SPATIALLY VARYING SINGLE IMAGE DEBLURRING USING CYCLEGANS

by

Gizem Esra Ünlü

B.S., Computer Engineering, Istanbul Technical University, 2017

B.S., Mathematics Engineering, Istanbul Technical University, 2017

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2019

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor Prof. Lale Akarun who has given me the opportunity to pursue masters education, for her expertise, guidance, and assistance during my studies. I would also like to thank Prof. Hasan Fehmi Ateş and Fatma Başak Aydemir for their valuable contribution and comments on my thesis.

I owe my deepest gratitude to my family for their endless love and support throughout my life. Without them, I wouldn't be the person I am today. I would especially like to thank my dear friend Deniz Engin for motivating me during hard moments and for being the voice of reason when I needed.

I would like to thank my friends and colleagues in Boğaziçi University and Perceptual Intelligence Laboratory for their endless support and valuable friendship, namely, Ahmet Alp Kındıroğlu, Alptekin Orbay, Çağrı Gökçe, Doğa Siyli, Mehmet Burak Kurutmaz, Oğulcan Özdemir, Ufuk Can Biçici, and Uras Mutlu. I would also like to thank my friend Alper Ahmetoğlu for sharing his knowledge tirelessly, and Erhan Çağrıcı for coming up with the most novel motivational notes.

ABSTRACT

SPATIALLY VARYING SINGLE IMAGE DEBLURRING USING CYCLEGANS

In this thesis, we propose a novel method that learns to deblur hand images in the presence of spatially-varying object motion blur and unpaired blurry/sharp training pairs. While some success has been achieved in image deblurring by learning disentangled representations from synthetically blurred data, these methods do not perform well when objects in the frame are moving rapidly; consequently resulting in inferior pose estimation performances. This commonly occurs when the hands of a signer moves abruptly in a sign language setting. We propose to solve these problems by disentangling blur information from image content (hand texture, background). Lack of non-corresponding training pairs is dealt with cross-cycle consistency losses in blurring/deblurring branches based on disentangled representations and spatially-variant blur is extracted from blur-degraded regions using partial convolutions. We test our results both qualitatively and quantitatively on a novel hand blur dataset consisting of real blurry images and sharp frames as well as a reference synthetically blurred dataset.

ÖZET

İMGELERDE UZAMSAL OLARAK DEĞİŞEN BULANIKLIĞIN ÇEVRİMSEL ÇEKİŞMELİ AĞLAR İLE GİDERİLMESİ

Bu tezde, uzamsal olarak değişken nesne hareket bulanıklığı ve eşleştirilmemiş bulanık/keskin eğitim çiftleri varlığında el görüntülerinden bulanıklığı kaldırmayı öğrenen yeni bir yöntem öneriyoruz. Sentetik bulanıklık eklenmiş verilerden ayrık gösterimleri öğrenerek görüntüde bulanıklığı kaldırma konusunda bir miktar başarı elde edilmiş olsa da, bu yöntemler hızla hareket eden nesneler var ise iyi performans göstermez; bu sebeple düşük poz tahmin performanslarının ortaya çıkması kaçınılmaz olur. Bu durum çoğu zaman işaret dili konuşan bir kişinin ellerini ani hareket ettirmesiyle meydana gelir. Bu sorunları imge içeriğinden (el dokusu, arka plan) bulanıklık bilgilerini kaldırarak çözmeyi öneriyoruz. Birbirlerine eş gelen eğitim çiftlerinin olmayışı, ayrık gösterimlere dayanarak bulanıklaştıran /bulanıklık kaldıran paralel ağlarla ve çapraz döngü tutarlılık kayıpları ile çözümlendirilmiş ve kısmi evrişimli ağ kullanılarak bulanık bölgelerden uzamsal değişken bilgi öğrenilmiştir. Sonuçlarımızı hem niteliksel hem de niceliksel olarak, gerçek bulanık görüntülerden ve keskin imgelerden oluşan bir veri kümesi ve sentetik bulanıklık içeren bir veri kümesi üzerinden sunuyoruz.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES viii
LIST OF TABLES xii
LIST OF SYMBOLS xiii
LIST OF ACRONYMS/ABBREVIATIONS xvi
1. INTRODUCTION 1
1.1. Motivation and Related Work 1
1.2. Contributions 3
1.3. Organization of the Thesis 4
2. GENERATIVE ADVERSARIAL NETWORKS 5
2.1. Generative Adversarial Network Foundations 5
2.2. Common Problems in GAN Training 7
2.3. A Stable Architecture: DCGAN 9
3. SINGLE IMAGE DEBLURRING 11
3.1. Classical Methods 11
3.1.1. Non-blind Deblurring 12
3.1.2. Blind Deblurring 14
3.2. Deblurring Methods for Object Motion 16
3.3. Learning Based Methods 18
4. SPATIALLY VARYING MOTION BLUR ELIMINATION 23
4.1. Disentangling Content and Spatially Variant Object Motion Blur . . . 23
4.2. Cross-Cycle Consistency Loss 25
4.3. Domain Adversarial Loss for HospiSign 27
4.4. Domain Adversarial Loss For CelebA 27
4.5. Perceptual Loss 28
4.6. Learning Objective 29

4.7. Testing . . . 29
4.8. Architecture of Blur Encoder with Partial Convolutions . . . 29
5. EXPERIMENTS AND RESULTS . . . 31
5.1. Datasets . . . 31
5.1.1. HospiSign . . . 31
5.1.1.1. Dataset Preparation . . . 31
5.1.2. CelebA . . . 32
5.2. Evaluation Metrics . . . 32
5.2.1. Inception Score . . . 32
5.2.2. Frechet Inception Distance . . . 33
5.2.3. Kernel Inception Score . . . 34
5.2.4. Peak Signal to Noise Ratio . . . 35
5.2.5. Structural Similarity Index . . . 35
5.3. Experimental Results and Discussion . . . 36
5.3.1. Implementation Details . . . 36
5.3.2. Compared Methods . . . 37
5.3.3. Results for CelebA . . . 37
5.3.4. Results for HospiSign . . . 37
5.3.5. Failure Cases . . . 41
5.3.5.1. CelebA . . . 41
5.3.5.2. Hand . . . 42
5.3.6. Additional Results . . . 42
6. CONCLUSION AND FUTURE WORK . . . 45
REFERENCES . . . 47

LIST OF FIGURES

Figure 1.1. Types of blur. (a)Sample image from blur detection dataset of [1]. (b) and (c) belong to [2] 2

Figure 1.2. Samples from the proposed real hand blur dataset 3

Figure 2.1. Overview of a Generative Adversarial Network for natural image generation. Two CNN-based opponents called generator and discriminator play an adversarial game, generating fake samples and distinguishing from real samples, respectively. 5

Figure 2.2. GAN training algorithm. [3] 7

Figure 2.3. DCGAN [4] generator architecture. Image taken from [4]. 10

Figure 2.4. DCGAN [4] disriminator architecture. Image taken from [4]. . . . 10

Figure 3.1. A deblurring example from CelebA [5] 11

Figure 3.2. A graphical explanation for non-blind deblurring (top row) and blind deblurring (bottom row). (Image taken from [6]'s deblurring dataset) 12

Figure 3.3. A graphical explanation for the ill-posed nature of deblurring.(Image taken from [6]'s deblurring dataset) 13

Figure 3.4. Non-blind deconvolution as an intermediate step in blind deblurring.(Image taken from [6]'s deblurring dataset) 14

Figure 3.5. Bright/saturated regions throughout blurry images can cause severe ringing artifacts after deblurring. Left column illustrates blurry images while the right column shows deblurring result. Areas with ringing artifacts have been magnified for better visualization.(Image taken from [6]'s deblurring dataset) 15

Figure 3.6. An example of synthetic motion blur on a text image. Different spatial locations are degraded by unidentical blurs. Left image shows sharp image and left image is synthetically blurred image.(Image taken from [6]'s deblurring dataset) 17

Figure 3.7. Object motion blur from our proposed dataset. The background contains no blur((b)-(c)). Blur degradation is around the hand area(a). 19

Figure 3.8. An Overview of [7]'s method for partial defocus/object motion blur. The task is split in to two succesive steps: foreground and background segmentation(matting) and blind deblurring on blurry parts. During the second step, blur kernel estimation and blur elimination is applied on both the background and foreground. (Image taken from [7]) 19

Figure 3.9. Learning-based deblurring methods require large amounts of data: earlier methods use corresponding sharp-blurry image pairs for learning, however, recent approaches can learn from uncorresponding data which enabled working on real blurry datasets as opposed to synthetically created datasets.(Image reprinted from [8]) 20

Figure 3.10. Overview of CycleGAN [9] architecture applied to deblurring. (a)Domain generators $G_{S \to B}$ and $G_{B \to S}$ learn mapping between sharp and blurry image domains. (b)Domain discriminators D_S and D_B assess how well the generators translate images between domains. (c) Cycle consistency losses regularize mappings between domains. The idea behind this is that by mapping a sample from one domain and back to its original domain, we want to arrive at the same point. 21

Figure 3.11. [10]'s deblurring architecture visualized with our novel hand dataset. Training with unpaired images is achieved via blurring/deblurring branches and cross-cycle consistency losses. 22

Figure 4.1. Our proposed network. 24

Figure 4.2. A toy example for the partial convolution's unmasking rule. For this case, two consecutive convolution operations are applied on a given mask image with a 3x3 kernel, unit stride and no padding. During the first convolution the mask for the upper left corner is not unmasked as it was not conditioned on any valid pixels. The bottom right corner is masked because its feature map was conditioned on two valid pixels. 26

Figure 5.1. Some sharp hand samples from our novel deblurring dataset. The top row shows the frames where the sharp hand images in the bottom row were taken. 33

Figure 5.2. Blurry hand images (bottom row) from our proposed hand deblurring dataset. The top row illustrates the full frames of the sign language dataset our dataset was collected from. 34

Figure 5.3. Comparison of state-of-the-art unpaired deblurring methods on our the novel hand blur dataset. 39

Figure 5.4. Failure cases on partially-blurred CelebA test set. The middle column illustrates the deblurring results while the first and last columns are the corresponding blurred and sharp images, respectively. 41

Figure 5.5. Failure cases on the proposed novel hand test set. First and third columns contain blurry images while second and fourth columns have the corresponding deblurring result. 42

Figure 5.6. Qualitative test results on novel blur dataset. The first column shows the mask for the hand area. Best viewed in color by zooming in. 43

Figure 5.7. Some results from state-of-the-art works on partially blurred CelebA dataset. The first column shows the mask that was used to blur the facial region in each image. Best viewed in color by zooming in. 44

LIST OF TABLES

Table 4.1. Proposed blur encoder architecture with partial convolution layers. The input consists of an RGB image and a blur region mask. The network outputs are two 1x8 vectors for blur feature mean and variance. 30

Table 5.1. Statistical information from the novel blur dataset for training and testing divisions 32

Table 5.2. CelebA quantitative results for unsupervised metrics 38

Table 5.3. CelebA quantitative results for full reference metrics 38

Table 5.4. Hospisign quantitative results 40

Table 5.5. Hospisign ablation study on how the use of partial convolution in the blur encoder improves evaluation metrics. 40

LIST OF SYMBOLS

b	Blurry Image Sample
B	Blurry Image Domain
B	Sliding Window Feature Values
c	Weight Clipping Value
D	Discriminator
D_{blur}	Blurry Image Discriminator
D_B	Blurry Domain Discriminator
D_{sharp}	Sharp Image Discriminator
D_S	Sharp Domain Discriminator
D_{KL}	KL Divergence
$\mathcal{D}$	Set of Lipschitz-1 Functions
E_{blur}	Blur Encoder
E^B_{con}	Blurry Image Content Encoder
E^S_{con}	Sharp Image Content Encoder
$\mathbb{E}$	Expectation
G	Generator
G_{blur}	Blurry Image Generator
G_{sharp}	Sharp Image Generator
$G_{B\to S}$	Sharp Domain Generator
$G_{S\to B}$	Blurry Domain Generator
k	Blur Kernel
$k(x, y)$	Kernel Function Given x and y
$\hat{k}$	Blur Kernel Estimate
K	Number of Discriminator Updates
L	Full Objective Function
L_{adv}	Adversarial Loss
L_{cc}	Cross-Cycle Consistency Loss
L_{D_B}	Blurry Domain Discriminator Loss

L_{D_S}	Sharp Domain Discriminator Loss
L_p	Perceptual Loss
m	Number of Samples
M	Binary Mask
M_b	Blurry Image Mask
M_s	Sharp Image Mask
n	Image Noise
N	Total Number of Training Iterations
$p(a)$	Probability of Event a
$p(a\|b)$	Conditional Probability of a given b
$P(s)$	Sharp Image Prior
$\mathbb{P}_{data}$	Data Generating Distribution
$\mathbb{P}_G$	Fake Data Distribution
$\mathbb{P}_z$	Prior Noise Distribution
$q(s)$	Variational Distribution
s	Sharp Image Sample
S	Sharp Image Domain
Tr	Transpose
$\hat{s}$	Sharp Image Estimate
x	Image Sample
W	Convolution Weight Matrix
$\mathcal{W}$	Network Parameters
z	Noise Sample
z_b	Blur Feature Vector
θ_d	Discriminator Weights
θ_g	Generator Weights
λ	Data Term Weight Factor
λ_{adv}	Adversarial Loss Weight Factor
λ_{cc}	Cross-Cycle Consistency Loss Weight Factor
λ_{gp}	Gradient Penalty Weight Factor

λ_p	Perceptual Loss Weight Factor
μ	Mean
$\rho_k(k)$	Blur Kernel Regularization Term
$\rho_s(s)$	Sharp Image Regularization Term
Σ	Covariance
ϕ_l	Pre-trained Network Output
$*$	Convolution Operator
$\in$	Element of
$\| \|_1$	L_1 Norm
$\| \|_2$	Frobenius Norm
$\propto$	Proportional To
∇	Gradient Operator
$\odot$	Hadamard Product

LIST OF ACRONYMS/ABBREVIATIONS

BD	Blind Deblurring
CNN	Convolutional Neural Network
DCGAN	Deep Convolutional Generative Adversarial Networks
EM	Earth Mover
FID	Frechet Inception Score
GAN	Generative Adversarial Network
IS	Inception Score
KID	Kernel Inception Score
LRELU	Leaky Rectified Linear Unit
MAP	Maximum a Posteriori
MMD	Maximum Mean Discrepancy
M_PSNR	Mask Peak Signal to Noise Ratio
NBD	Non-Blind Deblurring
PSNR	Peak Signal to Noise Ratio
RELU	Rectified Linear Unit
RL	Richardson-Lucy
ROI	Region of Interest
SSIM	Structural Similarity Index
Tanh	Hyperbolic Tangent Function
VB	Variational Bayesian
WGAN	Wasserstein Generative Adversarial Network

1. INTRODUCTION

1.1. Motivation and Related Work

Image blur, a common undesirable phenomenon in digital photography, may be caused by several factors such as a non-focusing camera, fast objects in the scene, or simply camera motion. Figure 1.1 illustrates the degradation caused by different types blur: as each type produces such diverse artefacts, a universal deblurring method as the solution is not possible. This has led the research community to focus on approaches that are geared towards a single blur type. To overcome the effects of blur, numerous techniques in computer vision literature have been proposed which are generally referred to as *deblurring* or *deconvolution*.

The degradation caused by blur is often unwanted not only because the outcome is not visually pleasing, but also since blur negatively affects the performances of several consequent computer vision tasks, such as face recognition/verification, pose estimation, and object recognition/tracking [11]. [12] showed that the performance of a pretrained object detection network deteriorates in the presence of blur and conversely, deblurring increases recognition success. In [10], the effect of deblurring on facial images is examined. Face verification accuracies increase in both frontal-to-frontal and frontal-to-profile tests when input images are deblurred. Thus, considering these adverse consequences of blur in images, the amount of research produced over the years in deblurring literature is not surprising.

Recent advances in deep learning rely on the presence of large datasets. However, real-life blurry data collection on large-scale is rather costly. In order to supply the need for big image deblurring datasets, researchers steered towards creating synthetically blurred datasets for sharp-blurred image pairs necessary for learning. Unfortunately, these synthetic datasets fall short in simulating the complex nature of authentic blur [1].



(a) Camera Motion



(b) Moving Object Blur

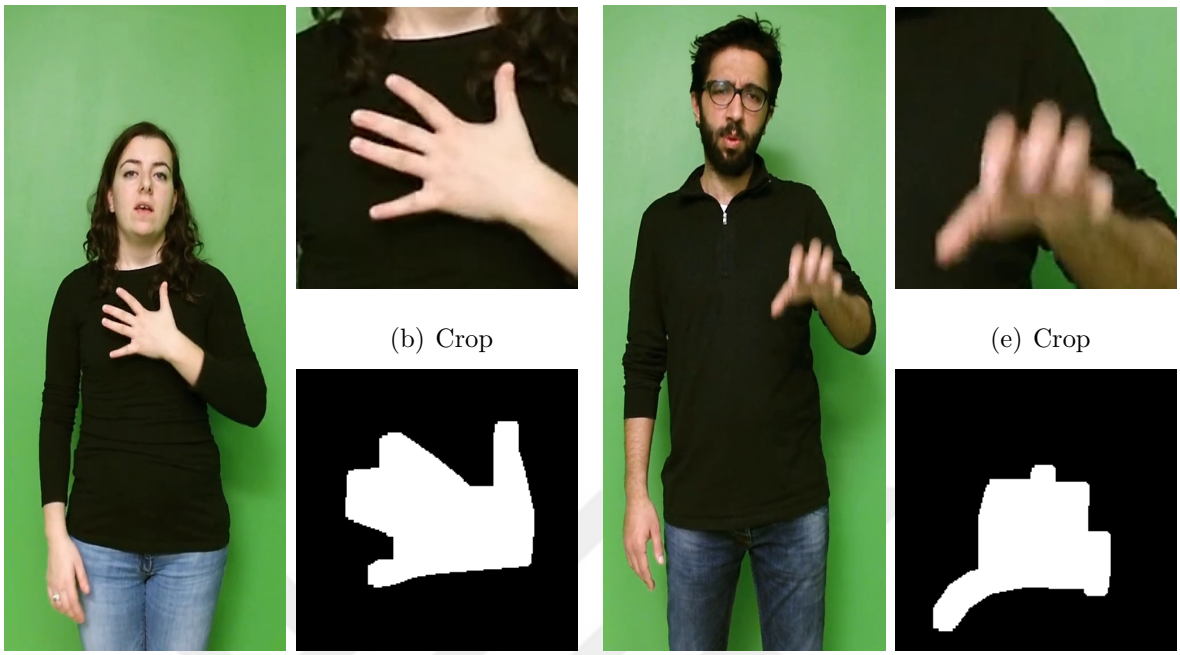


(c) Defocus Blur

Figure 1.1. Types of blur. (a)Sample image from blur detection dataset of [1]. (b) and (c) belong to [2]

Studies have shown that deep learning techniques supported by adversarial training [3] achieve state-of-the-art deblurring results on such man-made datasets. Yet, the trend is towards uniform camera motion blur that degrades the entire frame; no recent study focuses on the challenging case of spatially varying object motion blur where there is a blur-free static background and degradation is around the object. This type of localized blur can be observed in a sign language video where the person moves his/her hands rapidly and causes severe motion blur around this region see Figure 1.2). This presence of heavy blur on the hand region decreases Sign Language Recognition accuracy which is another negative outcome of blur [13].

Generative Adversarial Networks(GANs) [3] have positively affected numerous image restoration and enhancement tasks: realistic image generation, inpainting, super resolution, deblurring [14–16]. A leading work in single image deblurring with GANs is [17], which uses a multi-scale convolutional deblurring generator for recovering sharp



(a) Sharp frame (b) Crop (c) Mask (d) Blurry frame (e) Crop (f) Mask

Figure 1.2. Samples from the proposed real hand blur dataset

images from blurry ones and a discriminator that tries to distinguish deblurred images from their sharp counterparts. Building upon this, [12] approaches the idea more efficiently and reports results on a novel dataset that contains complex motion blurs caused by multiple sources. A more recent paper [10] learns to disentangle blur features from image contents in a blurry image, and solves the unpaired training data issue with a blurring branch and a cycle-consistency loss. However, these methods are not suitable for non-uniform blur and fail to produce desirable deblurring results, as can be expected.

1.2. Contributions

In this thesis, we have made the following contributions:

- We have collected a novel hand blur dataset consisting of real motion blur degraded samples for use in deep learning based deblurring research.
- We have formulated a novel weakly-supervised deblurring approach that can handle spatially varying blur when there are no corresponding sharp-blurry image

pairs.

- Experimental analysis of existing deblurring methods have been provided for spatially varying motion on both synthetic and real blur datasets.

We introduce a method that can learn to deblur images corrupted from a fast-moving object in a still background with non-corresponding sharp/blurry image pairs. Our method is built upon recent successful image-to-image translation approaches where the deblurring problem is posed as translating blurry images to a sharp image domain such as [15]. Spatially varying blur disentanglement from blurry images is achieved via a blur encoder with partial convolution layers [18] that attend to areas with blur degradation. The unpaired training data is handled by learning blurring/deblurring sub-networks using cross-cycle consistency losses Our method is tested firstly on CelebA [5] with motion blur applied to facial regions to imitate partial blur. Secondly, we present a novel hand blur dataset consisting of real blurry and sharp images that capture the complex nature of moving object blur (See Figure1.2). We show that our method achieves superior or comparable performance against state-of-the-art deblurring models using qualitative and quantitative evaluation metrics.

1.3. Organization of the Thesis

The organization of the rest of this theses is as follows: In Chapter 2, Generative Adversarial Network(GAN) foundations, various loss functions and regularization techniques as well as different architectures are explained in detail. The literature review for single image deblurring methods is showcased in Chapter 3. Following the related work, we give a detailed explanation of our spatially varying motion blur elimination method in Chapter 4. After the proposed method is described, experiments section is presented in Chapter 5 where we elaborate on the datasets used, experimental protocol and discussion of the results on several evaluation metrics used commonly in deblurring. Lastly, in Chapter 6, we share our closing thoughts on our work and make projections on possible directions spatially varying deblurring may take.

2. GENERATIVE ADVERSARIAL NETWORKS

Generative Adversarial Networks(GANs) [3] belong to a family of statistical models called generative methods that can generate new data instances from a probability distribution; in contrast to discriminative models which discriminate between the instances of a distribution. In this chapter we examine GANs [3] which are used frequently in recent work regarding image restoration tasks: inpainting [18], super-resolution [16], dehazing [19]. Image deblurring research has also benefited vastly from the introduction of adversarial training, inspiring a whole new branch under deep learning based blur elimination methods.

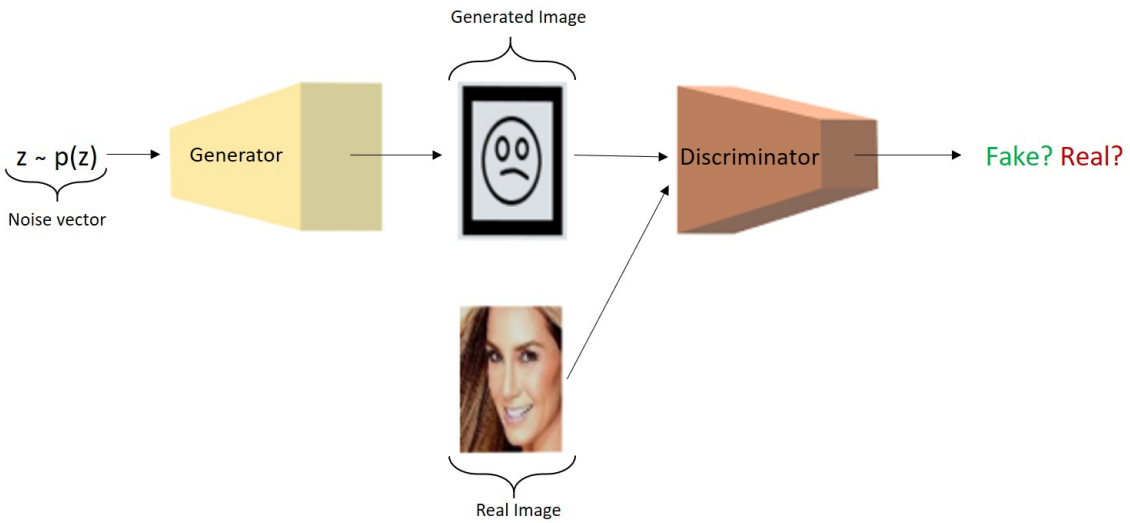


Figure 2.1. Overview of a Generative Adversarial Network for natural image generation. Two CNN-based opponents called generator and discriminator play an adversarial game, generating fake samples and distinguishing from real samples, respectively.

2.1. Generative Adversarial Network Foundations

Originally, GANs were developed for natural image generation, as the name suggests. This has applications in deblurring [10,20], inpainting [18], super-resolution [16], dehazing [19], security [21]. The idea behind GANs can be described as an adversarial game between two opponents: a generator and a discriminator. The generator pro-

duces fake images to fool the discriminator into believing that they are real and the discriminator conversely tries to distinguish between generated fake images and real images (See Figure2.1). More formally, let $x \sim \mathbb{P}_{data}(x)$ be an image coming from the real image data distribution and $z \sim \mathbb{P}_z(z)$ be a noise sample drawn from a prior distribution, typically chosen as normal distribution with zero mean and unit variance. Then, for generator G which maps a latent vector z to image space and discriminator D which is a scalar function that outputs the probability of a sample being fake, the value function for GAN is:

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_{data}(x)}[log(D(x))] + \mathop{\mathbb{E}}_{z\sim\mathbb{P}_z(z)}[log(1 - D(G(z)))] \tag{2.1}$$

In Eq. 2.1, the first term can be described as the log-probability of D predicting that a real image is genuine and the second term represents the log-probability of D determining if a sample generated by G is fake.

GANs have brought innovation and lots of benefits to the realistic data generation field. However, they are undeniably difficult to train. The GAN architecture consists of two sub-networks with different objectives. Thus, the training process should address this duality, which is a source of complexity. During training, the generator and discriminator updates are done alternately: the discriminator trains for one or more epochs; then the generator trains similarly and this process continues in the same manner until a stopping criterion is reached. The algorithm for this process as given in the original GAN paper [3] can be found in Figure 2.2.

The outer loop in Figure 2.2 determines the total number of iterations to be completed during learning. For each iteration, the second loop represents the discriminator training scheme where multiple updates are performed; K discriminator updates per generator update, precisely. Regular minibatch practice is applied for efficiency and stability by sampling m noise samples from the prior distribution and m samples from the real image distribution. After this, the update for the generator with respect to its gradient is performed by taking a minibatch of m noise samples from the prior

for $n = 1$ to N **do**

  for $k = 1$ to K **do**

    Sample minibatch of m noise samples $(z^{(1)}, ..., z^{(m)})$ from prior $\mathbb{P}_z(z)$;

    Sample minibatch of m data samples $(x^{(1)}, ..., x^{(m)})$ from data distribution $\mathbb{P}_{data}(x)$;

    Update discriminator with stochastic gradient ascent;

    $\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \Big[logD\Big(x^{(i)}\Big) + log\Big(1 - D\Big(G\Big(z^{(i)}\Big)\Big)\Big)\Big]$;

  end for

  Sample minibatch of m noise samples $z^{(1)}, ..., z^{(m)}$ from prior $\mathbb{P}_z(z)$;

  Update generator with stochastic gradient descent;

  $\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} log\Big(1 - D\Big(G\Big(z^{(i)}\Big)\Big)\Big)$;

end for

Figure 2.2. GAN training algorithm. [3]

distribution. It is important to emphasize that during discriminator training, the generator's weights are kept frozen and the discriminator, likewise. remains static during the generator phase.

2.2. Common Problems in GAN Training

An important aspect of GAN training is the instability problem. GANs suffer from a number of issues that compromise training: vanishing gradients, mode collapse, non-convergence. Vanishing gradients phenomenon is commonly observed at the beginning of training when discriminator outperforms generator by such a large margin that the generator fails to receive any feedback; hence, the name. Second failure case is mode collapse: the generator learns to map all input noise samples to one plausible output such that the discriminator is fooled but the generated images have minuscule variance defeats the purpose of the task at hand. The last mode of failure is the network not converging at all which can be intuitively explained by the nature of the problem: In an adversarial game where the opponents always take countermeasures,

convergence is hard and unstable.

To overcome all these challenges regarding instability, various novel loss functions, network architectures, and regularization techniques have been proposed. Particularly, since the publication of the original GAN paper [3], the research was particularly focused in coming up with novel losses. Even in [3], a modification to the proposed minimax loss is advised for the vanishing gradients problem:

$$\max_{G} \mathop{\mathbb{E}}_{z \sim \mathbb{P}_z(z)} [log(D(G(z)))] \tag{2.2}$$

where instead of minimizing the log-probability of the discriminator being correct, the generator now maximizes the log-probability of discriminator being mistaken. As it is stated in [3], this modification is purely heuristically motivated and does not have have any theoretical foundations. The discriminator loss remains the same as Eq. 2.1.

Another frequently used adversarial loss is based on Wasserstein-1 or Earth-Mover(EM) distance. Introduced as an extension to the regular GAN, Wasserstein GAN(WGAN) [22] minimizes the distance between the real data distribution and the generated fake distribution by using a practical approximation to the EM distance. They theoretically prove that the optimization of Wasserstein loss is valid with an interpretable metric in terms of generated image quality and the common problems of GANs such as mode collapse and divergence are alleviated. The formal definition of Wasserstein loss is as follows:

$$\min_{G} \max_{D \in \mathcal{D}} \mathop{\mathbb{E}}_{x \sim \mathbb{P}_{data}(x)} [D(x)] - \mathop{\mathbb{E}}_{z \sim \mathbb{P}_z(z)} [D(G(z))] \tag{2.3}$$

where $\mathcal{D}$ is the set of Lipschitz-1 and $D(x)$ is referred to as a *critic* rather than a discriminator since it outputs a value regarding the input image's authenticity as opposed to a probability of discriminating the image as fake or real. The critic function approximates the Wasserstein distance between $\mathbb{P}_{data}$ and $\mathbb{P}_G$ which is the distribution of generated images, up to a multiplicative constant.

To ensure the Lipschitz constraint on the critic function D which is approximated with a CNN for the image generation task, weight clipping on the network's parameters $\mathcal{W}$ is performed, e.g. $\mathcal{W} = [-c, c]$. However, this approach has obvious repercussions: setting c to a value too large will prolong convergence time and too high a value risks the occurrence of vanishing gradients problem. [23] proposes a gradient penalty as a different way of ensuring this Lipschitz constraint:

$$\mathop{\mathbb{E}}_{\widehat{x}\sim\mathbb{P}_{\widehat{x}}}[(\|\nabla_{\widehat{x}}D(\widehat{x})\|_2 - 1)^2] \tag{2.4}$$

where $\widehat{x} \sim \mathbb{P}_{\widehat{x}}$ are sampled uniformly along straight lines between the real data distribution and the generated data distribution.

2.3. A Stable Architecture: DCGAN

In the original GAN paper [3], the GAN network was proposed as a multilayer perceptron with fully connected layers: CNNs are known to be better suited to computer vision tasks. [4], though not the first to use GANs with convolutional layers, proposed a set of architectural rules and constraints to adversarial generation which became the cornerstone of most works that use GANs. Figure 2.4 and Figure 2.3 illustrate the discriminator and generator architectures of Deep Convolutional Generative Adversarial Networks(DCGAN). The innovations from [4] can be summarised as follows:

- A comprehensive study for finding a stable architecture for adversarial image generation with batch normalization in both generator and discriminator was conducted.
- Pooling layers were swapped with strided convolutions for downsizing operations.
- For generator, they use Rectified Linear Unit(RELU) after convolutional layers except the last which utilizes hyperbolic tangent(TanH), and Leaky Rectified Linear Unit(LRELU) for discriminator is recommended.

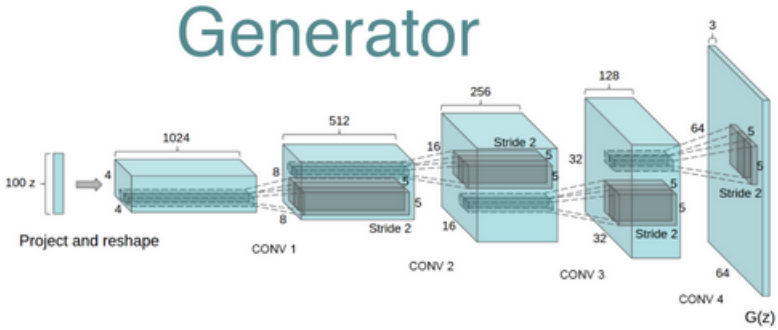


Figure 2.3. DCGAN [4] generator architecture. Image taken from [4].



Figure 2.4. DCGAN [4] disriminator architecture. Image taken from [4].

3. SINGLE IMAGE DEBLURRING

The aim of single image deblurring is to successfully recover the sharp image given its blurry counterpart (See Figure 3.1). Earlier methods can be inspected under two categories, blind and non-blind. More recently, learning based approaches have become popular among researchers. In this chapter, we will review blind/non-blind methods as well as learning based works by focusing on spatially-varying motion deblurring which is the main subject of this thesis.

3.1. Classical Methods

Given a blurry image b, image blur can be modelled as follows:

$$b = k * s + n \tag{3.1}$$

where s is the latent sharp image, k is the non-negative blur kernel, n is noise, and $*$ denotes the convolution operation. Non-blind approaches seek to recover the sharp image s given the blurry image b and the blur kernel k. Obtaining the latent sharp image from the observed blurry image when knowledge about the blur kernel is not present is in the scope of blind deconvolution techniques (See Figure 3.2).



(a) Blurred (b) Deblurred (c) Sharp

Figure 3.1. A deblurring example from CelebA [5]



Figure 3.2. A graphical explanation for non-blind deblurring (top row) and blind deblurring (bottom row). (Image taken from [6]'s deblurring dataset)

Image deblurring is a classical inverse problem that suffers from severe ill-posedness (See Figure 3.3). For non-blind methods, this arises from ill-posed blur kernels. For the non-blind case, the deblurring problem is ill-posed because the blur kernel is ill-conditioned [11]. The ill-posedness in blind deblurring is caused by the infinite number of (s, k) pairs that could result in the observed blurry image b, even if the blur kernel is not ill-posed [11]. Also, for the blind deblurring case, one unwanted solution to Eq. 3.1 can be the no-blur explanation where the kernel is selected as the identity operator that yields the solution $s = b$ [24]. To address these issues, the need for additional constraints on s, k or both are necessary to to contrain the space of possible solutions [24]: the Bayesian framework naturally suits this situation.

3.1.1. Non-blind Deblurring

Given a blurry image and the blur kernel affecting the image, non-blind methods try recover the underlying sharp image as accurately as possible [25–31]. Though blind methods have become more popular recently since it is not possible to know the blur kernel affecting captured real blurry images, improvements in non-blind approaches are important as many blind approaches have non-blind deconvolution as an intermediate

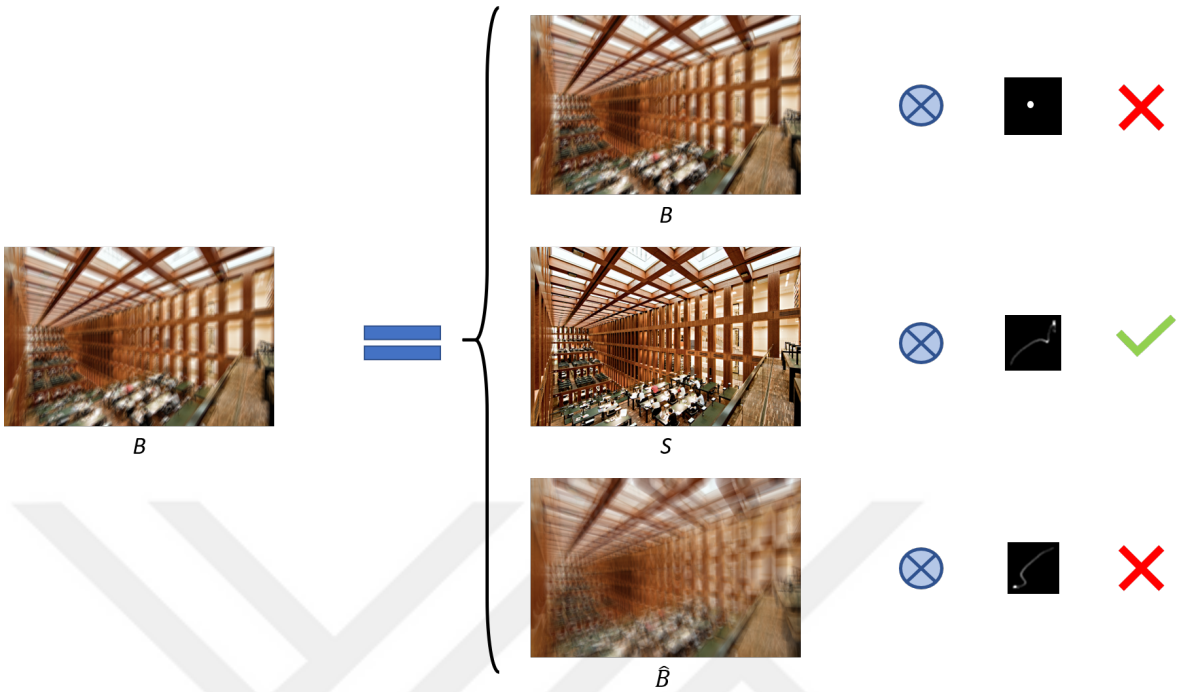


Figure 3.3. A graphical explanation for the ill-posed nature of deblurring.(Image taken from [6]'s deblurring dataset)

step [32] (See Figure 3.4). It was theoretically proven by [24] that splitting the blind deblurring problem into blur kernel estimation followed by non-blind deconvolution to obtain the latent sharp image allows for better results. Some classical methods for non-blind deblurring(NBD) are the Weiner Filter [27], Richardson-Lucy (RL) [28, 33] deconvolution; however, they are known to produce unwanted ringing artifacts [34].

More recently, it can be observed that many works in non-blind deblurring have adopted a maximum a posteriori (MAP) framework [34] of the form:

$$\hat{s} = arg\min_{s} \ \lambda \| s * \hat{k} - b \ \|^2 + P(s) \tag{3.2}$$

where s is latent sharp image, $\hat{s}$ is sharp image estimate, b is the observed blurry image, $\hat{k}$ is the given blur kernel estimate, λ is the weighting factor and P is a prior over the sharp images. The choice of prior has been studied extensively to suppress the ill-posed nature of NBD: [24, 26, 29, 32, 35] have proposed different image priors based on image gradients. One of the main problems deblurring algorithms suffer from are

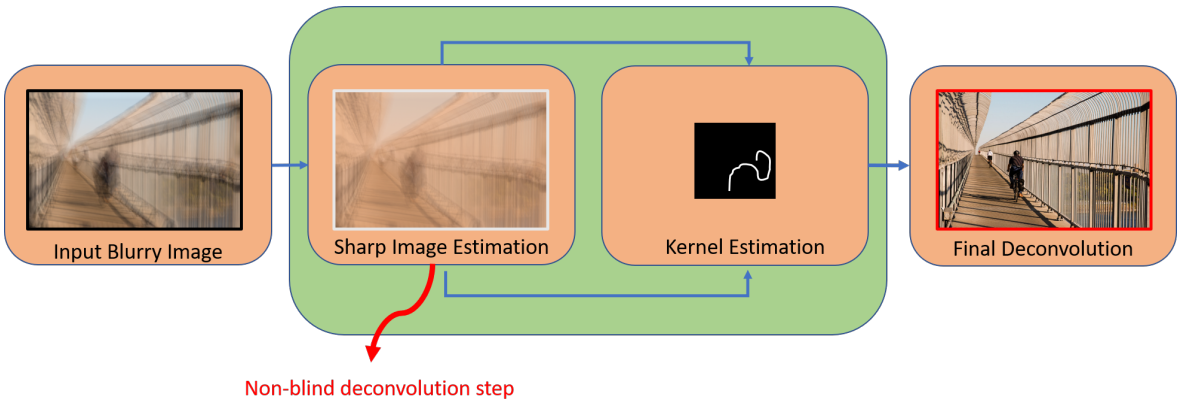


Figure 3.4. Non-blind deconvolution as an intermediate step in blind deblurring.(Image taken from [6]'s deblurring dataset)

ringing artifacts that occur after deconvolution: [36] utilizes residual deconvolutions to overcome ringing effects. Another problem that commonly occurs is caused by outliers in the blurry image. [37, 38] both use $L_1 - norm$ in the energy function that is more robust to impulse noise as opposed to Eq. 3.2. [25] presents a robust deconvolution algorithm that is advantageous for both impulse noise and pixel saturation. (See Figure 3.5)

3.1.2. Blind Deblurring

Blind deblurring (BD) methods try to obtain the latent sharp image along with the blur kernel [12, 39–43]. Bayesian inference is commonly used for deblurring (blind and non-blind) and two approaches for BD are MAP and Variational Bayesian (VB) frameworks. The deblurring problem with Bayesian inference can be formulated using the Bayes' Rule:

$$p(s,k|b) = \frac{p(b|s,k)p(s)p(k)}{p(b)} \tag{3.3}$$

where $p(s,k|b)$ is posterior distribution for latent sharp image and unknown blur kernel given observed blurry image, $p(b|x,k)$ is the likelihood term, $p(s)$ and $p(k)$ are image and blur priors, and lastly $p(b)$ is the evidence. The idea in MAP estimation is finding the maximum of Eq. 3.3 or similarly $p(s,k|b) \propto p(b|x,k)p(s)p(k)$. By taking the loga-

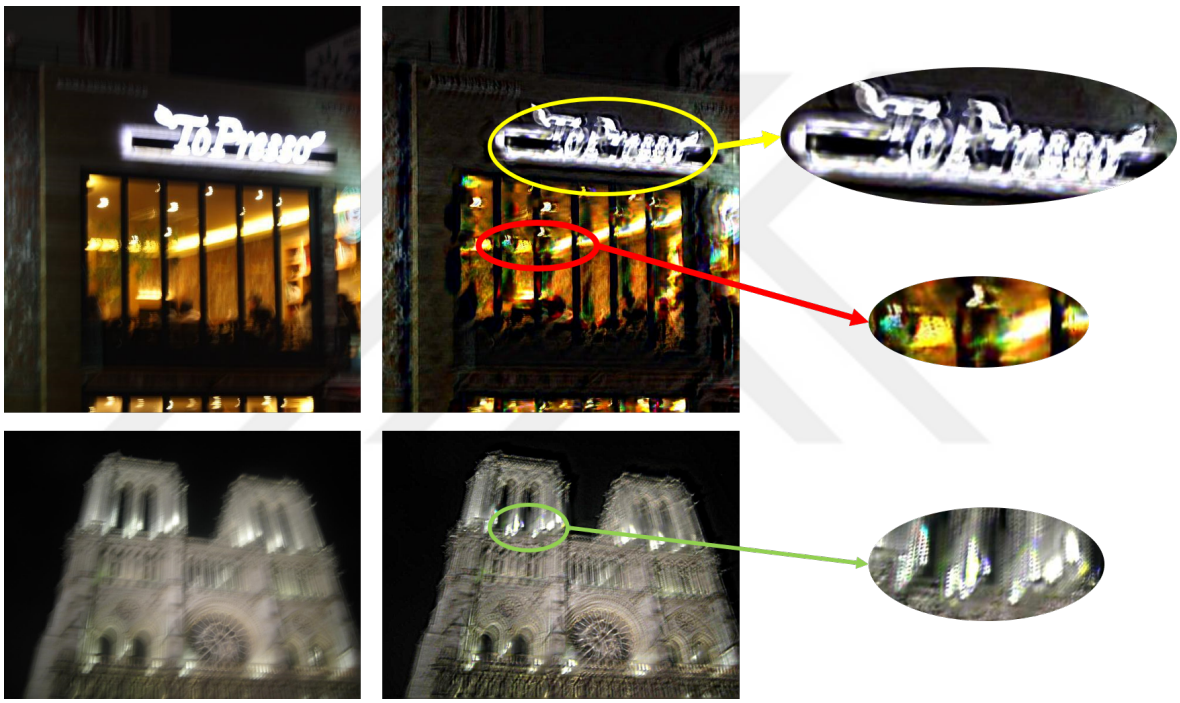


Figure 3.5. Bright/saturated regions throughout blurry images can cause severe ringing artifacts after deblurring. Left column illustrates blurry images while the right column shows deblurring result. Areas with ringing artifacts have been magnified for better visualization.(Image taken from [6]'s deblurring dataset)

rithm and eliminating regularizing constants, maximizing Eq. 3.3 becomes equivalent to the following [44]:

$$\min_{s,k} \; -2log\, p(s,k|b) = \min_{s,k} \frac{1}{\lambda}\|k * s - y\|^2 + \rho_s(s) + \rho_k(k) \tag{3.4}$$

where the first term on the right-hand side is referred to as the *data fitting term* and $\rho_s(s)$ and $\rho_k(k)$ are regularization terms for sharp images and blur kernel, respectively. Several works that use the MAP framework are [39,41,45] with different natural image priors. [24] proposed the MAP_k approach that estimates the blur kernel k first, then using a conventional non-blind deconvolution method, the latent image is obtained. MAP_k was theoretically proven to recover more accurate kernels as opposed to MAP. On the other hand, VB approaches calculate approximations when intractable integrals arise during a Bayesian framework process, e.g. marginalization over the latent sharp image in a MAP_k framework [46]. In deblurring research, *mean field VB* is used popularly where $p(s,k|b)$ is approximated by a variational distribution $q(s)$ by minimizing the Kullback-Leibler (KL) Divergence between the two distributions:

$$D_{KL}(q\|p) = \int q(s) log \frac{q(s)}{p(s,k|b)} ds \tag{3.5}$$

Some known works in VB include but are not limited to [47–49].

3.2. Deblurring Methods for Object Motion

To accurately examine existing deblurring approaches, one has to consider several factors such as the type of input accepted (e.g. single frame or multiple frames), the kind of blur present in the input and whether the degradation is spatially-varying or not(See Figure 3.6). The focus of this paper is on eliminating object motion blur from single images: this type of degradation naturally falls under the spatially-varying blur class. (See Figure 3.7).



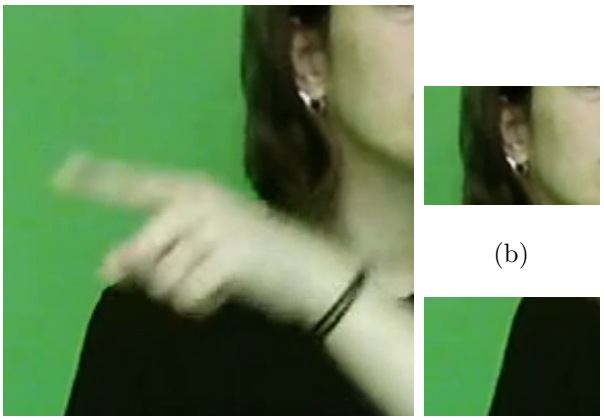
Figure 3.6. An example of synthetic motion blur on a text image. Different spatial locations are degraded by unidentical blurs. Left image shows sharp image and left image is synthetically blurred image.(Image taken from [6]'s deblurring dataset)

While talking about the taxonomy of deblurring methods, it is important to state the various types of blur and how each inspires its own plethora of deblurring techniques. Some common blur types are as follows: object motion blur, camera shake blur, defocus blur [11]. Ideally, a deblurring method is expected to handle any type of blur; however, this is a hard objective as each type corrupts an image in a different way. In this thesis, we focus specifically on object motion deblurring and the existing work in motion deblurring should be inspected carefully to distinguish between object motion blur and camera shake blur [50–53] which are both generally referred to as motion blur.

Earlier works approach the object motion blur as a joint deblurring and segmentation problem. [7] separates the blurry region using user input and a matting technique [54]; then, blur is eliminated using an existing method [55]. Figure 3.8 illustrates the general overview of [7]'s method. Similar to [7], a blur detection algorithm is proposed for blurry object segmentation. In [56], images are segmented into regions with different blurs by leveraging simple image statistics, which allows the method to handle multiple blurry objects in a dynamic scene. Another work [57] obtains pleasant deblurring and segmentation results in case of mild blur but quality decreases for more serious blur. A more recent approach [6] formulates the joint segmentation and motion blur elimination problem in a maximum a posteriori (MAP) setting in which soft-segmentation is aided by user-provided seeds.

3.3. Learning Based Methods

Differing from more traditional deblurring methods, learning based approaches that directly recover the sharp image without estimating the blur kernel have emerged in the Deep Learning and Convolutional Neural Network (CNN) [58] literature. Similar to any other deep learning application, CNN-based deblurring approaches require large datasets for learning but achieve visually-pleasing results. The introduction of adversarial training can be considered as a milestone for image deblurring with CNNs as considerable performance increase was achieved with their aid. Our work falls under



(a) (b) (c)

Figure 3.7. Object motion blur from our proposed dataset. The background contains no blur((b)-(c)). Blur degradation is around the hand area(a).

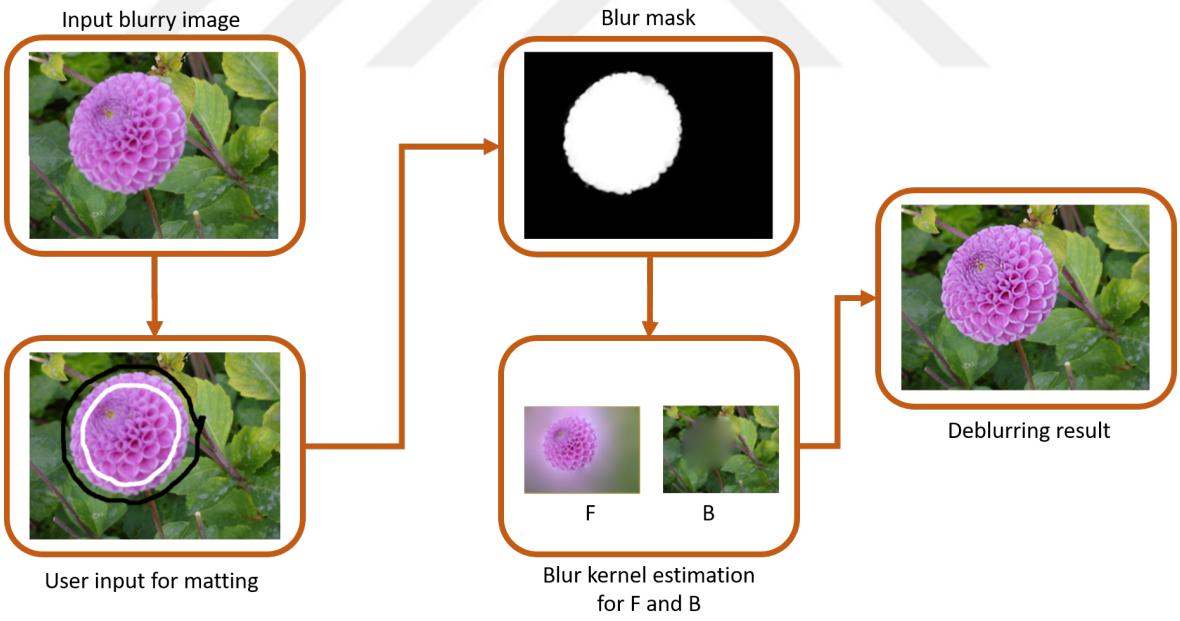


Figure 3.8. An Overview of [7]'s method for partial defocus/object motion blur. The task is split in to two succesive steps: foreground and background segmentation(matting) and blind deblurring on blurry parts. During the second step, blur kernel estimation and blur elimination is applied on both the background and foreground. (Image taken from [7])

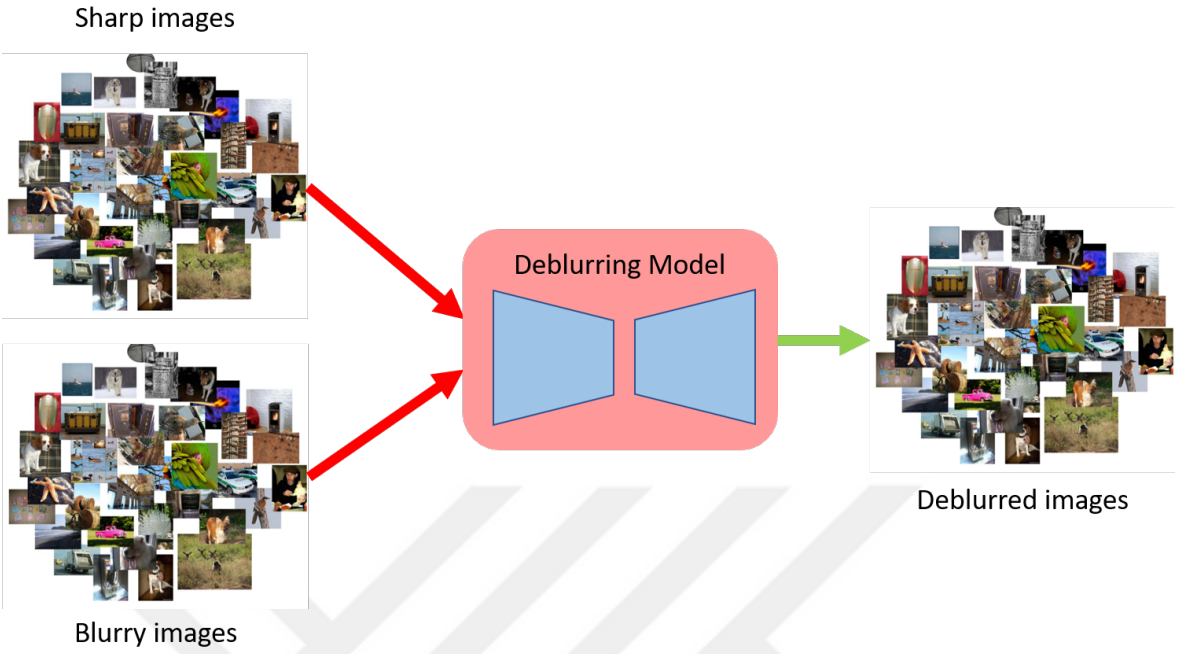


Figure 3.9. Learning-based deblurring methods require large amounts of data: earlier methods use corresponding sharp-blurry image pairs for learning, however, recent approaches can learn from uncorresponding data which enabled working on real blurry datasets as opposed to synthetically created datasets.(Image reprinted from [8])

this family of methods which are blind and kernel-free that directly learn to deblur a degraded image. Earlier work [30,59,60], use neural networks in a non-blind deblurring setting. [30] trains an end-to-end blind deblurring network on a dataset consisting of sharp and corresponding blurry images, with degrading performance in big blurs. Similar to image denoising, these works treat debluring as a general image enhancement task and utilize multi-layer perceptrons as building blocks for their networks. Two CNN-based methods geared towards object motion deblurring [17, 61] have created novel sharp-blurry datasets by averaging the frames from a high frame-rate camera to realistically simulate moving object blur and both apply a multi-scale approach. Inspired by [17], DeblurGAN [12] also uses the adversarial idea for the deblurring generator but with Wasserstein GAN [22] and gradient penalty [23]. In terms of the dataset, DeblurGAN modifies [17]'s dataset to obtain a camera-motion dataset which renders the whole method unsuitable for moving object deblurring. The follow-up work of [12] called DeblurGAN-v2 [62], uses the relativistic conditional GAN formulation with a

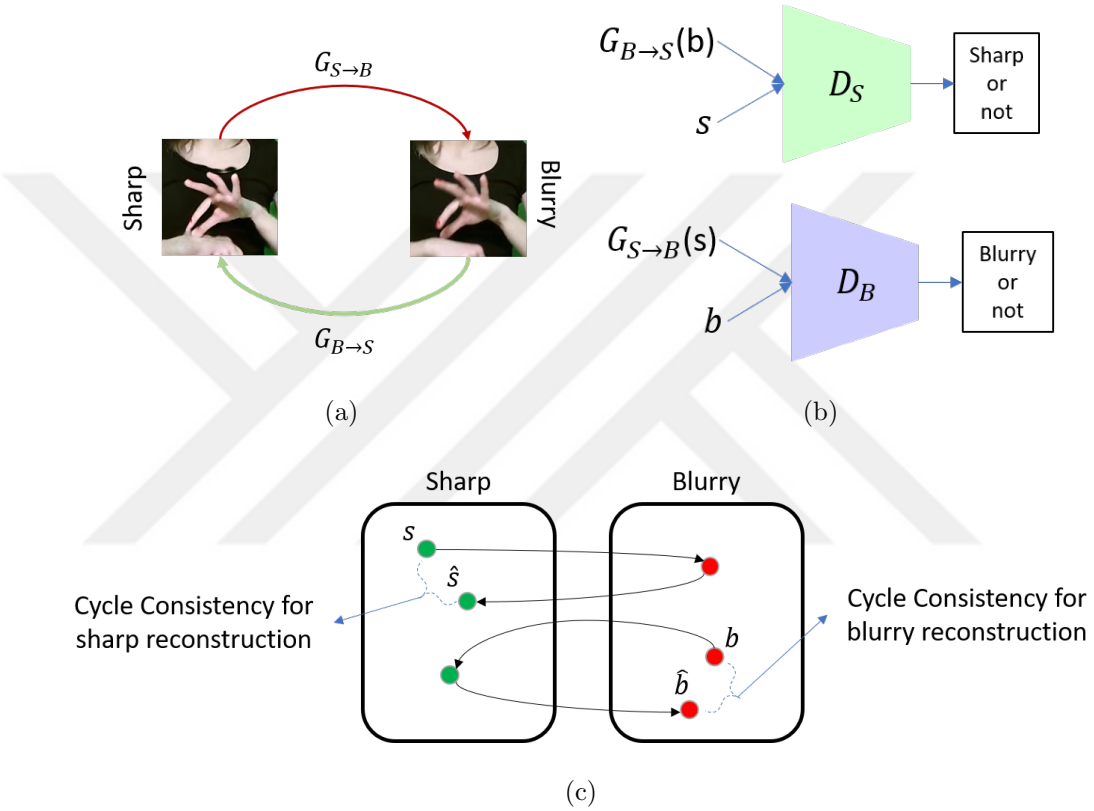


Figure 3.10. Overview of CycleGAN [9] architecture applied to deblurring. (a)Domain generators $G_{S\to B}$ and $G_{B\to S}$ learn mapping between sharp and blurry image domains. (b)Domain discriminators D_S and D_B assess how well the generators translate images between domains. (c) Cycle consistency losses regularize mappings between domains. The idea behind this is that by mapping a sample from one domain and back to its original domain, we want to arrive at the same point.

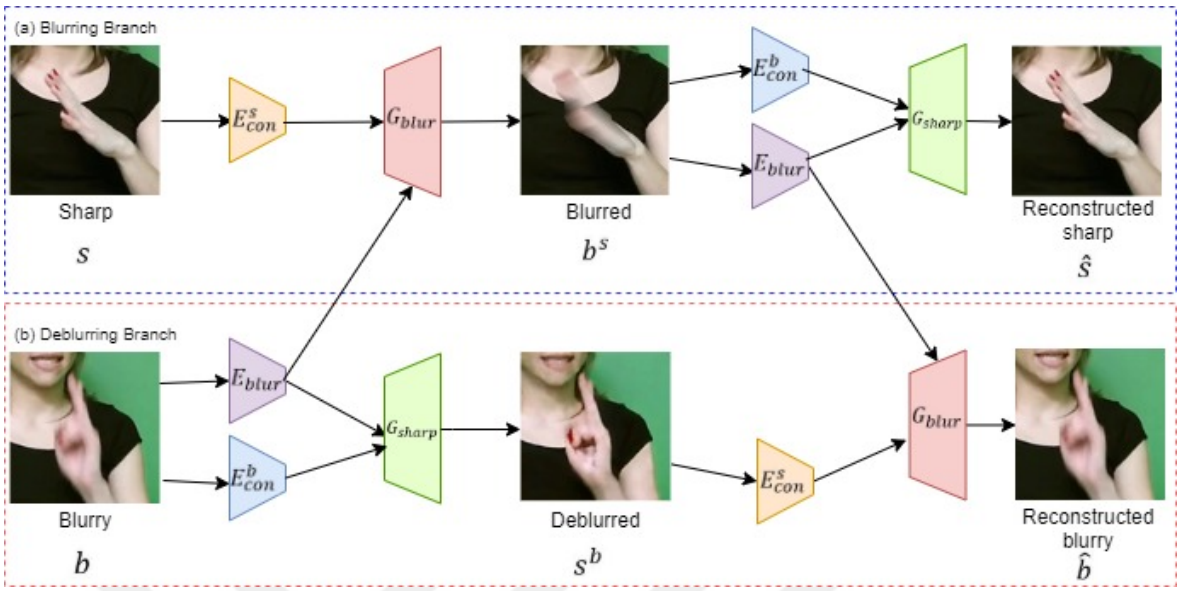


Figure 3.11. [10]'s deblurring architecture visualized with our novel hand dataset. Training with unpaired images is achieved via blurring/deblurring branches and cross-cycle consistency losses.

double-scale discriminator for increased efficiency and quality.

A newer branch of adversarial approaches [10, 20, 63] have emerged for blind kernel-free blur elimination inspired from image-to-image translation [9] (See Figure 3.10). [63] introduces a novel architecture with dense connections. However, this work is aimed towards camera-shake blur. [20] is the leading work in unsupervised data-driven deblurring in which unpaired blurry and sharp training data can be used. Following this idea, [10] introduces a blur generating branch on a deblurring branch to provide self-supervision to the blur disentanglement. [10]'s architecture is illustrared in Figure 3.11. Both [20] and [10] and all other deep learning based methods use artificially blurred datasets which leaves them open to criticism: Synthetic datasets used in training do not usually capture the complex nature of object motion blur and these methods often perform poorly in case of real blur degradation [1].

4. SPATIALLY VARYING MOTION BLUR ELIMINATION

Our goal is to deblur images degraded by spatially varying object motion using unpaired sharp and blurry data. The general overview of the network can be seen in Figure 4.1. The framework consists of two branches that are trained with a pair of non-corresponding sharp and blurry images: a deblurring branch where the blurry image is disentangled into its content and blur features to obtain the desired sharp image, and a blurring branch that generates a novel blurry image by combining the sharp image with the learned blur feature. To solve the spatially varying blur problem, we extract the blur features using an encoder network with partial convolution [18] layers that processes only the blurry parts of the image. The components of the network for blurry and sharp image domains, B and S, are as follows: content encoders $\{E^B_{con}, E^S_{con}\}$, image generators $\{G_{blur}, G_{sharp}\}$, image discriminators $\{D_{blur}, D_{sharp}\}$ and a blur encoder $\{E_{blur}\}$.The source of self-supervision for the above disentanglement is described in the following sections.

4.1. Disentangling Content and Spatially Variant Object Motion Blur

Inspired by [10], motion deblurring is posed as an image translation problem from the blurry domain B to the sharp domain S. To tackle this when there are no ground truth sharp/blurry image pairs, an approach based on disentangled representations is used and several strategies are followed to embed images into two spaces: a content space that captures shared information from each domain and a blur space for the blurry image domain. For content feature mapping to a domain-invariant space, disentanglement is achieved via weight sharing [64] in the final layers of content encoders $\{E^b_{con}, E^s_{con}\}$: this enables E^b_{con} to learn from E^s_{con} which is assumed to be a good content extractor by nature.

For blur feature learning, the blur attribute encoder $\{E_{blur}\}$ is expected to contain no content features belonging to the input blurry image b: this is accomplished by

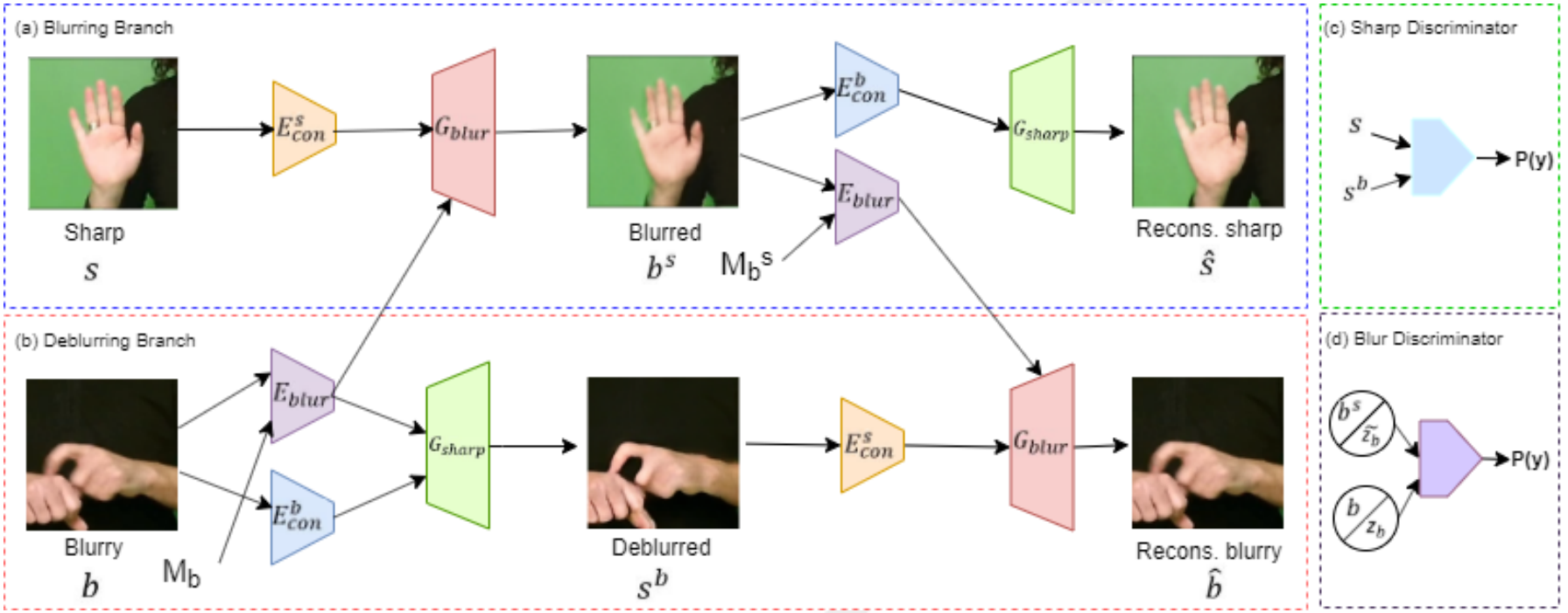


Figure 4.1. Our proposed network.

utilizing these features in generating novel blurry images from sharp images and region-of-interest(ROI) masks M_b. During training, the content feature $E^S_{con}(s)$ of the sharp training image and the blur feature $E_{blur}(b, M_b)$ from the blurry training image are fed into the blurry image generator G_{blur} to obtain b_s, which discourages E_{blur} to encode any content information. Furthermore, as opposed to [10], for the case of spatially varying motion blur degradation, we propose to use Partial Convolutional Layers [18] in E_{blur} to extract blur information only from the blurry parts of the image. For this, we use a mask corresponding to the blurry regions of a training image. For feature values B and corresponding binary mask values M in the current sliding convolution window, the partial convolution operation is defined as follows:

$$b^{'} = \begin{cases} W^T(B \odot M)\frac{1}{sum(M)} + b & if \;\; sum(M) > 0 \\ 0 & otherwise \end{cases} \tag{4.1}$$

where W is the weight matrix of the convolution kernel, b is the bias, and $\odot$ denotes the Hadamard product. The mask gets similarly updated after each partial convolution: the unmasking rule is given as:

$$m^{'} = \begin{cases} 1 & if \;\; sum(M) > 0 \\ 0 & otherwise \end{cases} \tag{4.2}$$

where the mask is removed if and only if the convolution was able to condition its output on at least one valid pixel value (See Figure 4.2). The use of partial convolution in the blur encoder allows the network to attend only to the moving blurry object with appropriate scaling applied in regards to the changing area of the object.

4.2. Cross-Cycle Consistency Loss

The content encoders allow us to map sharp and blurry training pairs onto a shared latent space and the blur encoder learns domain-specific blur representations: using this disentanglement, we can perform inter-domain image translation between

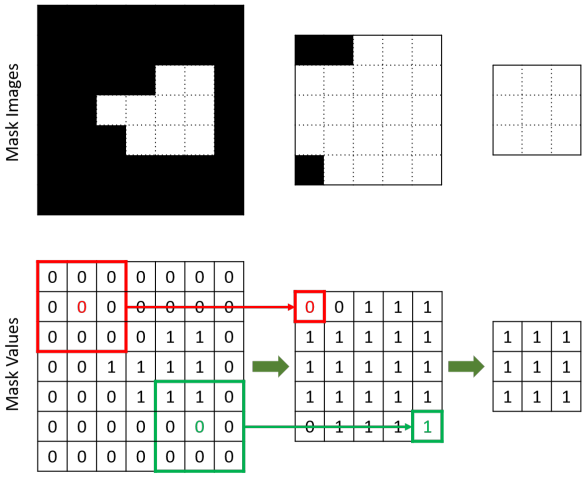


Figure 4.2. A toy example for the partial convolution's unmasking rule. For this case, two consecutive convolution operations are applied on a given mask image with a 3x3 kernel, unit stride and no padding. During the first convolution the mask for the upper left corner is not unmasked as it was not conditioned on any valid pixels. The bottom right corner is masked because its feature map was conditioned on two valid pixels.

sharp and blurry domains. Cross-cycle consistency [64] leverages the network's ability to do cyclic reconstructions. More explicitly, we do the first cross-translations to obtain blurred and deblurred images b^s and s^b using corresponding ROI masks M_b and M_{b^s} from h:

$$b^s = G_{blur}(E^S_{con}(s), E_{blur}(b, M_b)), s^b = G_{sharp}(E^B_{con}(b), E_{blur}(b, M_b)) \tag{4.3}$$

Then, second cross-translations are performed to reconstruct the original training pairs, completing the cycle:

$$\hat{b} = G_{blur}(E^S_{con}(s^b), E_{blur}(b^s, M_{b^s})), \hat{s} = G_{sharp}(E^B_{con}(b^s), E_{blur}(b^s, M_{b^s})) \tag{4.4}$$

Finally, the cross-cycle consistency losses on sharp and blurry images is given below:

$$L_{cc} = \mathop{\mathbb{E}}_{s\sim p(s)}[\|s - \hat{s}\|_1] + \mathop{\mathbb{E}}_{b\sim p(b)}[\|b - \hat{b}\|_1] \tag{4.5}$$

4.3. Domain Adversarial Loss for HospiSign

Adversarial loss is applied for pleasing image generation results in both blurring and deblurring branches. For sharp and blurry domains, domain discriminators $\{D_{blur}, D_{sharp}\}$ aim to distinguish between real and generated images while the generators $\{G_{blur}, G_{sharp}\}$ strive to produce images as close to real image distributions as possible. For the sharp domain, we use Wasserstein loss [22] with gradient penalty [23]: the sharp generator G_{sharp} maximizes the objective function against the sharp discriminator D_{sharp} which aims to minimize it:

$$L_{D_S} = \mathop{\mathbb{E}}_{s \sim p(s)} [D_{sharp}(s)] - \mathop{\mathbb{E}}_{b \sim p(b)} [D_{sharp}(s^b)] \tag{4.6}$$

where $s^b = G_{sharp}(E^B_{con}(b), z_b)$ is and $z_b = E_{blur}(b, M_b)$ denotes the disentangled blur feature.

The objective for the blurry image domain is defined similarly with a minor modification: We condition the blur discriminator D_{blur}'s input images with the corresponding blur attributes extracted by the blur encoder E_{blur}:

$$L_{D_B} = \mathop{\mathbb{E}}_{b \sim p(b)} [D_{blur}(b, z_b)] - \mathop{\mathbb{E}}_{s \sim p(s)} [D_{blur}(b^s, \widetilde{z_b})] \tag{4.7}$$

where $b^s = G_{blur}(E^S_{con}(s), z_b)$ and $\widetilde{z_b} = E_{blur}(b^s, M_{b^s})$. This condition is applied because we expect the blur feature z_b extracted from the original blurry image b to be similar to the feature $\widetilde{z_b}$ disentangled from the blurred image b^s: the blurred image is generated with the blur feature learned from the blurry training image.

4.4. Domain Adversarial Loss For CelebA

As opposed to the hand dataset which uses Wasserstein formulation, we use the original GAN loss [3] for the CelebA dataset. For sharp and blurry domains, domain discriminators $\{D_{blur}, D_{sharp}\}$ aim to distinguish between real and generated images

while the generators $\{G_{blur}, G_{sharp}\}$ strive to produce images as close to real image distributions as possible. The adversarial losses for blurry and sharp domains L_{D_B} and L_{D_S} respectively, are defined as:

$$\begin{aligned} L_{D_B} = & \mathop{\mathbb{E}}_{b\sim p(b)}[\log D_{blur}(b, z_b)] \\ & + \mathop{\mathbb{E}}_{s\sim p(s)}[\log(1 - D_{blur}(b^s, \widetilde{z_b}))] \end{aligned} \tag{4.8}$$

where $b^s = G_{blur}(E^S_{con}(s), z_b)$ and $\widetilde{z_b} = E_{blur}(b^s, M_{b^s})$

$$\begin{aligned} L_{D_S} = & \mathop{\mathbb{E}}_{s\sim p(s)}[\log D_{sharp}(s)] \\ & + \mathop{\mathbb{E}}_{b\sim p(b)}[\log(1 - D_{sharp}(s^b))] \end{aligned} \tag{4.9}$$

where $s^b = G_{sharp}(E^B_{con}(b), z_b)$ and $z_b = E_{blur}(b, M_b)$ denotes the disentangled blur feature.

4.5. Perceptual Loss

We impose a perceptual loss on blurred and deblurred images to reduce the unpleasant artifacts introduced by the generators G_{sharp} and G_{blur}. Proposed by [10], the distance between the features of blurry and deblurred images and sharp and blurred images, extracted from a pre-trained CNN can act as a perceptual similarity metric:

$$Lp = \|\phi_l(s_b) - \phi_l(b)\|_2^2 + \|\phi_l(b_s) - \phi_l(s)\|_2^2 \tag{4.10}$$

where $\phi_l(x)$ denotes the features extracted from the lth layer of a pre-trained network. For this work, $conv_{3,3}$ layer of VGG-19 [65] trained on Imagenet dataset [66] has been used.

4.6. Learning Objective

The full objective function is the weighted sum of the mentioned loss components:

$$L = \lambda_{adv} Ladv + \lambda_{cc} Lcc + \lambda_p Lp$$

where $Ladv = L_{D_S} + L_{D_B}$

4.7. Testing

During test time, a forward pass through the deblurring branch is performed to obtain the deblurred test image $s^{b^{test}}$:

$$s^{b^{test}} = G_{sharp}(z_c^{test}, z_b^{test}) \tag{4.11}$$

where $z_c^{test} = E_{con}^b(b^{test})$ is the content feature and $z_c^{test} = E_{blur}^b(b^{test}, M_{b^{test}})$ is the blur feature with ROI mask $M_{b^{test}}$

4.8. Architecture of Blur Encoder with Partial Convolutions

We present the architectural details of the blur encoder in Table 4.1. As can be seen, the encoder outputs two 1x8 vectors for blur feature mean and variance which are used to draw blur vectors from a Gaussian Distribution. The use of partial convolutions [18] allows the network to learn accurate blur representations by focusing on degraded regions and omitting information from sharp backgrounds while applying appropriate scaling with respect to the changing mask sizes for each frame. In our ablation study presented in Chapter 5, we compare the effect of partial convolutions against regular convolutions: the architecture for the latter is the same as Table 4.1 with regular convolutions instead of partial convolutions.

Table 4.1. Proposed blur encoder architecture with partial convolution layers. The input consists of an RGB image and a blur region mask. The network outputs are two 1x8 vectors for blur feature mean and variance.

Operation	Kernel	Strides	Feature Maps	Pad Type	Non-linearity
4 x H x W input					
Partial Convolution	7x7	1x1	64	Reflect	Relu
Partial Convolution	4x4	2x2	128	Reflect	Relu
Partial Convolution	4x4	2x2	256	Reflect	Relu
Partial Convolution	4x4	2x2	256	Reflect	Relu
Partial Convolution	4x4	2x2	256	Reflect	Relu
Partial Convolution	4x4	2x2	256	Reflect	Relu
Global Avg. Pooling	-	-	-	-	-
(Feed output to two dense layers for feature mean and variance vectors)					
Dense (mean)	-	-	8	-	-
Dense (variance)	-	-	8	-	-

5. EXPERIMENTS AND RESULTS

In this chapter, we present the dataset preparation processes, detailed explanations of the evaluation metrics used for analyzing deblurring performances, experimental results for our proposed deblurring method as well as the detailed comparisons to state-of-the art blur elimination techniques.

5.1. Datasets

This section presents detailed information on the datasets used for our work: The data collection stages as well as the preprocessing steps are explained carefully.

5.1.1. HospiSign

5.1.1.1. Dataset Preparation. HospiSign, a subset of BosphorusSign [67], is a sign language dataset which contains 41 different sign videos performed by 6 native users with phrases that are commonly used in a hospital setting. We have collected a novel dataset featuring sharp and blurry samples. Unlike previous datasets that use synthetically blurred training samples, we extract real blurry frames from videos that occur due to the rapid motion of the hands during signing. We annotate the frames as blurry and sharp using an automated tool as follows: Hand regions are detected using OpenPose's [68] wrist joints and region growing [69] is applied to segment moving hands from the static background. Lastly, a Canny edge detector finds edges in the hand region. Heavy edge presence indicates a sharp image. To eliminate any erroneous detections made by the proposed blur selection system, we manually correct both sharp and blurry image sets. The resulting dataset contains approximately 25K sharp and blurry training pairs and 3K blurry test images. All images have hand area masks obtained during the segmentation process. More details can be found in Table 5.1. Some sharp and blurry samples from our dataset can be seen in Figure 5.1 and Figure 5.2, respectively.

Table 5.1. Statistical information from the novel blur dataset for training and testing divisions

ID of Signer \ Data Split	# Sharp Tr.	# Blurry Tr.	# Blurry Tst.
User 2	5563	6002	500
User 3	4668	5310	500
User 4	4448	5097	500
User 5	6292	5380	500
User 6	2089	3000	500
User 7	2406	2943	500
Total	25466	27732	3000

5.1.2. CelebA

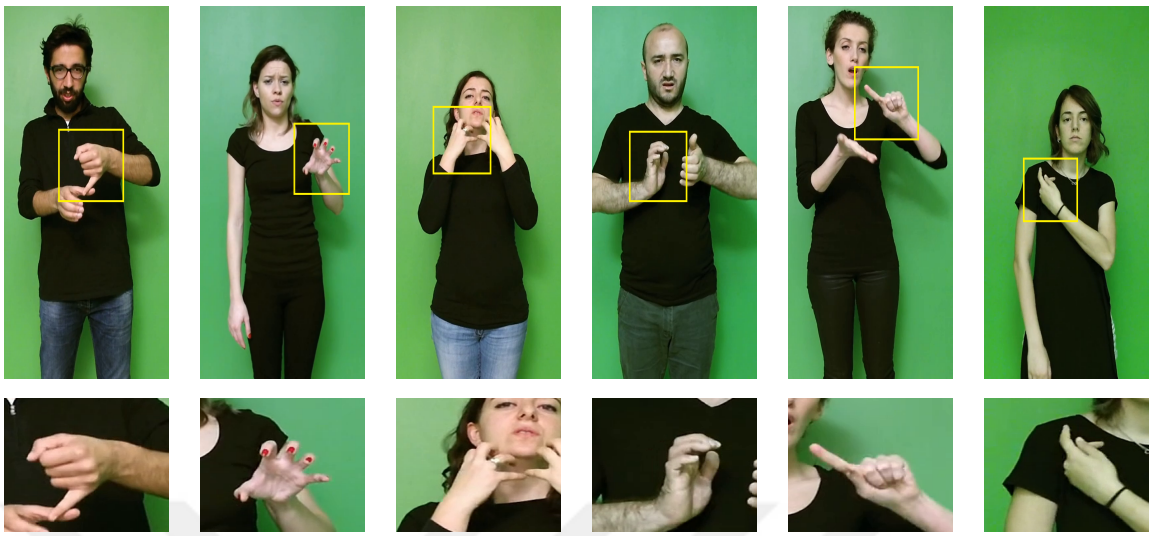
Dataset Preparation: We use the CelebA [5] dataset using a similar protocol as proposed in [10]. The dataset is randomly split into training/sharp(100K) training/blurred(100K) and test(2137) sets. The images in blurred and test sets are created with the blurring routine from DeblurGAN [12]. However, rather than blurring the entire image as in [5], we only blur the facial region(See Figure 5.7) to better fit our objective of deblurring in the presence of partial-blur.

5.2. Evaluation Metrics

Quantitative evaluation is conducted using the following metrics:

5.2.1. Inception Score

Inception Score(IS) [70] is a GAN-based metric proposed to assess the quality of generated images from a generative network. Higher IS scores signify better image



User 2 User 3 User 4 User 5 User 6 User 7

Figure 5.1. Some sharp hand samples from our novel deblurring dataset. The top row shows the frames where the sharp hand images in the bottom row were taken.

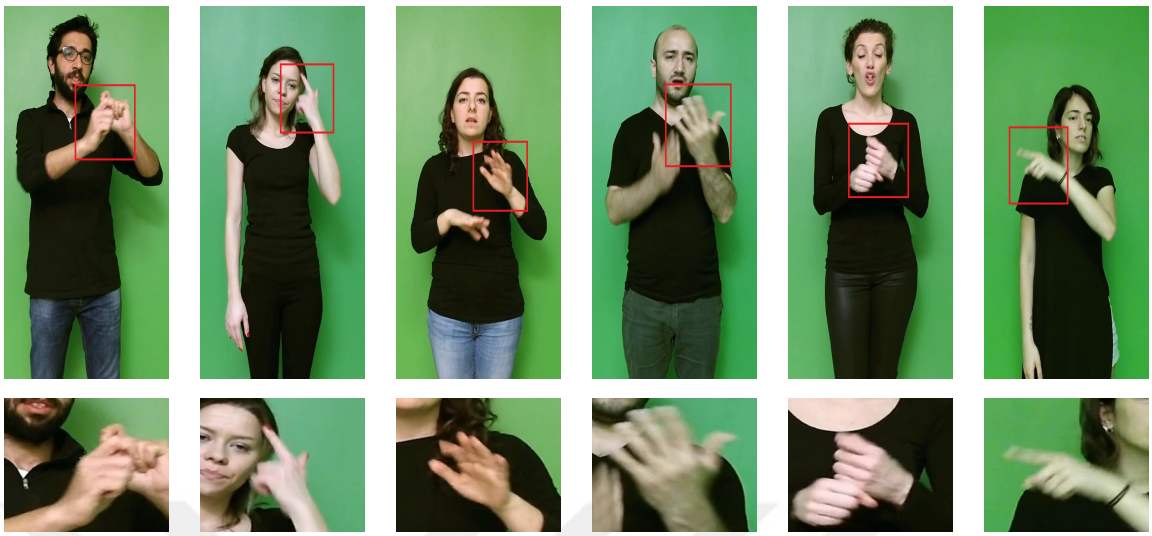
quality as well as diversity of a set of images. As the name suggests, a pretrained Inception network [71] is leveraged to obtain logits for the set of images, the IS is as follows:

$$IS(G) = exp(\ \mathbb{E}_{x \sim p_g(x)}\ D_{KL}(\ p(y|x)\ ||\ p(y)\ )\ ) \tag{5.1}$$

where $x \sim p_g(x)$ is an image sampled from the generated image distribution. KL-Divergence D_{KL} is calculated between the conditional label distribution $p(y|x)$ given an image and the marginal label distribution $p(y) = \mathbb{E}_{x \sim p_g(x)}[p(y|x)]$). Some criticisms for IS are that an image set with one sample per class will still score high even though intra-class diversity is not achieved and also that it is limited by the Inception network's class distribution.

5.2.2. Frechet Inception Distance

We use Frechet Inception Distance(FID) [72] to compare the statistics between the sharp image distribution and the deblurred image distribution. As opposed to IS, lower FID scores imply superior image quality as well as diversity among samples. Both



User 2 User 3 User 4 User 5 User 6 User 7

Figure 5.2. Blurry hand images (bottom row) from our proposed hand deblurring dataset. The top row illustrates the full frames of the sign language dataset our dataset was collected from.

IS and FID use Inception network [71] for lower dimensional image features. However, FID extracts features from an intermediate layer (e.g pool-3 layer) in contrast to IS which uses the logits layer. FID between real images and fake images is calculated by modelling Gaussian distributions from their features:

$$FID = \|\mu_r - \mu_g\|^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}}) \tag{5.2}$$

where (μ_r, Σ_r) and (μ_g, Σ_g) denote the mean vector and covariance matrix pairs for real and generated image sets.

5.2.3. Kernel Inception Score

[73] proposed Kernel Inception Distance(KID) against FID, which is biased to large number of samples. Lower scores for KID imply better performance. Similar to FID, KID is defined as the Maximum Mean Discrepancy (MMD) between Inception

features. For the choice of kernel function, the following polynomial kernel is selected:

$$k(x, y) = (\frac{1}{d}x^T y + 1)^T \tag{5.3}$$

where d is the dimension of Inception features, which for our case is 2048.

5.2.4. Peak Signal to Noise Ratio

Peak Signal to Noise Ratio (PSNR) is a standard full-reference metric commonly used in deblurring research. It can only be used when the uncorrupted initial sharp image is present alongside its corrupt (blurred) counterpart. Higher PSNR scores represent greater image quality. The definition for PSNR is as follows:

$$PSNR(x, \hat{x}) = 20\, log_{10}\,(\frac{MAX_f}{\sqrt{MSE(x, \hat{x})}}) \tag{5.4}$$

where MAX_f is the maximum pixel value for an image, e.g. for 8-bit image representation this value is 255, and MSE is the mean square error between sharp and deblurred images, calculated as:

$$MSE(x, \hat{x}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x_i})^2 \tag{5.5}$$

where, n is the number of samples.

We also apply a masked version of this metric: M-PSNR is calculated from the blurry areas of an image and is normalized with respect to the changing mask area.

5.2.5. Structural Similarity Index

Structural Similarity Index [46] is also a full-reference metric for generated image quality assessment and similar to PSNR, higher scores are preferred for higher image quality. As opposed to MSE and PSNR that are pixel-wise error metrics, SSIM was

developed as a structural similarity metric that appeals to human perception. Let x and $\hat{x}$ be two images of size $M \times M$, then, SSIM score is given by:

$$\begin{aligned} SSIM(x, \hat{x}) &= \frac{(2\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)}, \\ c_1 &= (k_1 L)^2, \\ c_2 &= (k_2 L)^2 \end{aligned} \tag{5.6}$$

where (μ_x,σ_x^2) and ($\mu_{\hat{x}}$,$\sigma_{\hat{x}}^2$) are mean-variance pairs for x and $\hat{x}$ and $\sigma_{x\hat{x}}$ is the covariance with respect to x and $\hat{x}$, $(k_1, k_2) = (0.01, 0.03)$ and $L = 255$.

5.3. Experimental Results and Discussion

We evaluate our method on two datasets: CelebA and a novel dataset, HospiSign, which contains real, spatially varying hand blur. In CelebA, we have added synthetically generated blur to the face area. This lets us use objective error metrics which measure the error between real and deblurred images. The second dataset, on the other hand, has spatially varying hand motion blur that varies from moderate to heavy.

5.3.1. Implementation Details

We implemented all our models using Tensorflow [74] deep learning framework. The network architectures are similar to [10]: The content encoder E_{con} consists of three convolution layers and four residual blocks, and the generators G are symmetric to E_{con} with fractionally-strided convolutions instead of regular convolutions. The blur encoder E_{blur} uses 7 partial convolution layers followed by fully connected layers for feature extraction. The discriminators apply a multi-scale architecture for increased performance. The code will be publicly available in [75].

We used the Adam Optimizer [76] with decay rates $(\beta_1, \beta_2) = (0.5, 0.999)$ during training. For CelebA dataset, we use a learning rate of 0.0001 with loss component weights as follows: $\lambda_{adv} = 1$, $\lambda_c c = 10$, $\lambda_p = 0.1$. The model is trained for 80 epochs

and we apply linear learning rate decay after 40 epochs. For HospiSign, the learning rate is 0.00005 with the following hyper-parameters: $\lambda_{adv} = 0.01$, $\lambda_{cc} = 10$, $\lambda_p = 0.5$ and the weight of gradient penalty $\lambda_{gp} = 100$. We train the model for 40 epochs and linear learning rate decay is applied after 20 epochs.

5.3.2. Compared Methods

We compare our method against some state-of-the-art deblurring methods [10, 12, 62, 77]. For [12, 62], the pretrained models provided by the authors are used during evaluation since both of them can only be trained in the presence of paired sharp/blurry samples. [10] and [77] are trained from scratch using both CelebA and our spatially variant blur dataset.

5.3.3. Results for CelebA

We present the quantitative results for CelebA dataset in Table 5.3.3. Figure 5.7 illustrates the visual comparisons for some sample images from the test set. Our method achieves superior performance in FID and obtains comparable scores across other metrics. Visual inspection of deblurring results in Figure 5.7 indicates that both [12] and [62] fall short in producing pleasing outcomes: [12] fails to perform sufficient deblurring on the test images while [62] produces unnatural artifacts. [77] performs well in PSNR/SSIM metrics but produces slightly discoloured images from the original sharp images. [10], the closest method to us, causes some minor artifacts around the eyes and the mouth area.

5.3.4. Results for HospiSign

The qualitative and quantitative results for our novel blur dataset can be inspected in Figure 5.3, Figure 5.6, and Table 5.4, respectively. Our method performs favourably across IS and KID metrics and produces visually pleasing results. Upon visual examination of the deblurred results in Figure 5.6, under-deblurring is present

Table 5.2. CelebA quantitative results for unsupervised metrics

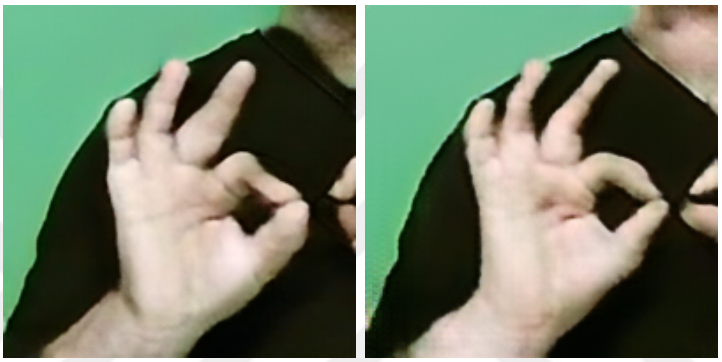
Method \ Metric	IS	FID	KID
Kupyn *et al.* [12]	**3.327790**	19.039589	0.010072
Kupyn *et al.* [62]	3.179018	20.562109	0.010018
Zhu *et al.* [77]	3.029768	13.614720	0.003701
Lu *et al.* [10]	3.055309	12.943586	0.002473
Ours	2.968421	**11.870880**	**0.001567**

Table 5.3. CelebA quantitative results for full reference metrics

Method \ Metric	PSNR	SSIM	M_PSNR
Kupyn *et al.* [12]	**27.853812**	**0.832676**	**25.355615**
Kupyn *et al.* [62]	23.963674	0.726193	22.146578
Zhu *et al.* [77]	27.317720	0.822587	25.178801
Lu *et al.* [10]	26.950048	0.804321	24.991829
Ours	24.138607	0.735460	22.914365



(a) Blurry (b) [77]



(c) Ours (d) [10]

Figure 5.3. Comparison of state-of-the-art unpaired deblurring methods on our the novel hand blur dataset.

for [12] and [62], as can be seen in the two right-most columns. On the other hand, [10] tends to erase blurry regions of the hand. [77] also shows signs of under-deblurring as seen in Figure 5.6d, but eliminates blur in Figure 5.6(k-r) and obtains the best FID score.

Ablation Study We conduct an ablation study to examine the effect of partial convolutions in the blur encoder network and report the evaluation metric results for HospiSign in Table 5.5. We start with the *Base Model* which is similar to [10] with blurring/deblurring branches but without the KL-divergence loss that was proposed in [64]. This loss aims to constrain blur features so that no content information is present in the blur feature: we do not use this loss in our ablation study or final network. In the next step, for blur extraction, we feed masked hand regions to E_{blur} which consists of regular convolutions: this leads to a slight improvement in terms of all metrics. Then, we switch to partial convolutions in E_{blur} which yields greater

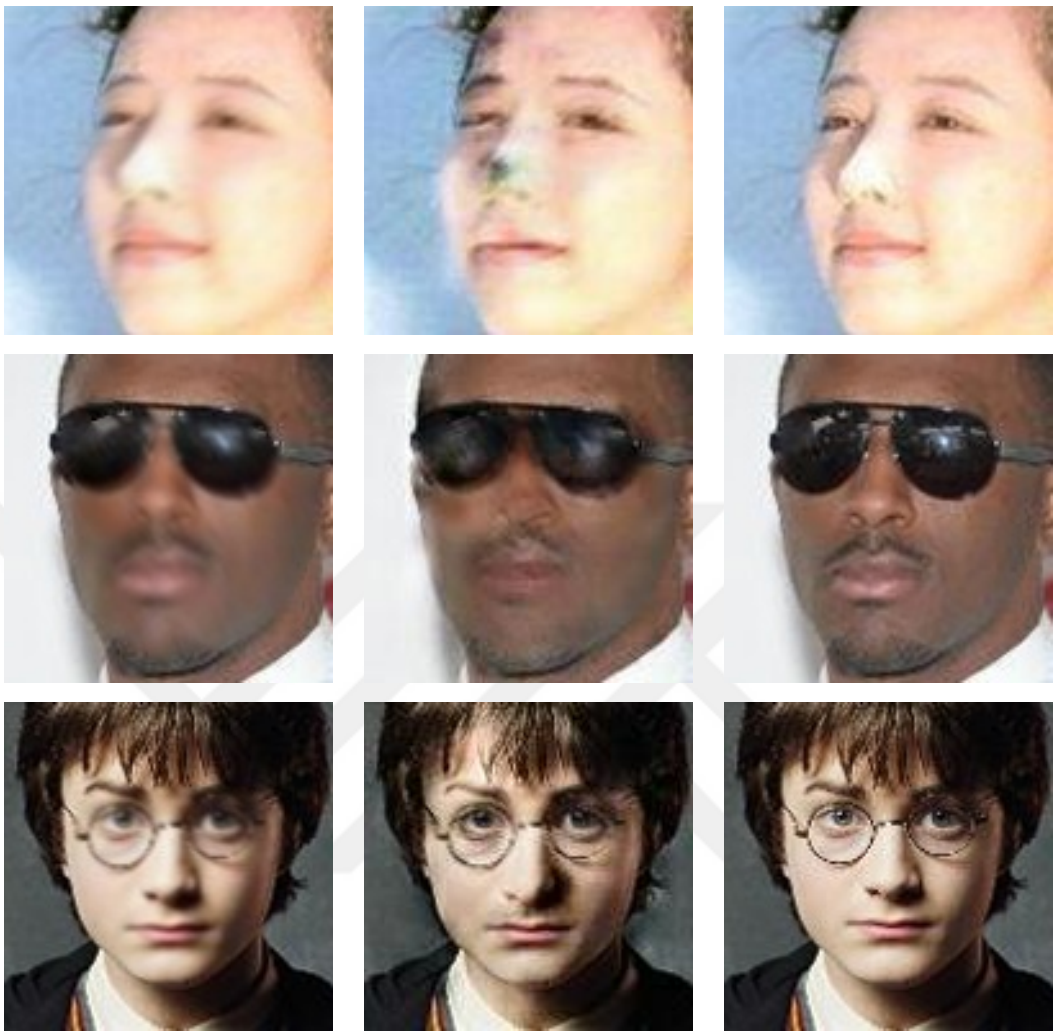
Table 5.4. Hospisign quantitative results

Method \ Metric	IS	FID	KID
Kupyn *et al.* [12]	3.214193	27.969915	0.015117
Kupyn *et al.* [62]	3.163555	24.333662	0.010501
Zhu *et al.* [77]	3.062458	**19.875273**	**0.006237**
Lu *et al.* [10]	**3.438400**	91.979790	0.081881
Ours	3.325758	49.791279	0.031744

Table 5.5. Hospisign ablation study on how the use of partial convolution in the blur encoder improves evaluation metrics.

Method	IS	FID	KID
Base Model	3.438400	91.979790	0.081881
+ Conv.	**3.557228**	86.352165	0.071378
+ Par-Conv	3.422889	57.677235	0.038770
+ Cond-Disc.	3.325758	**49.791279**	**0.031744**

enhancements to FID and KID which shows that attending the degraded regions allows for more accurate blur feature disentanglement. The last model is the proposed model of this work and demonstrates an even further boost for the metrics by conditioning the blur domain discriminator D_{blur} on blur features extracted by E_{blur}. It is demonstrated that the largest improvement is obtained via partial convolutions during blur feature extraction.

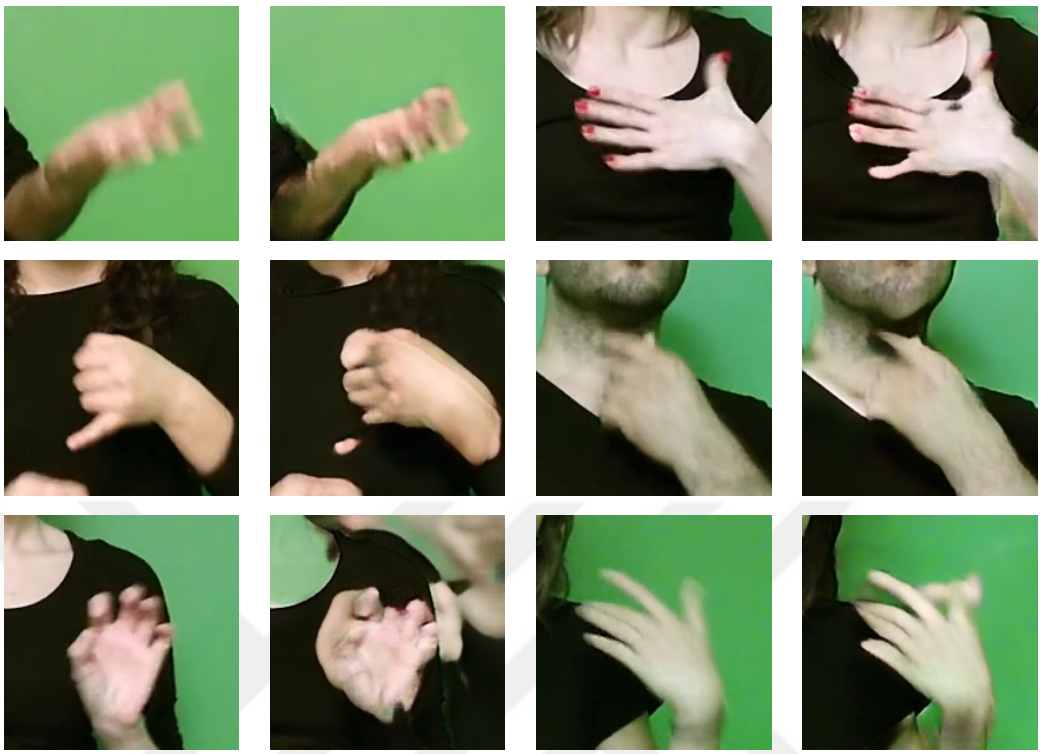


Blurred Deblurred Sharp

Figure 5.4. Failure cases on partially-blurred CelebA test set. The middle column illustrates the deblurring results while the first and last columns are the corresponding blurred and sharp images, respectively.

5.3.5. Failure Cases

5.3.5.1. CelebA. Some poorly deblurred results from the CelebA [5] dataset can be seen in Figure 5.4. Our network sometimes produces unwanted artifacts when the input image contains bright spots (e.g. Figure 5.4 top row) or if there are glasses present in the frame.



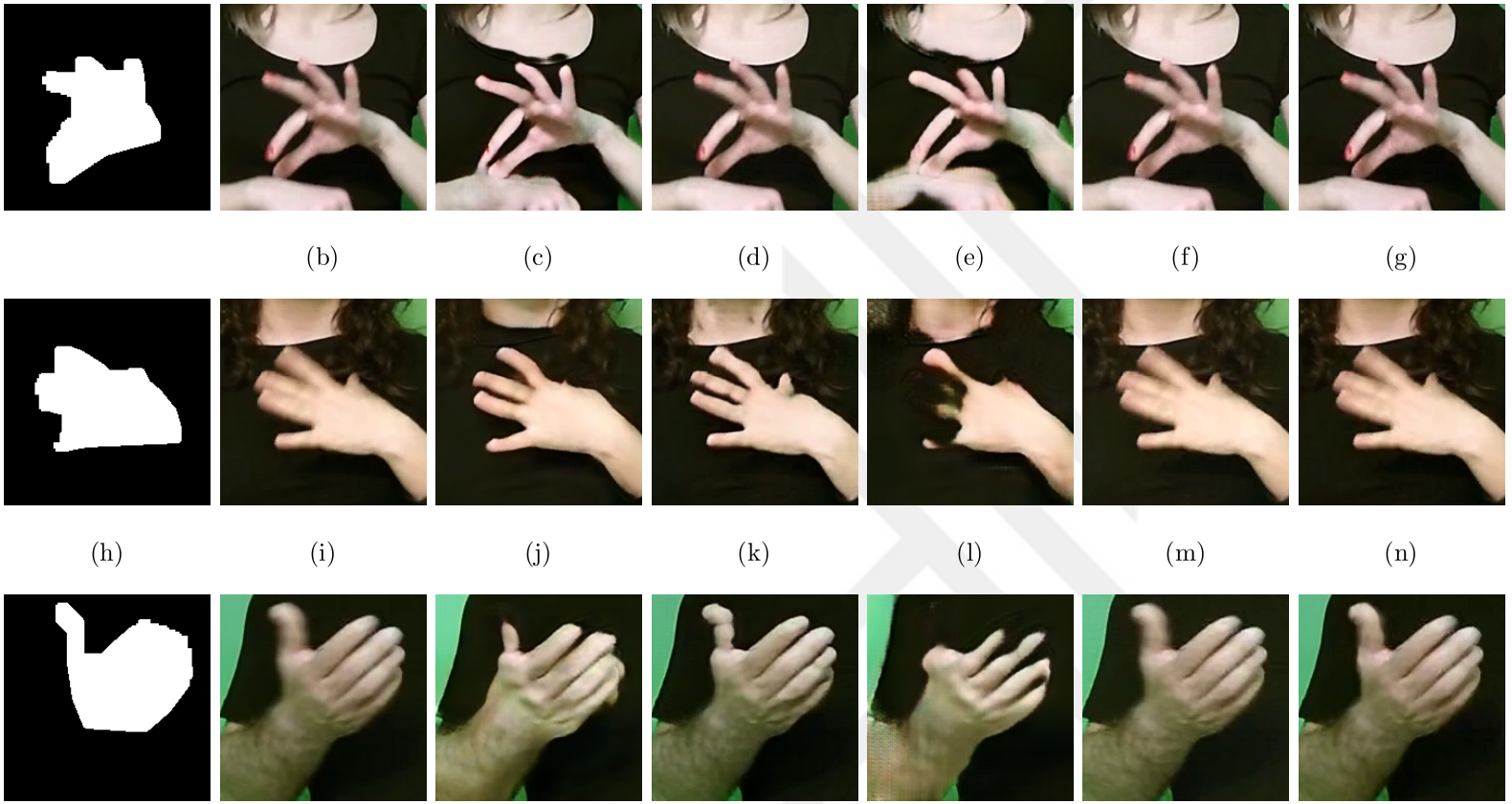
Blurry Deblurred Blurry Deblurred

Figure 5.5. Failure cases on the proposed novel hand test set. First and third columns contain blurry images while second and fourth columns have the corresponding deblurring result.

5.3.5.2. Hand. Some unsuccessful deblurring results may be seen in Figure 5.5 for our novel hand dataset. The network sometimes produces unwanted extra skin regions or erases parts of the fingers.

5.3.6. Additional Results

We showcase further qualitative results for HospiSign in Figure 5.6 and CelebA in Figure 5.7, with extensive comparisons between other methods.

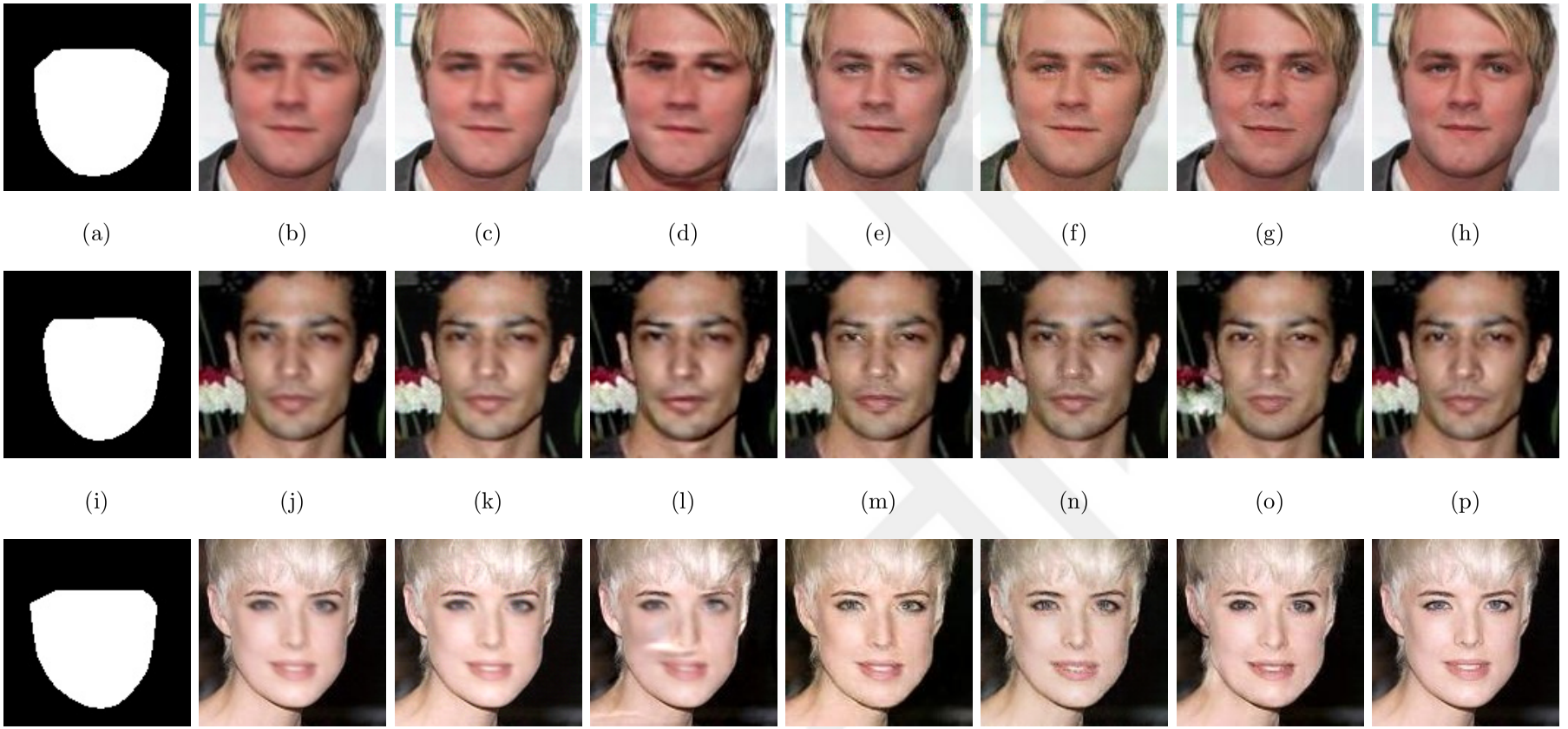


(b) (c) (d) (e) (f) (g)

(h) (i) (j) (k) (l) (m) (n)

(o) Blur Region (p) Blurred (q) Ours (r) [77] (s) [10] (t) [12] (u) [62]

Figure 5.6. Qualitative test results on novel blur dataset. The first column shows the mask for the hand area. Best viewed in color by zooming in.



(a) (b) (c) (d) (e) (f) (g) (h)

(i) (j) (k) (l) (m) (n) (o) (p)

(q) Blur Region (r) Blurred (s) [12] (t) [62] (u) [10] (v) [77] (w) Ours (x) Sharp

Figure 5.7. Some results from state-of-the-art works on partially blurred CelebA dataset. The first column shows the mask that was used to blur the facial region in each image. Best viewed in color by zooming in.

6. CONCLUSION AND FUTURE WORK

In this thesis, we propose a novel method for deblurring under spatially varying motion blur. The method is based on disentangling blur from images using unpaired blurry and sharp image pairs. We use cross-cycle consistency losses in blurring/deblurring branches based on disentangled representations and extract spatially-variant blur from blur-degraded regions using partial convolutions. We first test our method on the CelebA dataset with synthetically generated partial blur. Our method achieves superior or comparable performance with state-of-the art deblurring methods. The second, more challenging dataset is generated from sign language videos containing severe hand motion blur. Our method achieves satisfactory performance on this challenging dataset.

The introduction of deep learning techniques to deblurring led researchers to creating synthetically blurred datasets to supply the need for datasets with a large number of samples. Unfortunately, mimicking the degradation caused by object motion is quite complex and it was evident that learning from synthetic datasets and testing on real blurry data yields inferior blur elimination performances. With our proposed hand deblurring dataset containing real sharp and blurry samples, we aim to take a step in minimizing this need.

HospiSign consists of hand images that are corrupted by object motion blur around hand regions: This causes partial blur degradation resulting in blurry foregrounds and sharp backgrounds. Inspired by earlier blind methods that split the partial blur deblurring task into a user-guided blurry region mask extraction and a consequent blur kernel estimation from those areas, we follow a similar blur feature extraction scheme from masked regions. For this blur feature extractor, we use partial convolutions instead of regular convolutional layers to provide a normalization with respect to changing hand mask areas. To examine the performance of the proposed partial convolutions against regular convolutions, we conduct an ablation study where

we report results on several evaluation metrics. Starting with a base deblurring model, we separately add partial convolutions and regular convolutions to the blur encoder: It was shown that partial convolutions achieve a significantly greater enhancement against regular convolutions in terms of our metrics.

Our future work will focus on the improvement of sign language recognition performance with deblurring. So far, we have worked on single hand crops with blur from sign language video frames. An exciting direction to follow may be to treat head, right hand and left hand regions separately. Learning specialized deblurring models for each area may improve various sign language tasks. Moreover, supporting deblurring with hand pose estimation information may enhance performance as we have observed that deblurring removes blurry digits rather than creating a sharpening effect in some cases. This issue arises when in the presence of severe blur and may be alleviated by providing missing information in the form of hand landmarks.

REFERENCES

1. Lai, W.-S., J.-B. Huang, Z. Hu, N. Ahuja and M.-H. Yang, "A Comparative Study for Single Image Blind Deblurring", *IEEE Conferene on Computer Vision and Pattern Recognition*, 2016.

2. Shi, J., L. Xu and J. Jia, "Discriminative blur detection features", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2965–2972, 2014.

3. Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, "Generative adversarial nets", *Advances in neural information processing systems*, pp. 2672–2680, 2014.

4. Radford, A., L. Metz and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks", *arXiv preprint arXiv:1511.06434*, 2015.

5. Liu, Z., P. Luo, X. Wang and X. Tang, "Deep learning face attributes in the wild", *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

6. Pan, J., Z. Hu, Z. Su, H.-Y. Lee and M.-H. Yang, "Soft-segmentation guided object motion deblurring", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 459–468, 2016.

7. Dai, S. and Y. Wu, "Removing partial blur in a single image", *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2544–2551, June 2009.

8. Noroozi, M., A. Vinjimoor, P. Favaro and H. Pirsiavash, "Boosting self-supervised learning via knowledge transfer", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9359–9367, 2018.

9. Isola, P., J.-Y. Zhu, T. Zhou and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks", *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, July 2017.

10. Lu, B., J.-C. Chen and R. Chellappa, "Unsupervised domain-specific deblurring via disentangled representations", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10225–10234, 2019.

11. Wang, R. and D. Tao, "Recent Progress in Image Deblurring", *ArXiv*, Vol. abs/1409.6838, 2014.

12. Kupyn, O., V. Budzan, M. Mykhailych, D. Mishkin and J. Matas, "DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks", *2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8183–8192, June 2018.

13. Buehler, P., "Automatic learning of British sign language from signed tv broadcasts", , 2010.

14. Donahue, J. and K. Simonyan, "Large scale adversarial representation learning", *arXiv preprint arXiv:1907.02544*, 2019.

15. Yu, J., Z. Lin, J. Yang, X. Shen, X. Lu and T. S. Huang, "Generative image inpainting with contextual attention", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5505–5514, 2018.

16. Ledig, C., L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017.

17. Nah, S., T. H. Kim and K. M. Lee, "Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring", *The IEEE Conference on Computer Vision*

and Pattern Recognition (CVPR), July 2017.

18. Liu, G., F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions", *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 85–100, 2018.

19. Yang, X., Z. Xu and J. Luo, "Towards perceptual image dehazing by physics-based disentanglement and adversarial training", *Thirty-second AAAI conference on artificial intelligence*, 2018.

20. Madam, N. T., S. Kumar and A. N. Rajagopalan, "Unsupervised Class-Specific Deblurring", V. Ferrari, M. Hebert, C. Sminchisescu and Y. Weiss (Editors), *Computer Vision – ECCV 2018*, pp. 358–374, Springer International Publishing, Cham, 2018.

21. Samangouei, P., M. Kabkab and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models", *arXiv preprint arXiv:1805.06605*, 2018.

22. Arjovsky, M., S. Chintala and L. Bottou, "Wasserstein gan", *arXiv preprint arXiv:1701.07875*, 2017.

23. Gulrajani, I., F. Ahmed, M. Arjovsky, V. Dumoulin and A. Courville, "Improved Training of Wasserstein GANs", *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 5769–5779, Curran Associates Inc., USA, 2017, `http://dl.acm.org/citation.cfm?id=3295222.3295327`.

24. Levin, A., Y. Weiss, F. Durand and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms", *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1964–1971, IEEE, 2009.

25. Cho, S., J. Wang and S. Lee, "Handling outliers in non-blind image deconvolution",

Proceedings of the IEEE International Conference on Computer Vision, pp. 495–502, 11 2011.

26. Krishnan, D. and R. Fergus, "Fast Image Deconvolution using Hyper-Laplacian Priors", *Advances in Neural Information Processing Systems 22 - Proceedings of the 2009 Conference*, pp. 1033–1041, 01 2009.

27. Wiener, N., *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*, MIT Press, 1949.

28. Lucy, L. B., "An iterative technique for the rectification of observed distributions", *The astronomical journal*, Vol. 79, p. 745, 1974.

29. Zoran, D. and Y. Weiss, "From Learning Models of Natural Image Patches to Whole Image Restoration", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 479–486, 11 2011.

30. Schuler, C. J., M. Hirsch, S. Harmeling and B. Schölkopf, "Learning to Deblur", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, pp. 1439–1451, 2014.

31. Dong, J., J. Pan, D. Sun, Z. Su and M.-H. Yang, "Learning Data Terms for Non-blind Deblurring", V. Ferrari, M. Hebert, C. Sminchisescu and Y. Weiss (Editors), *Computer Vision – ECCV 2018*, pp. 777–792, Springer International Publishing, Cham, 2018.

32. Schmidt, U., C. Rother, S. Nowozin, J. Jancsary and S. Roth, "Discriminative non-blind deblurring", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 604–611, 2013.

33. Richardson, W. H., "Bayesian-based iterative method of image restoration", *JoSA*, Vol. 62, No. 1, pp. 55–59, 1972.

34. Vasu, S., V. Reddy Maligireddy and A. Rajagopalan, "Non-blind Deblurring: Handling Kernel Uncertainty with CNNs", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3272–3281, 2018.

35. Levin, A., R. Fergus, F. Durand and W. T. Freeman, "Deconvolution using natural image priors", *Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory*, Vol. 3, 2007.

36. Yuan, L., J. Sun, L. Quan and H.-Y. Shum, "Image deblurring with blurred/noisy image pairs", *ACM Transactions On Graphics (TOG)*, Vol. 26, No. 3, p. 1, 2007.

37. Bar, L., A. Brook, N. Sochen and N. Kiryati, "Deblurring of color images corrupted by impulsive noise", *IEEE Transactions on Image Processing*, Vol. 16, No. 4, pp. 1101–1111, 2007.

38. Xu, L. and J. Jia, "Two-phase kernel estimation for robust motion deblurring", *European conference on computer vision*, pp. 157–170, Springer, 2010.

39. Cho, S. and S. Lee, "Fast Motion Deblurring", *ACM Trans. Graph.*, Vol. 28, No. 5, pp. 145:1–145:8, Dec. 2009, `http://doi.acm.org/10.1145/1618452.1618491`.

40. Joshi, N., R. Szeliski and D. Kriegman, "PSF Estimation using Sharp Edge Prediction", *IEEE Conference on Computer Vision and Pattern Recognition*, 06 2008.

41. Krishnan, D., T. Tay and R. Fergus, "Blind deconvolution using a normalized sparsity measure", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 233 – 240, 07 2011.

42. Yan, Y., W. Ren, Y. Guo, R. Wang and X. Cao, "Image deblurring via extreme channels prior", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4003–4011, 2017.

43. Li, L., J. Pan, W.-S. Lai, C. Gao, N. Sang and M.-H. Yang, "Learning a discrim-

inative prior for blind image deblurring", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6616–6625, 2018.

44. Wipf, D. and H. Zhang, "Revisiting Bayesian blind deconvolution", *The Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 3595–3634, 2014.

45. Shan, Q., J. Jia and A. Agarwala, "High-quality motion deblurring from a single image", *Acm transactions on graphics (tog)*, Vol. 27, No. 3, p. 73, 2008.

46. Wang, Z., A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity", *IEEE transactions on image processing*, Vol. 13, No. 4, pp. 600–612, 2004.

47. Palmer, J., K. Kreutz-Delgado, B. D. Rao and D. P. Wipf, "Variational EM algorithms for non-Gaussian latent variable models", *Advances in neural information processing systems*, pp. 1059–1066, 2006.

48. Levin, A., Y. Weiss, F. Durand and W. T. Freeman, "Efficient marginal likelihood optimization in blind deconvolution", *CVPR 2011*, pp. 2657–2664, IEEE, 2011.

49. Babacan, S. D., R. Molina, M. N. Do and A. K. Katsaggelos, "Bayesian blind deconvolution with general sparse image priors", *European Conference on Computer Vision*, pp. 341–355, Springer, 2012.

50. Köhler, R., M. Hirsch, B. Mohler, B. Schölkopf and S. Harmeling, "Recording and Playback of Camera Shake: Benchmarking Blind Deconvolution with a Real-World Database", A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato and C. Schmid (Editors), *Computer Vision – ECCV 2012*, pp. 27–40, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

51. Wieschollek, P., M. Hirsch, B. Scholkopf and H. Lensch, "Learning blind motion deblurring", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 231–240, 2017.

52. Michaeli, T. and M. Irani, "Blind Deblurring Using Internal Patch Recurrence", D. Fleet, T. Pajdla, B. Schiele and T. Tuytelaars (Editors), *Computer Vision – ECCV 2014*, pp. 783–798, Springer International Publishing, Cham, 2014.

53. Whyte, O., J. Sivic, A. Zisserman and J. Ponce, "Non-uniform Deblurring for Shaken Images", *International Journal of Computer Vision*, Vol. 98, No. 2, pp. 168–186, Jun 2012, https://doi.org/10.1007/s11263-011-0502-7.

54. Levin, A., D. Lischinski and Y. Weiss, "A Closed-Form Solution to Natural Image Matting", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 30, pp. 228–42, 03 2008.

55. Dai, S. and Y. Wu, "Motion from blur", *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2008.

56. Levin, A., "Blind Motion Deblurring Using Image Statistics.", Vol. 19, pp. 841–848, 01 2006.

57. Hyun Kim, T., B. Ahn and K. Mu Lee, "Dynamic Scene Deblurring", *2013 IEEE International Conference on Computer Vision*, pp. 3160–3167, Dec 2013.

58. LeCun, Y., Y. Bengio and G. Hinton, "Deep learning", *Nature*, Vol. 521, No. 7553, pp. 436–444, 2015.

59. Schuler, C., H. Burger, S. Harmeling and B. Scholkopf, "A Machine Learning Approach for Non-blind Image Deconvolution", *Proceedings / CVPR, IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1067–1074, 06 2013.

60. Xu, L., J. S. Ren, C. Liu and J. Jia, "Deep convolutional neural network for image deconvolution", *Advances in neural information processing systems*, pp. 1790–1798, 2014.

61. Noroozi, M., P. Chandramouli and P. Favaro, "Motion Deblurring in the Wild",

V. Roth and T. Vetter (Editors), *Pattern Recognition*, pp. 65–77, Springer International Publishing, Cham, 2017.

62. Kupyn, O., T. Martyniuk, J. Wu and Z. Wang, "DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better", *ArXiv*, Vol. abs/1908.03826, 2019.

63. Ramakrishnan, S., S. Pachori, A. Gangopadhyay and S. Raman, "Deep Generative Filter for Motion Deblurring", *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 2993–3000, Oct 2017.

64. Lee, H.-Y., H.-Y. Tseng, Q. Mao, J.-B. Huang, Y.-D. Lu, M. Singh and M.-H. Yang, "DRIT++: Diverse Image-to-Image Translation via Disentangled Representations", *arXiv preprint arXiv:1905.01270*, 2019.

65. Simonyan, K. and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, 2014.

66. Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database", *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

67. Camgöz, N. C., A. A. Kındıroğlu, S. Karabüklü, M. Kelepir, A. S. Özsoy and L. Akarun, "BosphorusSign: A Turkish Sign Language Recognition Corpus in Health and Finance Domains", *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 1383–1388, European Language Resources Association (ELRA), Portorož, Slovenia, May 2016, https://www.aclweb.org/anthology/L16-1220.

68. Cao, Z., T. Simon, S.-E. Wei and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7291–7299, 2017.

69. Adams, R. and L. Bischof, "Seeded region growing", *IEEE Transactions on Pattern*

Analysis and Machine Intelligence, Vol. 16, No. 6, pp. 641–647, June 1994.

70. Salimans, T., I. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, "Improved techniques for training gans", *Advances in neural information processing systems*, pp. 2234–2242, 2016.

71. Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

72. Heusel, M., H. Ramsauer, T. Unterthiner, B. Nessler and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium", *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.

73. Binkowski, M., D. J. Sutherland, M. Arbel and A. Gretton, "Demystifying MMD GANs", *ArXiv*, Vol. abs/1801.01401, 2018.

74. Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning", *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

75. Unlu, G., *Spatially Varying Motion Deblurring*, 2019, `https://github.com/unlugi`, accessed in December 2019.

76. Kingma, D. P. and J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

77. Zhu, J.-Y., T. Park, P. Isola and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.