Universität Siegen

Naturwissenschaftlich-Technische Fakultät

On the Robustness and Generalization of Deep Learning Approaches for Image Classification and Reconstruction

DISSERTATION
zur Erlangung des Grades eines
Doktors der Naturwissenschaften

vorgelegt von
Kanchana Vaishnavi Gandikota, M.S.

eingereicht bei der
Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen

November 2023

Betreuer und erster Gutachter
Prof. Dr. rer. nat. Michael Moeller
Universität Siegen

Zweite Gutachterin
Prof. Dr.-Ing. Margret Keuper
Universtät Mannheim

Tag der mündlichen Prüfung
24.01.2024

Declaration of Authorship

I Kanchana Vaishnavi Gandikota MS., declare that this thesis titled “On Robustness and Generalization of Deep Learning Approaches for Image Classification and Reconstruction” and the work presented in it are my own. I confirm that:

- The work in this doctoral thesis was wholly done during my candidature for the degree of Doctor of Philosophy in natural sciences at the University of Siegen.
- This thesis, in the same or similar form, was not used previously to achieve academic grading in any other degree nor published elsewhere.
- The information used in this thesis from other resources is clearly declared and attributed in the text and in the references.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

ABSTRACT

As deep learning models begin to be deployed in real-world applications, characterizing their vulnerabilities, and improving their robustness is critical to ensure reliable performance. This thesis deals with a few aspects of robustness and generalizability of deep learning models for image classification and reconstruction.

We first address the problem of robustness and invariance of neural networks to spatial transformations that can be represented as group actions. We propose a simple strategy to achieve provable invariance with respect to group actions by choosing a unique element from the orbit of transformation group. Such a simple orbit mapping can be used with any standard network architecture and still achieve desired invariance. We investigate the robustness with respect to image rotations, provable orientation and scaling invariance of 3D point cloud classification. We demonstrate the advantages of our method in comparison with different approaches which incorporate invariance via training or architecture in terms of robustness and computational efficiency.

Next, we investigate the robustness of classical and deep learning approaches to ill-posed image recovery problems, with a focus on image deblurring and computer tomography reconstruction. We demonstrate the susceptibility of reconstruction networks to untargeted, targeted and localized adversarial attacks using norm-constrained additive perturbations and study the transferability of attacks. We find that incorporating the model knowledge can, but does not always result in improved robustness. Further, localized attacks which modify semantic meaning can still maintain a high consistency with the original measurement, which could be used to deal with the ill-posedness of image recovery.

While deep neural networks are successful in many image recovery tasks, these networks are typically trained for specific forward measurement processes, and therefore do not typically generalize to even small changes in the forward model. To deal with this, we explore the use of generative model priors for flexible image reconstruction tasks. We develop a generative autoencoder for light fields conditioned on the central view, and utilize this model as a prior for light field recovery. We adopt the approach of optimizing in the latent space of the conditional generator to minimize data discrepency with the measurement, and perform simultaneous optimization of both the latent code and the central view when the latter is unavailable. We demonstrate the applicability of this approach for generic light field recovery.

Finally, we demonstrate the use of recently proposed text conditioned image diffusion models for generic image restoration and manipulation. We demonstrate flexible image manipulation by using a simple deterministic forward and reverse processes, with reverse diffusion being conditioned on target text. For consistent image restoration, we modify the reverse diffusion process of text-to-image diffusion model to analytically enforce data consistency of the solution, and explore diverse contents of null-space using text guidance. This results in diverse solutions which are simultaneously consistent with input text and the degraded inputs.

ZUSAMMENFASSUNG

Da Deep-Learning-Modelle zunehmend in praxisnahen Anwendungen eingesetzt werden, ist die Charakterisierung ihrer Schwachstellen und Verbesserung ihrer Robustheit unerlässlich, um eine zuverlässige Leistung zu gewährleisten. Diese Arbeit beschäftigt sich mit einigen Aspekten der Robustheit und Generalisierbarkeit von Deep-Learning-Modellen für Klassifikation und Rekonstruktion von Bildern.

Wir befassen uns zunächst mit dem Problem der Robustheit und Invarianz neuronaler Netze gegenüber räumlichen Transformationen, die als Gruppenaktionen dargestellt werden können. Wir schlagen eine einfache Strategie vor, um eine nachweisbare Invarianz in Bezug auf Gruppenaktionen zu erreichen, indem wir ein eindeutiges Element aus dem Orbit der Transformationsgruppe auswählen. Eine solche einfache Orbit-Mapping kann mit jeder Standardnetzwerkarchitektur verwendet werden und erreicht dennoch die gewünschte Invarianz. Wir untersuchen die Robustheit gegenüber Bildrotationen, sowie nachweisbare Orientierungs- und Skalierungsinvarianz bei 3D-Punktwolken-Klassifikation. Wir zeigen die Vorteile unserer Methode im Vergleich zu verschiedenen Ansätzen, die die Invarianz über das Training oder die Architektur einbeziehen, in Bezug auf Robustheit und Berechnungseffizienz.

Als Nächstes untersuchen wir die Robustheit von klassischen und Deep-Learning-Ansätzen bei schlecht gestellten Bildwiederherstellungsproblemen, wobei der Schwerpunkt auf Bildschärfung und Computertomographie-Rekonstruktion liegt. Wir zeigen die Anfälligkeit von Rekonstruktionsnetzwerken gegenüber ungezielten, gezielten und lokalisierten Angriffen mit additiven Störungen, deren Norm beschränkt ist, und untersuchen die Übertragbarkeit der Angriffe. Wir stellen fest, dass die Einbeziehung des Modellwissens in manchen Fällen zu einer verbesserten Robustheit führt. Außerdem können lokalisierte Angriffe, die die semantische Bedeutung verändern, immer noch eine hohe Konsistenz mit der ursprünglichen Messung aufrechterhalten. Dies könnte genutzt werden, um damit umzugehen, dass Bildwiederherstellung ein schlecht gestelltes Problem ist.

Tiefe neuronale Netze sind zwar bei vielen Bildwiederherstellungsaufgaben erfolgreich, aber diese Netze werden in der Regel für bestimmte Vorwärtsmessprozesse trainiert und lassen sich daher in der Regel nicht einmal auf kleine Änderungen im Vorwärtsmodell verallgemeinern. Um dies zu ändern, untersuchen wir die Verwendung generativer Model-Priors für flexible Bildrekonstruktionsaufgaben. Wir entwickeln einen generativen Autoencoder für Lichtfelder, der sich auf die zentrale Ansicht bezieht, und verwenden dieses Modell als Prior für die Lichtfeldwiederherstellung. Wir verfolgen den Ansatz, im Latent Space des conditional Generators zu optimieren, um die Diskrepanz zwischen den Daten und der Messung zu minimieren, und führen eine gleichzeitige Optimierung sowohl des Latent Codes als auch der zentralen Ansicht durch, wenn letztere nicht verfügbar ist. Wir demonstrieren die Anwendbarkeit dieses Ansatzes für eine generische Lichtfeldwiederherstellung.

Schließlich demonstrieren wir die Verwendung von kürzlich vorgeschlagenen, textgesteuerten Bilddiffusionsmodellen für die allgemeine Wiederherstellung und Manipulation von Bildern. Wir demonstrieren eine flexible Bildmanipulation durch Verwendung eines einfachen deterministischen Vorwärts- und Rückwärtsprozesses, wobei die Rückwärtsdiffusion durch den Zieltext gesteuert wird. Für eine konsistente Bildwiederherstellung modifizieren wir den umgekehrten Diffusionsprozess des Text-Bild-Diffusionsmodells, um die Datenkonsistenz der Lösung analytisch zu erzwingen, und untersuchen verschiedene Inhalte des Nullraums unter Verwendung von Textsteuerung. Dies führt zu verschiedenen Lösungen, die sowohl mit dem Eingabetext als auch mit den verschlechterten Eingabebildern konsistent sind.

ACKNOWLEDGMENTS

I thank everyone who supported and accompanied me on my journey towards PhD. I express my deep gratitude to my advisor Prof. Michael Moeller for providing me an opportunity to pursue my doctorate under his guidance. During the course of my doctoral studies, he always encouraged me, and shared valuable advice and expertise. I thank him for his time, advice and support. He was very supportive of me pursuing research directions that I found interesting and promising. I am grateful to have a wonderful and empathetic supervisor without whom this journey would not be possible.

I thank all my collaborators and co-authors Prof. Micheal Moeller, Jonas Geiping , Zorah Laehner, Hannah Droege, Paramanand Chandramouli, Prof. Margret Keuper, Prof. Andreas Kolb, Shashank Agnihotri, Julia Grabinski, Adam Czaplinski. It was a great experience working with all of you. I am grateful to Prof. Margret Keuper, who kindly agreed to be the co-examiner of this thesis. I thank Prof. Andreas Kolb for accepting me as a Masters student at the University of Siegen, which eventually paved a way for me to get an opportunity to pursue PhD. I thank Prof. Kristof Van Laerhoven and Zorah Laehner for their encouragement and support in my KI starter application.

I thank the anonymous reviewers of all my papers, whose feedback helped me to improve my writing, presentation and motivated me to conduct thorough experiments. This thesis is built on top of the work of many others before me. I thank all the members of the scientific community from the past, present and future, whose work inspired and continues to inspire me in my research.

I thank Sarah Wagener for her help in all important organizational things. I thank Sarah Wagener, Hannah Droege, Hartmut Baumeister, Zorah Lähner and Nadine Hoffmann for their support in dealing with many things in our life in Germany. I thank Hannah Droege, Hartmut Bauermesiter, Zorah Laehner, Marcel Seelbach, Sarah Wagener, Tejaswini Medi, Sameer Adavani for all the friendly conversations. I thank Elek Serif and DRK kita for their support. I thank Bhaskar Choubey and Mansi Trivedi for their personal support, friendship and advice. I am forever grateful to Dr. Vera Vaillant and Christiane Weimann for their support.

I thank my parents Ramesh Kumar and Syamala, my parents-in-law Chandramouli, and Vijayalakshmi for their encouragement throughout my PhD. Thank you my brother Rajeev Kashyap and sister-in-law, Rekha for your support, and for remotely keeping company during the pandemic. I am really grateful for your support.

I thank you Paramanand for research collaboration, which started rather reluctantly on your part. I thank you for your encouragement, and support through many tight deadlines and stressful times. I cannot thank you and Ananda enough for your love and support. Balancing the demands of research and caregiving required a lot of effort, especially during the pandemic. I very well know Ananda that there were many times when you rather wanted me to spend time with you, rather than have me working on some deadline. Thank you very much for your patience, understanding and encouragement. I could not have pursued this long journey without the support of both of you.

to Maa

CONTENTS

IV Conclusions

Part I

INTRODUCTION

1 INTRODUCTION

In the last decade, deep learning models have achieved remarkable breakthroughs in several domains such as computer vision, natural language processing, time series analysis, audio processing, reinforcement learning to name a few. With increasingly available massive training datasets, and the development of advanced network architectures and training methods, deep neural networks continue to improve performance on several benchmarks across domains. To give an example of the progress achieved, the performance of image recognition on the ImageNet benchmark (Russakovsky et al., 2015) has reached a state-of-the-art accuracy of over 91% in 2023 (Chen et al., 2023a) in comparison to 63% accuracy provided by Alexnet (Krizhevsky et al., 2012) which marked the beginning of the deep learning era in computer vision. Similar strong empirical performance is demonstrated by deep networks on other computer vision tasks such as segmentation (Wang et al., 2023a), object detection (Li et al., 2023), image generation (Sauer et al., 2022; Dhariwal and Nichol, 2021) and restoration (Sun et al., 2022) on benchmark datasets.

While performance on benchmark datasets is certainly important, machine learning models deployed in the real-world could encounter inputs which are from a significantly different distribution than those seen in training. Consider the example of image recognition, a model deployed in the real world could encounter data from different weather conditions, occlusions, image corruptions, or changes in imaging hardware. Unfortunately, even the state-of-the-art models can fail on such test data when such variations are not encountered during training (Hendrycks and Dietterich, 2019). In the case of image reconstruction, where the task is to recover an image from a measurement obtained through some forward process, one may encounter variations in noise statistics, or there may be errors in calibrating the forward model. As many supervised learning methods for image reconstruction are trained for a specific forward process, variations such as these would significantly affect their performance. This lack of desired robustness of learned models, despite their excellent performance on different benchmarks, is concerning. As deep networks are starting to be deployed in the real world, it becomes increasingly important to characterize the vulnerabilities of these models, and improve their robustness and generalizability. In this thesis, we study and address different aspects of robustness and generalizability for deep learning methods for image classification and reconstruction. We now briefly introduce each of these issues.

ADVERSARIAL ROBUSTNESS Deep networks are susceptible to *adversarial examples* where visually imperceptible perturbations to inputs result in catastrophic failures of the state of the art neural networks (Szegedy et al., 2014; Goodfellow et al., 2015). While most works focus on the robustness of image recognition networks to additive adversarial perturbations, some works also analyze adversarial robustness for other computer vision tasks such as object detection (Xie et al., 2017), semantic segmentation (Gu et al., 2022b; Agnihotri and Keuper, 2023), image reconstruction (Antun et al., 2020; Genzel et al., 2022). Investigating the stability of deep learning based image reconstruction is particularly interesting due to the following reasons: when the inverse image reconstruction is ill-posed, multiple valid solutions can exist. This ill-posedness also implies a trade-off between the stability of the recovery algorithm and the accuracy it can achieve in terms of proximity to ground truth. Further, there exist classical approaches for image reconstruction with convergence guarantees. In this thesis, we take a closer look at these issues and study the stability and adversarial robustness of different classical and deep learning approaches to image recovery, and propose approaches to handle ill-posedness.

ROBUSTNESS TO SPATIAL TRANSFORMATIONS Beyond robustness to tiny additive perturbations, certain properties such as rotational, scale, and shift invariance are often highly desirable in applications like image recognition. Yet, even after training deep networks with millions of realistic images, these properties are not guaranteed, and networks are still susceptible to adversarial attacks with respect to these transformations (see e.g. (Engstrom et al., 2017; Finlayson et al., 2019; Zhao et al., 2020b; Lang et al., 2021)). To counteract the lack of robustness, one solution is to modify the training procedure, either by augmentation using transformed examples or training with transformed examples which change the model prediction, i.e. *adversarial training*. In fact, such a strategy could be applied to any general set of transformations, for instance, additive perturbations within a norm ball of a certain radius, or deformations via translations, rotations, and other spatial transformations within a predefined deformation measure. While these training strategies do provide gains in robustness, they do not guarantee invariance. In contrast, for more structured transformations which can be characterized as a *group*, network architectures which guarantee provable invariance have been proposed (Cohen and Welling, 2016; Weiler et al., 2018a), yet, many works focus only on finite groups. In this thesis, we propose an approach for provable invariance to continuous group transformations, a more challenging setting which is rather less addressed in the literature.

ROBUSTNESS TO MEASUREMENT MODEL CHANGES Many supervised learning methods for image reconstruction are trained for a specific forward process and noise model and any changes in these can adversely affect their performance. In contrast, classical energy minimization approaches allow such modifications by appropriate changes to the energy function, yet their performance is inferior to fully learned methods. One can ask if it is possible to retain the flexibility of energy minimization methods while improving their reconstruction performance by learning. In this thesis, we address this problem by using learned generative priors, which are trained in an unsupervised fashion independent of a specific measurement model in an energy minimization framework.

1.1 ORGANIZATION OF THESIS

In this thesis, we study different aspects of robustness of deep neural networks with application to image classification and reconstruction, and propose methods to improve robustness and generalizability. This thesis assumes that the reader already has a basic understanding of machine learning and deep learning. Part II has three chapters discussing the background on adversarial robustness, image reconstruction, and generative models. In Chapter 2, we provide the reader with an overview of different adversarial attack and defense techniques. As a significant part of this thesis deals with different issues related to stability and generalization in image reconstruction, we give a broad overview of image reconstruction in Chapter 3, which walks the reader through different approaches to image recovery, both classical and deep learning based techniques, and a variety of methods using a combination of both. In Chapter 4, we provide a general overview of generative models, and discuss generative adversarial networks (GANs), generative autoencoders, and diffusion models. We further discuss about the use of generative priors in image recovery. The major contributions of the thesis are laid out in Chapters 5-9 included in the Methodology part III. These methodology chapters have their own related work sections with a survey of publications directly related to the chapter, and also include some preliminaries specific to the chapter when necessary. This thesis concludes with some discussions and future directions in Chapter 10.

1.2 CONTRIBUTIONS

This section outlines the research contributions made in this thesis, and lists the publications based on these contributions.

CHAPTER 5 looks at the problem of obtaining provable robustness or invariance to transformations which can be modeled as continuous group actions. Examples of such transformations include rotations in 2D and 3D, translations, and scaling. In contrast to most existing approaches which address this issue either by modifying the training procedure or designing specific network architectures, we propose a simpler alternative by transforming the input itself before feeding it into the network. We derive a principled approach based on group theory to perform this input transformation which guarantees provable invariance to the desired group action, with any network architecture. Further, we empirically analyze the properties of different approaches which incorporate invariance, and demonstrate the advantages of our method in terms of robustness and computational efficiency. In particular, we investigate the robustness of classification with respect to image rotations (which can hold up to discretization artifacts) as well as the provable orientation and scaling invariance of 3D point cloud classification.

CHAPTER 6 studies the stability and adversarial robustness of different classical and deep learning approaches to image recovery, with a focus on image deblurring and computer tomography reconstruction. In contrast to prior works which mainly focus on untargeted attacks on image reconstruction, we also demonstrate susceptibility of image recovery networks to targeted and localized attacks. We show that localized attacks can be used beneficially to handle the ill-posedness of image recovery by allowing exploration of solution space with high data consistency.

CHAPTER 7 addresses the problem of generalizing deep learning based image reconstruction for different measurement models. In this chapter, we use deep generative models, specifically generative auto-encoders as priors in model based image reconstruction. As these models are trained without the supervision of specific measurement models, they can be incorporated as a prior into model-based optimization, and therefore extend to diverse reconstruction tasks. In contrast, the applicability of end-to-end networks trained in a supervised way is limited to the specific measurement model they have been trained on. We demonstrate the utility of generative autoencoder priors for light field recovery from diverse measurement models. We propose and train the first generative model, a conditional

Wassertein auto-encoder (CWAE) for 4D light field patches conditioned on the central view of the light field, and utilize this model as a prior in light field reconstruction. We take the approach of optimizing in the latent space of the CWAE generator to minimize data discrepancy with the measurement, and perform simultaneous optimization of both the latent code and the central view when the latter is unavailable. We perform diverse light field recovery tasks including light field view synthesis, spatial-angular super resolution, and reconstruction from coded projections. We demonstrate the advantages of our approach in comparison with end-to-end trained networks in terms of flexibility and robustness to corruptions and improved performance with respect to traditional model-based approaches on both synthetic and real scenes.

CHAPTER 8 addresses the problem of generalized open domain image manipulation through simple and intuitive text prompts. This chapter continues on the theme of the preceding chapter in using generative model priors, but the focus is on leveraging text-to-image diffusion based generative models for diverse image manipulation tasks, without explicit training or fine-tuning. We leverage a pretrained text-to-image generative model known as latent diffusion model, where text guided diffusion is performed in the latent space of a variational auto-encoder. We employ a deterministic forward diffusion in a lower dimensional latent space, and the desired manipulation is achieved by simply providing the target text to condition the reverse diffusion process. This ensures consistency with the input image while modifying the desired attributes. We demonstrate that this method can accomplish diverse image manipulation tasks, with advantages in terms of flexibility and computation times over competing baselines.

CHAPTER 9 introduces the problem of exploring open domain image restoration through text prompts. This chapter continues on the theme of the preceding chapters 7 and 8 in using generative model priors. This time, the focus is on exploring solutions to image restoration problems using text, in a zero-shot fashion without explicit training for this task. There are no existing works which attempt this problem. We develop an approach to utilize a pretrained text-to-image diffusion model where text conditioned diffusion is performed in a down-sampled pixel space, and modify its reverse diffusion process to analytically enforce data consistency of the solutions. This approach can recover diverse solutions that match the semantic meaning provided by the text prompt, while preserving data consistency with the degraded inputs. In contrast, most prior works for image recovery, even those utilizing generative model priors do not provide a mechanism to control the solutions, and tend to exhibit limited diversity in their solutions.

1.2.1 *Publications in the Thesis*

The research presented in this thesis is based on the following jointly authored papers. In the following * indicates equal contribution. At the beginning of each chapter, my contributions and the contributions of my collaborators are clearly specified.

1. Chapter 5: **K. V. Gandikota**, J. Geiping, Z. Lähner, A. Czapliński, Michael Moeller " A Simple Strategy to Provable Invariance via Orbit Mapping," *Proc. Asian Conference on Computer Vision (ACCV)*, 2022

2. Chapter 6:

   (a) **K. V. Gandikota**, P. Chandramouli, M. Moeller "On Adversarial Robustness of Deep Image Deblurring," *Proc. IEEE International Conference on Image Processing (ICIP)*, 2022.

   (b) **K. V. Gandikota**, P. Chandramouli, H. Droege, M. Moeller "Evaluating Adversarial Robustness of Low dose CT Recovery," *Proc. Medical Imaging with Deep Learning (MIDL)*, 2023

3. Chapter 7: P. Chandramouli*, **K. V. Gandikota***, A. Goerlitz, A. Kolb, M. Moeller "Generative models for generic light field reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* April 2022.

4. Chapter 8: P. Chandramouli, **K. V. Gandikota** "LDEdit: Towards Generalized Text Guided Image Manipulation via Latent Diffusion Models," *Proc. British Machine Vision Conference (BMVC)*, 2022.

5. Chapter 9: **K. V. Gandikota***, P. Chandramouli*, "Exploring Open Domain Image Super-Resolution through Text", in *ICML Workshop on Artificial Intelligence & Human-Computer Interaction*, 2023.

1.2.2 *Other Publications*

In addition, I also jointly co-authored the following publications which are not part of this thesis, during the course of my PhD. In these publications, I partially contributed to the ideas, design of experiments, and technical presentation. However, I was not involved in the actual implementation of experiments and generating results presented therein.

- P. Chandramouli, **K. V. Gandikota** "Blind single image reflection suppression for face images using deep generative priors," *Proc. IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2019.

- G. Hegde*, A. N. Ramesh*, **K. V. Gandikota***, R. Obermaisser, M. Moeller "A Simple Domain Shifting Network for Generating Low Quality Images" *Proc. 25th International Conference on Pattern Recognition (ICPR)*, 2020

- S. Agnihotri, **K. V. Gandikota**, J. Grabinski, P. Chandramouli, M. Keuper "On the unreasonable vulnerability of transformers for image restoration and an easy fix". *Proc. IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2023.

Chandramouli & Gandikota [2020] deals with reflection removal for facial photographs taken through transparent surfaces. This is a challenging and ill-posed problem due to the inherent ambiguity in separating the averaged colors and textures and assigning them to one of the two layers. To reduce ambiguity in single image reflection suppression, we utilize a deep generative autoencoder prior trained on facial images for the background layer, and develop an optimization scheme to recover reflection-free facial images using both the encoder and the decoder in model based optimization. Such an optimization would alleviate the problem of representation error between the actual image and its nearest neighbor in the range space of the generator. In contrast, using a latent space optimization would result in a higher representation error, which is exacerbated when the face image to be recovered is out of distribution to the training set. The proposed approach compares favorably to recent deep network based reflection separation approaches when evaluated on synthetic and real world facial images containing reflections.

Hegde et al. [2020], deals with the problem of classifying low quality images captured by robot camera without any low quality labeled training data specific to the classification task. To deal with this problem, we propose to train a regression network to translate high quality images to mimic images captured by a specific low quality camera, using unlabeled image pairs. This domain-shifting network can then be used to generate low quality images for novel classes. We train a classifier to learn an invariant representation across the source domain (high quality images) and the target domain (low quality images), by training on data from both the domains. We demonstrate the utility of this approach for zero-shot and unsupervised domain adaptation for low quality image recognition.

Agnihotri et al. [2023] also deals with adversarial robustness of deep learning based image deblurring similar to Chapter 4, yet the focus is on Transformer based architectures. In this work, we analyze adversarial robustness of a recent transformer based network (Zamir et al., 2022) and network architectures derived in (Chen et al., 2022a) by reducing the complexity of the network proposed by (Zamir et al., 2022), with modifications to attention mechanism, and activation functions. This study is particularly interesting in the light of recent works (Xie et al., 2020; Bai et al., 2021) which demonstrate the importance of the choice of activation function in boosting adversarial robustness. In tune with our observations in Chapter 4, we find that restoration networks, even Transformer based ones, are not inherently robust when trained using standard training protocols. We show that simple adversarial training using single-step attacks can significantly improve robustness, and uncover interesting effects due to the interplay of different attention mechanisms and nonlinearities on adversarially robust generalization.

Part II

BACKGROUND

2 ADVERSARIAL ROBUSTNESS

We consider a neural network $\mathscr{G}$ to be a function $\mathscr{G} : \mathscr{X} \times \mathbb{R}^p \rightarrow \mathscr{Y}$ that maps data $x \in \mathscr{X}$ from some suitable input space $\mathscr{X}$ to some prediction $\mathscr{G}(x;\theta) \in \mathscr{Y}$ in an output space $\mathscr{Y}$, where the way this mapping is performed depends on parameters θ. Assuming that the training and test data are identically distributed, the standard training of networks is based on principles of empirical risk minimization (ERM)

$$\min_{\theta} \sum_{\text{examples i}} \mathscr{L}\left(\mathscr{G}\left(x^i;\theta\right);y^i\right). \tag{2.1}$$

The parameters θ of network $\mathscr{G}$ are typically updated via gradient descent based steps using back-propagation (Rumelhart et al., 1986) to minimize the empirical risk (average loss) $\mathscr{L}$ on all the data samples (x^i, y^i) across the entire training set. Neural networks trained in this manner have achieved impressive performance across several computer vision benchmarks. However, when the assumption that the train and test data belong to the same underlying distribution is violated, the trained models fail. This is a very concerning issue that needs to be solved to deploy machine learning systems in safety critical applications.

2.1 ADVERSARIAL ATTACKS

An extreme version of fragility of the deep neural networks is demonstrated through adversarial examples. Adversarial examples are a result of imperceptible changes to inputs that are almost indistinguishable to human eye. Yet, these changes result in catastrophic failures of the state of the art neural networks (Szegedy et al., 2014; Goodfellow et al., 2015). A popular illustration of adversarial attacks is depicted in Fig. 2.1, where the addition of a tiny perturbation results in a trained classifier network misclassifying a panda as a gibbon. To a human observer, both the original image and the adversarial image look identical. Adversarial attacks are further categorized into “white box”, “black box” and “gray box” attacks. White box attacks are the strongest as the adversary has complete access to model parameters and thus can optimize for the adversarial examples accordingly.

$$\max_{\tilde{x} \in S_\delta(x)} \mathscr{L}\left(\mathscr{G}\left(\tilde{x};\theta\right);y\right), \tag{2.2}$$

where $S_\delta(x)$ represents the set of allowable examples that are similar to the original example x according to some measure of distortion. In contrast, in a black box attack, the adversary can only query the model on inputs, and update the adversarial perturbation based on the output labels or confidence scores. In the case of gray box attacks, the adversary has some access to the network setup, for instance, the architecture and the training procedure of the model may be known, but not the exact model parameters.

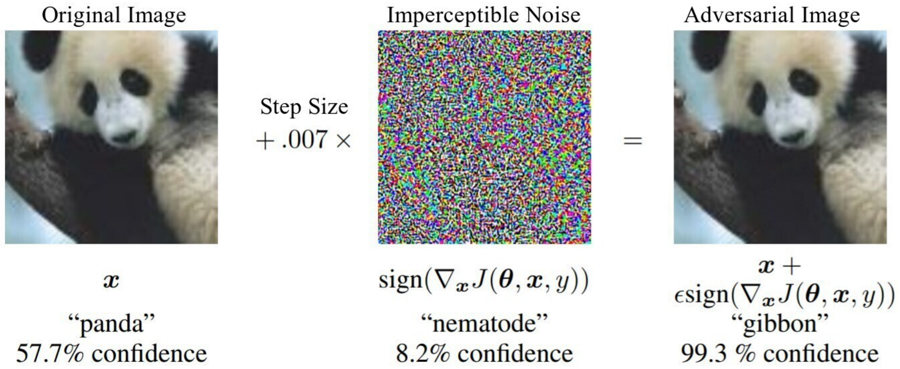

Original Image
Imperceptible Noise
Adversarial Image
Step Size
+ .007 ×
=
$\boldsymbol{x}$
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"panda"
57.7% confidence
"nematode"
8.2% confidence
"gibbon"
99.3 % confidence

Figure 2.1: Illustration of adversarial attack from Goodfellow et al. (2015) which causes a state of the art deep network classifier to predict an adversarially perturbed image of a panda as a gibbon with high confidence.

2.1.1 *Additive Adversarial Perturbations*

As image classification is the most extensively studied task in the context of adversarial robustness, in the following, we describe adversarial attacks for image classification. Similar attacks have also been extended to networks trained for other computer vision tasks such as semantic segmentation (Arnab et al., 2018; Agnihotri and Keuper, 2023; Rony et al., 2023), object detection (Fischer et al., 2017; Xie et al., 2017), image reconstruction (Antun et al., 2020; Choi et al., 2019). Among the different types of adversarial attacks, white box attacks with norm-constrained additive perturbations (Goodfellow et al., 2015; Madry et al., 2018) are the most widely considered attacks in the adversarial robustness literature. Here, the objective is to change the network prediction subject to some ℓ_p norm constraint on the additive perturbation. The attacks can further be categorized as *targeted* and *untargeted* attacks. When the objective is to only misclassify the input image, regardless of which wrong label is predicted, it becomes an untargeted attack:

$$\delta_{adv} = \arg\max_{\delta} \mathscr{L}\left(\mathscr{G}\left(x+\delta;\theta\right), y\right) \text{ s.t. } \|\delta\|_p \leq \epsilon. \tag{2.3}$$

When the goal is to find an additive image perturbation that makes the network under attack predict a specific target label $\tilde{y}$, it becomes a targeted attack:

$$\delta_{adv} = \arg\min_{\delta} \mathscr{L}\left(\mathscr{G}\left(x+\delta;\theta\right), \tilde{y}\right) \text{ s.t. } \|\delta\|_p \leq \epsilon. \tag{2.4}$$

where $\mathscr{L}$ is usually the cross entropy loss for attacks on image classification networks. Due to the non-convexity of neural network loss landscape, exactly solving the optimization problems eq. (2.3),eq. (2.4) is hard. For small neural networks, these can be solved exactly using mixed integer linear programming, which allows multiple additional constraints to be exactly imposed (Fischetti and Jo, 2018). However, this approach is computationally intractable for large deep neural networks used in practice. Common practical implementation of adversarial attacks employs approximate solutions using a limited number of gradient ascent based steps. Different attack algorithms have been proposed by varying the optimization algorithm used. The simplest of such attacks is the fast gradient sign method (FGSM) (Goodfellow et al., 2015) which uses a single projected gradient ascent step to obtain ℓ_∞ norm constrained adversarial examples. Untargeted adversarial perturbations with a perturbation budget of ϵ are generated using the FGSM attack as:

$$\delta_{adv} = \epsilon \cdot \text{sign}\left(\nabla_{\delta}\mathscr{L}\left(\mathscr{G}\left(x+\delta;\theta\right), y\right)\right) \tag{2.5}$$

Instead of a single step, basic iterative method (BIM) (Kurakin et al., 2018), iteratively applies FGSM multiple times with step size α, and clipping the resulting adversarial noise according to the perturbation budget at each iteration:

$$\delta^{t+1} = \text{clip}\left(\delta^{t} + \alpha \cdot \text{sign}\left(\nabla_{\delta^t}\mathscr{L}\left(\mathscr{G}(x+\delta^t;\theta), y\right)\right), [-\epsilon, \epsilon]\right) \tag{2.6}$$

This results in a stronger attack than single-step FGSM. Due to the highly non-convex nature of neural network loss-landscape, such signed gradient ascent based methods have demonstrated more success in finding adversarial perturbations (Bernstein et al., 2018) in comparison to the classical gradient ascent methods which were shown to be less effective (Athalye et al., 2018). The signed gradient step, in fact, corresponds to the normalized steepest ascent steps of the form

$$\delta^{t+1} = \delta^{t} + \underset{\|v\|_{\infty}\leq\alpha}{\arg\max}\; v^{T}\nabla_{\delta^t} f(\delta^t), \tag{2.7}$$

whose solution is given as

$$\delta^{t+1} = \delta^{t} + \alpha \cdot \text{sign}\left(\nabla_{\delta^t} f\left(\delta^t\right)\right), \tag{2.8}$$

where we denoted $\mathscr{L}\left(\mathscr{G}(x+\delta^t;\theta), y\right)$ as $f(\delta^t)$ for convenience. Clipping δ to lie in the range $[-\epsilon, \epsilon]$ corresponds to projection onto an ℓ_∞ norm ball of radius ϵ. This attack can be generalized to any ℓ_p norm constraint on the perturbation by projection $\mathscr{P}$ onto the corresponding norm ball of radius ϵ. For ℓ_2 norm constraints, this becomes:

$$\delta^{t+1} = \underset{\|\cdot\|_2\leq\epsilon}{\mathscr{P}}\left(\delta^{t} + \alpha \cdot \text{sign}\left(\nabla_{\delta^t} f\left(\delta^t\right)\right)\right), \text{ with } \underset{\|\cdot\|_2\leq\epsilon}{\mathscr{P}}(z) = \epsilon\frac{z}{\max\{\epsilon, \|z\|_2\}}. \tag{2.9}$$

The adversarial examples $(x+\delta_{adv})$ are subsequently clipped to lie in the range of valid intensities, which for image data corresponds to $[0,1]$. To take into account the non-convexity of the neural network loss landscape, the authors in Madry et al. (2018) propose to run the attack with multiple random restarts within the ℓ_p ball of choice to effectively explore the input space. With a slight abuse of nomenclature, the attack is referred to as projected gradient descent (PGD) attack. Due to the easy analytical solution for the projection, ℓ_p norm-constrained additive adversarial perturbations are the most commonly used attacks. Chapter 6 of this thesis is concerned with evaluating the robustness of deep neural networks for image reconstruction, where we consider ℓ_p norm-constrained additive perturbations. Apart from constraining the norm of perturbation, a few works (Carlini and Wagner, 2017b; Brendel et al., 2019; Rony et al., 2019) have also considered additive perturbations with the minimum norm. Carlini and Wagner (2017b) employ an augmented Lagrangian formulation resulting in a trade-off between misclassification loss and perturbation norm. Brendel et al. (2019) attack stays close to the decision boundary using the gradient for local linear approximation while minimizing the norm. Rony et al. (2019) iteratively reduce the perturbation budget of projected-gradient attacks at each step. We do not discuss these attacks (Carlini and Wagner,

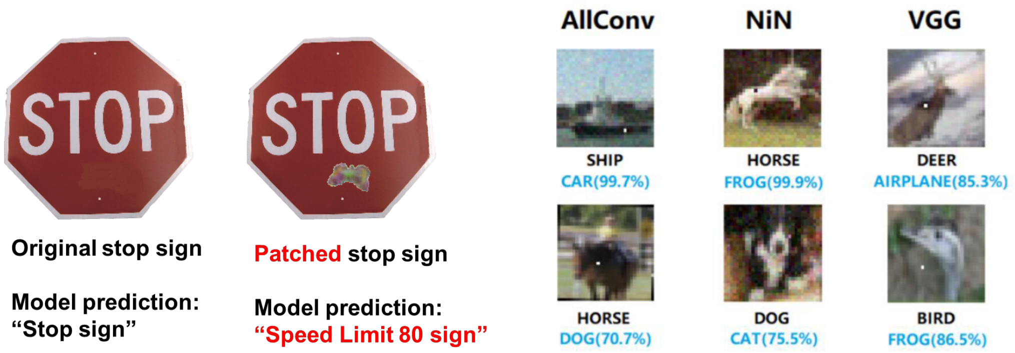


Figure 2.2: Illustration of localized adversarial attacks. Shown on the left is a patch attack taken from Yakura et al. (2020) due to an adversarial 'bug' on the stop sign. Shown on the right is an adversarial attack caused by modifying a single pixel, image taken from Su et al. (2017).

2017b; Brendel et al., 2019; Rony et al., 2019) in more detail, as these are beyond the scope of attacks used in this thesis.

2.1.2 *Beyond Additive Perturbations*

While additive perturbations with ℓ_p norm constraints are the most popular type of adversarial attacks, these attacks tend to modify every pixel in the image. However, the ℓ_p norm metric is not the only way to measure image similarity. A small translation or rotation of an image results in a transformed image with a high similarity with the original. Yet, such a transformation can induce a large change in the ℓ_p norm metric, which only measures pixel-wise similarity. A number of adversarial attacks have been proposed which impose constraints on alternate distortion measures between the original and perturbed images to obtain realistic adversarial samples. For instance, Wong et al. (2019); Wu et al. (2020a) propose to constrain the perturbation budget in terms of Wasserstein distance between the clean and adversarial samples. A few approaches have considered localized perturbations such as adversarial patches (Brown et al., 2017) or even modifying a single pixel in an image to change network prediction (Su et al., 2017). Some of these localized adversarial attacks are also physically realizable (Kurakin et al., 2018), for example, glasses to fool facial recognition (Sharif et al., 2016), stickers based on adversarial patches to attack traffic sign recognition systems (Eykholt et al., 2018), adversarial prints to fool object recognition systems (Wu et al., 2020b). Fig. 2.2 illustrates such localized attacks on deep learning based image recognition systems. In this thesis, we also develop localized adversarial attacks in the context of image reconstruction where an additive adversarial perturbation leads to a desired localized change in the reconstructed image.

Another highly concerning phenomenon is the susceptibility of neural networks to simple geometric transformations of images, which can occur due to natural factors such as viewpoint changes. Examples include vulnerabilities to small translations and rotations (Engstrom et al., 2017; Finlayson et al., 2019) or spatial deformations (Xiao et al., 2018), viewpoint changes (Dong et al., 2022). Fig. 2.3 depicts instances of such spatial adversarial examples. We can observe that trained classifiers can be misled by an adversarially optimized flow field, or even by translations and rotations. While transformations such as large rotations are rarely observed in natural images (such as an upside down tree), full rotational invariance is desirable in certain critical applications such as medical image classification, where images

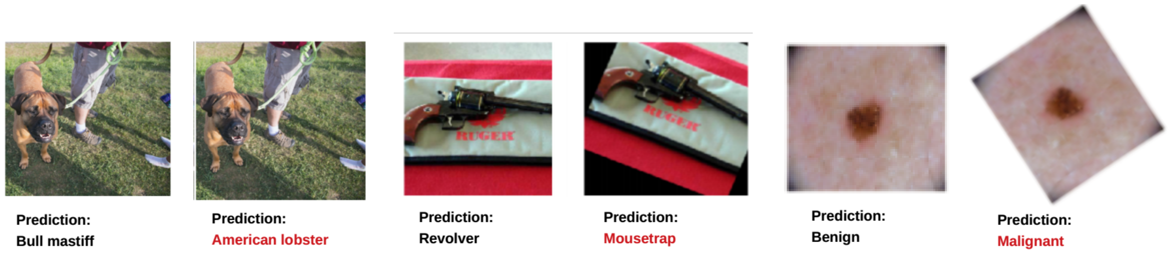


Figure 2.3: Illustration of geometric adversarial attacks constructed by i) adversarial spatial deformation field (left), taken from Xiao et al. (2018), ii) adversarial translation and rotation (middle), taken from Engstrom et al. (2019) and iii) adversarial rotation (right), taken from Finlayson et al. (2019). Adversarial predictions are denoted in red.

do not have an inherent orientation. For instance, a change in viewing angle must not change the classification of a lesion from that of benign to malignant or vice-versa. In this thesis, we develop approaches to obtain robustness and invariance against transformations such as rotations and translations which can be modeled as group actions.

2.2 ADVERSARIAL DEFENSE

For safe deployment of machine learning models in the real world, they need to be robust to small variations in test inputs. Several approaches have been proposed to improve the robustness of models to adversarial attacks. On the other hand, stronger adversarial attacks are proposed which with the knowledge of the defense mechanism, can bypass many defenses. In the following, we attempt to provide an overview of adversarial defense techniques.

2.2.1 *Preprocessing Techniques*

Simple preprocessing techniques have been used to defend models against adversarial attacks. These include the use of JPEG compression (Das et al., 2018), random resizing and padding (Xie et al., 2018), total variation minimization, bit depth reduction, image quilting (Guo et al., 2018). However, transformation based defenses against additive adversarial perturbations can be easily overcome by an adversary with the knowledge of the transformation being used. Even defenses employing non-differentiable transformations could be bypassed by using differentiable approximations (Athalye et al., 2018). While earlier works suggested that the use of randomized transformations is difficult to circumvent for an adversary (Raff et al., 2019), more recent work (Gao et al., 2022) shows that even the effectiveness of stochastic defenses can be reduced. Some defense mechanisms also use multiple transformed versions of input, for instance, randomized smoothing (Cohen et al., 2019) which makes predictions by feeding multiple copies of the input with additive Gaussian corruptions to the classifier being defended. Transformation based defenses have also been proposed for attacks other than additive perturbations, for example, using multiple randomized crops to defend against patch attacks (Lin et al., 2021) or pooling the features of multiple rotated versions of an image to improve robustness to rotations (Manay et al., 2006; Laptev et al., 2016). Some defense approaches propose to *purify* inputs by removing adversarial perturbations to reconstruct the clean samples. These include methods that reconstruct images using sparse representation techniques (Sun et al., 2019; Lu et al., 2022), or by projecting inputs onto the data manifold of trained generative models such as generative adversarial networks (GANs) (Samangouei et al., 2018), or more recently using diffusion-based generative models (Nie et al., 2022) by adding randomly sampled noise, followed by iterative denoising using a trained diffusion model. In chapter 5 of this thesis, we develop a technique that theoretically guarantees invariance of models to transformations such as rotations which form a group. This technique can be considered as a preprocessing step that undoes the transformation by *consistently* orienting an input before feeding the data into the actual network.

2.2.2 *Robustness by Architecture*

An alternate approach to improve robustness is through the use of robust network architectures. Different works study the influence of architecture on adversarial robustness at different levels of granularity ranging from choice of basic architecture itself, for instance, CNNs versus vision transformers (Shao et al., 2022; Bai et al., 2021), to specific blocks within a class of architectures, for example, residual blocks in (Huang et al., 2023), down to modifying specific components such as pooling (Grabinski et al., 2022) or skip connections (Li et al., 2020c) to improve adversarial robustness. While Shao et al. (2022) suggest relatively higher robustness of vision transformers in comparison to CNNs, Bai et al. (2021) demonstrates a similar degree of adversarial robustness for both under identical training protocols. A closer investigation by Croce and Hein (2022) on the effect of components such as patches, convolution, and attention revealed small architectural modifications can significantly improve robustness. Further, some recent works also propose to search for neural architectures (Huang et al., 2021; Guo et al., 2020) to obtain adversarially robust architectures. While all these methods mainly focus on additive adversarial perturbations, provable robustness to certain transformations is easier achieved via network architectures which include invariances to these transformations by design. Examples include equivariant/invariant networks for rotations (Oyallon and Mallat, 2015; Cohen and Welling, 2016; Worrall et al., 2017), and shifts (Sifre and Mallat, 2013; Rojas-Gomez et al., 2022).

2.2.3 *Adversarial Training*

Perhaps one of the most effective defenses to counteract the threat of adversarial examples is adversarial training. Instead of the standard ERM training, adversarial training employs the principles of robust optimization (Wald, 1945; Ben-Tal et al., 2009) to optimize for the maximum risk (loss) to confer better robustness properties. Instead of training directly on data samples, training is performed on corresponding adversarially generated samples (Gu and Rigazio, 2014; Goodfellow et al., 2015; Madry et al., 2018), which yields models robust to adversarial perturbations. For norm-bounded additive adversarial perturbations, adversarial training involves the following:

$$\min_{\theta} \sum_{\text{examples i}} \max_{\delta^i : \|\delta^i\| \leq \epsilon} \mathscr{L}\left(\mathscr{G}(x^i + \delta^i; \theta); y^i\right) \tag{2.10}$$

In principle, any of the methods introduced in section 2.1.1 may be used for obtaining solutions to the inner maximization, and these adversarial examples are in turn used for training the network (the outer minimization problem). While adversarial training using single-step attacks such as FGSM is less expensive (Goodfellow et al., 2015), models trained using FGSM adversarial training are still susceptible to iterative attacks (Kurakin et al., 2018; Tramèr et al., 2018). Further, adversarial training using FGSM examples could be affected by catastrophic over-fitting (Wong et al., 2020; Kim et al., 2021). Several recent works (Wong et al., 2020; Andriushchenko and Flammarion, 2020; Zhang et al., 2022; de Jorge Aranda et al., 2022) attempt to improve the effectiveness of single-step adversarial training, which remains an active area of research. In contrast, adversarial training using PGD generated adversarial examples results in a model that is robust to norm-bounded attacks (Madry et al., 2018). Further, models trained against PGD attacks have also been shown to be robust against other attacks (Zheng et al., 2019). However, adversarial training using PGD (with random restarts) is very expensive.

On the other hand, robustness gains through adversarial training are accrued at the cost of reduced accuracy on clean samples (Tsipras et al., 2019), which is undesirable. This can be partially mitigated by using both natural and adversarial examples during training. Alternately, Miyato et al. (2018); Zhang et al. (2019a) proposed a modified training scheme to minimize the difference in network output on clean and adversarial data, which mitigates this trade-off to some extent. Schmidt et al. (2018) suggested that adversarially robust generalization requires more data than for standard learning. Several follow-up works demonstrated improvements in robust generalization using additional data in adversarial training and regularization. Examples include the use of unlabeled examples (Carmon et al., 2019; Alayrac et al., 2019), data samples synthesized via data augmentation (Rebuffi et al., 2021; Gowal et al., 2021) or using generative models (Wang et al., 2023c). Yet, the achieved adversarial robustness is far from satisfactory. Even after all the advancements, the current state of the art[1] robust classification accuracy with perturbation budget of 8/255 in ℓ_∞ norm for CIFAR-100 is only 42.67%, even after using an additional 50 million synthetic images during training.

While we mainly discussed adversarial training against norm-constrained additive perturbations, similar defense mechanisms were also explored for alternate attack paradigms, including adversarial training against patch attacks (Rao et al., 2020) and spatial transformations (Engstrom et al., 2019; Yang et al., 2019). In the case of the latter, architectures that include a provable invariance to specific spatial transformations by design outperform adversarial training.

2.2.4 *Further Defense Mechanisms*

In the following, we briefly discuss defense mechanisms not encountered in this thesis, for the sake of providing an overall overview:

ADVERSARIAL EXAMPLE DETECTION Adversarial defense via detection works by rejecting inputs that are classified as adversarial by a detector (Metzen et al., 2017; Xu et al., 2017). The detection techniques can further be categorized as supervised and unsupervised detection methods, based on the knowledge of the attack mechanism or the lack thereof. Supervised methods train a detector to distinguish adversarial examples by training on both natural and adversarial examples, for instance, using features extracted from the defended network (Metzen et al., 2017; Carrara et al., 2018; Lu et al., 2017), or their statistics (Feinman

[1] `https://robustbench.github.io` accessed on April 21, 2023.

et al., 2017; Grosse et al., 2017; Roth et al., 2019). Unsupervised methods rely only on natural samples, for example, by utilizing the inconsistencies between transformed versions of inputs (Meng and Chen, 2017; Xu et al., 2017). Recent works (Carlini and Wagner, 2017a; Bryniarski et al., 2022), however, show that most state-of-the-art detection-based defenses could be bypassed by detection-aware attacks which can simultaneously fool the original classifier and the detector.

CERTIFIED ROBUSTNESS In contrast to the empirical defense mechanisms discussed so far, certified defense mechanisms attempt to produce a certificate that no attack can induce the network to produce outputs with error beyond a certain value. Training networks to have a verifiably low robust error would require computing worst-case error exactly at each training step, which is intractable beyond small networks. To deal with this intractability, (Wong and Kolter, 2018; Raghunathan et al., 2018) employ an alternate approach, which involves computing an upper bound for the worst-case loss to certify a network's robustness against all attacks for a given input. While this provides a certifiable bound, robust training networks using the upper bounds are still outperformed by adversarial training in empirical robustness evaluations.

3 IMAGE RECONSTRUCTION: FROM MODEL BASED APPROACHES TO NEURAL NETWORKS

The goal of image reconstruction is to recover an unknown image from indirect or distorted measurements. Let f, A, u and n represent the measurement, forward operator, ground truth image and measurement noise respectively, the measurement process can be described as,

$$f = A(u) + n. \tag{3.1}$$

When the forward measurement process is linear, eq. (3.1) can be expressed as

$$f = Au + n, \tag{3.2}$$

and recovering u becomes a linear inverse problem, which is what we focus on in this thesis. Depending on the operator A, we have different reconstruction problems, some examples are uniform deblurring, where A corresponds to a convolution with a blur kernel, super-resolution, where A is modeled as blur followed by downsampling, for computed tomography reconstruction, where A is given by the 2D Radon transform (Radon, 1986), compressive sensing, where A corresponds to an acquisition mask. In many image recovery problems encountered in practice, the measurements are incomplete, which causes the recovery problem to be ill-posed. Sometimes, the measurement operator is also unknown, in which case, it becomes a blind reconstruction problem. In the following, we discuss different approaches to recover the image $\hat{u}$ from the measurement f.

3.1 CLASSICAL APPROACHES

Classical approaches explicitly take into account the known forward model in the solution criteria. One of the earliest approaches to eq. (3.2) is the solution to the least squares problem $\underset{u}{\arg\min} \|Au - f\|^2$, given by $\hat{u} = A^\dagger f$, where $A^\dagger$ is the pseudo-inverse. This is the minimum norm solution to eq. (3.2) with no components in the null space of A, and satisfies perfect data consistency when there is no measurement noise. This however can overfit to measurement noise, and become very unstable in case of an ill-conditioned pseudo-inverse. Pseudo-inverse solutions can be stabilized by regularization or inexact computation of the pseudo-inverse, for example using reduced iterations of conjugate gradient (Hestenes et al., 1952). Further, even in the noise-less case, any solution of form $(A^\dagger f + u_\Delta)$ is also valid as long as u_Δ lies in the non-trivial null space of A. The following questions then arise: How to model the properties of meaningful/physically plausible solutions (prior knowledge)? How to recover solutions that are meaningful from the set of all solutions that are consistent with the measurements? There are two different classical paradigms to image recovery that address these questions - the *variational approach* and the *Bayesian approach*. Variational approach selects an optimization criterion, and incorporates prior knowledge through *regularization*. In the Bayesian approach, all unknowns are treated as stochastic quantities, and prior knowledge is incorporated into the problem formulation through the probabilistic models used to model the unknown quantities.

3.1.1 *Variational Methods for Image Recovery*

Variational approaches (Benning and Burger, 2018) find a minimizer of the energy function, with a term penalizing data discrepency and a regularization term to model prior knowledge about the latent image

$$\hat{u} = \arg\min_{u} E(A, u, f) + R(u). \tag{3.3}$$

Commonly used forms of the data discrepency term $E(A, u, f)$ include squared error $\frac{1}{2}\|Au - f\|_2^2$ or absolute error $\|Au - f\|_1$. The term $R(u)$ corresponds to regularization which encourages certain desired properties in a solution.

REGULARIZATION One of the earliest methods for regularizing ill-conditioned problems is Tikhonov regularization (Tikhonov, 1963), with $R(u)$ of the form $\|\Lambda u\|_2^2$. When Λ is a scaled identity matrix, this becomes ℓ_2 regularization on the latent image. The most commonly encountered priors in the literature are however sparsity and low-rank promoting priors in some transform domain (Mairal et al., 2014; Donoho and Elad, 2003), which encode the prior knowledge that clean noiseless images typically admit sparse representations. One example is the widely studied total-variation seminorm (TV) regularization $\|\nabla u\|_{2,1}$ (Rudin et al., 1992) which encourages sparsity in gradient domain, and was successfully applied in several image recovery tasks including non-blind deconvolution (Getreuer, 2012; Bioucas-Dias et al., 2006), CT reconstruction (Sidky et al., 2006; Chen et al., 2013), blind deconvolution (Perrone and Favaro, 2014). Further, sparsity based regularization have been proposed using ℓ_1 and nuclear norms based functions in transform domains such as wavelets (Figueiredo et al., 2007), curvelets (Starck et al., 2002), gradients (Xu et al., 2013), higher order gradients (Lefkimmiatis et al., 2013), or coefficients of over complete dictionary (Bruckstein et al., 2009; Elad and Aharon, 2006). These regularizers are grounded in theory (Donoho, 2006; Elad, 2010; Chambolle et al., 2010) and could achieve competitive results in many image recovery tasks. Apart from sparsity priors, other types of priors have also been proposed, for example, priors relying on local self-similarity (Freedman and Fattal, 2011), internal patch recurrence in images (Zontak and Irani, 2011; Michaeli and Irani, 2014). The choice of prior determines the structural properties of the solution (Rott Shaham and Michaeli, 2016), for instance, TV regularization encourages piece-wise constant solutions, whereas self-similarity prior promotes images with similar-looking structures at different scales. Further, the choice of regularizing prior also determines whether certain convergence guarantees can be provided for the variational approach, depending on the theoretical constraints satisfied by the regularizer.

SOLVING THE VARIATIONAL MINIMIZATION PROBLEM One simple way to solve the minimization problem eq. (3.3) would be using gradient descent with a suitably selected step size. When either the data consistency term $E(A, u, f)$ or the regularizer $R(u)$ is non-convex w.r.t u, using gradient descent can not guarantee convergence to the global optimum. On the other hand, when both $E(A, u, f)$ and $R(u)$ are convex, suitable methods from the convex optimization literature such as (Nikolova and Ng, 2005; Beck and Teboulle, 2009; Chambolle and Pock, 2011; Boyd et al., 2011) can be selected depending on the structure of $R(u)$ to accelerate convergence to the global optimum. Further, some of these methods can also be extended to non-convex energies (Bolte et al., 2018; Valkonen, 2021), however, they can only be guaranteed to converge to some stationary point and not the global optimum.

3.1.2 *Bayesian Approach for Image Recovery*

The Bayesian approach to inverse problems involves modeling the unknown image u and the measurement f as realizations of random variables (Kaipio and Somersalo, 2006) with respective distributions $P(u)$ and $P(f)$. The *prior* distribution $P(u)$ encodes the desired properties of the solution, with a suitably selected prior giving a high likelihood for 'good' images and a low likelihood otherwise. Some examples of Bayesian priors include generalized Gaussian priors (Bouman and Sauer, 1993), Markov random field models (Li, 1994), edge-preserving priors (Chantas et al., 2006), Bayesian priors based on wavelets (Bioucas-Dias, 2006), Gaussian mixture models on patches (Zoran and Weiss, 2011), and priors combining multiple probabilistic models such as field of experts (Roth and Black, 2009).

Given a measurement f, the inverse image to be recovered is characterized by its conditional distribution $P(u|f)$ known as the *posterior*, which is derived from the likelihood $P(f|u)$ and the prior $P(u)$ using Bayes theorem. The solutions to the Bayesian inversion problem may be obtained by sampling the posterior or through estimators such as the minimum mean square error (MMSE) estimate which is the conditional mean of the posterior, or the maximum-a-posteriori (MAP) estimate. Further, the MAP estimate can be related to the variational methods, with the data discrepency term corresponding to the likelihood, and the regularizer corresponding to the prior.

3.2 DEEP LEARNING FOR IMAGE RECONSTRUCTION

Following the success of deep neural networks in higher level vision tasks, deep learning approaches are increasingly being adopted in image reconstruction and restoration. These encompass a wide array of methods. In the following, we review the different deep learning approaches for such ill-posed image recovery problems.

3.2.1 *Fully Learned Methods*

These methods learn to directly invert the forward imaging model as

$$\hat{u} = \mathscr{G}(f;\theta). \tag{3.4}$$

Most commonly, the deep networks using direct inversion are trained in a supervised manner by minimizing some distortion measure between the network output with respect to the ground truth as

$$\min_{\theta} \sum_{\text{examples i}} \mathscr{L}(\mathscr{G}(f_i;\theta), u_i). \tag{3.5}$$

Earlier works (Xie et al., 2012; Burger et al., 2012; Kim et al., 2016; Zhang et al., 2017b) used simple pixel-wise ℓ_2 or ℓ_1 reconstruction losses with respect to the ground truth. Further improvements were observed by more advanced loss functions, for instance, losses based on structural similarity metric (Zhao et al., 2016), perceptual losses (Johnson et al., 2016) which employ deep features extracted from pretrained image classification networks, adversarial losses by discriminating between the network output and the clean data distribution (Wang et al., 2018a; Kupyn et al., 2019) in addition to pixel-wise losses, or even using task-specific learned loss functions (Mustafa et al., 2022). Along with improved loss functions, recent deep networks for image recovery (Zamir et al., 2022; Chen et al., 2022b; Mansour et al., 2022; Pang et al., 2022; Tu et al., 2022) adopted recent innovations in deep learning architectures such as transformers (Vaswani et al., 2017), MLP mixers (Tolstikhin et al., 2021) leading to further performance improvements. As the direct inversion approaches do not typically take into account the forward imaging model in the reconstruction process, they can also be used when the forward model is not known or cannot be modeled accurately, for example, in blind image restoration tasks (Noroozi et al., 2017; Zamir et al., 2022).

3.2.2 *Deep Neural Network Post-processors*

This involves a two-step approach where an initial reconstruction is obtained by an analytical reconstruction operator $B^{\dagger}(\cdot)$, which maps measurements to image space, and a post-processing network is trained to remove artefacts from this initial reconstruction:

$$\hat{u} = \mathcal{G}\left(B^{\dagger}(f)\right). \tag{3.6}$$

Examples of this approach include using a learned network $\mathcal{G}$ to remove artefacts from the initial solution obtained by pseudo-inverse operation or adjoint operation for compressive sensing (Mousavi et al., 2015) or under-sampled magnetic resonance imaging (Lee et al., 2017) or a filtered-back-projection operator for CT recovery (Chen et al., 2017). A common approach for such post-processing networks is to use residual learning (He et al., 2016) to recover the difference between initial reconstruction and ground truth. While this approach provides good reconstruction performance, this may not guarantee data consistent solutions. To address this, (Schwab et al., 2019) explicitly constrain the learned residual to be in the null space of the forward operator and provide convergence guarantees for their approach.

3.2.3 *Unrolled Optimization Networks*

While post-processing networks use the model knowledge once to obtain an initial estimate using a known operator to go from measurement space to image space, unrolled optimization (Gregor and LeCun, 2010) uses this knowledge to alternate between measurement and image spaces in an iterative algorithm with a fixed number of iterations, where some of the intermediate operations are learned using parameterized deep network modules. Starting with (Gregor and LeCun, 2010), several works explored unfolding different model-based algorithms, for instance, learned ISTA (Gregor and LeCun, 2010; Zhang and Ghanem, 2018), learned ADMM (Sun et al., 2016) learned gradient descent (Adler and Öktem, 2017; Gong et al., 2020), learned primal-dual (Adler and Öktem, 2018), proximal gradient algorithms (Mardani et al., 2018; Putzky and Welling, 2017). The learning can be performed at different levels of abstraction, from learning the same hyperparameters for all iterations which is run till convergence (Gong et al., 2020), to using different hyperparameters for each iteration, or learning neural network modules to approximate proximal steps for each iteration (Adler and Öktem, 2018). Instead of learning a different set of network parameters for the proximal step in each iteration, a few works (He et al., 2018; Gupta et al., 2018; Aggarwal et al., 2018) share the same network for the proximal steps across iterations.

In comparison with the fully learned approaches, unrolled networks tend to require less training data, and allow for more interpretable, and parameter-efficient learning (Monga et al., 2021). Further, learning on specific datasets allows them to capture data-dependent context resulting in better performance than classical approaches. As the unrolled networks are trained typically using a small number of unrolled steps, the inference is also faster in comparison to classical optimization-based approaches, which may need more iterations to converge. On the other hand, testing unrolled networks using more inference steps than used in training typically results in severe artifacts. Recent work (Gilton et al., 2021a) addresses this shortcoming through deep equilibrium models (Bai et al., 2019) which incorporate fixed-point convergence by construction, and sharing the same set of parameters across iterations. This allows the learned optimization scheme to be unrolled for arbitrary iterations without any degradation in reconstruction quality.

3.2.4 *Neural Network Priors for Variational Inference*

While unrolled optimization networks do learn modules that substitute proximal operation or gradient with respect to a regularizer at each step, these are not necessarily shared across unrolled steps. Further, task-specific supervised training means that these modules suffer from performance drop when there are modifications to the forward operator. An alternate approach is to use neural networks as priors in variational inference. In contrast to the dedicatedly trained networks, this approach endows the algorithm with the flexibility to handle different measurement models, while improving up on the performance of handcrafted priors. This class of methods includes learning regularizers, using trained networks such as denoisers, generative models, and even untrained neural networks as priors in variational image recovery. We now discuss each of these approaches.

Learned Regularizers

In this approach, one learns a regularizer or some parameters of a regularizer for subsequent use in an iterative optimization scheme. Some example non-deep learning methods to data-driven regularizers include learning linear sparsity promoting dictionaries (Bruckstein et al., 2009), Gaussian mixture models learned on image patches (Zoran and Weiss, 2011), field of experts (Roth and Black, 2009) which learn the distribution of filter responses on images, and learning task-specific regularizers (Gilboa, 2013). We now briefly discuss some deep learning approaches to learning regularizers. One approach is to explicitly parameterize the regularization functional using a neural network $R(u;\theta)$, (Li et al., 2020b; Lunz et al., 2018; Mukherjee et al., 2021; Kobler et al., 2020; Goujon et al., 2023) which may be trained based on different objectives. While Li et al. (2020b) use a neural network trained to penalize artifacts in the recovered solution, Kobler et al. (2020) train a neural network regularizer motivated by sparsity penalties. Lunz et al. (2018); Prost et al. (2021); Mukherjee et al. (2021) learn regularizers which are trained adversarially to distinguish between samples from the training data distribution and degraded samples. Instead of directly parameterizing the regularizers, Chang et al. (2017) learn a proximal operator with respect to regularizer, Heaton et al. (2022) learn projection operators onto clean data manifolds, and Moeller et al. (2019) learn a descent direction using parameterized deep networks. While learned regularizers improve reconstruction performance over handcrafted priors, they may not always guarantee convergence. Guaranteeing stability or convergence requires imposing additional constraints on the regularizer. Moeller et al. (2019) train networks to output descent direction with a provable convergence to a minimizer of the energy. Some works constrain the regularizer to ensure a convergent iterative scheme. Lunz et al. (2018); Mukherjee et al. (2021) impose Lipschitz-continuity on the regularizer via a soft-penalty, and Mukherjee et al. (2020) enforce convexity of regularizer using input convex networks (Amos et al., 2017) for stability and convergence.

Denoiser Priors

There are two main approaches to using denoiser priors for image recovery- as proximal operators, or in a functional representing the gradient of regularizer. Plug-and-Play (PnP) methods (Venkatakrishnan et al., 2013; Chan et al., 2016) replace proximal operators with respect to regularizer by generic denoisers such as non-local means (Buades et al., 2005) or BM3D (Dabov et al., 2007) in proximal splitting algorithms. Subsequently Zhang et al. (2017c); Meinhardt et al. (2017) proposed the use of pretrained neural network denoisers as proximal operators with good empirical results. In a follow-up work, Zhang et al. (2021b) proposed a more powerful denoiser for PnP image recovery. An alternate approach is regularization by denoising (RED) using denoisers D_θ in a regularization functional of the form $\langle u, u - D_\theta(u)\rangle$ (Bigdeli et al., 2017; Romano et al., 2017) in a gradient descent based scheme. While both PnP and RED approaches empirically provide very good reconstructions, they require strong conditions on the denoiser to have convergence guarantees. The denoiser replacing the proximal operator should be non-expansive, or in the RED framework, the denoiser should additionally have a symmetric Jacobian. These restrictive conditions are not satisfied by arbitrary denoising networks (Reehorst and Schniter, 2018). A few approaches constrain the denoiser to satisfy properties required for convergence, for instance, Ryu et al. (2019); Terris et al. (2021) train denoisers with constrained Lipschitz constants, and Cohen et al. (2021) derive image denoisers with symmetric Jacobians, Hasannasab et al. (2020) parameterize 1-Lipschitz operators for denoising. On the other hand, Sommerhoff et al. (2019) observed that enforcing non-expansiveness drastically decreased the denoising performance. Instead of constraining the denoisers, Sommerhoff et al. (2019) project the outputs of arbitrary denoisers onto the cone of descent directions to a given energy in a (proximal) gradient descent algorithm for provable convergence.

Generative Priors

This approach involves the use of trained generative models as priors for image recovery. Generative models such as generative adversarial networks (GANs) (Goodfellow et al., 2014), variational autoencoders (VAEs) (Kingma and Welling, 2013), diffusion based models (Ho et al., 2020) are trained to produce new samples from the underlying distribution of the training data. When the image to be recovered belongs to the distribution of images that a generative model is trained on, then this unknown image can be modeled by this generative model. Bora et al. (2017) first proposed the use of deep generative model priors for image recovery by optimizing for a vector in the smaller dimensional latent space of a trained GAN to minimize the reconstruction error:

$$\hat{u} = \mathscr{G}(\hat{z};\theta) \text{ s.t. } \hat{z} = \arg\min_{z} \|f - A\mathscr{G}(z;\theta)\|^2, \tag{3.7}$$

and demonstrated significant improvements over the classical priors for compressive sensing with very low measurements. Subsequently, several works developed different algorithms for image recovery using a variety of generative priors. We discuss these approaches in detail in Chapter 4.

Untrained Neural Network priors

Ulyanov et al. (2018) proposed that the structure of a randomly initialized convolutional generator can be a good prior to capture natural image statistics, which they referred to as 'Deep Image Prior (DIP)'. Ulyanov et al. (2018) use DIP to solve inverse problems such as denoising, inpainting, and super-resolution by optimizing the untrained network weights to minimize reconstruction error:

$$\hat{u} = \mathscr{G}(z_0;\hat{\theta}) \text{ s.t. } \hat{\theta} = \arg\min_{\theta} \|f - A\mathscr{G}(z_0;\theta)\|^2. \tag{3.8}$$

Subsequent works (Veen et al., 2020; Bostan et al., 2020; Baguer et al., 2020) extend the use of DIP to solve inverse problems such as compressed sensing, phase microscopy, and low dose CT recovery, with additional regularization. Ulyanov et al. (2018) used an over-parameterized UNet (Ronneberger et al., 2015) for $\mathscr{G}$ and suggested early stopping of the optimization in eq. (3.8) to prevent overfitting. Heckel et al. (2019) instead use an under-parameterized non-convolutional generator which prevents overfitting. More recent works (Chen et al., 2020b; Ho et al., 2021; Liu et al., 2023b) even search for neural architectures to be used as deep image priors.

3.3 WHAT MAKES A GOOD RECOVERY ALGORITHM?

Till now we have discussed approaches to inverse image reconstruction, ranging from classical approaches to deep learning methods, and a variety of methods using the combination of both. We now ask what are the desirable properties of a reconstruction algorithm $\mathscr{G}$.

Consistency The most common approach used to evaluate a reconstruction algorithm is by measuring discrepency of the solution provided by the method with respect to the ground truth, in terms of the reconstruction error. For a *good* reconstruction algorithm, it is also desirable that reconstruction error vanishes as measurement noise tends to zero. Yet, recent work (Blau and Michaeli, 2018) reported that deep networks trained to minimize such reconstruction error may suffer from a limited perceptual quality. Further, in case of ill-posed problems, the ground truth would be only one instance among many solutions that explain the measurement equally well. Therefore, a reconstruction algorithm that guarantees a high data consistency with the measurement, while satisfying certain desired image properties is more reasonable. For variational regularization methods, such guarantees are provided through fixed point convergence to the minimizer of energy (Benning and Burger, 2018). As discussed in the previous sections, many deep learning approaches do not come with such guarantees, and some guarantees may be obtained by explicitly constraining the network or network-based iterative schemes.

Reconstruction Quality It is desirable that a reconstruction algorithm produces natural-looking images with high perceptual quality, free from artifacts. This perceptual quality may be quantified using the deviation of the reconstructions from natural image statistics (Blau and Michaeli, 2018), through divergence between distributions of reconstructions and natural images (in the training data). Another popular approach to measuring perceptual quality is by measuring deviation between the deep features of reconstruction and ground truth extracted from pretrained networks (Zhang et al., 2018b).

Stability Another important desirable property for reconstruction algorithm is *stability*, in the sense that an algorithm's output varies smoothly with respect to changes in the input, i.e.,

$$\left\|\mathscr{G}(f+\delta)-\mathscr{G}(f)\right\| \to 0 \text{ as } \|\delta\| \to 0. \tag{3.9}$$

It is desirable that the maximum deviation in reconstruction with respect to change in measurement δ is controlled, this notion can be formalized by Lipschitz continuity. While stability is desirable, it can be in conflict with the objective of achieving high accuracy in terms of proximity to the ground truth, especially for ill-posed problems. Another approach to analyze worst-case stability is by analyzing the behaviour of the reconstruction method with worst-case inputs in the form of adversarial examples. We discuss in more detail the issues pertaining to the stability of reconstruction methods in Chapter 6, which deals with the adversarial robustness of image recovery methods.

Diversity Another desirable property for a reconstruction algorithm is the ability to sample diverse solutions, especially for ill-posed problems, where different solutions satisfy consistency equally well. Yet, most end-to-end trained networks produce a single solution out of several valid solutions. Among classical methods, Bayesian image recovery approaches allow sampling the solution space (by sampling the posterior). Among deep learning approaches, reconstruction methods utilizing conditional Bahat and Michaeli (2020); Lugmayr et al. (2020); Buhler et al. (2020) or unconditional generative models Menon et al. (2020); Montanaro et al. (2022) models permit sampling diverse solutions. A few prior works also propose to explore solution space using graphical inputs Bahat and Michaeli (2020) or semantic maps Buhler et al. (2020).

Robustness to Measurement Model Changes Another desirable property for an image reconstruction algorithm is robustness to changes in the measurement model. Classical variational approaches allow such modifications, for example, changes in the noise model, or modifications to the forward operator A, or any other change that can be modeled can be incorporated into the energy minimization by appropriate changes to the energy function. However, end-to-end trained neural network reconstructors, including the model-based unrolled networks suffer from a lack of adaptivity. This means that a network trained for a specific forward operator A, and noise model suffers from a significant performance drop if these are modified, and therefore have to be retrained for the new measurement model. To address this, Gilton et al. (2021b) propose fine-tuning based as well as training-free approaches to adapt trained models to variations in forward operator, whereas Gossard and Weiss (2022) propose training with different forward operators. Hu et al. (2023) show that unrolled networks based on deep equilibrium models Gilton et al. (2021a) are robust to changes in measurement model. In contrast, learned image priors such as learned regularizers, denoiser priors and generative priors are trained independent of a specific measurement model, and can be used in an energy minimization framework with an energy function tailored to any measurement model.

A significant part of this thesis deals with these issues of consistency, stability, and generalization in image reconstruction. We find that adversarial attacks lead to inaccurate results in terms of proximity to ground truth or clean reconstruction, but the resulting reconstructions may still maintain a good degree of data consistency. We exploit this observation to design adversarial attacks on CT recovery to produce diagnostically different solutions with high data consistency, using deterministic image recovery networks that otherwise produce a single arbitrary solution. We attempt to obtain *consistent* solutions to different image recovery problems using generative priors, by minimizing data discrepancy or by explicitly imposing data consistency. Thanks to generative priors which are trained independent of the specific measurement model, these approaches are also conferred with adaptivity and robustness to changes in the measurement model.

4 DEEP GENERATIVE MODELS AND APPLICATION TO IMAGE RECOVERY

4.1 OVERVIEW OF DEEP GENERATIVE MODELS

Deep generative models have emerged as an important tool for learning data representations in an unsupervised manner. Given a set of training samples, the goal of generative models is to learn an approximation of the distribution of the training data according to a chosen statistical criterion, and produce new samples from the underlying distribution. Generative models often consider a latent space with a known distribution, such as Gaussian, from which the latent variables are drawn, these are mapped to the data space by a generator. Different classes of generative models exist, depending on how the distance between the generated and actual distributions is measured and approximated during optimization. Popular deep generative models include generative adversarial networks GANs (Goodfellow et al., 2014), variational autoencoders (VAEs) (Kingma and Welling, 2013), normalizing flow-based models (Dinh et al., 2017) and diffusion-based models (Ho et al., 2020). As these models learn to model the distribution of training data, they can also serve as useful priors for inverse imaging tasks, when the image to be recovered belongs to the same distribution they are trained on.

Deep generative models can be broadly classified into likelihood-based generative models and implicit generative models, depending on how they represent probability distributions. Given a dataset of examples drawn from a distribution, likelihood-based models optimize for model parameters that maximize the log-likelihood of the training data or an approximation of the likelihood. Examples of likelihood-based generative models include variational autoencoders (Kingma and Welling, 2013), auto-regressive models (Salimans et al., 2017), energy based models (Grathwohl et al., 2020), normalizing flow based models (Dinh et al., 2017), and diffusion based or score based generative models (Song et al., 2021b; Ho et al., 2020). Among these, autoregressive and normalizing flow based models provide exact likelihoods, whereas VAEs optimize for a lower bound of the likelihood. Diffusion-based models learn the gradients of the log probability density function of the data distribution (also referred to as score). Generative adversarial networks (GANs) (Goodfellow et al., 2014) in contrast, do not compute the likelihoods of training samples, and are examples of implicit generative models. GAN training involves optimization through a zero-sum game between the generator and the discriminator. These different generative models have specific limitations. Autoregressive models have very high sampling costs due to sequential sampling of pixels, normalizing flows are limited to using specific invertible architectures, and the images produced by VAEs are often blurry. While GANs can achieve impressive image quality, they suffer from mode collapse, and the generated samples lack high diversity. Diffusion models, on the other hand, achieve high fidelity and diversity in the generated images (Nichol and Dhariwal, 2021; Ho et al., 2020). Yet, the generation process is slower due to iterative sampling, as opposed to VAEs and GANs which need only a single forward pass of a network to generate image samples. In the following, we provide an overview of three classes of generative models which we encounter in this thesis, including generative autoencoders, generative adversarial networks (GANs), and diffusion based generative models.

4.1.1 *Variational Autoencoders*

Autoencoders (Vincent et al., 2010) are networks trained to reconstruct the input data. An autoencoder learns two components: an encoder that transforms the input data into a latent code, and a decoder that reconstructs the input data from the latent code. Different techniques have been proposed to encourage autoencoders to learn useful representations, such as denoising, contractive and sparse autoencoders (Bengio et al., 2013). Typically, the latent codes are of much smaller dimension than the input data, and such an encoding is useful for dimensionality reduction, and the learned representations are useful for downstream tasks such as classification or anomaly detection. While the compressed representations are useful, these autoencoders cannot generate samples similar to training data. To turn an autoencoder into a model which is capable of generating new samples similar to training data, the latent space is regularized to approximate a known distribution called prior distribution, for example, a Gaussian distribution. Training is performed using a combination of reconstruction loss and a term penalizing the deviation between the distribution of encoder output, and the prior distribution used to model the latent space. New samples can then be generated by the decoder (also referred to as the generator) which maps the randomly drawn samples from the prior distribution to the data space.

We now briefly recall the variational autoencoders (VAE) which are the first generative autoencoders introduced in (Kingma and Welling, 2013). We assume that the data samples x come from an unknown distribution $p(x)$, and each data sample has a lower dimensional latent representation z, which follows a prior distribution $p_z(z)$. The objective is to find a distribution $p_\theta(x)$ that is the best fit to the underlying data distribution. This is achieved by maximizing the marginal likelihood or *evidence* given by

$$p_\theta(x) = \int_{\mathcal{Z}} p_\theta(x|z) p_z(z) dz, \tag{4.1}$$

where, $p_\theta(x|z)$ is referred to as likelihood and is approximated by a generative model, a neural network parameterized by θ (also referred to as the decoder in VAE). The $p_\theta(z|x)$ is the posterior whose approximation $q_\phi(z|x)$ is given by a probabilistic *encoder* parameterized by ϕ, which outputs the parameters of this conditional distribution. When the prior is assumed to be a multivariate standard Gaussian $p_z = \mathcal{N}(0, I_d)$, then the approximate posterior is given as $q_\phi(z|x) = \mathcal{N}\big(z\big|\mu(x), \Sigma(x)\big)$, where $\big(\mu(x), \Sigma(x)\big)$ are outputs of the encoder network. This is in contrast with the encoder in deterministic autoencoders which output a latent code z. As maximizing the true likelihood is intractable, Kingma and Welling (2013) train the two networks (the encoder parameterized by ϕ and the decoder parameterized by θ) together to minimize a lower bound of the logarithm of likelihood in eq. (4.1)

$$\log \underbrace{\mathbb{E}_{z\sim q_\phi}\big[\widehat{p}_\theta(x)\big]}_{p_\theta(x)} \geq \mathbb{E}_{z\sim q_\phi}\big[\log \widehat{p}_\theta(x)\big] = \underbrace{\mathbb{E}_{z\sim q_\phi}[\log p_\theta(x|z)]}_{\text{reconstruction}} - \underbrace{\mathcal{D}_{KL}\big[q_\phi(z|x)||p_z(z)\big]}_{\text{regularisation}}, \tag{4.2}$$

This bound is referred to as known as evidence lower bound (ELBO). The first term in the loss $\mathbb{E}_{z\sim q_\phi}[\log p_\theta(x|z)]$ corresponds to a *reconstruction* loss (mean squared error loss when z is Gaussian) between the output of the decoder and the original input x. $\mathcal{D}_{KL}(p||q)$ is the Kullback-Leibler divergence between prior distribution p_z and the approximate posterior distribution given by encoder output $q_\phi(z|x)$. Following the seminal work of VAEs by Kingma and Welling (2013), several works have proposed improvements, for example by modifying prior, tightening the gap between true likelihood and the lower bound, or changing the statistical criterion used to measure the distance between distributions. We refer to (Chadebec et al., 2022) for an overview and comparison between different generative autoencoders.

In chapter 7 of this thesis, we consider a class of generative autoencoders known as Wasserstein autoencoders (WAEs) introduced in (Tolstikhin et al., 2018) which utilize optimal transport (Villani, 2021) to measure the distance between the distributions p_z and q_z. In theory, WAEs can use any cost function including squared error to minimize discrepency between the input and the decoder output, and any statistical distance measure in the latent space. We consider WAEs optimized using a combination of mean squared error loss between input and decoder output, and a maximum mean discrepency penalty between encoder distribution and prior latent distribution, and use this model as prior for light field recovery.

4.1.2 *Generative Adversarial Networks*

Generative adversarial networks (GANs) (Goodfellow et al., 2014) do not explicitly define the likelihood of training data. Instead, GAN training involves optimizing a generator and a discriminator in parallel, such that the generator implicitly learns the distribution of training data. The generator G draws samples z from a lower dimensional latent space, for example, Gaussian noise, to map to images from the desired distribution, whereas the discriminator D learns to maximize the probability of correctly distinguishing between the training examples and the samples generated by G. Both the models G and D are trained together using back-propagation in an attempt to realize the Nash equilibrium of the following two-player mini-max game:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]. \tag{4.3}$$

While GANs produce impressive results on realistic image generation, they have two major drawbacks, in terms of difficulty in convergence, and mode collapse which leads to limited diversity of generated images. The adversarial training of the generator and the discriminator in a zero-sum game can become unstable. During training, a poorly trained generator may produce samples away from the original distribution, which can easily be distinguished by the discriminator. This further affects the trainability of the generator due to high confidence predictions made by the discriminator. On the other hand, the generator could learn to generate a limited number of realistic samples to fool the discriminator and repeat these across iterations. This affects the trainability of the discriminator, resulting in it being confined to a local minimum, as the discriminator does not see diverse generated samples. It is therefore crucial to realize a good equilibrium between the discriminator and generator.

Several strategies have been developed to stabilize and improve GAN training and convergence, including modifying training objective function, for example, using Wasserstein distance and gradient penalty (Arjovsky et al., 2017), regularization through noise (Jenni and Favaro, 2019), use of extrapolation based methods (Yadav et al., 2018; Daskalakis et al., 2018; Gidel et al., 2019) which take an additional step along a predicted gradient using look ahead step. Significant improvements in GAN generation quality were possible due to novel architectures and training schemes. We refer the reader to (Razavi-Far et al., 2022) for a recent overview of different training methods, architectures developed for GANs, and their applications. In the following, we briefly touch upon popular GAN variants. Karras et al. (2018a) proposed progressive training to stabilize GAN training for higher resolution images, which was improved upon by Big GAN (Brock et al., 2019) that remained the state-of-the-art GAN model for image synthesis on ImageNet (Russakovsky et al., 2015) till recently. Style-based GANs (Karras et al., 2019, 2021) achieved high photorealism on restricted datasets such as human faces, and have recently been scaled to ImageNet dataset in (Sauer et al., 2022) by incorporating additional class-information. StyleGANs include a mapping network that maps a lower dimensional latent code to a higher dimensional style space, which is subsequently mapped to image space by a synthesis network. This allows the disentangling of factors of variation into style and content, and therefore StyleGANs have been very popular in image manipulation (Collins et al., 2020; Shen et al., 2020; Zhu et al., 2020; Wu et al., 2021), and also in image reconstruction (Menon et al., 2020), for specific class of images like faces. Vector-quantized GAN (VQGAN) (Esser et al., 2021b) is another popular generative model, that has recently become popular for image manipulation (Crowson et al., 2022). In contrast to the usual GAN training, VQGAN employs a two-stage training approach where a vector quantized autoencoder is first trained adversarially to learn a codebook representation. This allows images to be represented as a sequence of codebook indices of corresponding image embeddings. Image generation is performed by a Transformer (Vaswani et al., 2017) trained on top of the codebook for codeword sequence prediction. In chapters 8 and 9 of this thesis, we compare with StyleGAN and VQGAN based approaches for text guided image manipulation and reconstruction.

4.1.3 *Diffusion based generative models*

Diffusion based generative models are a class of likelihood-based models built from a hierarchy of denoising auto-encoders (Vincent et al., 2008). These models have recently demonstrated high quality generative capabilities surpassing GANs (Ramesh et al., 2022; Dhariwal and Nichol, 2021; Nichol and Dhariwal, 2021), and have better training stability and mode coverage than GANs. Two main works in this field are Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020) and score-based Stochastic Differential Equation (SDE) (Song and Ermon, 2019; Song et al., 2021b). We now look into DDPM formulation.

Denoising Diffusion Probabilistic Models

The core idea of the generation process in diffusion models is iterative denoising via denoising autoencoders (Vincent et al., 2008). Denoising diffusion probabilistic models (DDPM) (Ho et al., 2020) define a forward diffusion process that gradually perturbs the input data into pure Gaussian noise. Image generation is achieved via a reverse process by gradually removing Gaussian noise, inspired by non-equilibrium thermodynamics (Sohl-Dickstein et al., 2015).

i)Forward process: Given a data sample x_0 sampled from data distribution $q(\mathbf{x})$, the forward diffusion process slowly adds Gaussian noise to data sample in T steps under Markov assumption, resulting in a series of noisy samples $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T$. The evolution of a noised sample $\mathbf{x}_t$ at time-step t is expressed as:

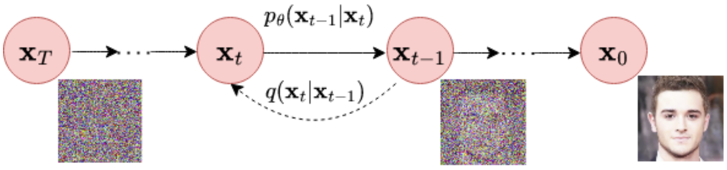

$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$
$\mathbf{x}_T$
$\mathbf{x}_t$
$\mathbf{x}_{t-1}$
$\mathbf{x}_0$
$q(\mathbf{x}_t|\mathbf{x}_{t-1})$

Figure 4.1: Depiction of forward and reverse diffusion process, image from (Ho et al., 2020)

$$
\begin{aligned}
q(\mathbf{x}_t|\mathbf{x}_{t-1}) &:= \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}\right), \\
i.e., \quad \mathbf{x}_t &= \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0,\mathbf{I}),
\end{aligned}
\tag{4.4}
$$

where, $\{\beta_t\}_{t=0}^{T}$ is the noise variance schedule, usually with $\beta_1 < \beta_2 \cdots < \beta_T$,and $\mathcal{N}$ represents the Gaussian distribution. For a large T, $\mathbf{x}_T$ becomes isotropic Gaussian noise. Using reparameterization trick, eq. (4.4) can be modified to sample at any time step t:

$$
\begin{aligned}
q(\mathbf{x}_t|\mathbf{x}_0) &:= \mathcal{N}\left(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}\right), \\
\text{with} \quad \alpha_t &= 1-\beta_t, \quad \bar{\alpha}_t = \prod_{i=0}^{t}\alpha_i.
\end{aligned}
\tag{4.5}
$$

ii) Learned reverse process: The reverse process $p_\theta(\mathbf{x}_0) := \int p_\theta\,(\mathbf{x}_{0:T} d\mathbf{x}_{1:T})$ learns to reverse the dynamics of the forward process by iterative denoising in T steps resulting in a sample from the distribution of training data. This is also a Markov chain with learned Gaussian denoising steps starting from $p(\mathbf{x}_T) = \mathcal{N}\,(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$ with transitions expressed as:

$$
\begin{aligned}
p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) &:= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_t(\mathbf{x}_t,\mathbf{x}_0), \sigma_t^2\mathbf{I}), \quad \text{where} \\
\boldsymbol{\mu}_t(\mathbf{x}_t,\mathbf{x}_0) &= \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \epsilon_\theta(\mathbf{x}_t, t)\frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\right) \quad \text{and} \\
\sigma_t^2 &= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t, \\
i.e., \quad \mathbf{x}_{t-1} &= \boldsymbol{\mu}_t(\mathbf{x}_t,\mathbf{x}_0) + \sigma_t z \quad \text{with} \quad z \sim \mathcal{N}(0,\mathbf{I}).
\end{aligned}
\tag{4.6}
$$

Clean images are generated by iterative sampling eq. (4.6) in the reverse diffusion process exploiting the learned neural network noise approximator ϵ_θ. Given an arbitrary noised version of data sample $\mathbf{x}_t$ at step t, DDPM training involves training a network ϵ_θ to predict noise at this step, i.e. to minimize the objective $\|\epsilon_t - \epsilon_\theta(\mathbf{x}_t, t)\|^2$. Score based generative models (Song et al., 2021b) have an alternate formulation, which results in similar forward and reverse processes. The training involves training a network to predict the score of the sample at an arbitrary noise level $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$. We refer the reader to (Luo, 2022) for a more detailed understanding of the different formulations.

While diffusion models have demonstrated impressive generative capabilities, high quality diffusion models are computationally expensive to train and have slower inference times than GANs. This is due to expensive Markovian sampling and iterative network evaluations required for generation. These problems can be alleviated by accelerated stochastic sampling techniques, or by performing diffusion in a smaller latent space (Rombach et al., 2022; Vahdat et al., 2021). Employing deterministic diffusion process (Song et al., 2021a) can also speed up inference, in addition to enabling high fidelity sample reconstruction.

4.2 GENERATIVE PRIORS FOR IMAGE RECOVERY

We now discuss the use of trained generative models as priors for image recovery. When an image to be recovered belongs to the distribution of images that a generative model is trained on, then this unknown image can be modeled as being in the range or closer to the range of the generative model, which is the key idea of using generative priors for image recovery. In the following, we deal with only zero-shot approaches utilizing pretrained generative models for image recovery. It is also possible to train conditional generative models for a specific recovery task, we do not discuss such methods here.

GAN Priors: The pioneering work in using deep generative model priors for image recovery is by Bora et al. (2017), which optimizes for a vector in the smaller dimensional latent space of a trained GAN to minimize the reconstruction error:

$$\hat{u} = \mathscr{G}(\hat{z};\theta) \quad \text{s.t.} \quad \hat{z} = \arg\min_{z} \| f - A\mathscr{G}(z;\theta)\|^2. \tag{4.7}$$

Bora et al. (2017) investigate incorporating an ℓ_2 regularization on z in eq. (4.7), which corresponds to a MAP estimate with respect to z, assuming a Gaussian prior on z. This approach was shown to significantly outperform the classical sparsity based priors for compressive sensing with very low measurements, and was later extended to non-linear inverse problems in (Bohra et al., 2022). While Bora et al. (2017) used simple gradient descent based algorithms to solve eq. (4.7), later works such as (Latorre et al., 2019a; Raj et al., 2019) also investigated the use of algorithms such as ADMM and projected gradient descent for image recovery using GAN priors. An advantage of latent space exploration is the ability to obtain diverse solutions by using different starting latent codes (Menon et al., 2020; Marinescu et al., 2021; Pan et al., 2021). Montanaro et al. (2022) show that this can be accelerated by finding latent space directions in the null space of the forward operator.

A major limitation of the latent space optimization eq. (4.7) is that samples outside the range manifold of the generator can not be reconstructed accurately resulting in a non-trivial representation error. Subsequent works attempt to reduce this representation error using different approaches. Dhar et al. (2018) allow a small deviation of the recovered image from the range of a generator with sparsity prior on the difference, which is extended to optimizing intermediate layer representations in (Daras et al., 2021). Pan et al. (2021) adopt a two-step approach of latent space optimization followed by fine-tuning both the latent vector and generator parameters.

VAE Priors: In addition to using GAN priors, Bora et al. (2017) also explore the approach of latent space optimization using VAEs, using the trained decoder (generator). This approach has similar limitations as GAN priors, it cannot recover images outside the range of the generator, leading to representation error. In chapter 7 of this thesis (also published in (Chandramouli et al., 2022)), we explore the use of generative autoencoders for variational image recovery. Our approach involves latent space optimization of a conditional generative autoencoder for generic light field recovery. More recently, Prost et al. (2023) utilize more powerful hierarchical VAEs and design an efficient Plug-and-Play algorithm for inverse problems. González et al. (2022) propose a framework for inverse problems using VAEs by considering a joint posterior distribution of latent and image space, with guaranteed convergence to a stationary point.

Flow Priors: Normalizing flow based models are a class of invertible generative models that learn to transform complex distribution such as images into a simpler distribution like Gaussian through a series of invertible transformations. By drawing random samples from the Gaussian distribution, image samples are generated by traversing through the model backward during testing. A few recent works have investigated the use of flow-based generative models for image recovery. Asim et al. (2020a) replace GAN prior in eq. (4.7) with a flow-based generator, with additional ℓ_2 regularization on z. Whang et al. (2021); Kothari et al. (2021) generalize this to arbitrary differentiable measurement operators and measurement noises using a maximum aposteriori framework, with Kothari et al. (2021) using a generalized version of flow models which progressively increase dimension from a low-dimensional latent space. Cai et al. (2023) instead use Langevin-based sampling using normalizing flow models for Bayesian inference. Runkel et al. (2023) jointly train two normalizing flows learning the distributions of measurement and unpaired ground truth images connected by a common latent space, and propose a point estimator optimizing data discrepancy term.

Diffusion Priors One could utilize diffusion models for image recovery either by training a conditional diffusion model for specific recovery tasks or by leveraging diffusion based image generative models for zero-shot image recovery. We are concerned with the latter variety, which exploits the knowledge of the degradation operator in a guidance mechanism to modify the sampling process. Earlier works (Jalal et al., 2021a; Kadkhodaie and Simoncelli, 2021) adopt Langevin dynamics for linear inverse problems and incorporate measurement guidance through the gradient of the least-squares data fidelity term. Choi et al. (2021); Chung et al. (2022c) propose to alternate between a standard reverse diffusion step and a projection step promoting measurement consistency on the intermediate noisy estimate. Subsequent works Chung et al. (2023, 2022b); Wang et al. (2023b); Lugmayr et al. (2022a); Kawar et al. (2022a); Song et al. (2023) predict clean sample at each reverse diffusion step, and use this estimate to promote measurement consistency. This can be achieved via a guidance function to guide the previous step through the gradient of the least-squares data fidelity term (Chung et al., 2023), or gradient based guidance from measurements through pseudo-inverse operation (Song et al., 2023) at each reverse step. Wang et al. (2023b); Lugmayr et al. (2022a); Kawar et al. (2022a) alternately employ the clean estimate in a consistency enforcing projection operation. While projection-based approaches are faster and do not need to backpropagate through network weights, they are restricted to inverse problems where a pseudo-inverse or its approximation can be computed. In contrast, gradient based measurement guidance can be applied to inverse problems, or even arbitrary guidance (Bansal et al., 2023), yet, it is more expensive as it requires back-propagation through the diffusion model weights at each iteration. In chapter 9 of this thesis, we adapt a projection based approach using range-null space decomposition (Wang et al., 2023b) to explore solutions of linear inverse problems through text.

Part III

METHODOLOGY

Declaration for Chapter 5 - Robustness to Group Transformations

This chapter is based on the paper Gandikota et al. (2022b) titled “A Simple Strategy to Provable Invariance via Orbit Mapping” co-authored by Kanchana Vaishnavi Gandikota, Jonas Geiping, Zorah Lähner, Adam Czapliński and Prof. Michael Moeller, published at the Asian Conference on Computer Vision (ACCV) 2022. Prof. Michael Moeller proposed the idea of developing a provably invariant method to image classification under group transformations. All the collaborators contributed to discussing ideas on achieving invariance. The idea of consistently orienting input prior to passing it to neural network classifiers was proposed by Zorah Lähner. Prof. Michael Moeller proposed the gradient based solution to achieve such consistent orientation, and proved invariance using this approach in section 5.4.1. Kanchana Vaishnavi Gandikota contributed to the discussions, reviewed the literature, performed all the experiments and comparisons for image classification, part of experiments with point clouds without spatial transformers, and contributed to writing major portions of the first draft of the paper. Jonas Geiping contributed to remaining experiments with point clouds and contributed to writing parts of experimental results with point clouds. Zorah Lähner contributed to writing isometry invariance. Adam Czapliński is a domain expert in group theory. Prof. Michael Moeller and Adam Czapliński contributed to mathematical formalism in the paper. The research was supervised by Prof. Michael Moeller. The paper benefited from several rounds of reviews where the reviewers provided several constructive suggestions. This feedback helped us to simplify and clarify our writing, and motivated us to provide a more comprehensive literature survey to place our work in the context of existing work, and resulted in thorough experiments which illustrate the benefit of our approach.

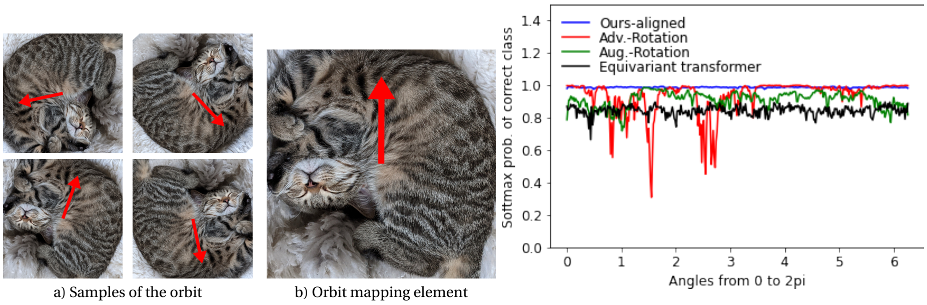
5 ROBUSTNESS TO GROUP TRANSFORMATIONS

Many applications require robustness, or ideally invariance, of neural networks to certain transformations of input data. In image classification, for instance, rotational, scale, and shift invariance are often highly desirable properties. While training deep networks with millions of realistic images in datasets like ImageNet (Russakovsky et al., 2015) confers some degree of in/equi-variance (Tensmeyer and Martinez, 2016; Olah et al., 2020; Lenc and Vedaldi, 2018), these properties however, cannot be guaranteed. On the contrary, networks are susceptible to adversarial attacks with respect to these transformations (see e.g. (Engstrom et al., 2017; Finlayson et al., 2019; Zhao et al., 2020b; Lang et al., 2021)), and small perturbations can significantly affect their predictions, as also discussed in the chapter 2. To counteract this behavior, the two major directions of research are to either modify the training procedure or the network architecture. Modifications of the training procedure replace the common training of a network $\mathscr{G}$ with parameters θ on training examples (x^i, y^i) via a loss function $\mathscr{L}$,

$$\min_{\theta} \sum_{\text{examples i}} \mathscr{L}(\mathscr{G}(x^i;\theta); y^i), \tag{5.1}$$

with a loss function that considers all perturbations in a given set S of transformations to be invariant towards. The most common choices are taking the mean loss of all predictions $\{\mathscr{G}(g(x^i);\theta) \mid g \in S\}$ (training with *data augmentation*), or the maximum loss among all predictions (*adversarial training*). However, such training schemes cannot guarantee provable invariance. In particular, training with data augmentation is far from being robust to transformations as illustrated in Fig. 5.1. The plot shows the softmax probabilities of the true label when feeding the exemplary image at rotations ranging from 0 to 2π into a network trained with rotational augmentation (green), adversarial training (red), and undoing rotations using a learned network (black). As we can see, rotational data augmentation is not sufficient to truly make a classification network robust towards rotations, and even the significantly more expensive adversarial training shows instabilities.

While modifications of the training scheme remain the best option for complex or hard-to-characterize transformations, more structured transformations, e.g., those arising from a group action, allow modifications to the network architecture to yield provable invariance. As opposed to previous works that largely rely on the ability to enlist all transformations of an input x (i.e., assume a finite *orbit*), we propose to make neural networks invariant by selecting a specific element from a (possibly infinite) orbit generated by a group action, through an application-specific *orbit mapping*. Simply put, we undo and fix the transformation or pose. Our proposed approach is significantly easier to train than adversarial training methods, and simultaneously results in better performance, robustness, and computational costs than adversarial training. We illustrate these findings on the rotation invariant classification of images (on which discretization artifacts from the interpolation after any rotation play a crucial role) as well as on the scale, rotation, and translation invariant classification of 3D point clouds. Our contributions in this chapter can be summarized as follows:



a) Samples of the orbit

b) Orbit mapping element

Figure 5.1: (Left) Picture of a cat in 4 different rotation samples from the continuous orbit of rotations. Our orbit mapping selects the element with mean gradient direction (marked in red) along a circle pointing upwards. (Right) Softmax probabilities of the true label when rotating an image by $0° - 360°$. Our method (in blue) is *robust for any angle*, which cannot be guaranteed through data augmentations (green) or adv. training (red).

- We present *orbit mapping*, a simple way to adapt neural networks to be in-(or equi)variant to transformations from sets S associated with a group action.
- We propose a gradient based orbit mapping strategy for image rotations, which can provably select a unique orientation for continuous image models.
- Our proposed orbit mapping consistently improves the robustness of standard networks to transformations even *without* additional changes in training or architecture.
- Existing invariant approaches also demonstrate a gain in robustness to discrete image rotations when combined with orbit mapping.
- We demonstrate orbit mappings to provable scale and orientation invariant 3D point cloud classification using well-known scale normalization and PCA.

5.1 RELATED WORK

Several approaches have been developed in the literature to encourage models to exhibit invariance or robustness to desired transformations of data. These include: i) data augmentation using desired transformations, ii) regularization to encourage network output to be robust to transformations on the input (Simard et al., 1991), iii) adversarial training (Engstrom et al., 2019; Wang et al., 2022a) and regularization (Yang et al., 2019), iv) unsupervised or self-supervised pretraining to learn transformation robust representations (Anselmi et al., 2016; Noroozi and Favaro, 2016; Komodakis and Gidaris, 2018; Zhang et al., 2019b; Gu and Yeung, 2021), v) parameterized learning of augmentations to learn invariances from training data(Wilk et al., 2018; Benton et al., 2020), vi) use of hand-crafted invariant shallow (Sheng and Shen, 1994; Yap et al., 2010; Tan, 1998; Lazebnik et al., 2005; Manthalkar et al., 2003) or deep (Bruna and Mallat, 2013; Sifre and Mallat, 2013; Oyallon and Mallat, 2015) features for downstream classification tasks vii) incorporating desired invariance properties into the network design (Cohen and Welling, 2016; Worrall et al., 2017; Weiler and Cesa, 2019; Zhang et al., 2020; Yu et al., 2020), and viii) train time/test time data transformation. Recent works Balunovic et al. (2019); Fischer et al. (2020) have also explored certifying the geometric robustness of networks. The approaches i)-v) can improve robustness but cannot yield provable invariance to transformations. Hand-crafting features can yield the desired invariance, but is difficult and often sacrifices accuracy. Provable invariance to a finite number of transformations is achievable by applying all such transformations to each input data point and pooling the corresponding features (Manay et al., 2006; Laptev et al., 2016). While this strategy can even be applied only during test time, it can not be extended to sets with infinitely many transformations. Recent approaches (Cohen and Welling, 2016; Ravanbakhsh et al., 2017; Weiler and Cesa, 2019) incorporate in-/equivariances when the desired transformations of the data can be formulated as a group action, e.g. enforcing equivariance in each layer separately. Layer-wise approaches for equivariance to finite groups such as (Cohen and Welling, 2016) typically use all possible transformations at each layer.

CANONICALIZATION Closely related to our approach are methods which align input to a normalized or canonical pose. The use of PCA or scale renormalization are well-known approaches to normalizing point clouds. However, PCA-based pose canonicalization is known to suffer from ambiguities, and learning based approaches (Xiao et al., 2020; Yu et al., 2020; Li et al., 2021) have been proposed for disambiguation. Several recent works directly leverage deep learning for 3d pose canonicalization, for example, training with ground truth poses (Rempe et al., 2020; Wang et al., 2019) or self-supervised learning (Sun et al., 2021; Spezialetti et al., 2020; Sajnani et al., 2022). For 2D images, PCA-based canonicalization is possible only with binary images (Rehman and Lee, 2018); the use of Radon transformations (Jafari-Khouzani and Soltanian-Zadeh, 2005) requires an expensive, fine discretization of continuous rotations. The use of spatial transformer networks (Jaderberg et al., 2015) is an alternate learning based approach to 2D/3D pose normalization which can be used along with an application-dependent coordinate transformation (Tai et al., 2019; Esteves et al., 2018b). Such learning-based approaches, however, require additional training with data augmentation and cannot guarantee invariance. Since our orbit mappings essentially select a canonical group orbit element, our work can be interpreted as a formalization of canonicalization for group transformations. In contrast to learning based approaches, we select a canonical element from the orbit using simple analytical solutions, which can improve robustness even without data augmentations.

PROVABLE ROTATIONAL IN-/EQUIVARIANCE IN 2D Several works (Sifre and Mallat, 2013; Oyallon and Mallat, 2015; Cohen and Welling, 2016; Marcos et al., 2017; Veeling et al., 2018; Marcos et al., 2016) have considered layer wise equivariance to discrete rotations using multiple rotated versions of filters at each layer, which was formalized using group convolutions in (Cohen and Welling, 2016). While Cohen and Welling (2016); Marcos et al. (2017); Veeling et al. (2018); Marcos et al. (2016) learn these filters by training, Sifre and Mallat (2013); Oyallon and Mallat (2015) make use of rotated and scaled copies of fixed wavelet filters at each layer. For equivariance to continuous rotations, Worrall et al. (2017) utilize circular harmonic filters at each layer. All these layer wise approaches for group equivariance in images were unified in a single framework in (Weiler and Cesa, 2019). Instead of layer-wise approaches, Fasel and Gatica-Perez (2006); Laptev et al. (2016); Henriques and Vedaldi (2017) pool the features of multiple rotated copies of images input to the network.

ROTATION INVARIANCE IN 3D Due to the different representations of 3D data (e.g. voxels, point clouds, meshes), many strategies exist. Some techniques for image invariances can be adapted to voxel representations, e.g. probing several rotations at test time (Wu et al., 2015; Wang et al., 2017), use of rotationally equivariant convolution kernels (Weiler et al., 2018b; Thomas et al., 2018; Fuchs et al., 2020). Spatial transformers have also been used to learn 3D pose normalization, e.g. in the classical PointNet architecture (Qi et al., 2017a), and its extension PointNet++ (Qi et al., 2017b) which additionally considers hierarchical and neighborhood information. While point clouds do not suffer from discretization artifacts after rotations, they struggle with less clear neighborhood information due to unordered coordinate lists. Zhang and Rabbat (2018) solve this by adding hierarchical graph connections to point clouds and using graph convolutions. However, the features learned using graph convolutions still depend on the rotation of the input data. Horie et al. (2020); Satorras et al. (2021) propose graph convolution networks equivariant to isometric transformations. Esteves et al. (2018a); Rao et al. (2019) project point clouds onto 2D sphere and employ spherical convolutions to achieve rotational equivariance. Deng et al. (2018) and Zhao et al. (2019) achieve rotation invariance on point clouds by considering pairs of features in the tangent plane of each point. While local operations and convolutions on the surface of triangular meshes are invariant to global rotations by definition (Monti et al., 2016), they however do not capture global information. MeshCNN (Hanocka et al., 2019) addresses this by adding pooling operations through edge collapse. Sharp et al. (2020) defines a representation independent network structure based on heat diffusion which can balance between local and global information.

5.2 PRELIMINARIES

We consider $\mathcal{G}$ to be a neural network parameterized by $\theta \in \mathbb{R}^p$ that maps data $x \in \mathcal{X}$ from some suitable input space $\mathcal{X}$ to some prediction $\mathcal{G}(x;\theta) \in \mathcal{Y}$ in an output space $\mathcal{Y}$. The question we attempt to tackle is how, for a given set $S \subset \{g : \mathcal{X} \to \mathcal{X}\}$ of transformations of the input data, we can achieve robustness or ideally *invariance* of $\mathcal{G}$ to S. We consider invariance of a network $\mathcal{G}$ with respect to transformations in S, where S induces a *group action* on $\mathcal{X}$, which is what we will assume about S for the remainder of this chapter. We begin by introducing the basic terminology used in the theory of groups. The reader is referred to (Rotman, 2012) for a detailed introduction to group theory.

Definition 1. ***Group:*** *A* group *is defined as a set S with a notion of* product *on its elements, which satisfies the following axioms*

(i) closure*: $a, b \in S \Longrightarrow$ the product $ab \in S$,*

(ii) associativity*: $(ab)c = a(bc)$, and*

(iii) inverse element*: for each $g \in S, \exists g^{-1} \in S$ such that $gg^{-1} = g^{-1}g = e \in S$, where e is the identity element satisfying $ge = eg = g, \forall\, g \in S$.*

A group is *abelian* if the group product is commutative ($gh = hg, \forall\, g, h \in S$). Each element $g \in S$ can be viewed as a transformation acting on an input space $\mathcal{X}$, $g : \mathcal{X} \mapsto \mathcal{X}$. The set of all rotations in a 2-D plane is an example of an infinite group, whereas a set of rotations by multiples of $\pi/2$ in the 2D plane is an example of a finite group. In this case of the rotation group, the consecutive application of two rotations becomes the group product.

Definition 2. ***Group Action:*** *A (left) group action of a group S with the identity element e, on a set X is a map $\sigma : S \times X \to X$, that satisfies*

(i) $\sigma(e, x) = x$ and

(ii) $\sigma(g, \sigma(h, x)) = \sigma(gh, x)$, $\forall g, h \in S$ *and* $\forall x \in X$.

When the action being considered is clear from the context, we write $g(x)$ instead of $\sigma(g, x)$.

Definition 3. ***Orbit:*** *The orbit of* $x \in \mathscr{X}$ *under the action of a group* S *is defined as the set of all possible transformations of* x*,*

$$S \cdot x = \{g(x) \mid g \in S\}, \tag{5.2}$$

The closure property of the group implies that the orbit of a point x is invariant under a group action on x, i.e., the orbit of $g(x)$ is the same as the orbit of x, for all $g \in S$. Continuing with our earlier example of a group of rotations on the 2-D plane, the orbit of an image under this group is the infinite set consisting of all rotated versions of the image. We now proceed to clarify the concepts of invariance and equivariance of functions with respect to group actions.

Definition 4. ***Invariant functions with respect to group actions:*** *A function* $\mathscr{G}$ *is said to be invariant to the action of a group* S *of transformations if*

$$\mathscr{G}(g(x)) = \mathscr{G}(x) \quad \forall x \in \mathscr{X},\ g \in S. \tag{5.3}$$

Definition 5. ***Equivariant functions with respect to group actions:*** *A function* $\mathscr{G}$ *is said to be equivariant to the action of a group* S *of transformations if*

$$\mathscr{G}(g(x)) = g(\mathscr{G}(x)) \quad \forall x \in \mathscr{X},\ g \in S. \tag{5.4}$$

The *equivariance* of $\mathscr{G}$ preserves the structure of transformations $g \in S$ of input data in the elements $y \in \mathscr{Y}$ (including, but not limited to, the case where $\mathscr{X} \equiv \mathscr{Y}$). In the case of functions represented as parameterized neural networks, the notion of invariance and equivariance can be expressed as:

$$\text{Invariant network:} \quad \mathscr{G}(g(x);\theta) = \mathscr{G}(x;\theta) \quad \forall x \in \mathscr{X},\ g \in S,\ \theta \in \mathbb{R}^p. \tag{5.5}$$

$$\text{Equivariant network:} \quad \mathscr{G}(g(x);\theta) = g(\mathscr{G}(x;\theta)) \quad \forall x \in \mathscr{X},\ g \in S,\ \theta \in \mathbb{R}^p. \tag{5.6}$$

An example of desired invariance could be rotation invariant classification of images or pointclouds, i.e. the neural network produces the same classification label, irrespective of the orientation of input. An instance of desired equivariance could be rotation equivariant segmentation, i.e. the predicted segmentation map should be rotated exactly in the same way as the rotated input.

5.3 PROPOSED APPROACH

Our idea is straightforward. We make neural networks invariant by consistently selecting a fixed element from the orbit of group transformations, i.e., we modify the input pose such that every element from the orbit of transformations maps to the same canonical element. For example, different rotated versions of an image are mapped to have the same orientation as visualized in Fig. 5.2. In conjunction with such *orbit mapping*, any standard network architecture can achieve provable invariance. In the following, we formalize our approach to achieve invariance.

5.3.1 *Invariant Networks w.r.t. Group Actions*

A basic observation for constructing invariant networks is that any network acting on the orbit of the input is automatically invariant to transformations in S:

Fact 1. ***Characterization of Invariant Functions via the Orbit:*** *Let S define a group action on $\mathscr{X}$. A network $\mathscr{G} : \mathscr{X} \times \mathbb{R}^p \to \mathscr{Y}$ is invariant under the group action of S if and only if it can be written as $\mathscr{G}(x;\theta) = \mathscr{G}_1(S \cdot x;\theta)$ for some other network $\mathscr{G}_1 : 2^{\mathscr{X}} \times \mathbb{R}^p \to \mathscr{Y}$.*

The above observation is based on the fact that $S \cdot x = S \cdot g(x)$ holds for any $g \in S$, provided that S is a group. Although not taking the general perspective of Fact 1, approaches, like (Laptev et al., 2016), which integrate (or sum over finite elements of) the mappings of $\mathscr{G}$ over a (discrete) group can be interpreted as instances of Fact 1 where $\mathscr{G}_1$ corresponds to the summation. Similar strategies of applying all transformations in S to the input x can be pursued for the design of equivariant networks. We now show that equivariant networks can be designed by applying all transformations in S to the input x.

Proposition 1. ***Characterization of Equivariant Functions via the Orbit:*** *Let S define a group action on $\mathscr{X}$. A network $\mathscr{G}$ is equivariant under the group action of S if it can be written as*

$$\mathscr{G}(x;\theta) = \mathscr{G}_1(\{g(\mathscr{G}_2(g^{-1}(x);\theta_2)) \mid g \in S\};\theta_1) \tag{5.7}$$

for some other arbitrary network $\mathscr{G}_2 : \mathscr{X} \times \mathbb{R}^{p_2} \to \mathscr{X}$, and a network $\mathscr{G}_1 : 2^{\mathscr{X}} \times \mathbb{R}^{p_1} \to \mathscr{X}$ that commutes with any element $h \in S$, i.e., for $h \in S$, and $Z \subset \mathscr{X}$, it satisfies $\mathscr{G}_1(h(Z);\theta_2) = h(\mathscr{G}_1(Z;\theta_2))$, where $h(Z)$ denotes the set obtained by the applying h to every element of Z.

Proof. We want to show that a network satisfying the condition (5) is equivariant. Let $h \in S$ be arbitrary. Note that

$$\{g \mid g \in S\} = \{h^{-1}g \mid g \in S\} \tag{5.8}$$

such that a substitution of variables from $g \in S$ to $z = h^{-1}g \in S$ (i.e., $g = hz$ and $z^{-1} = g^{-1}h$) yields

$$\begin{aligned}&\{g(\mathscr{G}_2(g^{-1}(h(x));\theta_2)) \mid g \in S\}\\ =&\{h(z(\mathscr{G}_2(z^{-1}(x);\theta_2))) \mid z \in S\}.\end{aligned}$$

This means that we can also write

$$\begin{aligned}\mathscr{G}(h(x);\theta) &= \mathscr{G}_1(\{h(z(\mathscr{G}_2(z^{-1}(x);\theta_2))) \mid z \in S\};\theta_1)\\ &= \mathscr{G}_1(h(\{z(\mathscr{G}_2(z^{-1}(x);\theta_2)) \mid z \in S\});\theta_1)\\ &= h(\mathscr{G}_1(\{z(\mathscr{G}_2(z^{-1}(x);\theta_2)) \mid z \in S\});\theta_1)\\ &= h(\mathscr{G}(x;\theta))\end{aligned}$$

which yields the desired equivariance under the assumed commutative property. □

The work Cohen and Welling (2016) can be interpreted as an instance of the construction in proposition 1, where equivariant linear layers w.r.t. rotations by 90 degrees are obtained by choosing $\mathscr{G}_2$ to be a simple convolution and $\mathscr{G}_1$ to be the summation over all (finitely many) elements of the set. Subsequently, they nest these layers with component-wise (and therefore inherently equivariant) non-linearities.

5.3.2 *Orbit Mappings*

While Fact 1 and proposition 1 are stated for general (even infinite) groups, realizations of such constructions from the literature often assume a finite orbit. In this chapter, we develop an efficient solution even for cases in which the orbit is not finite, and utilize Fact 1 in the most straightforward way: We propose to construct provably invariant networks $\mathscr{G}(x;\theta) = \mathscr{G}_1(S \cdot x;\theta)$ by simply using an

$$\textit{orbit mapping } h : \{S \cdot x \mid x \in \mathscr{X}\} \to \mathscr{X},$$

which uniquely selects a particular element from an orbit as a first layer in $\mathscr{G}_1$. Subsequently, we can proceed with any standard network architecture and Fact 1 still guarantees the desired invariance. A key in designing instances of orbit mappings is that they should not require enlisting all elements of $S \cdot x$ in order to evaluate $h(S \cdot x)$. Let us provide more concrete examples of orbit mappings.

Example 1 (Mean-subtraction)**.** *A common approach in data classification tasks is to first normalize the input by subtracting its mean. Considering $\mathscr{X} = \mathbb{R}^n$ and $S = \{g : \mathbb{R}^n \to \mathbb{R}^n \mid g(x) = x + a\mathbb{1},$ for some $a \in \mathbb{R}\}$, with $\mathbb{1} \in \mathbb{R}^n$ being a vector of all ones, input-mean-subtraction is an orbit mapping that selects the unique element from any $S \cdot x$ which has zero mean.*

Example 2 (Permutation invariance via sorting)**.** *Consider $\mathscr{X} = \mathbb{R}^n$, and S to be all permutations of vectors in $\mathbb{R}^n$, i.e., $S = \{s \in \{0,1\}^{n\times n} \mid \sum_i s_{i,j} = 1 \; \forall j, \;\; \sum_j s_{i,j} = 1 \; \forall i\}$. We could define an orbit mapping that selects the element from an orbit whose entries are sorted by magnitude in an ascending order.*

With the very natural condition that orbit mappings really select an element from the orbit, i.e., $h(S \cdot x) \in S \cdot x$, we can readily construct equivariant networks by applying the inverse mapping, see Appendix A. In our Example 2, undoing the sort operation at the end of the network allows to transfer from an invariant to an equivariant network.

As a final note, our concept of orbit mappings can further be generalized by h not mapping to the input space $\mathscr{X}$, but to a different representation, which can be beneficial for particular, complex groups. In geometry processing, for instance, an important group action is isometric deformations of shapes. A common strategy to handle these (c.f. (Ovsjanikov et al., 2012)) is to identify any shape with the eigenfunctions of its Laplace-Beltrami operator (Pinkall and Polthier, 1993), which represents a natural (generalized) orbit mapping. We refer to (Litany et al., 2017; Eisenberger et al., 2020; Huang et al., 2019) for exemplary deep learning applications.

5.4 APPLICATIONS

We will now present two specific instances of orbit mappings for handling continuous rotations of images as well as for invariances in 3D point cloud classification.

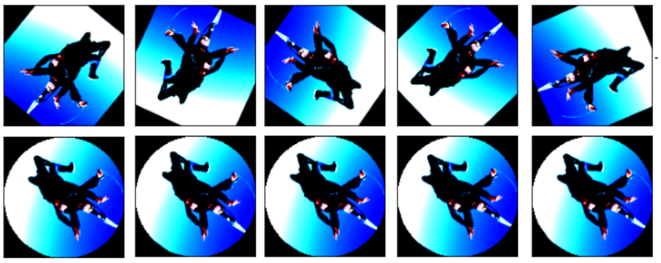


Figure 5.2: Images of different orientations (top) are consistently aligned with the proposed gradient-based orbit mapping (bottom).

5.4.1 *Invariance to continous image rotations*

Images as functions Let us consider the important example of invariance to continuous rotations of images. To do so, consider $\mathscr{X} \subset \{u : \Omega \subset \mathbb{R}^2 \to \mathbb{R}\}$ to represent images as functions. For the sake of simplicity, we consider grayscale images only, but this extends to color images in a straightforward way. In our notation $z \in \mathbb{R}^2$ represents spatial coordinates of an image (to avoid an overlap with our previous $x \in \mathscr{X}$, which we used for the input of a network). We set

$$\begin{aligned} &S = \{g : \mathscr{X} \to \mathscr{X} \mid g \circ u(z) = u(r(\alpha)z), \text{ for } \alpha \in \mathbb{R}\}, \\ &\text{and } r(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}. \end{aligned} \tag{5.9}$$

As S has infinitely many elements, approaches that worked well for rotations by 90 degrees like (Cohen and Welling, 2016) are not applicable anymore. We instead propose to uniquely select an element from the continuous orbit of rotation $g \in S$ by choosing a rotation that makes the average gradient of the image $\int_Z \nabla(g \circ u)(z)\; dz$ over a suitable set Z, e.g. a circle around the image center point upwards. It holds that

$$\begin{aligned} \nabla(g \circ u)(z) = r^T(\alpha)\nabla u(r(\alpha)z) \text{ such that} \\ \int_Z \nabla(g \circ u)(z) dz = \int_Z r^T(\alpha)\nabla u(r(\alpha)z)\, dz. \end{aligned}$$

Substituting $\varphi = r(\alpha)z$, we obtain

$$\int_Z r^T(\alpha)\nabla u(r(\alpha)z)\, dz = \int_{r^T(\alpha)Z} r^T(\alpha)\nabla u\big(\varphi\big)\, d\varphi = r^T(\alpha)\int_Z \nabla u\big(\varphi\big)\, d\varphi \tag{5.10}$$

where we used that Z is rotationally invariant. Thus, choosing a rotation that makes $\int_Z \nabla(g \circ u)(z)\; dz$ point upwards is equivalent to solving

$$r(\hat{\alpha}) = \text{argmax}_{r(\alpha)} \left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, r^T(\alpha)\int_Z \nabla u(\varphi)\; d\varphi \right\rangle \tag{5.11}$$

whose solution is given by $\hat{\alpha}$ such that

$$\begin{pmatrix} \cos\hat{\alpha} \\ \sin\hat{\alpha} \end{pmatrix} = \left(\frac{\int_Z \nabla u(z)\; dz}{\| \int_Z \nabla u(z)\; dz \|} \right). \tag{5.12}$$

Note that eq. (5.12) yields a unique solution to the maximization problem. Since a consistent pose is always selected[1], it is an invariant mapping. When $\int_Z \nabla u(z)\ dz = 0$, any $g \in S$ maximizes eq. (5.11). However, numerically $\int_Z \nabla u(z)\ dz$ rarely evaluates to exact zero, and its magnitude determines the stability of orbit mapping.

DISCRETIZATION For a discrete (grayscale) image given a matrix $\tilde{u} \in \mathbb{R}^{n_y \times n_x}$, we first apply Gaussian blur with a standard deviation of $\sigma = 1.5$ (to reduce the effect of noise and create a smooth image), and subsequently construct an underlying continuous function $u : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ by bilinear interpolation. For the set Z we choose two circles of radii 0.05 and 0.4 (for Ω being normalized to $[0,1]^2$). We approximate the integral by a sum over finite evaluations of the derivative along each circle, using exact differentiation of the continuous image model. This strategy can stabilize arbitrary rotations successfully as illustrated in Fig. 5.2. However, in practice, the magnitude of $\int_Z \nabla u(z)\ dz$ and interpolation artifacts affect the stability of the orbit mapping. We analyze the stability of the proposed gradient based orbit-mapping for discrete images in Appendix C, where we observe that the use of forward or central differences to approximate gradients further deteriorates the stability of orbit mapping. Since the orbit mapping for discrete images has instabilities, exact invariance to rotations cannot be guaranteed. Even when the integral values are large leading to a stable orbit mapping, our approach does not need to give the same rotation angle for semantically similar content, for example, different cars are not necessarily rotated to have the same orientation. Due to these reasons, our approach can further benefit from augmentation.

Stability of gradient based orbit mapping

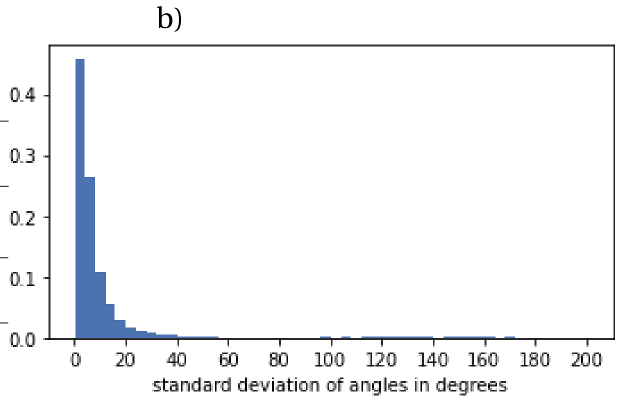
We now analyze the stability of our gradient based orbit mapping strategy for discrete images. While the proposed gradient based orbit mapping our approach leads to unique orientation as long as $\int_Z \nabla u(z)\ dz$ is non-zero, practically, the magnitude of $\int_Z \nabla u(z)\ dz$ and interpolation artifacts affect the stability of the orbit mapping. While one could possibly use forward or central differences to calculate gradients at pixels along approximate circles, this further deteriorates the stability of orbit mapping. This is seen in Tab. 5.1 a) which shows the mean standard deviation orientation of orbit-mapped images when input images rotated in steps of 1 degree using bilinear interpolation. We find that using forward differences to approximate the gradient has the most instability. In the section 5.C of the appendix, we derive a necessary condition for provable invariance using general convolution kernels (instead of gradients in x and y direction), where we show that forward differences do not satisfy these conditions for any rotation.

Tab. 5.1 b) shows the histogram of standard deviations in orientation for CIFAR10 images when calculating exact gradients along the circle. The standard deviation of predicted orientations of over 78% of the images is less than 10 degrees, and over 44% of images is less than 4 degrees, indicating a relatively stable orbit mapping for these images. However, a fraction of images also have a higher variance, in predicted orientation possibly due to small values of the integral. Tab. 5.1 c) shows that our gradient based orbit mapping is fairly robust to small additive Gaussian noise.

[1] Note that $r^T(\alpha) = r(-\alpha)$, therefore if the predicted rotation for $u(z)$ is β, then for $u(r(\gamma)z)$, it is $\beta - \gamma$, i.e the same element is consistently selected.

a)

Dataset	Exact	Central Diff.	Forward Diff
CIFAR10	10.46	12.47	23.89
CUB200	9.05	14.56	24.75



c)

Dataset	clean	σ^2=0.01	σ^2=0.05	σ^2=0.1
CIFAR10	10.46	11.36	14.08	16.69
CUB200	9.05	10.55	15.99	20.610

Table 5.1: Stability and robustness of proposed gradient based Orbit Mapping strategy. a) The mean standard deviation values of angles in degrees over the images in the dataset are reported when rotating images based on exact gradients computed along the circle using bilinear interpolation, and approximate gradients using finite differences along pixels closest to the circle. b) The histogram of standard deviations of the predicted orientation in degrees for CIFAR10. c) The mean standard deviation values of angles in degrees over the images in the CIFAR10 dataset are reported, for different levels of additive Gaussian noise.

Experiments

To evaluate our approach, we use orbit mapping in conjunction with image classification networks on three datasets: On CIFAR10, we train a Resnet-18 (He et al., 2016) from scratch. On the HAM10000 skin image dataset (Tschandl et al., 2018), we finetune an NFNet-F0 network (Brock et al., 2021), and on CUB-200 (Wah et al., 2011) we finetune a Resnet-50 (He et al., 2016), both pretrained on ImageNet. While the datasets CIFAR10 and CUB-200 have an inherent variance in orientation, for the HAM10000 skin lesion classification, exact rotation invariance is desirable. Finally, we also perform experiments with RotMNIST using state-of-the-art E2CNN network (Weiler and Cesa, 2019). The details of the protocol used for training all our networks as well as some additional experiments are provided in the Appendix E. We compare with following approaches on CIFAR10, HAM10000, and CUB-200:

(i) *Adversarial training*: $\min_{\theta} \sum_{\text{examples i}} \mathcal{L}(\mathcal{G}(\hat{x}^i;\theta);y^i)$, for $\hat{x}^i = \arg\max_{z\in S\cdot x^i} \mathcal{L}(\mathcal{G}(z);y^i)$. This is approximated by selecting the worst out of 10 different random rotations for each image in every iteration, following (Engstrom et al., 2019). It is referred to as Adv. in Tab. 5.2.

(ii) *Mixed mode training*: $\min_{\theta} \mathcal{L}_{\text{examples i}} (\mathcal{G}(\hat{x}^i;\theta);y^i) + \mathcal{L}(\mathcal{G}(x^i;\theta);y^i)$ which uses both natural and adversarial examples $\hat{x}^i$ (Yang et al., 2019).

(iii) *Adversarial training with regularization:* Use of adversarial logit pairing and KL-divergence regularizers (Yang et al., 2019) along with adversarial training (indicated as Adv.-ALP and Adv.-KL in Tab. 5.2):

   (a) *Adversarial logit pairing (ALP)*: $R_{ALP}(\mathcal{G},x^i,y^i) = \|\mathcal{G}(x^i;\theta) - \mathcal{G}(\hat{x}^i;\theta)\|_2^2$

   (b) *KL-divergence*:$R_{KL}(\mathcal{G},x^i,y^i) = D_{KL}(\mathcal{G}(x^i;\theta)||\mathcal{G}(\hat{x}^i;\theta))$

(iv) *Transformation invariant pooling (TIpool):* which is a provably invariant approach for discrete rotations (Laptev et al., 2016), where the features of multiple rotated copies of an input image are pooled before the final classification. We use 4 rotated copies of images rotated in multiples of 90 degrees.

Method	OM$^{\text{(Ours)}}$	CIFAR10			HAM10000			CUB200		
		Clean	Avg.	Worst	Clean	Avg.	Worst	Clean	Avg.	Worst
Std.	✗	**93.98**	40.06	1.31	93.82	91.73	82.52	**77.41**	53.45	8.07
	✓ Train+Test	87.99	84.12	68.60	93.31	91.38	87.96	71.19	71.56	58.80
RA	✗	85.54	75.99	44.71	93.30	90.81	82.30	69.89	70.12	41.01
	✓ Train+Test	85.40	81.82	71.09	93.41	92.13	88.55	70.35	70.72	57.54
STN	✗	83.74	78.86	54.03	–	–	–	–	–	–
ETN	✗	84.39	80.30	64.08	92.47	90.85	84.32	64.14	66.95	52.85
Adv.	✗	69.32	68.54	50.21	92.28	91.87	85.04	64.54	64.07	42.82
Mixed	✗	91.15	68.37	17.15	93.71	92.13	84.53	68.56	65.91	42.87
Adv.-KL	✗	72.28	70.29	51.05	92.54	91.79	85.42	64.47	64.65	43.04
Adv.-ALP	✗	71.25	70.30	52.29	92.89	91.84	85.98	64.63	64.34	43.63
TIpool	✗	93.56	66.46	20.22	93.19	91.87	88.16	76.80	74.90	59.04
	✓ Train+Test	91.94	**88.77**	76.26	**93.83**	92.05	**89.81**	76.82	**77.18**	**69.19**
TIpool-RA	✗	91.40	84.65	67.28	93.39	91.87	88.12	73.47	74.71	62.82
	✓ Train+Test	90.47	87.92	**80.07**	93.68	**92.78**	89.30	74.78	75.89	67.78

Table 5.2: Comparison of orbit mapping *(OM)* with training and architecture based methods. Robustness to rotations is compared using the average and worst case accuracies over 5 runs with test images rotated in steps of 1° using bilinear interpolation.

(v) *Spatial transformer networks (STN):* which learns to undo the transformation by training using appropriate data augmentation (Jaderberg et al., 2015).

(vi) *Equivariant transformer networks (ETN):* which additionally uses appropriate coordinate transformation along with a learned spatial transformer to undo the transformation (Tai et al., 2019).

We also compare with the simple baseline of augmenting with random rotations, referred to as RA in Tab. 5.2. Additionally, we also compare with (Benton et al., 2020), an approach which learns the distribution of augmentations on the task of rotated CIFAR10 classification, referred to as Augerino in Tab. 5.3. We use 4 samples from the learned distribution of augmentations during both training and test. We would also like to point out that adversarial training using the worst of 10 samples roughly increases the training effort of the underlying model by a factor of 5.

Results

We measure the accuracy on the original testset(*Clean*), as well as the average (*Avg.*) and worst-case (*Worst*) accuracies in the orbit of rotations discretized in steps of 1 degree, where '*Worst*' counts an image as misclassified as soon as there exists a rotation at which the network makes a wrong prediction.

As we can see in Tab. 5.2, networks trained without rotation augmentation perform poorly in terms of both, the average and worst-case accuracy if the data set contains an inherent orientation. While augmenting with rotations during training results in improvements, there is still a huge gap (∼ 30% for CIFAR10 and CUB200) between the average and worst-case accuracies. While adversarial training approaches (Engstrom et al., 2019; Yang et al., 2019) improve the performance in the worst case, there is a clear drop in the clean and average accuracies when compared to data augmentation. Learned approaches to correct orientation i.e. STN (Jaderberg et al., 2015), ETN (Tai et al., 2019) show an improvement over adversarial training schemes in terms of average and worst case accuracies, when training from scratch, with ETN demonstrating even higher robustness than plain STNs. While pooling over features of rotated versions of image provides provable invariance to

Train	OM	Clean	Average			Worst-case		
			Nearest	Bilinear	Bicubic	Nearest	Bilinear	Bicubic
Std.	✗	**93.98±0.32**	35.12±0.81	40.06±0.44	42.81±0.50	0.79±0.38	1.31±0.13	2.22±0.17
	✓ Train+Test	87.99±0.43	72.40±0.33	84.12±0.55	86.61±0.49	34.57±0.94	68.60±0.81	74.49±0.84
RA	✗	85.54±0.72	80.47±0.74	75.99±0.72	79.47±0.65	45.50±0.83	44.71±0.74	50.50±0.78
	✓ Test	79.26±0.42	74.93±0.51	69.31±0.65	73.94±0.63	48.93±0.75	52.18±0.91	58.69±0.78
	✓ Train+Test	85.40±0.57	84.37±0.58	81.82±0.59	84.82±0.52	66.22±0.75	71.09±1.01	76.44±0.89
RA-combined	✗	92.42±0.21	80.90±0.64	82.23±0.74	82.71±0.69	36.98±1.27	48.07±1.66	49.51±1.47
	✓ Test	82.55±0.86	76.33±0.95	77.93±0.68	78.42±0.64	45.44±1.32	60.23 ±1.24	62.18±1.33
	✓ Train+Test	86.69±0.12	84.06±0.21	85.27±0.23	86.06±0.20	61.75±0.76	75.29±0.42	77.25±0.27
Adv.	✗	69.32±1.61	61.73±1.12	68.54±0.68	68.00±0.31	36.95±0.97	50.21±0.55	49.73±0.98
Mixed	✗	91.15±0.15	54.55±0.40	68.37±0.66	68.48±0.37	3.86±0.13	17.15±1.25	16.85±0.93
Adv.-KL	✗	72.28±2.05	62.60±1.72	70.29±1.42	69.84±1.29	32.60±0.74	51.05±2.47	51.11±1.03
Adv.-ALP	✗	71.25±0.97	62.36±2.19	70.30±1.50	69.71±1.22	33.98±1.44	52.29±1.76	52.57±1.57
STN	✗	83.74±0.50	81.94±0.51	78.86±0.73	82.21±0.55	51.23±1.01	54.03±1.36	59.65±1.31
ETN	✗	84.39±0.09	82.98±0.28	80.30±0.55	83.31±0.31	59.40±0.76	64.08±0.78	68.75±0.83
Augerino	✗	83.68±0.76	80.17±0.70	82.27±0.69	81.69±0.72	52.44±0.66	60.36±1.00	60.63±0.94
TIpool	✗	93.56±0.25	55.96±0.39	66.46±1.36	70.70±0.77	3.14±1.09	20.22±1.51	27.88±1.09
TIpool-RA	✗	91.40±0.17	87.50±0.24	84.65±0.51	87.31±0.29	66.52±1.31	67.28±1.03	72.35±0.83
TIpool	✓Train+Test	91.94±0.38	78.66±0.83	88.77±0.51	**90.76±0.40**	42.01±1.07	76.26±1.12	81.46±1.02
TIpool-RA	✓Train+Test	90.47±0.36	**89.37±0.36**	87.92±0.36	89.91±0.34	**74.51±0.79**	80.07±0.69	83.76±0.60
TIpool-RA combined	✓Train+Test	91.09±0.40	89.02±0.30	**90.13±0.34**	90.64±0.30	70.18±1.12	**82.71±0.62**	**84.26±0.41**

Table 5.3: Effect of augmentation on robustness to rotations with different interpolations. Shown are clean accuracy on standard CIFAR10 test set, average and worst-case accuracies on rotated test set with mean and standard deviations over 5 runs.

discrete rotations, this approach is still susceptible to continuous image rotations. The robustness of this approach to continuous rotations is boosted by rotation augmentation, with improvements over even learned transformers. Note that using TI-pooling with 4 rotated copies increases the computation by 4 times. In contrast, our orbit mapping effortlessly leads to significant improvements in robustness even without augmenting with rotations, with performance better than adversarial training, learned transformers, and discrete invariance based approaches. Since our orbit mapping for discrete images has some instabilities, our approach also benefits from augmentation with image rotations. Further, when combined with the discrete invariant approach (Laptev et al., 2016), we obtain the best accuracies for average and worst case rotations.

Even when finetuning networks, we observe that orbit mapping readily improves robustness to rotations over standard training, even without the use of augmentations. Furthermore, the combination of orbit mapping with the discrete invariant approach of pooling over rotated features yields the best performance. For the birds dataset with inherent orientation, undoing rotations using ETN significantly improves robustness when compared to adversarial training schemes, which only marginally improve robustness over rotation augmentation. We found it difficult to train STN with higher accuracies (*Clean*/*Avg.*/*Worst*) than plain augmentation with rotated images for CUB200 and HAM10000, despite extensive hyperparameter optimization, therefore we do not report the numbers here[2]. When the data itself does not contain a prominent orientation as in the HAM10000 data set, the general trend in accuracies still holds (*Clean*>*Avg.*>*Worst*), but the drops in accuracies are not drastic, and adversarial training schemes provide improvements over undoing transormations using ETN. Further, orbit mapping and pooling over rotated images provide comparable improvements in robustness, with their combination achieving the best results.

[2] We use a single spatial transformer as opposed to multiple STNs used in (Jaderberg et al., 2015) and train on randomly rotated images.

Train.	OM	D4/C4			D16/C16		
		Clean	Avg.	Worst	Clean	Avg.	Worst
Std.	✗	98.73±0.04	98.61±0.04	96.84±0.08	99.16±0.03	99.02±0.04	98.19±0.08
Std.	✓(Train+Test)	98.86±0.02	98.74±0.03	98.31±0.05	99.21±0.01	99.11±0.03	98.82±0.06
RA.	✗	99.19±0.02	99.11±0.01	98.39±0.05	99.31±0.02	99.27±0.02	98.89±0.03
RA.	✓(Train+Test)	98.99±0.03	98.90±0.01	98.60±0.02	99.28±0.02	99.23±0.01	99.04±0.02

Table 5.4: Effect of orbit mapping and rotation augmentation on RotMNIST classification using regular D4/C4 and D16/C16 E2CNN models. Shown are clean accuracy on standard test set and average and worst-case accuracies on test set rotated in steps of 1 degree, with mean and standard deviations over 5 runs.

DISCRETIZATION ARTIFACTS: It is interesting to see that while consistently selecting a single element from the continuous orbit of rotations leads to provable rotational invariance when considering images as continuous functions, discretization artifacts, and boundary effects still play a crucial role in practice, and rotations cannot be fully stabilized. As a result, there is still a discrepancy between the average and worst case accuracies, and the performance is further improved when our approach also uses rotation augmentation. Motivated by the strong effect the discretization seems to have, we investigate different interpolation schemes used to rotate the image in more detail: Tab. 5.3 shows the results of different training schemes with and without our orbit mapping (*OM*) obtained with a ResNet-18 architecture on CIFAR-10 when using different types of interpolation. Besides standard training (*Std.*), we use rotation augmentation (*RA*) using the Pytorch-default of nearest-neighbor interpolation, a combined augmentation scheme (*RA-combined*) that applies random rotation only to a fraction of images in a batch using at least one nearest neighbor, one bilinear and one bicubic interpolation. The adversarial training and regularization from (Engstrom et al., 2019; Yang et al., 2019) are trained using bilinear interpolation (following the authors' implementation).

Results show that interpolation used in image rotation impacts accuracies in all the baselines. Most notably, the worst-case accuracies between different types of interpolation may differ by more than 20%, indicating a huge influence of the interpolation scheme. Adversarial training with bi-linear interpolation still leaves a large vulnerability to image rotations with nearest neighbor interpolation. Further, applying an orbit mapping at test time to a network trained with rotated images readily improves its worst case accuracy, however, there is a clear drop in clean and average case accuracies, possibly due to the network not having seen doubly interpolated images during training. While our approach without rotation augmentation is also vulnerable to interpolation effects, it is ameliorated when using orbit mapping along with rotation augmentation. We observe that including different augmentations (RA-combined) improves the robustness significantly. Combining the orbit mapping with the discrete invariant approach (Laptev et al., 2016) boosts the robustness, with different augmentations further reducing the gap between clean, average case, and worst case performance.

EXPERIMENTS WITH ROTMNIST We investigate the effect of orbit mapping on RotMNIST classification. This dataset has 12000 training and 50000 test samples of randomly rotated MNIST digits. We use the state of the art network from (Weiler and Cesa, 2019) employing regular steerable equivariant models (Weiler et al., 2018a). This model uses 16 rotations and flips of the learned filters (with flips being restricted till layer3). We also compare with a variation of the same architecture with 4 rotations. We refer to these models as D16/C16 and D4/C4 respectively. We train and evaluate these models using their publicly available code[3].

[3] code url https://github.com/QUVA-Lab/e2cnn_experiments

Augment.	Unscaling	with STN			without STN		
		Clean	Avg.	Worst	Clean	Avg.	Worst
[0.8, 1.25]	✗	86.15± 0.52	24.40±1.56	0.01±0.02	85.31±0.39	33.57±2.00	2.37±0.06
[0.8, 1.25]	✓(Train+Test)	**86.15± 0.28**	**86.15± 0.28**	**86.15± 0.28**	85.25±0.43	85.25±0.43	85.25±0.43
[0.8, 1.25]	✓(Test)	86.15± 0.52	85.59±0.79	85.59±0.79	85.31±0.39	83.76±0.35	83.76±0.35
[0.1, 10]	✗	85.40±0.46	47.25±1.36	0.04±0.05	75.34±0.84	47.58±1.69	1.06±0.87
[0.1, 10]	✓(Test)	85.40±0.46	85.85±0.73	85.85±0.73	75.34±0.84	81.45±0.56	81.45±0.56
[0.001, 1000]	✗	33.33± 7.58	42.38± 1.54	2.25±0.22	5.07±2.37	25.42±0.73	2.24±0.11
[0.001, 1000]	✓(Train+Test)	85.66± 0.39	85.66± 0.39	85.66± 0.39	85.05±0.43	85.05±0.43	85.05±0.43

Table 5.5: Scaling invariance in 3D pointcloud classification with PointNet trained on modelnet40, with and without data augmentation, with and without STNs or scale normalization. Mean and standard deviations over 10 runs are reported.

RA	STN	PCA	Clean	Rotation		Translation	
				Avg.	Worst	Avg.	Worst
✗	✓	✗	**86.15±0.52**	10.37±0.18	0.09±0.07	10.96±1.22	0.00±0.00
✗	✗	✗	85.31±0.39	10.59±0.25	0.26±0.10	6.53±0.12	0.00±0.00
✗	✓	✓(Train+Test)	74.12± 1.80	74.12± 1.80	74.12± 1.80	74.12± 1.80	74.12± 1.80
✗	✗	✓(Train+Test)	75.36±0.70	**75.36±0.70**	**75.36±0.70**	**75.36±0.70**	**75.36±0.70**
✓	✓	✗	72.13± 5.84	72.39± 5.60	35.91± 4.87	5.35±0.98	0.00±0.00
✓	✗	✗	63.93±0.65	64.75±0.57	45.53±0.29	3.90±0.71	0.00±0.00
✓	✓	✓(Test)	72.13± 5.84	72.96± 5.85	72.96± 5.85	72.96± 5.85	72.96± 5.85
✓	✗	✓(Test)	64.56±0.91	64.56±0.91	64.56±0.91	64.56±0.91	64.56±0.91
✓	✓	✓(Train+Test)	72.84±0.77	72.84±0.77	72.84±0.77	72.84±0.77	72.84±0.77
✓	✗	✓(Train+Test)	74.84±0.86	74.84±0.86	74.84±0.86	74.84±0.86	74.84±0.86

Table 5.6: Rotation and translation invariances in 3D pointcloud classification with PointNet trained on modelnet40, with and without rotation augmentation, with and without STNs or PCA. Mean and standard deviations over 10 runs are reported.

Results in Tab. 5.4 indicate that even for these state-of-the-art models, there is a discrepancy between the accuracy on the standard test set and the worst case accuracies, and their robustness can be further improved by orbit mapping. Notably, orbit mapping significantly improves worst case accuracy (by around 1.5%) for D4/C4 steerable model trained without augmenting using rotations, showing gains in robustness even over naively trained D16/C16 model of much higher complexity. Training with augmentation leads to improvement in robustness, with orbit mapping providing gains further in robustness. However, artifacts due to double interpolation affect the performance of orbit mapping.

5.4.2 *Invariances in 3D Point Cloud Classification*

Invariance to orientation and scale is often desired in networks classifying objects given as 3D point clouds. Popular architectures, such as PointNet (Qi et al., 2017a) and its extensions (Qi et al., 2017b), rely on the ability of spatial transformer networks to learn such invariances by training on large datasets and extensive data augmentations. We analyze the robustness of these networks to transformations with experiments using Pointnet on *modelnet40* dataset (Wu et al., 2015). We compare the class accuracy of the final iterate for the clean validation set *(Clean)*, and transformed validation sets in the average *(Avg.)* and worst-case *(Worst)*. We show that PointNet performs better with our orbit mappings than with augmentation alone.

In this setting, $\mathcal{X} = \mathbb{R}^{d \times N}$ are N many d-dimensional coordinates (usually with $d = 3$). The desired group actions for invariance are left-multiplication with a rotation matrix, and multiplication with any number $c \in \mathbb{R}^+$ to account for different scaling. We also consider translation by adding a fixed coordinate $c_t \in \mathbb{R}^3$ to each entry in $\mathcal{X}$. Desired invariances in point cloud classification range from class-dependent variances to geometric properties. For example, the classification of airplanes should be invariant to the specific wing shape, as well as the scale or translation of the model. While networks can learn some invariance

Augmentation			STN	OM	Clean	Scaling		Rotation		Translation	
Scale	RA	Translation		All		Avg.	Worst	Avg.	Worst	Avg.	Worst
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✓	✗	72.13± 5.84	19.74± 4.01	0.16± 0.42	72.39± 5.60	35.91± 4.87	5.35±0.98	0.00±0.00
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✓	✓ Test	67.38± 7.96	64.88± 12.16	64.88± 12.16	64.88± 12.16	64.88± 12.16	64.88± 12.16	64.88± 12.16
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✓	✓ Train+Test	**77.52±1.03**	**77.52±1.03**	**77.52±1.03**	**77.52±1.03**	**77.52±1.03**	**77.52±1.03**	**77.52±1.03**
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✗	✗	63.93±0.65	12.85±0.29	0.27±0.55	64.75±0.57	45.53±0.29	3.90±0.71	0.00±0.00
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✗	✓Test	64.71±0.92	57.10±1.14	57.10±1.14	57.10±1.14	57.10±1.14	57.10±1.14	57.10±1.14
$[0.8, 1.25]$	✓	$[-0.1, 0.1]$	✗	✓Train+Test	74.41±0.58	74.41±0.58	74.41±0.58	74.41±0.58	74.41±0.58	74.41±0.58	74.41±0.58

Table 5.7: Combined Scale, rotation and translation invariances in 3D pointcloud classification with PointNet trained on modelnet40, with data augmentation and analytical inclusion of each invariance. Mean and standard deviations over 10 runs are reported.

from training data, our experiments show that even simple transformations like scaling and translation are not learned robustly outside the scope of what was provided in the training data, see Tabs. 5.5, 5.6, 5.7. This is surprising, considering that both can be undone by centering around the origin and re-scaling.

SCALING: Invariance to scaling can be achieved in the sense of Sec. 5.3 by scaling input point-clouds by the average distance of all points to the origin. Our experiments show that this leads to robustness against much more extreme transformation values without the need for expensive training, both for average as well as worst-case accuracy. We tested the worst-case accuracy on the following scales: $\{0.001, 0.01, 0.1, 0.5, 1.0, 5.0, 10, 100, 1000\}$. While our approach performs well on all cases, training PointNet on random data augmentation in the range of possible values actually reduces the accuracy on clean, not scaled test data. This indicates that the added complexity of the task cannot be well represented within the network although it includes spatial transformers. Even when restricting the training to a subset of the interval of scales, the spatial transformers cannot fully learn to undo the scaling, resulting in a significant drop in average and worst-case robustness, see Tab. 5.5. While training the original Pointnet including the desired invariance in the network achieves the best performance, dropping the spatial transformers from the architecture results in only a tiny drop in accuracy with significant gains in training and computation time[4]. This either indicates that in the absence of rigid deformation, the spatial transformers do not add much knowledge and are strictly inferior to modeling invariance, at least on this dataset.

ROTATION AND TRANSLATION In this section, we show that 3D rotations and translations exhibit similar behavior and can be more robustly treated via orbit mapping than through data augmentation. This is even more meaningful than scaling as both have three degrees of freedom and sampling their respective spaces requires a lot more examples. For rotations, we choose the unique element of the orbit to be the rotation of $\mathscr{X}$ that aligns its principle components with the coordinate axes. The optimal transformation involves subtracting the center of mass from all coordinates and then applying the singular value decomposition $X = U\Sigma V$ of the point cloud X up to the arbitrary orientation of the principle axes, a process also known as PCA. Rotation and translation can be treated together, as undoing the translation is a substep of PCA, but Tab. 5.6 also shows separate results for both. To remove the sign ambiguity in the principle axes, we choose signs of the first row of U and encode them into a diagonal matrix D, such that the final transform is given by $\hat{X} = XV^\top D$. We apply this rotational alignment to PointNet with and without spatial transformers and evaluate its robustness to rotations in average-case and worst-case when rotating the validation dataset in 16 × 16 increments (i.e.

[4] Model size of PointNet with STNs is 41.8 MB, and without STNs 9.8 MB

with 16 discrete angles along each of the two angular degrees of freedom of a 3D rotation). We test robustness to translations in average-case and worst-case for the following shifts in each of *x*, *y* and *z* directions: $\{-10.0, -1.0, -0.5, -0.1, 0.1, 0.5, 1.0, 10.0\}$. Tab. 5.6 shows that PointNet trained without augmentation is susceptible in worst-case and average-case rotations and even translations. The vulnerability to rotations can be ameliorated in the average-case by training with random rotations, but the worst-case accuracy is still significantly lower, even when spatial transformers are employed. Also notable is the high variance in performance of Pointnets with STNs trained using augmentations. On the other hand, explicitly training and testing with stabilized rotations using PCA does provide effortless invariance to rotations and translations, even without augmentation. Interestingly, the best accuracy here is reached when training PointNet entirely without spatial transformers, which offer no additional benefits when the rotations are stabilized. The process for invariance against translation is well-known and well-used due to its simplicity and robustness. We show that this approach arises naturally from our framework, and that its extension to rotational invariance inherits the same numerical behavior, i.e., provable invariance outperforms learning to undo the transformation via data augmentation.

COMBINED INVARIANCE TO SCALING, ROTATION, TRANSLATION. Our approach can be extended to make a model simultaneously invariant to scaling, rotations and translations. In this setup, we apply a PCA alignment before normalizing the scale of input point cloud. Tab. 5.7 shows that PointNet trained with such combined orbit mapping does achieve the desired invariances.

5.5 DISCUSSION AND CONCLUSIONS

We proposed a simple and general way of incorporating invariances to group actions in neural networks by uniquely selecting a specific element from the orbit of group transformations. This guarantees provable invariance to group transformations for 3D point clouds, and demonstrates significant improvements in robustness to continuous rotations of images with a limited computational overhead. However, for images, a large discrepancy between the theoretical provable invariance (in the perspective of images as continuous functions) and the practical discrete setting remains. We conjecture that this is related to discretization artifacts when applying rotations that change the gradient directions, especially at low resolutions. Notably, such artifacts appear more frequently in artificial settings, e.g. during data augmentation or when testing for worst-case accuracy, than in photographs of rotating objects that only get discretized once. While we found a consistent advantage of enforcing the desired invariance via orbit mapping rather than training alone, a combination of data augmentation and orbit mappings yields additional advantages (in cases where discretization artifacts prevent a provable invariance of the latter). Moreover, our orbit mapping can be combined with existing invariant approaches for improved robustness.

APPENDIX

5.A EXTENSION OF ORBIT MAPPING TO EQUIVARIANT NETWORKS

Proposition 2 (Orbit mapping for equivariant networks)**.** *Let h be an orbit mapping that satisfies $h(S\cdot x)\in S\cdot x$ for all x. Any network $\mathscr{G}:\mathscr{X}\times\mathbb{R}^p\to\mathscr{X}$ that can be written as*

$$\mathscr{G}(x;\theta)=\hat{g}^{-1}(\mathscr{G}_2(\hat{g}(x);\theta)) \tag{5.13}$$

for an arbitrary network $\mathscr{G}_2:\mathscr{X}\times\mathbb{R}^p\to\mathscr{X}$ and $\hat{g}\in S$ denoting the element that satisfies $\hat{g}(x)=h(S\cdot x)$ is equivariant.

Proof. We want to show that a network satisfying the condition (5.13) is equivariant. Consider an input $a=r(x)$ to the network, where r denotes an arbitrary element of S. We first need to determine the element $\tilde{g}\in S$ such that $\tilde{g}(a)=h(S\cdot a)$. From the definition of the orbit, it follows that $S\cdot x=S\cdot r(x)$, such that our orbit mapping satisfies remains the same, i.e., $h(S\cdot x)=h(S\cdot a)=\hat{g}(x)$. Solving the equation $\tilde{g}(a)=\hat{g}(x)$ with $a=r(x)$, i.e., $x=r^{-1}(a)$ for $\tilde{g}$ yields $\tilde{g}=\hat{g}r^{-1}$. Now it follows that

$$\begin{aligned}\mathscr{G}(r(x);\theta)=\mathscr{G}(a;\theta)&=\tilde{g}^{-1}(\mathscr{G}_2(\tilde{g}(a);\theta))\\&=r(\hat{g}^{-1}(\mathscr{G}_2(\tilde{g}(a);\theta)))\\&=r(\hat{g}^{-1}(\mathscr{G}_2(\hat{g}(x);\theta)))\\&=r(\mathscr{G}(x;\theta)),\end{aligned}$$

which concludes the proof. □

5.B A DISCUSSION ON ISOMETRY INVARIANCE

Here, we will elaborate on how the functional map framework (Ovsjanikov et al., 2012) can be seen as an application of our orbit mapping for isometry invariance. Functional maps are a widely used method to find correspondences between isometric shapes, and we will show here that the framework fits within our proposed theory. Non-rigid correspondence is a notoriously hard problem, and joint optimization within a larger framework makes it even more complex. To resolve this the idea of functional maps is to change the representation of the correspondence from point-wise to function-wise. By choosing the eigenfunctions of the Laplace-Beltrami operator [31] as the basis for functions on the shapes, the problem becomes a least squares problem aligning suitable descriptor functions in the space of functions.

Here, $F\in\mathscr{F}(\mathscr{X})$ and $G\in\mathscr{F}(\mathscr{Y})$ are descriptor functions on the shapes $\mathscr{X}$ and $\mathscr{Y}$ respectively. They are assumed to take similar values on corresponding points on $\mathscr{X},\mathscr{Y}$, and generate the designated orbit element within our framework. These descriptors are projected onto the eigenfunctions of $\mathscr{X},\mathscr{Y}$, named Φ,Ψ respectively. These projections are the chosen elements of the orbit we will align, and, for isometries and sufficiently comparable descriptors, the projections can be aligned by an orthogonal transformation generating the group action which is exactly the functional map C. The vanilla functional map optimization looks like this:

$$\underset{C\in O(k)}{\arg\min}\|C\Phi^{-1}F-\Psi^{-1}G\|_2^2 \tag{5.14}$$

Functional maps are often used when shape correspondence is required within another framework, and has been used in many deep learning applications [7],[16],[22]. Due to its wide application, we will not provide extra experiments to show its efficacy but want to emphasize that this is a possible implementation of our theory.

5.C INVARIANCE TO IMAGE ROTATIONS USING CONVOLUTION KERNELS

Let $u(z)$ denote the continuous image function with $z\in\mathbb{R}^2$ representing the spatial coordinates of an image. The invariance set for the orbit of continuous image rotations is

$$S=\{g:\mathscr{X}\to\mathscr{X}\mid g(u)(z)=(u\circ r(\alpha))(z),\text{ for }\alpha\in\mathbb{R}\},$$
$$\text{and } r(\alpha)=\begin{pmatrix}\cos(\alpha) & -\sin(\alpha)\\ \sin(\alpha) & \cos(\alpha)\end{pmatrix}\text{ is the rotation matrix.}$$

Let us consider two kernels $k_i:\mathbb{R}^2\to\mathbb{R}$, $i=\{1,2\}$. We now investigate the convolution of a kernel with a rotated image $(u\circ r(\alpha))(z)$

$$\begin{aligned}(k_i*u\circ r(\alpha))(z)&=\int_{\mathbb{R}^2}k_i(x)(u\circ r(\alpha))(z-x)dx\\ &=\int_{\mathbb{R}^2}k_i(x)u(r(\alpha)z-r(\alpha)x)dx\\ &=\int_{\mathbb{R}^2}k_i(r^T\varphi)u(r(\alpha)z-\varphi)d\varphi\\ &\text{with }\varphi=r(\alpha)x\end{aligned}$$

Now assume

$$\begin{pmatrix}k_1(r^T(\alpha)\varphi)\\ k_2(r^T(\alpha)\varphi)\end{pmatrix}=r^T(\alpha)\begin{pmatrix}k_1(\varphi)\\ k_2(\varphi)\end{pmatrix}. \tag{5.15}$$

Then

$$\begin{aligned}\begin{pmatrix}(k_1*(u\circ r(\alpha)))(z)\\ (k_2*(u\circ r(\alpha)))(z)\end{pmatrix}&=\int_{\mathbb{R}^2}r^T(\alpha)\begin{pmatrix}k_1(\varphi)\\ k_2(\varphi)\end{pmatrix}u(r(\alpha)z-\varphi)d\varphi.\\ &=r^T(\alpha)\begin{pmatrix}(k_1*u)(r(\alpha)z)\\ (k_2*u)(r(\alpha)z)\end{pmatrix}\end{aligned}$$

Then for a suitable set Z which makes the integral rotationally invariant, (e.g. circles around image center)

$$\int_Z\begin{pmatrix}(k_1*(u\circ r(\alpha)))(z)\\ (k_2*(u\circ r(\alpha)))(z)\end{pmatrix}dz=r^T(\alpha)\int_Z\begin{pmatrix}(k_1*u)(\varphi)\\ (k_2*u)(\varphi)\end{pmatrix}d\varphi \tag{5.16}$$

And we can determine the optimal rotation as solution to

$$\hat{g} = \arg\max_{g \in S} \left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \int_Z \begin{pmatrix} k_1 * u \\ k_2 * u \end{pmatrix} (z) \, dz \right\rangle \tag{5.17}$$

whose solution is given by $\hat{\alpha}$ such that

$$\begin{pmatrix} \cos\hat{\alpha} \\ \sin\hat{\alpha} \end{pmatrix} = \frac{\int_Z \begin{pmatrix} k_1 * u \\ k_2 * u \end{pmatrix} (z) \, dz}{\left\| \int_Z \begin{pmatrix} k_1 * u \\ k_2 * u \end{pmatrix} (z) \, dz \right\|} \tag{5.18}$$

We can see that (5.15) is a necessary condition to ensure invariance to image rotations using orbit mapping with (5.18) employing convolution kernels k_1 and k_2. For discrete convolution kernels, eq. (5.15) is not exactly satisfied for arbitrary rotations due to discretization problem. We can deduce necessary conditions on discrete kernels k_1 and k_2 to satisfy eq. (5.15) for rotations in multiples of 90^o. For square kernels k_1 and k_2 of size $N \times N$, we find that

$$k_1[i,j] = k_1[N-i+1, N-j+1] \text{ and} \tag{5.19}$$

$$k_2 = k_1 \circ r(-90^o) \tag{5.20}$$

are necessary to satisfy the condition (5.15) for $\alpha = 90^o$.
For $N = 2$, this gives kernels of the form

$$k_1 = \begin{pmatrix} a & b \\ -b & -a \end{pmatrix} \text{ and } k_2 = \begin{pmatrix} -b & a \\ -a & b \end{pmatrix}$$

For $N = 3$,

$$k_1 = \begin{pmatrix} a & b & c \\ d & 0 & -d \\ -c & -b & -a \end{pmatrix} \text{ and } k_2 = \begin{pmatrix} -c & d & a \\ -b & 0 & b \\ -a & -d & c \end{pmatrix}$$

Note that computing gradients using central differences satisfies (5.19) and (5.20), whereas using forward differences does not satisfy these conditions. Therefore, we observe more instabilities in orbit mapping when forward differences are used for gradient computation, see Tab. 5.1.

5.D DETAILS ABOUT THE EXPERIMENTAL SETTING

ROTATION INVARIANCE FOR IMAGES For our experiments with image rotational invariance, we used Pytorch(v.1.8.1), python(v.3.8.8), torchvision(v.0.9.1). The exact training protocol is provided below.

CIFAR10 We trained a Resnet18 (He et al., 2016) on the CIFAR 10 dataset, using stochastic gradient descent with initial learning rate 0.1, momentum 0.9, and weight decay 5e-4. Additionally, we trained a small Convnet and a linear model which used an initial learning rate of 0.01. For all the models, the learning rate is decayed by a factor of 0.5 whenever the validation loss does not decrease for 5 epochs. Training data is augmented using random horizontal flips, random crops of size 32 after zero-padding by 4 pixels. We divide the training data into train (80%) and validation (20%) sets. Networks are trained for 150 epochs with a batch size of 128 and we report the results on the test set using the model with best validation accuracy. The experiments with CIFAR10 were performed partially on a machine with one Nvidia TITAN RTX, and partially on a machine with 4 NVIDIA GeForce RTX 2080 GPUs.

HAM10000 We fine-tuned an ImageNet pretrained[5] NFNet-F0 (Brock et al., 2021) on HAM10000 dataset (Tschandl et al., 2018). The dataset is split into 8912 train and 1103 validation images using stratified split, ensuring there are no duplicates with the same lesion ids in the train and validation sets. Training data is augmented using random horizontal and vertical flips and color jitter, and randomly oversample the minority classes to mitigate class imbalance. The network is finetuned for 5 epochs, with a batch size of 128 and learning rate of 1e-4, weight decay of 5e-4 using Adam optimizer (Kingma and Ba, 2014) with exponential learning rate decay, with factor 0.2. For training using TI-pool which uses 4 rotated copies of images, we reduce the batch size to 32 to fit the GPU memory. For experiments with STN, we use a 3-layered CNN with convolution filers of size 3 × 3 followed by 2 fully connected layers for pose prediction. For experiment with ETN, we use a CNN with 4 conv layers with 64 channels and 2 fully connected layers for pose prediction. We report results using final iterate on the validation set. The experiments with HAM10000 dataset were partially performed on a machine with one NVIDIA TITAN RTX card, and partially on machine with 4 NVIDIA GeForce RTX 2080 GPUs.

CUB200 This is a small dataset containing 11,788 images of birds, split into 5994 images for training and 5794 test images. Since training a network from scratch gives low accuracies (around 35% clean accuracy with Resnet-50), we instead perform finetuning using an ImageNet pretrained Resnet-50 from pytorch torchvision (v.0.9.1) on CUB-200 dataset (Wah et al., 2011). The training data is augmented using random horizontal flips, and random resized crops of size 224. The network is finetuned for 60 epochs with a batch size of 128 and initial learning rate of 1e-4, using Adam optimizer (Kingma and Ba, 2014), weight decay of 5e-4, with exponential learning rate decay, with factor 0.9. For training using TI-pool which uses 4 rotated copies of images, we reduce the batch size to 64 to fit in the GPU memory. For experiment with ETN, we use a CNN with 4 conv layers with 64 channels and 2 fully connected layers for pose prediction. We report the accuracies using the final iterate on the test set. The experiments on the CUB-200 dataset were performed on a machine with 4 NVIDIA GeForce RTX 2080 GPUs.
The three image datasets including HAM10000 dataset (Tschandl et al., 2018) used in this chapter are publicly available and widely used in machine learning literature. To the best of our knowledge, they do not contain offensive content or personally identifiable information.

ROTATION AND SCALE INVARIANCE FOR 3D POINT CLOUDS We investigate invariance to rotations and scale for 3D point clouds with the task of point cloud classification on the *modelnet40* dataset (Wu et al., 2015). For this dataset note the asset descriptions at `https://modelnet.cs.princeton.edu/`.We use the resampled version of `shapenet.cs.stanford.edu/media/modelnet40_normal_resampled.zip`. We follow the hyperparameters of (Qi et al., 2017a,b) with improvements from the implementation of (Yan, 2019) on which we

[5] pretrained model from `https://github.com/rwightman/pytorch-image-models` licensed Apache 2.0

base our experiments. We train a standard PointNet for 200 epochs with a batch size of 24 with Adam (Kingma and Ba, 2014) with a base learning rate of 0.001, weight decay of 0.0001. During training, we sample 1024 3D points from every example in *Modelnet40*, randomly scale with a scale from the interval [0.8, 1.25], and randomly translate by an offset of up to 0.1 - if not otherwise mentioned in our experiments. This is the training procedure proposed in (Yan, 2019). However, we always train the model for the full 200 epochs and report final *class* accuracy based on the final result - we do not report instance accuracy. We further report invariance tests based on the final model. We evaluate rotational invariance by testing on 16×16 regularly spaced angles from $[0, 2\pi]$, rotating along xy and yz axes. We evaluate scaling invariance by testing the scales $\{0.001, 0.01, 0.1, 0.5, 1.0, 5.0, 10, 100, 1000\}$. All experiments for this dataset were run on three single GPU office machines, containing an NVIDIA TITAN Xp, and two GTX 2080ti, respectively.

5.E ADDITIONAL NUMERICAL RESULTS

Discretization effects in CUB200 We further investigate the effect of discretization using different interpolation schemes for rotation on higher resolution on the CUB-200 dataset (trained at 224x224 resolution) fine-tuned using Resnet-50. Tab. 5.8 shows the results of different training schemes with and without our orbit mapping (*OM*) obtained when using different interpolation schemes for rotation. Besides standard training (*Std.*), we use rotation augmentation (*RA*), and the adversarial training and regularization from (Engstrom et al., 2019; Yang et al., 2019). Even for this higher resolution dataset, the worst-case accuracies between different types of interpolation may differ by more than 15%.

Train	OM	Clean.	Average			Worst-case		
			Nearest	Bilinear	Bicubic	Nearest	Bilinear	Bicubic
Std.	✗	**77.41±0.33**	37.67±0.35	52.45±0.29	51.87±0.31	3.19±0.49	8.07±0.35	8.16±0.33
	✓Train+Test	71.19±0.34	63.35±0.30	71.56±0.34	70.93±0.35	40.63±0.48	58.80±0.39	59.02±0.41
RA.	✗	69.89±0.28	67.61±0.33	70.12±0.34	68.83±0.37	34.88±0.47	41.01±0.41	40.50±0.43
	✓Test	69.41±0.31	69.19±0.32	69.27±0.29	68.53±0.38	48.63±0.43	56.28±0.39	55.86±0.40
	✓Train+Test	70.35±0.46	69.41±0.23	70.72±0.18	70.37±0.34	47.92±0.26	57.54±0.39	57.62±0.14
Advers.	✗	64.54±0.17	53.74±0.65	64.07±0.25	63.22±0.54	26.63±0.79	42.82±0.60	42.44±0.55
Mixed	✗	68.56±0.46	57.17±0.60	65.91±0.42	65.76±0.51	28.06±0.58	42.87±0.32	42.92±0.38
Advers.-KL	✗	64.47±0.35	53.93±0.35	64.65±0.26	64.02±0.34	26.94±0.46	43.04±0.63	42.61±0.37
Advers.-ALP	✗	64.63±0.31	55.56±0.67	64.34±0.17	63.21±0.24	29.55±0.69	43.63±0.21	43.48±0.32
ETN	✗	64.14±0.24	64.26±0.65	66.95±0.42	64.32±0.62	43.33±1.01	52.85±1.12	49.72±1.31
TIpool	✗	76.80±0.25	60.67±0.79	74.90±0.15	74.82±0.24	36.06±1.12	59.04±0.37	59.50±0.41
TIpool-RA	✗	73.47±0.48	72.30±0.51	74.71±0.29	73.65±0.36	57.22±0.64	62.82±0.56	62.31±0.42
TIpool	✓Train+Test	76.82±0.15	68.50±0.58	**77.18±0.18**	**77.04±0.16**	49.85±0.65	**69.19±0.36**	**69.64±0.33**
TIpool-RA	✓Train+Test	74.78±0.20	**73.79±0.48**	75.89±0.17	75.07±0.16	**59.57±0.57**	67.78±0.20	67.64±0.18

Table 5.8: Effect of augmentation and including gradient based orbit mapping *(OM)* on robustness to rotations with different interpolations for CUB200 classification using Resnet50. Shown are clean accuracy on standard test set and average and worst-case accuracies on rotated test set. Mean and standard deviations over 5 runs are reported.

In particular, adversarial training with bi-linear interpolation is still vulnerable to image rotations with nearest-neighbor interpolation. The learned ETN also exhibits a similar behavior. While our approach is also affected by the interpolation effects, the vulnerability to nearest neighbor interpolation is ameliorated when using rotation augmentation. We obtain the best results by combining orbit mapping with the discrete invariant approach (Laptev et al., 2016)

Effect of Network architecture for CIFAR10 To investigate the effectiveness of our approach, we experiment with three different network architectures: *i) a linear network, ii) a 5-layer convnet ii) a Resnet18.* We compare the performance of our orbit mapping approach with

Network	Train	OM	Std.	Average			Worst-case		
				Nearest	Bilinear	Bicubic	Nearest	Bilinear	Bicubic
Linear	Std.	✗	**38.89±0.17**	25.31±0.21	25.57±0.22	25.48±0.24	2.50±0.11	3.56±0.17	3.26±0.11
		✓Train+Test	31.87±0.10	**31.25±0.04**	**31.58±0.05**	**31.33±0.04**	13.08±0.23	18.85±0.21	18.21±0.21
	RA	✗	29.73±0.18	30.66±0.03	30.77±0.03	30.72±0.03	14.30±0.42	18.31±0.29	16.94±0.37
		✓Test	30.60±0.13	30.52±0.07	30.65±0.08	30.54±0.09	16.83±0.47	21.17±0.28	20.37±0.26
		✓Train+Test	31.06±0.26	31.07±0.11	31.27±0.10	31.13±0.09	**19.19±0.28**	**24.25±0.31**	**23.68±0.31**
	Advers.	✗	28.82±0.77	29.46±0.60	29.62±0.56	29.36±0.56	11.45±0.81	14.20±0.93	13.65±0.55
Convnet	Std.	✗	**86.12±0.33**	32.01±0.32	35.97±0.26	38.15±0.36	0.85±0.09	0.57±0.06	0.89±0.14
		✓Train+Test	76.13±0.96	64.34±0.35	71.21±0.96	**74.61±0.84**	25.78±0.49	49.60±0.79	55.57±0.81
	RA	✗	75.03±0.99	71.77±0.84	65.45±0.66	70.22±0.66	27.96±0.50	27.06±0.61	32.51±0.53
		✓Test	70.12±0.64	67.64±0.55	61.03±0.67	66.09±0.71	39.01±0.57	42.88±0.90	49.39±0.68
		✓Train+Test	74.30±0.77	**73.24±0.58**	69.52±0.53	73.38±0.59	**46.25±0.54**	**53.36±0.57**	**59.04±0.53**
	Advers.	✗	72.96±0.95	62.08±0.59	**74.29±0.88**	73.86±0.76	26.24±0.43	50.99±0.54	52.46±0.51
Resnet18	Std.	✗	**93.98±0.32**	35.12±0.81	40.06±0.44	42.81±0.50	0.79±0.38	1.31±0.13	2.22±0.17
		✓ Train+Test	87.99±0.43	72.40±0.33	**84.12±0.55**	**86.61±0.49**	34.57±0.94	68.60±0.81	74.49±0.84
	RA	✗	85.54±0.72	80.47±0.74	75.99±0.72	79.47±0.65	45.50±0.83	44.71±0.74	50.50±0.78
		✓ Test	79.26±0.42	74.93±0.51	69.31±0.65	73.94±0.63	48.93±0.75	52.18±0.91	58.69±0.78
		✓ Train+Test	85.40±0.57	**84.37±0.58**	81.82±0.59	84.82±0.52	**66.22±0.75**	**71.09±1.01**	**76.44±0.89**
	Advers.	✗	69.32±1.61	61.73±1.12	68.54±0.68	68.00±0.31	36.95±0.97	50.21±0.55	49.73±0.98

Table 5.9: Comparing rotational invariance using training schemes vs. orbit mapping for CIFAR10 classification using *i) Linear network ii) 5-layer Convnet iii) Resnet18*. Shown are the mean clean accuracy and the average and worst case accuracies when test images are rotated in steps of 1 degree. The mean and standard deviation values over 5 runs are reported.

Method	Std.	STN	ETN	Adv.	OM
Train-time/epoch	18.05±0.05	18.90±0.05	18.89±0.07	72.09±0.18	18.59±0.04

Table 5.10: Average training time per epoch in seconds for different approaches to incorporate rotation invariance, with Resnet18 as base architecture for CIFAR10 classification. Training time corresponds to runs on a machine with a single Titan-RTX GPU.

training schemes, i.e. augmentation and adversarial training for rotational invariance in Tab. 5.9. For all the three architectures considered, our orbit mapping together with rotation augmentation consistently results in the most accurate predictions in the worst case.

Comparing Computation Complexity for CIFAR10 In Tab. 5.10, the training times using different approaches are compared for rotation-invariant CIFAR10 classification. It can be noted that the proposed gradient based orbit mapping is significantly easier and computationally cheaper to train in comparison with other approaches for incorporating invariance. In contrast, adversarial training is the most computationally expensive approach.

Comparing Computational Complexity of ROTMNIST Tab. 5.11 compares the computational complexity of the D4/C4 and D16/C16 models. The D16/C16 model has significantly higher computational complexity than the D4/C4 model, though the number of learnable parameters is nearly the same. The network size of the D16/C16 network is higher due to more rotated copies of the filters, resulting in larger training and inference times. Orbit mapping adds no learnable parameters and increases training time very marginally (~0.3 seconds/epoch). Training times correspond to runs on a machine with a single Titan-RTX GPU.

OM	D4/C4	D16/C16
	Train-time/epoch	Train-time/epoch
✗	4.47 s	41.89 s
✓	4.78 s	42.08 s

Table 5.11: Comparing computational complexity of D4/C4 and D16/C16 models. Orbit mapping adds no learnable parameters and increases training time very marginally (~0.3 seconds/epoch). Training times correspond to runs on a machine with single Titan-RTX GPU.

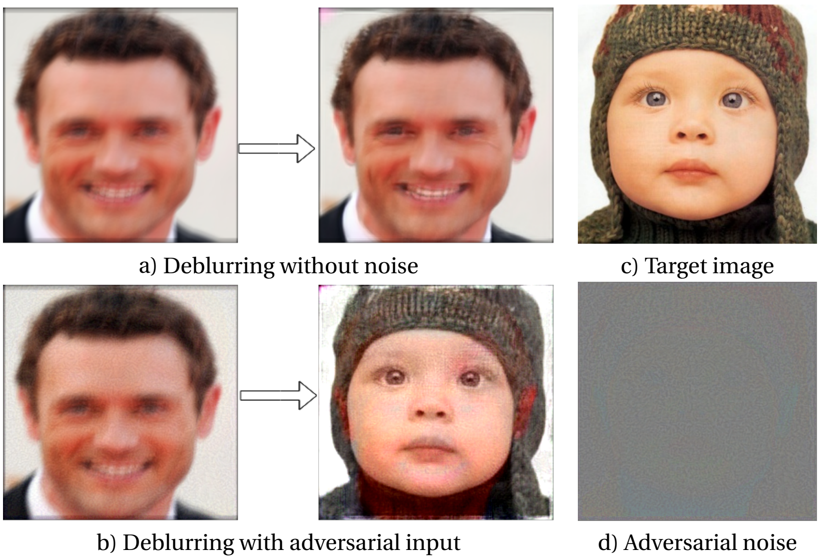
Declaration for Chapter 6 - Adversarial Robustness of Deep Image Recovery

This chapter is based on two papers- Gandikota et al. (2022a) titled "On adversarial robustness of deep image deblurring" co-authored by Kanchana Vaishnavi Gandikota, Paramanand Chandramouli and Prof. Michael Moeller, published at IEEE International Conference on Image Processing (ICIP) 2022, and, Gandikota et al. (2023) titled "Evaluating Adversarial Robustness of Low dose CT Recovery" co-authored by Kanchana Vaishnavi Gandikota, Paramanand Chandramouli, Hannah Dröge and Prof. Michael Moeller, published at Medical Imaging with Deep Learning (MIDL) 2023.

Kanchana Vaishnavi Gandikota and Paramanand Chandramouli jointly proposed this project idea on analyzing robustness of image recovery networks, focusing on robustness to targeted attacks. In Gandikota et al. (2022a) which analyzed robustness of image deblurring methods, Paramanand Chandramouli contributed to experiments with DeblurGAN, Kanchana Vaishnavi Gandikota reviewed the literature, contributed to the experimental evaluation and writing the first draft of the paper. Kanchana Vaishnavi Gandikota proposed to extend these ideas to CT reconstruction in Gandikota et al. (2023). Kanchana Vaishnavi Gandikota came up with the idea of robustness evaluation in terms of both error in image reconstruction and measurement consistency. Prof. Michael Moeller proposed the idea of localizing perturbations in the sinogram space, proposed evaluating Bregmann distances, provided code for implementing the same, and wrote the parts of the paper related to Bregman distances. Kanchana Vaishnavi Gandikota reviewed the literature, conducted experiments evaluating robustness to untargeted, localized and universal attacks, and contributed to writing the first draft of the paper. Hannah Droege trained iRadonmap network for CT recovery, and provided code for locating clinically relevant regions in CT images. Paramanand Chandramouli and Prof. Michael Moeller were involved in discussing the ideas and contributed to improving the writing. The research was supervised by Prof. Michael Moeller.

Additional follow-up experiments evaluating robustness of more recent restoration networks are provided in the appendix.

6 ADVERSARIAL ROBUSTNESS OF DEEP IMAGE RECOVERY


a) Deblurring without noise

c) Target image

b) Deblurring with adversarial input

d) Adversarial noise

Figure 6.1: Example targeted attack on DeblurGANv2 Kupyn et al. (2019), with adversarial noise level set to 8/255.

Following the success of deep neural networks for higher level computer vision applications, deep learning approaches are increasingly adopted in image recovery and restoration tasks (Zamir et al., 2022; Eboli et al., 2020; Ge et al., 2020; Pelt et al., 2018; Kuanar et al., 2019). While the vulnerabilities and instabilities of neural networks to adversarial examples are widely studied for image classification, they are less studied in the context of image recovery. This chapter attempts to fill this gap by taking a closer look at the adversarial robustness of image recovery.

The notion of robustness itself is very different for classification and reconstruction problems. For a classifier, robustness can be characterized by the minimal perturbation which can cause a sample to cross the decision boundary, leading to a change in classification outcome. In contrast, for reconstruction tasks, the outputs are not discrete labels, and there is no notion of a decision boundary. Robustness can instead be characterized by measuring the maximum change in the reconstruction with respect to change in the input. Furthermore, the ill-posedness of inverse imaging tasks means multiple valid solutions exist for the same measurement, in contrast to classification, where typically there is one salient object in an input image that belongs to a single class. This ill-posedness also implies a trade-off between the stability of the recovery algorithm, and the accuracy it can achieve in terms of proximity to ground truth. Moreover, there exist classical approaches for image reconstruction with convergence guarantees. To take these into account, in this chapter, we investigate the adversarial robustness of a variety of model-based methods, and deep neural networks for image recovery including model-inspired architectures.

We consider two example image recovery problems, image deblurring and low-dose computed tomography. While existing works focus on untargeted attacks on image reconstruction, we investigate the susceptibility of networks to targeted attacks, untargeted attacks, and also attacks aiming to modify a localized region in the reconstruction. Fig. 6.1 shows an example targeted attack on DeblurGANv2 (Kupyn et al., 2019), a popular image deblurring network. A tiny additive perturbation to the input is sufficient to change the network output from the image of a man to that of a baby. We can see that this adversarial reconstruction is clearly inconsistent with the measurement. While methods that take into account the forward measurement model (here, the blur operator) are relatively stable to such extreme targeted changes, they can still be susceptible to localized changes and untargeted attacks, indicating the necessity for robust networks for image recovery.

Let us now introduce the specific reconstruction problems of image deblurring and computed tomography reconstruction. We consider the forward measurement model from the chapter 3,

$$f = A(u) + n. \tag{6.1}$$

In the case of *image deblurring, A* corresponds to the blur operator. Blur occurs due to relative motion between cameras and objects in the scene during the exposure time, or due to sub-optimal focal settings. Recovering sharp images $\hat{u}$ from blurred inputs is a well-studied research problem. When the blur is uniform, the blur operation can be characterized using a convolution with blur kernel, and recovering $\hat{u}$ becomes a linear inverse problem. Even for non-blind deblurring, i.e. deblurring with a known blur operator, sharp image recovery is an ill-posed problem. When the blur operator is also unknown, it is referred to as blind deblurring, which is even more severely ill-posed, as multiple pairs of A and u can produce the same blurred observation f.

Computed tomography (CT) involves recording attenuated X-ray radiation projected at different angles by a scanner rotating around a target. The recorded measurements are arranged into a sinogram, from which a CT image is reconstructed. In this case, the forward operator A is a linear operator given by the 2D Radon transform (Radon, 1986), which models the attenuation of the radiation passing through the target by calculating line integral along the path of an X-ray beam. The measurement f is the sinogram, which consists of the recorded integrals for different distances and measurement angles. The degree of attenuation varies depending on the density of the tissue, and recording this attenuation at different angles aids in creating a detailed image of the internal structures of the target. This non-invasive imaging technique is widely used in medical diagnosis. While the accuracy and resolution of CT images improve with the number of X-ray beams used, exposure of patients to X-rays poses serious health risks. To reduce this risk, different solutions to low-dose CT acquisition have been proposed under the ALARA (as low as reasonably achievable) principle (Slovis, 2002; Newman and Callahan, 2011). These protocols can be broadly classified into two categories- i) adjusting the settings on the CT scanner tube to reduce the total number of X-ray photons and ii) recording measurements from fewer projection angles. Either or both of these approaches may be adopted to reduce radiation dosage. However, there exists a trade-off between dose reduction during CT acquisition and diagnostic quality. A lower number of X-ray photons degrades reconstruction quality due to increased image noise level. On the other hand, CT recovery from fewer projection angles can suffer from severe artefacts. Further, sparse-view CT is an ill-posed problem, and there can be many valid solutions for the same measurement.

While one could solve such inverse problems using the linear pseudo-inverse, it is highly sensitive to noise. Linearly filtering in Fourier space, commonly referred to as filtered back projection (FBP) (Feldkamp et al., 1984), is one standard classical approach to CT recovery.

For image deblurring, Wiener filtering, which involves linear filtering in Fourier space to achieve an optimal trade-off between inverse filtering and noise suppression is a standard classical approach. Variational approaches to such ill-posed reconstruction problems employ energy minimization with suitable priors using iterative algorithms. Examples include (Sidky et al., 2006; Chen et al., 2013) for CT reconstruction, (Getreuer, 2012; Bioucas-Dias et al., 2006; Krishnan and Fergus, 2009) for non-bind deblurring. Blind deblurring methods (Perrone and Favaro, 2014; Pan et al., 2016; Chen et al., 2019) employ alternate minimization with suitable priors to obtain both the image and the blur operator. In Chapter 3, we have discussed in detail different approaches to image recovery. In the following, we briefly discuss related work for deep learning methods specific to image deblurring and CT recovery and discuss prior works on adversarial attacks on image recovery.

6.1 RELATED WORK

6.1.1 *Deep learning for Image Deblurring and CT Recovery*

IMAGE DEBLURRING: Recently image restoration has witnessed a paradigm shift from classical approaches to using deep neural networks. We refer to (Koh et al., 2021; Su et al., 2022) for a detailed survey and comparison of deep learning based image deblurring methods. Neural network approaches to blind deblurring typically learn to invert the blur operation directly using a trained neural network (Kupyn et al., 2019; Zamir et al., 2021) from large datasets of sharp and blurry image pairs to recover clean images. However, there are also methods that explicitly include the estimation of a blur operator (Schuler et al., 2015; Chakrabarti, 2016). For non-blind deblurring, the knowledge of a blur operator has been successfully integrated into neural networks, by unrolling fixed steps of optimization algorithms with learned operators (Gong et al., 2020; Eboli et al., 2020; Bertocchi et al., 2020), or by using known deconvolution techniques in feature space (Dong et al., 2020) within the network. A few works (Vasu et al., 2018; Nan and Ji, 2020) also take into account kernel uncertainty in non-blind deblurring. In addition to end-to-end trained networks for image deblurring, neural networks are also used in iterative recovering sharp images from blurred observations, for example, by using trained denoisers as proximal operators for plug and play reconstruction (Meinhardt et al., 2017), or using trained generative priors (Asim et al., 2020b).

CT RECOVERY: Deep learning approaches to CT reconstruction tasks encompass a wide array of methods. These include deep neural network post-processors which denoise an initial reconstruction from the filtered-back-projection operator (Chen et al., 2017; Jin et al., 2017; Yang et al., 2018; Zhang et al., 2018c; Pelt et al., 2018; Kuanar et al., 2019), fully learned methods such as iRadonmap (He et al., 2020) and ADAPTIVE-Net (Ge et al., 2020), which also learn the filtered back projection operation in addition to learning the post-processor, unrolled optimization networks which unroll fixed iterations of algorithms such as gradient descent, primal-dual hybrid gradient, projected gradient descent with learned parameters (Adler and Öktem, 2017; Aggarwal et al., 2018; Adler and Öktem, 2018). In addition to end-to-end trained networks, the works (He et al., 2018; Gupta et al., 2018) learn projection or proximal step, Baguer et al. (2020) use untrained neural network prior (Ulyanov et al., 2018), and Song et al. (2022) use generative models trained on CT images in an iterative energy minimization.

In this chapter, we analyze the adversarial robustness of end-to-end trained deep networks, including, fully learned approaches, networks taking into account the forward operator and model-based architectures, which can recover solutions in a single forward pass. In addition, we consider the classical approaches for CT recovery, including filtered back projection and energy minimization with TV prior. We exclude iterative approaches involving deep network priors in our experiments due to high computational complexity.

6.1.2 *Adversarial Attacks on Image Reconstruction:*

While adversarial attacks on neural networks have been first introduced and extensively studied in the context of image classification (Szegedy et al., 2014; Goodfellow et al., 2015; Madry et al., 2018), analysis of adversarial robustness for reconstruction has received less attention. In the context of image restoration, Choi et al. (2019) shows that several state-of-the-art trained networks for image super-resolution are susceptible to adversarial perturbations, however, they consider only direct inversion models, with a focus on untargeted attacks. To the best of our knowledge, such attacks have not been shown for deblurring prior to our work. Recent works starting from (Antun et al., 2020; Raj et al., 2020) demonstrated the susceptibility of image reconstruction networks to adversarial attacks. While Antun et al. (2020) study instabilities of MRI reconstruction networks by adding perturbations in the image domain, Raj et al. (2020) consider adversarial perturbations in measurement domain and perform adversarial training using auxiliary network to generate adversarial examples. However, these works consider mainly untargeted attacks for networks performing direct inversion or post-processing. Cheng et al. (2020) perform adversarial attacks to generate tiny features that cannot be recovered well by MRI reconstruction networks and propose adversarial training to improve the network's sensitivity to such features. Darestani et al. (2021); Morshuis et al. (2022) show that adversarial perturbations can alter diagnostically relevant regions in recovered MRI images. In the context of CT recovery, Huang et al. (2018) perform preliminary investigations whether additive adversarial perturbations lead to incorrect reconstruction of an existing lesion. Closely related to our work, Genzel et al. (2022); Wu et al. (2022) also investigate the adversarial robustness of different approaches for CT recovery. They mainly considered untargeted attacks, with some preliminary experiments in (Genzel et al., 2022) on targeted changes indicating that reconstruction networks are largely robust to targeted changes. In contrast, we conduct an in-depth study on the effect of adversarial attacks on the recovery algorithm by measuring reconstruction quality in terms of similarity with the ground truth in the image domain, as well as consistency of the reconstruction with the input sinogram in the measurement domain. We investigate the susceptibility of image recovery methods to untargeted attacks, targeted attacks, universal attacks, and localized adversarial attacks targeting diagnostically relevant or semantically meaningful regions.

6.2 STABILITY OF IMAGE RECONSTRUCTION

Consider the problem of reconstructing u from the measurement model eq. (6.1). For simplicity, we assume A to be a linear operator. A desirable property for the reconstruction algorithm is *stability*, in the sense that the algorithm's output varies smoothly with respect to changes in the input. The notion of stability may be characterized using Lipschitz continuity. If a reconstruction algorithm $\mathcal{G}$ satisfies

$$\|\mathcal{G}(f+\delta)-\mathcal{G}(f)\| \leq L\|\delta\|, \tag{6.2}$$

then $\mathcal{G}$ is a Lipschitz continuous mapping with Lipschitz constant L. From the point of view of stability, a small value of L is desirable to ensure that the maximal change in the reconstruction with respect to a small change in measurement remains small and bounded. Yet, this stability comes at the cost of reconstruction performance. For instance, Sommerhoff et al. (2019) observed that enforcing non-expansiveness ($L \leq 1$) drastically decreased the denoising performance of neural network denoisers. This trade-off becomes clearer as we go beyond denoising towards more ill-posed recovery problems (Gottschling et al., 2020). A large change in image space Δ_u can still result in only a tiny change $\delta = A\Delta_u$ in the measurement space when a large component of Δ_u is in the null-space of A. In such cases, a small value of L implies the inability of $\mathcal{G}$ to accurately reconstruct the ground truth. Conversely, a deterministic recovery algorithm $\mathcal{G}$ which can recover both u and $u+\Delta_u$ accurately will have a much larger L, and is therefore less stable. Despite these trade-offs, the Lipschitz constant remains an important tool to evaluate the stability of reconstruction methods. However, analyzing the stability of neural networks in terms of the Lipschitz constant is difficult, owing to the high complexity involved in its exact computation, even for moderately sized neural networks (Jordan and Dimakis, 2020). As a result most works (Anil et al., 2019; Weng et al., 2018; Virmaux and Scaman, 2018; Combettes and Pesquet, 2020) only compute approximations and upper bounds for the Lipschitz constant.

In contrast, it is easier to analyze the stability of classical approaches. The stability of the standard linear techniques can be analyzed via the singular values of the reconstruction operator, see, e.g. (Bauermeister et al., 2020) for learning linear reconstructions in such a context. In the case of variational energy minimization approaches for linear inverse problems, a stability estimate shown in (Burger et al., 2007) is

$$\|f_1-f_2\|^2 \geq \|Au_1-Au_2\|^2 + 2\langle p_1-p_2, u_1-u_2\rangle, \quad p_1 \in \partial R(u_1),\ p_2 \in \partial R(u_2), \tag{6.3}$$

where the term $\langle p_1-p_2, u_1-u_2\rangle$ is the 'symmetric Bregman distance' with respect to the convex regularizer R. While it is difficult to estimate exact error bounds for neural network based reconstruction, empirical analysis of stability can be done using adversarial attacks for both model based approaches and neural networks. In this chapter, we empirically analyze robustness of image deblurring and CT reconstruction for different model based approaches and deep learning approaches.

6.3 ADVERSARIAL ATTACKS ON IMAGE RECOVERY

Adversarial attacks on image recovery methods make small changes to the inputs causing unpredictable large changes in the output. In this work, we consider robustness to tiny L_∞ norm bounded additive perturbations in the measurement domain. We assume that the parameters of the neural network $\mathcal{G}$ or the recovery algorithm is fully known to the attacker, i.e. the white box setting.

Untargeted Attacks: Here the aim is to find an additive L_∞ norm constrained perturbation in the measurement domain that maximizes the reconstruction error:

$$\delta_{adv} = \arg\max_{\delta} \left\|\mathcal{G}\left(f+\delta\right) - \mathcal{G}\left(f\right)\right\|_2 \text{ s.t. } \|\delta\|_\infty \leq \epsilon. \tag{6.4}$$

Targeted Attacks: Here the goal is to find an additive perturbation in the measurement domain that produces a reconstruction close to a target image $\tilde{u}$ subject to L_∞ constraints on the perturbation,

$$\delta_{adv} = \arg\min_{\delta} \left\|\mathcal{G}\left(f+\delta\right) - \tilde{u}\right\|_2 \text{ s.t. } \|\delta\|_\infty \leq \epsilon. \tag{6.5}$$

In the case of CT recovery, attempting to reconstruct a target image totally different from the original solution does not produce meaningful results. We instead perform a localized attack which attempts to modify only clinically relevant regions in the reconstruction.

Localized Attacks: Here the goal is to find an additive L_∞ norm constrained perturbation that produces a change the visual appearance and alters predicted malignancy in a localized clinically relevant region. We utilize an adversarially trained classifier $\mathcal{G}_\theta$ trained to classify chest CT nodules to guide the attack towards a plausible change in visual appearance locally. Note that using a non-robust classifier in the attack would cause misclassification even without perceptible changes in reconstruction. Our localized attack can be formulated as:

$$\delta_{adv} = \arg\max_{\delta} E\left(\mathcal{G}_\theta\left(g_c\left(\mathcal{G}(f+\delta)\right)\right), y\right) \text{ s.t. } \|\delta\|_\infty \leq \epsilon. \tag{6.6}$$

where $g_c(\cdot)$ crops the region of interest, and $y = \mathcal{G}_\theta\left(g_c\left(\mathcal{G}\left(f\right)\right)\right)$ is the predicted label for the region of interest in the clean reconstruction. $E(\cdot)$ refers to the energy function (loss) to be maximized for binary classification of nodules, which is the binary cross entropy loss. To ensure that the degradation remains localized, and to avoid artifacts at the boundary of the local region, we apply a smoothed mask to the adversarial noise setting at every step. The mask is calculated as the sinogram of the Gaussian-smoothed spatial mask corresponding to the region of interest, normalized to have a maximum value of 1.

Universal Attacks: Here we aim to find an input-agnostic L_∞ norm constrained adversarial perturbation that maximizes the reconstruction error of a recovery method $\mathcal{G}$ for any input. This input-agnostic perturbation is optimized over a set of examples:

$$\delta_{uniadv} = \arg\max_{\delta} \sum_{\text{examples i}} \left\|\mathcal{G}\left(f_i+\delta\right) - \mathcal{G}\left(f_i\right)\right\|_2 \text{ s.t. } \|\delta\|_\infty \leq \epsilon. \tag{6.7}$$

We solve the constrained optimization problems eq. (6.4), eq. (6.5), eq. (6.6), eq. (6.7) using the projected gradient descent (PGD) algorithm (Madry et al., 2018). The adversarial examples $\left(f+\delta_{adv}\right)$ are finally clipped to lie in the range of valid intensities of f.

6.4 EXPERIMENTS AND RESULTS

6.4.1 *Deblurring*

We use the following networks in our experiments: i) DeblurGANv2 (Kupyn et al., 2019) and ii) MPRNet (Zamir et al., 2021), which are end-to-end trained networks for dynamic image (blind) deblurring, as well as iii) (Gong et al., 2020), a learned recurrent gradient descent network, and iv) (Dong et al., 2020), which performs Wiener deconvolution in the feature space of

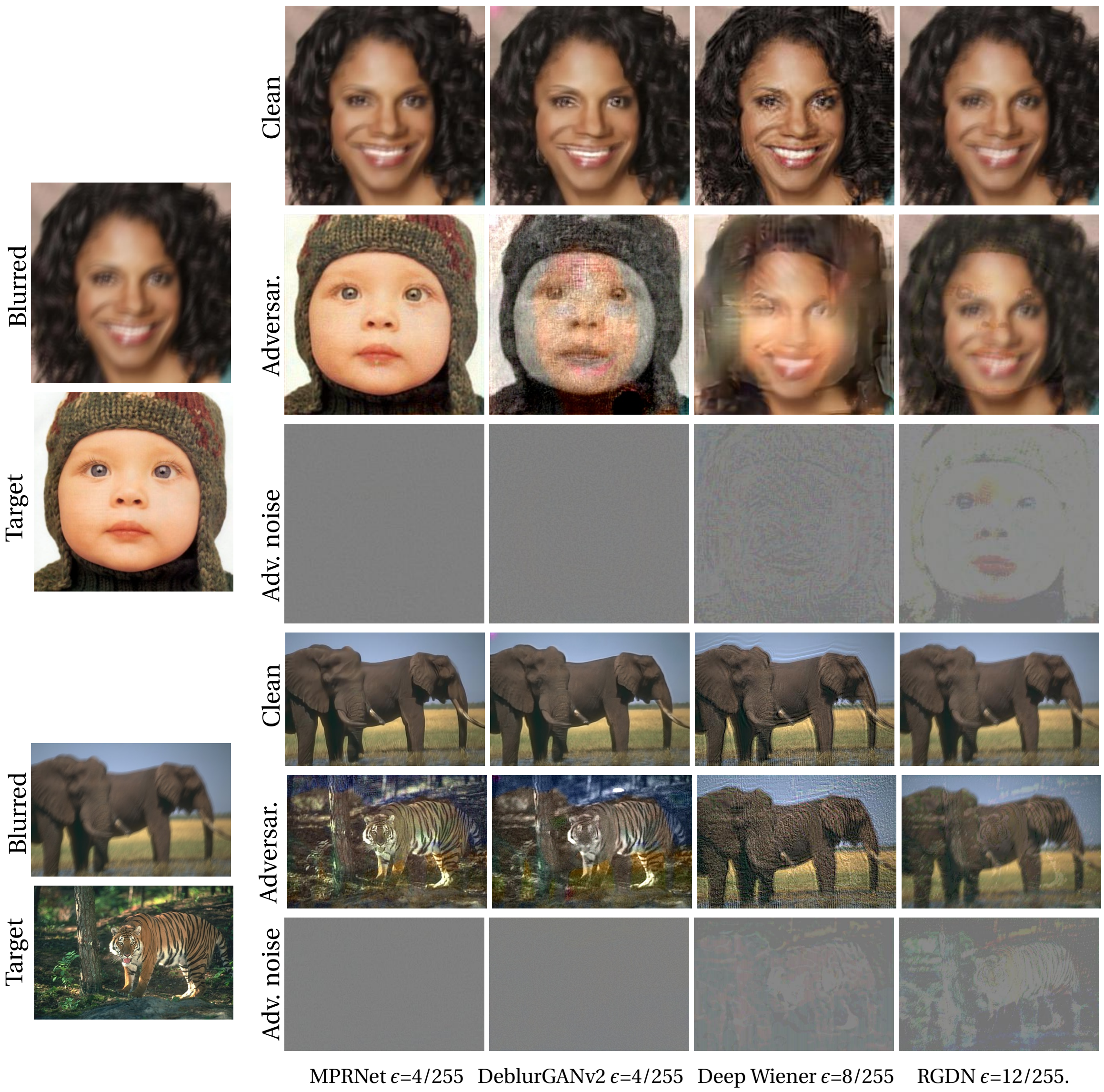


Figure 6.2: Example targeted adversarial attacks on deblurring networks MPRNet Zamir et al. (2021), DeblurGANv2 Kupyn et al. (2019),RGDN Dong et al. (2020), deep Wiener network Gong et al. (2020) on Face images. Blurred images in rows 1 and 2 are generated using blur kernels ‘1’ and ‘2’ of size 19 × 19 and 17 × 17 in the dataset of Sun et al. (2013) respectively.

neural networks. The non-blind deblurring networks (Gong et al., 2020; Dong et al., 2020) use the knowledge of blur operator during reconstruction. We use publicly available trained models of all these networks made available by the authors. For attacks on DeblurGANv2 (Kupyn et al., 2019), we choose the version using the inception backbone since it achieves the best results. In their experiments, Gong et al. (2020) can unroll the recurrent gradient descent network for an arbitrary number of steps till a stopping criterion is satisfied. However, it becomes prohibitively complex to perform adversarial attacks for a high number of unrolled steps. In our experiments, we limit the number of unrolled steps to 10 for crafting adversarial inputs, but evaluate robustness to the same inputs using a network with 50 unrolled steps. We create a synthetic dataset of 80 blurred images generated by convolving uniform blur kernels of the dataset in (Sun et al., 2013) with sharp images, a subset of images from CelebA-HQ dataset resized to 256 × 256, and a subset of images from the Berkeley segmentation dataset (BSDS300). For the targeted attacks, we use the image ’baby’ from Set5, and the image ‘108005’ from BSDS300 as the targets for face images and BSDS images respectively. In

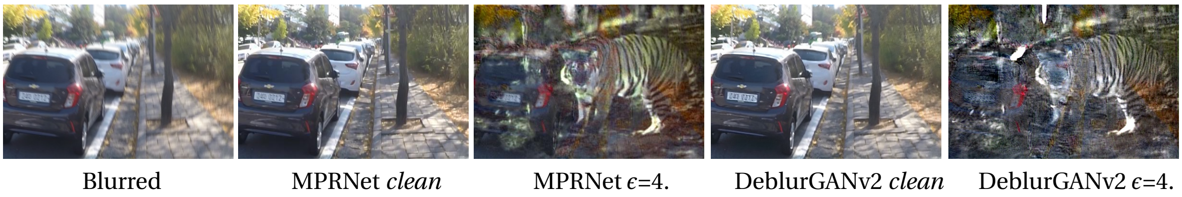
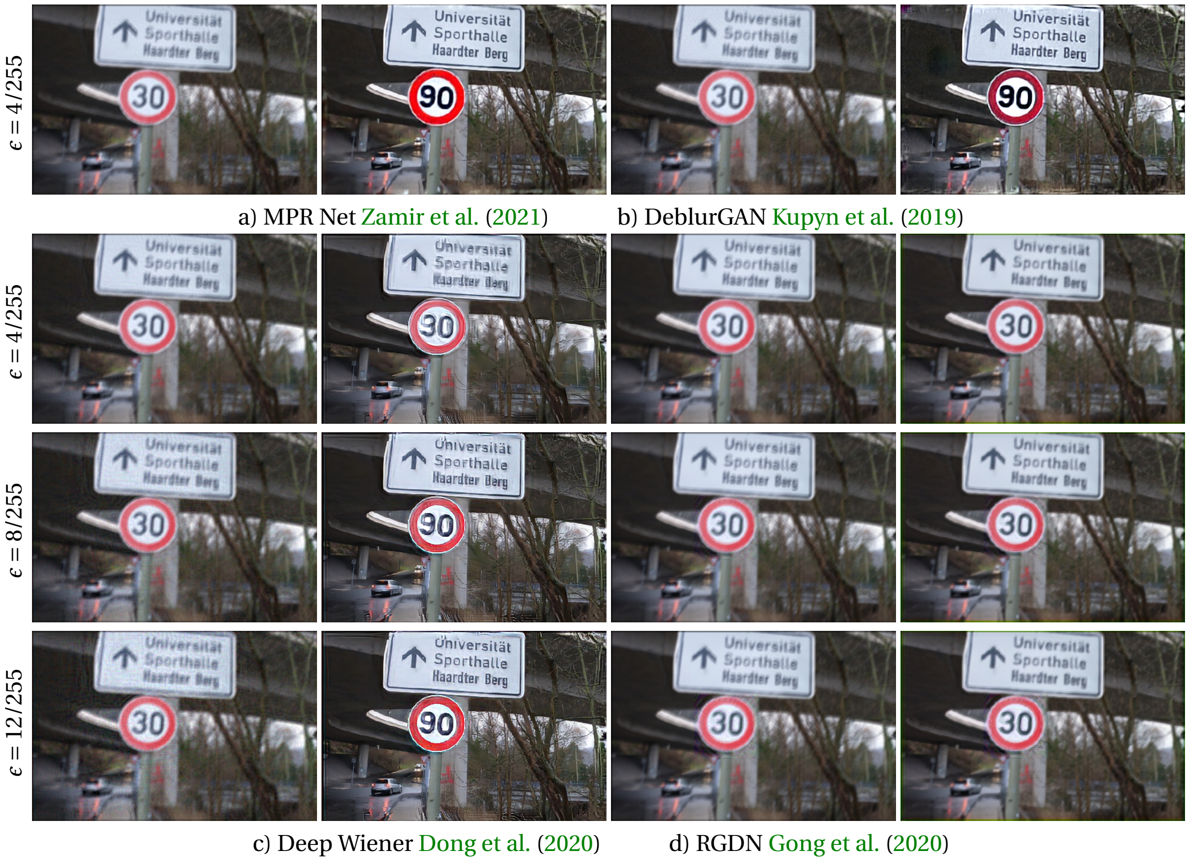


Figure 6.3: Targeted attack with dynamically blurred input on DeblurGANv2 Kupyn et al. (2019) and MPRNet Zamir et al. (2021).



a) MPR Net Zamir et al. (2021) b) DeblurGAN Kupyn et al. (2019)

c) Deep Wiener Dong et al. (2020) d) RGDN Gong et al. (2020)

Figure 6.4: Illustration of localized targeted attack. For each approach, the first column is the adversarial input and the second column is the network prediction.

the appendix, we additionally evaluate more recent dynamic deblurring networks including MIMO-UNet (Cho et al., 2021), and transformer based methods, Uformer (Wang et al., 2022c), Stripformer (Tsai et al., 2022), Restormer (Zamir et al., 2022), NAFNet (Chen et al., 2022a) only on the set of blurred face images. We do not evaluate these methods on blurred BSDS images due to GPU memory constraints in processing these images in their full resolution. We use a step size of $1e-2$ and use 25 PGD steps and 50 PGD steps to craft untargeted and targeted adversarial perturbations. We measure the effect of attack on reconstruction by measuring peak-signal-to-noise-ratio (PSNR) and normalized-cross-correlation (NCC), with respect to ground truth for untargeted attacks, and PSNR and NCC with respect to both the ground truth and the target image for the targeted attacks.

Targeted Attacks

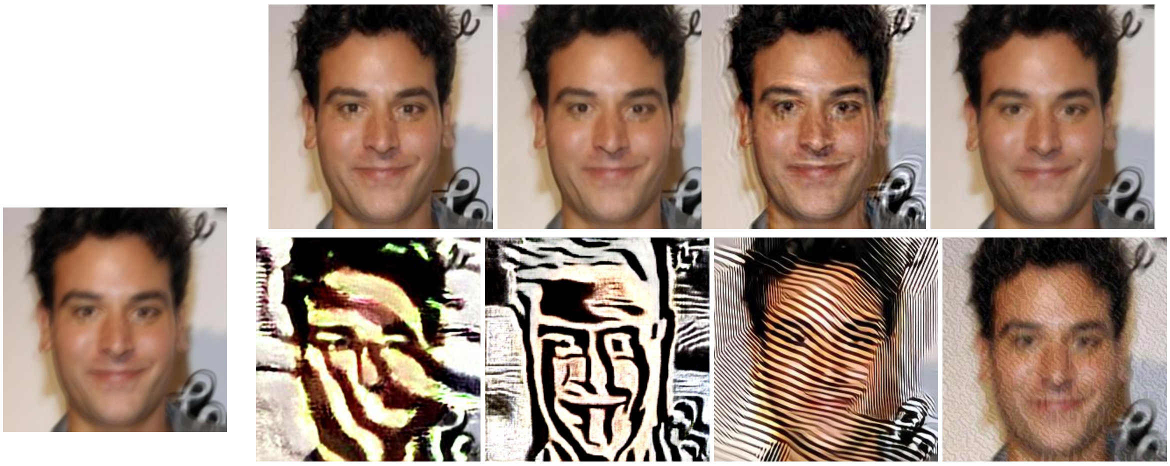
We evaluate the robustness of deblurring networks to targeted attacks that try to make the networks generate an image that is close to a target image. Fig. 6.2 illustrates example targeted

			Targeted attacks						Untargeted attacks		
	Method	Clean	Similarity to source PSNR/NCC			Similarity to target PSNR/NCC			Similarity to source PSNR/NCC		
			$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$	$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$	$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$
Faces	MPRNet	26.81/0.976	9.77/0.409	9.75/ 0.408	9.75/0.408	26.80/ 0.986	26.94/0.987	26.94/0.987	11.29/ 0.645	8.88/0.498	8.582/0.465
	DeblurGANv2	27.13/0.982	10.26/0.419	10.09/0.413	10.081/0.412	20.57/0.950	21.66/0.963	21.76/0.963	5.65/-0.063	5.36/-0.137	5.23/-0.175
	DeepWiener	22.91/0.951	19.28/0.906	17.07/0.858	15.58/0.810	11.31/0.565	12.55/0.659	13.69/0.728	11.65/0.593	10.12/0.511	9.10/0.456
	RGDN	26.98/0.981	24.55/0.961	23.32/0.954	22.06/0.934	10.19/0.452	10.57/0.496	10.95/0.537	24.13/0.966	22.67/0.953	21.43/0.939
BSD	MPRNet	24.41/ 0.944	12.52/0.155	12.30/0.130	12.27/0.126	23.15/0.899	24.14/0.919	24.22/0.921	10.59/0.487	9.04/0.398	8.12/0.319
	DeblurGANv2	24.04/0.941	12.43/0.174	12.35/0.164	12.27/0.153	19.94/0.805	20.92/0.841	21.27/0.854	5.90/0.173	5.64/0.147	5.57/0.085
	DeepWiener	21.44/0.443	21.08/0.882	19.43/0.844	18.08/0.792	13.11/0.139	14.10/0.223	15.00/0.311	14.15/0.607	11.43/0.478	10.15/0.416
	RGDN	24.23/0.943	22.61/0.921	21.78/0.907	20.89/0.887	12.83/ 0.106	13.29/0.166	13.75/0.226	22.08/0.907	21.11/0.884	20.22/0.859

Table 6.1: Comparison of PSNR and normalized cross correlation (NCC) values with respect to source image for untargeted attacks, PSNR and NCC with respect to source and target images for targeted attacks. Evaluation of robustness of MPRNet Zamir et al. (2021), DeblurGANv2Kupyn et al. (2019), Deep Wiener network Dong et al. (2020), RGDN Gong et al. (2020) on blurred face images (top) and blurred images from BSD dataset (bottom).

attacks on the deblurring methods (Zamir et al., 2021; Kupyn et al., 2019; Dong et al., 2020; Gong et al., 2020). Even though the blind deblurring methods (Zamir et al., 2021; Kupyn et al., 2019) are not trained using uniform blur models, they generate sharper images with less visible artifacts when the inputs are clean, and the blur kernels are not too large. However, they are also most susceptible to targeted attacks, shockingly turning a woman into a baby or an elephant into a tiger with adversarial noise strength as low as 4/255. In contrast, the non-blind methods are more robust and do not produce such extreme changes in the output, even for higher strengths of adversarial noise. Adversarial perturbation for the non-blind networks also shows a clear pattern of target images, in contrast to the blind deblurring methods. However, the visual quality of the non-blind network outputs (Dong et al., 2020; Gong et al., 2020) is lower even without adversarial noise. On our test data, the deep Wiener network (Dong et al., 2020) produces sharper results, but with visible ringing artifacts, and the recurrent gradient decent network (Gong et al., 2020) outputs still have a residual blur.

The quantitative evaluation provided in Tab. 6.1 confirms the trend of higher susceptibility of the blind deblurring approaches to targeted attacks, showing higher similarity with the target image in terms of PSNR and normalized cross-correlation (NCC) than with respect to the actual ground truth. While the blind dynamic deblurring approaches (Zamir et al., 2021; Kupyn et al., 2019) are not trained using uniform blur models, we find that similar adversarial vulnerabilities occur even with dynamically blurred images from the test sets of (Zamir et al., 2021; Kupyn et al., 2019), see Fig. 6.3.



Blurred MPRNet $\epsilon = 4/255$ DeblurGANv2 $\epsilon = 4/255$ Deep Wiener $\epsilon = 4/255$ RGDN $\epsilon = 8/255$

Figure 6.5: Untargeted adversarial attack on deblurring networks MPRNet Zamir et al. (2021), DeblurGANv2 Kupyn et al. (2019), Deep Wiener Dong et al. (2020), RGDN Gong et al. (2020). Blur kernel '6' of size 21 × 21 in the dataset of Sun et al. (2013) is used.

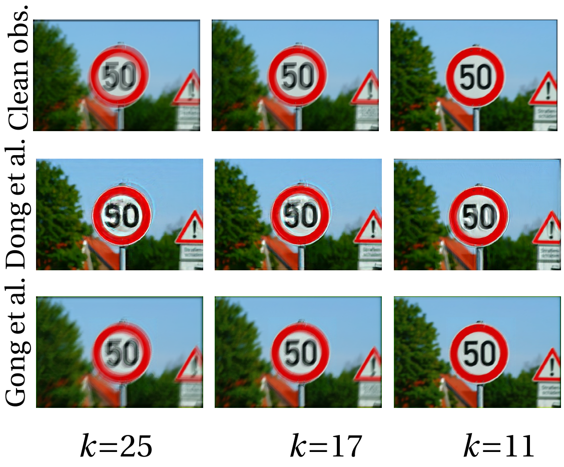


Figure 6.6: Effect of kernel size on ease of targeted attack on non-blind deblurring networks Deep Wiener Dong et al. (2020), RGDN Gong et al. (2020) with $\epsilon = 4/255$.

We now investigate the susceptibility of networks to targeted attacks where the target image is modified at a small localized region. Fig. 6.4 shows such a targeted attack on the deblurring networks, where the target image has the speed limit sign modified from '30' to '90'. The attack on blind deblurring networks is successful even at $\epsilon = 4/255$. Among the non-blind networks, an attack on deep Wiener network (Dong et al., 2020) is clearly successful at $\epsilon = 12/255$, while the target features begin to manifest at even lower values of ϵ. The learned gradient descent approach (Gong et al., 2020) is the most difficult to attack, barely producing target features even for $\epsilon = 12/255$. As the size of the blur kernel becomes larger, deblurring becomes more ill-posed, which can affect the stability of the reconstruction. We investigate this effect by evaluating the adversarial robustness of non-blind networks to targeted attacks by fixing the adversarial noise level to $4/255$, and varying the blur kernel size $\{25, 17, 11\}$. Here the target has a change in the speed limit sign from '50' to '90'. As the blur effect in the input reduces, we expect the networks to be more robust to attacks, which is confirmed by the results in Fig. 6.6. The robustness of deep Wiener deblurring (Dong et al., 2020) improves as the inputs are less and less blurred, and the learned gradient method (Gong et al., 2020) is least susceptible to attack. This robustness however comes at the cost of reduced deblurring performance on clean inputs.

Untargeted Attacks

In Tab. 6.1 and Fig. 6.5 we provide of effect of untargetted attacks on deblurring networks which increase the reconstruction loss. While the blind deblurring networks are highly susceptible to untargeted attacks, we find even the non-blind method of deep Wiener filtering also being unstable, even at low adversarial noise strengths.

In all our experiments, we find that the blind deblurring methods (Kupyn et al., 2019; Zamir et al., 2021) are most susceptible to adversarial perturbations. One reason is that blind deconvolution is inherently more ill-posed, making the reconstruction problem more unstable. Moreover, both the methods (Kupyn et al., 2019; Zamir et al., 2021) use only clean data during training, whereas the methods (Gong et al., 2020; Dong et al., 2020) also add noise to blurry inputs during training. Recent work (Genzel et al., 2022) shows the addition of noise during training as an effective way to improve the adversarial robustness of deep CT reconstruction. However, training with noise is not sufficient to guarantee adversarial robustness, as seen from the results of deep Wiener deconvolution (Dong et al., 2020), which is more prone to attacks than the learned gradient descent approach (Gong et al., 2020), despite being trained with additive noise as augmentation.

6.4.2 *CT Reconstruction*

DATASET We conduct experiments with low-dose parallel beam (LoDoPaB) CT dataset (Leuschner et al., 2021), consisting of data pairs of simulated low-intensity measurements for sampling 513 out of 1000 parallel beams and corresponding ground truth human chest CT images from the LIDC/IDRI dataset (Armato III et al., 2011). In the appendix, we include experiments with LoDoPaB_200 obtained by sampling 200 projection beams from 1000 parallel projection beams for the same dataset of ground truth chest images.

BASELINES We evaluate the robustness of the following approaches: i) Filtered back projection (FBP) ii) FBP-Unet (Chen et al., 2017) post-processing FBP outputs, iii) iRadonmap (He et al., 2020), which also learns back projection in addition to pre-processing, iv) LearnedGD, learned gradient descent (Adler and Öktem, 2017) v) Learned Primal-Dual (Adler and Öktem, 2018) vi) Total Variation regularization. For the learned methods ii)-v), we use the pretrained models[1] from (Baguer et al., 2020) trained on the full training set excluding iRadonmap (which we trained ourselves to full convergence). For FBP, we employ the Hann filter with a low-pass cut-off of 0.6410, the best setting for this dataset in (Baguer et al., 2020). When attacking FBP-Unet, we also backpropagate through the FBP operation. For TV minimization, we used 500 gradient descent steps, with a TV weight of 1e-3, and the attack backpropagates through all the gradient descent steps.

ATTACK SETTINGS We perform untargeted attacks eq. (6.4) using step size of $1e-3$ and 20 PGD steps and choose the best adversarial noise from 5 random restarts. We perform universal attack eq. (6.7) on each method by optimizing a single $L\infty$ norm constrained untargeted adversarial perturbation for hundred examples using step size of $1e-3$ with PGD steps using Adam optimizer for 50 epochs. We perform adversarial attacks effecting localized changes eq. (6.6) using the step size of $1e-3$ and iterate for a maximum of 50 PGD steps till the local patch is misclassified. We choose the best adversarial noise from 5 random restarts. For the localized attacks, we obtain the locations of regions of interest corresponding to ground truth from the LIDC-IDRI dataset. For malignancy classification, we use a Basic ResNet model (Al-Shabi et al., 2019) adversarially trained on nodule patches from the LIDC-IDRI dataset (Armato III et al., 2011). We exclude the images where the patch surrounding the nodule does not lie fully within the central cropped region of the LoDoPAB dataset. For malignancy classification, we consider a 'Basic ResNet' model from (Al-Shabi et al., 2019) for nodule classification, and utilize the adversarially trained model from (Dröge et al., 2022). We consider additive perturbations are L_∞ norm bounded by 1%, 2.5%, and 5% of the intensity range of the clean observation. The code for our experiments is publicly available at `https://github.com/KVGandikota/robustness-low-dose-ct`.

PERFORMANCE METRICS: In the following f, f_δ, $\hat{u}$ and $\hat{u}_\delta$ denote the clean and adversarial sinogram measurements and the corresponding recovered CT images respectively. We measure the PSNR, SSIM, and the TV Bregman distance of the reconstructions with clean and adversarial inputs with respect to the ground truth (setting the corresponding subgradient to zero if the norm of the gradient is below a threshold of 10^{-5}, which we consider to be

[1] `https://github.com/oterobaguer/dip-ct-benchmark`

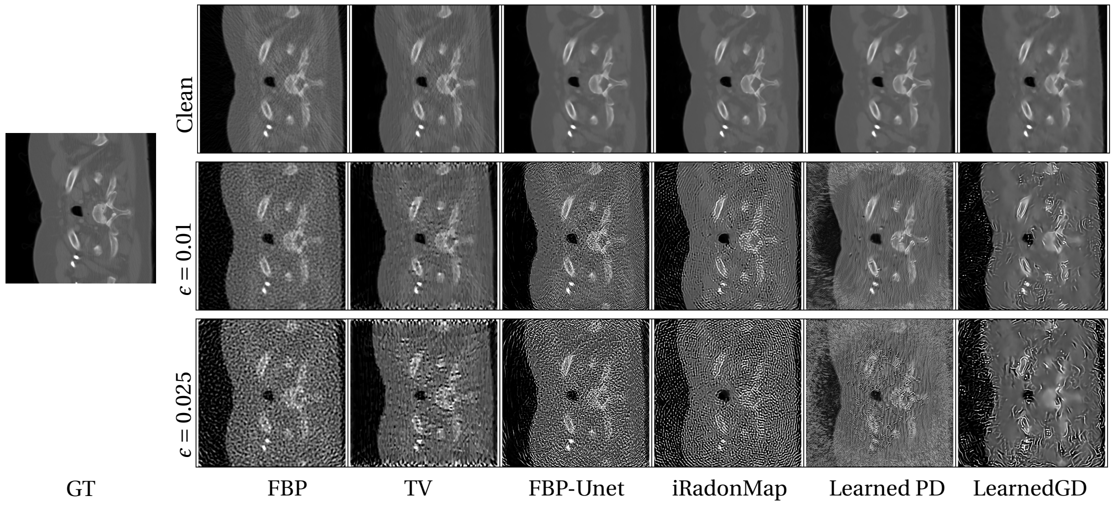


Figure 6.7: Untargeted attack on CT reconstruction methods for ϵ values 0.01 and 0.025.

’numerically zero’). We also measure the data consistency of the reconstructions with respect to the clean and adversarial sinograms in terms of PSNR. Further, we empirically compute a lower bound for Lipschitz constant of each method as

$$L_b(\mathcal{G}) = \left(\frac{\| \mathcal{G}\left(f_\delta\right) - \mathcal{G}\left(f\right) \|}{\|\delta\|} \right)_{\max}$$

which is the maximum value obtained across the test set of 100 CT images for the three adversarial noise levels with 5 random restarts (a total of 1500 examples). For localized attacks, we additionally compare the PSNR values in the local region, and the region exterior to it, for reconstructions with clean and adversarial inputs.

Untargeted Attacks:

Tab. 6.2 and Fig. 6.7 illustrate the results of untargeted attacks eq. (6.4). The results demonstrate that in the absence of adversarial noise, the neural network approaches provide qualitatively better reconstructions than FBP and TV minimization. However, their reconstructions are also more susceptible to adversarial perturbations despite training with inputs corrupted by Poisson noise. Among the deep learning approaches, the learned primal-dual network which provides the best reconstructions from clean inputs is also the most unstable to perturbations, whereas the learned gradient descent is more stable. This is also reflected in the empirical Lipschitz lower bound which is the highest for LearnedPD. This high sensitivity to adversarial attacks is surprising as LearnedPD also encourages data consistency in its (fixed number of) iterations. Among the classical methods, FBP and TV minimization have similar stability in terms of PSNR and L_b, while TV is better in terms of SSIM and Bregman distance as one would have hoped considering the provable stability eq. (6.3). Interestingly, the adversarial perturbations do not heavily affect the data consistency of the recovered CT images for all the methods. The adversarially affected CT reconstructions from LearnedPD with an extremely low average PSNR (0.36 dB) still have a good data consistency (28.7 dB) with the input measurement, showing instabilities typical to unregularized solutions to the recovery problem. Results of similar untargeted attack on the LoDoPAB_200 dataset are provided in Tab. 6.8 of the appendix.

Method	$\hat{u}$	$(A\hat{u}, f)$	ϵ	$\hat{u}_\delta$	$(A\hat{u}_\delta, f)$	$(A\hat{u}_\delta, f_\delta)$	(f, f_δ)	L_b
	PSNR/SSIM/$\mathrm{d}_{\mathrm{Breg}}$	PSNR		PSNR/SSIM/$\mathrm{d}_{\mathrm{Breg}}$	PSNR	PSNR	PSNR	Empir
FBP	30.37/0.738/0.018	33.82	0.01	25.18/0.448/0.029	33.36	33.37	40.20	15.03
			0.025	18.68/0.194/0.049	31.47	31.43	32.51	
			0.05	13.02/0.074/0.081	28.46	28.34	26.91	
TV	31.62/0.763/0.018	36.52	0.01	25.20/0.615/0.026	35.62	35.72	40.36	16.52
			0.025	18.32/0.365/0.044	32.51	33.24	32.71	
			0.05	12.99/0.150/0.077	28.66	30.01	27.22	
FBP-Unet	35.47/0.837/0.013	36.47	0.01	18.39/0.287/0.081	35.06	35.71	40.28	46.71
			0.025	12.18/0.095/0.152	29.82	30.95	32.77	
			0.05	7.38/0.034/0.227	24.86	25.93	27.39	
iRadonMap	33.94/0.810/0.014	36.03	0.01	17.98/0.326/0.062	29.62	29.90	40.22	43.80
			0.025	10.85/0.084/0.140	24.07	24.51	32.60	
			0.05	6.24/0.026/0.215	21.50	21.98	27.16	
LearnedPD	35.73/0.842/0.012	36.46	0.01	9.47/0.164/0.230	25.27	25.50	40.48	143.39
			0.025	3.38/0.030/0.467	23.05	23.38	32.95	
			0.05	0.36/ 0.008/0.623	28.28	28.72	27.17	
LearnedGD	34.55/0.815/0.014	36.43	0.01	21.14/0.504/0.036	35.18	35.62	40.39	30.48
			0.025	13.90/0.291/0.069	31.62	32.82	32.80	
			0.05	8.64/0.180/0.099	28.11	29.64	27.50	

Table 6.2: Comparison of robustness to untargeted attacks on different CT reconstruction methods using 20 attack iterations on first 100 samples LoDoPAB testset.

Source Noise	FBP	FBP-Unet	iRadonMap	LearnedGD	LearnedPD	TV
Clean	30.37/0.738	35.47/0.837	33.94/0.810	34.55/0.815	35.73/0.842	31.62/0.763
FBP	**18.68/0.194**	16.19/0.139	15.41/0.131	16.04/0.138	16.19/0.151	18.32/0.191
FBP-Unet	22.03/0.325	**12.19/0.095**	16.33/0.173	17.98/0.279	14.10/0.125	21.38/0.318
iRadonMap	20.72/0.284	15.18/0.152	**10.86/0.084**	15.45/0.197	16.01/0.171	19.88/0.273
LearnedGD	21.17/0.375	15.42/0.275	15.96/0.271	**13.90/0.290**	15.28/0.241	20.08/0.383
LearnedPD	26.39/0.553	25.33/0.604	26.19/0.590	26.23/0.603	**3.38/0.030**	26.52/0.563
TV	19.19/0.365	16.94/0.289	16.78/0.305	16.66/0.280	16.75/0.333	**18.32/0.365**

Table 6.3: Evaluating transferability of adversarial noises for ϵ=0.025. Bold indicates performance of a model when attacked using noise optimized for the same model.

Transferability of Adversarial Examples:

Transferability of adversarial examples is studied in the context of image classification networks, to examine the possibility of black-box attacks. We investigate the transferability of adversarial examples across different CT recovery methods, i.e. we test whether an adversarial example crafted for a "source" CT recovery method also reduces the quality of reconstruction of a different "target" method for CT recovery. Tab. 6.3 summarizes the results of transferability for CT recovery methods, for ϵ value of 0.025. The results demonstrate that the adversarial examples are indeed transferable across different methods to some extent. The adversarial examples for classical methods FBP and TV are highly transferable across methods significantly reducing the reconstruction quality. The adversarial examples of neural network methods FBP-Unet, iRadonMap, and LearnedGD are also transferable to other network based approaches. While LearnedPD is significantly affected by the adversarial examples generated for other methods, the adversarial examples optimized for LearnedPD are the least effective in reducing the performance of other methods.

Universal Attacks & Transferability:

We perform input-agnostic attack universal attack eq. (6.7) by optimizing over a set of 100 samples, by optimizing over 50 epochs. Tab. 6.4 shows the effect of this adversarial perturbation on the optimized examples and its generalizability on a different 100 examples not seen during optimization. The results indicate that CT recovery methods can also be affected by universal attacks. Further, these input-agnostic perturbations are more effective than the input-specific perturbations in reducing the performance of the reconstruction algorithm, even though the perturbation is not optimized for the specific example. This trend is seen across the reconstruction methods, with universal perturbation resulting in a further reduction of around 2 dB PSNR compared to the input-specific perturbations. This effect is possibly due to the significantly increased number of total attack iterations.

In addition to input-specific adversarial examples, we also study the transferability of input-agnostic universal perturbations across different CT recovery methods. Tab. 6.5 summarizes the results of such transferability test for ϵ value of 0.05. The results indicate that even universal adversarial perturbations are transferable across different methods. This indicates the possibility of crafting fully black box attacks on CT recovery. The universal perturbations optimized for FBP and TV are the most transferable to other methods, indicating the vulnerabilities of the classical approaches are also shared by the end-to-end trained deep networks. Among the end-to-end trained networks, the universal perturbations optimized for iRadonMap are the most transferable to other networks and even the classical methods, despite being a fully learned approach. In contrast, universal perturbation optimized for LearnedPD is least transferable to other methods.

		FBP	FBP-Unet	iRadonMap	LearnedGD	LearnedPD	TV
Optimized	Clean	30.37/0.738	35.47/0.837	33.95/0.810	34.55/0.815	35.73/0.843	31.62/0.763
	$\epsilon = 0.01$	22.87/0.340	17.96/0.223	15.58/0.263	18.41/0.542	7.19/0.139	22.86/0.408
	$\epsilon = 0.025$	15.73/0.116	9.93/0.055	8.24/0.031	10.07/0.308	0.401/0.013	15.03/0.120
	$\epsilon = 0.05$	9.87/0.036	4.49/0.023	3.303/0.011	3.80/0.179	-3.71/0.003	8.76/0.032
Unseen	Clean	30.53/0.714	35.67/0.824	34.19/0.799	34.74/ 0.802	35.92/0.829	31.87/0.750
	$\epsilon = 0.01$	23.27/ 0.337	18.59/0.225	16.29/ 0.262	19.04/0.538	7.93/0.161	23.27/0.404
	$\epsilon = 0.025$	16.19/0.115	10.43/0.0525	8.80/0.030	10.64/0.309	1.24/0.016	15.47/0.119
	$\epsilon = 0.05$	10.34/ 0.036	4.95/0.022	3.82/0.0108	4.32/0.183	-2.95/0.003	9.24/0.031

Table 6.4: Universal adversarial attack on CT recovery. PSNR/SSIM values for clean samples and samples affected by additive universal perturbation are shown.

Source Noise	FBP	FBP-Unet	iRadonMap	LearnedGD	LearnedPD	TV
Clean	30.53/0.714	35.67/0.824	34.19/0.799	34.74/ 0.802	35.92/0.829	31.87/0.750
FBP	**10.34/0.036**	9.90/0.031	8.74/0.025	7.68/0.021	10.62/0.041	9.28/0.031
FBP-Unet	14.42/0.098	**4.95/0.022**	9.06/0.035	9.26/ 0.095	7.77/0.042	12.21/0.077
iRadonMap	13.02/0.0706	9.61/0.049	**3.82/0.0108**	7.38/0.042	10.99/0.057	10.95/0.052
LearnedGD	15.60/0.188	13.52/0.220	10.38/0.112	**4.32/0.183**	9.69/0.109	12.68/0.170
LearnedPD	23.07/0.358	21.42/0.444	19.45/0.232	23.54/0.453	**-2.95/0.003**	20.76/0.261
TV	10.38/0.037	9.76/0.032	8.62/0.025	7.61/0.022	10.54/0.042	**9.24/0.031**

Table 6.5: Evaluating transferability of universal adversarial noises for ϵ=0.05.

Localized Attacks:

Tab. 6.6 and Fig. 6.8 provide the results of our experiments with localized attacks eq. (6.6) on different CT recovery methods. Sample reconstructions from different methods with

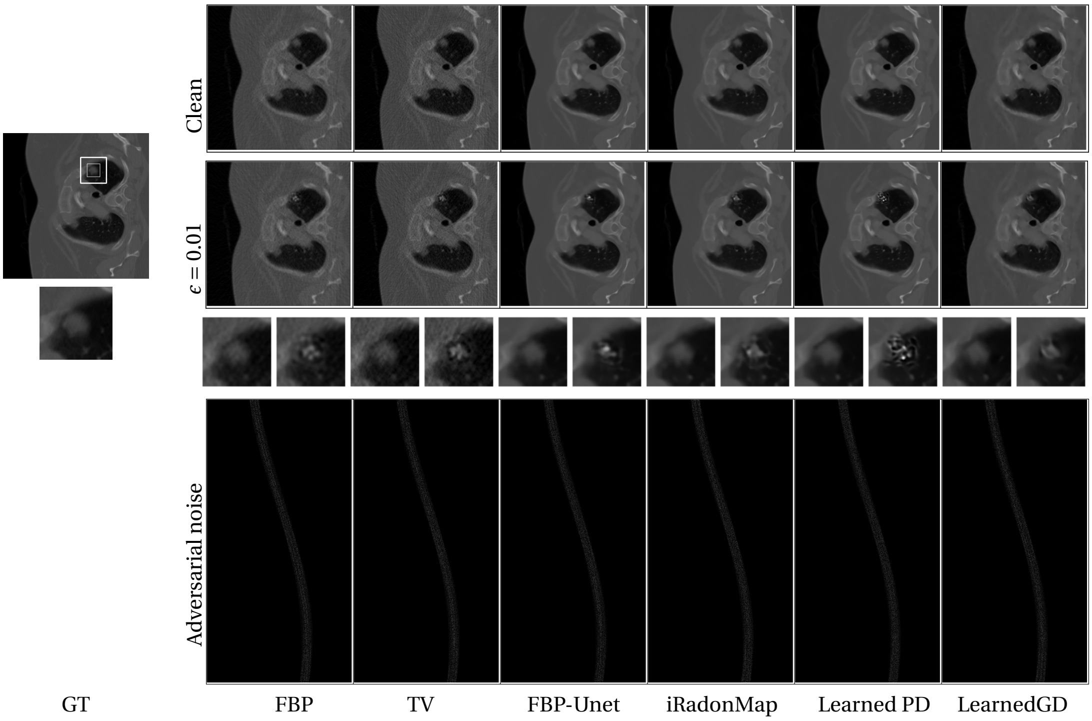


Figure 6.8: Localized attack on CT reconstruction methods. for $\epsilon = 0.01$. First and second row illustrate clean and adversarial reconstructions for each method. The third row shows the cropped patches from the clean (left) and adversarial (right) reconstructions. Adversarial noise in the fourth row is multiplied by ×25 for visibility.

clean and adversarial inputs are compared in Fig. 6.8. The adversarial noise that produces the localized changes is also depicted. The results clearly demonstrate visible alteration in the region of interest $\hat{u}_i$ indicated by the inner square marked in the ground truth image. Our attack successfully achieves this modification, barely affecting the reconstruction in the exterior region $\hat{u}_e$ using an extremely low noise level.

Tab. 6.6 summarizes our results for localized attacks for three levels of adversarial noise. The subscripts i and e denote the restriction to the interior and exterior of the local region to be attacked. Due to masking, the magnitudes of additive perturbation are extremely small, with high PSNR values between the clean and adversarial inputs for all noise levels. Still, our attack is almost always successful in producing local degradations that change the malignancy prediction (100% success rate on our test set for all the methods). This is also reflected in the steep PSNR drop in the local region $\hat{u}_i$, while the PSNR in the exterior region is mostly unaffected. While the classical approaches are more robust to untargeted attacks, they are also sensitive to local changes. This is a direct consequence of the ill-posedness of the recovery problem, as we observe nearly similar data consistency of the recovered $\hat{u}_\delta$ with both clean and adversarial inputs. In a recent work Dröge et al. (2022) demonstrate that the CT images of varying malignancy levels can be solutions to the same measurement with a high data consistency, but by modifying the reconstruction loss. Our localized attacks also show that the adversarial noise necessary to change the malignancy is extremely small for a variety of methods and the resulting solutions demonstrate high data consistency with both clean and adversarial inputs. While one way to use adversarial attacks beneficially is to use them in training, i.e. adversarial training to make models robust, our work shows another beneficial application of attacks. One could utilize localized attacks to efficiently explore diagnostically different reconstructions with a very high degree of data consistency with

Method	$\hat{u}$	$\hat{u}_i\|\hat{u}_e$	$(A\hat{u}, f)$	ϵ	$\hat{u}_\delta$	$\hat{u}_{\delta_i}\|\hat{u}_{\delta_e}$	$(A\hat{u}_\delta, f)$	$(A\hat{u}_\delta, f_\delta)$	(f, f_δ)
	PSNR/SSIM	PSNR	PSNR		PSNR/SSIM	PSNR	PSNR	PSNR	PSNR
FBP	30.86/0.787	31.45|30.86	33.81	0.01	30.60/0.782	22.29|30.83	33.79	33.77	55.09
				0.025	30.35/0.772	20.93|30.67	33.75	33.42	47.55
				0.05	29.97/0.751	19.89|30.34	33.70	32.63	41.08
TV	32.36/0.829	31.84|32.37	36.52	0.01	32.00/0.825	22.70|32.32	36.48	36.42	54.77
				0.025	31.62/0.812	21.26|32.07	36.46	35.66	46.97
				0.05	30.65/0.767	20.28|31.15	36.35	33.59	40.11
FBP-Unet	36.94/0.909	35.67|36.95	36.50	0.01	34.85/0.902	19.43|36.61	36.46	36.43	55.11
				0.025	33.79/0.889	17.82|35.87	36.37	35.84	47.83
				0.05	33.15/0.877	17.27|35.11	36.11	34.42	41.90
iRadonMap	35.25/0.888	34.07|35.27	36.09	0.01	33.70/0.883	18.85|35.12	36.03	36.03	55.32
				0.025	32.68/0.875	16.53|34.76	35.95	35.52	48.08
				0.05	30.60/0.808	15.32|32.73	35.55	33.39	40.81
LearnedPD	37.22/0.913	35.97|37.23	36.49	0.01	33.15/0.854	18.34|35.08	36.28	36.10	53.74
				0.025	29.90/0.753	16.15|31.57	35.33	34.57	45.41
				0.05	25.05/0.559	14.52|25.72	33.29	31.74	38.41
LearnedGD	35.80/0.891	34.86|35.82	36.49	0.01	34.86/0.886	22.02|35.71	36.46	36.42	55.29
				0.025	34.49/0.883	20.98|35.53	36.42	35.99	48.41
				0.05	34.12/0.875	21.11|35.04	36.28	34.72	42.44

Table 6.6: Comparison of robustness to localized attacks on different CT reconstruction method evaluated on 100 samples LoDoPAB testset.

sinogram. This can be used by a medical doctor to choose the most plausible reconstruction in making a diagnosis.

6.5 DISCUSSION AND CONCLUSIONS

In this chapter, we evaluated the adversarial robustness of image recovery for two example tasks, image deblurring and low dose CT reconstruction. We showed that state-of-the-art deep networks for deblurring can be highly susceptible to adversarial attacks, even drastically changing the reconstruction to a different target image, which is clearly inconsistent with the measurement. In contrast, non-blind networks are less susceptible to such drastic changes, yet, they are affected be local changes and untargeted attacks.

Our analysis of the adversarial robustness of CT recovery shows that deep learning methods are more sensitive to untargeted adversarial examples than classical approaches. Even model-inspired unrolled networks are susceptible to adversarial examples, even though they encourage data consistency within the network. While the quality of the recovered CT images degrades, we find that the recovered images still exhibit a good degree of consistency with the measurements. In contrast, restoration networks can produce images which are highly inconsistent with the measurements. This difference in robustness with respect to measurement consistency could be due to the fact that restoration networks have to cope with varying degradation operators across examples. Even if we consider non-blind deblurring networks, the blur operator is different for different examples. The problem becomes more severe for blind dynamic deblurring networks. In contrast, CT recovery networks are trained for the same forward operator for the entire training data, and use the same forward operator during testing with natural and adversarial inputs. This could possibly explain why the reconstructions have a reasonably high measurement consistency under attack, even for methods that do not take into account the forward operator (for instance, iRadonMap). It is therefore interesting to study the worst-case performance when there are uncertainties in the knowledge of the forward operator, or when there is a drift in the forward operator in the test phase when compared to training.

In addition to measurement consistency, it is also desirable that the reconstruction methods produce realistic results, even when inputs are perturbed by a tiny amount. We find that image recovery networks fail at this, even though they are augmented with noise during training. This underlines the importance of improving the adversarial robustness of image recovery methods through better regularization, robust training, or by developing more robust architectures. We demonstrated that adversarial perturbations can be transferable across CT recovery methods. Further, we showed the feasibility of universal attacks, and showed that perturbations can transfer across different CT recovery methods, indicating the possibility of fully black box attacks on image recovery. Interestingly the universal perturbations crafted for the classical approaches, FBP and TV minimization are the most transferable to network based methods, even to the methods not utilizing the forward operator, indicating that these methods share some common directions of vulnerability.

We also find that the classical methods and deep learning methods for CT recovery are similarly affected by adversarial examples targeting small localized regions. Moreover, such attacks are successful for extremely small perturbations already, such that the resulting reconstructions have high data consistency with original measurements. Therefore, the proposed localized attacks could can aid in dealing with uncertainties in ill-posed image recovery, by allowing exploration of the solution space of the reconstruction problem.

APPENDIX

6.A ADDITIONAL RESULTS FOR IMAGE DEBLURRING

Method	Clean	Targeted attacks						Untargeted attacks		
		Similarity to source PSNR/NCC			Similarity to target PSNR/NCC			Similarity to source PSNR/NCC		
		$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$	$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$	$\epsilon = 4$	$\epsilon = 8$	$\epsilon = 12$
MIMO-UNet	24.84/0.967	9.78/0.412	9.71/0.404	9.70/0.403	25.55/0.983	26.85/0.988	27.00/0.989	5.45/0.046	5.26/0.006	5.26/0.006
Uformer	24.32/0.963	10.44/0.470	10.05/0.438	10.00/0.433	22.38/0.961	24.36/0.979	24.86/0.982	6.74/0.092	5.97/-0.053	5.56/-0.163
Restormer	24.37/0.963	9.84/0.413	9.76/0.406	9.75/0.406	27.38/0.988	28.88/0.992	28.94/0.993	5.54/0.016	5.39/0.004	5.39/0.004
Stripformer	23.38/0.967	13.01/0.597	12.33/0.554	12.08/0.534	13.34/0.725	14.66/0.726	15.17/0.800	10.36/0.546	9.44/0.493	8.97/0.441
NAFNet	24.62/0.965	11.75/0.543	11.11/0.511	10.98/0.503	18.19/0.887	18.99/0.915	19.17/0.919	11.76/0.382	5.12/0.045	5.09/0.045

Table 6.7: Comparison of PSNR and normalized cross correlation (NCC) values with respect to source image for untargeted attacks, PSNR and NCC with respect to source and target images for targeted attacks, for MIMO-UNet Cho et al. (2021), Uformer Wang et al. (2022c), Restormer Zamir et al. (2022), Stripformer Tsai et al. (2022), NAFNet Chen et al. (2022a) on blurred face images.

Tab. 6.7 summarizes the results of our experiments evaluating the adversarial robustness of more recent dynamic deblurring networks MIMO-UNet (Cho et al., 2021), Uformer (Wang et al., 2022c), Restormer (Zamir et al., 2022), Stripformer (Tsai et al., 2022) and NAFNet (Chen et al., 2022a) on a set of blurred face images. The results in Tab. 6.7 indicate that these dynamic deblurring networks are highly susceptible to adversarial examples. Further, the networks are also susceptible to targeted attacks, showing a higher similarity with the target image than the actual ground truth both in terms of PSNR and NCC. Yet, the degree of susceptibility varies across the different networks. Among the evaluated networks, Stripformer (Tsai et al., 2022) demonstrates better robustness to both targeted and untargeted attacks, yet it has a poorer performance on clean inputs. In terms of robustness to targeted attacks, the networks MIMO-UNet (Cho et al., 2021), and Restormer (Zamir et al., 2022) are the least robust, producing images highly similar to the target, followed by Uformer (Wang et al., 2022c) and NAFNet

Method	$\hat{u}$	$(A\hat{u}, f)$	ϵ	$\hat{u}_\delta$	$(A\hat{u}_\delta, f)$	$(A\hat{u}_\delta, f_\delta)$	(f, f_δ)	$\|\delta\|^2$	L_b
	PSNR/SSIM	PSNR		PSNR/SSIM	PSNR	PSNR	PSNR		Empir
FBP	28.38/0.649	34.14	0.01	25.26/0.465	33.69	33.69	40.03	0.093	29.69
			0.025	19.77/0.233	31.84	31.75	32.09	0.581	
			0.05	14.36/0.096	28.78	28.52	26.12	2.292	
TV	28.94/0.652	37.47	0.01	24.88/0.520	36.58	36.54	40.10	0.092	33.98
			0.025	18.91/0.302	33.32	33.74	32.20	0.565	
			0.05	13.72/0.126	29.13	30.16	26.33	2.177	
FBP-Unet	33.55/0.799	36.50	0.01	19.37/0.384	34.52	35.244	40.14	0.091	97.56
			0.025	12.82/0.115	28.33	29.23	32.31	0.551	
			0.05	8.38/0.036	23.26	23.97	26.52	2.074	
iRadonMap	32.39/0.778	36.3	0.01	18.46/0.546	30.22	30.58	40.08	0.092	125.21
			0.025	9.40/0.231	19.55	19.88	32.27	0.554	
			0.05	5.39/0.051	14.92	15.12	26.65	2.01	
LearnedPD	33.64/0.802	36.50	0.01	17.75/0.412	34.23	34.92	40.11	0.092	108.48
			0.025	10.56/0.153	31.26	33.08	32.34	0.548	
			0.05	5.94/0.053	31.91	33.66	26.57	2.047	
LearnedGD	32.49/0.776	36.46	0.01	22.44/0.583	35.41	35.67	40.35	0.086	61.95
			0.025	15.66/0.418	32.09	33.01	32.72	0.499	
			0.05	10.89/0.301	29.10	30.02	27.19	1.773	

Table 6.8: Comparison of robustness to untargeted attacks on different CT reconstruction methods using 20 attack iterations on 100 samples LoDoPAB200 testset.

(Chen et al., 2022a) with decreasing similarity with respect to the target, as evidenced by the lower PSNR and NCC values with respect to the target. NAFNet demonstrates better robustness to untargeted attacks at lower adversarial noise strength of 4/255, yet becomes more susceptible to adversarial noises of higher strength. This wide variability in the robustness of dynamic deblurring networks calls for a more thorough investigation into the architectures and training protocols to understand their effects on robustness. We leave this to future work.

6.B ADDITIONAL RESULTS FOR CT RECOVERY

Untargeted Attacks on LoDoPAB_200 Tab. 6.8 summarizes the results of our untargeted attacks on the LoDoPAB_200 dataset, where the measurements are generated using 200 projection beams. Similar to our results on the LoDoPAB dataset, we find that classical approaches are more robust to untargeted attacks. However, on this dataset, the fully learned approach of iRadon Map is the most unstable method, followed by LearnedPD. LearnedGD is stable among the network based methods. Further, the methods show a general trend of a higher value of L_b on LoDoPAB_200 dataset in comparison with LoDoPAB dataset indicating higher instabilities as the reconstruction from 200 projection beams is more severely ill-posed than from 513 projections.

Qualitative Results Fig. 6.9 shows the results of localized attacks on 20 example CT images in the LoDoPAB test set. For each method, the local patches extracted from clean and adversarial reconstructions are shown.

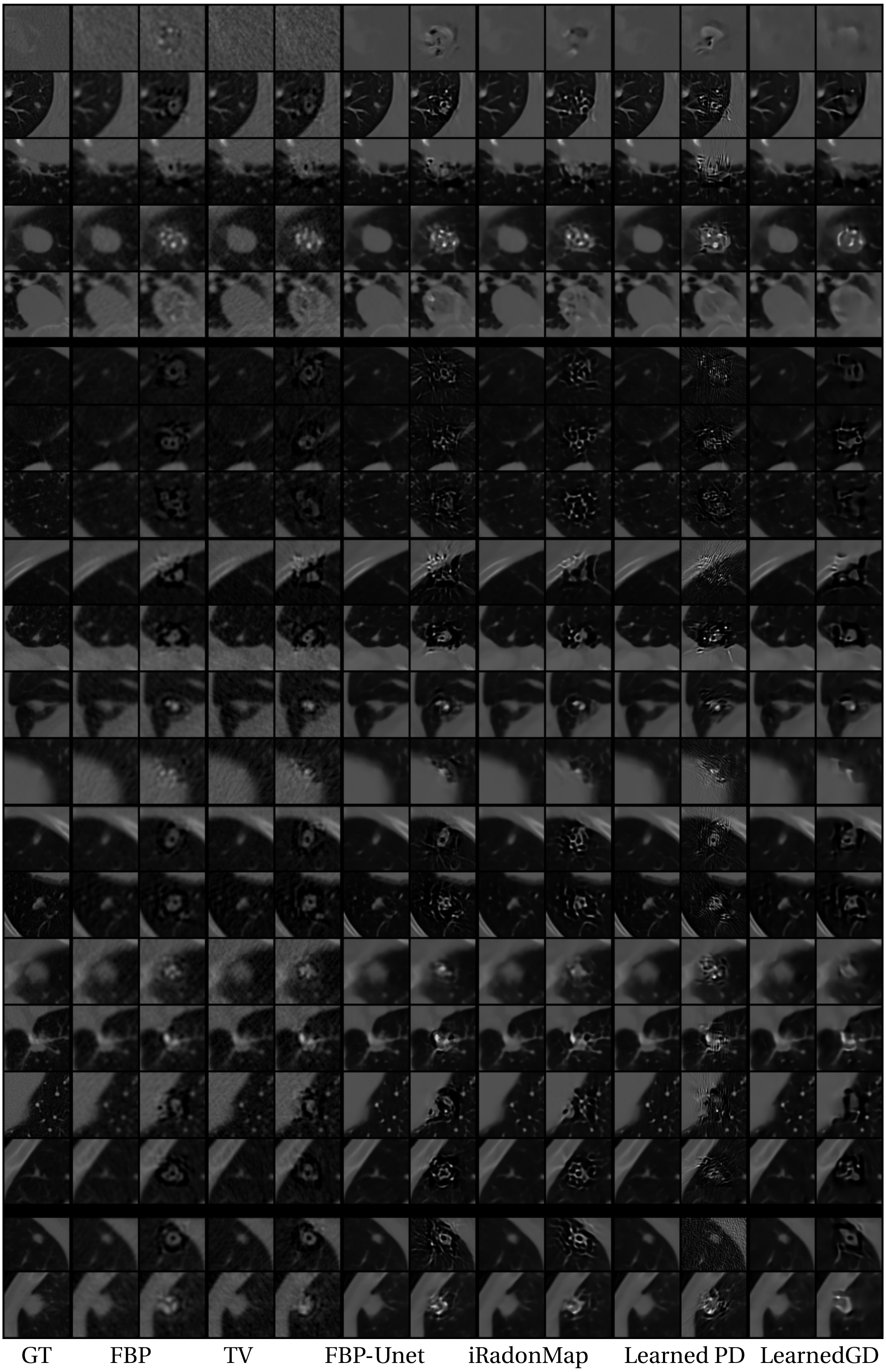


Figure 6.9: Result of localized attacks on 20 images. For each method left patch is from clean reconstruction and right is the result of attack.

Declaration for Chapter 7 - Light Field Reconstruction with Generative Priors

This chapter is based on the paper Chandramouli et al. (2022) titled "A generative model for generic light field reconstruction" co-authored by Paramanand Chandramouli, Kanchana Vaishnavi Gandikota, Andreas Goerlitz, Prof. Andreas Kolb, and Prof. Michael Moeller, published in IEEE Transactions on Pattern Analysis and Machine Intelligence, (TPAMI) April 2022. Paramanand Chandramouli and Kanchana Vaishnavi Gandikota are joint first authors of this paper.

Paramanand Chandramouli proposed this project idea of building a generative model for light fields, and using it as a prior in light field recovery. Paramanand Chandramouli contributed to training the generative models- an unconditional Wasserstein autoencoder, and subsequently a conditional Wasserstein autoencoder for light field patches. He proposed and implemented an encoder-decoder based optimization scheme for light field recovery, and evaluated the baseline methods. Kanchana Vaishnavi Gandikota implemented the optimization using the conditional generator which showed improvement over the former approach. Paramanand Chandramouli contributed to writing related work on generative models, description of model architecture and training. Kanchana Vaishnavi Gandikota contributed to writing introduction, related work on light field recovery, describing light field recovery algorithm, experimental setup, conducting experiments for light field recovery from different forward measurement processes, and writing up the results. Prof. Andreas Kolb and Prof. Michael Moeller provided several suggestions, and improved the structure and clarity of the writing. The research was supervised by Prof. Michael Moeller.

7 LIGHT FIELD RECONSTRUCTION WITH GENERATIVE PRIORS

In the previous chapter, we analyzed the stability of image recovery methods through worst-case additive perturbations. In this chapter, we utilize generative priors for image recovery to deal with lack of generalizability in end-to-end trained networks. As discussed in Chapter 3, end-to-end trained networks for inverse problems suffer from a lack of generalizability, and cannot handle changes in noise or measurement process. On the other hand, variational approaches, including those using neural network based priors can adapt to such changes by suitable modifications to energy function in model based optimization. Among these approaches, we adopt the approach of using generative model priors to address the problem of generalizing deep learning based image reconstruction for different measurement models. Specifically, we investigate the use of generative auto-encoders as priors in model based image reconstruction. As these models are trained without the supervision of specific measurement models, they can be incorporated as a prior into model-based optimization and therefore extend to diverse reconstruction tasks. In this chapter, we demonstrate the utility of generative auto-encoder priors, for light field recovery from diverse measurement models.

We consider 4D light fields which capture a scene from different viewpoints along a plane. Acquiring the high dimensional light fields is an involved process, and as we will see in section 7.1.1, there are different ways to acquire measurements from which light fields can be computationally reconstructed. As discussed in Chapter 3, classical variational approaches to such inverse imaging problems determine the solution as the minimizer of an energy function composed of a data discrepency term and a suitable regularizer which characterizes the desirable properties of light fields, such as a learned dictionary on light fields (Marwah et al., 2013). Several recent works (Kalantari et al., 2016; Gupta et al., 2017; Inagaki et al., 2018; Yeung et al., 2018; Vadathya et al., 2019) instead learn a deep neural network to map from the measurements to light fields by training on paired datasets of measurement and ground truth light fields, and achieved significant improvements in reconstruction performance compared to classical approaches for the specific trained task. Yet, these lack the flexibility of classical methods, and have to be retrained as soon as the measurement process or the noise statistics change. While there exist several hybrid approaches which exploit the expressive power of neural networks without losing the flexibility of energy minimization methods, by using neural networks as priors in energy minimization. Interestingly, such approaches have not yet been exploited for light field recovery, most likely due to the high complexity of light field data.

In this chapter, we attempt to fill in this gap by introducing the first generative model for light field data and using it as a prior for generic light field reconstruction tasks. The key idea is to model the distribution of light fields using a class of generative autoencoders (Tolstikhin et al., 2018). Once the training is complete, we use our generative model as a prior in different light field reconstruction problems in an energy minimization framework. Due to the high complexity and variability of the light field data, generating light fields in a consistent fashion is highly challenging. In this chapter, we consider only *small baseline light fields*, and we address this challenge by training a conditional Wasserstein autoencoder (CWAE) for light field patches, with the decoder conditioned on the central patch. The advantage of our approach is that the conditional generative model learned on patches can readily generalize to a variety of scene content, while being small enough to be amenable for training.

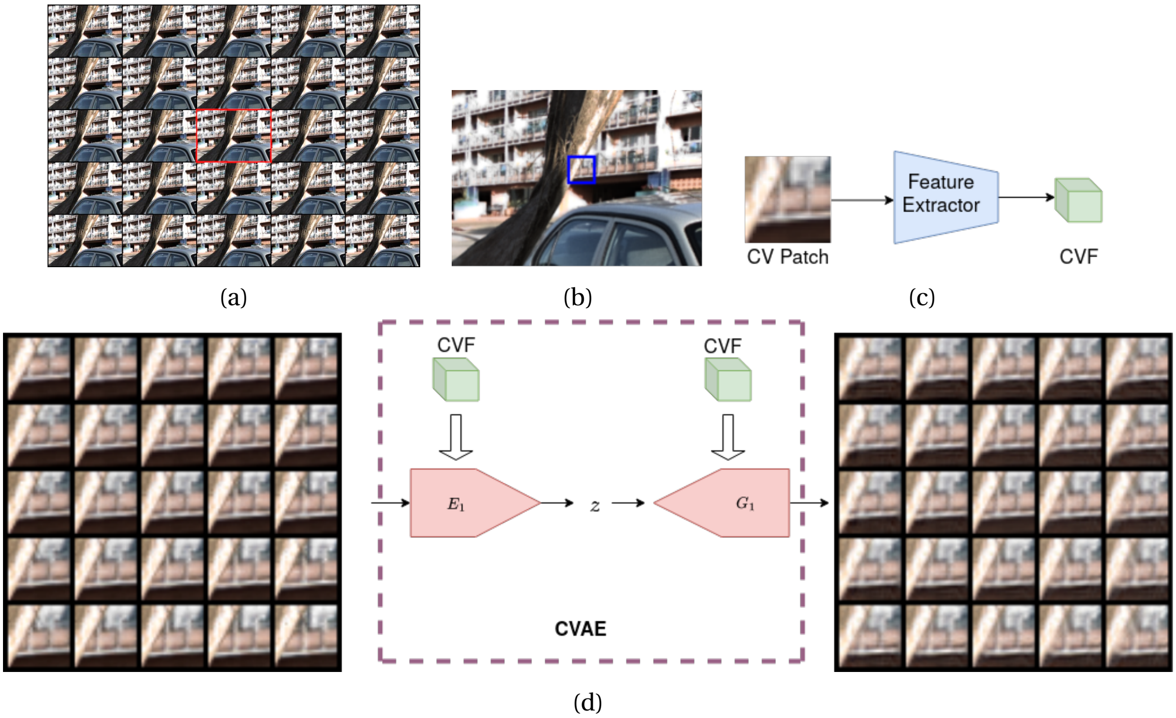


Figure 7.1: (a) A full 5 × 5 LF, with central view marked in red. (b) Central view (CV) extracted from (a), with a small patch of this central view marked in blue. (c) This patch passes through a convolutional feature extractor to output central view features (CVF). (d) The encoder E_1 of the CWAE maps an LF patch to a latent variable z, while the generator G_1 of the CWAE maps z back to the LF patch using CVF as an additional input.

Fig. 7.1(d) shows the schematic of the CWAE. The CWAE, consists of an encoder E_1 that takes an LF patch as input and returns a low-dimensional latent code z. The generator G_1 maps this latent code back to the LF patch. A convolutional feature extractor Fig. 7.1(c) provides features of the central view of the light field patch as an additional input to both the encoder and generator of the CWAE. Consequently, both the encoder and the generator utilize the information from the central patch. In the reconstruction of the light field patch shown in Fig. 7.1 (d), we observe that the generator can map the encoded latent variable along with the features of the central view to a light field patch which looks similar to the input light field patch. This indicates that the encoder has learned to encode properties such as disparity and occlusion in the latent space, such that the generator can reconstruct the LF patch just from this latent code and the central view features.

We utilize this model as a prior in light field reconstruction. We take the approach of optimizing in the latent space of CWAE generator to minimize data discrepency with respect to the measurement, and perform simultaneous optimization of both the latent code and the central view when the latter is unavailable. We perform diverse light field recovery tasks including light field view synthesis, spatial-angular super resolution, and reconstruction from coded projections. We demonstrate the advantages of the proposed approach in comparison with end-to-end trained networks in terms of flexibility and robustness to corruptions, and improved performance with respect to traditional model-based approaches on both synthetic and real scenes.

7.1 PRELIMINARIES

We begin with an introduction to light fields and the associated inverse imaging problems.

7.1.1 *Light Fields*

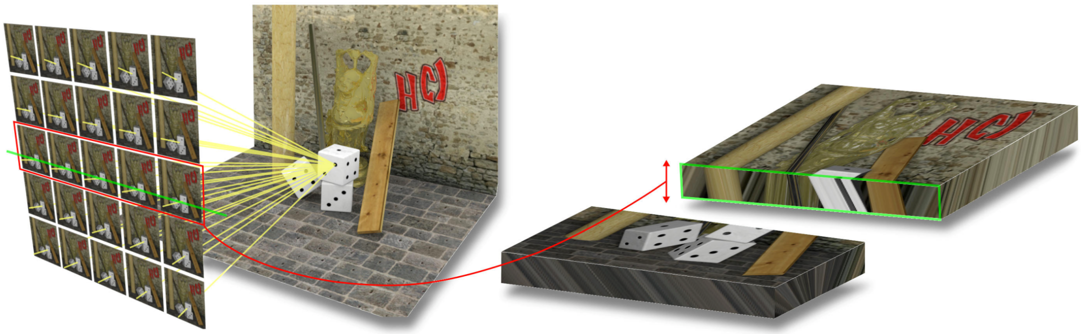


Figure 7.2: Illustration of a discrete 4D light field using a two-plane parameterization. On the right, epipolar plane image(EPI) obtained by slicing all the images in a light field along a scanline is shown. Image from light field archive of Heidelberg collaboratory for image processing.

A light field (LF) is a vector function that models light rays in a scene as a function of position, orientation, time, and frequency. The space of all possible light rays is given by the seven-dimensional plenoptic function (three spatial coordinates (x, y, z), two angles indicating direction (θ, ϕ), time and frequency), and the magnitude of each ray is given by its radiance. In this chapter, we restrict ourselves to four-dimensional static light fields (Levoy and Hanrahan, 1996), which can be represented using a two-plane parameterization by constraining each ray to have the same value at every point along its direction of propagation. The 4D light fields are represented using the continuous plenoptic function $L(\mathbf{x}, \mathbf{v})$ that denotes the radiance of the scene emitted at the spatial position $\mathbf{x}$ and in the angular direction $\mathbf{v}$. Instead of frequency, we have three colour channels typical of digital colour representation, yet, the three colour channels are excluded when counting the dimensionality of the light field, as these are generally treated independently as three 4D signals. Light fields can accurately model geometrically complex scenes, and have a wide range of applications such as the precise free viewpoint rendering of a 3D scene or the estimation of geometries or materials of objects in a scene.

In practice, it is not possible to acquire a continuous light field, rather it is common to acquire a discrete version of the light field, where a scene is captured from a discrete set of viewpoints on a plane. Discrete 4D light field can be captured using exhaustive and expensive hardware setups comprising dozens of cameras in a camera-rig (Wilburn et al., 2005), or by using *plenoptic cameras* that utilize microlens arrays placed in front of the imager of a standard 2D camera (Ng et al., 2005), or by using coded mask based cameras (Liang et al., 2008a). While camera-rigs allow for larger baselines with full spatial resolutions as that of image sensor, they have a rather sparse angular resolution. In contrast, plenoptic cameras allow recording dense light fields with a single exposure, yet they are restricted to capturing a rather small baseline, and they have a trade-off between the spatial resolution of each sub-aperture image (view obtained from a specific angular coordinate) and the angular resolution of the light field. On the other hand, coded aperture or mask based cameras can capture a light field with the same spatial resolution as that of an image sensor, but there is a trade-off between quality of the light field and the number of acquired images.

One could address the different trade-offs in each of these acquisition setups, and computationally improve the quality and resolution of the captured light fields. This could involve improving the angular resolution using view synthesis methods, or improving both the spatial and the angular resolution of the captured light fields, or improving the quality of recovered light fields from coded measurements. Let us consider a discretized version of light field $L(\mathbf{x}, \mathbf{v})$, where the angular resolution for each axis is N_v, and the spatial resolution of each view is $N_x \times N_y$. In this chapter, we consider 3 different LF reconstruction problems: (i) Light field view synthesis/ view upsampling, (ii) Spatial-angular super-resolution, and (iii) Light field recovery from coded aperture images. We now consider the specific forward measurement process for each of these reconstruction tasks.

(i) *View synthesis / Angular super-resolution:* The task of view synthesis is to recover a dense set of sub-aperture images (SAIs) from a sparse subset of input views. The forward model can be considered to be a point-wise multiplication of the light field with a binary mask M, whose value is 1 at the known views, and 0 at all other locations, leading to

$$f(\mathbf{x}, \mathbf{v}) = L(\mathbf{x}, \mathbf{v}) \odot M(\mathbf{x}, \mathbf{v}). \tag{7.1}$$

where $\odot$ is the point-wise multiplication operator.

(ii) *Spatial and angular super-resolution using central view:* Here the task is to recover all SAIs from a sparse subset of spatially down-sampled input views. The corresponding forward measurement model can be written as

$$f(\mathbf{x}, \mathbf{v}) = (L(\mathbf{x}, \mathbf{v}) \odot M(\mathbf{x}, \mathbf{v}))_{\downarrow_{s(\mathbf{v})}}. \tag{7.2}$$

where M is a binary mask which is non-zero only at known views, and $\downarrow_{s(\mathbf{v})}$ is the spatial down-sampling operation of the known views. We assume that the central view is available in full resolution which aids in spatial upsampling of novel views, i.e. the downsampling factor is 1, for the central view.

(iii) *Coded aperture:* Coded aperture images are the result of optical multiplexing only along angular dimension. In a continuous setting, the coded aperture image formation model can be written as

$$f(\mathbf{x}) = \int L(\mathbf{x}, \mathbf{v}) M(\mathbf{v}) d\mathbf{v} \tag{7.3}$$

where M represents the coded mask, which depends on the angles $\mathbf{v}$, but not on the spatial position. In the discrete setting, the measurement is a weighted sum of the view images, with weights provided by the coded mask.

Recovering good quality light field with high spatial and angular resolution from the measurements in each of these settings eq. (7.1), eq. (7.2), eq. (7.3) can be posed as a linear inverse problem of the form

$$f = A(u) + n, \tag{7.4}$$

where u represents the discrete light field to be recovered, A is the problem-dependent linear operator and n is the additive noise. In this chapter, we attempt to solve these problems using deep generative models trained on light field data in an energy minimization framework.

As discussed in Chapters 3 and 4, generative models can be used as priors in various image reconstruction and manipulation tasks (Bora et al., 2017; Li et al., 2017; Bau et al., 2019a). We briefly recall the approaches relevant to this chapter. Bora et al. (2017) propose to model the solution to inverse problem as belonging to the range manifold of the generative model, and optimize in the latent space of the generative model (GAN or VAE generator) for a latent code which maps to an image that minimizes data discrepency using gradient descent based updates. More sophisticated optimization schemes such as projected gradient descent, ADMM have also been used in conjunction with GAN priors for optimization in the latent space (Shah and Hegde, 2018; Hegde, 2018; Latorre et al., 2019b). The problem with restricting the solution to the range of the generator is that it leads to non-trivial representation error, even when the measurement operator is an identity, as the generator cannot accurately represent any image. In this chapter, we propose the approaches of latent space optimization using a conditional generator with application to light field recoverywhich can alleviate the issue of representation error. We consider a class of generative autoencoders known as Wasserstein autoencoders (WAEs) introduced in (Tolstikhin et al., 2018) which are trained using a combination of mean squared error loss between input and decoder output, and a maximum mean discrepency penalty between encoder distribution and prior latent distribution. We now review specific related work on light field recovery.

7.2 RELATED WORK

7.2.1 *Light Field Reconstruction*

Light field reconstruction has been performed from different measurement models, such as coded aperture (Liang et al., 2008b; Veeraraghavan et al., 2007; Babacan et al., 2012), compressed sensing (Ashok and Neifeld, 2010; Marwah et al., 2013), novel view synthesis and angular super-resolution (Wanner and Goldluecke, 2013; Shi et al., 2014; Schedl et al., 2015; Vagharshakyan et al., 2018; Jin et al., 2020), spatial angular super-resolution aided by high resolution central view (Wang et al., 2016) and also light-field image in-painting and focal stack reconstruction in (Blocker and Fessler, 2019). Since virtually all such measurement models make the problem of recovering light fields eq. (7.4) an *ill-posed* problem, a natural strategy is to consider regularized energy minimization methods, for example (Marwah et al., 2013; Vagharshakyan et al., 2018). Alternately, one could estimate depth maps (Jeon et al., 2015; Sajjadi et al., 2016) or disparity maps which could be subsequently used to synthesize light fields, see (Chaurasia et al., 2013; Wanner and Goldluecke, 2013) for examples. Recently learning-based approaches have also been applied in light field recovery for coded aperture in (Inagaki et al., 2018; Vadathya et al., 2019), compressed sensing in (Gupta et al., 2017), view synthesis and angular super-resolution in (Yeung et al., 2018; Kalantari et al., 2016; Wu et al., 2019a,b; Wang et al., 2018b; Navarro and Sabater, 2021), spatial and angular super-resolution in (Gul and Gunturk, 2018; Meng et al., 2021) as well as view extrapolation for wide baseline light fields in (Srinivasan et al., 2019; Mildenhall et al., 2019). While neural network-based reconstruction schemes (Inagaki et al., 2018; Vadathya et al., 2019; Gupta et al., 2017; Yeung et al., 2018; Meng et al., 2021, 2019; Kalantari et al., 2016; Navarro and Sabater, 2021) outperform traditional approaches to LF reconstruction by a large margin, they are applicable to specific measurement models only, i.e., they are not flexible in adapting to modifications of the measurement process. We note that (Nabati et al., 2018) is a deep network-based approach for compressive LF recovery, which also takes a mask as an input to the deep network, achieving flexibility with respect to different masks for compressive sensing.

Learning light field representations has been addressed previously since the data is high dimensional and contains redundant information. Representations based on sparse coding have been utilized to perform inference tasks such as disparity estimation (Heber and Pock, 2014; Johannsen et al., 2016) and light field reconstruction (Marwah et al., 2013). Alperovich et al. (2018) have shown that an autoencoder trained on stacks of epipolar-plane images (EPI) can learn useful LF representations which can be used for supervised training for disparity estimation and intrinsic decomposition. Recently, there have been efforts to synthesize a light field from a single image in (Srinivasan et al., 2017; Ivan et al., 2019; Chen et al., 2020a). Srinivasan et al. (2017) train an end-to-end network which is based on depth estimation from a single image and subsequent warping to render light field. CNN-based appearance flow estimation is used in (Ivan et al., 2019), to accomplish LF synthesis from a single image. Chen et al. (2020a) synthesize a light field from a single image without estimating any depth map using a deep neural network employing GAN loss. Generating a light field from a single view can have several possible solutions. The approaches (Srinivasan et al., 2017; Ivan et al., 2019; Chen et al., 2020a) output a fixed light field for a given input image. In contrast, our CWAE can generate different LF patches for the same input patch, by sampling in the latent distribution.

7.3 GENERATIVE MODEL FOR GENERIC LIGHT FIELD RECOVERY

7.3.1 *Conditional Generative Model for Light Fields*

Though light field data has high dimensionality, patches of light fields lie in a manifold of much lower dimension owing to their redundant structure (Alperovich et al., 2018). Therefore, training generative models for LF patches instead of full light fields is a promising alternative. Moreover, the representation learned on the small LF patches can generalize to a wide variety of different light fields independent of any specific class of objects. We introduce generative models for 4D light field patches based on a class of generative autoencoders known as Wasserstein autoencoders (Tolstikhin et al., 2018). In addition to the autoencoder MSE loss between input and output, these models have a maximum mean discrepency (MMD) penalty between the encoder distribution, and the prior latent distribution, instead of the Kullback-Leibler (KL) divergence penalty found in the traditional variational autoencoders. The loss function is given as

$$\text{Total loss} = \text{MSE loss} + \lambda \cdot \text{MMD loss} \tag{7.5}$$

We propose a generative model for LF patches, a conditional Wasserstein autoencoder (CWAE), conditioned on the central view. We trained the model for LF patches of spatial resolution 25 × 25. The angular resolution of the LF patch is chosen to be the same as the angular resolution of the light field to be reconstructed (5 × 5 and 7 × 7 in our experiments). Although we restrict the spatial extent of an LF patch to 25 × 25 pixels, due to diverse possibilities of texture content, parallax effects, and occlusion effects, representing any patch with a generative model would still be a difficult task. Therefore, we develop a model which is conditioned on the patch corresponding to the central view. With the central patch being fed into the network as an additional input, the encoder only needs to encode the additional information to represent the parallax and occlusion effects in the light field. The decoder learns to utilize the information from the central view to map the latent variable to the light field. The schematic of the CWAE with its main components is illustrated in Fig. 7.3. Features of the central view are extracted from a convolutional feature extractor at different layers (CVF_1 and CVF_2), which are together referred to here as the central view features (CVF). These are simultaneously fed to both the encoder and the generator. The feature extractor is jointly trained along with the encoder and generator. We employ 3D and 2D convolutions in our architecture as an alternative to computationally expensive 4D convolutions. To realize this, the encoder blocks Enc_1 and Enc_2 in E_1 (Fig. 7.3 (d)) take the input 4D LF patch as a set of 3D LF patches by splitting them along the horizontal and vertical view dimensions, respectively. The outputs of these encoder blocks are together fed into a common encoder Enc_3, along with a set of central view features CVF_1. The output of Enc_3 together with central view features CVF_2 are further encoded by fully connected layers to output latent code z. The generator G_1, takes in the latent code and central view features CVF_2 which first pass through linear fully connected layers, followed by a common partial decoder Dec_1. This decoder's output together with central view features CVF_1, simultaneously pass through the row and column decoders Dec_2 and Dec_3. These features are together input to a final 4D convolutional layer. Further details of CWAE network architecture are provided in Appendix 7.A.

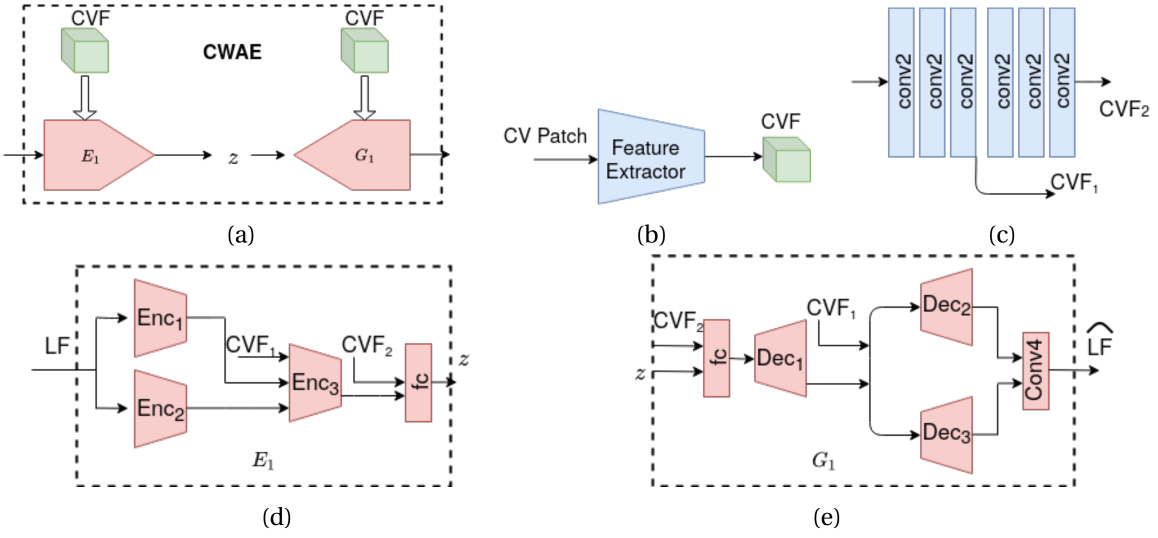


Figure 7.3: (a) Schematic of CWAE. (b) Central view feature (CVF) extraction. (c) Architecture of feature extractor, CVF={CVF_1,CVF_2}. (d) Schematic of encoder E_1 of CWAE. (e) Schematic of generator G_1 of CWAE

7.3.2 *Reconstruction from Generative Model*

To illustrate the performance of the CWAE, Fig. 7.4 depicts sample reconstructions (encoding and decoding) from our CWAE for 4 LF patches. We handle colored light field inputs by reconstructing each color channel separately. In the second row of Fig. 7.4, we observe that our CWAE can reconstruct the input LF patches quite accurately. It captures the disparity across different views, and is able to realistically estimate pixel values that are not present in the central view due to the parallax. To demonstrate the efficacy of the CWAE latent code

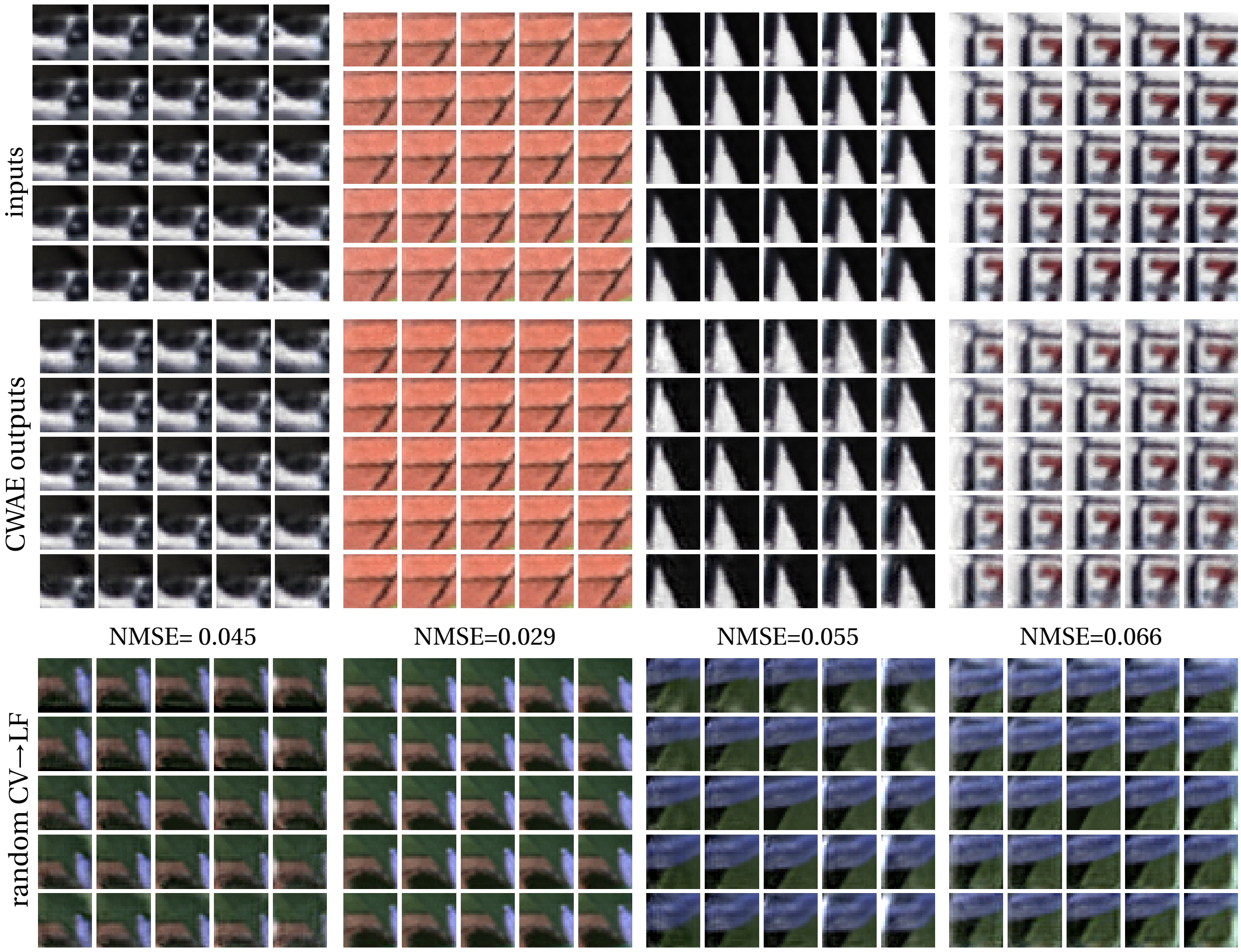


Figure 7.4: Sample reconstruction from CWAE. The first two rows are input LF patches and corresponding reconstructions from CWAE. The third row shows the CWAE mapping of an arbitrary central patch to an LF patch with disparity similar to input LF patch, using the latent code corresponding to the second row. Reported numbers are normalized RMSE (NMSE) values of the reconstructions with respect to the corresponding input patches.

in encapsulating different properties of the input LF patch, we show the generation of a light field from an arbitrarily chosen central patch in the third row of Fig. 7.4. The latent representation of the LF patch shown in the first row is used for generating this output. As we can see, the result is a new LF patch with disparity values similar to the input LF patch in the first row of Fig. 7.4. This indicates that the latent vector indeed encodes an understanding of the geometry of the scene. In the following, we develop LF recovery techniques which exploit the strength of our CWAE.

7.3.3 *Generic Light Field Recovery*

Light field recovery from measurements as seen in Sec. 7.1.1 is an inherently ill-posed problem, and needs strong priors to obtain acceptable solutions. We consider two scenarios: i) the central view is available, and ii) the central view is not available. We now proceed to solve the LF reconstruction problems in both the cases using our CWAE from Sec. 7.3.1.

Central view available

In some LF recovery applications such as view synthesis, or spatial angular super-resolution, one can assume that the central view is known. For such scenarios, we utilize our CWAE model for reconstruction. Given the central view, the generator of CWAE is trained to always map a latent code to a light field patch. Therefore, we optimize over the latent space similar to (Bora et al., 2017; Li et al., 2017) to obtain a latent code that best captures the scene geometry corresponding to the observations. However, unlike (Bora et al., 2017; Li et al., 2017), we use a conditioned generative model, which additionally takes the central view as input. More specifically, we solve

$$\min_{z} \| f - A(G_1(z)) \|_2^2, \tag{7.6}$$

where G_1 is the generator of CWAE, and A is the operator corresponding to the measurement from angular subsampled views or from spatial and angular subsampled views, assuming the central view is present. We minimize eq. (7.6) locally using Adam (Kingma and Ba, 2014), a gradient-based optimization algorithm. After finding a local minimum $\hat{z}$ of eq. (7.6), $G_1(\hat{z})$ is considered to be our final light field estimate.

Central view not available

In LF recovery applications such as recovery from coded aperture, the central view is not available. Even in this case, we can utilize the generator of CWAE for reconstruction. The only difference is that we now optimize both for the latent code z and the central view $\mathbf{c}$. We solve the following optimization problem

$$\min_{z,\mathbf{c}} \| f - A(G_1(z, \mathbf{c})) \|_2^2, \tag{7.7}$$

where A is the forward measurement operator. We solve this problem using Adam optimizer to obtain local minimizers $\hat{z}$ and $\hat{\mathbf{c}}$. We find our final LF estimate as $G_1(\hat{z}, \hat{\mathbf{c}})$.

7.3.4 *Experiments*

To be able to compare with recent network-based approaches on small baseline light fields, we evaluate view synthesis from sparsely sampled views for LFs with angular resolution 7 × 7. We evaluate LF recovery for view synthesis, spatial-angular super-resolution and coded aperture for LFs with angular resolution 5 × 5. The code and trained models are publicly available at `https://github.com/KVGandikota/Generative-Light-Field-Models/`.

Baselines:

We obtain the performance references for the reconstruction tasks using both, model- and network-based approaches for comparisons. For 7 × 7 view synthesis, we compare with the recent neural network-based technique of (Wu et al., 2019b). For comparison with a traditional approach, we report the performance of the depth-based approach of (Jeon et al., 2015) as reported in (Wu et al., 2019b). The dictionary-based approach of Marwah et al. (2013), developed for compressed sensing, is a flexible technique, which can be used with any observation model. We use their open sourced code[1] which is available for LFs of angular resolution 5 × 5. We use this as a reference for model-based approaches on all the 3 recovery tasks for 5 × 5 LFs. For the best performance of (Marwah et al., 2013), we always compute their result obtained by averaging over overlapping patches with stride 1. Additionally, for comparison with a recent neural network baseline, we compare with (Jin et al., 2020) for 5 × 5 view synthesis. We use their publicly available code to retrain their model for this task. For reconstruction from coded aperture, we compare to the neural network based approach of Inagaki et al. (2018).

Datasets:

For training the generative models, we used the following datasets: i) The training set used by Kalantari et al. (2016), ii) the training set used in CNN-based depth estimation for light fields by Heber and Pock (2016), and iii) the training set used in encoder-decoder-based light field intrinsic (Alperovich et al., 2018). These datasets contain a significant number of samples with effects such as occlusions and specular reflections. We create a training set by randomly cropping 250K LF patches of resolution 5 × 5 × 25 × 25 in gray scale from these datasets and use them for training the CWAE with angular resolution 5 × 5. Similarly, a training set of 250K LF patches of resolution 7 × 7 × 25 × 25 was created to train the CWAE with angular resolution 7 × 7. The datasets from (Alperovich et al., 2018) and (Heber and Pock, 2016) have high disparity, therefore we down-scale those light fields spatially by a factor of 1.4 before extracting patches from this data. We investigate the effect of training with these datasets by training a separate CWAE on each of them. The comparison of sample reconstructions using these models with our model trained on all the three datasets is provided in the appendix. Furthermore, we also study the performance of our generative model for LF patches of different spatial extents, which is provided in the appendix.

We evaluate the light field recovery on synthetic and real datasets. Specifically, for LFs of angular resolution 5 × 5, we evaluate the recovery from all the tasks on the light fields "Dino", "Kitchen", "Medieval 2" and "Tower" from the synthetic New HCI dataset (HCI, 2018). Furthermore, we evaluate coded aperture reconstruction on the real light field from (Inagaki et al., 2018). We evaluate view synthesis for LFs of angular resolution 7 × 7 on the test set of Kalantari et al. (2016) which contains 30 real light fields captured by a Lytro Illum. Further, we also evaluate 7 × 7 view synthesis on the LFs 'Reflective 9', 'Reflective 13', 'Reflective 22', 'Reflective 27', 'Reflective 29', 'Occlusions 16', and 'Bikes12' from Stanford Lytro light field archive (Sunder Raj et al., 2016), which contain significant reflections, transparencies, specularities and occlusions.

Generative model training:

We used Pytorch 1.1.0 for all our experiments. For training the CWAEs, we used mini-batches of size 128 and trained the models for 5 × 5 and 7 × 7 views with spatial extent of 25 × 25 pixels for 150 epochs. We used Adam optimizer (Kingma and Ba, 2014), with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. We set the initial learning rate to 10^{-3}, which is decreased by a factor of 2 after 30 epochs, further by a factor of 5 after first 50 epochs and finally by a factor of 10 after 100 epochs. For both the models, we choose the factor λ in eq. (7.5) to be 100.

[1] http://web.media.mit.edu/~gordonw/CompressiveLightFieldPhotography/

Dataset	$3 \times 3 \rightarrow 7 \times 7$				5 views$\rightarrow 7 \times 7$	
	EPICNN	Ours	Ours$^{\mathrm{OL}}$	depth-based†	Ours	Ours$^{\mathrm{OL}}$
30 scenes	41.16	38.53	39.77	34.42	38.29	39.57
7 scenes	41.24	39.62	40.48	-	39.07	40.00

Table 7.1: Average PSNR of novel views in dB for 7 × 7 view synthesis on test test of 30 scenes from (Kalantari et al., 2016), and 7 scenes from (Sunder Raj et al., 2016) containing reflections, refractions and occlusions. Comparisons with EPICNN (Wu et al., 2019b), and depth based method (Jeon et al., 2015) are shown. † indicates PSNR values of (Jeon et al., 2015) are as reported in (Wu et al., 2019b).

LF recovery:

Since our generative models are trained on gray scale patches, we divide the input into patches of suitable dimensions and use our generative models on all color channels separately. We initialize the latent code z with a random sample drawn from the same posterior distribution that was used for the latent space during the training of the generative model (i.e. isotropic Gaussian with variance of 2). We observed that different random initializations of z lead to similar quality of reconstruction. For recovery from coded aperture, the central view is not available. In this case, we initialize the central view with the coded image itself scaled between 0 and 1. We solve the LF reconstruction tasks using Adam optimizer as discussed in Sec. 7.3.3, until convergence.

7.3.5 *Results*

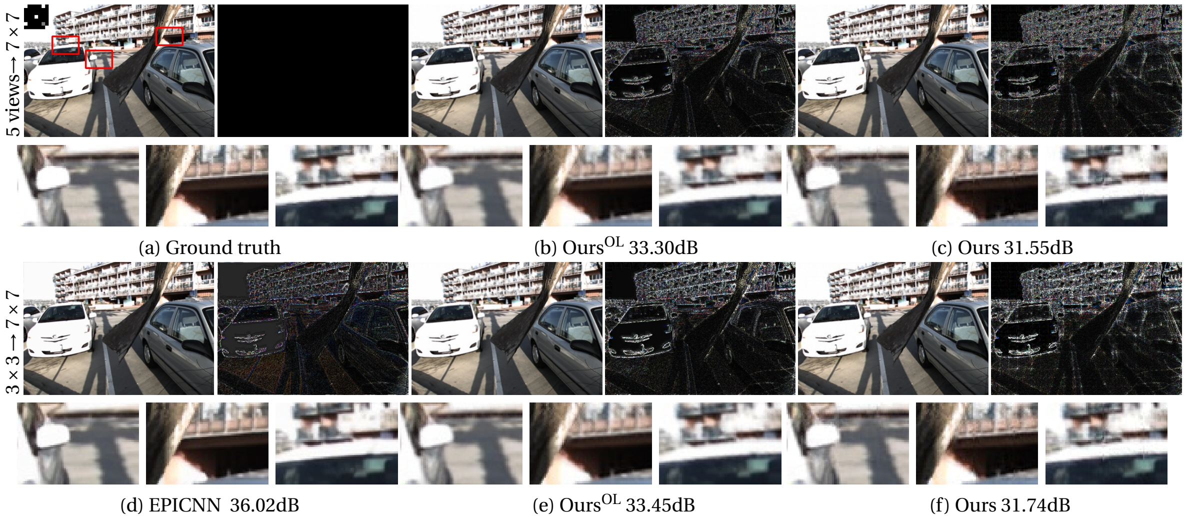
We now evaluate the efficacy of our approach on different LF recovery tasks. We perform quantitative evaluation in terms of PSNR and also qualitative evaluation by comparing light field views of our approach with ground truth and baseline methods and show the corresponding error maps. Additional visual comparisons of the reconstructed LFs are provided in the appendix.

View synthesis 7×7*:*

We compare our approach with the recent CNN-based technique of Wu et al. (2019b) for LF reconstruction from sparsely sampled input views using epi-polar images (EPI). We consider upsampling the angular resolution from 3 × 3 to 7 × 7. Since central view is available for this task, our approach uses CWAE for reconstruction. We use the publicly available trained model of Wu et al. (2019b)[2] for evaluating their approach. We also report the performance of a traditional depth estimation-based approach from (Wu et al., 2019b) for this task, where the depth is estimated using the approach of Jeon et al. (2015), followed by a novel view synthesis by warping the input views following (Chaurasia et al., 2013). Apart from the specific case of 3 × 3 input views, our method can still be applicable if any arbitrary set of views are given as input along with the central view. To demonstrate this flexibility, we also show 7 × 7 LF reconstruction from 5 randomly chosen input views including the central view. The mask used for selecting the 5 input views is provided in the inset of Fig. 7.5 a). Since view extrapolations cannot be handled by EPICNN (Wu et al., 2019b), we show visual comparison only with the ground truth for this task.

Results of our quantitative evaluation on 30 real LFs of Kalantari et al. (2016) test set and 7 scenes selected from Stanford Lytro dataset (Sunder Raj et al., 2016) are provided

[2] `https://github.com/GaochangWu/lfepicnn`



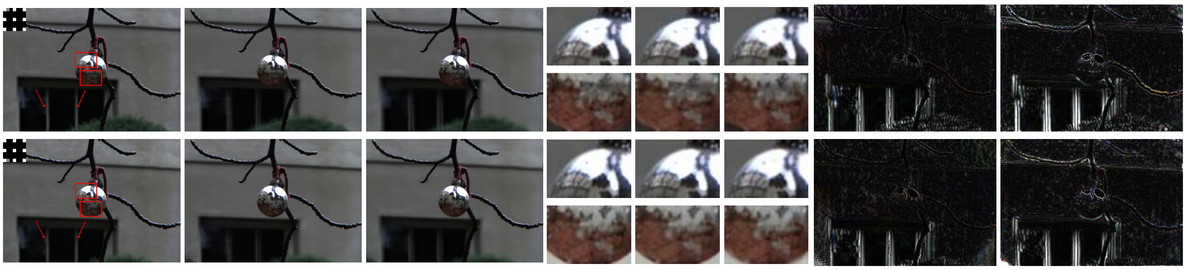
(a) Ground truth (b) Ours^{OL} 33.30dB (c) Ours 31.55dB

(d) EPICNN 36.02dB (e) Ours^{OL} 33.45dB (f) Ours 31.74dB

Figure 7.5: Result of 7 × 7 view synthesis for the LF 'Cars'. Shown is the novel view at angular location (6,6), depicted as gray location in the inset. The mask for selecting 5 input views is shown in the inset of ground truth view. Figures in the first row a)−c) depict ground truth view, and the results of our approach using 5 input views with and without overlapping patches in that order. Figures d)−f) in the second row provide visual comparison of novel views generated using approach of EPICNN Wu et al. (2019b), and our approach using 3 × 3 angular views. Error maps and zoomed in patches are depicted along with corresponding novel views, with error magnified by a factor of 10. Results best viewed when zoomed in.

in Tab. 7.1. 'Ours^{OL}' indicates our reconstruction using overlapping patches with stride 5. Following Wu et al. (2019b), we show the result of average PSNR of the luminance component of novel synthesized views. For brevity, we report only average PSNRs of the LFs in each test set. Quantitative comparisons for individual LFs are provided in the appendix. For the task of view upsampling from 3 × 3 to 7 × 7, we compute the average PSNRs of the 40 novel views. For this task, we find that our performance is approaching the CNN-based method of Wu et al. (2019b), with a PSNR reduction of only 1.4 dB when we use overlapping patches, and 2.6 dB when non-overlapping patches are used on (Kalantari et al., 2016) test set. Our approach also outperforms the depth-based approach using the method of Jeon et al. (2015) by a large margin. Further, our performance is close to the method of Wu et al. (2019b) on the scenes selected from (Sunder Raj et al., 2016) as well, with PSNR reduction of only 0.8 dB and 1.6 dB respectively, when overlapping and non-overlapping patches are used. Even when the number of known views is reduced to 5, our average PSNR of 44 novel views is 39.57 dB on the 30 scenes (Kalantari et al., 2016) with a reduction of only 0.2 dB, and average PSNR of 40.00 dB with a reduction of 0.48 dB on the 7 scenes from (Sunder Raj et al., 2016), demonstrating the strength of our approach.

A qualitative comparison of the synthesized views for the task of 7 × 7 view synthesis is provided in Fig. 7.5 for the LF 'Cars' from the 30 scenes test set. The newly synthesized view at angular location (6, 6) (depicted by gray location in the inset) are shown. The first row of Fig. 7.5 (a)−(c) gives a visual comparison of the results of our approach with the ground truth when 5 input views are used. Visually, it can be seen that our approach provides a reasonable reconstruction quality even when using a limited number of input views. The second row of Fig. 7.5 (d)−(f) compares our method with the approach of EPICNN (Wu et al., 2019b), for the task of 3 × 3 → 7 × 7 angular super resolution. In terms of reconstruction quality, our approach performs slightly worse than (Wu et al., 2019b). However, this is to be expected as Wu et al. (2019b) uses network specifically trained for this task. In contrast, we obtain a comparable reconstruction quality with flexible input views. It can be noticed from the error



a) Ground truth b) EPICNN 35.76 dB c) Ours$^{\text{OL}}$35.06 dB a) b) c) b) EPICNN c) Ours$^{\text{OL}}$

Figure 7.6: Visual comparison of our synthesized views (depicted by gray locations in the inset) for the task of $3 \times 3 \rightarrow 7 \times 7$ view synthesis for the LF 'Reflective13'. Columns $1-3$ depict a) the ground truth views, the result using b) EPICNN Wu et al. (2019b) and c) our approach using overlapping patches respectively. Columns $4-6$, the patches of columns $1-3$. Columns $7-8$ depict the error maps corresponding to columns $2-3$ with error magnified by a factor of 10. The brightness of the zoomed in patches is increased for better illustration. Average PSNR in dB of 40 novel views is shown.

maps and zoomed in patches that our approach preserves the details fairly well. Further, we can observe that there are errors at the patch boundaries when non-overlapping patches are used. These errors are reduced due to averaging effect when overlapping patches are used.

In Fig. 7.6, we illustrate our reconstruction of the LF 'Reflective13' from the Stanford Lytro dataset and compare it with the approach of Wu et al. (2019b) for the task of $3 \times 3 \rightarrow 7 \times 7$ view synthesis. The novel synthesized views at angular locations $(1,2)$ and $(7,2)$ (depicted by gray location in the inset of ground truth) are shown. This is a challenging scene which contains a highly reflective ball in the foreground, and high disparities (4 pixels between adjacent views) in the background. We can observe from the synthesized views and corresponding error maps that our approach provides reconstructions which are slightly worse than (Wu et al., 2019b). On closer inspection of the zoomed in patches, we can observe that our method can reconstruct well the reflections which are slowly varying across views (patches on the top in each row). We observe reasonable reconstruction even when there is a high variability in the reflections across views (patches on the bottom). However, our approach cannot handle high disparities in the background causing ghosting artifacts, as seen in the corresponding regions in the reconstructions, which are indicated by red arrows in the ground truth views. We observe that the approach (Wu et al., 2019b) also cannot handle such large disparities, which are also evident in the error maps.

View synthesis with corrupted inputs:

CLEAN CENTRAL VIEW AVAILABLE To further demonstrate our flexibility vis-a-vis end to end trained networks, we consider the task of $3 \times 3 \rightarrow 7 \times 7$ angular super resolution and compare our reconstruction with EPICNN (Wu et al., 2019b), when inputs are corrupted. We assume that the central view is clean and the remaining 8 views are corrupted by different distortions. The qualitative and quantitative comparison of our reconstructions with the approach of EPICNN (Wu et al., 2019b), with corrupted input views is provided in Fig. 7.7 and in Tab. 7.2 for the LF 'Cars'. The reconstructed view at angular location $(6,6)$ is depicted.

	Clean	$\sigma = 0.05$	$\sigma = 0.1$	S&P	50% pixels
EPICNN	36.02	33.34	29.95	25.02	13.60
Ours	31.74	31.75	31.67	31.66	31.68
Ours$^{\text{OL}}$	33.45	33.47	33.41	33.35	33.39

Table 7.2: $3 \times 3 \rightarrow 7 \times 7$ view synthesis result on the LF 'Cars', when input views other than central view are corrupted. PSNR values in dB in comparison with EPICNN Wu et al. (2019b) are shown

With additive Gaussian noise of standard deviation $\sigma = 0.05$ in 8 input views, the PSNR of the reconstructed views using (Wu et al., 2019b) drops from 36.02 dB to 33.34 dB. When we increase the noise level to $\sigma = 0.1$ this value further drops to 29.95 dB. This degradation in the quality of reconstruction is also evident from the error maps in Fig. 7.7. In contrast, our reconstruction quality is robust to addition of noise.

We also consider corruption of input views with salt-and-pepper noise with a probability of 0.05. Even in this case, the performance of Wu et al. (2019b) is severely affected, with PSNR reduction of 11 dB compared to the clean case, where as our performance only shows a marginal decrease of 0.1 dB. We note that we employ an L_1 loss, as it is more suited to handle salt and pepper noise when compared to the traditional L_2 loss in eq. (7.6). This demonstrates the flexibility of our energy minimization-based approach in adapting to different noise statistics. When we use an L_2 loss instead, our PSNR dropped by about 2 dB compared to the clean case. Finally, when 50% pixels are randomly dropped from the 8 known views, the neural network-based approach of Wu et al. (2019b), completely fails in reconstruction. In contrast, we can incorporate an additional mask corresponding to the missing pixel locations in our optimization, and consequently our reconstructions remain robust to this distortion.

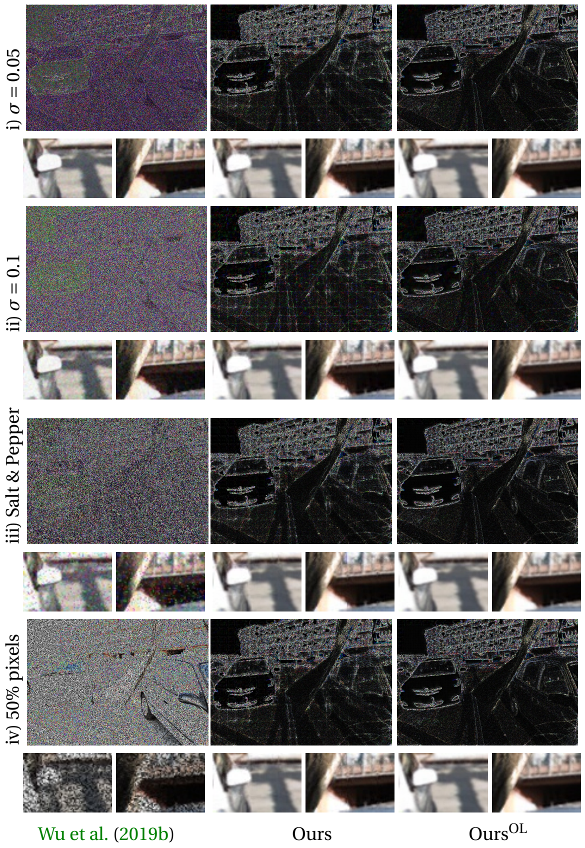


Figure 7.7: Novel view at angular location (6,6) for the task $3 \times 3 \rightarrow 7 \times 7$ view synthesis. Columns $1-3$ depict the result using EPICNN Wu et al. (2019b) and our approach using non-overlapping patches and overlapping patches respectively. Shown are the zoomed in patches of the reconstructed views and error maps with error magnified by a factor of 10. Among the 3×3 input views, central view is clean. For the the remaining 8 views, we consider the following corruptions (rows i–iv) i) additive Gaussian noise $\sigma = 0.05$. ii) additive Gaussian noise $\sigma = 0.1$ iii) salt and pepper noise with a probability of occurrence of 0.05. iv) 50% pixels randomly dropped from views. Results best viewed by zooming in.

VIEW SYNTHESIS WITH CORRUPT CENTRAL VIEW We consider the task of $3 \times 3 \rightarrow 7 \times 7$ angular super resolution and compare our reconstruction with (Wu et al., 2019b), when input views including the central view are corrupted are corrupted by different distortions. Since our approach is crucially dependent on the central view, we investigate the effect of considering an additional total variation (TV) penalty on the central view to deal with noise. We initialize the central view to be the observed corrupted central view and optimize jointly for the central view and the latent code. The qualitative and quantitative comparison of our reconstructions with the approach of Wu et al. (2019b), with corrupted input views is provided in Fig. 7.8 and in Tab. 7.3 for the LF 'Cars'. We show the reconstructed view at angular location (6,6). We consider 4 different scenarios, where the input views are corrupted by additive Gaussian noise of variance 0.05 and 0.1, salt and pepper noise with a probability of occurrence of 0.05, and 50% of pixels randomly dropped from the input views. We can observe reconstructions

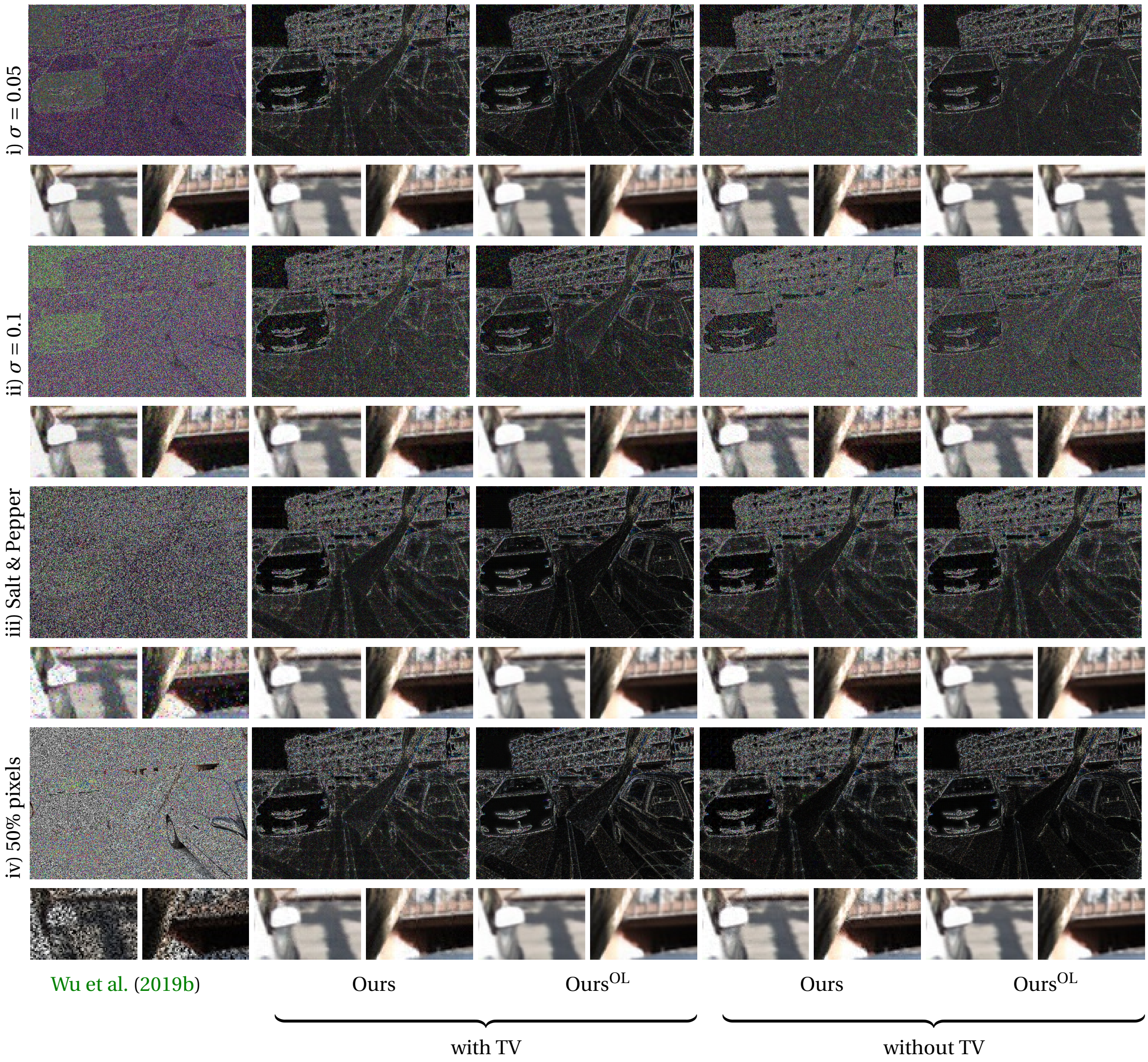


Figure 7.8: Visual comparison of our approach with Wu et al. (2019b) (first column) on the novel view at angular location (6, 6) for the task 3 × 3 → 7 × 7. Columns 2 – 3 depict our result with TV regularization, and columns 4 – 5 depict our reconstruction without TV regularization. Shown are the zoomed in patches of the reconstructed views and error maps with error magnified by a factor of 10. We consider the following corruptions to all the input views i) additive Gaussian noise $\sigma = 0.05$. ii) additive Gaussian noise $\sigma = 0.1$ iii) salt and pepper noise with a probability of occurrence of 0.05. iv) 50% pixels randomly dropped from views. Results best viewed by zooming in.

using (Wu et al., 2019b) are highly sensitive to corruptions in the input views, with the sharp drop of 12.3 dB in average PSNR, when compared to reconstruction using clean inputs. In contrast, our approach results in PSNR drop of only 0.63 dB and 1.2 dB when compared to our reconstruction with clean input views, when overlapping and non-overlapping patches are used. Additional TV regularization further improves our reconstruction under noise. In this case, the drop in average PSNR when compared to reconstructions with clean inputs is only 0.43 dB and 0.59 dB when overlapping and non-overlapping patches are used.

With additive Gaussian noise of standard deviation $\sigma = 0.05$ and $\sigma = 0.1$, reconstructions using (Wu et al., 2019b) show PSNR drops of 3.37 dB and 7.42 dB with respect to reconstructions when inputs are clean. This drop in reconstruction quality is also visible in the error maps in Fig. 7.8. Since the central view is also corrupted, our method also shows a performance drop, albeit much lower than (Wu et al., 2019b). For the same noise levels, we observe a PSNR drop of 0.35 dB and 2.52 dB with our approach using overlapping patches, when

Corruption	EPICNN	with TV		without TV	
		Ours	Ours$^{\mathrm{OL}}$	Ours	Ours$^{\mathrm{OL}}$
None	36.02	31.80	33.48	31.74	33.45
Gaussian noise $\sigma = 0.05$	32.65	31.62	33.32	30.98	33.10
Gaussian noise $\sigma = 0.1$	28.60	30.81	32.52	28.71	30.93
Salt&Pepper noise	23.21	31.00	33.27	30.83	33.23
50% Pixel drop	10.43	31.40	33.08	31.51	34.01

Table 7.3: $3 \times 3 \rightarrow 7 \times 7$ view synthesis result on the LF 'Cars', when input views including the central view are corrupted. Shown are PSNR values in dB for our approach and EPICNN Wu et al. (2019b).

compared to our reconstructions with clean input views. With additional TV regularization on the central view, this PSNR drop further reduces to around 0.16 dB and 0.96 dB with additive Gaussian of standard deviation $\sigma = 0.05$ and $\sigma = 0.1$ respectively. When the input views are corrupted by salt-and-pepper noise with a probability of 0.05, the reconstructions using (Wu et al., 2019b) are strongly affected, with PSNR drop of 12.8 dB compared to the clean case. In contrast, our PSNR drops by only 0.2 dB, since we use an L_1 reconstruction loss to tackle salt and pepper noise. When we use an L_2 loss, our PSNR drops by 4.9 dB and 6 dB in comparison to using L_1 loss, for overlapping and non-overlapping patches respectively. This is because L_2 loss is poorly suited to handle salt and pepper noise. When 50% pixels are randomly dropped from the input views, reasonable recovery is not provided by (Wu et al., 2019b). In our approach, we also incorporate the mask corresponding to the missing pixel locations in our optimization, and therefore our approach can effectively handle this distortion. This flexibility to handle different distortions is possible in the framework of energy minimization.

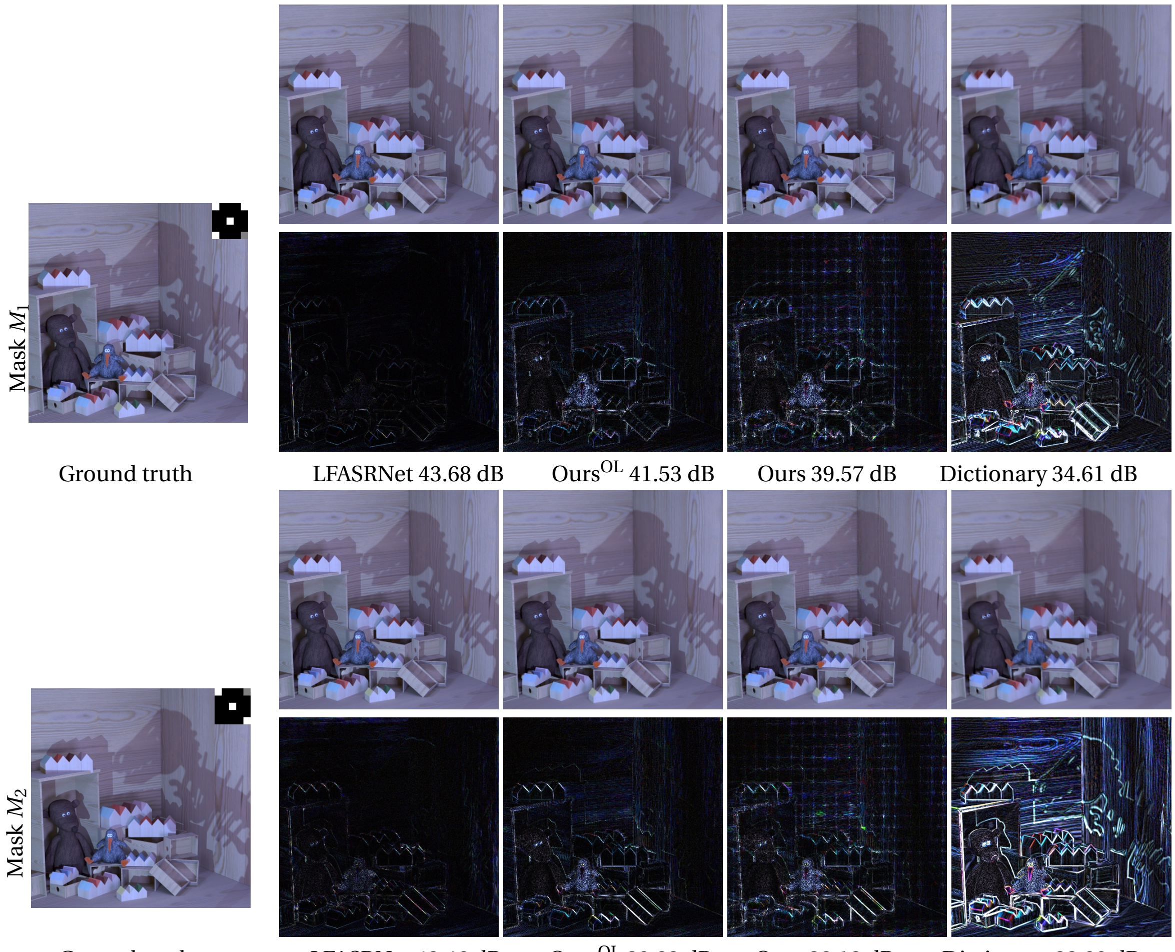
View synthesis 5×5*:*

We compare our approach for view synthesis with dictionary based approach of Marwah et al. (2013) and geometric warping based light field angular super-resolution network (LFASRNet) (Jin et al., 2020) for two different input views using masks M_1 and M_2. For evaluating the performance of (Jin et al., 2020) we use separate networks trained end-to-end for view synthesis with each of the masks.

The results of our quantitative evaluation on synthetic HCI data are summarized in Tab. 7.4, where the PSNR of the reconstructed light fields is presented. Our approach without considering overlapping patches is superior by 2.63 dB and 3.13 dB to the dictionary-based approach of Marwah et al. (2013) with overlapping patches with stride 1, for masks M_1 and M_2, respectively in terms of average PSNR. Our performance further improves when we consider overlapping patches with stride 5, where our approach is better by 4 dB and 4.4 dB, respectively for M_1 and M_2. Further, the end-to-end trained LFASRNet (Jin et al., 2020) performs the best, with an improvement in average PSNR of 1.69 dB compared to our approach with overlapping patches for both M_1 and M_2.

	Method	Dino	Kitchen	Medieval2	Tower
Mask M_1	Ours	39.57	33.59	34.86	31.24
	Ours$^{\mathrm{OL}}$	41.53	34.95	35.94	32.30
	Dictionary	34.61	30.30	32.19	28.45
	LFASRNet	43.68	37.01	36.75	34.00
Mask M_2	Ours	38.18	33.06	34.55	30.28
	Ours$^{\mathrm{OL}}$	39.83	34.41	35.66	31.31
	Dictionary	32.99	29.83	31.51	27.67
	LFASRNet	42.46	36.29	36.25	32.97

Table 7.4: Quantitative comparison of 5×5 view synthesis with dictionary based approach Marwah et al. (2013) and LFASRNet Jin et al. (2020). PSNR values in dB are shown.



Ground truth LFASRNet 43.68 dB Ours$^{\mathrm{OL}}$ 41.53 dB Ours 39.57 dB Dictionary 34.61 dB

Ground truth LFASRNet 42.46 dB Ours$^{\mathrm{OL}}$ 39.83 dB Ours 38.18 dB Dictionary 32.99 dB

Figure 7.9: Result of view synthesis of the LF 'Dino'. Masks M_1 and M_2 are provided as inset of the ground truth views. The columns 1 – 5 show the views depicted by gray location in the inset corresponding to i) the ground truth, and synthesized novel views using ii) LFASRNet Jin et al. (2020) iii) – iv) our approach with overlapping patches and without overlapping patches and v) the dictionary based method of Marwah et al. (2013), respectively. Columns 6 – 9 illustrate the error maps corresponding to the reconstructed views in columns 2 – 5, with errors magnified by a factor of 10. Shown are the PSNR values in dB of the reconstructed LFs. (Results best viewed zoomed in).

A qualitative comparison of the synthesized views is provided for the LF 'Dino' for mask M_1 and M_2 in Fig. 7.9. The locations of known views are depicted in white in the inset of Fig. 7.9, and gray represents the location of the reconstructed view. Extrapolating novel views away from known views is difficult. Even for this challenging case, we observe the quality of our reconstruction with both, overlapping and non-overlapping patches, is better and sharper compared to the reconstruction from the dictionary-based approach of Marwah et al. (2013). The neural network approach of Jin et al. (2020) provides even better reconstruction, which is expected with end-to-end networks specifically trained for each of the masks. This is also evident from the error maps shown in Fig. 7.9. We can observe that averaging effect of overlapping patches mitigates the errors at the patch boundaries in comparison to our approach without overlapping patches.

Spatial and angular super-resolution 5 × 5*:*

Fig. 7.10 provides a visual comparison of our LF reconstruction with the approach of Marwah et al. (2013) for the task of spatial-angular super-resolution on the LF 'Kitchen'. The masks used for the measurements are provided in the inset of ground truth view of the LF 'Kitchen' in

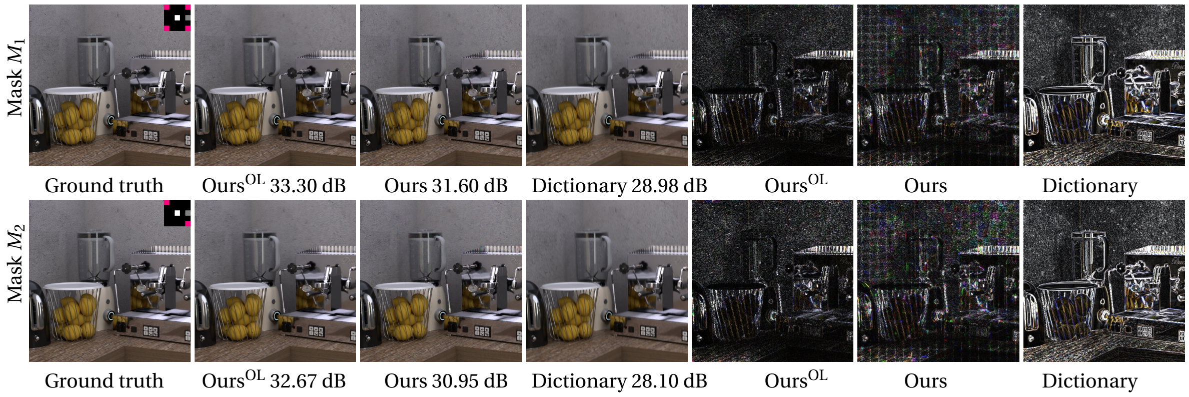


Figure 7.10: Result of spatial angular super-resolution of the LF 'Kitchen'. Masks M_1 and M_2 are provided as inset of the ground truth views. Central view in full resolution is depicted in white. Measurements at the locations in red are spatially down-sampled by a factor of 3. The columns 1 – 4 from left to right show the views depicted by gray location in the inset corresponding to the i) ground truth, and synthesized views using ii)–iii) our approach with overlapping patches and without overlapping patches, and iv) dictionary based approach of Marwah et al. (2013). Columns 5 – 7 illustrate the error maps corresponding to the reconstructed views in columns 2 – 4, with error magnified by a factor of 10.(Results best viewed zoomed in)

Fig. 7.10. The central view is available in full resolution and is depicted in white. Views in red are spatially down-sampled by a factor of 3. It can be observed that our reconstruction of the novel view (depicted in gray in the inset) with both overlapping patches and non-overlapping patches is of superior quality compared to the reconstruction from the approach of Marwah et al. (2013). This is further substantiated by the error maps shown in the Fig. 7.10, which depict a much lower error in our reconstruction. Tab. 7.5 provides a quantitative comparison of our method with the dictionary-based approach of Marwah et al. (2013). Again, on average our approach outperforms the approach of Marwah et al. (2013) by more than 2 dB without, and by more than 4 dB with overlapping patches.

	Method	Dino	Kitchen	Medieval2	Tower
Mask1	Ours	37.18	31.60	33.27	29.95
	Ours^{OL}	39.71	33.30	34.87	31.15
	Dictionary	33.07	28.98	31.26	27.93
Mask2	Ours	35.84	30.95	32.78	28.99
	Ours^{OL}	38.11	32.67	34.50	30.23
	Dictionary	31.70	28.10	30.26	26.93

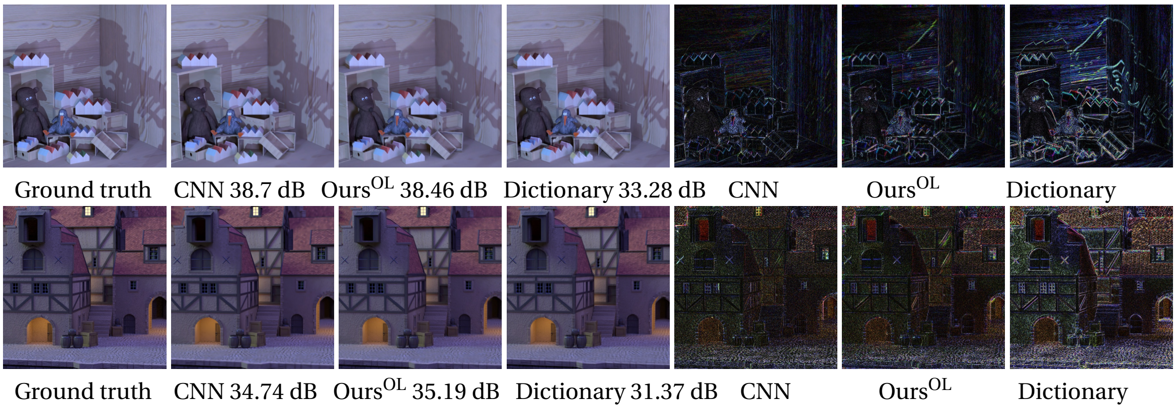
Table 7.5: Spatial-angular super-resolution: PSNR values in dB shown for our reconstructions and those recovered by dictionary based method of Marwah et al. (2013).

Coded aperture 5 × 5*:*

We evaluate the LF recovery from 2 coded aperture observations for our approach, CNN based coded aperture recovery method of Inagaki et al. (2018) and the dictionary based method of Marwah et al. (2013), using two different coded mask sets 'Normal' and 'Rotated' (available from (Inagaki et al., 2018)), and denote them by M_1 and M_2, respectively. The quantitative evaluation on synthetic data is summarized in Tab. 7.6. To evaluate the approach of Inagaki et al. (2018), we use the publicly available trained reconstruction network corresponding to M_1. For

	Method	Dino	Kitchen	Medieval2	Tower
Mask1	Ours	34.97	31.07	32.90	29.02
	Ours^{OL}	38.46	33.29	35.19	30.43
	CNN	38.7	33.78	34.74	31.63
	Dictionary	33.28	29.00	31.37	27.81
Mask2	Ours	34.34	31.03	32.49	28.47
	Ours^{OL}	38.00	33.14	34.84	29.86
	CNN	37.50†	33.00†	34.00†	31.00†
	Dictionary	32.86	29.40	31.42	27.33

Table 7.6: Comparing recovery from coded aperture with CNN based approach Inagaki et al. (2018) and dictionary based approach Marwah et al. (2013). PSNR values in dB are shown. † indicates approximate PSNR values for the mask M_2 are taken from Inagaki et al. (2018).



Ground truth CNN 38.7 dB Ours^{OL} 38.46 dB Dictionary 33.28 dB CNN Ours^{OL} Dictionary

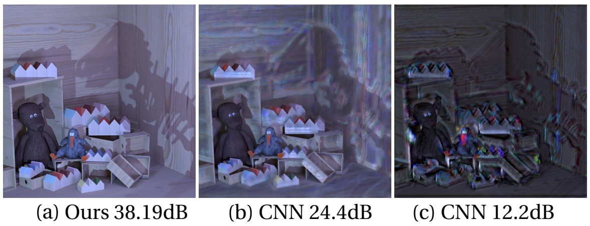
Ground truth CNN 34.74 dB Ours^{OL} 35.19 dB Dictionary 31.37 dB CNN Ours^{OL} Dictionary

Figure 7.11: Coded aperture reconstruction using the coded mask M_1 of Inagaki et al. (2018). The column 1 depicts the the bottom right ground truth LF view. Columns 2 – 4 depict the reconstructed views using the CNN based approach for coded aperture recovery Inagaki et al. (2018), our approach and the dictionary based approach of Marwah et al. (2013) respectively. The error maps corresponding to the views in columns 2 – 4 are illustrated in the columns 5 – 7, with errors magnified by a factor of 10. PSNR values of recovered LFs are shown. (Results best viewed zoomed in).

M_2, we reproduce the values reported in Inagaki et al. (2018), since a trained network is not publicly available. Even without overlapping patches, our method gives superior PSNR values when compared to the model-based approach of Marwah et al. (2013), with improvement of 1.6 dB for both M_1 and M_2. However, our method is worse by 2.7 dB and 2.3 dB for M_1 and M_2 when compared to Inagaki et al. (2018). When we use overlapping patches with stride 5, the average PSNR on the test set for our method is comparable to the end-to-end trained model of Inagaki et al. (2018) and is better by 3.97 dB and 3.71 dB for M_1 and M_2 when compared to Marwah et al. (2013). For qualitative evaluation, we show sample LF reconstructions using coded masks M_1 on the LFs 'Dino' and 'Medieval' in Fig. 7.11. We can observe that our approach provides a reasonably good recovery, with performance comparable to an end-to-end trained network. Our recovery is also more accurate when compared to Marwah et al. (2013).

To demonstrate the vulnerability of the end-to-end trained reconstruction pipeline, we altered the coded aperture mask from the set of M_1 and then perform LF recovery using the method of Inagaki et al. (2018). Minor changes were applied to only one of the two masks in the set M_1. First, we swap the values of the mask at locations with coordinates $(0,0)$ and $(0,2)$. With this tiny change, the performance of (Inagaki et al., 2018) dropped from 38.7 db to 24.3 db on the 'Dino' LF. When we swap the values at three sets of locations, the method of Inagaki et al. (2018) completely failed to reconstruct a meaningful light field (yielding a PSNR of 12.2 dB).

In contrast, the effect of these changes on our approach is marginal, since our optimization scheme explicitly takes the mask as an input. With the first swap in the mask, our PSNR changed to 38.52 dB, compared to 38.46 dB of the original mask, when we use overlapping patches. With three swaps, the PSNR value for our reconstruction is 38.19 dB, demonstrating our flexibility. Views from the reconstructed LFs are shown in Fig. 7.12.



(a) Ours 38.19dB (b) CNN 24.4dB (c) CNN 12.2dB

Figure 7.12: Effect of minor alterations to the coded mask on reconstruction. Shown are reconstructions of the top left view: (a) Our reconstruction with 3 swaps in the mask. (b) Reconstruction using Inagaki et al. (2018) with 1 swap. (c] Reconstruction using Inagaki et al. (2018) with 3 swaps.

We apply our reconstruction method on the real observations obtained in the work of Inagaki et al. (2018). In Fig. 7.13, we show a specific view obtained from our reconstruction along with the corresponding result obtained by Inagaki et al. (2018). Close-ups near the occlusion boundaries for two different views (with appropriate vertical alignment) in Fig. 7.13 (c) and (d) show a comparable quality of our approach (left columns) to the results obtains by Inagaki et al. (2018) (right columns).

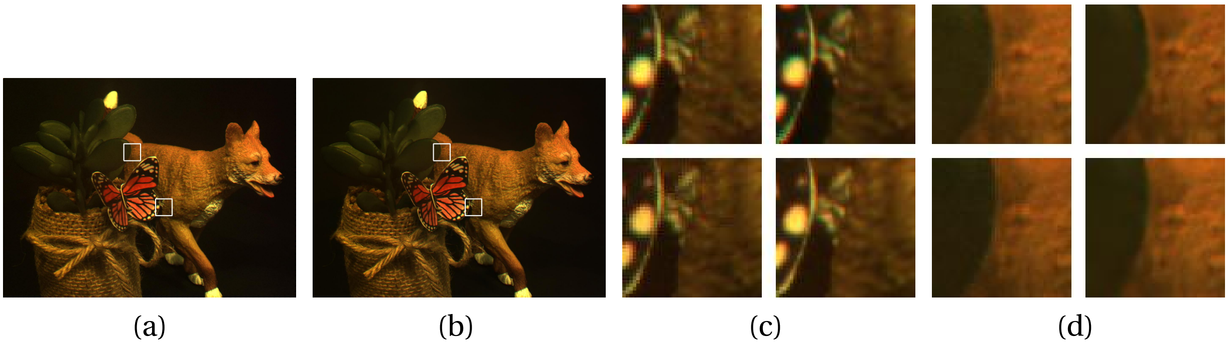


Figure 7.13: Real result using the observation of Inagaki et al. (2018). (a) Central view from our reconstructed light field. (b) Corresponding view from the result of Inagaki et al. (2018). (c) and (d) left half shows patches from two different views of our reconstruction and right half similarly shows patches from the result of Inagaki et al. (2018).

We also attempted comparisons with other model-based approaches (Blocker and Fessler, 2019; Vagharshakyan et al., 2018). We note that these works have not considered view synthesis with arbitrary masks or coded aperture reconstruction. As Vagharshakyan et al. (2018) uses an iterative approach that regularizes the epipolar plane images, it works well with a regular pattern of input views. We found it not to be directly applicable for view extrapolations, while our model remains flexible with respect to the pattern of input views. Moreover, we found that (Blocker and Fessler, 2019) crucially depends on a good initial estimate for view extrapolation. Finally, we found that the performance of (Blocker and Fessler, 2019) on coded aperture reconstruction was worse than that of (Marwah et al., 2013). Therefore, we have not included these comparisons in our results.

Due to patch based processing, and optimization steps required to reconstruct light fields, our reconstruction times are longer. For 7 × 7 view synthesis, our approach takes nearly 12 minutes on a Nvidia GeForce RTX 2080 Ti machine to reconstruct a full Lytro image (of size $376 \times 541 \times 3 \times 7 \times 7$), which requires 150 update steps per patch. For 5 × 5 view synthesis and spatial-angular super-resolution, our approach requires 12 minutes to reconstruct LF of size $512 \times 512 \times 3 \times 5 \times 5$, when 250 update steps per patch are used. Another limitation of our approach is that our reconstructions are not satisfactory when the disparity between adjacent views is greater than two pixels. Since the spatial extent of our generative models is only 25 × 25, it is difficult for our model to capture large disparities in a low-dimensional latent representation, as the views tend to be significantly different. To overcome this limitation, one needs to train a generative model with higher capacity by using LF patches of a larger spatial extent. Since our work is the first attempt to develop generative light field models, we consider this to be beyond the scope of this work.

7.4 CONCLUSION

We developed the first autoencoder-based generative model conditioned on the central view for 4D light field patches for generic reconstruction. We developed algorithms for generic light field reconstruction by exploiting the strengths of our generative model and evaluated our approach on three different LF reconstruction tasks. Experimental results indicate that our approach leads to high quality reconstructions with a performance superior to other optimization-based approaches, while being only slightly worse, but significantly more flexible and robust than end-to-end trained networks. We believe that our experimental results are very promising and can serve as a starting point for further research on generative light field models.

APPENDIX

7.A NETWORK ARCHITECTURE FOR LF GENERATIVE MODEL

As described in the Section. 7.3.1, our conditional generative model consists of three main components- i) central view feature extractor, ii) encoder and iii) generator. Consider a light field (LF) patch of angular dimension 5×5 and spatial extent of size 25×25. The central view feature extractor takes the middle 25×25 2-D patch as input to extract the features that are denoted as CVF_1 and CVF_2 in the paper. For the encoder Enc_1, the input 4D LF patch is rearranged by considering it to be 5 channels of 3D tensors of size 5×25×25. Each channel is composed of a set of horizontal views and corresponds to only a specific location along the vertical view direction. Similarly, for Enc_2, the 4D LF patch is rearranged wherein each channel corresponds to a specific horizontal view location. Consequently, the angular information along the horizontal and vertical view directions is first encoded separately by the encoder blocks Enc_1 and Enc_2. The outputs of these are combined in a common encoder Enc_3, along with a set of central view features. Output of Enc_3 together with another set of central view features are further encoded by fully connected layers to output latent code. The generator mirrors the encoder architecture, with a common decoder Dec_1 whose output together with the central view features are further decoded by the decoders Dec_2 and Dec_3 to reconstruct angular information in horizontal and vertical view directions, which is combined in a final 4D convolutional layer.

In the following, we provide the architectural details of the components of CWAE. We use the following notation to describe convolutional mappings. $C^{F}_{a\to b}\downarrow_{S}$ represents convolution filter mapping from channel dimension of a to b with filter size of F and stride S. $C^{F}_{a\to b}\uparrow_{S}$ represents fractional strided convolution (transpose convolution) filter mapping from channel dimension of a to b with filter size of F and stride S.

Feature extractor:

$$C^{(3,3)}_{1\to 6}\downarrow_{(1,1)}\to C^{(3,3)}_{6\to 10}\downarrow_{(2,2)}\to C^{(3,3)}_{10\to 20}\downarrow_{(1,1)}\to C^{(3,3)}_{20\to 40}\downarrow_{(1,1)}\to C^{(3,3)}_{40\to 50}\downarrow_{(2,2)}\to C^{(3,3)}_{50\to 60}\downarrow_{(1,1)}$$

Partial row/column encoders Enc1, Enc2: $C^{(3,3,3)}_{5\to 20}\downarrow_{(1,1,1)}\to C^{(3,3,3)}_{20\to 40}\downarrow_{(1,2,2)}\to C^{(3,3,3)}_{40\to 60}\downarrow_{(1,1,1)}$

Partial common encoder Enc3: $C^{(3,3)}_{140\to 200}\downarrow_{(1,1)}\to C^{(3,3)}_{200\to 250}\downarrow_{(2,2)}\to C^{(3,3)}_{250\to 300}\downarrow_{(1,1)}$

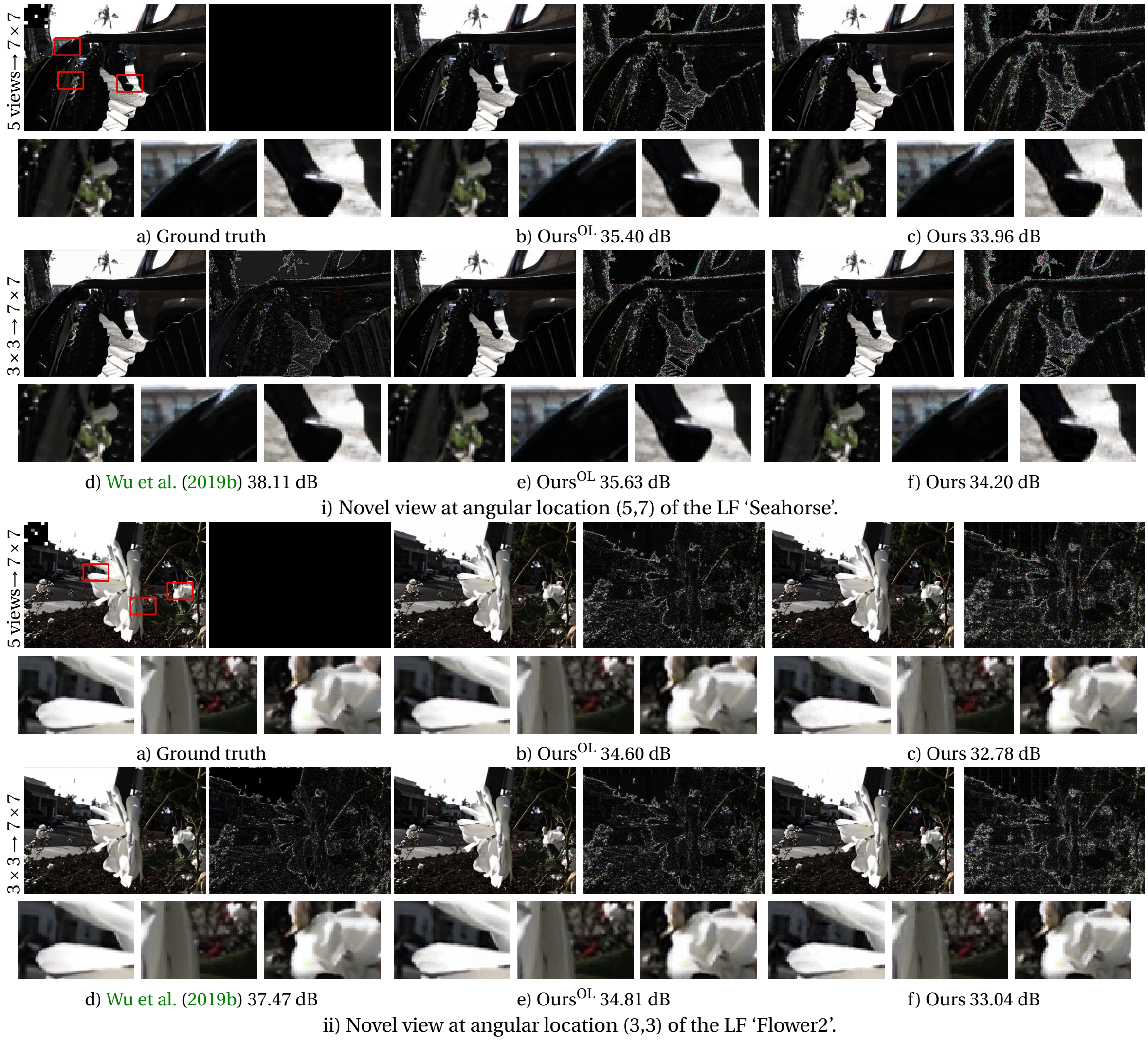
Partial common decoder of Dec1: $C^{(3,3)}_{300\to 250}\uparrow_{(1,1)}\to C^{(3,3)}_{250\to 200}\uparrow_{(2,2)}\to C^{(3,3)}_{200\to 120}\uparrow_{(1,1)}$

Partial row/column decoder Dec2, Dec3: $C^{(3,3,3)}_{140\to 80}\uparrow_{(1,1,1)}\to C^{(3,3,3)}_{80\to 40}\uparrow_{(1,2,2)}\to C^{(3,3,3)}_{40\to 20}\uparrow_{(1,1,1)}$

All the convolutional layers except the last layer of the generator are followed by batch norm and ReLU non-linearity. We fix the latent dimension of CWAE to be 160. We used isotropic Gaussian prior, with variance of 2 for the latent space. The architecture is same for both the angular resolutions 5 × 5 and 7 × 7, except for padding in the first convolutional layer.

7.B ADDITIONAL RESULTS

7 × 7 *View Synthesis*



a) Ground truth b) Ours^{OL} 35.40 dB c) Ours 33.96 dB

d) Wu et al. (2019b) 38.11 dB e) Ours^{OL} 35.63 dB f) Ours 34.20 dB

i) Novel view at angular location (5,7) of the LF ‘Seahorse’.

a) Ground truth b) Ours^{OL} 34.60 dB c) Ours 32.78 dB

d) Wu et al. (2019b) 37.47 dB e) Ours^{OL} 34.81 dB f) Ours 33.04 dB

ii) Novel view at angular location (3,3) of the LF ‘Flower2’.

Figure 7.14: Result of 7 × 7 view synthesis for the LFs i) ‘Seahorse’ ii) ‘Flower2’. Shown are the novel view at angular locations, depicted as gray location in the inset. The mask for selecting 5 input views is shown in the inset of ground truth view. Figures in the first row a)−c) depict ground truth view, and the results of our approach using 5 input views with and without overlapping patches in that order. Figures d)−f) in the second row provide visual comparison of novel views generated using approach of Wu *et al.* Wu et al. (2019b), and our approach using 3 × 3 angular views. Error maps and zoomed in patches are depicted along with corresponding novel views, with error magnified by a factor of 10. Results best viewed when zoomed in.

Results of our quantitative evaluation on 30 real LFs from the test set of Kalantari et al. (2016) and 7 real LFs selected from Stanford Lytro archive (Sunder Raj et al., 2016) are provided in Tab. 7.7. 'Ours$^{\text{OL}}$' indicates our reconstruction using overlapping patches with stride 5. For each LF, we report the result of average PSNR of the luminance component of novel synthesized views.

We show additional qualitative results of 7 × 7 LF reconstruction from 5 input views, and 3 × 3 input views from Kalantari testset in Fig. 7.14. The selected 5 input views are depicted in white and the novel view displayed is depicted in gray in the inset of the ground truth views. Shown here are ground truth and reconstructed views for the LFs 'Seahorse' and 'Flower2' from 30 real scenes set using our approach and network based approach of Wu et al. (2019b). Reconstructed views along with corresponding error maps and zoomed-in patches are provided for visual comparison. Zoomed-in patches show good reconstruction quality at occlusion boundaries.

Fig. 7.15 shows visual comparison for 7×7 LF reconstruction from 3×3 input views for the LF 'Reflective22', which contains specularities and transparencies. Novel views at locations depicted in gray in the inset of the ground truth views are shown, along with error maps and zoomed-in patches. We observe reasonably good quality of reconstruction with our approach. Closer inspection of zoomed-in patches reveals that specularities and transparencies are also handled reasonably well by our approach, however, the reconstruction is slightly blurred when compared to the end-to-end trained network (Wu et al., 2019b).

LF	3 × 3 → 7 × 7			5 views→ 7 × 7	
	EPICNN	Ours	Ours$^{\text{OL}}$	Ours	Ours$^{\text{OL}}$
Seahorse	38.11	34.20	35.63	33.96	35.40
Rock	38.24	32.86	34.93	32.55	34.71
Flower1	37.73	33.37	34.96	33.14	34.77
Flower2	37.47	33.04	34.81	32.78	34.60
Cars	36.02	31.74	33.45	31.55	33.30
1085	43.03	41.72	42.31	41.27	41.85
1086	43.75	42.80	43.70	42.40	43.27
1184	43.75	43.23	43.65	43.10	43.53
1187	43.20	42.11	42.80	42.00	42.72
1306	42.74	39.47	40.86	39.29	40.69
1312	45.66	44.33	45.55	44.14	45.39
1316	42.78	40.23	41.11	40.09	41.00
1317	41.67	39.39	40.20	39.24	40.07
1320	39.97	35.62	37.02	35.35	36.80
1321	46.07	44.62	45.72	44.43	45.55
1324	46.06	47.39	48.04	47.24	47.94
1325	44.16	43.00	43.92	42.85	43.78
1327	40.76	37.18	38.32	37.03	38.22
1328	44.19	41.35	42.82	41.05	42.55
1340	45.38	46.12	47.01	45.99	46.92
1389	45.63	44.76	46.35	44.60	46.23
1390	45.95	46.29	47.06	46.17	46.94
1411	36.13	32.84	33.84	32.68	33.69
1419	39.30	36.08	36.95	35.82	36.70
1528	36.28	30.91	32.68	30.50	32.36
1541	36.84	31.77	33.76	31.39	33.49
1554	33.54	28.78	30.21	28.46	29.93
1555	35.89	31.28	32.88	31.00	32.65
1586	42.44	38.98	40.88	38.75	40.74
1743	42.12	40.52	41.77	39.94	41.25
Avg. 30 scenes	41.16	38.53	39.77	38.29	39.57
Reflective9	45.82	46.47	47.01	46.10	46.69
Reflective13	35.76	34.03	35.06	33.73	34.87
Reflective22	43.17	41.78	42.45	41.20	41.93
Reflective27	43.75	43.92	44.36	43.47	44.06
Reflective29	43.40	41.67	42.65	41.19	42.25
Occlusions16	36.23	32.53	33.56	31.56	32.64
Bikes12	40.58	36.97	38.32	36.15	37.58
Avg. 7 scenes	41.24	39.62	40.48	39.07	40.00

Table 7.7: PSNR values in dB for 7 × 7 view synthesis for individual scenes in the test sets- 30 scenes from Kalantari et al. (2016), and 7 scenes containing reflections, refractions and occlusions from Sunder Raj et al. (2016). Comparison with EPICNN Wu et al. (2019b) is shown.

5 × 5 *View Synthesis*

We provide additional qualitative results for 5 × 5 LF recovery from a sparse subset of input views in Fig. 7.16 for the LFs 'Kitchen' and 'Medieval2' from the synthetic HCI dataset. We include a comparison of novel reconstructed views with ground truth and the reconstructions using approaches of Marwah et al. (2013); Jin et al. (2020). The superior quality of our reconstructions can be inferred from our reconstructions and the error maps in Fig. 7.16.

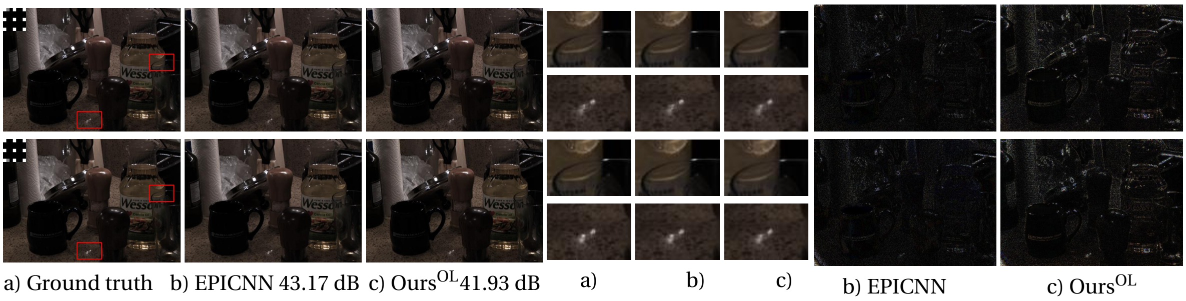


Figure 7.15: Visual comparison of our synthesized views (depicted by gray locations in the inset) for the task of 3 × 3 → 7 × 7 view synthesis for the LF 'Reflective22'. Columns 1 – 3 depict a) the ground truth views, the result using b) EPICNN Wu et al. (2019b) and c) our approach using overlapping patches respectively. Columns 4 – 6, the patches of columns 1 – 3. Columns 7 – 8 depict the error maps corresponding to columns 2 – 3 with error magnified by a factor of 10. The brightness of the zoomed in patches is increased for better illustration. Average PSNR in dB of 40 novel views is shown.

5 × 5 *Spatial angular super-resolution*

Additional visual comparisons for 5 × 5 LF recovery for the task of spatial angular super resolution are provided in Fig. 7.17 for the LFs 'Dino' and 'Medieval2' from the synthetic HCI dataset. Error maps show lower error in our reconstructions when compared to dictionary based approach of Marwah et al. (2013).

Effect of TV regularization on coded aperture reconstruction:

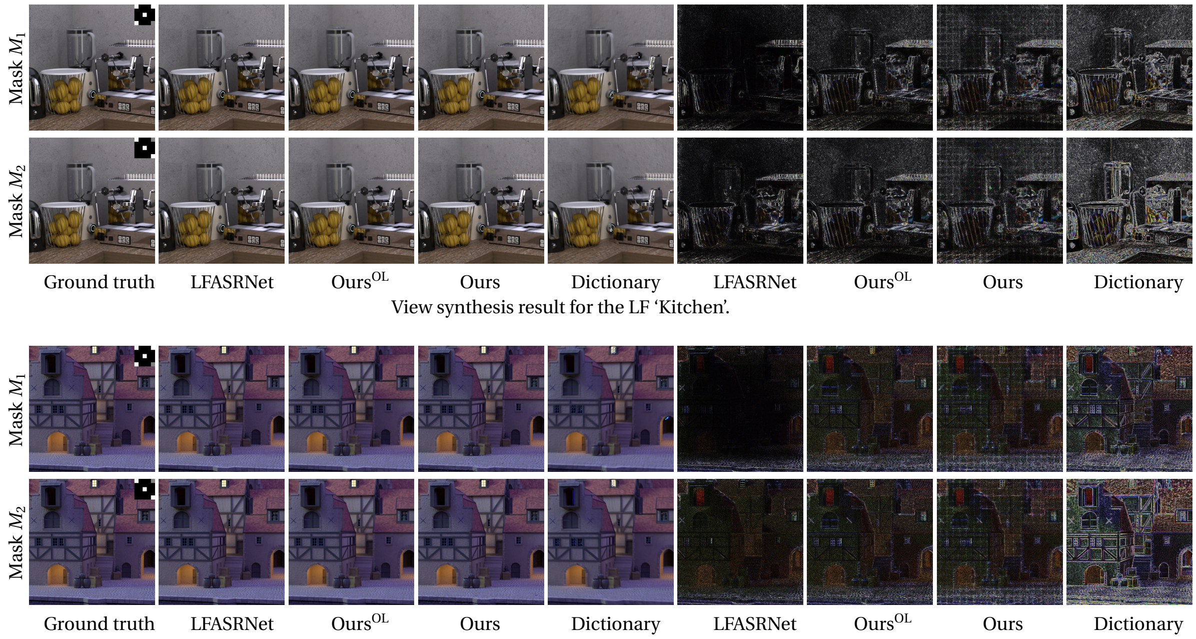
As discussed in Sec. 5 of the paper, we optimize for both the central view and the latent code, when the central view is unavailable. We also experimented with an additional total variation (TV) penalty on the central view for recovery from coded aperture. Tab. 7.8 shows the PSNR comparison, illustrating the effect of this additional regularization on coded aperture reconstruction. We can observe that using additional TV regularization on the central view results in a very marginal improvement when we do not use overlapping patches. When overlapping patches are used we obtain similar PSNR values, with and without this additional regularization. Visual comparisons for this task is provided in Fig. 7.18 for the LFs 'Dino' and 'Kitchen'. Visually, the reconstructions with and without TV regularization appear similar.

LF	without TV		with TV	
	Ours	Ours^{OL}	Ours	Ours^{OL}
Dino	34.97	38.46	35.26	38.49
Kitchen	31.07	33.29	31.20	33.32
Medieval2	32.90	35.19	32.92	35.20
Tower	29.02	30.43	29.06	30.42

Table 7.8: Effect of additional TV regularization on coded aperture reconstruction. Shown are PSNR values in dB.

7.B.1 *Effect of training with different datasets*

We trained our generative model using patches from light fields in the datasets provided by Kalantari et al. (2016), Heber and Pock (2016) and HCI (2018). From a total of 250K patches, 86K patches were from (Kalantari et al., 2016) consisting of real LFs captured through Lytro Illum, 117K patches were from (Heber and Pock, 2016) (synthetic LF data), and 47K patches were from (HCI, 2018) (synthetic LF data). To investigate the effect of training data, we trained separate generative models on LF patches extracted from each of the three datasets. We compare sample LF reconstructions from sparse input views using the following models: i) Model$_1$ trained on all the three datasets, ii) Model$_2$ trained on data from set of Kalantari



View synthesis result for the LF 'Kitchen'.

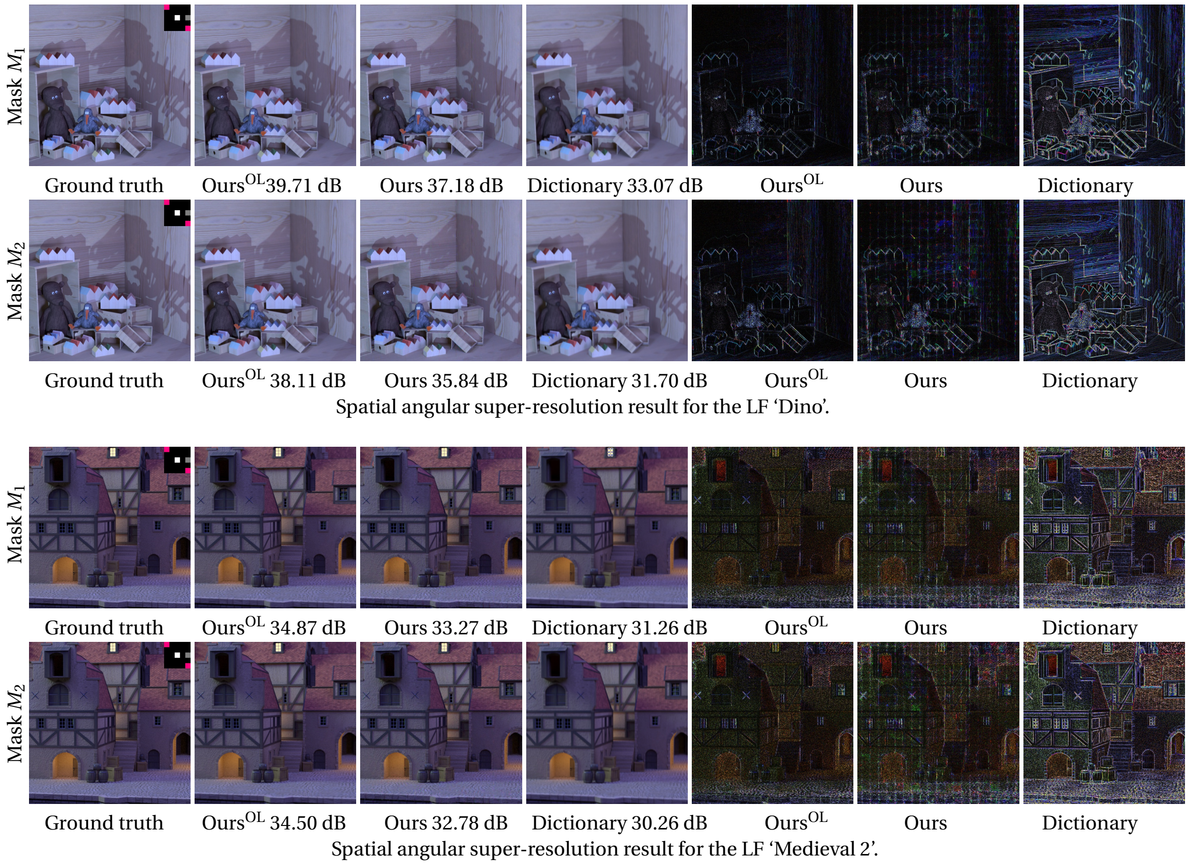
View synthesis result for the LF 'Medieval 2'.

Figure 7.16: Result of 5 × 5 view synthesis. Masks M_1 and M_2 are provided as inset of the ground truth views. The columns 1 – 5 show the views depicted by gray location in the inset corresponding to the ground truth and synthesized novel views using LFASRNet Jin et al. (2020), our approach with and without overlapping patches, and the dictionary based approach of Marwah et al. (2013), respectively in order. Columns 6 – 9 illustrate the error maps corresponding to the reconstructed views in columns 2 – 4, with error magnified by a factor of 10. (Results best viewed zoomed in)

et al. (2016), iii) Model$_3$ trained on data from (Heber and Pock, 2016), and iv) Model$_4$ trained on data in (HCI, 2018). We note that because of high disparities present in the datasets from (Heber and Pock, 2016) and (HCI, 2018), they are spatially down-scaled by a factor of 1.4 before LF patch extraction. Comparison for sample 5 × 5 view reconstruction on the synthetic LF 'Kitchen' and a real LF 'Cars' using each of these models is provided in Fig. 7.19. We can observe that the Model$_1$ trained on all the 3 datasets performs the best on both the LFs, whereas the Model$_4$ trained with data from (HCI, 2018) performs poorest. This is because Model$_1$ is trained with the largest and diverse training set among all the four models and therefore can generalize better, whereas the training set of Model$_4$ has the least number of training samples, resulting in poorer quality reconstructions. Further, Model$_2$ trained on real dataset of Kalantari et al. (2016) gives better reconstruction on the real LF 'Cars' compared to the models trained on synthetic datasets. However, on the synthetic LF 'Kitchen', which has higher disparities than typically found in Lytro LFs, Model$_3$ is better than Model$_2$.

7.B.2 *Effect of size of generative model*

To investigate the variation in performance with respect to the size of the generative model, we trained 7 × 7 generative models with different spatial extents 16 × 16 and 32 × 32 having 1.78M and 4.25M trainable parameters respectively, in addition to our original model with spatial extent 25x25 having 3.85 M parameters. We conduct 3 × 3 → 7 × 7 view interpolation experiment on 30 scenes of Kalantari test set using each of these models using our approach without overlapping patches. We found that reconstructions using both the models were only slightly worse compared to our original model, with an average PSNR of 38.25 dB using the model with spatial extent 16 × 16 and 38.31 dB using the model with spatial extent 32 × 32,



Spatial angular super-resolution result for the LF 'Dino'.

Spatial angular super-resolution result for the LF 'Medieval 2'.

Figure 7.17: 5 × 5 Spatial angular super-resolution results. Masks M_1 and M_2 are provided as inset of the ground truth views. Central view in white is available in full resolution. Views in red are down-sampled by a factor of 3. The columns 1 – 4 show the views depicted by gray location in the inset corresponding to the ground truth and synthesized novel views using our approach with and without overlapping patches, and the dictionary based approach of Marwah et al. (2013) respectively, in order. Columns 5 – 7 illustrate the error maps corresponding to the reconstructed views in columns 2 – 4, with error magnified by a factor of 10. PSNR values in dB of the reconstructed LF are provided.

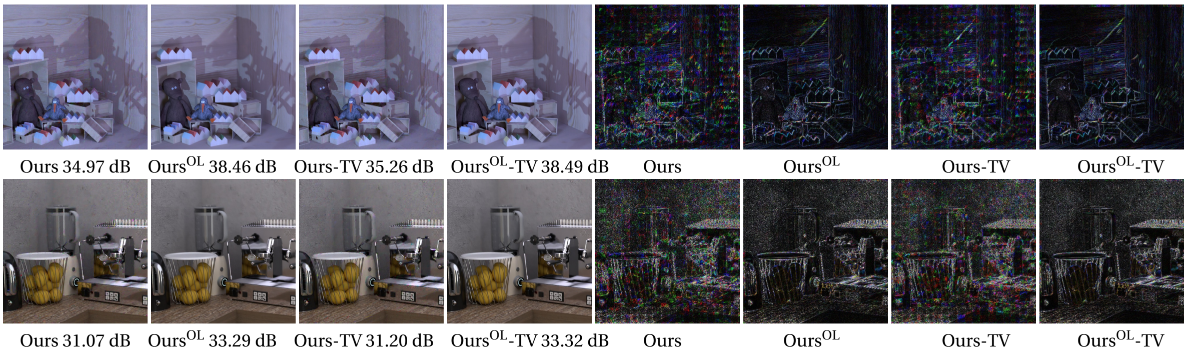


Figure 7.18: Effect of additional TV regularization on 5 × 5 coded aperture reconstruction. The columns 1 – 2 show our reconstruction without and with overlapping patches when TV regularization is not employed. The columns 3 – 4 show our reconstruction without and with overlapping patches when additional TV regularization is used. Columns 5 – 8 illustrate the error maps corresponding to the reconstructed views in columns 1 – 4, with error magnified by a factor of 10.(Results best viewed zoomed in)

whereas our original model leads to an average PSNR of 38.54 dB. However, these models also resulted in higher inference times, with the smaller model requiring 28 minutes for reconstructing full real Lytro LF due to a larger number of patches, and the larger model requiring 12.6 minutes, due to a slightly larger model size, compared to 11.8 minutes required by our approach.

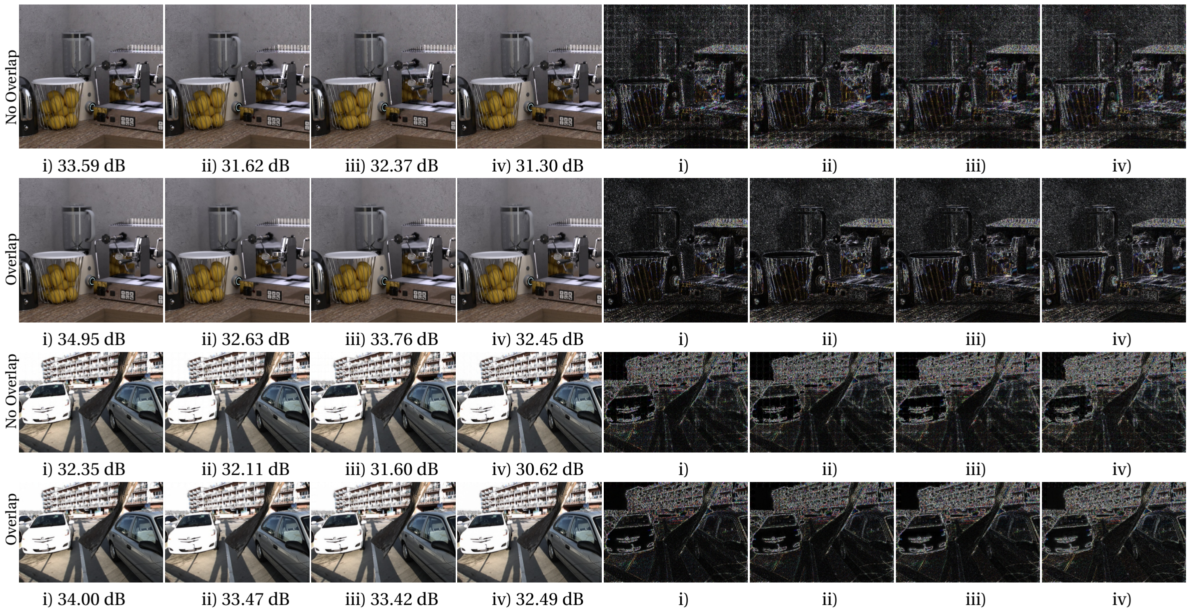


Figure 7.19: Comparison of 5 × 5 view reconstructions using models trained from each of the datasets (Kalantari et al., 2016; Heber and Pock, 2016; HCI, 2018). Shown is 5 × 5 view synthesis result using the mast M_1 for the LFs 'Kitchen' and 'Cars', shown is novel view at location (5,5). The rows 1 – 2 show our reconstruction without and with overlapping patches. The columns 1 – 4 depict our reconstruction using all the models trained on i) all the 3 datasets, ii) dataset of (Kalantari et al., 2016), iii) dataset of (Heber and Pock, 2016), iv) dataset of (HCI, 2018). Columns 5 – 8 illustrate the error maps corresponding to the reconstructed views in columns 1 – 4, with error magnified by a factor of 10. PSNR values in dB of the reconstructed LF are shown. (Results best viewed zoomed in)

Declaration for Chapter 8 - Generalized Text Guided Image Manipulation

This chapter is based on the paper Chandramouli et al. (2022) titled “LDEdit: Towards generalized text guided image manipulation via latent diffusion models” co-authored by Paramanand Chandramouli and Kanchana Vaishnavi Gandikota, published at the British Machine Vision Conference (BMVC) 2022.

Paramanand Chandramouli proposed this project idea of text guided image editing using latent diffusion models, and using denoising diffusion implicit models (DDIM) sampling process to achieve manipulations with near cycle consistency with input image. Kanchana Vaishnavi Gandikota proposed to use DDIM with controlled stochasticity to deal with manipulations that are difficult to achieve through deterministic sampling. Paramanand Chandramouli proposed and developed mask based editing to avoid undesired changes in images. Both the authors designed the experiments, and contributed to generating the results provided in the paper. Kanchana Vaishnavi Gandikota reviewed literature, designed and set up the user study, and contributed to writing all the sections in the first draft of the paper.

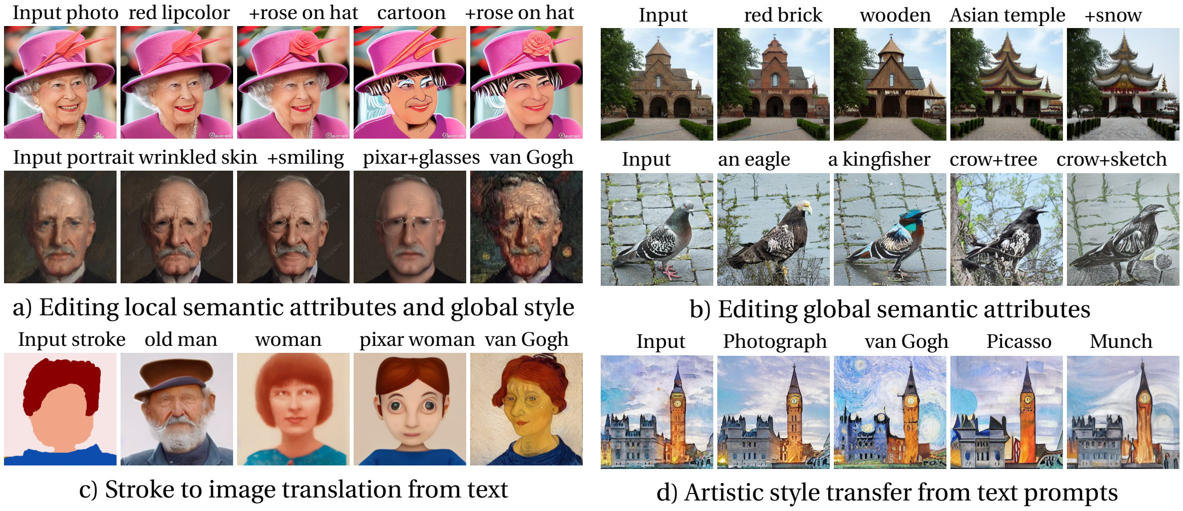
Most of the test images and text prompts used in evaluation are taken from the works VQGAN+CLIP Crowson et al. (2022) and DiffusionCLIP Kim et al. (2022), and GLIDE Nichol et al. (2022). All other images are taken from `pixabay.com` from the selection of free images available with contents license that allows users to use, modify and adapt content for free in non-commercial works without having to attribute the authors.

8 GENERALIZED TEXT GUIDED IMAGE MANIPULATION

Using natural language descriptions is an intuitive and easy way for humans to communicate visual concepts. Hence, a tool that can automatically manipulate images using textual descriptions can greatly ease editing. This requires a careful control to modify only the relevant semantic attributes and styles, while preserving the desired content representations. However, accomplishing this is highly challenging, especially when manipulating open-domain images using arbitrary text prompts. As a result, many existing works allow manipulations that are restricted to specific image classes (Patashnik et al., 2021; Xia et al., 2021; Gal et al., 2022; Kim et al., 2022) or a specific manipulation task (Avrahami et al., 2022; Nichol et al., 2022; Kwon and Ye, 2022). Further, some of these methods require fine-tuned models (Kim et al., 2022; Gal et al., 2022) for specific text prompts, further limiting their utility for flexible open domain image manipulation. Only a few prior works (Liu et al., 2020; Crowson et al., 2022) handle open domain image manipulation from text prompts. While Liu et al. (2020) focuses on semantically simple transformations, Crowson et al. (2022) allows a more general image generation as well as manipulation. Yet, Crowson et al. (2022) employs an expensive and time-consuming optimization to achieve these manipulations.

In this chapter, we attempt to develop a fast and flexible approach to open domain image manipulation using arbitrary text prompts. Our goal is to accomplish a wide range of manipulations from text prompts ranging from a simple change in colour of an object, to modification of multiple semantic attributes of an image, and artistic styles, all with a single model. Our work is inspired by the recent developments in realistic image generation with language guidance (Ramesh et al., 2021; Ding et al., 2021; Rombach et al., 2022; Ramesh et al., 2022). In particular, we leverage the recently proposed *Latent Diffusion Model (LDM)* (Rombach et al., 2022) which performs diffusion in a smaller dimensional latent space of trained convolutional auto-encoders, to provide higher inference speed and computational efficiency. Further, we utilize the idea of deterministic diffusion proposed in Denoising Diffusion Implicit Models (DDIM) (Song et al., 2021a) which can enable faster inference and high fidelity sample reconstruction. Our key idea is the use of a shared latent representation as a link between the source image and the desired target. To this end, we employ a deterministic DDIM sampling in the forward diffusion in the latent space of LDM. We use the same latent code along with the target text prompt to condition the reverse diffusion process, effectively achieving the desired transformation in the input image, while automatically maintaining consistency with the original content representation. Using this technique, we can accomplish a variety of image manipulation tasks using the pretrained LDM, in a zero-shot fashion without further optimization or fine-tuning. Further, by introducing controlled stochasticity, we can trade-off diversity for fidelity with the original image. This is especially useful when the desired target is very different from the original input. We refer to our approach as *LDEdit*.

Fig 8.1 illustrates the diverse image manipulation tasks that can be accomplished by our LDEdit using only text prompts. We can modify objects in the image while largely preserving the original pose or structure, see Fig. 8.1 b). LDEdit can accomplish simultaneous global style manipulation as well as fine-grained (multiple) attribute changes such as changes in expression, wrinkles, and makeup while preserving identity in human faces, see Fig. 8.1 a).



a) Editing local semantic attributes and global style

b) Editing global semantic attributes

c) Stroke to image translation from text

d) Artistic style transfer from text prompts

Figure 8.1: LDEdit can edit local and global semantic attributes and also perform artistic style transfer on real-world images using a single model.

Further, without requiring an input mask, simple local edits such as adding a flower on a woman's hat, or eyeglasses are achieved through text alone. Our approach can operate on diverse types of input images such as natural photographs, paintings, sketches, and strokes. By providing an intuitive target text prompt " a photograph of a woman" or a "pixar animation of a woman", our method can translate from stroke to a semantically consistent image in the corresponding domain, see Fig. 8.1 c). We can observe realistic details are hallucinated while transferring to the domain of natural photos, for example, wrinkles in the picture of an old man, in Fig. 8.1 c), or details in the clock Fig. 8.1 d). Further, artistic style transfer is also achieved via simple text prompts, such as "a Picasso style painting". It can be seen that our approach can accomplish manipulations that are semantically and stylistically consistent with the given target text prompt, while remaining faithful to original content.

By offering significant advantages in flexibility, faster run-times, and the capability to generate diverse samples in parallel, LDEdit can facilitate efficient user-guided editing. Our experimental results demonstrate that LDEdit can accomplish diverse manipulation tasks, in addition to achieving performance close to recent state-of-the-art baselines.

8.1 RELATED WORK

IMAGE GENERATIVE MODELS Ever since the seminal works of VAEs (Kingma and Welling, 2013) and GANs (Goodfellow et al., 2014), image generative models have achieved significant improvements, and modern generative models can generate highly photo-realistic images (Brock et al., 2019; Razavi et al., 2019; Karras et al., 2020, 2021; Esser et al., 2021b; Dhariwal and Nichol, 2021; Song et al., 2021a). While GANs (Goodfellow et al., 2014) achieve high quality generation, they are difficult to train and are prone to mode collapse. Likelihood-based models, (Kingma and Welling, 2013; Razavi et al., 2019) on the other hand, have a stable training and capture more diversity. Score based (Song and Ermon, 2019; Song et al., 2021b) or denoising diffusion (Ho et al., 2020; Sohl-Dickstein et al., 2015) models are a new class of likelihood-based models built from a hierarchy of denoising auto-encoders (Vincent et al., 2008). These models have recently demonstrated generative capabilities surpassing GANs (Dhariwal and Nichol, 2021; Nichol and Dhariwal, 2021). Yet, high quality diffusion models are computationally expensive to train, and have slower inference times than GANs, due to the expensive Markovian sampling and iterative network evaluations required for diffusion. These problems can be alleviated by accelerated stochastic sampling techniques or by performing diffusion in a smaller latent space (Rombach et al., 2022; Vahdat et al., 2021). Employing deterministic diffusion process (Song et al., 2021a) can also speed up inference, in addition to enabling high fidelity sample reconstruction, which can be exploited for image recovery and manipulation.

IMAGE MANIPULATION As images can be manipulated in various ways, (for example, artistic style, image translation, semantic manipulation, local edits), a variety of methods exist. Approaches for image translation include CNN based optimization using style and content images (Gatys et al., 2016), conditional GANs trained on pair of domains (Isola et al., 2017; Zhu et al., 2017; Almahairi et al., 2018; Zhao et al., 2020a), GANs for multi-domain translation (Choi et al., 2018, 2020) and more recently, conditional diffusion models (Sasaki et al., 2021; Saharia et al., 2022a). An alternate approach (Zhu et al., 2016; Brock et al., 2017) is to manipulate images in the latent space of pretrained GANs. StyleGANs (Karras et al., 2020, 2021) are a popular choice for such latent space editing due to their disentanglement properties in the latent space (Collins et al., 2020; Shen et al., 2020; Zhu et al., 2020; Abdal et al., 2020; Gu et al., 2020; Wu et al., 2021). This is achieved through optimization or by using encoders for GAN inversion (Tov et al., 2021; Richardson et al., 2021; Alaluf et al., 2021). However, GAN inversion may not yield faithful reconstruction (Bau et al., 2019b). Improving StyleGAN inversion for editing is an active area of research (Tov et al., 2021; Alaluf et al., 2021; Dinh et al., 2022; Wang et al., 2022b; Alaluf et al., 2022). In contrast to GANs, diffusion models can readily be leveraged for inpainting (Lugmayr et al., 2022a) and stroke guided image editing (Meng et al., 2022) and even unpaired image translation (Su et al., 2023).

TEXT GUIDED GENERATION AND MANIPULATION: Starting from (Mansimov et al., 2016) many works proposed different methods to generate images from text prompts. Earlier works employed RNNs (Mansimov et al., 2016) and GANs (Reed et al., 2016; Zhang et al., 2017a; Xu et al., 2018; Zhang et al., 2017a, 2018a; Zhu et al., 2019; Li et al., 2019; Zhang et al., 2021a; Zhu et al., 2022) for text guided image synthesis, and manipulation (Dong et al., 2017; Li et al., 2020a; Nam et al., 2018). Nevertheless, these works are often restricted to class-specific image generation, and are trained on smaller datasets. In the recent past, there has been a rapid surge in vision-language models, with the developments in cross-modal contrastive learning (Radford et al., 2021; Jia et al., 2021) and powerful text-to-image generative models (Ramesh et al., 2021; Nichol et al., 2022; Ramesh et al., 2022; Saharia et al., 2022c). These models are trained on massive datasets to learn joint image-text distributions. Some of these models (Ramesh et al., 2021; Ding et al., 2021; Gafni et al., 2022) use autoregressive(AR) transformers for generation, while some others (Nichol et al., 2022; Ramesh et al., 2022; Saharia et al., 2022c) employ diffusion based models for the generation task. However, training these models for high quality generation requires massive computational resources. To address this, some recent works (Gu et al., 2022c; Tang et al., 2022; Rombach et al., 2022; Bond-Taylor et al., 2022; Esser et al., 2021a; Hu et al., 2022) instead perform the diffusion in a lower dimensional latent space resulting in faster training and inference. In our work, we exploit Latent Diffusion Models (LDM) (Rombach et al., 2022) as they offer good reconstruction quality, and latency, and perform diffusion in a continuous latent space.

Method	Image input	Text input	Semantic	Artistic Style	local edits	Comments
			(Global)			
DiffusionCLIPKim et al. (2022)	Class-Specific	Predefined	✓	✓	✗	Separately fine-tuned models for each task
StyleCLIPPatashnik et al. (2021)	Class-specific	Arbitrary	✓	✗	✗	Includes versions with and without optimization
GLIDENichol et al. (2022)	Open domain	Arbitrary	✗	✗	✓	Trained model for inpainting with mask input
CLIPStylerKwon and Ye (2022)	Open domain	Arbitrary	✗	✓	✗	Test-time optimization w/o pretrained generator
VQGAN+CLIPCrowson et al. (2022)	Open domain	Arbitrary	✓	✓	Limited	Optimization with pretrained generator
LDEdit(Ours)	Open-domain	Arbitrary	✓	✓	✓	A pretrained LDM is used

Table 8.1: Comparison of recent state of the methods for text guided image manipulation.

Contrastive Language-Image Pretraining (CLIP) model (Radford et al., 2021) is a cross-modal encoder that provides a similarity score between an image and a caption. Several recent approaches to text guided image synthesis (Galatolo et al., 2021; Crowson et al., 2022; Liu et al., 2021, 2023a; Couairon et al., 2022; Paiss et al., 2022) steer pretrained generative models (Brock et al., 2019; Esser et al., 2021b; Dhariwal and Nichol, 2021) towards a user provided text prompts using the similarity score provided by the CLIP model. This approach of CLIP guided latent space navigation is directly applicable for image manipulation (Crowson et al., 2022), mask guided local editing (Bau et al., 2021; Avrahami et al., 2022), semantic manipulation of class-specific images (Patashnik et al., 2021; Yu et al., 2022; Abdal et al., 2022) via StyleGAN inversion (Alaluf et al., 2021). CLIP has also been applied to fine-tune the output domain and style (Gal et al., 2022; Kim et al., 2022) of class-specific image generators. While these approaches are promising, optimization in latent space for each text-prompt is expensive and time-consuming. On the other hand, the fine-tuned models are fast, but are restricted to the specific fine-tuned tasks. Further, class-specific generators are not suited for the manipulation of open domain images. Instead of using pretrained generative models, some recent works employ test-time optimization for each image and target text, using CLIP, for tasks such as local object appearance (Bar-Tal et al., 2022), global texture-style manipulation (Kwon and Ye, 2022), rendering drawings (Frans et al., 2021; Chen et al., 2021), however such optimization is task specific, and is expensive requiring many augmentations. Tab. 8.1 provides an overview comparing the pros and cons of recent methods for text guided manipulation. As we can see, our approach and VQGAN+CLIP (Crowson et al., 2022) can accomplish flexible manipulation tasks. Additionally, our approach allows fast manipulations.

8.2 PRELIMINARIES

8.2.1 *Denoising Diffusion Implicit Models*

Recall from Chapter 4 that denoising diffusion probabilistic models (DDPM) have a Markovian forward process where random Gaussian noise is gradually added to a sample, following a fixed noise variance schedule, and a learned reverse process that reverses the dynamics of this forward process via iterative denoising. Denoising Diffusion Implicit Models(DDIM) (Song et al., 2021a) construct a deterministic diffusion process which can utilize a trained DDPM model, to allow a much faster sampling in the reverse process. We recall DDPM forward process eq. (4.5):

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{(1-\bar{\alpha}_t)}\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{8.1}$$

where $\bar{\alpha}_t = \prod_{i=0}^{t} 1 - \beta_i$, with $\{\beta_t\}_{t=0}^{T}$ being the noise variance schedule. The trained noise approximator ϵ_θ can predict ϵ at t from $\mathbf{x}_t$, without the knowledge of $\mathbf{x}_0$. Rewriting eq. (8.1), Song et al. (2021a) obtain an estimate of clean sample $\mathbf{x}_{0|t}$ from $\mathbf{x}_t$ as follows:

$$\mathbf{x}_{0|t} = \left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}\right). \tag{8.2}$$

Song et al. (2021a) utilize this to construct fully deterministic forward and reverse sampling processes, enabling fast transformation into latent space $\mathbf{x}_T$, and fast inversion to $\mathbf{x}_0$ from the latent space. The deterministic DDIM forward process can be expressed as:

$$\mathbf{x}_{t+1} = \sqrt{\bar{\alpha}_{t+1}}\mathbf{x}_{0|t} + \sqrt{1-\bar{\alpha}_{t+1}}\epsilon_\theta(\mathbf{x}_t, t), \tag{8.3}$$

and the deterministic reverse DDIM process is expressed as:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}}\epsilon_\theta(\mathbf{x}_t, t), \tag{8.4}$$

where, $\bar{\alpha}_0 := 1$ by definition. Further, Song et al. (2021a) introduce stochasticity into the sampling process, resulting in a reverse DDIM process that can be expressed as

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta(x_t, t) + \sigma_t^2\xi, \tag{8.5}$$

where $\xi \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Varying σ leads to different generative processes with the same model ϵ_θ. When σ_t is set to 0, the DDIM sampling becomes fully deterministic, enabling fast inversion of the noised latent variable to the original images ($\mathbf{x}_0$ in our case). For different subsequences τ in $[1, \ldots, T]$ Song et al. (2021a) consider σ of the form:

$$\sigma_{\tau_i}(\eta) = \eta\sqrt{(1-\bar{\alpha}_{\tau_{i-1}})/(1-\bar{\alpha}_{\tau_i})}\sqrt{1-\bar{\alpha}_{\tau_i}/\bar{\alpha}_{\tau_{i-1}}}, \tag{8.6}$$

where the hyperparameter $\eta \in \mathbb{R}_{\geq 0}$ controls the degree of stochasticity, with $\eta = 1$ leading to the original DDPM generative process and $\eta = 0$ leading to DDIM.

8.2.2 *Latent Diffusion Models*

The main idea of latent diffusion models (LDM) (Rombach et al., 2022) is to perform diffusion in the latent space of an autoencoder to improve speed and computational efficiency. Given an image $x_{\text{src}} \in \mathbb{R}^{H\times W\times C}$, the encoder $\mathcal{E}$ maps x_{src} into a down-sampled latent code $z_0 = \mathcal{E}(x_{\text{src}})$, and the decoder $\mathcal{D}$ is trained to recover the image from this latent. This encoding results in a lossy compression, i.e. $\|\mathcal{D}(\mathcal{E}(x_{\text{src}})) - x_{\text{src}}\|$ is non-zero, which is a trade-off for computational efficiency. Following encoding into latent space, diffusion steps can be performed via DDPM or DDIM, but in z_t for $t \in [1, T]$ instead of $\mathbf{x}_t$. The diffusion process can additionally be conditioned on user inputs such as text prompts $\epsilon_\theta(z_t, t, \tau_{\tilde{\theta}}(y))$. Here, the text-prompts y are tokenized using transformers $\tau_{\tilde{\theta}}$ (Vaswani et al., 2017) for conditioning the diffusion process.

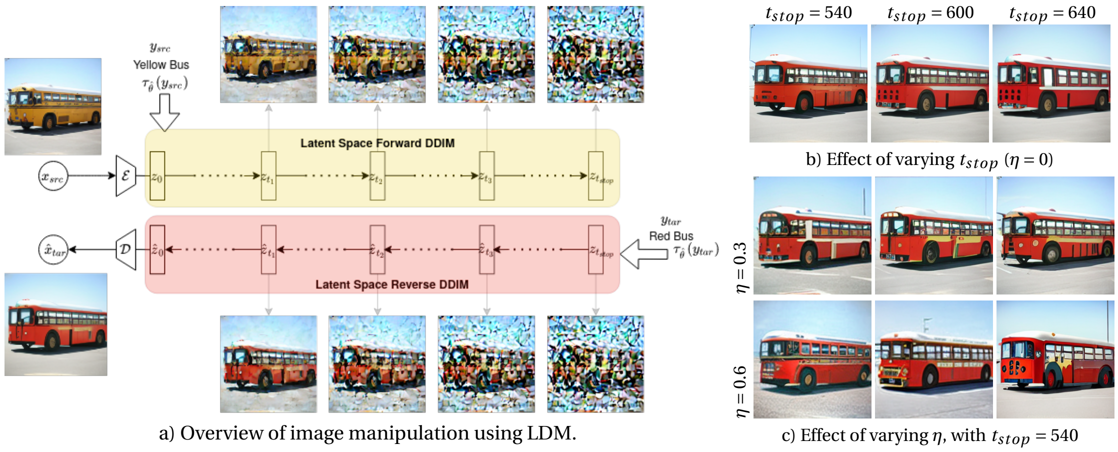
8.3 TEXT DRIVEN MANIPULATION WITH LDEDIT

In this section, we show how LDMs trained for text-to-image generation can be adapted for image manipulation. Our main idea is to use a common shared latent representation between the source image and the desired target, which is made possible by a deterministic diffusion process. The source image x_{src} is mapped to a latent code z_0 by the encoder $\mathscr{E}$, and forward diffusion is performed until the time step $t_{stop} < T$ using DDIM sampling, conditioned on the source text prompt y_{src} as:

$$\begin{aligned} z_{t+1} &= \sqrt{\bar{\alpha}_{t+1}} z_{0|t,y_{src}} + \sqrt{1-\bar{\alpha}_{t+1}}\epsilon_\theta(z_t, t, \tau_{\tilde{\theta}}(y_{src}))), \\ \text{where,} \quad z_{0|t,y_{src}} &= \left(\frac{z_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(z_t, t, \tau_{\tilde{\theta}}(y_{src}))}{\sqrt{\bar{\alpha}_t}} \right). \end{aligned} \tag{8.7}$$

The reverse diffusion conditioned on the target text prompt y_{tar} starts from the same noised latent code $z_{t_{stop}}$ to arrive at $\hat{z}_0$:

$$\begin{aligned} z_{t-1} &= \sqrt{\bar{\alpha}_{t-1}} z_{0|t,y_{tar}} + \sqrt{1-\bar{\alpha}_{t-1}}\epsilon_\theta(y_t, t, \tau_{\tilde{\theta}}(y_{tar})), \\ \text{where,} \quad z_{0|t,y_{tar}} &= \left(\frac{z_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(z_t, t, \tau_{\tilde{\theta}}(y_{tar}))}{\sqrt{\bar{\alpha}_t}} \right). \end{aligned} \tag{8.8}$$



a) Overview of image manipulation using LDM.

b) Effect of varying t_{stop} ($\eta = 0$)

c) Effect of varying η, with $t_{stop} = 540$

Figure 8.2: a) Overview of LDEdit, illustrating forward and reverse diffusion in latent space of autoencoder. b) and c) illustrate the effects of varying time steps t_{stop} and stochasticity hyperparameter η respectively

Due to deterministic sampling, a near cycle-consistency is automatically maintained between source and target images (Su et al., 2023). Fig. 8.2 a) provides an overview of our approach, with an example where a source image with y_{src} 'a yellow bus', is transformed according to the y_{tar} 'a red bus' in a straightforward way. The visualized results obtained by decoding latents sampled in $[1, t_{stop}]$ during the forward and reverse diffusion process demonstrate the gradual transformation in the reverse process. Additionally, we can also introduce controlled stochasticity by varying η eq. (8.6), which can produce diverse outputs as seen in Fig. 8.2 c), with magnitude of η controlling consistency with the original image. Further, Fig. 8.2 b) shows that changing the number of DDIM steps can also lead to some variance in our results. In the following section, we demonstrate that this technique can accomplish a variety of

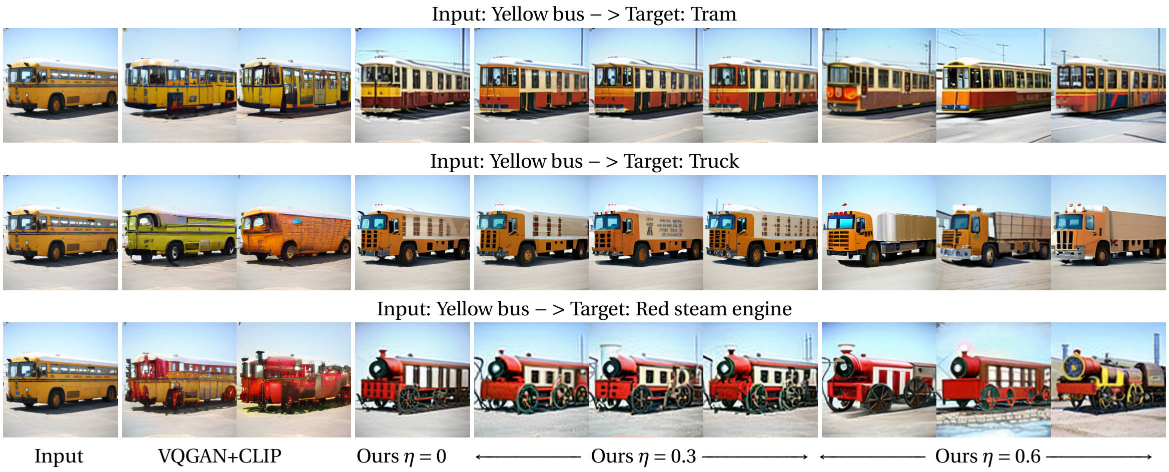

Input: Yellow bus – > Target: Tram
Input: Yellow bus – > Target: Truck
Input: Yellow bus – > Target: Red steam engine
Input
VQGAN+CLIP
Ours $\eta = 0$
Ours $\eta = 0.3$
Ours $\eta = 0.6$

Figure 8.3: Comparison with VQGAN+CLIP Crowson et al. (2022): Manipulation results of a yellow bus according to target texts 'a tram', 'a truck', and 'a red steam engine'.

image manipulation tasks using the pretrained LDM, in a zero-shot fashion without further optimization or fine-tuning.

8.4 EXPERIMENTS

We perform all our experiments with different image manipulation tasks using the text-to-image LDM (Rombach et al., 2022) with a downsampling factor of 8, pretrained using the openly available LAION dataset (Schuhmann et al., 2021) containing open-domain image-text pairs. We do not fine-tune this model for any task. We set $t_{stop} \in [300, 640]$ out of the total 1000 steps and use fewer (20-80) steps between $[1, t_{stop}]$ in the deterministic forward and reverse diffusion. We perform experiments on both class-specific and open-domain images and compare with VQGAN+CLIP (Crowson et al., 2022) which is versatile to handle general manipulation tasks. In addition, we also compare with class-specific approaches (Patashnik et al., 2021; Xia et al., 2021) and fine-tuned models (Gal et al., 2022; Kim et al., 2022) on the domain-specific tasks. Comparisons with the baseline methods and run-time comparisons are performed with images of dimension 256 × 256.

We first demonstrate our method on the task of manipulating an image of a yellow bus according to the target prompts: 'a tram', 'a truck' and 'a red steam engine'. Fig. 8.3 illustrates the results of this manipulation. The results indicate that LDEdit is able to manipulate the input according to the target texts even with a simple DDIM forward and reverse process with $\eta = 0$. Further, by increasing η, our method is able to generate an assortment of diverse samples that are consistent with the pose of the yellow bus in the input image. The diversity increases as the parameter η is increased. We also illustrate the results obtained by VQGAN+CLIP (Crowson et al., 2022) on this task using two sets of hyper-parameters for comparison. While Crowson et al. (2022) can successfully transform the input image to that of 'a tram', we were unable to obtain satisfactory results for the other two tasks, despite manual hyper-parameter tuning.

We further test our approach on manipulating images from diverse classes using test images from (Kim et al., 2022). We compare our performance with the generic approach of VQGAN+CLIP (Crowson et al., 2022) and DiffusionCLIP (Kim et al., 2022), a state of the art method using class-specific models fine-tuned for the specific target texts. Fig. 8.4 illustrates

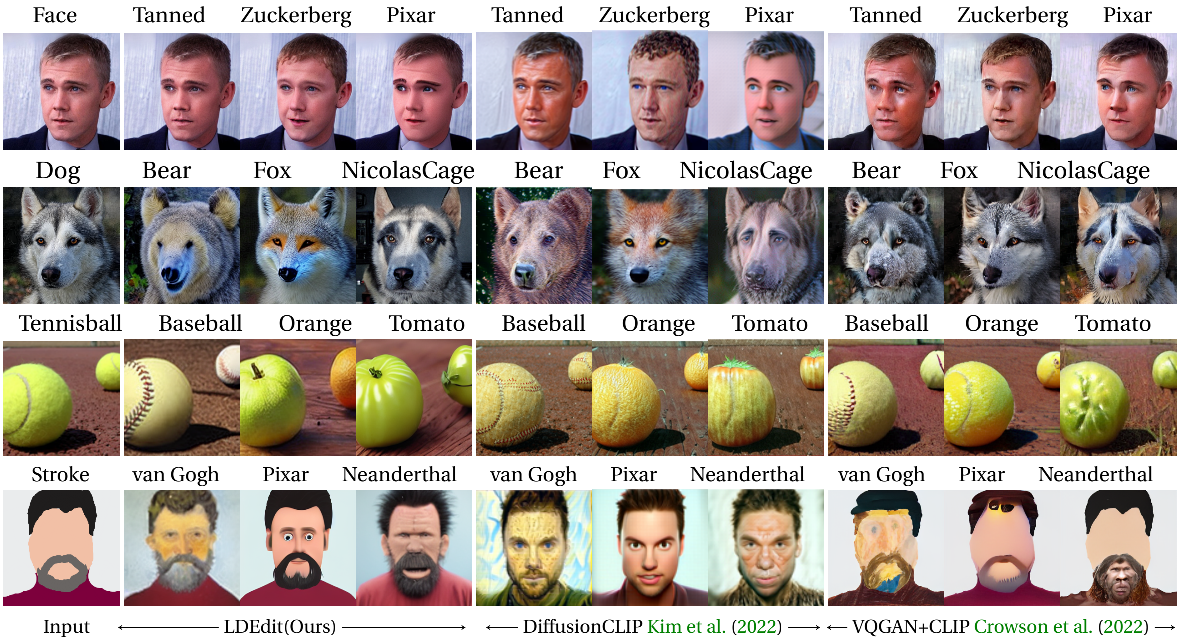


Figure 8.4: Visual comparison of image manipulation task with DiffusionCLIP Kim et al. (2022) and VQGAN+CLIP Crowson et al. (2022). Our LDEdit can successfully transform input image into target classes while retaining the original pose. Test images and target prompts from Kim et al. (2022).

the results of this experiment. As DiffusionCLIP uses specific fine-tuned models on these tasks, it can effortlessly accomplish the desired manipulations. Yet, the manipulation results provided by DiffusionCLIP, occasionally do not preserve the content in the source image, for example, the color of the dog in the row 2, moustache and color of clothing in the stroke painting in the row 4 are not preserved. On the other hand, VQGAN+CLIP preserves source content better, yet, struggles to achieve desired changes when the target is highly different from the input. Despite not being fine-tuned for the specific tasks, our LDEdit can accomplish the manipulations quite well. The task of manipulating a stroke image according to the target prompts is particularly challenging, as the input image lacks details. Handling such manipulation requires introducing stochasticity in the forward process, without which it is not possible to produce the desired edits.

We further perform multiple manipulation tasks on face images, including semantic (multi)-attribute manipulation, style transfer, domain manipulation and compare with the recent state-of-the-art methods which are trained for face manipulation (Kim et al., 2022; Patashnik et al., 2021; Gal et al., 2022; Xia et al., 2021). The StyleGAN based methods (Patashnik et al., 2021; Gal et al., 2022; Xia et al., 2021) employ the same encoders for GAN inversion as per the original setting in their work. Further, we include a comparison with CLIP-Styler (Kwon and Ye, 2022) a CLIP guided texture manipulation approach, and VQGAN+CLIP (Crowson et al., 2022) which can perform flexible image manipulation. Fig. 8.5 illustrates our results. While StyleGAN inversion based approaches (Patashnik et al., 2021; Gal et al., 2022; Xia et al., 2021) can manipulate semantic attributes see Fig.8.5 c), they struggle to reconstruct face images in atypical poses, see Fig.8.5 a). Unexpected details present in the original image such as the hand on the face are completely removed or distorted in the reconstructions. Since such atypical faces are hardly encountered during training, StyleGAN inversion results in a high representation error. Similarly, it is hard to transfer to a different style for example, a watercolour painting, or domain, for example, zombie using StyleGAN latent space search alone

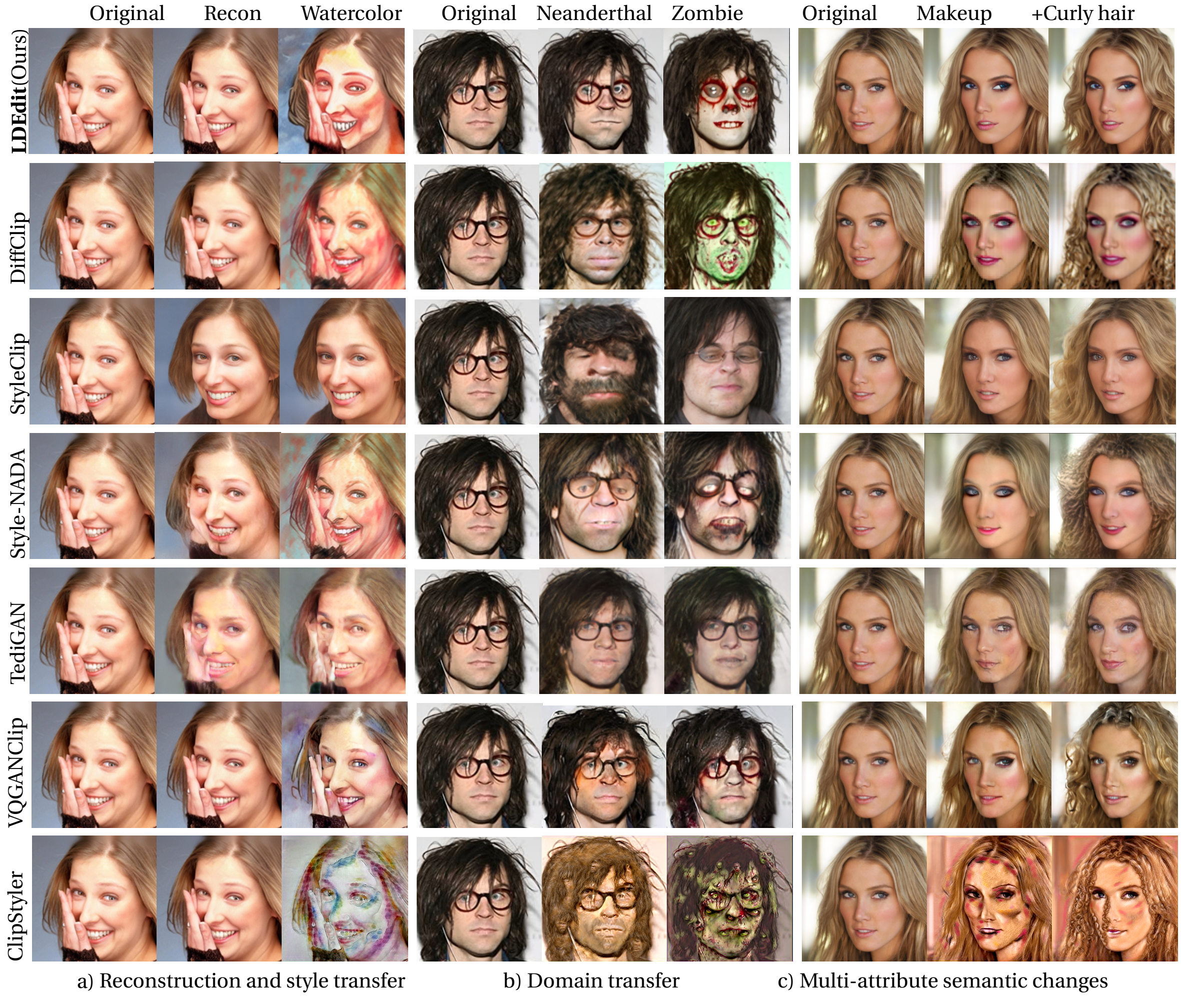


Figure 8.5: Comparison with recent baselines: DiffusionCLIP Kim et al. (2022) StyleCLIP Patashnik et al. (2021), StyleGAN-NADA Gal et al. (2022), TEDIGAN Xia et al. (2021), CLIPStyler Kwon and Ye (2022), VQGAN+CLIP Crowson et al. (2022). Test images and target prompts from Kim et al. (2022)

Fig.8.5 a) and b). StyleGAN-NADA instead enable these manipulations using domain-specific fine-tuning. On the other hand, ClipStyler (Kwon and Ye, 2022) can only accomplish global texture manipulations, and the result may drift away from the original colour palette. Among the compared methods, LDEdit, DiffusionCLIP (Kim et al., 2022) and VQGAN+CLIP(Crowson et al., 2022) accomplish the different manipulation tasks in addition to achieving good reconstructions, preserving identity better than StyleGAN inversion based methods. Interestingly, though VQGAN+CLIP and LDEdit are trained on generic images, these methods are still able to perform on par with state-of-the-art fine-tuned DiffusionCLIP (Kim et al., 2022) on these tasks.

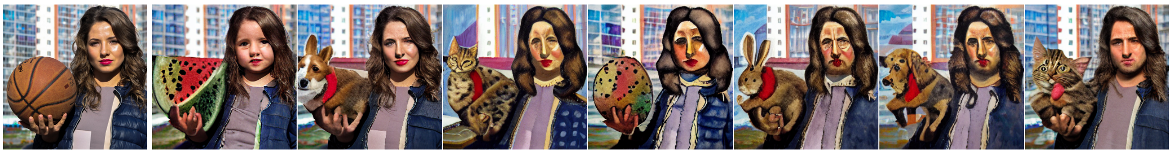


Figure 8.6: Simultaneous editing of multiple attributes and objects of an image. Shown from left to right are (i) input (ii) girl+watermelon (iii) woman+corgi (iv) paint+cat+old woman (v) paint + boy+ big egg (vi) paint + man + rabbit (vii) paint + man + dog (viii) man+cat

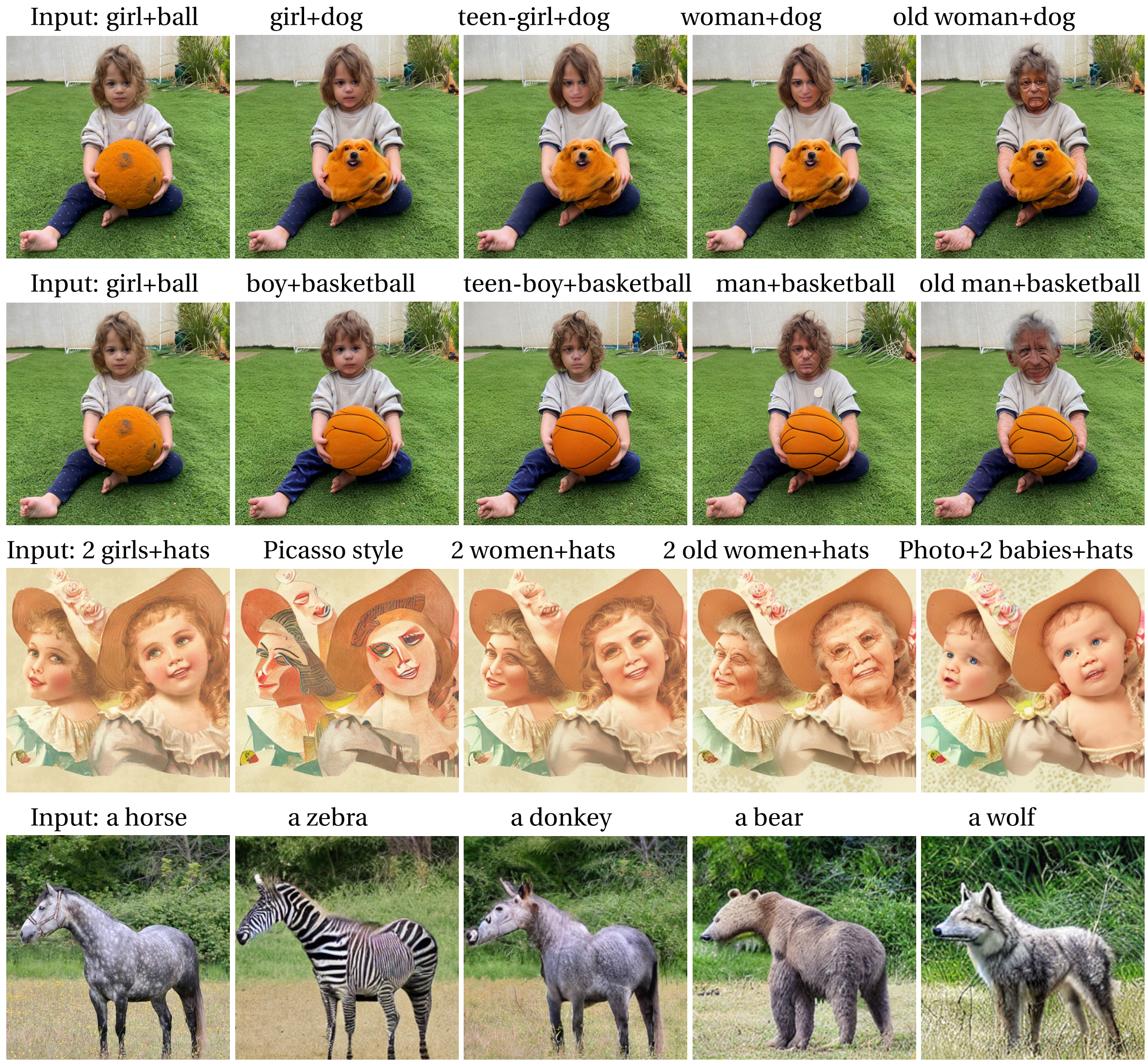
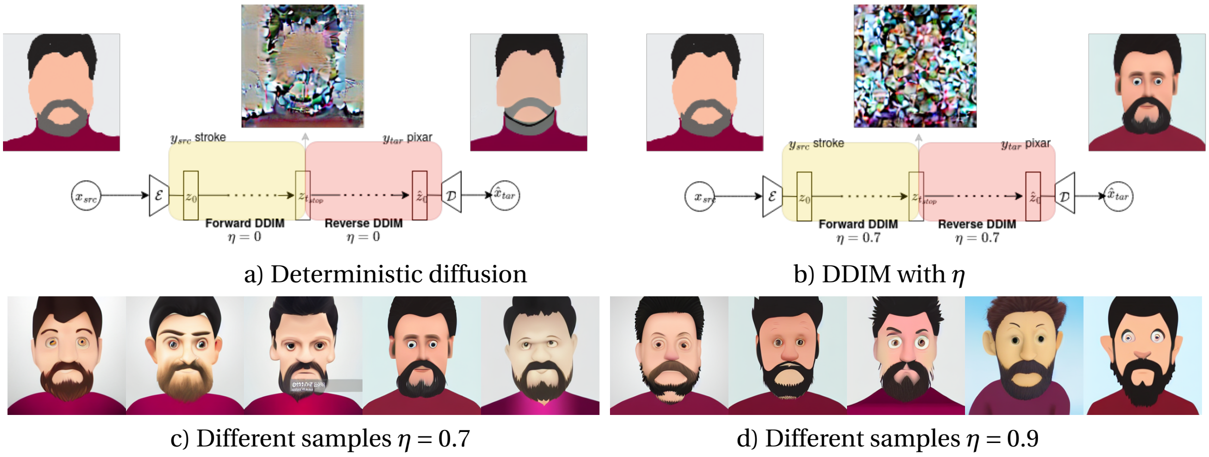


Figure 8.7: More results of image manipulation using LDEdit

It is also possible to achieve further challenging manipulations involving simultaneous changes in multiple attributes, local manipulations, and artistic style changes as seen in Fig. 8.6. While the LDM model is trained on generation of images of dimension 256×256, due to fully convolutional nature of the autoencoder, our method can be applied to images of higher resolution using the same model. Fig. 8.7 shows further example results of image manipulation using LDEdit, with image resolution 512 × 512. It is seen that our method can achieve varied transformations in a straightforward way. The first two rows show simultaneous manipulation of the girl and the ball. The third row shows style transfer to a painting or a photo and semantic manipulation of the age of two girls. Interestingly, LDEdit can effect such transformations with little or no stochasticity, $\eta = 0$ or 0.1, such that the background remains largely unaffected. The final row shows manipulating a horse to other species, for example, a zebra, a donkey, a bear, and a wolf. These transformations required a higher η of 0.3 for zebra and donkey, and η of 0.8 for bear and wolf. However, higher values of η result in more changes in the background.

Effect of Stochasticity



a) Deterministic diffusion

b) DDIM with η

c) Different samples $\eta = 0.7$

d) Different samples $\eta = 0.9$

Figure 8.8: Effect of η in diffusion process. Purely deterministic DDIM process cannot achieve desired target when the original input lacks details.

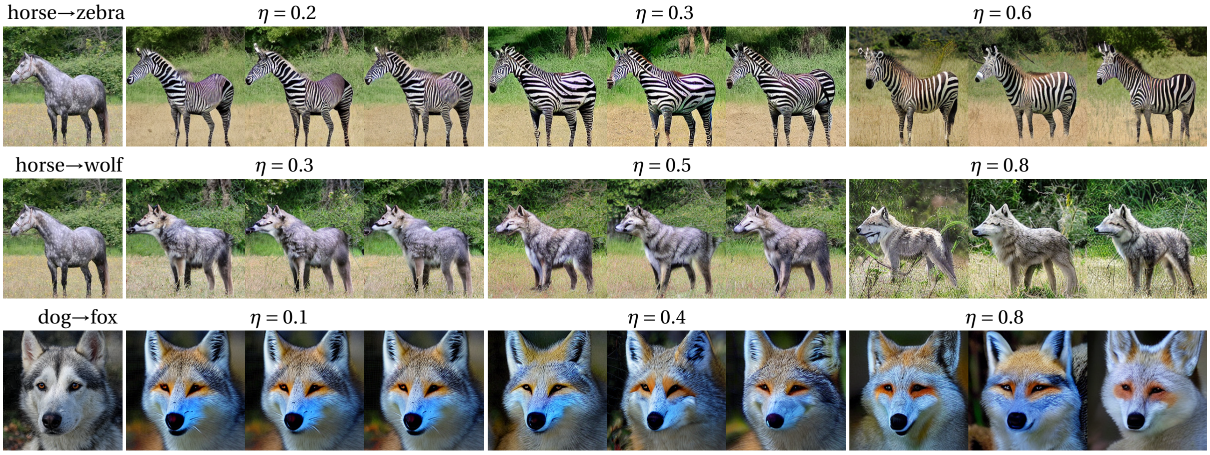


Figure 8.9: Sample results for different η using LDEdit. As the value of η increases, the diversity of samples increases.

In our approach, we proposed to perform a deterministic DDIM sampling, to ensure that consistency is maintained with the original image. However, when the input image lacks details, such as a stroke image, doing a deterministic forward produces a latent code which lacks any details, see Fig. 8.8 a). On the other hand, the introduction of stochasticity through η can aid in hallucinating details not present in the original image, Fig. 8.8 b). With $\eta = 1$, DDIM becomes equivalent to DDPM sampling, which results in more diverse samples. Note that our method may sometimes result in images with text like artifacts, as seen in Fig. 8.8 c). More examples of image manipulation of LDEdit by varying η are shown in Fig. 8.9. As the value of η increases, the diversity of samples improves. However, there are more perceptible changes in the background, see rows 1 and 2 of Fig. 8.9.

Failure Cases

In some cases, our method may fail to produce desired manipulations as seen in Fig. 8.10. With an input text prompt of 'a deer with antlers', we obtain manipulated images where the antlers are misplaced. In other cases, we obtain features of target objects additionally in

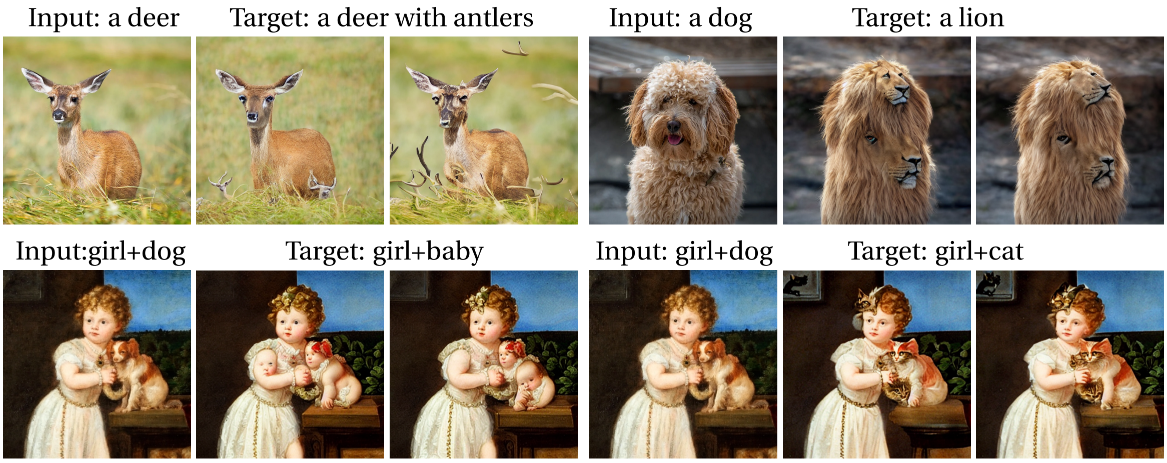


Figure 8.10: Failure cases of image manipulation using LDEdit

undesired locations, such as a baby face on the girl's hand, or a cat face in the hair and in the background picture frame. These undesired effects can be avoided by using a mask, which can aid in the localization of edits.

Editing with Masks

Our method can be modified to include a user-specified mask which specifies the regions where significant changes are needed. Similar mask-guided editing has also been shown in (Avrahami et al., 2022; Nichol et al., 2022). The user-specified mask is also down-sampled such that it has the same spatial extent as the latent code. Let $z_{t_{stop}}$ be the latent code after forward diffusion, the desired localized edit can be obtained by performing the reverse diffusion process on multiple copies of $z_{t_{stop}}$, by changing the target text for the respective masked regions. For seamless blending of the masked and unmasked regions, the latent codes corresponding to the different copies are combined at each diffusion step. This even allows us to specify different levels of stochasticity for the different masked regions. Fig. 8.11 shows the result of such masked editing. We can see that our approach successfully results in a seamless local editing, without requiring expensive optimization.

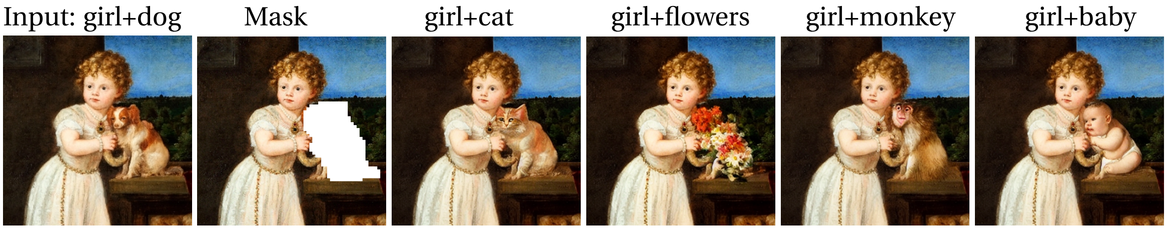


Figure 8.11: Masked image manipulation using LDEdit

User Study

We conduct user studies to compare user preference of image manipulation results of our method with VQGAN+CLIP (Crowson et al., 2022) and DiffusionCLIP (Kim et al., 2022). Users participated in two surveys, where they were provided with a source image, target text description, and the results obtained with LDEdit and base-line method (VQGAN+CLIP or DiffusionCLIP) in a random order, and voted their preferred image manipulation using a survey platform. We obtained a total of 1120 votes from 32 participants for comparing LDEdit with VQGAN+CLIP and 950 votes from 38 participants for comparing LDEdit with DiffusionCLIP. For comparison with both the baselines, we included a combination of face images and general images (on manipulations demonstrated in DiffusionCLIP (Kim et al., 2022) paper). On faces, the manipulated attributes include makeup, tanned, curly hair, changing gender, domain change to zombie, and Neanderthal. We also include an example of translating a stroke image to pixar, neanderthal, and van Gogh painting styles. In general image manipulation, we include manipulating an input building, bus, dog, and a tennis ball. Additionally, for comparison with VQGAN+CLIP, we include examples of manipulating an image of a bird and multiple local object manipulations. In human evaluation, the results of LDEdit were preferred 83.87% of the time in the survey comparing LDEdit with VQGAN+CLIP, whereas user preference for LDEdit is 49.15% in the survey comparing LDEdit with DiffusionCLIP.

Run-time

Tab. 8.2 provides a comparison of GPU memory requirements and run-times of different text based image manipulation methods. The experiments were conducted on a computer with AMD Ryzen 9 3950X 16-Core Processor and NVIDIA GeForce RTX 3090 with 24GB GPU memory. The run-times are highest for VQGAN+CLIP (Crowson et al., 2022) (in the order of minutes), which requires an expensive optimization. Further, VQGAN+CLIP requires a different number of iterations to achieve the desired edit depending on the target prompt, leading to variable run-times. The run-times of both DiffusionCLIP (Kim et al., 2022) and our proposed LDEdit are significantly lower, with LDEdit having smaller run-times due to diffusion in smaller dimensional latent space. It is to be noted that DiffusionCLIP (Kim et al., 2022) needs to be fine-tuned for specific text prompts using a set of images (~ 30-50 images for each prompt), which takes 2 – 6 minutes. Our method also scales well in terms of performing manipulations on multiple images in parallel, in contrast to VQGAN+CLIP, where manipulation on only 2 images could be performed in parallel.

Method	#images	GPU Memory	run-time	(n_{for}, n_{rev})
LDEdit	1	8831MB	2.02s± 5.58 ms	(25,25)
LDEdit	24	16947MB	22.6±169ms	(25,25)
LDEdit	1	8831MB	6.05s±35.6 ms	(75,75)
LDEdit	24	16947MB	67.2s±704ms	(75,75)
VQGAN+CLIP Crowson et al. (2022)	1	10413 MB	4-6 mins	–
VQGAN+CLIP Crowson et al. (2022)	2	18933 MB	5-8 minutes	–
DiffusionCLIP Kim et al. (2022)	1	5385MB	11.54s±66.3ms	(200,40)
DiffusionCLIP Kim et al. (2022)	1	5385MB	4.01s±10.5ms	(40,40)
DiffusionCLIP Kim et al. (2022)	24	15257MB	156.94s±470ms	(200,40)

Table 8.2: Comparing inference times and GPU memory usage of LDEdit with VQGAN+CLIP (Crowson et al., 2022) and DiffusionCLIP (Kim et al., 2022). Images are of dimension 256 × 256. n_{for} and n_{rev} refer to the number of forward and reverse diffusion steps. Mean and standard deviation of run-times over 10 runs are reported for LDEdit and DiffusionCLIP.

8.5 DISCUSSION AND CONCLUSIONS

We proposed LDEdit, a fast and flexible approach to open domain image manipulation using arbitrary text prompts. Our approach utilizes a recent text-to-image latent diffusion model to achieve zero-shot manipulation. Experiments demonstrate that the proposed method can accomplish fast and diverse manipulation making our approach a versatile tool to facilitate efficient user-guided editing. As with other image generation and manipulation methods, there is a potential for LDEdit to be misused by bad actors for generating deep-fakes and doctored pictures for propaganda. Further, since LDEdit leverages a pre-trained text to image latent diffusion model, our approach inherits the inherent biases of its training dataset, including, but not limited to gender, age, and ethnicity of people and cultural biases.

Our work is among the first methods to employ latent diffusion models for general text guided image manipulation. Concurrent works such as (Avrahami et al., 2023) have also proposed to use latent diffusion models for specific editing tasks such as mask guided editing. Following the release of latent diffusion models trained on much larger vision-language datasets, popularly referred to as *Stable Diffusion*, a flurry of text guided image manipulation methods have been proposed. Some of these methods such as (Couairon et al., 2023) obviate the requirement of user-specified mask by automatically estimating the region of edits. A few more recent works (Mokady et al., 2023; Wallace et al., 2023) also perform image manipulation without unwanted changes by improving upon DDIM to affect more precise inversion. Research in text guided generation and vision language models is progressing at a very rapid pace, and we expect more powerful generation and manipulation tools to be available in future.

Declaration for Chapter 9 - Text Guided Explorable Image Restoration

This chapter is based on the rejected submission to ICCV 2023 titled "Text Guided Explorable Image Restoration" co-authored by Kanchana Vaishnavi Gandikota and Paramanand Chandramouli, and subsequent improvements to this submission. A shorter version of this work Gandikota and Chandramouli (2023) was accepted and presented at ICML 2023 Workshop on Artificial Intelligence & Human-Computer Interaction (non-archival). Paramanand Chandramouli and Kanchana Vaishnavi Gandikota are joint first authors of this work.

Kanchana Vaishnavi Gandikota and Paramanand Chandramouli jointly proposed this project idea of text guided image restoration using pretrained text-to-image diffusion models. Paramanand Chandramouli conducted initial experiments for super-resolution from average down-sampling through null-space consistency in a single stage of unCLIP model. Kanchana Vaishnavi Gandikota proposed to impose consistency in both the stages, which allowed restoration tasks in full-resolution, and implemented super-resolution for bicubic downsampling. Paramanand Chandramouli proposed embeddings averaging trick to deal with structurally inconsistent restoration results. Both the authors collaborated in designing the experiments, and contributed equally to generating the results provided in the paper. Kanchana Vaishnavi Gandikota reviewed literature, designed and set up the user study, and contributed to writing all the sections in the first draft of the paper.

9 TEXT GUIDED EXPLORABLE IMAGE RESTORATION

In this chapter, we revisit the problem of image recovery and attempt to provide a solution to deal with the ill-posedness of inverse image reconstruction. Consider the example of recovering a high resolution image from a very low resolution input. This is a highly ill-posed linear inverse problem, and there can be many high resolution images which map to the same low resolution input exactly. As discussed in Chapter 3, solutions to such inverse problems could be obtained both by classical approaches, end-to-end trained deep networks, and a variety of approaches combining deep learning with classical methods. While end-to-end trained deep networks (Fuoli et al., 2021; Chen et al., 2023b; Fang et al., 2022) achieve state-of-the-art performance, most of the prior works still recover only a single arbitrary image out of many possible solutions. On the other hand, stochastic estimators can recover a range of possible solutions that correspond to the degraded input. Such approaches were explored recently in the context of super-resolution (Menon et al., 2020; Bahat and Michaeli, 2020; Lugmayr et al., 2020; Li et al., 2022a), image deblurring (Whang et al., 2022), and jpeg decompression (Bahat and Michaeli, 2021). A few works also propose to explore the solution space, for instance, using graphical user inputs (Bahat and Michaeli, 2020, 2021) or semantic maps (Buhler et al., 2020) for exploring solutions to image restoration, or using a trained classifier for exploring solutions of CT recovery (Dröge et al., 2022). While this is interesting, natural language offers a simpler and more intuitive way to communicate semantic concepts such as age, emotion, color, and other attributes. Therefore, a method which guides image restoration through text can greatly ease the exploration of semantically meaningful solutions.

In this chapter, we propose a zero-shot approach to image restoration using simple and intuitive text prompts. Our goal is to generate reconstructions semantically similar to input text, while preserving data consistency with the degraded input, without explicitly training for specific degradations. Towards this goal, we utilize a recently proposed diffusion based text-to-image (T2I) generative model DALL-E2 (Ramesh et al., 2022) trained for text guided image generation, and adapt it for restoration by modifying the reverse diffusion process. We incorporate the range-null space decomposition (Schwab et al., 2018; Bahat and Michaeli, 2020; Chen and Davies, 2020; Wang et al., 2023b) into the reverse diffusion to analytically enforce data consistency of the solutions, while exploring diverse contents of null-space using text guidance. Due to faster diffusion in the down-sampled space of DALL-E2 and analytic consistency enforcement without expensive back-propagation through network weights, the proposed approach allows efficient exploration of data consistent solutions to image restoration tasks via text prompts, using a few network evaluations. As illustrated in Fig 9.1, this approach can greatly improve the diversity of outputs, while providing semantically meaningful content.

We focus on extreme image super-resolution with large upscale factors, as this problem is severely ill-posed, and allows exploration of a larger solution space. We also demonstrate the applicability of our method on image colorization and inpainting. Further, we extend the text guided image manipulation techniques encountered in chapter 8 using image generative models (Karras et al., 2019; Esser et al., 2021b) with CLIP guidance (Radford et al., 2021) to the task of text guided image restoration, by additionally encouraging data consistency in

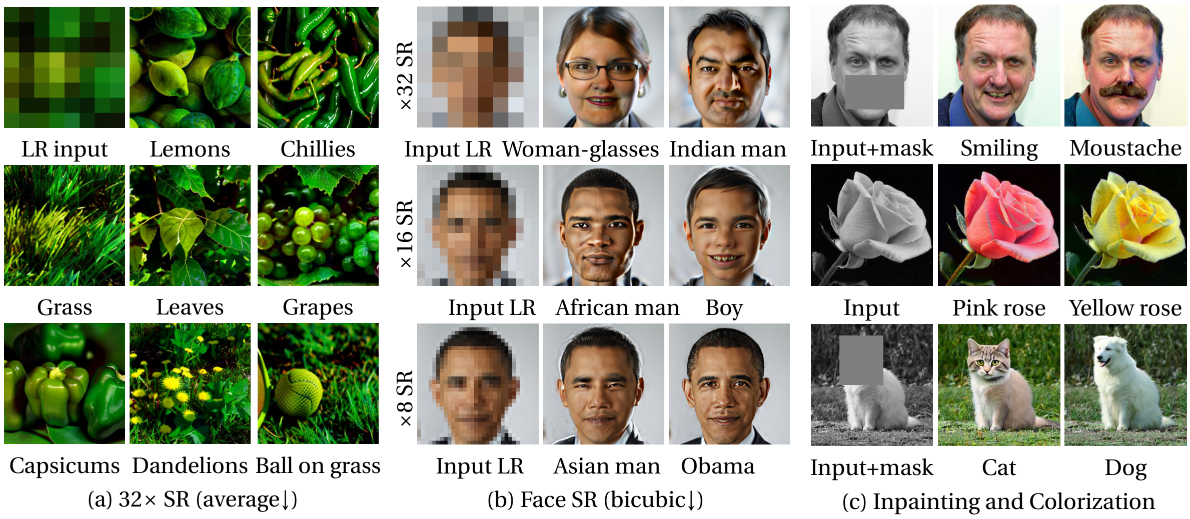


Figure 9.1: **Text guided image restoration.** We explore multiple perfectly consistent reconstructions to image restoration problems through text prompts, while achieving perfect data consistency with the given inputs for all solutions. Shown from left to right are a) extreme super-resolution of natural images, b) face super-resolution c) inpainting and colorization. Prompts of the form *'A photograph of {key word}'* are used. Reconstructions with the corresponding keyword are depicted.

the optimization process. Extensive experimental evaluations demonstrate the benefit of the proposed approach in terms of flexibility, speed, and diversity in generated solutions which enables efficient text guided exploration. Our work opens up a promising direction of developing efficient tools for text guided exploration of solutions to image recovery problems.

9.1 RELATED WORK

We now discuss specific related work on approaches that allow sampling diverse solutions to image recovery, diffusion models in image recovery, and text based image recovery. We refer the reader to Chapters 3 and 4, for a more detailed background on different approaches to image recovery, and the use of generative model priors in inverse reconstruction problems. Further, we refer to Chapter 8 for a background on text-to-image generative models.

DIVERSE SOLUTIONS TO IMAGE RESTORATION Stochastic restoration algorithms can produce variable outputs, which is desirable for ill-posed problems. Recent work (Ohayon et al., 2023) shows that such stochastic algorithms can generate images with high perceptual quality while being more robust than deterministic approaches. Existing deep learning approaches to stochastic image restoration utilize conditional (Bahat and Michaeli, 2020; Lugmayr et al., 2020; Buhler et al., 2020) or unconditional generative models (Menon et al., 2020; Montanaro et al., 2022) such as GANs (Menon et al., 2020; Bahat and Michaeli, 2020; Buhler et al., 2020), normalizing flow based models (Lugmayr et al., 2020; Jo et al., 2021b) and more recently diffusion based models (Kawar et al., 2022a; Chung et al., 2022c; Wang et al., 2023b; Song et al., 2023). Conditional generative model based approaches (Bahat and Michaeli, 2020; Lugmayr et al., 2020; Saharia et al., 2023; Li et al., 2022a; Peng and Li, 2020; Jo et al., 2021a; Saharia et al., 2022b; Whang et al., 2022) are typically *trained* for the specific restoration task, while allowing stochastic sampling to generate diverse reconstructions. A few of these methods also guarantee consistency of the reconstruction with input either by an explicit projection operation (Bahat and Michaeli, 2020), or by exploiting the inherent invertibility of the (flow-based) generative models (Lugmayr et al., 2020; Jo et al., 2021b).

Methods proposed in recent challenge benchmarks (Lugmayr et al., 2021, 2022b) also make use of conditional generative models to learn the solution space of super-resolution.

Among these, a few prior works attempt to explore the solution space using graphical inputs (Bahat and Michaeli, 2020) or semantic maps (Buhler et al., 2020). However, they are still restricted to specific classes e.g. faces, or trained for specific degradation, e.g. specific super-resolution factors. To the best of our knowledge, there is no existing method which allows exploration of solutions space for different restoration tasks on open domain images through text or any other guidance.

DIFFUSION PRIORS FOR IMAGE RESTORATION There are two approaches to using diffusion based models for image recovery tasks. One approach is to train a conditional diffusion model for specific restoration tasks (Saharia et al., 2023; Li et al., 2022a; Saharia et al., 2022b; Whang et al., 2022). Alternatively, one could utilize diffusion based image generative models for the task of image restoration in a zero-shot fashion. Most of such zero-shot approaches assume that the forward degradation operator is known, and exploit this knowledge in a guidance mechanism to modify the sampling process. A few works Chung et al. (2022a); Murata et al. (2023) address blind inverse problems. In the following, we discuss different zero-shot approaches with known degradation models, which is what we consider in this chapter. Jalal et al. (2021a) adopt Langevin dynamics for linear inverse problems and incorporate guidance through the gradient of the least-squares data fidelity term. Kadkhodaie and Simoncelli (2021) propose a score based method to solve linear inverse problems using stochastic gradient ascent to sample from the implicit prior of a trained blind denoiser. Choi et al. (2021) propose an iterative refinement of latent variables for super-resolution which substitutes the low-frequency component in each sampling step with the corresponding information from the measurement. To deal with noise in the intermediate latents, Choi et al. (2021) utilize a low-pass filter to obtain the low frequency components, this, however, does not guarantee exact consistency. Lugmayr et al. (2022a) propose a diffusion based approach for inpainting which replaces unmasked regions in the intermediate steps by corresponding regions in the measurement corrupted by suitable noise strength, to impose data consistency during the sampling process. Chung et al. (2022c) utilize an initial estimate provided by a super-resolution network to reduce the number of reverse diffusion steps, and alternate between a standard reverse diffusion step and a non-expansive mapping to impose data consistency on the intermediate noisy estimate for consistent super-resolution. While this improved performance in comparison with earlier methods, such iterations accumulate errors due to noise in the intermediate steps. This is solved in (Chung et al., 2022b) which guides the reverse process through the gradient of the least-squares data fidelity term using clean prediction at each reverse diffusion step, followed by a non-expansive mapping to impose data consistency. While this provides impressive performance on linear noise-less inverse problems, imposing strict measurement consistency through a projection step is not ideal when measurements are noisy, and imposing data consistency constraints is difficult for non-linear measurements. Therefore, Chung et al. (2023) discard the projection step in (Chung et al., 2022b) to solve general noisy inverse problems. Song et al. (2023) modify the guidance function in (Chung et al., 2023) to incorporate gradient-based guidance from measurements by applying pseudo-inverse operation. While the approaches (Chung et al., 2022b, 2023; Song et al., 2023) result in improved reconstruction quality and consistency, these methods require higher computation times as they require back-propagation through the diffusion model weights at each iteration. Kawar et al. (2021) also consider noisy inverse reconstruction problems whose forward operator can be decomposed by singular value decomposition to perform reverse diffusion in spectral space, which is improved upon in (Kawar et al., 2022a) in terms of speed and performance. This was further extended in (Kawar et al., 2022b) to the task of the noise-less non-linear inverse problem of JPEG artifact correction. Wang et al. (2023b) predict a clean sample at each reverse diffusion step, and perform a projection operation that rectifies the clean prediction at each step to explicitly satisfy data consistency. This method is equivalent to (Kawar et al., 2022a) in the noise-less case. These approaches (Kawar et al., 2022a; Wang et al., 2023b) also achieve high measurement consistency, and reconstruction quality while being fast, as they do not require expensive back-propagation through the weights of a diffusion model. More recently, Mardani et al. (2023) adopt diffusion models in a regularization by denoising (RED) framework, and Zhu et al. (2023) demonstrate their utility for plug-and-play image restoration as an effective alternative to the standard Gaussian denoisers. In this chapter, we modify the approach of Wang et al. (2023b) to perform linear restoration tasks using text-to-image diffusion models performing text conditioned diffusion in down-sampled pixel space. Our choice is motivated by the performance and computational efficiency of (Wang et al., 2023b) which does not require expensive back-propagation through the weights of a diffusion model, in contrast to the competing baselines (Chung et al., 2022b, 2023; Song et al., 2023).

TEXT GUIDED IMAGE RESTORATION A recent work (Ma et al., 2022) also combines text features into super-resolution network architectures using attention and train separate models for text guided image super-resolution in an end-to-end manner for each dataset and super-resolution factor. Further, some recent approaches (Chen et al., 2018; Bahng et al., 2018; Kim et al., 2019) also trained deep networks for colorizing gray-scale images, using text inputs. These methods, however, are trained for specific tasks and datasets and are not generalizable.

9.2 PRELIMINARIES

9.2.1 *Range-Null Space Decomposition:*

Let us consider ill-posed image restoration tasks where the measurement process is modeled as a linear operator. In a noiseless case, this can be represented as

$$f = Au. \tag{9.1}$$

When AA^T is invertible, applying the pseudoinverse $A^\dagger = A^T(AA^T)^{-1}$ produces the minimum norm solution $A^\dagger f$ to the ill-posed problem eq. (9.1), with perfect data consistency. Any other sample of form $(A^\dagger f + u_\delta)$ also satisfies perfect data consistency, as long as u_δ lies in the null space of A. Note that u can be decomposed as:

$$u \equiv A^\dagger Au + (I - A^\dagger A)u. \tag{9.2}$$

We can note that $A^{\dagger}Au$ satisfies exact consistency with $AA^{\dagger}Au \equiv f$, and the component $(I - A^{\dagger}A)u$ is in the null space of A, with $A(I - A^{\dagger}A)u \equiv \mathbf{0}$. Given an approximate solution $\bar{u}$, this decomposition can be used to construct a solution,

$$\hat{u} = A^{\dagger}f + (I - A^{\dagger}A)\bar{u}, \tag{9.3}$$

such that $\hat{u}$ satisfies perfect data consistency (Bahat and Michaeli, 2020; Wang et al., 2023b).

9.2.2 *Denoising Diffusion Null Space Models*

Denoising Diffusion Null Space Models (DDNM) (Wang et al., 2023b) is a recently proposed technique for zero-shot image restoration using pretrained diffusion based image generative models (Ho et al., 2020). The core idea of DDNM is to utilize the range-null space decomposition in the reverse diffusion process.

To maintain consistency in notation with eq. (9.1), we assume the distribution of desired images is $q(u)$. The reverse diffusion process involves Gaussian transitions $p_{\theta}(u_{t-1}|u_t, u_0)$ utilizing a learned noise approximator $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$. We have seen in section 8.2.1, that a clean estimate of a sample can be obtained at any iteration in the reverse diffusion process:

$$u_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(u_t - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(u_t, t)\sqrt{1-\bar{\alpha}_t}\right). \tag{9.4}$$

Wang et al. (2023b) rectify this estimate to impose consistency with the measurement at each step in the reverse process by refining its null-space component using range space null-space decomposition eq. (9.3). The rectified estimate $\hat{u}_{0|t}$ satisfying data consistency is obtained as:

$$\hat{u}_{0|t} = A^{\dagger}f + (I - A^{\dagger}A)u_{0|t}. \tag{9.5}$$

This rectified estimate is used in subsequent sampling from $p(u_{t-1}|u_t, \hat{u}_{0|t})$. The advantage of the DDNM approach is that it can reconstruct consistent solutions, without expensive backpropagation through the generator weights. The limitation of this method is that it can only be used in reconstruction tasks where the pseudo-inverse operator $A^{\dagger}$ can be computed efficiently.

9.2.3 *DALL-E2 unCLIP*

We now describe DALL-E2 (Ramesh et al., 2022) the text-to-image generative model, we employ for text guided restoration. DALL-E2 consists of: *i) a diffusion based prior* to produce CLIP image embeddings (Radford et al., 2021) z_i from encodings of the input prompt c, *ii) a conditional diffusion based decoder* $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ to generate images conditioned on CLIP image embeddings and text prompts, and *iii) a diffusion based super-resolution module* $\boldsymbol{\zeta}_{\boldsymbol{\theta}}$ to obtain a high resolution output. The prior is trained to produce CLIP image embedding given a text caption, and the diffusion decoder is trained to invert the CLIP image encoder. The whole framework is referred to as unCLIP, as it generates images by inverting the embeddings of the CLIP image encoder. The text conditioned diffusion is performed in a down-sampled pixel space for improved computational efficiency, yielding a lower resolution image, which is super-resolved in a subsequent diffusion based super-resolution module to obtain a higher resolution output.

9.3 METHOD

Given a degraded image f with a known degradation operator A, our goal is to generate data consistent solutions $\hat{u}$ whose attributes can be varied using input text prompts c:

$$\begin{aligned} \textit{Data Consistency}: \quad & A\hat{u} \equiv f, \\ \textit{Semantic Consistency}: \quad & \hat{u} \sim q(u|c), \end{aligned} \tag{9.6}$$

where, $q(u|c)$ denotes the distribution of images u with semantic meaning provided by the text prompt c. To obtain data consistent reconstructions satisfying semantic meaning provided by text prompt, we employ null space consistency enforcement proposed in DDNM (Wang et al., 2023b) in the conditional reverse diffusion process of the unCLIP model (Ramesh et al., 2022). We call this approach DDNM-unCLIP. As the text conditioned diffusion is performed in a low resolution pixel space (64×64), followed by a diffusion based super-resolution at a higher resolution (256×256), we adapt DDNM to deal with a two-stage diffusion process to recover images u of high resolution. We first recover a lower resolution version u_{LR} by using a modified measurement A_{LR} which takes into account the downsampling $(\downarrow 4)$ operation for text conditioned diffusion in low resolution. Let z denote the CLIP image embeddings produced by the prior model for input text prompt, and ϵ_θ denotes diffusion based text conditioned decoder, the current estimate of the low resolution clean image at each step is given by:

$$u_{LR_{0|t}} = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(u_{LR_t} - \epsilon_\theta(u_{LR_t}, t|z) \sqrt{1 - \bar{\alpha}_t} \right), \tag{9.7}$$

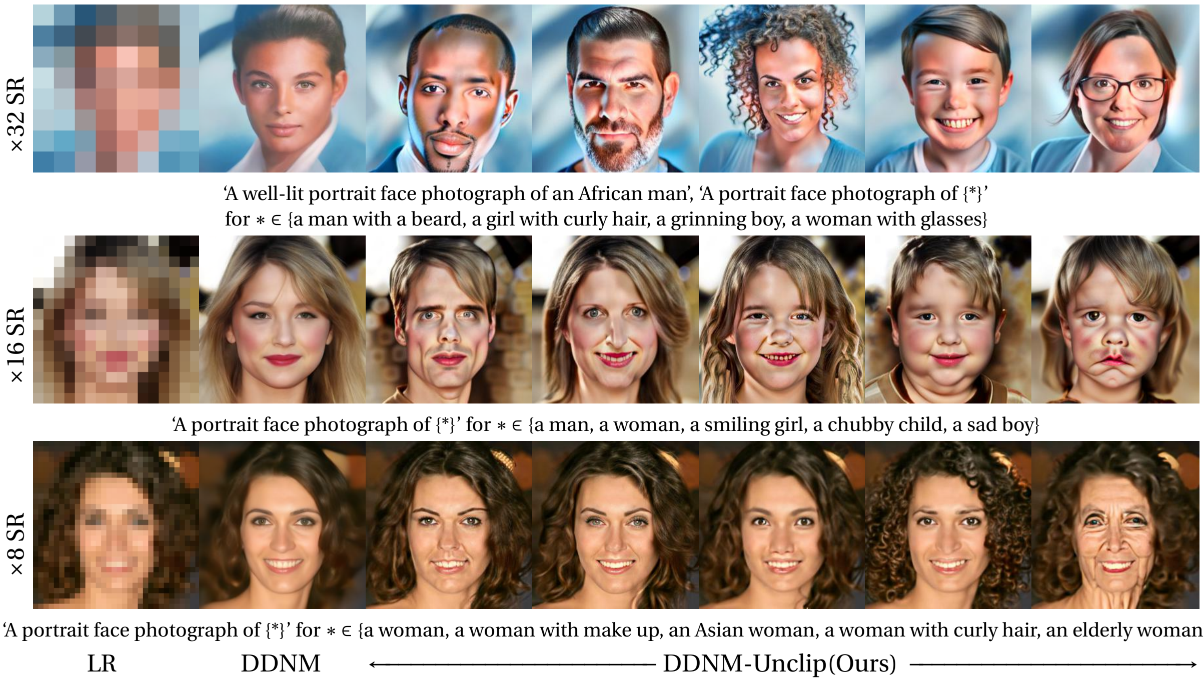
and the consistency rectified estimate following DDNM is given as

$$\hat{u}_{LR_{0|t}} = A^{\dagger}{}_{LR} f + (I - A^{\dagger}{}_{LR} A_{LR}) u_{LR_{0|t}}. \tag{9.8}$$

For the subsequent diffusion for super-resolution using the model ζ_θ, we consider the actual measurement operator A, with null space consistency rectification using DDNM. This two step process is summarized in Algorithm 1. In practice, we accelerate the reconstruction by starting at an earlier time step $t_0 < T$, instead of starting from random noise for both of the reverse diffusion processes and use a fewer number of steps between $[1, t_0]$ in the reverse diffusion. Due to this acceleration, and text conditioned diffusion in the smaller dimensional pixel space, DDNM-unCLIP can produce fast reconstructions. Furthermore, DDNM consistency enforcement in the decoder and super-resolution modules ensures perfect data consistency of the resulting solutions. Fig. 9.2 shows that this approach can generate data consistent reconstructions with the desired semantic attributes.

Algorithm 1 DDNM unCLIP sampling.

```
$u_{LR_T} \sim \mathcal{N}(0, I)$
**for** $t = T, \ldots, 1$ **do**
    $u_{LR_{0|t}} = \frac{1}{\sqrt{\bar{\alpha}_t}} \left( u_{LR_t} - \epsilon_\theta(u_{LR_t}, t|z) \sqrt{1 - \bar{\alpha}_t} \right)$
    $\hat{u}_{LR_{0|t}} = A^{\dagger}{}_{LR} f + (I - A^{\dagger}{}_{LR} A_{LR}) u_{LR_{0|t}}$
    $u_{LR_{t-1}} \sim p_1(u_{LR_{t-1}} | u_{LR_t}, \hat{u}_{LR_{0|t}})$
**end for**
$u_{LR} \leftarrow u_{LR_0}$
$u_T \sim \mathcal{N}(0, I)$
**for** $t = T, \ldots, 1$ **do**
    $u_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}} \left( u_t - \zeta_\theta(u_t, t|u_{LR}) \sqrt{1 - \bar{\alpha}_t} \right)$
    $\hat{u}_{0|t} = A^{\dagger} f + (I - A^{\dagger} A) u_{0|t}$
    $u_{t-1} \sim p(u_{t-1} | u_t, \hat{u}_{0|t})$
**end for**
**return** $u_0$
```



'A well-lit portrait face photograph of an African man', 'A portrait face photograph of {*}' for $* \in$ {a man with a beard, a girl with curly hair, a grinning boy, a woman with glasses}

'A portrait face photograph of {*}' for $* \in$ {a man, a woman, a smiling girl, a chubby child, a sad boy}

'A portrait face photograph of {*}' for $* \in$ {a woman, a woman with make up, an Asian woman, a woman with curly hair, an elderly woman}

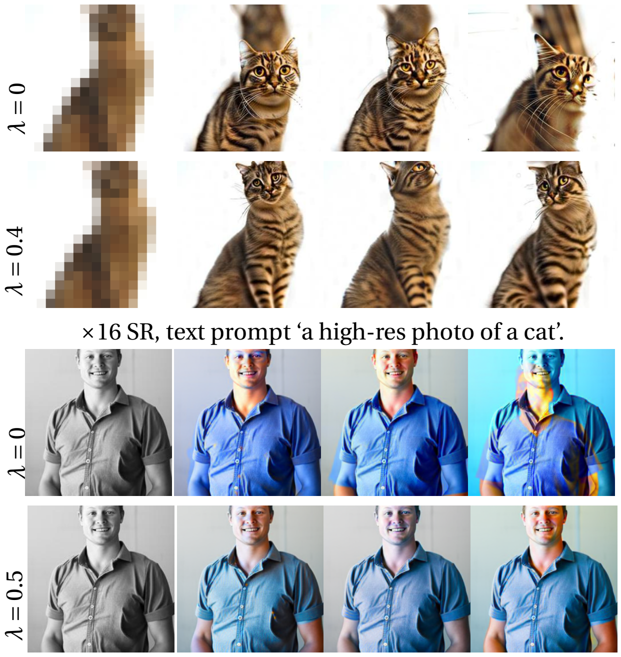
Figure 9.2: Visual comparison of face super-resolution with respect to DDNM Wang et al. (2023b) for scales ×8, ×16 and ×32.

Embeddings Averaging Trick

While DDNM-unCLIP sampling generates reconstructions consistent with both text and measurements, it can still result in unrealistic images when the image embedding z_i as imagined by the prior does not structurally align with the observation. To alleviate this, we propose to modify z_i for better structural consistency with the degraded input:

$$z_i = (1 - \lambda) z_{i_{prior}} + \lambda \mathcal{E}(A^{\dagger} f), \tag{9.9}$$

where, $\mathscr{E}$ is the CLIP image encoder used in training the DALL-E2 unCLIP model. This embedding averaging trick improves the structural consistency of the image embedding with the input observation. We found reasonable outputs with $\lambda \in [0, 0.6]$ with lower values of λ for higher super-resolution factors. We also find this embedding averaging trick useful for zero-shot colorization with text inputs, for improving the structural consistency of the results. For all the image colorization results using DDNM-unCLIP, we use the averaging trick with $\lambda = 0.5$. We can observe in Fig. 9.3 that embedding averaging leads to more realistic looking reconstructions.



×16 SR, text prompt ‘a high-res photo of a cat’.

Colorization, text input ‘a man wearing a blue shirt’

Figure 9.3: Reconstructions with and without the proposed embeddings averaging trick for image super-resolution and colorization with text input.

Implementing A and $A^\dagger$

For the tasks of inpainting and colorization, we consider forward operators of a simple form :

COLORIZATION: A is the operator $\begin{bmatrix}\frac{1}{3} & \frac{1}{3} & \frac{1}{3}\end{bmatrix}$ that averages the intensities in the red, green and blue channels at each pixel $\begin{bmatrix}r & g & b\end{bmatrix}^\top$ into a grayscale value $\left[\frac{r}{3} + \frac{g}{3} + \frac{b}{3}\right]$. The corresponding pseudo-inverse is $A^\dagger = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^\top$.

INPAINTING: A is the binary mask operator, whose pseudo-inverse $A^\dagger \equiv A$.

SUPER-RESOLUTION: In case of down-sampling by averaging with scale n, A becomes the average pooling operator, and corresponding $A^\dagger$ would replicate the pixels n^2 many times. When the low resolution images are generated through bicubic down-sampling, we construct the pseudo-inverse using the singular value decomposition (SVD) following (Kawar et al., 2022a)

$$A = U\Sigma V^T, \qquad A^\dagger = V\Sigma^\dagger U^T.$$

When the forward operator is a composition of multiple simple operators, $A = A_1 \ldots A_n$, the corresponding pseudo-inverse can be obtained as $A^\dagger = A_n^\dagger \ldots A_1^\dagger$. We use such a composition in our experiments where we perform both inpainting and colorization.

9.4 EXPERIMENTS AND RESULTS

We perform our experiments using the publicly available implementation of the unCLIP model (Lee et al., 2022). This model is trained on 115M image-text pairs including COYO-100M, CC3M, and CC12M (Lee et al., 2022) to generate images of resolution 256 × 256. The unCLIP decoder of this model is trained at the resolution 64 × 64, i.e., u_{LR} is of resolution

64×64, and the subsequent super-resolution model at resolution 256×256. We use this model directly for restoration tasks without further fine-tuning. We set t_{stop} as 800 out of the total 1000 steps and use fewer steps between $[1, t_{stop}]$ in both the reverse diffusion process (40 for text conditioned decoding and 10 for super-resolution). For obtaining the image embeddings we perform 25 reverse diffusion steps using the prior model.

We focus on extreme super-resolution of images with large super-resolution factors ×8, ×16, ×32, as this problem is severely under-determined and allows exploration of a larger solution space, and is, therefore, an ideal setting to test our method on exploring diverse solutions. In contrast, input imposes stronger constraints on the solutions for super-resolution at smaller scale factors, limiting their diversity and explorability.

DATASETS AND EVALUATION METRICS: To evaluate consistency between the generated result and the input text prompt, we use CLIP score (Radford et al., 2021) using the ViT-B/16 CLIP model. For super resolutions with large factors, PSNR/SSIM which measure consistency with ground truth are not effective metrics to measure reconstruction performance, as multiple solutions can lead to the same low resolution image. Measuring low resolution consistency by calculating PSNR between the input LR image and the downsampled version of the solution is a better alternative to measure reconstruction performance and has been used in recent challenges for learning super-resolution space (Gu et al., 2022a; Lugmayr et al., 2021). Owing to the use of DDNM based approach, all our solutions satisfy exact consistency, achieving LR PSNR values > 50 dB. We evaluate the reconstruction quality in terms of NIQE score (Mittal et al., 2012), a no-reference image quality estimator to have a quantitative measurement of realism. For the task of super-resolution, we also conduct a user study to evaluate how the users rate the plausibility of reconstruction and semantic consistency with the text prompt.

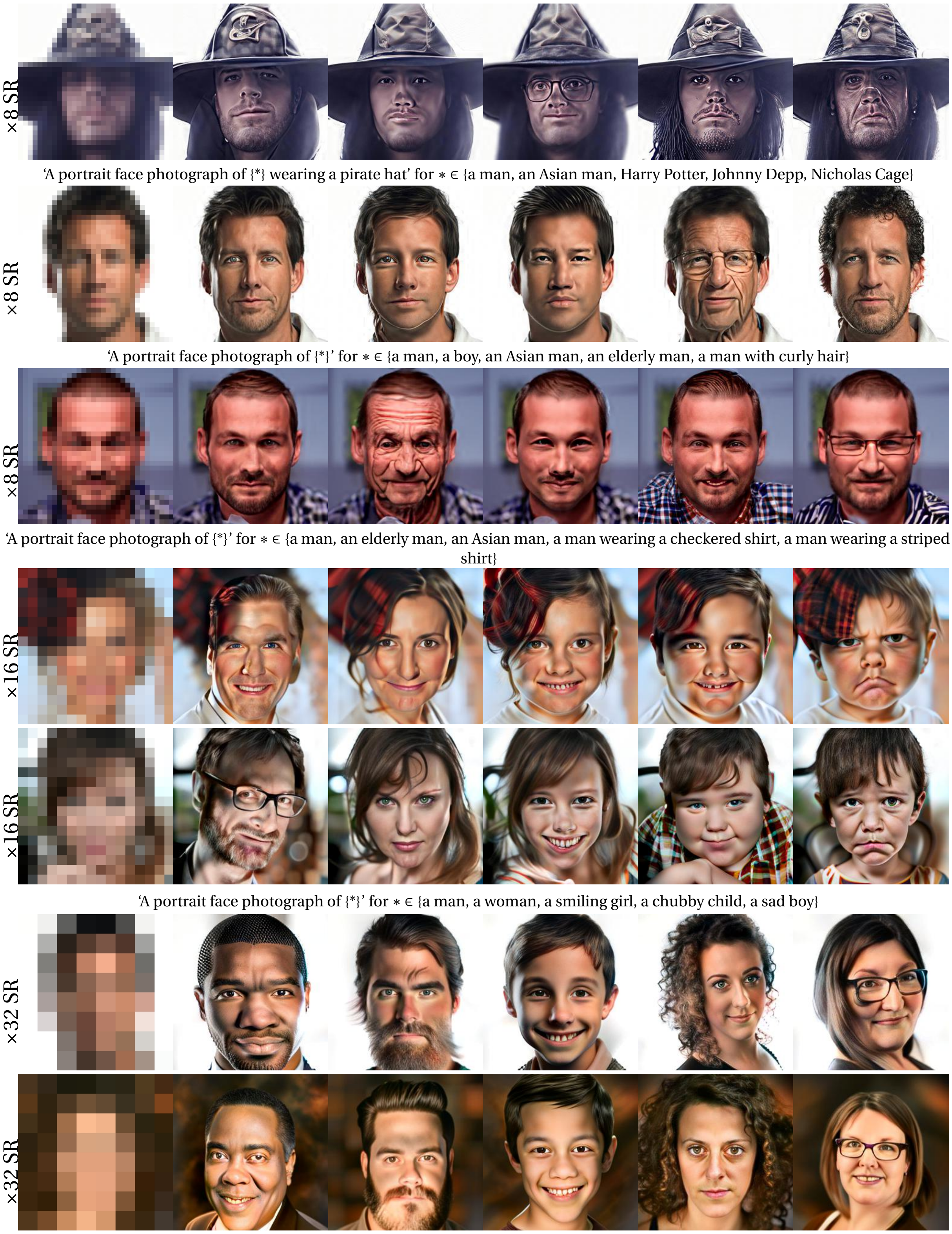
We perform a comparison of DDNM-unCLIP with vanilla DDNM[1] using a diffusion model trained on CelebA-HQ faces (Karras et al., 2018b), on a subset of 100 images from the CelebA-HQ dataset. We provide different captions for different super-resolution factors, as described in Tab. 9.1. We further quantitatively evaluate the super-resolution performance of DDNM-unCLIP on images from the Set5 (Bevilacqua et al., 2012) (5 images), Oxford Pets (Parkhi et al., 2012) (a subset of 200 images). For Set5, we provide captions- a photograph of {a baby face with knitted cap, a side view of a child's face, a black bird perched on a tree has a big colorful beak, a butterfly, a woman face with head band}. For Oxford Pets, we use the captions from the dataset[2] which are automatically generated using BLIP (Li et al., 2022b) and GPT-3(Brown et al., 2020). We show qualitative results for open-domain image super-resolution using images and captions from the SBU Captions (Ordonez et al., 2011) dataset and the nocaps dataset (Agrawal et al., 2019). All our experiments are performed for an output resolution of 256×256. The LR images are obtained via bicubic interpolation.

SR	Caption
8	a portrait face photograph
16	a portrait face photograph of {a man, a woman, a smiling girl, a chubby child, a sad boy} well-lit portrait face photograph of an African man
32	a portrait face photograph of {a man with a beard, a girl with curly hair, a grinning boy, a woman with glasses}

Table 9.1: Text prompts used for quantitative evaluation CelebA-HQ.

[1] https://github.com/wyhuai/DDNM

[2] https://huggingface.co/datasets/Multimodal-Fatima/OxfordPets_test



‘A portrait face photograph of {*} wearing a pirate hat’ for ∗ ∈ {a man, an Asian man, Harry Potter, Johnny Depp, Nicholas Cage}

‘A portrait face photograph of {*}’ for ∗ ∈ {a man, a boy, an Asian man, an elderly man, a man with curly hair}

‘A portrait face photograph of {*}’ for ∗ ∈ {a man, an elderly man, an Asian man, a man wearing a checkered shirt, a man wearing a striped shirt}

‘A portrait face photograph of {*}’ for ∗ ∈ {a man, a woman, a smiling girl, a chubby child, a sad boy}

‘A portrait face photograph of {*}’ for ∗ ∈ {a man of African descent, a man with a beard, a grinning boy, a girl with a curly hair, a woman with glasses}

Figure 9.4: Qualitative results for face super-resolution using DDNM-unCLIP.

FACE IMAGE SUPER-RESOLUTION: Tab. 9.2 provides the results of our quantitative evaluation of images from the CelebA-HQ data set. We compare our model with DDNM using a diffusion model trained on CelebA-HQ faces. Both DDNM and DDNM-unCLIP achieve

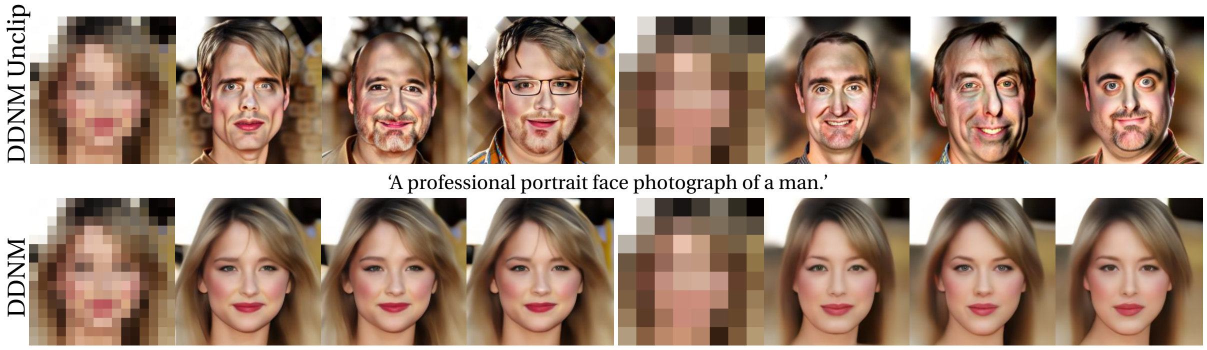


Figure 9.5: Exploring multiple consistent solutions for the same text prompt. Using DDNM with T2I models generates data consistent images of high diversity in content, pose, background and lighting, with semantics meaning matching the input prompt. In contrast, DDNM without any text inputs shows limited variations in the recovered solutions. Results for ×16 SR (left) and ×32 SR (right).

perfect LR PSNR (>50 dB) due to analytic consistency enforcement.

	×8		×16		×32	
	CLIP	NIQE	CLIP	NIQE	CLIP	NIQE
DDNM	0.3330	8.01	0.1979	8.79	0.1989	9.20
DDNM-unCLIP	0.3252	5.41	0.2972	6.43	0.3165	5.97

Table 9.2: Comparing DDNM-unCLIP with DDNM (Wang et al., 2023b) in terms of CLIP score and NIQE score on face image super-resolution.

However, the vanilla DDNM offers no scope of controlling the output using text and therefore, the DDNM generated images do not have high similarity with the captions in terms of CLIP score, except for the neutral caption 'a portrait face photograph of a person' used for ×8 super-resolution task, which is true for all the images in the CelebA-HQ dataset. On the other hand, using text guidance with DDNM-unCLIP results in solutions that are semantically consistent with the input text prompt. This is reflected in the improved CLIP scores using DDNM-unCLIP. Tab. 9.2 also shows a better image quality index for reconstructions obtained using DDNM-unCLIP in terms of NIQE score. While DDNM using the CelebA-HQ diffusion model produces more photo-realistic face images, the poorer performance of vanilla DDNM could be because of blurrier reconstructions.

Figs. 9.2 and 9.4 depict sample qualitative results of our experiments with CelebA-HQ dataset. While the vanilla DDNM achieves perfectly data consistent solutions within the training distributions, it offers little scope for exploration through text. On the other hand, DDNM-unCLIP can recover images with great diversity. As the ill-posedness of the recovery problem becomes more severe at high super-resolution factors (×32), it is possible to recover a wide variety of outputs with challenging attributes in age, race, gender, appearance, and accessories. It is especially hard for generative models trained on datasets with limited attribute variability to overcome their training bias to recover such images. For lower super-resolution factors, the input has stronger constraints on the solutions, and therefore, it is not meaningful to explore drastic variations in semantic attributes. We therefore provide different text prompts with attributes that seem plausible for the given low resolution input. The results in Figs. 9.2 and 9.4 demonstrate that our approach can successfully produce reconstructions with the desired attributes. More qualitative results for face super-resolution are provided in the appendix.

Fig 9.5 compares the reconstructions of DDNM-unCLIP with DDNM using a diffusion model trained on CelebA-HQ faces. Multiple recovered solutions using DDNM-unCLIP are shown for a single text prompt. The results show a wide variety in pose, backgrounds, lighting, and content while being semantically consistent with the input text. In contrast, the DDNM

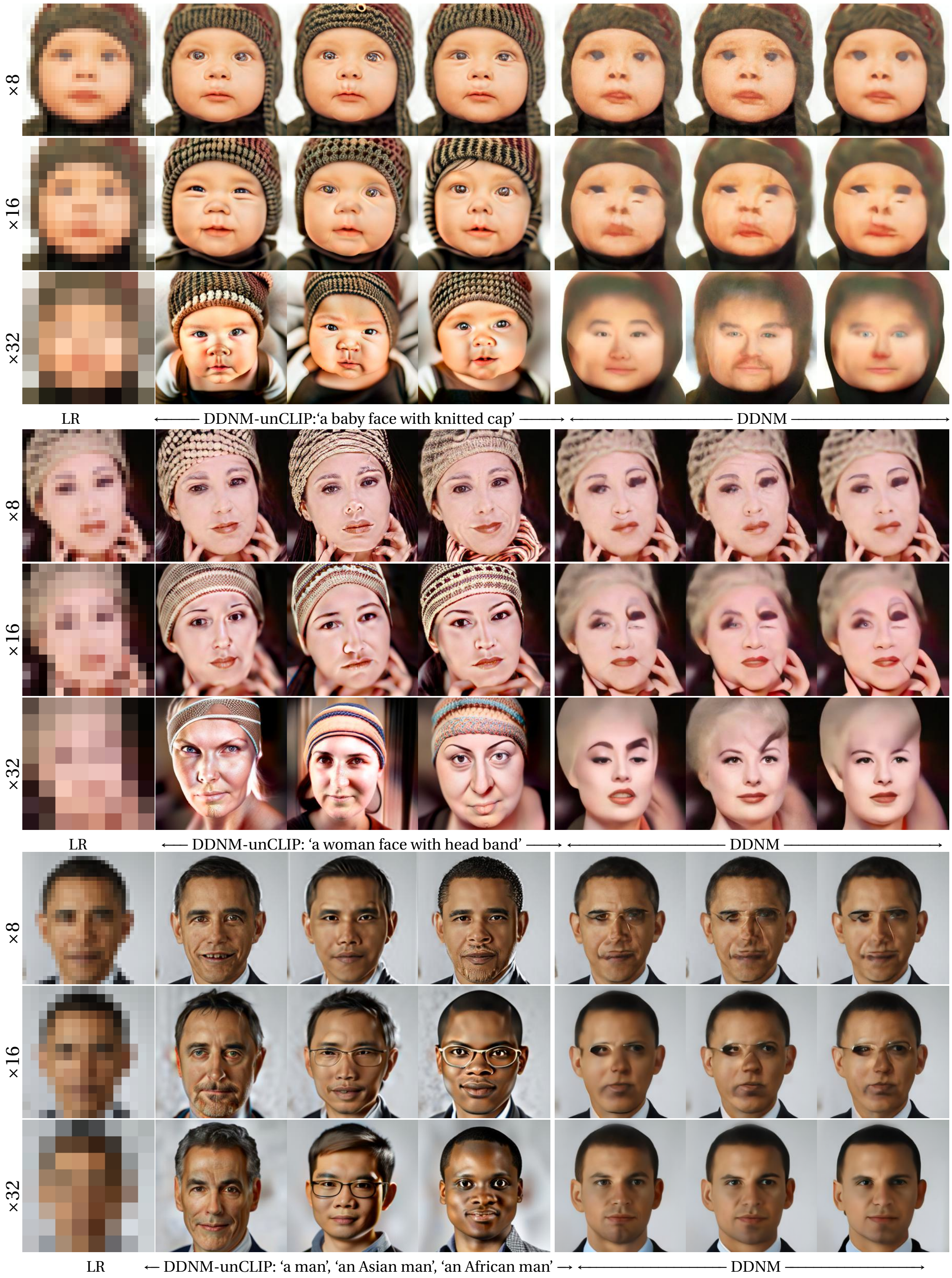

×8
×16
×32
LR
DDNM-unCLIP:‘a baby face with knitted cap’
DDNM
×8
×16
×32
LR
DDNM-unCLIP: ‘a woman face with head band’
DDNM
×8
×16
×32
LR
DDNM-unCLIP: ‘a man’, ‘an Asian man’, ‘an African man’
DDNM

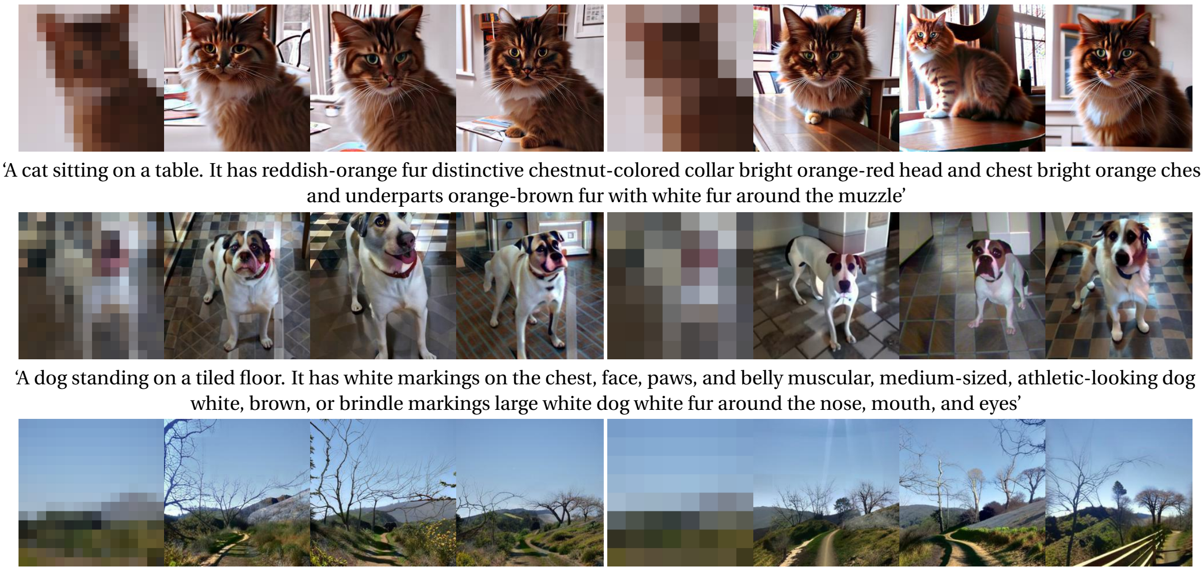
Figure 9.6: Comparing DDNM-unCLIP with DDNM Wang et al. (2023b) on images out of distribution to CelebA

reconstructions show only a small diversity in the recovered solutions, even when starting the algorithm from different noisy latents. This could be due to the inherent bias in the training set, which has limited variations in lighting and pose.

On Bias and Generalization in Reconstruction using Class-Specific Diffusion Models

Intrigued by the limited diversity in the outputs of DDNM using the diffusion model trained on CelebA-HQ dataset, we test DDNM on images which are slightly out of distribution to the CelebA-HQ dataset. We perform super-resolution on downsampled versions of the following test images: two face images from Set5, and an image of Obama with upscaling factors of 8, 16, and 32. The results are illustrated in Fig. 9.6. For the test image of a baby face from Set5, the solutions recovered by vanilla DDNM using diffusion model trained on CelebA-HQ are quite blurred for scales 8 and 16, with results of 16× super-resolution having more artifacts. For 32× super-resolution, DDNM produces blurred faces of adults. As a baby face is out of distribution of CelebA-faces, the recovered solutions are not satisfactory. Even for the face of a woman from Set5, the results using the diffusion model trained on CelebA-HQ are not satisfactory, as this face is not aligned (aligning the face images is a common pre-processing step used in the training generative models on the CelebA-HQ dataset). As a result, the diffusion model tries to hallucinate eyes as though the image is aligned, and the resulting artifacts from the null-space component are visible in the results of 16× super-resolution. The results of ×32 super-resolution are sharper, still, there are artifacts on the head. This shows that a diffusion model trained on a specific domain does not generalize well to images that are slightly out of distribution. We note that these effects are due to the limitations and biases of the trained generative model and its training data, and not the reconstruction algorithm of DDNM. Adapting the same algorithm with unCLIP with reasonable text prompts produces visually better results, and fewer artifacts, as seen in Fig. 9.6.

We further test DDNM and DDNM-unCLIP on super-resolution from the downsampled version of the Obama image. This test image of the downsampled version of Obama resulted in an image of a distinctly white man when upsampled using StyleGAN based reconstruction (Menon et al., 2020), creating a controversy on biases in machine learning algorithms. In this context, Salminen et al. (2020) analyzed the biases in StyleGAN where it is observed that StyleGAN generates predominantly pictures of white people (72.6%), this racial bias is inherited by algorithms using StyleGAN for reconstruction (Menon et al., 2020). Jalal et al. (2021b) proposed an alternate sampling approach to deal with such biases. We test both DDNM and DDNM-unCLIP on this down-sampled test image. The results of the DDNM algorithm contain severe artifacts for scales 8 and 16. Despite running the DDNM algorithm multiple times starting with different noisy latent codes, and with different number of inference steps, we did not obtain results free from such artifacts. The results of DDNM using the CelebA-HQ pretrained diffusion model show improvement at scale 32. However, the results showed only a limited diversity in terms of pose, expression, and perceived race of the reconstructed face. In contrast to the diffusion model trained on CelebA-HQ faces, the results of DDNM-unCLIP demonstrate greater diversity in pose, expression, age, lighting, and background, and the use of text effortlessly enables the reconstruction of faces with varying personal attributes. While this improves demographic diversity through text inputs, one must note that text-to-image models are also not free from biases. For instance, with a race-neutral prompt such as ‘photograph of a face of a man’, DDNM-unCLIP provides images with limited



‘A cat sitting on a table. It has reddish-orange fur distinctive chestnut-colored collar bright orange-red head and chest bright orange chest and underparts orange-brown fur with white fur around the muzzle’

‘A dog standing on a tiled floor. It has white markings on the chest, face, paws, and belly muscular, medium-sized, athletic-looking dog white, brown, or brindle markings large white dog white fur around the nose, mouth, and eyes’

‘not a cloud in the sky (marin trail)’

Figure 9.7: Exploring multiple consistent solutions for the same text prompt for open domain image super-resolution. Results for ×16 SR (left) and ×32 SR (right) using DDNM-unCLIP.

demographic diversity.

OPEN DOMAIN IMAGE SUPER-RESOLUTION

Tab. 9.3 shows the CLIP scores for reconstructions using DDNM-unCLIP for the test images from Set5 and Oxford Pets datasets. The obtained CLIP scores on the reconstructions are close to the CLIP scores achieved by unCLIP on text conditioned generation[3] (0.3081 on conceptual captions and 0.3192 on MS-COCO datasets).

Dataset	×8	×16	×32
Oxford-Pets	0.3180	0.3293	0.3297
Set5	0.3219	0.3349	0.3322

Table 9.3: Evaluating alignment of DDNM-unCLIP reconstructions with text using CLIP score for test images from Oxford-Pets and Set5.

Fig. 9.7 illustrates sample qualitative results of our experiments for test images from Oxford Pets (Parkhi et al., 2012), and SBU Captions (Ordonez et al., 2011) datasets. The text prompts used for recovery are provided along with the images. We observe that even for general open domain images, DDNM-unCLIP is able to recover diverse solutions consistent with the text prompt. More qualitative results on open domain image super-resolution using images and text prompts from Oxford Pets (Parkhi et al., 2012) and nocaps dataset (Agrawal et al., 2019) are provided in the appendix.

User Study

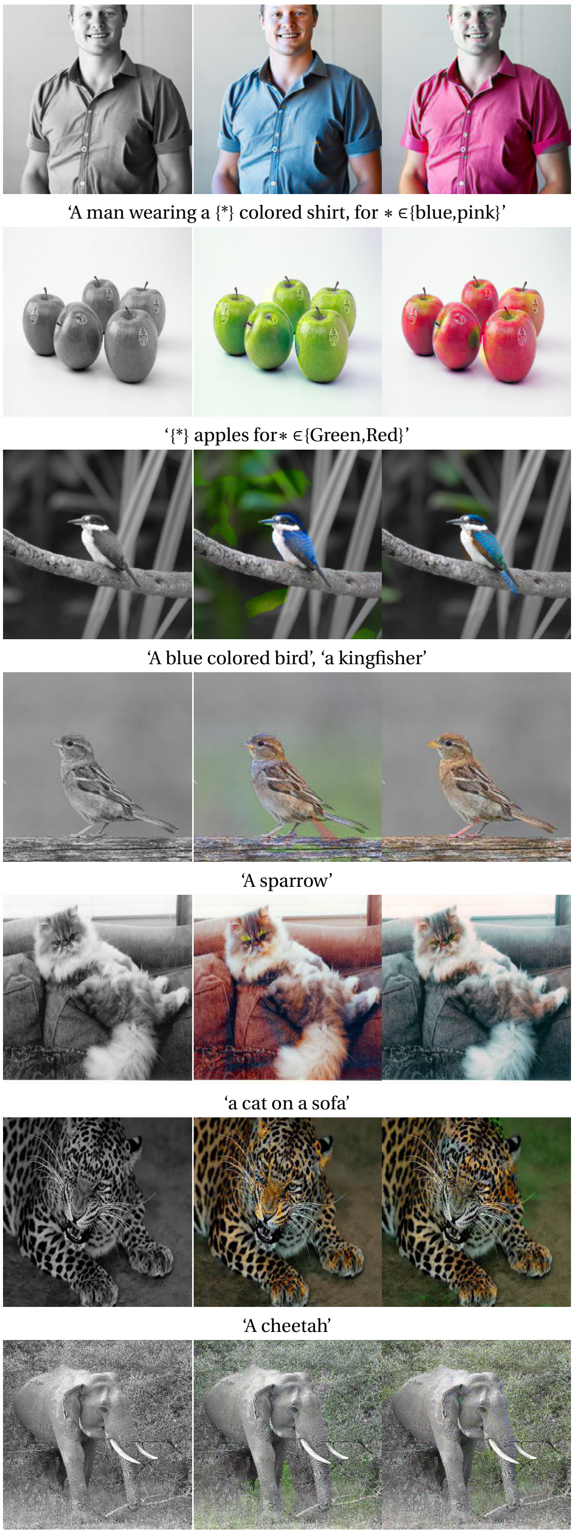
We perform a user study to evaluate the realism and semantic matching with text prompts on our results. The users participated in a survey where they voted on whether or not each super-resolved output looks plausible, and has semantic meaning corresponding to the input text prompt, using a survey platform. This survey included results from the super-resolution of faces, and natural images using DDNM-unCLIP[4].

[3] https://huggingface.co/kakaobrain/karlo-v1-alpha

[4] All the reconstructed images used in the user study were obtained without the embeddings average trick, (i.e. $\lambda = 0.0$), as this idea was developed after conducting the user study.

FACE SUPER-RESOLUTION: This included a user evaluation by 35 participants on the results of ×8, ×16, ×32 super-resolution on 10 different low-resolution images for 5 text prompts, a total of 150 face super-resolution results generated using DDNM-unCLIP. Low resolution images are generated using images from CelebA-HQ (Karras et al., 2018b) downsampled to a resolution 256 × 256. For ×8 face super-resolution, we provide 5 different text prompts suitable for each of the 10 inputs. Some of the examples included in the survey with text prompts and the corresponding super-resolution outputs obtained using DDNM-unCLIP for different super-resolution factors are provided in Figs. 9.2, 9.4, 9.12 and in the Figs. 9.14a and 9.14b of the appendix. Among the different text prompts, we found that the participants had the least preference for the results of ‘an elderly man/woman’ as the results had unnaturally high wrinkles and contained higher ringing artifacts. User rating for plausibility and semantic consistency with text for 8×, 16× and 32× super-resolution was **65.4%**, **63.5%**, and **74.1%** respectively.

OPEN DOMAIN SUPER-RESOLUTION: This included evaluation of DDNM-unCLIP on ×16 super-resolution using low resolution images generated from Oxford Pets (Parkhi et al., 2012), SBU Captions (Ordonez et al., 2011) and nocaps (Agrawal et al., 2019) datasets on 100 reconstructions by 35 participants. Some of the examples used in this evaluation are visualized in Figs. 9.15, 9.17 and 9.10. Participants rated **57.2%** of the images as realistic and semantically consistent.



‘A man wearing a {*} colored shirt, for ∗ ∈{blue,pink}’

‘{*} apples for∗ ∈{Green,Red}’

‘A blue colored bird’, ‘a kingfisher’

‘A sparrow’

‘a cat on a sofa’

‘A cheetah’

‘An Elephant’

Figure 9.8: Qualitative results for colorization.

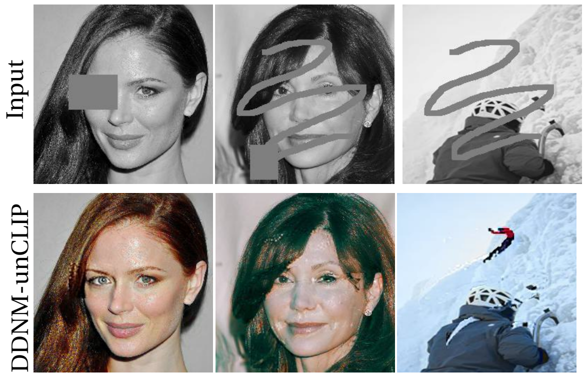


Figure 9.9: Qualitative results on composition of colorization and inpainting for prompts 'a photograph of a woman' for the first two images, 'a mountaineer climbing a snow mountain' for the third image

SR factor	DDNM-unCLIP	DDNM
8×	4.56s±21.4ms	4.77s±21.7ms
16×	4.56s±24.1ms	4.73s±21.1ms
32×	4.55s±18.5 ms	4.74s±21.5 ms

Table 9.4: Timing comparison of DDNM Wang et al. (2023b) with DDNM-unCLIP(Ours)

IMAGE COLORIZATION: Fig. 9.8 demonstrates qualitative examples of colorization using DDNM-unCLIP. The results show consistent colorization according to the text prompts.

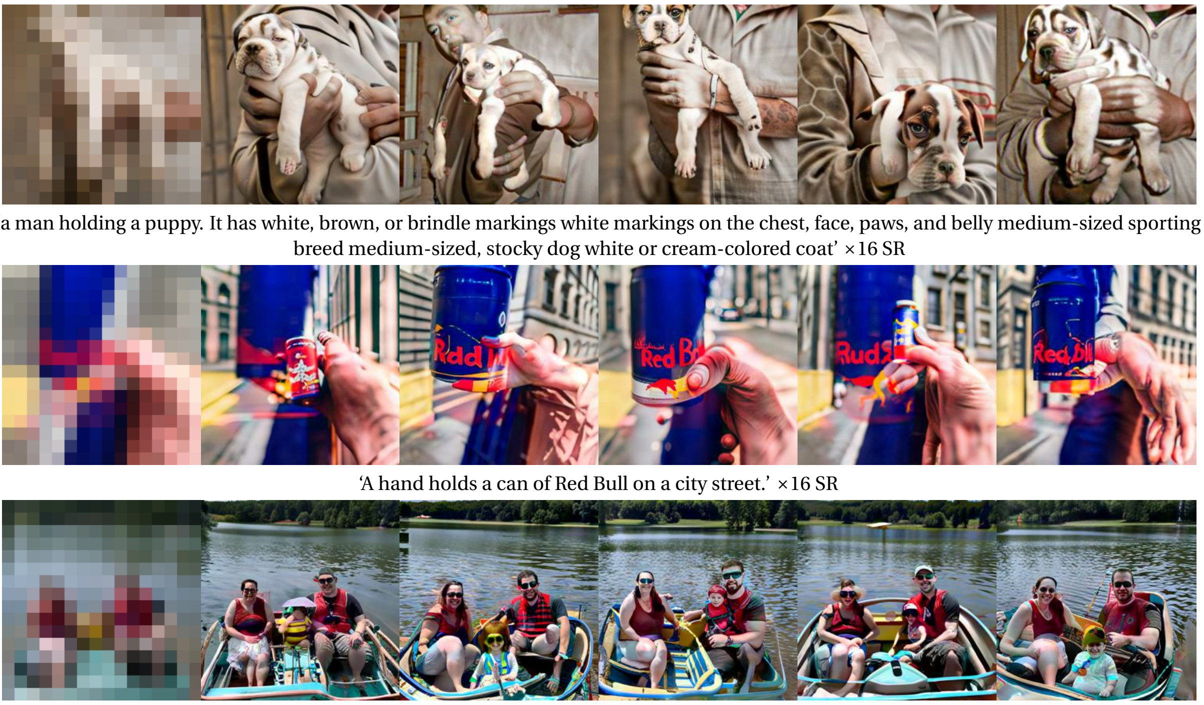
COMPOSITE DEGRADATIONS: Fig. 9.9 provides sample reconstructions for the composition of colorization and inpainting for two face images and an open domain natural image along with the input text prompts. The results show recovered images consistent with the text prompt.

Timing

Tab. 9.4 provides a comparison of run-times between DDNM and DDNM-unCLIP. The experiments were conducted on a computer with AMD Ryzen 93950X 16-Core processor and NVIDIA GeForce RTX 3090 with 24GB GPU memory. DDNM uses 100 reverse diffusion steps, whereas DDNM-unCLIP requires a total of 75 reverse diffusion steps (25 steps with the prior, 40 steps of text guided diffusion, and 10 steps of the super-resolution model). Both approaches have nearly similar run times, as seen in Tab. 9.4.

Failure Cases

Fig. 9.10 demonstrates some failure cases of DDNM-unCLIP, where it failed to generate satisfactory results for each of the 5 random samples generated. We observe that the model struggles with accurately representing holding objects in hands, and produces incorrect human anatomies. These images were also rated uniformly as unrealistic in our user study. These undesired effects are ameliorated to an extent in ×8 super-resolution as it imposes stronger constraints on generated images, see row 3 in Fig. 9.16, which shows better reconstruction of a hand holding an object. Sometimes, the resulting images have weird backgrounds and lighting effects, especially when it is hard to match the input prompt with the degraded image. We can see in Figs. 9.12, 9.11 that our embeddings averaging trick can reduce failure cases and artifacts, and improve photo-realism in difficult scenarios. Further, we observe that the recovered images for certain prompts such as 'an elderly man/woman' contain exaggerated features such as highly wrinkled skin to the point that they no longer look realistic.



'a man holding a puppy. It has white, brown, or brindle markings white markings on the chest, face, paws, and belly medium-sized sporting breed medium-sized, stocky dog white or cream-colored coat' ×16 SR

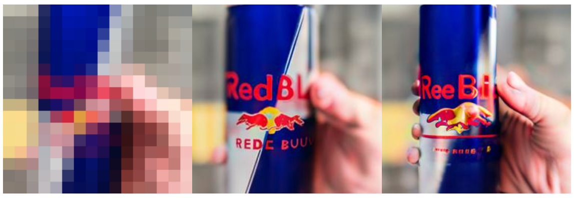
'A hand holds a can of Red Bull on a city street.' ×16 SR

'Mommy, daddy and kid in a paddle boat on the lake' ×16 SR

Figure 9.10: Some failure cases of DDNM-unCLIP.

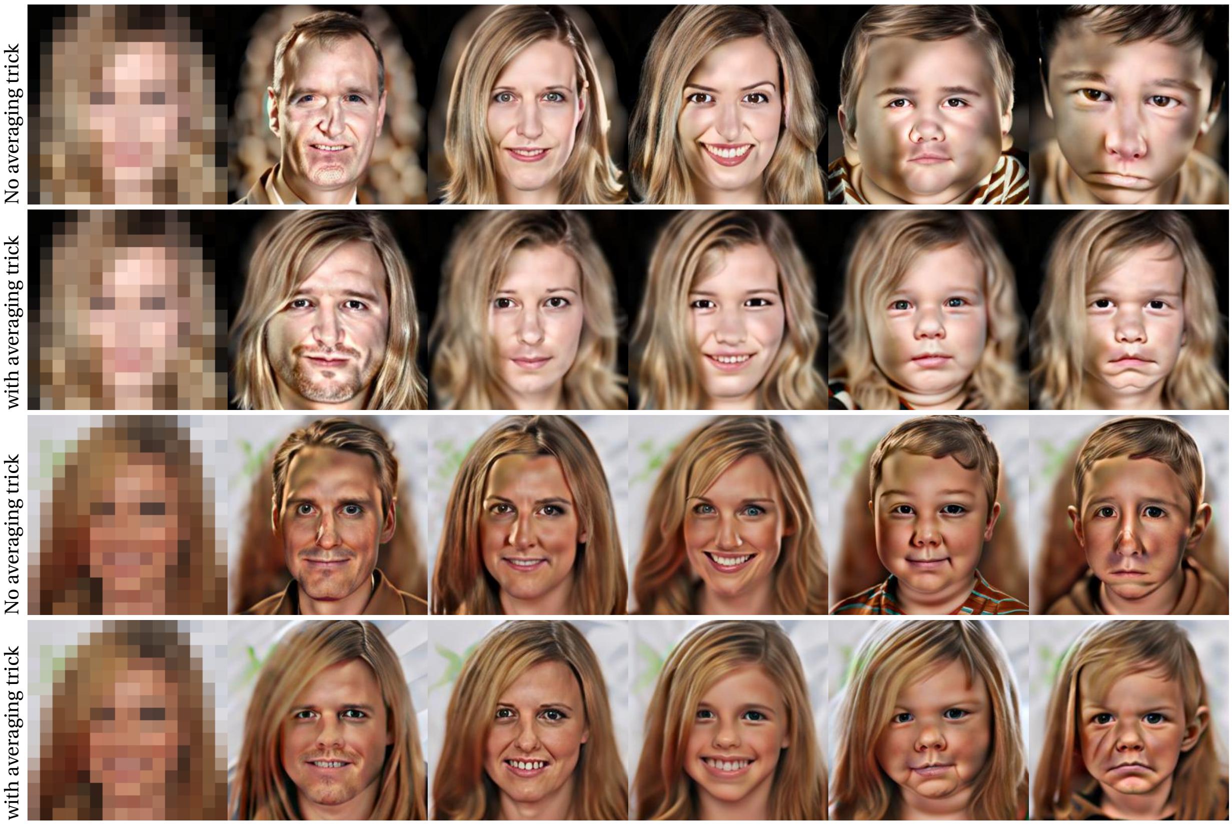
Alternative Methods to Text Guided Restoration

While we proposed to provide a solution to the problem of text guided image restoration using a text-to-image diffusion model, one may also leverage pretrained image generative models for this task. Recall from Chapter 8 the text guided image editing methods of VQGAN+CLIP (Crowson et al., 2022) and StyleCLIP (Patashnik et al., 2021), which optimize in the latent space of a trained GANs (VQGAN (Esser et al., 2021b) and StyleGAN (Karras et al., 2019) respectively) using similarity score provided by CLIP (Radford et al., 2021). We adapt these approaches to text guided restoration using a combined objective of reducing reconstruction loss and increasing CLIP similarity score with the input text. For optimization with VQGAN+CLIP, we follow the approach of Crowson et al. (2022) employing quantized latents with ℓ_2 regularization on the latent vector, and additionally incorporate reconstruction loss. For optimization with StyleGAN+CLIP, we experimented with both latent space optimization and style-space optimization using reconstruction loss and CLIP loss. We found that optimizing in the style-space yields better results. Since these approaches have a trade-off between reconstruction error and alignment with text, the results do not satisfy perfect measurement consistency. We perform preliminary experiments with these two methods, the qualitative results of this experiment are provided in Fig. 9.13. For optimization with VQGAN+CLIP, we obtained an LR PSNR 26-29 dB, for the four examples shown in Fig. 9.13. Further, different examples need varying numbers of steps to converge to a plausible result. The results in the bottom row of Fig. 9.13 for the text prompts 'a man' and 'an elderly



'A hand holds a can of Red Bull on a city street.' ×16 SR

Figure 9.11: Improved results with embeddings averaging for general image SR.



'A portrait face photograph of {*}' for $*\in$ {a man, a woman, a smiling girl, a chubby child, a sad boy}

Figure 9.12: Embeddings averaging trick improves photorealism in difficult examples ×16 face super-resolution.

man' were obtained in 1600 iterations requiring less than 4 minutes. The reconstructions in the top row corresponding to 'a woman' and 'a young girl' took 5200 and 4600 iterations requiring about 13 minutes and 11 minutes respectively. For style space optimization with StyleGAN, we obtain LR PSNR of around 27-30 dB for plausible reconstructions. The results were obtained using 250 steps, requiring around 2 minutes per image.

Apart from DALL-E2 unCLIP, we also studied the applicability of DDNM to *Stable Diffusion* (Rombach et al., 2022), another popular text-to-image diffusion model for the task of super-resolution. However since the diffusion process happens in the latent space, it is not straightforward to impose null-space consistency on the intermediate estimates in the Stable Diffusion model. We show in the appendix that this does not lead to desirable solutions.

9.5 DISCUSSION AND CONCLUSIONS

In this chapter, we introduced text guided exploration of solutions to image restoration problems. We proposed a zero-shot approach that utilizes a pretrained text-to-image diffusion based generative model to yield solutions that are simultaneously consistent with the input text as well as the degraded observation. Our approach can achieve a significantly higher diversity in recovered solutions in comparison with a method using class-specific generative models. The performance of this method depends on and is limited by the generative capabilities of the pretrained generative model, in our case, DALL-E2 unCLIP. The method inherits the biases of the data used to train the unCLIP model. This is also reflected in the results which sometimes lack photorealism, and have an oil painting like effect. This is likely an

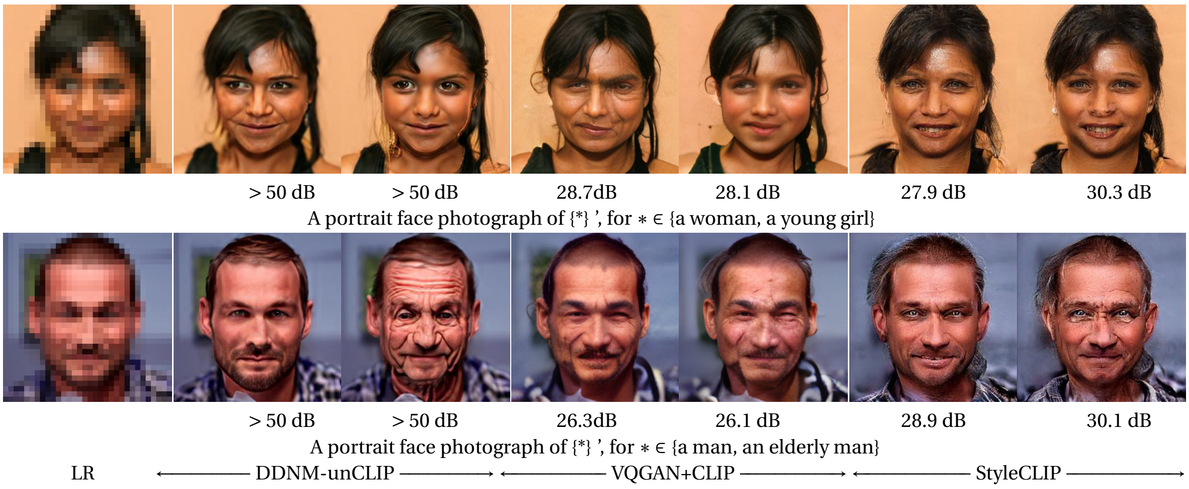

> 50 dB
> 50 dB
28.7dB
28.1 dB
27.9 dB
30.3 dB
A portrait face photograph of {*} ’, for ∗ ∈ {a woman, a young girl}
> 50 dB
> 50 dB
26.3dB
26.1 dB
28.9 dB
30.1 dB
A portrait face photograph of {*} ’, for ∗ ∈ {a man, an elderly man}
LR
DDNM-unCLIP
VQGAN+CLIP
StyleCLIP

Figure 9.13: Comparing DDNM-unCLIP with with VQGAN+CLIP and StyleGAN+CLIP adapted to the task of restoration.

artefact of training data which included paintings and other art in addition to photographs in the dataset. In contrast, a model trained only on photographs of faces can produce more photo-realistic faces, yet it can severely lack generalization to out-of-distribution data.

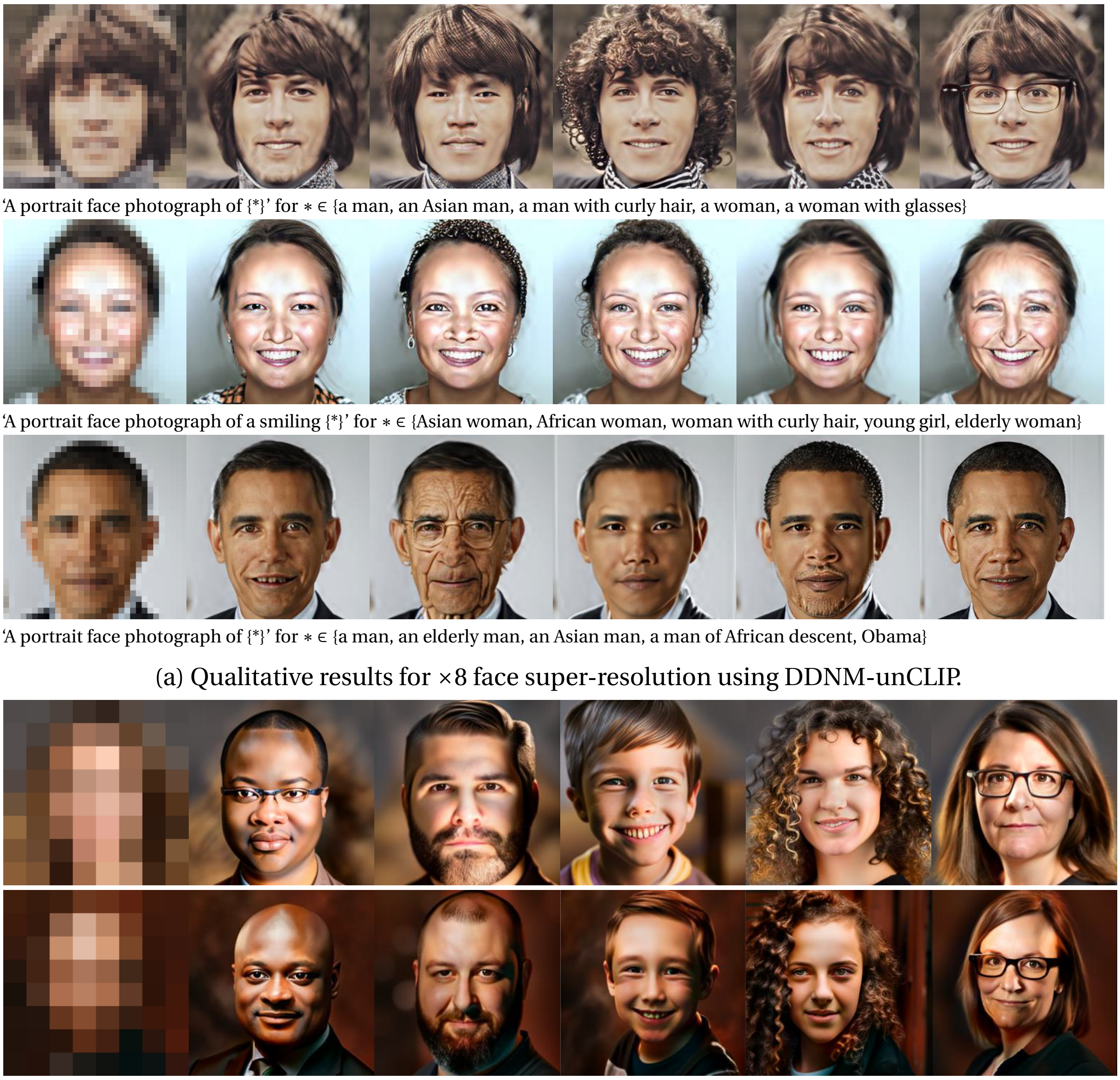
We observed that DDNM-unCLIP sometimes produces perceptually implausible solutions, with large artifacts. When this is caused by a mismatch of the image embedding and the measurement, it can be fixed to an extent by our proposed embedding averaging. On the other hand, not every text prompt is meaningful for every observation. When certain patterns or objects indicated by the input text cannot be present in the image, the corresponding objects or patterns cannot be recovered without severe artifacts or unrealistic images. In this case, it is not the failure of the approach or the model, rather it can help the users determine the plausibility of a solution. In view of this, evaluation of text guided restoration is highly subjective, and any quantitative evaluation in terms of image quality metrics is only meaningful only when the input text prompts are well aligned and plausible for a given degraded measurement. A separate subjective human ranking on alignment with text and plausibility would better evaluate the performance.

While we utilized unCLIP for text guided restoration, it is also interesting to explore text guided restoration with other pixel-space text-to-image diffusion models such as Imagen (Saharia et al., 2022c), when the open source versions of these models become available. It is also interesting to devise algorithms which can be applied for text guided reconstruction using latent diffusion models (Rombach et al., 2022), for example, by backpropagating through the weights of the generator. Alternately, one could also attempt to utilize CLIP to guide pretrained image diffusion models towards target text similar to (Bansal et al., 2023), and additionally impose measurement consistency at each diffusion step. Since we make use of range-null space decomposition, our proposed method requires the degradation operator and its pseudo-inverse in its inference, which may not be available in general restoration tasks. It is interesting to extend text guided disambiguation to non-linear and blind restoration tasks in the future.

APPENDIX

9.A ADDITIONAL QUALITATIVE RESULTS

SUPER-RESOLUTION For face super-resolution, we provide additional qualitative results for scales ×8, ×32 in Figs. 9.14a and 9.14b respectively. For natural image super-resolution, we provide additional qualitative results using images from Oxford Pets in Fig. 9.15 and images and captions from nocaps dataset Agrawal et al. (2019) for the task of ×8 and ×16 super resolution in Figs. 9.16, 9.17.



'A portrait face photograph of {*}' for $* \in$ {a man, an Asian man, a man with curly hair, a woman, a woman with glasses}

'A portrait face photograph of a smiling {*}' for $* \in$ {Asian woman, African woman, woman with curly hair, young girl, elderly woman}

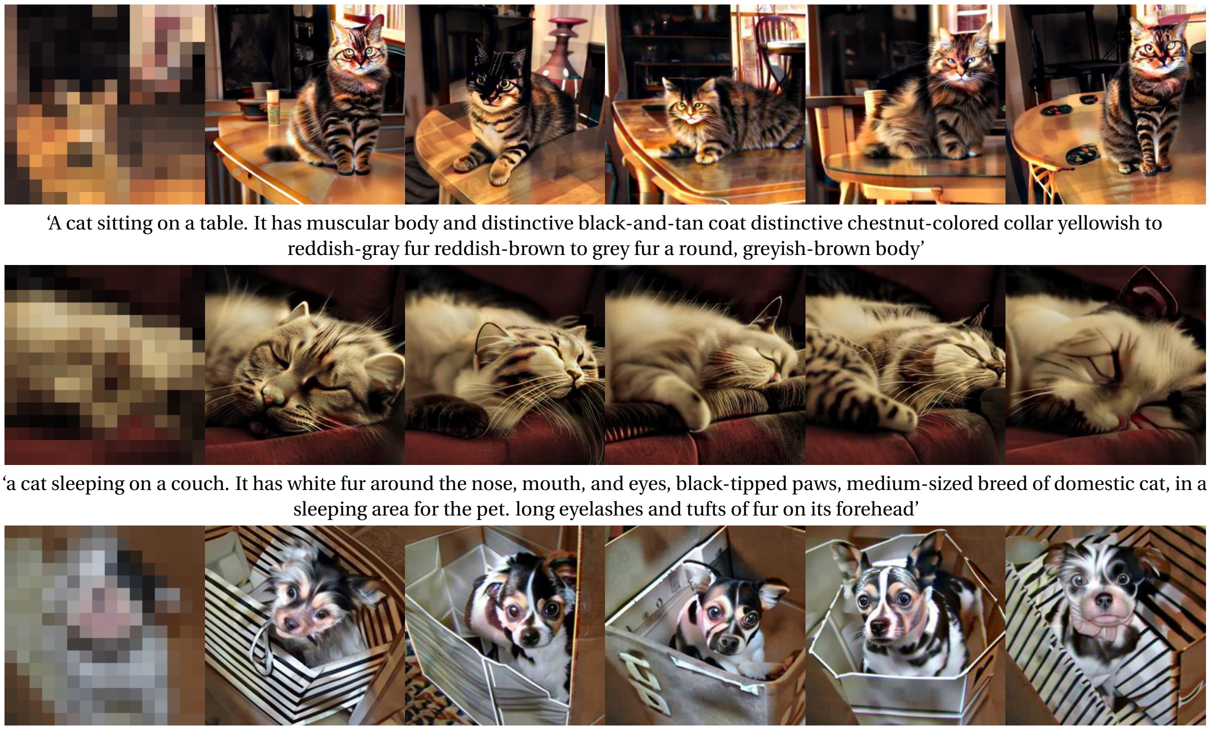
'A portrait face photograph of {*}' for $* \in$ {a man, an elderly man, an Asian man, a man of African descent, Obama}

(a) Qualitative results for ×8 face super-resolution using DDNM-unCLIP.

'A portrait face photograph of {*}' for $* \in$ {a man of African descent, a man with a beard, a grinning boy, a girl with a curly hair, a woman with glasses}

(b) Qualitative results for ×32 face super-resolution using DDNM-unCLIP.

Figure 9.14: Qualitative results for face super-resolution using DDNM-unCLIP.



‘A cat sitting on a table. It has muscular body and distinctive black-and-tan coat distinctive chestnut-colored collar yellowish to reddish-gray fur reddish-brown to grey fur a round, greyish-brown body’

‘a cat sleeping on a couch. It has white fur around the nose, mouth, and eyes, black-tipped paws, medium-sized breed of domestic cat, in a sleeping area for the pet. long eyelashes and tufts of fur on its forehead’

‘a small dog in a box. It has white, brown, or brindle markings black facial markings Black facial markings black, white, gray, fawn, or brindle color dark facial markings’

Figure 9.15: Exploring multiple consistent solutions for the same text prompt ×16 SR on samples from Oxford Pets dataset. The text prompt used for guidance is provided.

9.B CHOICE OF T2I MODEL

While we used DALL-E2 unCLIP in our experiments, we also studied the applicability of *Stable diffusion* Rombach et al. (2022) for text-guided image super-resolution using DDNM. In the case of Stable diffusion, the diffusion process happens in the latent space of a variational auto-encoder, and it is not straightforward to adapt DDNM to this model.

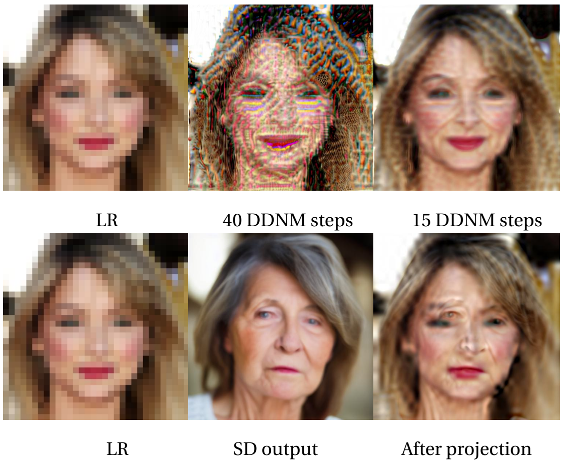
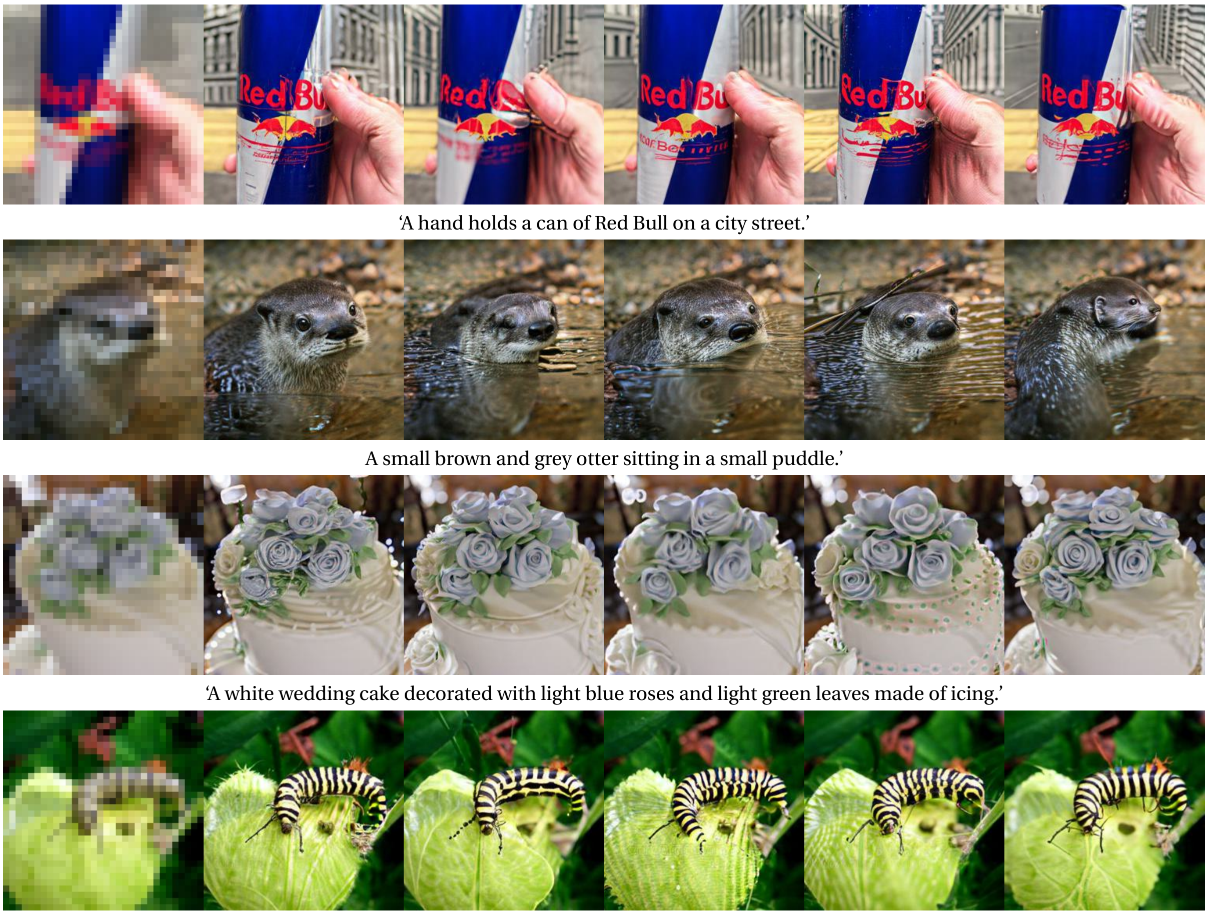


Figure 9.18: (Top) Stable Diffusion with DDNM using VAE decoder & encoder. (Bottom) Stable Diffusion result without DDNM in reverse diffusion. Results for the text prompt ‘a portrait face photograph of an elderly woman’

We attempted to enforce DDNM consistency in the text conditioned Stable diffusion model. At each step in the reverse diffusion process, we estimate the clean latent variable z_0 and decode it to image space and enforce data consistency. This data consistent image is then encoded again to latent space to resume reverse diffusion. While this approach achieves data consistency, we find that it is highly unstable due to the lossy nature of the variational autoencoder. As the number of inference steps increases, it results in unrealistic images with heavy artifacts. When the number of inference steps is low, the artifacts reduce, however, the resulting images are blurry, see the top row of Fig. 9.18. On the other hand, we observed that using the interpolated low resolution image as an initial estimate in the diffusion process



‘A hand holds a can of Red Bull on a city street.’

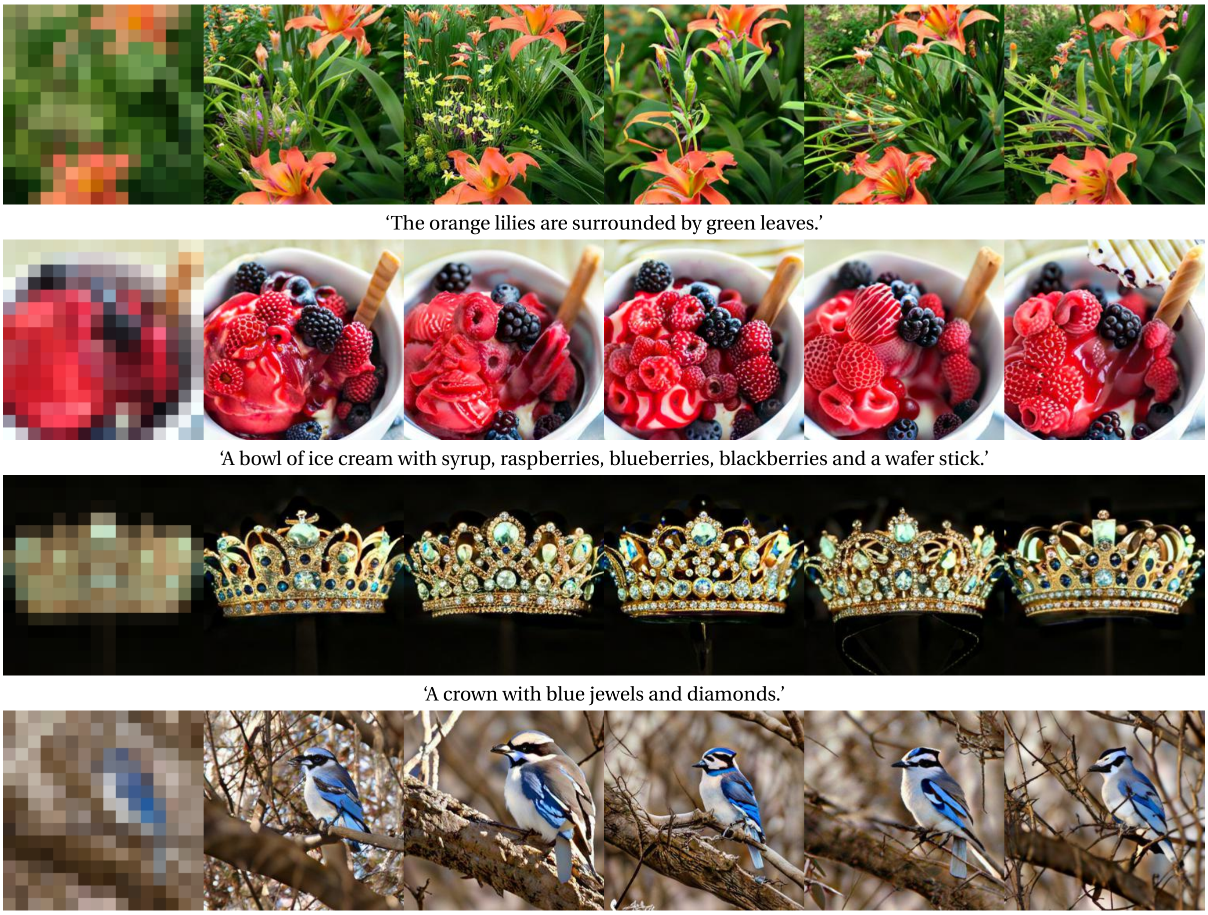
A small brown and grey otter sitting in a small puddle.’

‘A white wedding cake decorated with light blue roses and light green leaves made of icing.’

‘A striped caterpillar on a green plant.’

Figure 9.16: Exploring multiple consistent solutions for the same text prompt ×8 SR on samples from nocaps dataset. The text prompt used for guidance is provided. 5 randomly generated samples are shown

can lead to a totally different image without any guidance or consistency enforcement in the intermediate steps. While the output of the Stable Diffusion model in this case is well-aligned with the text prompt, it is not consistent with the measurement. As a result, the corresponding null-space contents are not aligned with the pseudo-inverse solution. An example is illustrated in the bottom row of Fig. 9.18, where the null space projection adds high frequency details of the elderly woman onto the pseudo inverse reconstruction. Adapting the Stable diffusion model for restoration may require a different guidance mechanism which back-propagates the reconstruction loss through the decoder. In contrast, using the unCLIP model allows us to perform DDNM for text guided reverse diffusion in the (down-sampled) pixel space, which easily generates data-consistent images that are aligned with text.



‘The orange lilies are surrounded by green leaves.’

‘A bowl of ice cream with syrup, raspberries, blueberries, blackberries and a wafer stick.’

‘A crown with blue jewels and diamonds.’

‘A fat blue jay sitting on a broken limb of a leafless tree.’

Figure 9.17: Exploring multiple consistent solutions for the same text prompt ×16 SR on samples from nocaps dataset. The text prompt used for guidance is provided. 5 randomly generated samples are shown

Part IV

CONCLUSIONS

10 CONCLUSIONS AND FUTURE WORK

10.1 SUMMARY

The topic of robustness and generalization in deep learning has recently received a lot of attention from the research community, focusing on different aspects including robustness to adversarial examples, distribution shifts, and spurious correlations among others. This dissertation analyzed and addressed some specific aspects of robustness and generalization for deep learning methods in computer vision:

- We studied the robustness and invariance of deep learning based classifiers to geometric transformations in Chapter 5. This was motivated by the real world necessity of machine learning models to be robust to label-preserving spatial transformations. We developed a simple solution to integrate invariance into arbitrary networks under any infinite group actions by selecting a provably unique element from the orbit. This task-independent strategy can be extended to obtain equivariance by applying the corresponding inverse transformation to the network output. We demonstrated the practical application of this approach for achieving rotation invariant image classification and scale and orientation invariance in point cloud classification.

- We showed in Chapter 6 that deep models trained for image recovery are susceptible to adversarial examples, even producing drastically different target images. We found that such extreme vulnerabilities are more common in blind restoration networks which have to cope with varying degradation operators across examples. On the other hand, we found that reconstruction networks trained for a specific forward operator are relatively more robust in terms of measurement consistency, even under adversarial attacks. We devised a localized attack to modify the visual appearance of clinically relevant regions while maintaining high consistency with the original measurement. Such attacks can be used to explore semantically or diagnostically different solutions in the solution space and can aid in dealing with uncertainties in ill-posed image recovery. We also demonstrated the feasibility and transferability of universal attacks across image recovery methods.

- We developed a flexible solution to light field recovery from measurements obtained through different forward operators using conditional generative priors in Chapter 7. Although end-to-end trained deep network solutions exist for each of these settings, they are affected by even small changes to the forward model. While the use of generative priors offers the desired flexibility, extending such an approach to light field recovery is challenging, due to the inherent difficulty in training a generative model for high dimensional light fields for varying scene content. We handled these challenges by training a generative model on light field patches conditioned on the central view and devised a recovery method to perform optimization of both the latent code and the central view. We demonstrated the advantages of this approach in comparison with end-to-end trained networks for light field recovery in terms of flexibility and robustness to corruptions, and improved performance with respect to traditional model-based approaches.

- We developed a simple method for generalized image manipulation through text prompts leveraging a text-to-image latent diffusion model in Chapter 8. Our method utilized deterministic forward and reverse processes with target text conditioning the reverse process, which automatically resulted in a near cycle-consistency between the source image and the manipulation result, while modifying the desired attributes. We showed that introducing controlled stochasticity into this sampling process aids the manipulation process when the target text is highly different from input, and showed that this method is also amenable to editing with additional mask inputs. We demonstrated advantages in terms of speed and flexibility, in comparison with previous image manipulation methods using text inputs.

- We introduced the problem of exploring solutions to open domain image restoration through text prompts in Chapter 9. We developed a zero-shot approach that utilized a pretrained text-to-image diffusion model by modifying its reverse diffusion process to analytically enforce consistency of the solutions with measurements. We introduced an embeddings average trick which can improve the plausibility of solutions. We showed that this approach can recover diverse reconstructions which preserve data consistency with the degraded inputs while being semantically consistent with input text. In contrast, class-specific generative priors do not yield satisfactory reconstructions, when the image is even slightly out of distribution to the training set.

10.2 DISCUSSION AND FUTURE DIRECTIONS

In summary, this thesis makes some contributions towards improving the robustness and generalization of deep learning methods for image recovery and classification. We provide a few suggestions for future work below.

While we proposed a solution to integrate invariances into arbitrary deep networks under infinite group actions in Chapter 5 through analytic canonicalization of orientation, more recent work Kaba et al. (2023) proposes to learn the canonicalization. In the future, it would also be interesting to extend learned canonicalization to achieve local invariances of objects in an image, as it is difficult to obtain analytical solutions to the same.

Our analysis of the adversarial robustness of image recovery methods in Chapter 6 revealed that different image recovery methods have different degrees of robustness. This motivates us to build more stable approaches to image recovery. Our follow-up work Agnihotri et al. (2023) shows that image recovery networks achieving similar performance for clean inputs can have widely different degrees of robustness with adversarial training, depending on the architectural components used. One interesting direction is to explore in more detail how the architecture design choices affect the performance and robustness of image recovery networks, and design recovery networks that are robust and highly performant through architecture search. As we have observed in Chapter 6, reconstruction networks trained for a specific forward operator are relatively more robust in terms of measurement consistency. Yet, these reconstructions are affected by severe artifacts under perturbations.

In this context, it is important to develop suitable regularization to restrict deep network solutions to satisfy certain desired properties while maintaining measurement consistency.

We have also seen in Chapters 6 and 9 that reconstructions with different semantic or diagnostic meanings can satisfy measurement consistency equally well. In this context, it is important to develop methods that can produce reconstructions with a measure of uncertainty quantification which can be useful in safety critical applications, instead of providing single point estimates. While we devised localized attacks in Chapter 6, which can serve to explore diagnostically different solutions, we focused only on the reconstruction of 2D slices. We look forward to extending the solution space exploration to 3D CT volume reconstruction where the adjacent slices can impose stronger constraints on reconstruction. While this may be achieved through similar localized attacks, it can also be achieved by guiding reconstructions through pre-trained generative models which naturally allow sampling the solution space using a suitable guiding function. While such methods have been explored in the context of image restoration Buhler et al. (2020) and in Chapter 9, we look forward to extending these to other recovery tasks.

We have seen that image recovery using generative priors offers advantages over end-to-end trained approaches in terms of flexible image recovery from a variety of forward measurement processes, as well as the ability to sample a multitude of solutions. Yet, even such methods have difficulty dealing with samples that are less represented in training distribution, and the samples that are even slightly out of distribution to the training data. While we observe improved recovery with text-to-image models trained on much larger datasets, these models are also not free from the biases of their datasets. While balancing the training data is one way to fix this problem, it is also necessary to devise improved training and sampling mechanisms to obtain high fidelity samples even for attributes that are less represented in training data.

(2018). Heidelberg collaboratory for image processing: 4d light field dataset. http://hci-lightfield.iwr.uni-heidelberg.de/.

Abdal, R., Qin, Y., and Wonka, P. (2020). Image2stylegan++: How to edit the embedded images? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8296–8305.

Abdal, R., Zhu, P., Femiani, J., Mitra, N. J., and Wonka, P. (2022). Clip2stylegan: Unsupervised extraction of stylegan edit directions. In *ACM SIGGRAPH*.

Adler, J. and Öktem, O. (2017). Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007.

Adler, J. and Öktem, O. (2018). Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 37(6):1322–1332.

Aggarwal, H. K., Mani, M. P., and Jacob, M. (2018). Modl: Model-based deep learning architecture for inverse problems. *IEEE Transactions on Medical Imaging*, 38(2):394–405.

Agnihotri, S., Gandikota, K. V., Grabinski, J., Chandramouli, P., and Keuper, M. (2023). On the unreasonable vulnerability of transformers for image restoration - and an easy fix. In *IEEE/CVF International Conference on Computer Vision Workshops*, pages 3707–3717.

Agnihotri, S. and Keuper, M. (2023). Cospgd: a unified white-box adversarial attack for pixel-wise prediction tasks. *ArXiv*, abs/2302.02213.

Agrawal, H., Desai, K., Wang, Y., Chen, X., Jain, R., Johnson, M., Batra, D., Parikh, D., Lee, S., and Anderson, P. (2019). Nocaps: Novel object captioning at scale. In *IEEE/CVF International Conference on Computer Vision*, pages 8948–8957.

Al-Shabi, M., Lan, B. L., Chan, W. Y., Ng, K.-H., and Tan, M. (2019). Lung nodule classification using deep local–global networks. *International journal of computer assisted radiology and surgery*, 14(10):1815–1819.

Alaluf, Y., Patashnik, O., and Cohen-Or, D. (2021). Restyle: A residual-based stylegan encoder via iterative refinement. In *IEEE/CVF International Conference on Computer Vision*, pages 6711–6720.

Alaluf, Y., Tov, O., Mokady, R., Gal, R., and Bermano, A. (2022). Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18511–18521.

Alayrac, J.-B., Uesato, J., Huang, P.-S., Fawzi, A., Stanforth, R., and Kohli, P. (2019). Are labels required for improving adversarial robustness? *Advances in Neural Information Processing Systems*, 32.

Almahairi, A., Rajeshwar, S., Sordoni, A., Bachman, P., and Courville, A. (2018). Augmented cyclegan: Learning many-to-many mappings from unpaired data. In *International Conference on Machine Learning*, pages 195–204. PMLR.

Alperovich, A., Johannsen, O., Strecke, M., and Goldluecke, B. (2018). Light field intrinsics with a deep encoder-decoder network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9145–9154.

Amos, B., Xu, L., and Kolter, J. Z. (2017). Input convex neural networks. In *International Conference on Machine Learning*, pages 146–155. PMLR.

Andriushchenko, M. and Flammarion, N. (2020). Understanding and improving fast adversarial training. *Advances in Neural Information Processing Systems*, 33:16048–16059.

Anil, C., Lucas, J., and Grosse, R. (2019). Sorting out Lipschitz function approximation. In *36th International Conference on Machine Learning*, volume 97, pages 291–301. PMLR.

Anselmi, F., Leibo, J. Z., Rosasco, L., Mutch, J., Tacchetti, A., and Poggio, T. (2016). Unsupervised learning of invariant representations. *Theoretical Computer Science*, 633:112–121.

Antun, V., Renna, F., Poon, C., Adcock, B., and Hansen, A. C. (2020). On instabilities of deep learning in image reconstruction and the potential costs of ai. *National Academy of Sciences*, 117(48):30088–30095.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

Armato III, S. G., McLennan, G., Bidaut, L., McNitt-Gray, M. F., Meyer, C. R., Reeves, A. P., Zhao, B., Aberle, D. R., Henschke, C. I., Hoffman, E. A., et al. (2011). The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931.

Arnab, A., Miksik, O., and Torr, P. H. S. (2018). On the robustness of semantic segmentation models to adversarial attacks. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Ashok, A. and Neifeld, M. A. (2010). Compressive light field imaging. In *Three-Dimensional Imaging, Visualization, and Display 2010 and Display Technologies and Applications for Defense, Security, and Avionics IV*, volume 7690, page 76900Q. International Society for Optics and Photonics.

Asim, M., Daniels, M., Leong, O., Ahmed, A., and Hand, P. (2020a). Invertible generative models for inverse problems: mitigating representation error and dataset bias. In *International Conference on Machine Learning*, pages 399–409. PMLR.

Asim, M., Shamshad, F., and Ahmed, A. (2020b). Blind image deconvolution using deep generative priors. *IEEE Transactions on Computational Imaging*, 6.

Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *35th International Conference on Machine Learning*, volume 80, pages 274–283. PMLR.

Avrahami, O., Fried, O., and Lischinski, D. (2023). Blended latent diffusion. *ACM Transactions on Graphics (TOG)*, 42(4):1–11.

Avrahami, O., Lischinski, D., and Fried, O. (2022). Blended diffusion for text-driven editing of natural images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218.

Babacan, S. D., Ansorge, R., Luessi, M., Mataran, P. R., Molina, R., and Katsaggelos, A. K. (2012). Compressive light field sensing. *IEEE Transactions on Image Processing*, 21(12):4746–4757.

Baguer, D. O., Leuschner, J., and Schmidt, M. (2020). Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Problems*, 36(9):094004.

Bahat, Y. and Michaeli, T. (2020). Explorable super resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2716–2725.

Bahat, Y. and Michaeli, T. (2021). What's in the image? explorable decoding of compressed images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2908–2917.

Bahng, H., Yoo, S., Cho, W., Park, D. K., Wu, Z., Ma, X., and Choo, J. (2018). Coloring with words: Guiding image colorization through text-based palette generation. In *European Conference on Computer Vision*.

Bai, S., Kolter, J. Z., and Koltun, V. (2019). Deep equilibrium models. *Advances in Neural Information Processing Systems*, 32.

Bai, Y., Mei, J., Yuille, A. L., and Xie, C. (2021). Are transformers more robust than cnns? *Advances in Neural Information Processing Systems*, 34:26831–26843.

Balunovic, M., Baader, M., Singh, G., Gehr, T., and Vechev, M. (2019). Certifying geometric robustness of neural networks. *Advances in Neural information processing systems*, 32.

Bansal, A., Chu, H.-M., Schwarzschild, A., Sengupta, S., Goldblum, M., Geiping, J., and Goldstein, T. (2023). Universal guidance for diffusion models. *arXiv preprint arXiv:2302.07121*.

Bar-Tal, O., Ofri-Amar, D., Fridman, R., Katen, Y., and Dekel, T. (2022). Text2live: Text-driven layered image and video editing. In *European Conference on Computer Vision*.

Bau, D., Andonian, A., Cui, A., Park, Y., Jahanian, A., Oliva, A., and Torralba, A. (2021). Paint by word. *arXiv preprint arXiv:2103.10951*.

Bau, D., Strobelt, H., Peebles, W., Wulff, J., Zhou, B., Zhu, J.-Y., and Torralba, A. (2019a). Semantic photo manipulation with a generative image prior. *ACM Trans. Graph.*, 38(4):59:1–59:11.

Bau, D., Zhu, J.-Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., and Torralba, A. (2019b). Seeing what a gan cannot generate. In *IEEE/CVF International Conference on Computer Vision*, pages 4502–4511.

Bauermeister, H., Burger, M., and Moeller, M. (2020). Learning spectral regularizations for linear inverse problems. In *NeurIPS 2020 Workshop on Deep Learning and Inverse Problems*.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal of Imaging Sciences*, 2:183–202.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust optimization*, volume 28. Princeton university press.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

Benning, M. and Burger, M. (2018). Modern regularization methods for inverse problems. *Acta Numerica*, 27:1–111.

Benton, G. W., Finzi, M., Izmailov, P., and Wilson, A. G. (2020). Learning invariances in neural networks from training data. In *Advances in Neural information processing systems*.

Bernstein, J., Wang, Y.-X., Azizzadenesheli, K., and Anandkumar, A. (2018). signsgd: Compressed optimisation for non-convex problems. *arXiv preprint arXiv:1802.04434*.

Bertocchi, C., Chouzenoux, E., Corbineau, M.-C., Pesquet, J.-C., and Prato, M. (2020). Deep unfolding of a proximal interior point method for image restoration. *Inverse Problems*, 36(3):034005.

Bevilacqua, M., Roumy, A., Guillemot, C., and Alberi-Morel, M. L. (2012). Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *British Machine Vision Conference (BMVC)*. BMVA press.

Bigdeli, S. A., Zwicker, M., Favaro, P., and Jin, M. (2017). Deep mean-shift priors for image restoration. *Advances in Neural Information Processing Systems*, 30.

Bioucas-Dias, J. M. (2006). Bayesian wavelet-based image deconvolution: A gem algorithm exploiting a class of heavy-tailed priors. *IEEE Transactions on Image Processing*, 15(4):937–951.

Bioucas-Dias, J. M., Figueiredo, M. A., and Oliveira, J. P. (2006). Total variation-based image deconvolution: a majorization-minimization approach. In *2006IEEEInternational Conference on Acoustics Speech and Signal Processing Proceedings*, volume 2, pages II–II. IEEE.

Blau, Y. and Michaeli, T. (2018). The perception-distortion tradeoff. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237.

Blocker, C. J. and Fessler, J. A. (2019). Blind unitary transform learning for inverse problems in light-field imaging. In *IEEE International Conference on Computer Vision Workshops*, pages 0–0.

Bohra, P., Pham, T.-a., Dong, J., and Unser, M. (2022). Bayesian inversion for nonlinear imaging models using deep generative priors. *IEEE Transactions on Computational Imaging*, 8:1237–1249.

Bolte, J., Sabach, S., Teboulle, M., and Vaisbourd, Y. (2018). First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151.

Bond-Taylor, S., Hessey, P., Sasaki, H., Breckon, T. P., and Willcocks, C. G. (2022). Unleashing transformers: parallel token prediction with discrete absorbing diffusion for fast high-resolution image generation from vector-quantized codes. In *European Conference on Computer Vision*, pages 170–188. Springer.

Bora, A., Jalal, A., Price, E., and Dimakis, A. G. (2017). Compressed sensing using generative models. In *34th International Conference on Machine Learning-Volume 70*, pages 537–546. JMLR. org.

Bostan, E., Heckel, R., Chen, M., Kellman, M., and Waller, L. (2020). Deep phase decoder: self-calibrating phase microscopy with an untrained deep neural network. *Optica*, 7(6):559–562.

Bouman, C. and Sauer, K. (1993). A generalized gaussian image model for edge-preserving map estimation. *IEEE Transactions on Image Processing*, 2(3):296–310.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Publishers.

Brendel, W., Rauber, J., Kümmerer, M., Ustyuzhaninov, I., and Bethge, M. (2019). Accurate, reliable and fast robustness evaluation. *Advances in neural information processing systems*, 32.

Brock, A., De, S., Smith, S. L., and Simonyan, K. (2021). High-performance large-scale image recognition without normalization. *arXiv preprint arXiv:2102.06171*.

Brock, A., Donahue, J., and Simonyan, K. (2019). Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Brock, A., Lim, T., Ritchie, J., and Weston, N. (2017). Neural photo editing with introspective adversarial networks. In *International Conference on Learning Representations*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Brown, T. B., Mané, D., Roy, A., Abadi, M., and Gilmer, J. (2017). Adversarial patch. In *Machine Learning and Computer Security Workshop, Neural Information Processing Systems*.

Bruckstein, A. M., Donoho, D. L., and Elad, M. (2009). From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM review*, 51(1):34–81.

Bruna, J. and Mallat, S. (2013). Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886.

Bryniarski, O., Hingun, N., Pachuca, P., Wang, V., and Carlini, N. (2022). Evading adversarial example detection defenses with orthogonal projected gradient descent. In *International Conference on Learning Representations*.

Buades, A., Coll, B., and Morel, J.-M. (2005). A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 60–65 vol. 2.

Buhler, M. C., Romero, A., and Timofte, R. (2020). Deepsee: Deep disentangled semantic explorative extreme super-resolution. In *Asian Conference on Computer Vision*.

Burger, H. C., Schuler, C. J., and Harmeling, S. (2012). Image denoising: Can plain neural networks compete with BM3D? In *IEEE Int. Conf. Computer Vision and Pattern Recognition*, pages 2392–2399.

Burger, M., Resmerita, E., and He, L. (2007). Error estimation for bregman iterations and inverse scale space methods in image restoration. *Computing*, 81(2):109–135.

Cai, Z., Tang, J., Mukherjee, S., Li, J., Schönlieb, C. B., and Zhang, X. (2023). Nf-ula: Langevin monte carlo with normalizing flow prior for imaging inverse problems. *arXiv preprint arXiv:2304.08342*.

Carlini, N. and Wagner, D. (2017a). Adversarial examples are not easily detected: Bypassing ten detection methods. In *10th ACM workshop on artificial intelligence and security*, pages 3–14.

Carlini, N. and Wagner, D. (2017b). Towards evaluating the robustness of neural networks. In *2017IEEEsymposium on security and privacy (sp)*, pages 39–57. IEEE.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. (2019). Unlabeled data improves adversarial robustness. *Advances in neural information processing systems*, 32.

Carrara, F., Becarelli, R., Caldelli, R., Falchi, F., and Amato, G. (2018). Adversarial examples detection in features distance spaces. In *European Conference on Computer Vision Workshops*, pages 0–0.

Chadebec, C., Vincent, L., and Allassonnière, S. (2022). Pythae: Unifying generative autoencoders in python-a benchmarking use case. *Advances in Neural Information Processing Systems*, 35:21575–21589.

Chakrabarti, A. (2016). A neural approach to blind motion deblurring. In *European Conference on Computer Vision*, pages 221–235.

Chambolle, A., Caselles, V., Cremers, D., Novaga, M., and Pock, T. (2010). An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9(263-340):227.

Chambolle, A. and Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40:120–145.

Chan, S. H., Wang, X., and Elgendy, O. A. (2016). Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98.

Chandramouli, P., Gandikota, K. V., Goerlitz, A., Kolb, A., and Moeller, M. (2022). A generative model for generic light field reconstruction. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 44(04):1712–1724.

Chang, R. J., Li, C.-L., Poczos, B., Vijaya Kumar, B., and Sankaranarayanan, A. C. (2017). One network to solve them all–solving linear inverse problems using deep projection models. In *IEEE International Conference on Computer Vision*, pages 5888–5897.

Chantas, G. K., Galatsanos, N. P., and Likas, A. C. (2006). Bayesian restoration using a new nonstationary edge-preserving image prior. *IEEE Transactions on Image Processing*, 15(10):2987–2997.

Chaurasia, G., Duchene, S., Sorkine-Hornung, O., and Drettakis, G. (2013). Depth synthesis and local warps for plausible image-based navigation. *ACM Transactions on Graphics*, 32(3).

Chen, B., Ruan, L., and Lam, M.-L. (2020a). Lfgan: 4d light field synthesis from a single rgb image. *ACM Trans. Multimedia Comput. Commun. Appl.*, 16(1).

Chen, D. and Davies, M. E. (2020). Deep decomposition learning for inverse imaging problems. In *European Conference on Computer Vision*, pages 510–526. Springer.

Chen, G., Dumay, A., and Tang, M. (2021). diffvg+CLIP: Generating painting trajectories from text. *preprint*.

Chen, H., Zhang, Y., Kalra, M. K., Lin, F., Chen, Y., Liao, P., Zhou, J., and Wang, G. (2017). Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE Transactions on Medical Imaging*, 36(12):2524–2535.

Chen, J., Shen, Y., Gao, J., Liu, J., and Liu, X. (2018). Language-based image editing with recurrent attentive models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8721–8729.

Chen, L., Chu, X., Zhang, X., and Sun, J. (2022a). Simple baselines for image restoration. In *European Conference on Computer Vision*.

Chen, L., Fang, F., Wang, T., and Zhang, G. (2019). Blind image deblurring with local maximum gradient prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Chen, X., Liang, C., Huang, D., Real, E., Wang, K., Liu, Y., Pham, H., Dong, X., Luong, T., Hsieh, C.-J., Lu, Y., and Le, Q. V. (2023a). Symbolic discovery of optimization algorithms.

Chen, X., Wang, X., Zhou, J., Qiao, Y., and Dong, C. (2023b). Activating more pixels in image super-resolution transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22367–22377.

Chen, Y.-C., Gao, C., Robb, E., and Huang, J.-B. (2020b). Nas-dip: Learning deep image prior with neural architecture search. In *16th European Conference on Computer Vision*, pages 442–459. Springer.

Chen, Z., Jin, X., Li, L., and Wang, G. (2013). A limited-angle ct reconstruction method based on anisotropic tv minimization. *Physics in Medicine & Biology*, 58(7):2119.

Chen, Z., Zhang, Y., Gu, J., Kong, L., Yuan, X., et al. (2022b). Cross aggregation transformer for image restoration. *Advances in Neural Information Processing Systems*, 35:25478–25490.

Cheng, K., Calivá, F., Shah, R., Han, M., Majumdar, S., and Pedoia, V. (2020). Addressing the false negative problem of deep learning mri reconstruction models by adversarial attacks and robust training. In *3rd Conference on Medical Imaging with Deep Learning*. PMLR.

Cho, S.-J., Ji, S.-W., Hong, J.-P., Jung, S.-W., and Ko, S.-J. (2021). Rethinking coarse-to-fine approach in single image deblurring. In *IEEE/CVF international conference on computer vision*, pages 4641–4650.

Choi, J., Kim, S., Jeong, Y., Gwon, Y., and Yoon, S. (2021). Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*.

Choi, J.-H., Zhang, H., Kim, J.-H., Hsieh, C.-J., and Lee, J.-S. (2019). Evaluating robustness of deep image super-resolution against adversarial attacks. In *IEEE/CVF International Conference on Computer Vision*.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., and Choo, J. (2018). Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *IEEE conference on computer vision and pattern recognition*, pages 8789–8797.

Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. (2020). Stargan v2: Diverse image synthesis for multiple domains. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197.

Chung, H., Kim, J., Kim, S., and Ye, J. C. (2022a). Parallel diffusion models of operator and image for blind inverse problems.

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. (2023). Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*.

Chung, H., Sim, B., Ryu, D., and Ye, J. C. (2022b). Improving diffusion models for inverse problems using manifold constraints. In *Advances in Neural Information Processing Systems*.

Chung, H., Sim, B., and Ye, J. C. (2022c). Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12413–12422.

Cohen, J., Rosenfeld, E., and Kolter, Z. (2019). Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR.

Cohen, R., Blau, Y., Freedman, D., and Rivlin, E. (2021). It has potential: Gradient-driven denoisers for convergent solutions to inverse problems. *Advances in Neural Information Processing Systems*, 34:18152–18164.

Cohen, T. and Welling, M. (2016). Group equivariant convolutional networks. In *International Conference on Machine Learning*, pages 2990–2999.

Collins, E., Bala, R., Price, B., and Susstrunk, S. (2020). Editing in style: Uncovering the local semantics of gans. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5771–5780.

Combettes, P. L. and Pesquet, J.-C. (2020). Lipschitz certificates for layered network structures driven by averaged activation operators. *SIAM Journal on Mathematics of Data Science*, 2(2):529–557.

Couairon, G., Grechka, A., Verbeek, J., Schwenk, H., and Cord, M. (2022). Flexit: Towards flexible semantic image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18270–18279.

Couairon, G., Verbeek, J., Schwenk, H., and Cord, M. (2023). Diffedit: Diffusion-based semantic image editing with mask guidance. In *The Eleventh International Conference on Learning Representations*.

Croce, F. and Hein, M. (2022). On the interplay of adversarial robustness and architecture components: patches, convolution and attention. In *ICML Workshop on New Frontiers in Adversarial Machine Learning*.

Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L., and Raff, E. (2022). Vqgan-clip: Open domain image generation and editing with natural language guidance. In *European Conference on Computer Vision*.

Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. (2007). Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095.

Daras, G., Dean, J., Jalal, A., and Dimakis, A. (2021). Intermediate layer optimization for inverse problems using deep generative models. In *International Conference on Machine Learning (ICML)*.

Darestani, M. Z., Chaudhari, A. S., and Heckel, R. (2021). Measuring robustness in deep learning based compressive sensing. In *International Conference on Machine Learning*, pages 2433–2444. PMLR.

Das, N., Shanbhogue, M., Chen, S.-T., Hohman, F., Li, S., Chen, L., Kounavis, M. E., and Chau, D. H. (2018). Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 196–204.

Daskalakis, C., Ilyas, A., Syrgkanis, V., and Zeng, H. (2018). Training GANs with optimism. In *International Conference on Learning Representations*.

de Jorge Aranda, P., Bibi, A., Volpi, R., Sanyal, A., Torr, P., Rogez, G., and Dokania, P. (2022). Make some noise: Reliable and efficient single-step adversarial training. *Advances in Neural Information Processing Systems*, 35:12881–12893.

Deng, H., Birdal, T., , and Ilic, S. (2018). Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *European Conference on Computer Vision*.

Dhar, M., Grover, A., and Ermon, S. (2018). Modeling sparse deviations for compressed sensing using generative models. In *International Conference on Machine Learning*, pages 1214–1223. PMLR.

Dhariwal, P. and Nichol, A. (2021). Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34.

Ding, M., Yang, Z., Hong, W., Zheng, W., Zhou, C., Yin, D., Lin, J., Zou, X., Shao, Z., Yang, H., and Tang, J. (2021). CogView: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). Density estimation using real NVP. In *International Conference on Learning Representations*.

Dinh, T. M., Tran, A. T., Nguyen, R., and Hua, B.-S. (2022). Hyperinverter: Improving stylegan inversion via hypernetwork. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Dong, H., Yu, S., Wu, C., and Guo, Y. (2017). Semantic image synthesis via adversarial learning. In *IEEE International Conference on Computer Vision*, pages 5706–5714.

Dong, J., Roth, S., and Schiele, B. (2020). Deep wiener deconvolution: Wiener meets deep learning for image deblurring. *Advances in Neural Information Processing Systems*, 33.

Dong, Y., Ruan, S., Su, H., Kang, C., Wei, X., and Zhu, J. (2022). Viewfool: Evaluating the robustness of visual recognition to adversarial viewpoints. *Advances in Neural Information Processing Systems*, 35:36789–36803.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306.

Donoho, D. L. and Elad, M. (2003). Maximal sparsity representation via l1 minimization. *Proceedings National Academy of Sciences*, 100(5):2197–2202.

Dröge, H., Bahat, Y., Heide, F., and Möller, M. (2022). Explorable data consistent ct reconstruction. In *British Machine Vision Conference*.

Eboli, T., Sun, J., and Ponce, J. (2020). End-to-end interpretable learning of non-blind image deblurring. In *European Conference on Computer Vision*, pages 314–331. Springer.

Eisenberger, M., Toker, A., Leal-Taixé, L., and Cremers, D. (2020). Deep shells: Unsupervised shape correspondence with optimal transport. In *Advances in Neural information processing systems*.

Elad, M. (2010). *Sparse and redundant representations: from theory to applications in signal and image processing*, volume 2. Springer.

Elad, M. and Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. (2019). Exploring the landscape of spatial robustness. In *International Conference on Machine Learning*, pages 1802–1811.

Engstrom, L., Tsipras, D., Schmidt, L., and Madry, A. (2017). A rotation and a translation suffice: Fooling cnns with simple transformations. *arXiv preprint arXiv:1712.02779*, 1(2):3.

Esser, P., Rombach, R., Blattmann, A., and Ommer, B. (2021a). Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. *Advances in Neural Information Processing Systems*, 34:3518–3532.

Esser, P., Rombach, R., and Ommer, B. (2021b). Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883.

Esteves, C., Allen-Blanchette, C., Makadia, A., and Daniilidis, K. (2018a). Learning so (3) equivariant representations with spherical cnns. In *European Conference on Computer Vision*, pages 52–68.

Esteves, C., Allen-Blanchette, C., Zhou, X., and Daniilidis, K. (2018b). Polar transformer networks. In *International Conference on Learning Representations*.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. (2018). Robust physical-world attacks on deep learning visual classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1625–1634.

Fang, J., Lin, H., Chen, X., and Zeng, K. (2022). A hybrid network of cnn and transformer for lightweight image super-resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1103–1112.

Fasel, B. and Gatica-Perez, D. (2006). Rotation-invariant neoperceptron. In *International Conference on Pattern Recognition*, volume 3, pages 336–339. IEEE.

Feinman, R., Curtin, R. R., Shintre, S., and Gardner, A. B. (2017). Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*.

Feldkamp, L. A., Davis, L. C., and Kress, J. W. (1984). Practical cone-beam algorithm. *Journal of the Optical Society of America A, Optics and image science*, 1(6):612–619.

Figueiredo, M. A., Bioucas-Dias, J. M., and Nowak, R. D. (2007). Majorization–minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.*, 16:2980–2991.

Finlayson, S. G., Bowers, J. D., Ito, J., Zittrain, J. L., Beam, A. L., and Kohane, I. S. (2019). Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289.

Fischer, M., Baader, M., and Vechev, M. (2020). Certified defense to image transformations via randomized smoothing. In *Advances in Neural information processing systems*, volume 33.

Fischer, V., Mummadi, C. K., Metzen, J. H., and Brox, T. (2017). Adversarial examples for semantic segmentation and object detection. In *International Conference on Learning Representations Workshops*.

Fischetti, M. and Jo, J. (2018). Deep neural networks and mixed integer linear optimization. *Constraints*, 23(3):296–309.

Frans, K., Soros, L. B., and Witkowski, O. (2021). Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *arXiv preprint arXiv:2106.14843*.

Freedman, G. and Fattal, R. (2011). Image and video upscaling from local self-examples. *ACM Transactions on Graphics (TOG)*, 30(2):1–11.

Fuchs, F., Worrall, D., Fischer, V., and Welling, M. (2020). Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural information processing systems*, 33.

Fuoli, D., Van Gool, L., and Timofte, R. (2021). Fourier space losses for efficient perceptual image super-resolution. In *IEEE/CVF International Conference on Computer Vision*, pages 2360–2369.

Gafni, O., Polyak, A., Ashual, O., Sheynin, S., Parikh, D., and Taigman, Y. (2022). Make-a-scene: Scene-based text-to-image generation with human priors. In *European Conference on Computer Vision*, pages 89–106. Springer.

Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13.

Galatolo, F. A., Cimino, M. G., and Vaglini, G. (2021). Generating images from caption and vice versa via clip-guided generative latent space search. *arXiv preprint arXiv:2102.01645*.

Gandikota, K. V. and Chandramouli, P. (2023). Exploring open domain image super-resolution through text. In *ICML 2023 Workshop on Artificial Intelligence & Human-Computer Interaction*.

Gandikota, K. V., Chandramouli, P., Dröge, H., and Moeller, M. (2023). Evaluating adversarial robustness of low dose CT recovery. In *Medical Imaging with Deep Learning*.

Gandikota, K. V., Chandramouli, P., and Moeller, M. (2022a). On adversarial robustness of deep image deblurring. In *IEEE International Conference on Image Processing*, pages 3161–3165.

Gandikota, K. V., Geiping, J., Laehner, Z., Czapliński, A., and Moeller, M. (2022b). A simple strategy to provable invariance via orbit mapping. In *Asian Conference on Computer Vision (ACCV)*, pages 3500–3518.

Gao, Y., Shumailov, I., Fawaz, K., and Papernot, N. (2022). On the limitations of stochastic pre-processing defenses. In *Advances in Neural Information Processing Systems*.

Gatys, L. A., Ecker, A. S., and Bethge, M. (2016). Image style transfer using convolutional neural networks. In *IEEE conference on computer vision and pattern recognition*, pages 2414–2423.

Ge, Y., Su, T., Zhu, J., Deng, X., Zhang, Q., Chen, J., Hu, Z., Zheng, H., and Liang, D. (2020). Adaptive-net: deep computed tomography reconstruction network with analytical domain transformation knowledge. *Quantitative Imaging in Medicine and Surgery*, 10(2):415.

Genzel, M., Macdonald, J., and März, M. (2022). Solving inverse problems with deep neural networks-robustness included. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Getreuer, P. (2012). Total variation deconvolution using split bregman. *Image Processing On Line*, 2:158–174.

Gidel, G., Berard, H., Vignoud, G., Vincent, P., and Lacoste-Julien, S. (2019). A variational inequality perspective on generative adversarial networks. In *International Conference on Learning Representations*.

Gilboa, G. (2013). Expert regularizers for task specific processing. In *4th International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, pages 24–35. Springer.

Gilton, D., Ongie, G., and Willett, R. (2021a). Deep equilibrium architectures for inverse problems in imaging. *IEEE Transactions on Computational Imaging*, 7:1123–1133.

Gilton, D., Ongie, G., and Willett, R. (2021b). Model adaptation for inverse problems in imaging. *IEEE Transactions on Computational Imaging*, 7:661–674.

Gong, D., Zhang, Z., Shi, Q., v. d. Hengel, A., Shen, C., and Zhang, Y. (2020). Learning deep gradient descent optimization for image deconvolution. *IEEE Transactions on Neural Networks and Learning Systems*, 31(12):5468–5482.

González, M., Almansa, A., and Tan, P. (2022). Solving inverse problems by joint posterior maximization with autoencoding prior. *SIAM Journal on Imaging Sciences*, 15(2):822–859.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Gossard, A. and Weiss, P. (2022). Training adaptive reconstruction networks for inverse problems. *arXiv preprint arXiv:2202.11342*.

Gottschling, N. M., Antun, V., Adcock, B., and Hansen, A. C. (2020). The troublesome kernel: why deep learning for inverse problems is typically unstable. *arXiv preprint arXiv:2001.01258*.

Goujon, A., Neumayer, S., Bohra, P., Ducotterd, S., and Unser, M. (2023). A neural-network-based convex regularizer for inverse problems. *IEEE Transactions on Computational Imaging*, pages 1–15.

Gowal, S., Rebuffi, S.-A., Wiles, O., Stimberg, F., Calian, D. A., and Mann, T. (2021). Improving robustness using generated data. In *Advances in Neural Information Processing Systems*.

Grabinski, J., Jung, S., Keuper, J., and Keuper, M. (2022). Frequencylowcut pooling-plug & play against catastrophic overfitting. In *17th European Conference on Computer Vision*, pages 36–57. Springer.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. (2020). Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*.

Gregor, K. and LeCun, Y. (2010). Learning fast approximations of sparse coding. In *27th international conference on international conference on machine learning*, pages 399–406.

Grosse, K., Manoharan, P., Papernot, N., Backes, M., and McDaniel, P. (2017). On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*.

Gu, J., Cai, H., Dong, C., Ren, J. S., Timofte, R., Gong, Y., Lao, S., Shi, S., et al. (2022a). Ntire 2022 challenge on perceptual image quality assessment. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 951–967.

Gu, J., Shen, Y., and Zhou, B. (2020). Image processing using multi-code gan prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3012–3021.

Gu, J. and Yeung, S. (2021). Staying in shape: learning invariant shape representations using contrastive learning. In *Conference on Uncertainty in Artificial Intelligence*, volume 161, pages 1852–1862. PMLR.

Gu, J., Zhao, H., Tresp, V., and Torr, P. H. S. (2022b). Segpgd: An effective and efficient adversarial attack for evaluating and boosting segmentation robustness. In *European Conference on Computer Vision*. Springer Nature Switzerland.

Gu, S., Chen, D., Bao, J., Wen, F., Zhang, B., Chen, D., Yuan, L., and Guo, B. (2022c). Vector quantized diffusion model for text-to-image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706.

Gu, S. and Rigazio, L. (2014). Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*.

Gul, M. S. K. and Gunturk, B. K. (2018). Spatial and angular resolution enhancement of light fields using convolutional neural networks. *IEEE Transactions on Image Processing*, 27(5):2146–2159.

Guo, C., Rana, M., Cisse, M., and van der Maaten, L. (2018). Countering adversarial images using input transformations. In *International Conference on Learning Representations*.

Guo, M., Yang, Y., Xu, R., Liu, Z., and Lin, D. (2020). When nas meets robustness: In search of robust architectures against adversarial attacks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 631–640.

Gupta, H., Jin, K. H., Nguyen, H. Q., McCann, M. T., and Unser, M. (2018). Cnn-based projected gradient descent for consistent ct image reconstruction. *IEEE Transactions on Medical Imaging*, 37(6):1440–1453.

Gupta, M., Jauhari, A., Kulkarni, K., Jayasuriya, S., Molnar, A., and Turaga, P. (2017). Compressive light field reconstructions using deep learning. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 11–20.

Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., and Cohen-Or, D. (2019). Meshcnn: A network with an edge. *ACM Transactions on Graphics (TOG)*, 38(4):90:1–90:12.

Hasannasab, M., Hertrich, J., Neumayer, S., Plonka, G., Setzer, S., and Steidl, G. (2020). Parseval proximal neural networks. *Journal of Fourier Analysis and Applications*, 26:1–31.

He, J., Wang, Y., and Ma, J. (2020). Radon inversion via deep learning. *IEEE Transactions on Medical Imaging*, 39(6):2076–2087.

He, J., Yang, Y., Wang, Y., Zeng, D., Bian, Z., Zhang, H., Sun, J., Xu, Z., and Ma, J. (2018). Optimizing a parameterized plug-and-play admm for iterative low-dose ct reconstruction. *IEEE Transactions on Medical Imaging*, 38(2):371–382.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heaton, H., Fung, S. W., Lin, A. T., Osher, S., and Yin, W. (2022). Wasserstein-based projections with applications to inverse problems. *SIAM Journal on Mathematics of Data Science*, 4(2):581–603.

Heber, S. and Pock, T. (2014). Shape from light field meets robust pca. In *European Conference on Computer Vision*, pages 751–767. Springer.

Heber, S. and Pock, T. (2016). Convolutional networks for shape from light field. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3746–3754.

Heckel, R. et al. (2019). Deep decoder: Concise image representations from untrained non-convolutional networks. In *International Conference on Learning Representations*.

Hegde, C. (2018). Algorithmic aspects of inverse problems using generative models. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 166–172. IEEE.

Hendrycks, D. and Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*.

Henriques, J. F. and Vedaldi, A. (2017). Warped convolutions: Efficient invariance to spatial transformations. In *International Conference on Machine Learning*, pages 1461–1469. PMLR.

Hestenes, M. R., Stiefel, E., et al. (1952). Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409–436.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc.

Ho, K., Gilbert, A., Jin, H., and Collomosse, J. (2021). Neural architecture search for deep image prior. *Computers & graphics*, 98:188–196.

Horie, M., Morita, N., Hishinuma, T., Ihara, Y., and Mitsume, N. (2020). Isometric transformation invariant and equivariant graph convolutional networks. In *International Conference on Learning Representations*.

Hu, J., Shoushtari, S., Zou, Z., Liu, J., Sun, Z., and Kamilov, U. S. (2023). Robustness of deep equilibrium architectures to changes in the measurement model. In *IEEEInternational Conference on Acoustics, Speech and Signal Processing*, pages 1–5.

Hu, M., Wang, Y., Cham, T.-J., Yang, J., and Suganthan, P. (2022). Global context with discrete diffusion in vector quantised modelling for image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11502–11511.

Huang, H., Wang, Y., Erfani, S., Gu, Q., Bailey, J., and Ma, X. (2021). Exploring architectural ingredients of adversarially robust deep neural networks. *Advances in Neural Information Processing Systems*, 34:5545–5559.

Huang, R., Rakotosaona, M.-J., Achlioptas, P., Guibas, L., and Ovsjanikov, M. (2019). Operatornet: Recovering 3d shapes from difference operators. In *International Conference on Computer Vision (ICCV)*.

Huang, S., Lu, Z., Deb, K., and Boddeti, V. (2023). Revisiting residual networks for adversarial robustness. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Huang, Y., Würfl, T., Breininger, K., Liu, L., Lauritsch, G., and Maier, A. (2018). Some investigations on robustness of deep learning in limited angle tomography. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 145–153. Springer.

Inagaki, Y., Kobayashi, Y., Takahashi, K., Fujii, T., and Nagahara, H. (2018). Learning to capture light fields through a coded aperture camera. In *European Conference on Computer Vision*, pages 418–434.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Ivan, A., Park, I. K., et al. (2019). Synthesizing a 4d spatio-angular consistent light field from a single image. *arXiv preprint arXiv:1903.12364*.

Jaderberg, M., Simonyan, K., Zisserman, A., and Kavukcuoglu, K. (2015). Spatial transformer networks. In *Advances in Neural information processing systems*.

Jafari-Khouzani, K. and Soltanian-Zadeh, H. (2005). Radon transform orientation estimation for rotation invariant texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):1004–1008.

Jalal, A., Arvinte, M., Daras, G., Price, E., Dimakis, A. G., and Tamir, J. (2021a). Robust compressed sensing mri with deep generative priors. In *Advances in Neural Information Processing Systems*, volume 34, pages 14938–14954. Curran Associates, Inc.

Jalal, A., Karmalkar, S., Hoffmann, J., Dimakis, A., and Price, E. (2021b). Fairness for image generation with uncertain sensitive attributes. In *International Conference on Machine Learning*, pages 4721–4732. PMLR.

Jenni, S. and Favaro, P. (2019). On stabilizing generative adversarial training with noise. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Jeon, H.-G., Park, J., Choe, G., Park, J., Bok, Y., Tai, Y.-W., and So Kweon, I. (2015). Accurate depth map estimation from a lenslet light field camera. In *IEEE conference on computer vision and pattern recognition*, pages 1547–1555.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR.

Jin, J., Hou, J., Yuan, H., and Kwong, S. (2020). Learning light field angular super-resolution via a geometry-aware network. In *AAAI conference on artificial intelligence*, volume 34, pages 11141–11148.

Jin, K. H., McCann, M. T., Froustey, E., and Unser, M. (2017). Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522.

Jo, Y., Oh, S. W., Vajda, P., and Kim, S. J. (2021a). Tackling the ill-posedness of super-resolution through adaptive target generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16236–16245.

Jo, Y., Yang, S., and Kim, S. J. (2021b). Srflow-da: Super-resolution using normalizing flow with deep convolutional block. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 364–372.

Johannsen, O., Sulc, A., and Goldluecke, B. (2016). What sparse light field coding reveals about scene structure. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3262–3270.

Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer.

Jordan, M. and Dimakis, A. G. (2020). Exactly computing the local lipschitz constant of relu networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 7344–7353. Curran Associates, Inc.

Kaba, S.-O., Mondal, A. K., Zhang, Y., Bengio, Y., and Ravanbakhsh, S. (2023). Equivariance with learned canonicalization functions. In *International Conference on Machine Learning*, pages 15546–15566. PMLR.

Kadkhodaie, Z. and Simoncelli, E. P. (2021). Stochastic solutions for linear inverse problems using the prior implicit in a denoiser. In *Advances in Neural Information Processing Systems.*

Kaipio, J. and Somersalo, E. (2006). *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media.

Kalantari, N. K., Wang, T.-C., and Ramamoorthi, R. (2016). Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)*, 35(6):193.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018a). Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations.*

Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018b). Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations.*

Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., and Aila, T. (2021). Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863.

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119.

Kawar, B., Elad, M., Ermon, S., and Song, J. (2022a). Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems.*

Kawar, B., Song, J., Ermon, S., and Elad, M. (2022b). JPEG artifact correction using denoising diffusion restoration models. In *NeurIPS 2022 Workshop on Score-Based Methods.*

Kawar, B., Vaksman, G., and Elad, M. (2021). SNIPS: Solving noisy inverse problems stochastically. In *Advances in Neural Information Processing Systems*.

Kim, G., Kwon, T., and Ye, J. C. (2022). Diffusionclip: Text-guided diffusion models for robust image manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435.

Kim, H., Jhoo, H. Y., Park, E., and Yoo, S. (2019). Tag2pix: Line art colorization using text tag with secat and changing loss. In *IEEE/CVF International Conference on Computer Vision*, pages 9055–9064.

Kim, H., Lee, W., and Lee, J. (2021). Understanding catastrophic overfitting in single-step adversarial training. In *AAAI Conference on Artificial Intelligence*, volume 35, pages 8119–8127.

Kim, J., Lee, K., and Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kobler, E., Effland, A., Kunisch, K., and Pock, T. (2020). Total deep variation for linear inverse problems. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7549–7558.

Koh, J., Lee, J., and Yoon, S. (2021). Single-image deblurring with neural networks: A comparative survey. *Computer Vision and Image Understanding*, 203.

Komodakis, N. and Gidaris, S. (2018). Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*.

Kothari, K., Khorashadizadeh, A., de Hoop, M., and Dokmanić, I. (2021). Trumpets: Injective flows for inference and inverse problems. In *Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161, pages 1269–1278. PMLR.

Krishnan, D. and Fergus, R. (2009). Fast image deconvolution using hyper-laplacian priors. *Neural Information Processing Systems*, 22.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25.

Kuanar, S., Athitsos, V., Mahapatra, D., Rao, K., Akhtar, Z., and Dasgupta, D. (2019). Low dose abdominal ct image reconstruction: An unsupervised learning based approach. In *IEEE international conference on image processing*, pages 1351–1355.

Kupyn, O., Martyniuk, T., Wu, J., and Wang, Z. (2019). Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *IEEE/CVF International Conference on Computer Vision*.

Kurakin, A., Goodfellow, I. J., and Bengio, S. (2018). Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC.

Kwon, G. and Ye, J. C. (2022). Clipstyler: Image style transfer with a single text condition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18062–18071.

Lang, I., Kotlicki, U., and Avidan, S. (2021). Geometric adversarial attacks and defenses on 3d point clouds. In *2021 International Conference on 3D Vision (3DV)*.

Laptev, D., Savinov, N., Buhmann, J. M., and Pollefeys, M. (2016). Ti-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 289–297.

Latorre, F., Cevher, V., et al. (2019a). Fast and provable admm for learning with generative priors. In *Advances in Neural Information Processing Systems*.

Latorre, F., eftekhari, A., and Cevher, V. (2019b). Fast and provable admm for learning with generative priors. In *Advances in Neural Information Processing Systems 32*, pages 12004–12016. Curran Associates, Inc.

Lazebnik, S., Schmid, C., and Ponce, J. (2005). A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278.

Lee, D., Kim, J., Choi, J., Kim, J., Byeon, M., Baek, W., and Kim, S. (2022). Karlo-v1.0.alpha on coyo-100m and cc15m. `https://github.com/kakaobrain/karlo`.

Lee, D., Yoo, J., and Ye, J. C. (2017). Deep residual learning for compressed sensing mri. In *2017IEEE14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 15–18.

Lefkimmiatis, S., Ward, J. P., and Unser, M. (2013). Hessian Schatten-norm regularization for linear inverse problems. *IEEE Transactions on Image Processing*, 22(5):1873–1888.

Lenc, K. and Vedaldi, A. (2018). Understanding image representations by measuring their equivariance and equivalence. *International Journal of Computer Vision*, 127(5):456–476.

Leuschner, J., Schmidt, M., Baguer, D. O., and Maass, P. (2021). Lodopab-ct, a benchmark dataset for low-dose computed tomography reconstruction. *Scientific Data*, 8(1):1–12.

Levoy, M. and Hanrahan, P. (1996). Light field rendering. In *23rd annual conference on Computer graphics and interactive techniques*, pages 31–42.

Li, B., Qi, X., Lukasiewicz, T., and Torr, P. (2019). Controllable text-to-image generation. *Advances in Neural Information Processing Systems*, 32.

Li, B., Qi, X., Lukasiewicz, T., and Torr, P. H. (2020a). Manigan: Text-guided image manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7880–7889.

Li, C., Li, L., Geng, Y., Jiang, H., Cheng, M., Zhang, B., Ke, Z., Xu, X., and Chu, X. (2023). Yolov6 v3. 0: A full-scale reloading. *arXiv preprint arXiv:2301.05586*.

Li, F., Fujiwara, K., Okura, F., and Matsushita, Y. (2021). A closer look at rotation-invariant deep point cloud analysis. In *International Conference on Computer Vision (ICCV)*, pages 16218–16227.

Li, H., Schwab, J., Antholzer, S., and Haltmeier, M. (2020b). Nett: Solving inverse problems with deep neural networks. *Inverse Problems*, 36(6):065005.

Li, H., Yang, Y., Chang, M., Chen, S., Feng, H., Xu, Z., Li, Q., and Chen, Y. (2022a). Srdiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479:47–59.

Li, J., Li, D., Xiong, C., and Hoi, S. (2022b). BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR.

Li, M., He, L., and Lin, Z. (2020c). Implicit Euler skip connections: Enhancing adversarial robustness via numerical stability. In *37th International Conference on Machine Learning*, volume 119, pages 5874–5883. PMLR.

Li, S. Z. (1994). Markov random field models in computer vision. In *Third European Conference on Computer Vision*, pages 361–370. Springer.

Li, Y., Liu, S., Yang, J., and Yang, M.-H. (2017). Generative face completion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3911–3919.

Liang, C.-K., Lin, T.-H., Wong, B.-Y., Liu, C., and Chen, H. H. (2008a). Programmable aperture photography: Multiplexed light field acquisition. *ACM Transactions on Graphics*, 27(3):1–10.

Liang, C.-K., Lin, T.-H., Wong, B.-Y., Liu, C., and Chen, H. H. (2008b). Programmable aperture photography: Multiplexed light field acquisition. *ACM Transactions on Graphics*, 27(3):55:1–55:10.

Lin, W.-Y., Sheikholeslami, F., jinghao shi, Rice, L., and Kolter, J. Z. (2021). Certified robustness against adversarial patch attacks via randomized cropping. In *ICML 2021 Workshop on Adversarial Machine Learning*.

Litany, O., Remez, T., Rodolà, E., Bronstein, A., and Bronstein, M. (2017). Deep functional maps: Structured prediction for dense shape correspondences. In *International Conference on Computer Vision (ICCV)*.

Liu, X., Gong, C., Wu, L., Zhang, S., Su, H., and Liu, Q. (2021). Fusedream: Training-free text-to-image generation with improved clip+ gan space optimization. *arXiv preprint arXiv:2112.01573*.

Liu, X., Lin, Z., Zhang, J., Zhao, H., Tran, Q., Wang, X., and Li, H. (2020). Open-edit: Open-domain image manipulation with open-vocabulary instructions. In *European Conference on Computer Vision*, pages 89–106. Springer.

Liu, X., Park, D. H., Azadi, S., Zhang, G., Chopikyan, A., Hu, Y., Shi, H., Rohrbach, A., and Darrell, T. (2023a). More control for free! image synthesis with semantic diffusion guidance. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 289–299.

Liu, Y., Li, J., Pang, Y., Nie, D., and Yap, P.-t. (2023b). The devil is in the upsampling: Architectural decisions made simpler for denoising with deep image prior. In *International Conference on Computer Vision*.

Lu, B., Liu, J., and Xiong, H. (2022). Transformation-based adversarial defense via sparse representation. In *IEEE International Conference on Image Processing (ICIP)*, pages 1726–1730.

Lu, J., Issaranon, T., and Forsyth, D. (2017). Safetynet: Detecting and rejecting adversarial examples robustly. In *IEEE International Conference on Computer Vision*, pages 446–454.

Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., and Van Gool, L. (2022a). Repaint: Inpainting using denoising diffusion probabilistic models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471.

Lugmayr, A., Danelljan, M., and Timofte, R. (2021). Ntire 2021 learning the super-resolution space challenge. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 596–612.

Lugmayr, A., Danelljan, M., Timofte, R., Kim, K.-w., Kim, Y., Lee, J.-y., et al. (2022b). Ntire 2022 challenge on learning the super-resolution space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Lugmayr, A., Danelljan, M., Van Gool, L., and Timofte, R. (2020). Srflow: Learning the super-resolution space with normalizing flow. In *European Conference on Computer Vision*, pages 715–732. Springer.

Lunz, S., Öktem, O., and Schönlieb, C.-B. (2018). Adversarial regularizers in inverse problems. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada*, pages 8507–8516.

Luo, C. (2022). Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*.

Ma, C., Yan, B., Lin, Q., Tan, W., and Chen, S. (2022). Rethinking super-resolution as text-guided details generation. In *30th ACM International Conference on Multimedia*, pages 3461–3469.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

Mairal, J., Bach, F., and Ponce, J. (2014). Sparse modeling for image and vision processing. *Foundations and Trends® in Computer Graphics and Vision*, 8(2-3):85–283.

Manay, S., Cremers, D., Hong, B.-W., Yezzi, A. J., and Soatto, S. (2006). Integral invariants for shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1602–1618.

Mansimov, E., Parisotto, E., Ba, J. L., and Salakhutdinov, R. (2016). Generating images from captions with attention. In *International Conference on Learning Representations*.

Mansour, Y., Lin, K., and Heckel, R. (2022). Image-to-image MLP-mixer for image reconstruction.

Manthalkar, R., Biswas, P. K., and Chatterji, B. N. (2003). Rotation and scale invariant texture features using discrete wavelet packet transform. *Pattern Recognition Letters*, 24(14):2455–2462.

Marcos, D., Volpi, M., Komodakis, N., and Tuia, D. (2017). Rotation equivariant vector field networks. In *IEEE International Conference on Computer Vision*, pages 5048–5057.

Marcos, D., Volpi, M., and Tuia, D. (2016). Learning rotation invariant convolutional filters for texture classification. In *International Conference on Pattern Recognition*, pages 2012–2017. IEEE.

Mardani, M., Song, J., Kautz, J., and Vahdat, A. (2023). A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*.

Mardani, M., Sun, Q., Donoho, D., Papyan, V., Monajemi, H., Vasanawala, S., and Pauly, J. (2018). Neural proximal gradient descent for compressive imaging. *Advances in Neural Information Processing Systems*, 31.

Marinescu, R., Moyer, D., and Golland, P. (2021). Bayesian image reconstruction using deep generative models. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*.

Marwah, K., Wetzstein, G., Bando, Y., and Raskar, R. (2013). Compressive light field photography using overcomplete dictionaries and optimized projections. *ACM Transactions on Graphics (TOG)*, 32(4):46.

Meinhardt, T., Moller, M., Hazirbas, C., and Cremers, D. (2017). Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *IEEE International Conference on Computer Vision*, pages 1781–1790.

Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. (2022). SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*.

Meng, D. and Chen, H. (2017). Magnet: a two-pronged defense against adversarial examples. In *2017 ACM SIGSAC conference on computer and communications security*, pages 135–147.

Meng, N., So, H. K.-H., Sun, X., and Lam, E. Y. (2021). High-dimensional dense residual convolutional neural network for light field reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3):873–886.

Meng, N., Zeng, T., and Lam, E. Y. (2019). Spatial and angular reconstruction of light field based on deep generative networks. In *IEEE International Conference on Image Processing*, pages 4659–4663.

Menon, S., Damian, A., Hu, S., Ravi, N., and Rudin, C. (2020). Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2445.

Metzen, J. H., Genewein, T., Fischer, V., and Bischoff, B. (2017). On detecting adversarial perturbations. In *International Conference on Learning Representations*.

Michaeli, T. and Irani, M. (2014). Blind deblurring using internal patch recurrence. In *13th European Conference on Computer Vision*, pages 783–798. Springer.

Mildenhall, B., Srinivasan, P. P., Ortiz-Cayon, R., Kalantari, N. K., Ramamoorthi, R., Ng, R., and Kar, A. (2019). Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*.

Mittal, A., Soundararajan, R., and Bovik, A. C. (2012). Making a “completely blind” image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212.

Miyato, T., Maeda, S.-i., Koyama, M., and Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993.

Moeller, M., Mollenhoff, T., and Cremers, D. (2019). Controlling neural networks via energy dissipation. In *IEEE/CVF International Conference on Computer Vision*, pages 3256–3265.

Mokady, R., Hertz, A., Aberman, K., Pritch, Y., and Cohen-Or, D. (2023). Null-text inversion for editing real images using guided diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047.

Monga, V., Li, Y., and Eldar, Y. C. (2021). Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44.

Montanaro, A., Valsesia, D., and Magli, E. (2022). Exploring the solution space of linear inverse problems with gan latent geometry. In *IEEE International Conference on Image Processing*, pages 1381–1385.

Monti, F., Boscaini, D., Masci, J., Rodolá, E., Svoboda, J., and Bronstein, M. M. (2016). Geometric deep learning on graphs and manifolds using mixture model cnns. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Morshuis, J. N., Gatidis, S., Hein, M., and Baumgartner, C. F. (2022). Adversarial robustness of mr image reconstruction under realistic perturbations. In *International Workshop on Machine Learning for Medical Image Reconstruction*, pages 24–33. Springer.

Mousavi, A., Patel, A. B., and Baraniuk, R. G. (2015). A deep learning approach to structured signal recovery. In *2015 53rd annual allerton conference on communication, control, and computing (Allerton)*, pages 1336–1343. IEEE.

Mukherjee, S., Carioni, M., Öktem, O., and Schönlieb, C.-B. (2021). End-to-end reconstruction meets data-driven regularization for inverse problems. In *Advances in Neural Information Processing Systems*, volume 34, pages 21413–21425.

Mukherjee, S., Dittmer, S., Shumaylov, Z., Lunz, S., Öktem, O., and Schönlieb, C.-B. (2020). Learned convex regularizers for inverse problems. *arXiv:2008.02839v2*.

Murata, N., Saito, K., Lai, C.-H., Takida, Y., Uesaka, T., Mitsufuji, Y., and Ermon, S. (2023). Gibbsddrm: A partially collapsed gibbs sampler for solving blind inverse problems with denoising diffusion restoration. In *International Conference on Machine Learning*.

Mustafa, A., Mikhailiuk, A., Iliescu, D. A., Babbar, V., and Mantiuk, R. K. (2022). Training a task-specific image reconstruction loss. In *IEEE/CVF winter conference on applications of computer vision*, pages 2319–2328.

Nabati, O., Mendlovic, D., and Giryes, R. (2018). Fast and accurate reconstruction of compressed color light field. In *2018 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11.

Nam, S., Kim, Y., and Kim, S. J. (2018). Text-adaptive generative adversarial networks: Manipulating images with natural language. In *Advances in Neural Information Processing Systems*, volume 31, pages 42–51. Curran Associates, Inc.

Nan, Y. and Ji, H. (2020). Deep learning for handling kernel/model uncertainty in image deconvolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Navarro, J. and Sabater, N. (2021). Learning occlusion-aware view synthesis for light fields. *Pattern Analysis and Applications*, 24(3):1319–1334.

Newman, B. and Callahan, M. J. (2011). Alara (as low as reasonably achievable) ct 2011—executive summary. *Pediatric radiology*, 41(2):453–455.

Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., and Hanrahan, P. (2005). Light field photography with a hand-held plenoptic camera. *Stanford Tech. Report CTSR*, 2005(2):1–11.

Nichol, A. Q. and Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR.

Nichol, A. Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., and Chen, M. (2022). GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *39th International Conference on Machine Learning*, volume 162, pages 16784–16804. PMLR.

Nie, W., Guo, B., Huang, Y., Xiao, C., Vahdat, A., and Anandkumar, A. (2022). Diffusion models for adversarial purification. In *International Conference on Machine Learning*, pages 16805–16827. PMLR.

Nikolova, M. and Ng, M. K. (2005). Analysis of half-quadratic minimization methods for signal and image recovery. *SIAM Journal on Scientific computing*, 27(3):937–966.

Noroozi, M., Chandramouli, P., and Favaro, P. (2017). Motion deblurring in the wild. In *Pattern Recognition: 39th German Conference, GCPR 2017, Basel, Switzerland, September 12–15, 2017, Proceedings 39*, pages 65–77. Springer.

Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer.

Ohayon, G., Adrai, T. J., Elad, M., and Michaeli, T. (2023). Reasons for the superiority of stochastic estimators over deterministic ones: Robustness, consistency and perceptual quality. In *International Conference on Machine Learning*, pages 26474–26494. PMLR.

Olah, C., Cammarata, N., Voss, C., Schubert, L., and Goh, G. (2020). Naturally occurring equivariance in neural networks. *Distill*, 5(12):e00024–004.

Ordonez, V., Kulkarni, G., and Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24.

Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., and Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics*, 31(4).

Oyallon, E. and Mallat, S. (2015). Deep roto-translation scattering for object classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Paiss, R., Chefer, H., and Wolf, L. (2022). No token left behind: Explainability-aided image classification and generation. In *European Conference on Computer Vision*. Springer-Verlag.

Pan, J., Sun, D., Pfister, H., and Yang, M.-H. (2016). Blind image deblurring using dark channel prior. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1628–1636.

Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C. C., and Luo, P. (2021). Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7474–7489.

Pang, J., Jiang, C., Chen, Y., Chang, J., Feng, M., Wang, R., and Yao, J. (2022). 3d shuffle-mixer: An efficient context-aware vision learner of transformer-mlp paradigm for dense prediction in medical volume. *IEEE Transactions on Medical Imaging*.

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. V. (2012). Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., and Lischinski, D. (2021). StyleCLIP: Text-driven manipulation of stylegan imagery. In *IEEE/CVF International Conference on Computer Vision*, pages 2085–2094.

Pelt, D. M., Batenburg, K. J., and Sethian, J. A. (2018). Improving tomographic reconstruction from limited data using mixed-scale dense convolutional neural networks. *Journal of Imaging*, 4(11).

Peng, S. and Li, K. (2020). Generating unobserved alternatives: A case study through super-resolution and decompression. *OpenReview*.

Perrone, D. and Favaro, P. (2014). Total variation blind deconvolution: The devil is in the details. In *2014IEEEConference on Computer Vision and Pattern Recognition*, pages 2909–2916.

Pinkall, U. and Polthier, K. (1993). Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*.

Prost, J., Houdard, A., Almansa, A., and Papadakis, N. (2021). Learning local regularization for variational image restoration. In *Scale Space and Variational Methods in Computer Vision: 8th In-*

ternational Conference, SSVM 2021, Virtual Event, May 16–20, 2021, Proceedings, pages 358–370. Springer.

Prost, J., Houdard, A., Almansa, A., and Papadakis, N. (2023). Inverse problem regularization with hierarchical variational autoencoders. *arXiv preprint arXiv:2303.11217*.

Putzky, P. and Welling, M. (2017). Recurrent inference machines for solving inverse problems.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017a). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017b). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural information processing systems*.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *38th International Conference on Machine Learning*, pages 8748–8763. PMLR.

Radon, J. (1986). On the determination of functions from their integral values along certain manifolds. *IEEE Transactions on Medical Imaging*, 5(4):170–176.

Raff, E., Sylvester, J., Forsyth, S., and McLean, M. (2019). Barrage of random transforms for adversarially robust defense. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6528–6537.

Raghunathan, A., Steinhardt, J., and Liang, P. S. (2018). Semidefinite relaxations for certifying robustness to adversarial examples. *Advances in neural information processing systems*, 31.

Raj, A., Bresler, Y., and Li, B. (2020). Improving robustness of deep-learning-based image reconstruction. In *International Conference on Machine Learning*, volume 119 of *PMLR*, pages 7932–7942.

Raj, A., Li, Y., and Bresler, Y. (2019). Gan-based projector for faster recovery with convergence guarantees in linear inverse problems. In *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In *38th International Conference on Machine Learning*, pages 8821–8831. PMLR.

Rao, S., Stutz, D., and Schiele, B. (2020). Adversarial training against location-optimized adversarial patches. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 429–448. Springer.

Rao, Y., Lu, J., and Zhou, J. (2019). Spherical fractal convolutional neural networks for point cloud recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–460.

Ravanbakhsh, S., Schneider, J., and Poczos, B. (2017). Equivariance through parameter-sharing. In *International Conference on Machine Learning*, pages 2892–2901. PMLR.

Razavi, A., van den Oord, A., and Vinyals, O. (2019). Generating diverse high-fidelity images with vq-vae-2. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Razavi-Far, R., Ruiz-Garcia, A., Palade, V., and Schmidhuber, J. (2022). *Generative adversarial learning: architectures and applications*. Springer.

Rebuffi, S.-A., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. (2021). Data augmentation can improve robustness. In *Advances in Neural Information Processing Systems*.

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. (2016). Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, volume 48, pages 1060–1069. PMLR.

Reehorst, E. T. and Schniter, P. (2018). Regularization by denoising: Clarifications and new interpretations. *IEEE Transactions on Computational Imaging*, 5(1):52–67.

Rehman, H. Z. U. and Lee, S. (2018). Automatic image alignment using principal component analysis. *IEEE Access*, 6:72063–72072.

Rempe, D., Birdal, T., Zhao, Y., Gojcic, Z., Sridhar, S., and Guibas, L. J. (2020). Caspr: Learning canonical spatiotemporal point cloud representations. *Advances in Neural information processing systems*, 33:13688–13701.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., and Cohen-Or, D. (2021). Encoding in style: a stylegan encoder for image-to-image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2287–2296.

Rojas-Gomez, R. A., Lim, T.-Y., Schwing, A., Do, M., and Yeh, R. A. (2022). Learnable polyphase sampling for shift invariant and equivariant convolutional networks. *Advances in Neural Information Processing Systems*, 35:35755–35768.

Romano, Y., Elad, M., and Milanfar, P. (2017). The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Rony, J., Hafemann, L. G., Oliveira, L. S., Ayed, I. B., Sabourin, R., and Granger, E. (2019). Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4322–4330.

Rony, J., Pesquet, J.-C., and Ben Ayed, I. (2023). Proximal splitting adversarial attack for semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20524–20533.

Roth, K., Kilcher, Y., and Hofmann, T. (2019). The odds are odd: A statistical test for detecting adversarial examples. In *36th International Conference on Machine Learning*, volume 97, pages 5498–5507. PMLR.

Roth, S. and Black, M. J. (2009). Fields of experts. *International Journal of Computer Vision*, 82(2):205–229.

Rotman, J. J. (2012). *An introduction to the theory of groups*, volume 148. Springer Science & Business Media.

Rott Shaham, T. and Michaeli, T. (2016). Visualizing image priors. In *14th European Conference on Computer Vision*, pages 136–153. Springer.

Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Runkel, C., Moeller, M., Schönlieb, C.-B., and Etmann, C. (2023). Learning posterior distributions in underdetermined inverse problems. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 187–209. Springer.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Ryu, E., Liu, J., Wang, S., Chen, X., Wang, Z., and Yin, W. (2019). Plug-and-play methods provably converge with properly trained denoisers. In *International Conference on Machine Learning*, pages 5546–5557. PMLR.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. (2022a). Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. (2022b). Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. (2022c). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., and Norouzi, M. (2023). Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4713–4726.

Sajjadi, M. S. M., Köhler, R., Schölkopf, B., and Hirsch, M. (2016). Depth estimation through a generative model of light field synthesis. In *Pattern Recognition*, pages 426–438, Cham. Springer International Publishing.

Sajnani, R., Poulenard, A., Jain, J., Dua, R., Guibas, L. J., and Sridhar, S. (2022). Condor: Self-supervised canonicalization of 3d pose for partial shapes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16969–16979.

Salimans, T., Karpathy, A., Chen, X., and Kingma, D. P. (2017). PixelCNN++: Improving the pixelCNN with discretized logistic mixture likelihood and other modifications. In *International Conference on Learning Representations*.

Salminen, J., Jung, S.-g., Chowdhury, S., and Jansen, B. J. (2020). Analyzing demographic bias in artificially generated facial pictures. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–8.

Samangouei, P., Kabkab, M., and Chellappa, R. (2018). Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In *International Conference on Learning Representations*.

Sasaki, H., Willcocks, C. G., and Breckon, T. P. (2021). Unit-ddpm: Unpaired image translation with denoising diffusion probabilistic models. *arXiv preprint arXiv:2104.05358*.

Satorras, V. G., Hoogeboom, E., and Welling, M. (2021). E(n) equivariant graph neural networks. In *International Conference on Machine Learning*, volume 139, pages 9323–9332. PMLR.

Sauer, A., Schwarz, K., and Geiger, A. (2022). Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10.

Schedl, D. C., Birklbauer, C., and Bimber, O. (2015). Directional super-resolution by means of coded sampling and guided upsampling. In *2015 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10. IEEE.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. (2018). Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31.

Schuhmann, C., Kaczmarczyk, R., Komatsuzaki, A., Katta, A., Vencu, R., Beaumont, R., Jitsev, J., Coombes, T., and Mullis, C. (2021). Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *NeurIPS Workshop Datacentric AI*. Jülich Supercomputing Center.

Schuler, C. J., Hirsch, M., Harmeling, S., and Schölkopf, B. (2015). Learning to deblur. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1439–1451.

Schwab, J., Antholzer, S., and Haltmeier, M. (2018). Deep null space learning for inverse problems: convergence analysis and rates. *Inverse Problems*, 35.

Schwab, J., Antholzer, S., and Haltmeier, M. (2019). Deep null space learning for inverse problems: convergence analysis and rates. *Inverse Problems*, 35(2):025008.

Shah, V. and Hegde, C. (2018). Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4609–4613. IEEE.

Shao, R., Shi, Z., Yi, J., Chen, P.-Y., and Hsieh, C.-J. (2022). On the adversarial robustness of vision transformers. *Transactions on Machine Learning Research*.

Sharif, M., Bhagavatula, S., Bauer, L., and Reiter, M. K. (2016). Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *ACM SIGSAC conference on computer and communications security*, pages 1528–1540.

Sharp, N., Attaiki, S., Crane, K., and Ovsjanikov, M. (2020). Diffusion is all you need for learning on surfaces. *CoRR*, abs/2012.00888.

Shen, Y., Gu, J., Tang, X., and Zhou, B. (2020). Interpreting the latent space of GANs for semantic face editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9243–9252.

Sheng, Y. and Shen, L. (1994). Orthogonal fourier–mellin moments for invariant pattern recognition. *Journal of the Optical Society of America*, 11(6):1748–1757.

Shi, L., Hassanieh, H., Davis, A., Katabi, D., and Durand, F. (2014). Light field reconstruction using sparsity in the continuous fourier domain. *ACM Transactions on Graphics (TOG)*, 34(1):12.

Sidky, E. Y., Kao, C.-M., and Pan, X. (2006). Accurate image reconstruction from few-views and limited-angle data in divergent-beam ct. *Journal of X-ray Science and Technology*, 14(2):119–139.

Sifre, L. and Mallat, S. (2013). Rotation, scaling and deformation invariant scattering for texture discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Simard, P., Victorri, B., LeCun, Y., and Denker, J. (1991). Tangent prop-a formalism for specifying selected invariances in an adaptive network. In *Advances in neural information processing systems*, volume 4.

Slovis, T. L. (2002). The alara concept in pediatric ct: myth or reality? *Radiology*, 223(1):5–6.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR.

Sommerhoff, H., Kolb, A., and Moeller, M. (2019). Energy dissipation with plug-and-play priors. In *NeurIPS 2019 Workshop on Solving Inverse Problems with Deep Networks*.

Song, J., Meng, C., and Ermon, S. (2021a). Denoising diffusion implicit models. In *International Conference on Learning Representations*.

Song, J., Vahdat, A., Mardani, M., and Kautz, J. (2023). Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*.

Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Song, Y., Shen, L., Xing, L., and Ermon, S. (2022). Solving inverse problems in medical imaging with score-based generative models. In *International Conference on Learning Representations*.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021b). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

Spezialetti, R., Stella, F., Marcon, M., Silva, L., Salti, S., and Di Stefano, L. (2020). Learning to orient surfaces by self-supervised spherical cnns. *Advances in Neural information processing systems*, 33.

Srinivasan, P. P., Tucker, R., Barron, J. T., Ramamoorthi, R., Ng, R., and Snavely, N. (2019). Pushing the boundaries of view extrapolation with multiplane images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 175–184.

Srinivasan, P. P., Wang, T., Sreelal, A., Ramamoorthi, R., and Ng, R. (2017). Learning to synthesize a 4d rgbd light field from a single image. In *IEEE International Conference on Computer Vision*, pages 2243–2251.

Starck, J.-L., Candès, E. J., and Donoho, D. L. (2002). The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11:670–684.

Su, J., Vargas, D. V., and Sakurai, K. (2017). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23:828–841.

Su, J., Xu, B., and Yin, H. (2022). A survey of deep learning approaches to image restoration. *Neurocomputing*, 487:46–65.

Su, X., Song, J., Meng, C., and Ermon, S. (2023). Dual diffusion implicit bridges for image-to-image translation. In *International Conference on Learning Representations*.

Sun, B., Tsai, N.-h., Liu, F., Yu, R., and Su, H. (2019). Adversarial defense by stratified convolutional sparse coding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11447–11456.

Sun, J., Li, H., Xu, Z., et al. (2016). Deep admm-net for compressive sensing mri. *Advances in neural information processing systems*, 29.

Sun, L., Cho, S., Wang, J., and Hays, J. (2013). Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography*.

Sun, L., Sakaridis, C., Liang, J., Jiang, Q., Yang, K., Sun, P., Ye, Y., Wang, K., and Gool, L. V. (2022). Event-based fusion for motion deblurring with cross-modal attention. In *European Conference on Computer Vision*, pages 412–428. Springer.

Sun, W., Tagliasacchi, A., Deng, B., Sabour, S., Yazdani, S., Hinton, G. E., and Yi, K. M. (2021). Canonical capsules: Self-supervised capsules in canonical pose. *Advances in Neural information processing systems*, 34.

Sunder Raj, A., Lowney, M., Shah, R., and Wetzstein, G. (2016). Stanford lytro light field archive.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. In *International Conference on Learning Representations*.

Tai, K. S., Bailis, P., and Valiant, G. (2019). Equivariant transformer networks. In *International Conference on Machine Learning*, pages 6086–6095. PMLR.

Tan, T. (1998). Rotation invariant texture features and their use in automatic script identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(7):751–756.

Tang, Z., Gu, S., Bao, J., Chen, D., and Wen, F. (2022). Improved vector quantized diffusion models. *arXiv preprint arXiv:2205.16007*.

Tensmeyer, C. and Martinez, T. (2016). Improving invariance and equivariance properties of convolutional neural networks.

Terris, M., Repetti, A., Pesquet, J.-C., and Wiaux, Y. (2021). Enhanced convergent pnp algorithms for image restoration. In *IEEE International Conference on Image Processing (ICIP)*, pages 1684–1688.

Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. (2018). Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*.

Tikhonov, A. N. (1963). Solution of incorrectly formulated problems and the regularization method. In *Soviet Math. Dokl.*, volume 5, pages 1035–1038.

Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. (2018). Wasserstein auto-encoders. In *International Conference on Learning Representations*.

Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al. (2021). Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272.

Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., and Cohen-Or, D. (2021). Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14.

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*.

Tsai, F.-J., Peng, Y.-T., Lin, Y.-Y., Tsai, C.-C., and Lin, C.-W. (2022). Stripformer: Strip transformer for fast image deblurring. In *European Conference on Computer Vision*, pages 146–162. Springer.

Tschandl, P., Rosendahl, C., and Kittler, H. (2018). The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1):1–9.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. (2019). Robustness may be at odds with accuracy. In *International Conference on Learning Representations*.

Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., and Li, Y. (2022). Maxim: Multi-axis mlp for image processing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5769–5780.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). Deep image prior. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454.

Vadathya, A. K., Girish, S., and Mitra, K. (2019). A unified learning based framework for light field reconstruction from coded projections. *IEEE Transactions on Computational Imaging*.

Vagharshakyan, S., Bregovic, R., and Gotchev, A. (2018). Light field reconstruction using shearlet transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):133–147.

Vahdat, A., Kreis, K., and Kautz, J. (2021). Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302.

Valkonen, T. (2021). First-order primal–dual methods for nonsmooth non-convex optimisation. *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging: Mathematical Imaging and Vision*, pages 1–42.

Vasu, S., Maligireddy, V. R., and Rajagopalan, A. (2018). Non-blind deblurring: Handling kernel uncertainty with cnns. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Veeling, B. S., Linmans, J., Winkens, J., Cohen, T., and Welling, M. (2018). Rotation equivariant cnns for digital pathology. In *International Conference on Medical image computing and computer-assisted intervention*, pages 210–218. Springer.

Veen, D. V., Jalal, A., Soltanolkotabi, M., Price, E., Vishwanath, S., and Dimakis, A. G. (2020). Compressed sensing with deep image prior and learned regularization.

Veeraraghavan, A., Raskar, R., Agrawal, A., Mohan, A., and Tumblin, J. (2007). Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. *ACM TOG*, 26(3):69.

Venkatakrishnan, S. V., Bouman, C. A., and Wohlberg, B. (2013). Plug-and-play priors for model based reconstruction. In *2013IEEEGlobal Conference on Signal and Information Processing*, pages 945–948. IEEE.

Villani, C. (2021). *Topics in optimal transportation*, volume 58. American Mathematical Society.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *25th international conference on Machine learning*, pages 1096–1103.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12).

Virmaux, A. and Scaman, K. (2018). Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems*, 31.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The caltech-ucsd birds-200-2011 dataset.

Wald, A. (1945). Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, pages 265–280.

Wallace, B., Gokul, A., and Naik, N. (2023). Edict: Exact diffusion inversion via coupled transformations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22532–22541.

Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., and Guibas, L. J. (2019). Normalized object coordinate space for category-level 6d object pose and size estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651.

Wang, P.-S., Liu, Y., Guo, Y.-X., Sun, C.-Y., and Tong, X. (2017). O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis. *ACM Transactions on Graphics (SIGGRAPH)*, 36(4).

Wang, R., Yang, Y., and Tao, D. (2022a). Art-point: Improving rotation robustness of point cloud classifiers via adversarial rotation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14371–14380.

Wang, T., Zhang, Y., Fan, Y., Wang, J., and Chen, Q. (2022b). High-fidelity gan inversion for image attribute editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Wang, W., Dai, J., Chen, Z., Huang, Z., Li, Z., Zhu, X., Hu, X., Lu, T., Lu, L., Li, H., et al. (2023a). Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14408–14419.

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., and Change Loy, C. (2018a). Esrgan: Enhanced super-resolution generative adversarial networks. In *European Conference on Computer Vision Workshops*, pages 0–0.

Wang, Y., Liu, F., Wang, Z., Hou, G., Sun, Z., and Tan, T. (2018b). End-to-end view synthesis for light field imaging with pseudo 4dcnn. In *European Conference on Computer Vision*, pages 333–348.

Wang, Y., Liu, Y., Heidrich, W., and Dai, Q. (2016). The light field attachment: Turning a dslr into a light field camera using a low budget camera ring. *IEEE transactions on visualization and computer graphics*, 23(10):2357–2364.

Wang, Y., Yu, J., and Zhang, J. (2023b). Zero-shot image restoration using denoising diffusion null-space model. In *International Conference on Learning Representations*.

Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., and Li, H. (2022c). Uformer: A general u-shaped transformer for image restoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17683–17693.

Wang, Z., Pang, T., Du, C., Lin, M., Liu, W., and Yan, S. (2023c). Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638*.

Wanner, S. and Goldluecke, B. (2013). Variational light field analysis for disparity estimation and super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):606–619.

Weiler, M. and Cesa, G. (2019). General E(2)-Equivariant Steerable CNNs. In *Advances in Neural information processing systems*.

Weiler, M., Geiger, M., Welling, M., Boomsma, W., and Cohen, T. (2018a). 3d steerable cnns: Learning rotationally equivariant features in volumetric data. In *Advances in Neural information processing systems*.

Weiler, M., Geiger, M., Welling, M., Boomsma, W., and Cohen, T. (2018b). 3d steerable cnns: learning rotationally equivariant features in volumetric data. In *Advances in Neural information processing systems*, pages 10402–10413.

Weng, L., Zhang, H., Chen, H., Song, Z., Hsieh, C.-J., Daniel, L., Boning, D., and Dhillon, I. (2018). Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning*, pages 5276–5285. PMLR.

Whang, J., Delbracio, M., Talebi, H., Saharia, C., Dimakis, A. G., and Milanfar, P. (2022). Deblurring via stochastic refinement. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16293–16303.

Whang, J., Lei, Q., and Dimakis, A. (2021). Solving inverse problems with a flow-based noise model. In *38th International Conference on Machine Learning*, volume 139, pages 11146–11157. PMLR.

Wilburn, B., Joshi, N., Vaish, V., Talvala, E.-V., Antunez, E., Barth, A., Adams, A., Horowitz, M., and Levoy, M. (2005). High performance imaging using large camera arrays. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, page 765–776, New York, NY, USA. Association for Computing Machinery.

Wilk, M. v. d., Bauer, M., John, S., and Hensman, J. (2018). Learning invariances using the marginal likelihood. In *Advances in Neural information processing systems*, pages 9960–9970.

Wong, E. and Kolter, Z. (2018). Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International conference on machine learning*, pages 5286–5295. PMLR.

Wong, E., Rice, L., and Kolter, J. Z. (2020). Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*.

Wong, E., Schmidt, F., and Kolter, Z. (2019). Wasserstein adversarial examples via projected sinkhorn iterations. In *International Conference on Machine Learning*, pages 6808–6817. PMLR.

Worrall, D. E., Garbin, S. J., Turmukhambetov, D., and Brostow, G. J. (2017). Harmonic networks: Deep translation and rotation equivariance. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wu, G., Liu, Y., Dai, Q., and Chai, T. (2019a). Learning sheared epi structure for light field reconstruction. *IEEE Transactions on Image Processing*, 28:3261–3273.

Wu, G., Liu, Y., Fang, L., Dai, Q., and Chai, T. (2019b). Light field reconstruction using convolutional network on epi and extended applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1681–1694.

Wu, K., Wang, A., and Yu, Y. (2020a). Stronger and faster wasserstein adversarial attacks. In *International Conference on Machine Learning*, volume 119, pages 10377–10387. PMLR.

Wu, W., Hu, D., Cong, W., Shan, H., Wang, S., Niu, C., Yan, P., Yu, H., Vardhanabhuti, V., and Wang, G. (2022). Stabilizing deep tomographic reconstruction: Part b. convergence analysis and adversarial attacks. *Patterns*, 3(5):100475.

Wu, Z., Lim, S.-N., Davis, L. S., and Goldstein, T. (2020b). Making an invisibility cloak: Real world adversarial attacks on object detectors. In *16th European Conference on Computer Vision*. Springer.

Wu, Z., Lischinski, D., and Shechtman, E. (2021). Stylespace analysis: Disentangled controls for stylegan image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12863–12872.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Xia, W., Yang, Y., Xue, J.-H., and Wu, B. (2021). Tedigan: Text-guided diverse face image generation and manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2256–2265.

Xiao, C., Zhu, J.-Y., Li, B., He, W., Liu, M., and Song, D. (2018). Spatially transformed adversarial examples. In *International Conference on Learning Representations*.

Xiao, Z., Lin, H., Li, R., Geng, L., Chao, H., and Ding, S. (2020). Endowing deep 3d models with rotation invariance based on principal component analysis. In *IEEE International Conference on Multimedia and Expo (ICME)*. IEEE.

Xie, C., Tan, M., Gong, B., Yuille, A., and Le, Q. V. (2020). Smooth adversarial training. *arXiv preprint arXiv:2006.14536*.

Xie, C., Wang, J., Zhang, Z., Ren, Z., and Yuille, A. (2018). Mitigating adversarial effects through randomization. In *International Conference on Learning Representations*.

Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., and Yuille, A. (2017). Adversarial examples for semantic segmentation and object detection. In *International Conference on Computer Vision*. IEEE.

Xie, J., Xu, L., and Chen, E. (2012). Image denoising and inpainting with deep neural networks. In *Advances in Neural Information Processing Systems*, pages 341–349.

Xu, L., Zheng, S., and Jia, J. (2013). Unnatural l0 sparse representation for natural image deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1107–1114.

Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., and He, X. (2018). AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1316–1324.

Xu, W., Evans, D., and Qi, Y. (2017). Feature squeezing: Detecting adversarial examples in deep neural networks. In *Machine Learning and Computer Security Workshop, Advances in Neural Information Processing Systems*.

Yadav, A., Shah, S., Xu, Z., Jacobs, D., and Goldstein, T. (2018). Stabilizing adversarial nets with prediction methods. In *International Conference on Learning Representations*.

Yakura, H., Akimoto, Y., and Sakuma, J. (2020). Generate (non-software) bugs to fool classifiers. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 1070–1078.

Yan, X. (2019). Pointnet/pointnet++ pytorch.

Yang, F., Wang, Z., and Heinze-Deml, C. (2019). Invariance-inducing regularization using worst-case transformations suffices to boost accuracy and spatial robustness. *Advances in Neural information processing systems*, pages 14757–14768.

Yang, Q., Yan, P., Zhang, Y., Yu, H., Shi, Y., Mou, X., Kalra, M. K., Zhang, Y., Sun, L., and Wang, G. (2018). Low-dose ct image denoising using a generative adversarial network with wasserstein distance and perceptual loss. *IEEE Transactions on Medical Imaging*, 37(6):1348–1357.

Yap, P.-T., Jiang, X., and Chichung Kot, A. (2010). Two-dimensional polar harmonic transforms for invariant image representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1259–1270.

Yeung, H. W. F., Hou, J., Chen, J., Chung, Y. Y., and Chen, X. (2018). Fast light field reconstruction with deep coarse-to-fine modeling of spatial-angular clues. In *European Conference on Computer Vision*, pages 137–152.

Yu, R., Wei, X., Tombari, F., and Sun, J. (2020). Deep positional and relational feature learning for rotation-invariant point cloud analysis. In *European Conference on Computer Vision*.

Yu, Y., Zhan, F., Wu, R., Zhang, J., Lu, S., Cui, M., Xie, X., Hua, X.-S., and Miao, C. (2022). Towards counterfactual image manipulation via clip. In *30th ACM International Conference on Multimedia*.

Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., and Yang, M.-H. (2022). Restormer: Efficient transformer for high-resolution image restoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., Yang, M.-H., and Shao, L. (2021). Multi-stage progressive image restoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zhang, H., Koh, J. Y., Baldridge, J., Lee, H., and Yang, Y. (2021a). Cross-modal contrastive learning for text-to-image generation. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. (2017a). StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision*, pages 5908–5916.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. (2018a). Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1947–1962.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. (2019a). Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR.

Zhang, J. and Ghanem, B. (2018). Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Zhang, J., Yu, M.-Y., Vasudevan, R., and Johnson-Roberson, M. (2020). Learning rotation-invariant representations of point clouds using aligned edge convolutional neural networks. In *2020 International Conference on 3D Vision (3DV)*, pages 200–209. IEEE.

Zhang, K., Li, Y., Zuo, W., Zhang, L., Van Gool, L., and Timofte, R. (2021b). Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017b). Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*.

Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017c). Learning deep CNN denoiser prior for image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Zhang, L., Qi, G.-J., Wang, L., and Luo, J. (2019b). Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2547–2555.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018b). The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Zhang, Y. and Rabbat, M. (2018). A graph-cnn for 3d point cloud classification. In *International Conference on Acoustics, Speech and Signal Processing*.

Zhang, Y., Zhang, G., Khanduri, P., Hong, M., Chang, S., and Liu, S. (2022). Revisiting and advancing fast adversarial training through the lens of bi-level optimization. In *39th International Conference on Machine Learning*, pages 26693–26712. PMLR.

Zhang, Z., Liang, X., Dong, X., Xie, Y., and Cao, G. (2018c). A sparse-view ct reconstruction method based on combination of densenet and deconvolution. *IEEE Transactions on Medical Imaging*, 37(6):1407–1417.

Zhao, H., Gallo, O., Frosio, I., and Kautz, J. (2016). Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57.

Zhao, Y., Birdal, T., Deng, H., and Tombari, F. (2019). 3d point capsule networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhao, Y., Wu, R., and Dong, H. (2020a). Unpaired image-to-image translation using adversarial consistency loss. In *European Conference on Computer Vision*, pages 800–815. Springer.

Zhao, Y., Wu, Y., Chen, C., and Lim, A. (2020b). On isometry robustness of deep 3d point cloud models under adversarial attacks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zheng, T., Chen, C., and Ren, K. (2019). Distributionally adversarial attack. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 2253–2260.

Zhu, J., Gao, L., Song, J., Li, Y.-F., Zheng, F., Li, X., and Shen, H. T. (2022). Label-guided generative adversarial network for realistic image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhu, J., Shen, Y., Zhao, D., and Zhou, B. (2020). In-domain gan inversion for real image editing. In *European Conference on Computer Vision*, pages 592–608, Berlin, Heidelberg. Springer-Verlag.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *European conference on computer vision*, pages 597–613. Springer.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pages 2223–2232.

Zhu, M., Pan, P., Chen, W., and Yang, Y. (2019). Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5802–5810.

Zhu, Y., Zhang, K., Liang, J., Cao, J., Wen, B., Timofte, R., and Gool, L. V. (2023). Denoising diffusion models for plug-and-play image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (NTIRE)*.

Zontak, M. and Irani, M. (2011). Internal statistics of a single natural image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 977–984, Los Alamitos, CA, USA. IEEE Computer Society.

Zoran, D. and Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In *International Conference on Computer Vision*, pages 479–486. IEEE.