Unsupervised Class-Specific Deblurring

Nimisha Thekke Madam[0000−0003−1945−1189], Sunil Kumar[0000−0001−7476−0537], and Rajagopalan A.N[0000−0002−0006−6961]

Indian Institute of Technology, Madras, India
ee13d037@ee.iitm.ac.in
http://www.ee.iitm.ac.in/ipcvlab/

Abstract. In this paper, we present an end-to-end deblurring network designed specifically for a class of data. Unlike the prior supervised deep-learning works that extensively rely on large sets of paired data, which is highly demanding and challenging to obtain, we propose an unsupervised training scheme with unpaired data to achieve the same. Our model consists of a Generative Adversarial Network (GAN) that learns a strong prior on the clean image domain using adversarial loss and maps the blurred image to its clean equivalent. To improve the stability of GAN and to preserve the image correspondence, we introduce an additional CNN module that reblurs the generated GAN output to match with the blurred input. Along with these two modules, we also make use of the blurred image itself to self-guide the network to constrain the solution space of generated clean images. This self-guidance is achieved by imposing a scale-space gradient error with an additional gradient module. We train our model on different classes and observe that adding the reblur and gradient modules helps in better convergence. Extensive experiments demonstrate that our method performs favorably against the state-of-the-art supervised methods on both synthetic and real-world images even in the absence of any supervision.

Keywords: Motion blur · deblur · reblur · unsupervised learning · GAN · CNN.

1 Introduction

Blind-image deblurring is a classical image restoration problem which has been an active area of research in image and vision community over the past few decades. With increasing use of hand-held imaging devices, especially mobile phones, motion blur has become a major problem to confront with. In scenarios where the light present in the scene is low, the exposure time of the sensor has to be pumped up to capture a well-lit scene. As a consequence, camera shake becomes inevitable resulting in image blur. Motion blur also occurs when the scene is imaged by fast-moving vehicles such as cars and aircrafts even in low-exposure settings. The problem escalates further in data-deprived situations comprising of only a single blurred frame.

Blind-deblurring can be posed as an image-to-image translation where given a blurred image y in blur domain, we need to learn a non-linear mapping $\mathcal{M}$:y

$\rightarrow x$ that maps the blurred image to its equivalent clean image x in the clean domain. Many recent deep learning based deblurring networks [27, 28, 18] estimate this mapping when provided with large sets of $\{y_i, x_i\}_{i=1}^{N}$ paired training data. Even though these networks have shown promising results, the basic assumption of availability of paired data is too demanding. In many a situation, collecting paired training data can be difficult, time-consuming and expensive. For example, in applications like scene conversion from day to night and image dehazing, the availability of paired data is scarce or even non-existent.

This debilitating limitation of supervised deep networks necessitates the need for unsupervised learning approaches [42, 41, 21] from unpaired datasets. In an unsupervised setting, the user collects two sets of images from two marginal distributions in both domains but sans pair-wise correspondences.Then the task is to infer the joint distribution using these images. In this paper, we aim to develop an unsupervised learning framework for blind-deblurring from a single blurred frame *without the need for the corresponding ground truth clean data*. Rather, our network relies on unlabeled image data from blur and clean domains to perform domain-specific deblurring.

Related works: There is a vast literature on motion deblurring spanning both conventional and deep learning techniques. Similarly, of late there are works on unsupervised image translations gaining popularity due to lack of availability of paired data. We provide a brief description of these two topics below.

Motion deblurring is a long-studied topic in imaging community. To avoid shot noise due to low amount of available photons in low light scenarios, the exposure time is increased. Hence, even a small camera motion is enough to create motion blur in the recorded image due to averaging of light energy from slightly different versions of the same scene. While there are several deblurring works that involve usage of multiple frames [35, 24], the problem becomes very ill-posed in data-limited situations where the user ends up with a single blurred frame. This entails the need for single-image blind-deblurring algorithms.

To overcome the ill-posedness of single image-blind deblurring, most of the existing algorithms [31, 39, 11] rely on image heuristics and assumptions on the sources of the blur. The most widely used image heuristics are sparsity prior, the unnatural l_0 prior [39] and dark channel prior [31]. Assumptions on camera motion are imposed in the form of kernel sparsity and smoothness of trajectory. These heuristics are used as priors and iterative optimization schemes are deployed to solve for camera motion and latent clean frame from a single-blurred input. Even though these methods are devoid of any requirement of paired data, they are highly dependent on the optimization techniques and prior selection.

With deep learning coming to the forefront, several deep networks [27, 28, 18] have been proposed that perform the task of blind deblurring from a single image. These methods work end-to-end and skip the need for the camera motion estimation and directly provide the clean frame when fed with the blurred image thus overcoming the tedious task of prior selection and parameter tuning. But the main disadvantage with existing deep-learning works is that they require close supervision warranting large amounts of paired datasets for training.

Unsupervised learning: The recent trend in deep learning is to use unpaired data to achieve domain transfer. With the seminal work of Goodfellow [10], GANs have been used in multiple areas of image-to-image translations. The key to this success is the idea of an adversarial loss that forces the generated images to be indistinguishable from real images thus learning the data domain. Conditional GANs (cGAN) [20, 15, 40] have made progress recently for cross-domain image-to-image translation in supervised settings. The goal remains the same in unsupervised settings too i.e; to relate the two domains. One way to approach the problem is by enforcing a common representation across the domains by using shared weights with two GANs as in [21, 3, 22]. The fundamental objective here is to use a pair of coupled GANs, one for the source and one for the target domain, whose generators share their high-layer weights and whose discriminators share their low-layer weights. In this manner, they are able to generate invariant representations which can be used for unsupervised domain transfer.

Following this, the works in [41, 42] propose to use a cycle consistency loss on the image space itself rather than asking for invariant feature space. Here too the GANs are used to learn each individual domain and then cross model term with cyclic consistency loss is used to map between domains. Apart from these methods, there are neural style transfer networks [6, 16, 7] that is also used for image-to-image translation with unsupervised data. The idea here is to combine the 'content' features of one image with the 'style' of another image (like famous paintings). These methods use matching of Gram matrix statistics of pre-trained deep features to achieve image translation between two specific images. On the other hand, our main focus is to learn the mapping between two image collections (rather than two specific images) from different domains by attempting to capture correspondences between higher-level appearance structures.

Class-specific Methods: Of late, domain-specific image restoration methods [5, 33, 40, 1, 2, 36, 37] are gaining relevance and attracting attention due to the inaccuracy of generic algorithms to deal with real-world data. The general priors learned from natural images are not necessarly well-suited for all classes and often lead to deterioration in performance. Recently, class-specific information has been employed in carrying out deblurring which outperforms blanket prior-based approaches. An exemplar-based deblurring for faces was proposed by Pan et al. in [29]. Anwar et al. [1] introduced a method to restore attenuated image frequencies during convolution using class-specific training examples. Deep learning networks too have attempted the task of class-specific deblurring. Text deblurring network in [12] and deep face deblurring network in [5] are a notable few amongst these.

Following these works, we also propose in this paper a domain-specific deblurring architecture focusing mainly on face, text, and checkerboard classes using a single GAN framework. Faces and texts are considered important classes and many restoration techniques have focused on them explicitly. We also included the checkerboard class to study our network performance and to ease the task of parameter tuning akin to [33]. GAN is used in our network to learn a strong class-specific prior on clean data. The discriminator thus learned captures the semantic domain knowledge of a class but fails to capture the content, colors, and

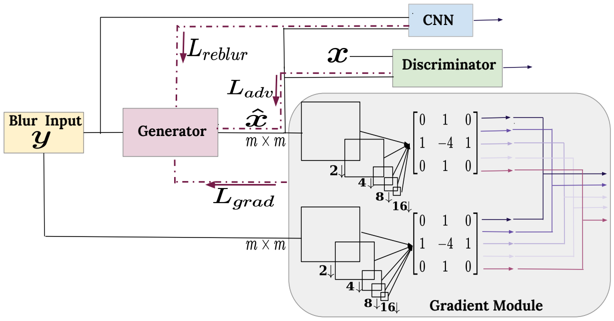


Fig. 1. Our network with GAN, reblur module and scale-space gradient module.

structure properly. These are usually corrected with supervised loss functions in regular networks which is not practical in our unsupervised setting. Hence, we introduce self-guidance using the blurred data itself. Our network is trained with unpaired data from clean and blurred domains. A comprehensive diagram of our network is shown in Fig.1.
The main contributions of our work are

- To the best of our knowledge, this is the first ever data-driven attempt at unsupervised learning for the task of deblurring.
- To overcome the shortcomings of supervision due to unavailability of paired data and to help the network converge to the right solution, we propose self-guidance with two new additional modules
  - A self-supervised reblurring module that guides the generator to produce a deblurred output corresponding to the input blurred image.
  - A gradient module with the key notion that down-sampling decreases gradient matching error and constrains the solution space of generated clean images.

2 Unsupervised deblurring

A naive approach to unsupervised deblurring would be to adopt existing networks (CoGAN [22], DualGAN [41], CycleGAN [42]) designed for image translations and train them for the task of image restoration. However, a main issue with such an approach is that most of the unsupervised networks discussed thus far are designed for a specific task of domain transformation such as face-to-sketch synthesis, day-to-night etc where the transformations are well-defined. In image deblurring, the transformation from blur to clean domain is a many-to-one mapping while clean to blur is the vice versa depending on the extent and nature of blur. Thus, it is difficult to capture the domain knowledge with these existing architectures (see experiments section for more on this). Also, the underlying idea in all these networks is to use a pair of GANs to learn the domains, but usually training GANs is highly unstable [34, 8] and thus using two GANs simultaneously escalates in stability issues in the network. Instead of using a second GAN to learn the blur domain, we use a CNN network for reblurring the output of GAN and a gradient module to constrain the solution space. A detailed description of each module is provided below.

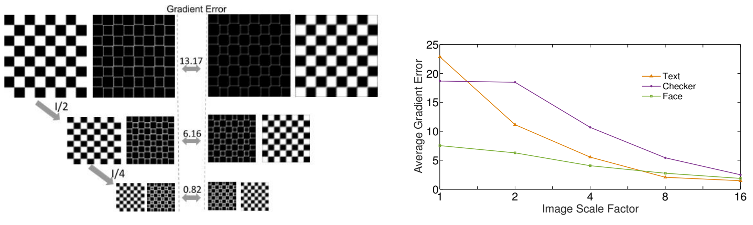
GAN proposed by Goodfellow [10] consists of two networks (a generator and a discriminator) that compete to outperform each other. Given the discriminator D, the generator tries to learn the mapping from noise to real data distribution so as to fool D. Similarly, given the generator G, the discriminator works as a classifier that learns to distinguish between real and generated images. The function of learning GAN is a min-max problem with the cost function

$$E(D,G) = \max_{D}\min_{G} \underset{x\sim P_{data}}{E}[\log D(x)] + \underset{z\sim P_z}{E}[\log(1-D(G(z)))]. \tag{1}$$

where z is random noise and x denotes the real data. This work was followed by conditional GANs (cGAN) [26] that use a conditioning input in the form of image [15], text, class label etc. The objective remains the same in all of these i.e, the discriminator is trained to designate higher probability to real data and lower to the generated data. Hence, the discriminator acts as a data prior that learns clean data domain similar to the heuristics that are used in conventional methods. This motivated us to use GANs for learning the mapping from blur to clean domain using the discriminator as our data prior. In our network, the input to generator G is a blurred image $y \in Y$ and the generator maps it to a clean image $\hat{x}$ such that the generated image $\hat{x} = G(y)$ is indistinguishable from clean data (where clean data statistics are learned from $\tilde{x}$s $\in X$).

Self-supervision by reblurring (CNN Module) The goal of GAN in our deblurring framework is to reach an equilibrium where P_{clean} and $P_{generated}$ are close. The alternating gradient update procedure (AGD) is used to achieve this. However, this process is highly unstable and often results in mode collapse [9]. Also, an optimal G that translates from $Y \rightarrow X$ does not guarantee that an individual blurred input y and its corresponding clean output x are paired up in a meaningful way, i.e, there are infinitely many mappings G that will induce the same distribution over $\hat{x}$ [42]. This motivated the use of reconstruction loss ($||\hat{x} - x||^2$) and perceptual loss ($||\Phi_i((\hat{x}) - \Phi_i(x)||^2$, where Φ_i represents VGG module features extracted at the i^{th} layer) along with the adversarial loss in many supervised learning works [27, 20, 15, 40, 38], to stabilize the solution and help in better convergence. But, these cost functions require high level of supervision in the form of ground truth clean reference images (x) which are not available in our case. This restricts the usage of these supervised cost functions in our network. To account for the unavailability of paired dataset, we use the blurred image y itself as a supervision to guide in deblurring. Ignatov et al. [14] have used a similar reblurring approach with a constant Gaussian kernel to correct for colors in camera mapping. We enforce the generator to produce result ($\hat{x}$) that when reblurred using the CNN module will furnish back the input. Adding such a module ensures that the deblurred result has the same color and texture comparable to the input image thereby constraining the solution to the manifold of images that captures the actual input content.

Gradient matching Module With a combined network of GAN and CNN modules, the generator learns to map to clean domain along with color preservation. Now, to enforce the gradients of the generated image to match its corresponding clean image, a **gradient module** is used in our network as given in Fig. 1. Gradient matching resolves the problem of over-sharpening and ringing in the results. However, since we do not have access to the reference image, determining the desired gradient distribution to match with is difficult. Hence, we borrow a heuristic from [25] that takes advantage of the fact that shrinking a blurry image y by a factor of α results in a image y^{α} that is α times sharper than y. Thus, we use the blurred image gradients at different scales to guide the deblurring process. At the highest scales, the gradients of blurred and generated output match the least but improve while going down in scale space. A visual diagram depicting this effect is shown in Fig. 2(a) where the gradients of a blurred and clean checker-board at different scales are provided. Observe that, at the highest scale, the gradients are very different and as we move down in scale the gradients start to look alike and the L_1 error between them decreases. The plot in Fig. 2(b) is the average per pixel L_1 error with respect to scale for 200 images from each of text, checker-board and face datasets. In all these data, the gradient error decreases with scale and hence forms a good guiding input for training our network.



(a) (b)

Fig. 2. (a) Scale space gradient error. (b) Average decrease in gradient error with respect to down scaling.

3 Loss Functions

A straightforward way for unsupervised training is by using GAN. Given large unpaired data $\{x_i\}_{i=1}^{M}$ and $\{y_j\}_{j=1}^{N}$ in both domains, train the parameters (θ) of the generator to map from $y \rightarrow x$ by minimizing the cost

$$L_{\text{adv}} = \min_{\theta} \frac{1}{N} \sum_{i} \log(1 - D(G_{\theta}(y_i))) \tag{2}$$

Training with adversarial cost alone can result in color variations or missing finite details (like eyes and nose in faces or letters in case of texts) in the generated

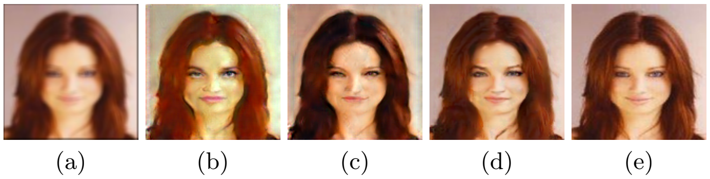


Fig. 3. Effect of different cost functions. (a) Input blurred image to the generator, (b) result of unsupervised deblurring with just the GAN cost in eq. (2), (c) result obtained by adding the reblurring cost in eq. (3) with (b), (d) result obtained with gradient cost in eq. (4) with (c), and (e) the target output.

outputs but the discriminator can still end up classifying it as real instead of generated data. This is because discriminating between real and fake does not depend on these small details (see Fig. 3(b), the output of GAN alone wherein eyes and colors are not properly reconstructed).

With the addition of the reblurring module, the generator is more constrained to match the colors and textures of the generated data (see Fig. 3(c)). The generated clean image from generator $\hat{x} = G(y)$ is again passed through the CNN module to obtain back the blurred input. Hence the reblurring cost is given as

$$L_{\text{reblur}} = ||y - \text{CNN}(\hat{x})||_2^2 \tag{3}$$

Along with the above two costs, we also enforce the gradients to match at different scales (s) using the gradient cost defined as

$$L_{\text{grad}} = \sum_{s \in \{1,2,4,8,16\}} \lambda_s |\nabla y_{s\downarrow} - \nabla \hat{x}_{s\downarrow}| \tag{4}$$

where ∇ denotes the gradient operator. A Laplacian operator $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ is used to calculate the image gradients at different scales and λ_s values are set as [0.0001, 0.001, 0.01, 0.1, 1] for $s = \{1, 2, 4, 8, 16\}$, respectively. Adding the gradient cost removes unwanted ringing artifacts at the boundary of the image and smoothens the result. It is evident from the figure that with inclusion of supporting cost functions corresponding to reblurring and gradient, the output (Fig. 3(d)) of the network becomes comparable with the ground truth (GT) image (Fig. 3(e)). Hence, the generator network is trained with a combined cost function given by

$$L_G = \gamma_{\text{adv}} L_{\text{adv}} + \gamma_{\text{reblur}} L_{\text{reblur}} + \gamma_{\text{grad}} L_{\text{grad}} \tag{5}$$

4 Network Architecture

We followed a similar architecture for our generator and discriminator as proposed in [40], which has shown good performance for blind super-resolution,

Table 1. (a) The proposed generator and discriminator network architecture. conv ↓ indicates convolution with stride 2 which in effect reduces the output dimension by half and d/o refers to dropout. (b) Reblurring CNN module architecture

Module	**Generator**										**Discriminator**					
Layers	conv	conv	conv	conv	conv	conv	conv	conv	conv	conv	conv↓	conv↓	conv↓	conv↓	conv ↓	fc
Kernel Size	5	5	5	5	5	5	5	5	5	5	4	4	4	4	4	-
Features	64	128	128	256 d/o (0.2)	256 d/o (0.2)	128	128	64	64	3	64	128	256	512	512	-

(a)

Module	**CNN**					
Layers	conv	conv	conv	conv	conv	tanh
Kernel Size	5	5	5	5	5	
Features	64	64	64	64	3	

(b)

with slight modification in the feature layers. The network architecture of GAN with filter sizes and the number of feature maps at each stage is provided in Table 1(a). Each convolution (conv) layer in the generator is followed by batch-normalization and non-linearity using Rectified Linear Unit (ReLU) except the last layer. A hyper tangential (Tanh) function is used at the last layer to constrain the output to $[-1, 1]$. The discriminator is a basic 6-layer model with each convolution followed by a Leaky ReLU except the last fully connected (fc) layer which is followed by a Sigmoid. Convolution with stride 2 is used in most layers to go down in dimension and the details of filter size and feature maps are provided in Table 1(a). The reblurring CNN architecture is a simple 5-layer convolutional module provided in Table 1(b). The gradient module is operated on-the-fly for each batch of data using GPU based convolution with the Laplacian operator and downsampling depending on the scaling factor with '*nn*'modules.

We used Torch for training and testing with the following options: ADAM optimizer with momentum values $\beta_1 = 0.9$ and $\beta_2 = 0.99$, learning rate of 0.0005, batch-size of 32 and the network was trained with the total cost as provided in eq. (5). The weights for different costs were initially set as γ_{adv}=1, γ_{grad}=.001 and γ_{reblur}=0.01 to ensure that the discriminator learns the clean data domain. After around 100K iterations the adversarial cost was weighted down and the CNN cost was increased so that the clean image produced corresponds in color and texture to the blurred input. Hence, the weights were readjusted as γ_{adv}=0.01, γ_{grad}=0.1 and γ_{reblur}=1 and the learning rate was reduced to 0.0001 to continue training. Apart from these, to stabilize the GAN, during training we used drop-out of 0.2 at the fourth and fifth convolution layers of the generator and used a smooth labeling of real and fake labels following [34].

5 Experiments

The experiments section is arranged as follows: (i) training and testing datasets, (ii) comparison methods, (iii) quantitative results, metrics used and comparisons, and (iv) visual results and comparisons.

Table 2. Quantitative comparisons on face, text, and checkerboard datasets.

		Face dataset			Text dataset				Checkerboard dataset		
	Method	**PSNR**	**SSIM**	**KSM**	**PSNR**	**SSIM**	**KSM**	**CER**	**PSNR**	**SSIM**	**KSM**
Conventional Methods	Pan et al. [30]	-	-	-	16.19	0.7298	0.8628	0.4716	11.11	0.3701	0.7200
	Pan et al. [31]	19.38	0.7764	0.7436	17.48	0.7713	0.8403	0.3066	13.91	0.5618	0.7027
	Xu et al. [39]	20.28	0.7928	0.7166	14.22	0.5417	0.7991	0.2918	8.18	0.2920	0.6034
	Pan et al.[29]	22.36	0.8523	0.7197	-	-	-	-	-	-	-
Deep learning Methods	Nah et al. [27]	24.12	0.8755	0.6229	18.72	0.7521	0.7467	0.2643	18.07	0.6932	0.6497
	Hradiš et al. [12]	-	-	-	24.28	0.9387	0.9435	0.0891	18.09	0.6788	0.6791
Unsupervised technique	Zhu et al. [42]	8.93	0.4406	0.2932	13.19	0.5639	0.8363	0.2306	21.92	0.8264	0.6527
	Ours	22.80	0.8631	0.7536	23.22	0.8792	0.9376	0.126	20.61	0.8109	0.7801

5.1 Dataset Creation

For all classes, we used 128 × 128 sized images for training and testing. The dataset generation for training and testing of each of these classes is explained below. Note that our network was trained for each of these classes separately.

Camera Motion Generation: In our experiments, to generate the blur kernels required for synthesizing the training and test sets, we used the methodology described by Chakrabarthi in [4]. The blur kernels are generated by randomly sampling six points in a limited size grid (13 × 13), fitting a spline through these points, and setting the kernel values at each pixel on this spline to a value sampled from a Gaussian distribution with mean 1 and standard deviation of 0.5, then clipping these values to be positive, and normalizing the kernel to have unit sum. A total of 100K kernels were used for creating the dataset.

Face Dataset: We use the aligned CelebA face dataset [23] for creating the training data for our case. CelebA is a large-scale face attributes dataset of size 178 × 218 with more than 200K aligned celebrity images. We selected 200K images from it, resized each to 128 × 128 and divided it into two groups of 100K images each. Then, we use the blur kernels generated with [4] to blur one set of images alone and the other set is kept intact. This way, we generate the clean and blur face data (without any correspondence) for training the network.

Text Dataset: For text images, we use the training dataset of Hradiš et al. [12] which consists of images with both defocus blur generated by anti-aliased disc and motion blur generated by random walk. They have provided a large collection of 66K text images of size 300×300. We use these images for creating the training dataset and use the test data provided by them for testing our network. We first divide the whole dataset into two groups of 33K each with one group containing clean data alone and other containing the blurred data. We took care to avoid any overlapping pairs in the generated set. We then cropped 128×128 patches from these sets to obtain the training set of around 300K images in both clean and blur set.

Checkerboard Dataset: We took a clean checkerboard image of size 256 × 256 and applied random rotations and translations to it and cropped out 128 ×128 (avoiding boundary pixels) to generate a set of 100K clean images. The clean images are then partitioned into two sets of 50K images each to ensure that there are no corresponding pairs available during training. To one set we

Table 3. Quantitative comparisons on face and text on real handshake motion [17].

Class	PSNR in (dB)	SSIM	KSM
Text	21.92	0.8968	0.8811
Face	21.40	0.8533	0.7794

apply synthetic motion blur to create the blurred images by convolving with linear filters and the other set is kept as such. We used a linear approximation of camera motion and parametrized it with length l and rotation angle θ. For the dataset creation, considering the size of input images, we selected the maximum value of l to be in the range $[0, 15]$ and varied θ from $[0, 180]^o$. We use *rand* function of MATLAB to generate 50K such filters. Following similar steps, a test set consisting of 5000 images is also created.

5.2 Comparison Methods

We compare our deblurring results with three classes of approaches, (a) State-of-art conventional deblurring approaches which use prior based optimization, (b) Supervised deep learning based end-to-end deblurring approaches, and (c) latest unsupervised image-to-image translation approaches.

Conventional Single image deblurring: We compare with the state-of-the-art conventional deblurring works of Pan et al. [31] and Xu et al. [39] that are proposed for natural images. In addition to this, for face deblurring we used the deblurring work in [29] that is designed specifically for faces. Similarly for text, we compared with the method in [30] that uses prior on text for deblurring. Quantitative results are provided by running their codes on our test dataset.

Deep supervised deblurring: In deep learning, for quantitative analysis on all classes, we compared with end-to-end deblurring work of [27] and additionaly for text and checkerboard we also compared with [12]. The work in [27] is a general dynamic scene deblurring framework and [12] is proposed for text deblurring alone. Note that all these methods use paired data for training and hence are supervised. Besides these for visual comparisons on face deblurring, we also compared with [5] on their images since the trained model was not available.

Unsupervised image-to-image translation: We train the cycleGAN [42] network, proposed for unpaired domain translations, for deblurring task. The network is trained from scratch for each class separately and quantitative and visual results are reported for each class in the following sections.

5.3 Quantitative Analysis

For quantitative analysis, we created the test sets for which the ground truth was available to report the metrics mentioned below. For text dataset, we used the test set provided in [12] itself. And for checkerboard, we used synthetic motion parametrized with $\{l, \theta\}$. For faces, we created test sets using the kernels generated from [4].

Quantitative Metrics: We have used PSNR (in dB), SSIM and Kernel Similarity Measure(KSM) values for comparing the performance of different state of art deblurring algorithms on all the classes. For texts, apart from these metrics, we also use Character Error Rate (CER) to evaluate the performance of various deblurring algorithms.

CER [12] is defined as $\frac{i+s+d}{n}$, where, n is total number of characters in the image, i is the minimal number of character insertions, s is the number of substitutions and d is the number of deletions required to transform the reference text into its correct OCR output. We used ABBYY FineReader 11 to recognize the text and its output formed the basis for evaluating the mean CER. Smaller the CER value, better the performance of the method.

Kernel Similarity Measure: In general practice, the deblurring efficiency is evaluated through PSNR, SSIM metric or with visual comparisons. These commonly used measures (MSE) are biased towards smooth outputs due to 2-norm form. Hence, Hu et al. [13] proposed KSM to evaluate deblurring in terms of the camera motion estimation efficiency. KSM effectively compare estimated kernels ($\hat{K}$) evaluated from the deblurred output with the ground truth (K). It is computed as $S(K, \hat{K}) = \max_{\gamma} \rho(K, \hat{K}, \gamma)$ where $\rho(.)$ is the normalized cross-correlation function given by ($\rho(K, \hat{K}, \gamma) = \frac{\sum_{\tau}(K(\tau).K(\hat{\tau+\gamma}))}{||K||.|\hat{K}||}$) and γ is the possible shift between the two kernels. The larger the value, the better the kernel estimate and indirectly the better the deblurring performance.

Results and Comparisons: For fair comparison with other methods, we used the codes provided by the respective authors on their website. Table 2 summarizes the quantitative performance of various competitive methods along with our network results for all the three classes. A set of 30 test images from each class is used to evaluate the performance reported in the table. It is very clear from the results that our unsupervised network performs on par with competitive conventional methods as well as supervised deep networks. Conventional methods are highly influenced by parameter selection. We used the default settings for arriving at the results for conventional methods. The results could perhaps be improved further by fine-tuning the parameters for each image but this is a time-consuming task. Though deep networks perform well for class-specific data, their training is limited by the lack of availability of large collections of paired data. It can be seen from Table 2 that our network (without data pairing) is able to perform equally well when compared to the class-specific supervised deep method [12] for text deblurring. We even outperform the dynamic deblurring network of [27] in most cases. The cycleGAN [42] (though unsupervised) struggles to learn the blur and clean data domains. It can be noted that, for checkerboard, cycleGAN performed better than ours in terms of PSNR and SSIM. This is because checkerboard had simple linear camera motion. Because blur varied for text and faces (general camera motion) the performance of cycleGAN also deteriorated (refer to the reported values).

Real Handshake Motion: In addition, to test the capabilities of our trained network on real camera motion, we also created test sets for face and text classes using the real camera motion dataset from [17]. Camera motion provided in [17]

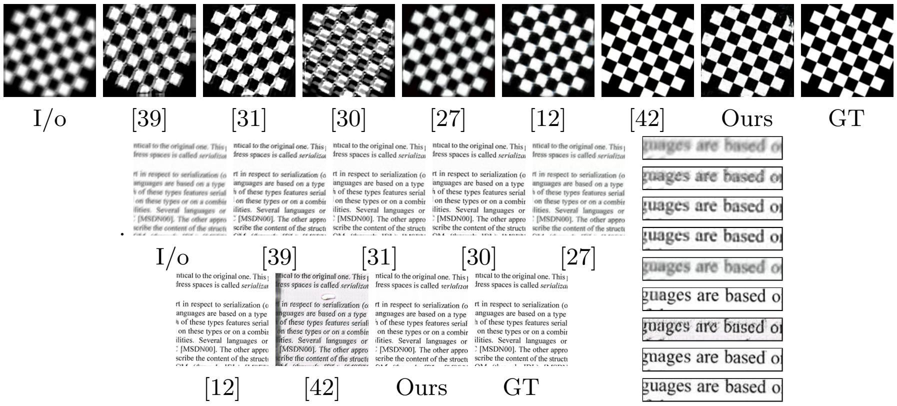


Fig. 4. Visual comparison on checkerboard deblurring. Input blurred image, deblurred results from conventional methods [39], [31] and [30], results from supervised network in [27], [12] and unsupervised network [42], our result and the GT clean image are provided in that order.

contains 40 trajectories of real camera shake by humans who were asked to take photographs with relatively long exposure times. These camera motions are not confined to translations, but consist of non-uniform blurs, originating from real camera trajectories. The efficiency of our proposed network in deblurring images affected by these real motions is reported in Table 3. Since long exposure leads to heavy motion blur which is not within the scope of this work, we use short segments of the recorded trajectory to introduce small blurs. We generated 40 images for both text and faces using 40 trajectories and used our trained network to deblur them. Table 3 shows the PSNR, SSIM between the clean and deblurred images and KSM between the estimated and original motion. The handshake motion in [17] produces space-varying blur in the image and hence a single kernel cannot be estimated for the entire image. We used patches (32×32) from the image and assumed space-invariant blur over the patch to extract the kernel and computed the KSM. This was repeated on multiple patches and an average KSM is reported for the entire image. The KSM, PSNR, and SSIM are all high for both the classes signifying the effectiveness of our network to deal with real camera motions.

5.4 Visual Comparisons

The visual results of our network and competitive methods are provided in Figs. 4 and 5. Fig. 4 contains the visual results for text and checkerboard data. Comparisons are provided with [39, 31] and [30]. The poor performance of these methods can be attributed to the parameter setting (we took the best amongst a set of parameters that gave highest PSNR). Most of these results have ringing artifacts. Now, to analyse the performance of our network over supervised

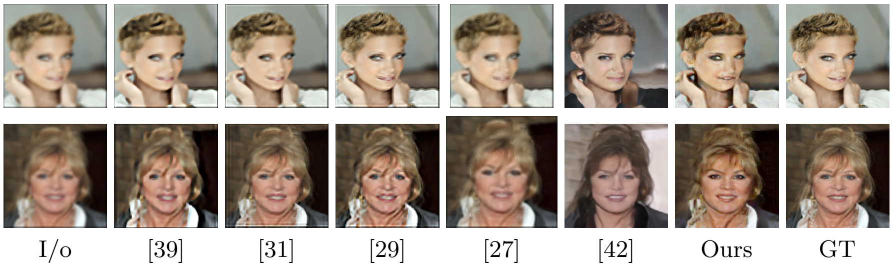


Fig. 5. Visual comparisons on face deblurring.

networks, we compared with the dynamic deblurring network of [27] and class-specific deblurring work of [12]. From the visual results it can be clearly observed that even though the method in [27] gave good PSNR in Table 2 it is visually not sharp and some residual blur remains in the output. The supervised text deblurring network [12] result for checkerboard was sharp but the squares were not properly reconstructed. For completeness, we also trained the unsupervised cycleGAN [42] network separately for each of these classes and the results so obtained are also provided in the figure. The inefficiency of cycleGAN to capture the clean and blur domains simultaneously is reflected in the text results. On the contrary, our unsupervised network produces sharp and legible (see the patches of texts) results in both these classes. Our network outperforms existing conventional methods and at the same time works on par with the text-specific deblurring method of [12]. Visual results on face deblurring are provided in Fig. 5. Here too we compared with conventional methods [39, 31] as before and the exemplar-based face-specific deblurring method of [29]. Though these results are visually similar to the GT, the effect of ringing is high with default parameter settings. The results from deep learning work of [27] is devoid of any ringing artifacts but are highly oversmoothened. Similarly, CycleGAN [42] fails to learn the domain properly and the results are quite different from the GT. On the otherhand, our results are sharp and visually appealing. While competitive methods failed to reconstruct the eyes of the lady in Fig. 5 (second row), our method reconstructs the eyes and produces sharp outputs comparable to GT.

We also tested our network against the latest deep face deblurring work of [5]. Since the trained model for their network was not available, we ran our network on the images provided in their paper. These are real world blurred images from dataset of Lai et al. [19] and from arbitrary videos. The results obtained are shown in Fig. 6. It can be cleraly seen that our method though unupervised can perform at par with the supervised method of [5] and even outperforms it in some examples. The results are sharper with our network; it can be clearly noticed that the eyes are eyebrows are reconstructed well with our network (first and second rows last columns) when compared to [5].

Human perception ranking: We conducted a survey with 50 users to analyze the visual quality of our deblurring. This was done for face and text datasets separately. The users were provided with 30 sets of images from each class grouped

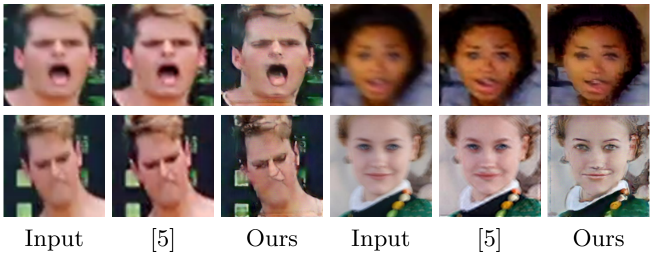


Fig. 6. Visual comparison with the latest face deblurring work of [5].

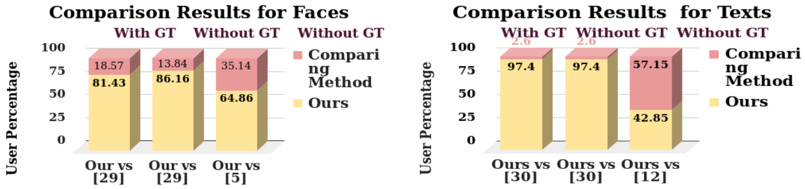


Fig. 7. Summarization of survey: Human rating of our network results against [29] and [5] for faces and [30] and [12] for texts.

into two sections depending on the presence or absence of reference image. In the first group consisting of 10 sets of images, the users were provided with blurred image, ground truth reference, our deblurred result and output from [29]/ [5] or [30]/ [12], based on their visual perception. And in the second group with 20 sets of images the references were excluded. From the face survey result provided in Fig. 7, it can be observed that 81% of the time the users preferred our results over the competitive method [29] when GT was provided and 86% of the time our result was preferred when GT was not provided. For texts, the users preferred our output 97% of the time over the conventional method [30] with or without GT. Also, it can be observed that our method matches well with [12]. 43% of the users opted our method while 57% voted for [12]. More results (on testset and real dataset from [32]), discussions on loss functions, details of survey and limitations of the network are provided in the supplementary material.

6 Conclusions

We proposed a deep unsupervised network for deblurring class-specific data. The proposed network does not require any supervision in the form of corresponding data pairs. We introduced a reblurring cost and scale-space gradient cost that were used to self-supervise the network to achieve stable results. The performance of our network was found to be at par with existing supervised deep networks on both real and synthetic datasets. Our method paves the way for unsupervised image restoration, a domain where availability of paired dataset is scarce.

References

1. Anwar, S., Phuoc Huynh, C., Porikli, F.: Class-specific image deblurring. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 495–503 (2015)
2. Anwar, S., Porikli, F., Huynh, C.P.: Category-specific object image denoising. IEEE Transactions on Image Processing **26**(11), 5506–5518 (2017)
3. Aytar, Y., Castrejon, L., Vondrick, C., Pirsiavash, H., Torralba, A.: Cross-modal scene networks. IEEE transactions on pattern analysis and machine intelligence (2017)
4. Chakrabarti, A.: A neural approach to blind motion deblurring. In: European Conference on Computer Vision. pp. 221–235. Springer (2016)
5. Chrysos, G., Zafeiriou, S.: Deep face deblurring. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017)
6. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2414–2423 (2016)
7. Gatys, L.A., Ecker, A.S., Bethge, M., Hertzmann, A., Shechtman, E.: Controlling perceptual factors in neural style transfer. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
8. Gomez, A.N., Huang, S., Zhang, I., Li, B.M., Osama, M., Kaiser, L.: Unsupervised cipher cracking using discrete gans. arXiv preprint arXiv:1801.04883 (2018)
9. Goodfellow, I.: Nips 2016 tutorial: Generative adversarial networks (2016)
10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
11. Gupta, A., Joshi, N., Zitnick, C.L., Cohen, M., Curless, B.: Single image deblurring using motion density functions. In: European Conference on Computer Vision. pp. 171–184. Springer (2010)
12. Hradiš, M., Kotera, J., Zemcík, P., Šroubek, F.: Convolutional neural networks for direct text deblurring. In: Proceedings of BMVC. vol. 10 (2015)
13. Hu, Z., Yang, M.H.: Good regions to deblur. In: European Conference on Computer Vision. pp. 59–72. Springer (2012)
14. Ignatov, A., Kobyshev, N., Vanhoey, K., Timofte, R., Van Gool, L.: Dslr-quality photos on mobile devices with deep convolutional networks. In: the IEEE Int. Conf. on Computer Vision (ICCV) (2017)
15. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. arxiv (2016)
16. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
17. Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., Harmeling, S.: Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In: European Conference on Computer Vision. pp. 27–40. Springer (2012)
18. Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. arXiv preprint arXiv:1711.07064 (2017)
19. Lai, W.S., Huang, J.B., Hu, Z., Ahuja, N., Yang, M.H.: A comparative study for single image blind deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1701–1709 (2016)

20. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. arXiv preprint (2016)
21. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems. pp. 700–708 (2017)
22. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in neural information processing systems. pp. 469–477 (2016)
23. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (2015)
24. Ma, Z., Liao, R., Tao, X., Xu, L., Jia, J., Wu, E.: Handling motion blur in multi-frame super-resolution. In: Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR). pp. 5224–5232 (2015)
25. Michaeli, T., Irani, M.: Blind deblurring using internal patch recurrence. In: European Conference on Computer Vision. pp. 783–798. Springer (2014)
26. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
27. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
28. Nimisha, T., Singh, A.K., Rajagopalan, A.: Blur-invariant deep learning for blind-deblurring. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017)
29. Pan, J., Hu, Z., Su, Z., Yang, M.H.: Deblurring face images with exemplars. In: European Conference on Computer Vision. pp. 47–62. Springer (2014)
30. Pan, J., Hu, Z., Su, Z., Yang, M.H.: Deblurring text images via l0-regularized intensity and gradient prior. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2901–2908 (2014)
31. Pan, J., Sun, D., Pfister, H., Yang, M.H.: Blind image deblurring using dark channel prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1628–1636 (2016)
32. Punnappurath, A., Rajagopalan, A.N., Taheri, S., Chellappa, R., Seetharaman, G.: Face recognition across non-uniform motion blur, illumination, and pose. IEEE Transactions on image processing **24**(7), 2067–2082 (2015)
33. Rengarajan, V., Balaji, Y., Rajagopalan, A.: Unrolling the shutter: Cnn to correct motion distortions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2291–2299 (2017)
34. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems. pp. 2234–2242 (2016)
35. Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., Wang, O.: Deep video deblurring for hand-held cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1279–1288 (2017)
36. Teodoro, A.M., Bioucas-Dias, J.M., Figueiredo, M.A.: Image restoration with locally selected class-adapted models. In: IEEE International Workshop on Machine Learning for Signal Processing (MLSP). pp. 1–6 (2016)
37. Ulyanov, D., Vedaldi, A., Lempitsky, V.S.: Deep image prior. CoRR **abs/1711.10925** (2017), http://arxiv.org/abs/1711.10925
38. Xie, J., Xu, L., Chen, E.: Image denoising and inpainting with deep neural networks. In: Advances in neural information processing systems. pp. 341–349 (2012)

39. Xu, L., Zheng, S., Jia, J.: Unnatural l0 sparse representation for natural image deblurring. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 1107–1114. IEEE (2013)
40. Xu, X., Sun, D., Pan, J., Zhang, Y., Pfister, H., Yang, M.H.: Learning to super-resolve blurry face and text images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 251–260 (2017)
41. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. arXiv preprint (2017)
42. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593 (2017)