Chapter 6
Blind Super-resolution of Faces for Surveillance

T. M. Nimisha and A. N. Rajagopalan

6.1 Introduction

Super-resolution (SR) refers to a class of techniques that derive a high resolution image from its low resolution (LR) counterpart. A vast amount of literature exists on SR spanning both multi and single image approaches. The classical approaches in SR use sub-pixel motion across multiple low resolution (LR) frames. These works [3, 11] typically assume that the blur encountered in the LR images is only due to downsampling and that the camera is static while capturing LR frames. The only motion addressed in these frameworks is the *inter*-frame motion which is used to infer the underlying high resolution (HR) image.

While multi-frame approaches supplement missing information in one frame from another, availability of multiple frames cannot always be assured. Single image SR [12, 17] is a lot more ill-posed and works by hallucinating the missing data or by exploiting patch-recurrences within an image across different scales. Of-late, many deep learning approaches have been proposed [9, 22, 24] that address the single image SR problem. However, all these methods assume that the blur encountered in the LR frame is only due to downsampling action.

Estimating an HR frame directly from a single motion blurred LR frame is highly ill-posed and is of great relevance in surveillance scenarios. Motion blur is an inevitable phenomenon that co-occurs with long exposure photography. Blur is considered as a nuisance in many image processing algorithms and inverting it is a difficult proposition. Many works exist [6, 23, 40, 49] that focus on the issue of removing motion blur due to camera shake from images. All these works aim for

T. M. Nimisha
Indian Institute of Technology Madras, Chennai, India
e-mail: nimiviswants@gmail.com

A. N. Rajagopalan (✉)
Electrical Engineering, Indian Institute of Technology Madras, Chennai, India
e-mail: raju@ee.iitm.ac.in

N. K. Ratha et al. (eds.), *Deep Learning-Based Face Analytics*, Advances in Computer
Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-74697-1_6

deblurring as the main goal and do not really consider resolution enhancement. SR and deblurring are well-studied problems but are treated as independent topics. Only a few works [33, 41, 47, 49] exist in the literature that addresses both SR and motion deblurring.

The challenge in arriving at an SR image escalates when the underlying LR frames have motion blur artifacts. These situations arise when the subject of interest is far away from the camera and the subject/camera is moving. In these situations, the observed images will be degraded both by motion blur and the downsampling action. Since face recognition (FR) systems are of great use nowadays and are employed as biometric in many areas, a motion distorted LR probe image that deviates significantly from that of the gallery image reduces recognition accuracy. This necessitates the need for single image blind SR. The class of algorithms that estimates an HR image from LR irrespective of artifacts due to motion blur are referred to as blind SR algorithms. It is interesting to note that motion blur occurs due to averaging of several warped instances of the clean frame during exposure. Thus, a single blurred LR frame by default aggregates information from multiple clean frames. Hence, scope exists to harness this aggregated information for deblurring as well as super-resolving.

Performing blind SR sequentially can lead to poor results. The error from the first stage (either SR or deblurring) can propagate to the second and worsen the final output. We propose here a blind SR framework that jointly deblurs and upsamples the probe images to help in achieving better recognition rates for FR systems. Prior works that have addressed the blind SR problem [26, 33, 41, 49], for instance, assumed a multi-frame approach. In contrast, ours is a single image blind SR specifically aimed at improving the accuracy of face recognition systems in surveillance applications.

In this work, we explore invariant feature learning for the purpose of single image SR from motion blurred frames. We employ a deep learning framework for achieving the task at hand. With the underlying idea that natural images follow a sparse distribution and that a shallow dictionary can capture invariance in a sparse domain, we attempt to generalize this invariance to deep non-linear networks. Our network consists of an Encoder-Decoder pair that learns the clean high resolution data domain. This is followed by a Generative Adversarial Network (GAN) that is trained to produce *blur and resolution invariant* features from LR blurred frames. The learned representations are processed by the Decoder to get the final result. We deploy this framework for face surveillance applications where the collected probe images are highly distorted.

6.2 Related Works

Deblurring and SR, though two extensively studied topics, have mostly been dealt with independently. SR frameworks assume static camera leading to LR images affected by downsampling alone. These methods neglect the effect of motion artifacts. Similarly, deblurring approaches assume the availability of high resolution frames and do not work well at a lower resolution. Hence, the performance of these methods drops considerably when the assumptions of blur/resolution do not hold. We discuss here in brief conventional and deep learning based works on deblurring and SR. These can be mainly classified as single image and multi-image approaches.

Super-resolution: Existing works in SR can be broadly classified into two categories (1) Multi-image approaches: Methods [3, 11] that utilize inter-frame sub-pixel motion in the LR frames to restore the HR image and (2) Single image based: These techniques either resort to exemplars or patch-recurrence (also termed 'image hallucination') [12, 17] or patch-based learning [48, 50] to create the HR image. *Single* image SR techniques (which is the focus of our work) employ a database of LR and HR image pairs to learn the correspondences between LR and HR image patches [48, 50]. The patch correspondences thus learned are used during testing to map an LR image to its most likely HR version. However, these techniques are known to hallucinate HR details that may not even be present in the true HR image. Based on the observation that patches in a natural image tend to recur within the same image, both at the same as well as at different scales, the works in [12, 17] sought to combine the strengths of both traditional multi-image SR as well as example-based SR. Recently, deep learning and generative networks have also made forays into computer vision and image processing, and their influence and impact are growing rapidly by the day. Single image SR with deep networks [9, 22, 24] have shown remarkable results that outperform traditional methods. Dong et al. [9] introduce a skip connection-based network that learns residual features for SR. The work in [22] uses a GAN architecture to produce photo-realistic SR outputs from a single LR frame. It is important to note that state-of-the-art SR techniques achieve remarkable results of resolution enhancement only when there is no motion blur in the LR input.

Deblurring: Many methods exist [7, 19, 49] that rely on information from multiple frames captured using video or burst mode and work by harnessing the information from these frames to solve for the underlying original (latent) image. Single image blind deblurring is considerably more challenging as the blur kernel, as well as the latent image, must be estimated from just one observation. Works in [6, 23, 40] perform an iterative approach to solve for the latent image and blur kernel. Most of these methods employ priors on the underlying clean image and motion to stabilize the optimization process. The most widely used priors are total variational regularizer [5, 35], sparsity prior on image gradients, l_1/l_2 image regularization [21], the unnatural l_0 prior [46], and the very recent dark channel prior [32] for images. Even though such prior-based optimization schemes have shown promise, the extent to which a prior is able to perform under general conditions is questionable [21]. Some priors (such as the sparsity prior on image gradient) even tend to favor blurry results [23]. In a majority of situations, the final result requires judicious selection of the prior, its weight, as well as tuning of other parameters. With the advancement in computation and availability of large datasets, deep learning-based deblurring too has come of age. Xu et al. [45] proposed a deep deconvolutional network for non-blind single image deblurring (i.e, the kernel is fixed and known apriori). Schuler et al. [39] came up with a neural architecture that mimics traditional iterative deblurring approaches. Chakrabarti [4] trained a patch-based neural network to estimate the kernel at each patch and employed a traditional non-blind deblurring method in the final step to arrive at the deblurred result. The above-mentioned methods attempt to estimate the blur kernel using a deep network, but finally perform non-blind deblurring outside of the network to get the deblurred result. Any error in the kernel estimate (due to poor edge content, saturation or noise in the image) will impact deblurring quality. Moreover, the final non-blind deblurring step typically assumes a prior (such as sparsity on the gradient of latent image), which again necessitates a careful selection of prior weights; else the deblurred result will be imperfect. Hence, kernel-free approaches are very much desirable. Recent works [30, 31] skips the need for kernel estimation and directly solve for the deblurred frame. But these works are restricted to deblurring and cannot perform resolution improvement.

Blind SR from motion blurred LR images: In situations where the LR frames are affected by motion blur, super-resolution makes little sense without compensating for the effect of the unknown motion blur. Sroubek et al. [41] address the blind SR problem by building a regularized energy function and minimizing it alternately with respect to the original HR image and the camera motion. The method of Ma et al. [26] is based on the premise that the same region is not equally blurred across frames. They propose a temporal region selection scheme to select the least blurred pixels from each frame. The works in [33, 49] perform the joint tasks of alignment, deblurring, and resolution enhancement. It should be noted that the blind SR techniques mentioned above are all multi-frame approaches. Single image blind SR is a much more involved problem and there are at present no traditional approaches to solve it. Very recently, Xu et al. [47] proposed a deep learning algorithm to solve the blind SR problem. They used discriminative image prior based on GAN that semantically favors clear high-resolution images over blurry low-resolution ones and directly regresses for the HR image. In contrast, ours is a sparse coding-based approach and we solve for the HR image by using an invariant feature representation.

6.3 Learning Invariant Features for Faces

Sensory data, including natural images, are sparse in nature and can be described as a superposition of small number of atoms such as edges and surfaces [27]. Dictionary learning methods are built on this very basis. Various image restoration tasks have been attempted with dictionaries (including deblurring and SR). With an added condition that these representations should be invariant to the blur or resolution in the image, dictionary methods have performed these tasks individually by learning coupled dictionaries [43, 48]. However, dictionaries capture only linearities in the data. Blurring process involves non-linearities (high frequencies are suppressed more), hence dictionary methods do not generalize across blurs.

In this chapter, we extend the notion of invariant representations to deep networks that can capture non-linearities in the data. Generalization of dictionary methods using deep networks to capture non-linearities is not new. The work in [44] com-

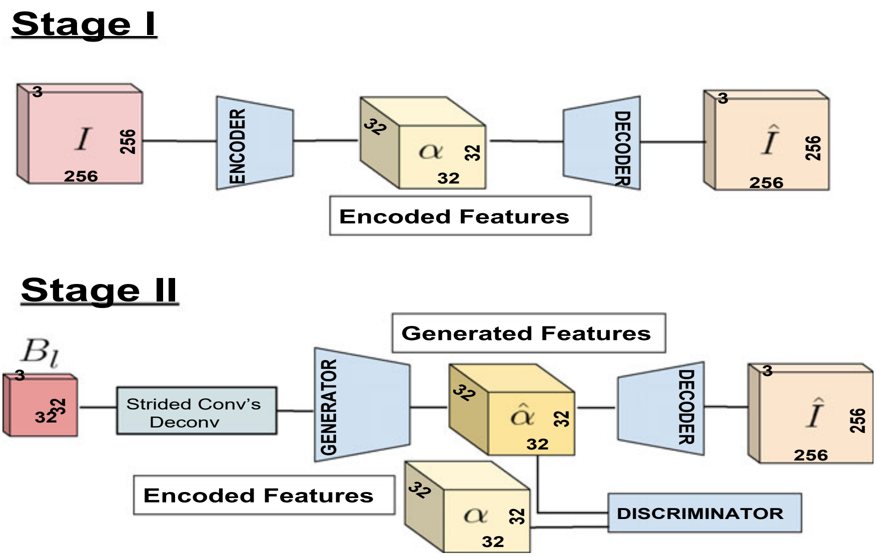

Fig. 6.1 Illustration of proposed architecture

bines sparse coding and denoising encoders for the task of denoising and inpainting. Deep neural networks, in general, have yielded good improvements over conventional methods for various low-level image restoration problems including SR [10], and inpainting and denoising [34, 44]. These networks are learned end-to-end by training with lots of example-data from which the network learns the mapping to undo distortions. We investigate the possibility of such a deep network for the task of single image blind SR. The idea of learning invariant representations is borrowed from our earlier work [31] with the main difference being that the problem we are addressing here is that of a single blind SR rather than just deblurring [31].

Similar to [31], we first require a good feature representation that can capture HR image-domain information. Autoencoders (AE) are apt for this task and have shown great success in unsupervised learning by encoding data to a compact form [15]. Once a good representation is learned for clean HR patches, the next step is to produce an invariant representation (as in [43, 48]) from blurred LR data. We propose to use a GAN for this purpose which involves training of a generator and discriminator that compete with each other. The purpose of the generator is to confuse the discriminator by producing clean features from blurred LR data that are similar to the ones produced by the autoencoder so as to achieve invariance. The discriminator, on the other hand, tries to beat the generator by identifying the clean and blurred features.

A schematic of our proposed architecture is shown in Fig. 6.1. The main difference in architecture vis-a-vis [31] is our generator now has to perform joint SR and deblurring. Since the input LR is of a lower dimension than the HR image, we include fractional strided convolutions in the initial stages of the generator. The number of these layers depends on the SR factor.

Akin to dictionary methods, our encoder-decoder architecture learns a representation in non-linear space. In dictionary approaches, an input HR patch I is sparsely represented with the dictionary atoms D_{HR} as $I = D_{HR}\alpha$. Our encoder-decoder module can be equated to this but in non-linear space. The encoder can be thought of as an inverse dictionary D_{HR}^{-1} that projects the incoming HR data into a sparse representation and decoder (D_{HR}) reconstructs the input from the sparse representation. Generator training can be treated as learning the blur LR dictionary that can project the blurred LR data B_l into the same sparse representation of I, i.e, $\alpha = D_{HR}^{-1}I = D_{b_{LR}}^{-1}B_l$. Once training is done, the input LR blurry image (B_l) is passed through the generator to get an invariant feature which when projected to the decoder yields the deblurred HR result as $\hat{I} = D_{HR}\alpha = D_{HR}D_{b_{LR}}^{-1}B_l$.

Thus, by associating the feature representation learned by the autoencoder with GAN training, our model is able to perform single image blind SR in an end-to-end manner for face dataset. Ours is a kernel-free approach and does away with the tedious task of modeling and selection of prior.

The main contributions of our work are as follows:

- We propose a compact end-to-end regression network that directly estimates the clean HR image from a single blurred LR frame without the need for optimal prior selection and weighting, as well as blur kernel estimation.
- The proposed architecture consists of an autoencoder in conjunction with a generative network for producing blur and resolution invariant features to guide the process.
- The network has shown performance gain in FR surveillance systems and produces good quality face reconstruction from its blurred LR counterpart.

6.4 Network Architecture

Our network consists of an AE that learns the clean HR image domain and a GAN that generates invariant features. We train our network in two stages. We first train an AE to learn the clean image manifold. This is followed by the training of a generator that can produce clean features from a blurred LR image which when fed to the decoder gives the deblurred HR output. Note that instead of combining the task of data representation, SR, and deblurring into a single network, we relegate the task of data-learning to the AE and use this information to guide blind SR. Details of the architecture and the training procedure are explained next.

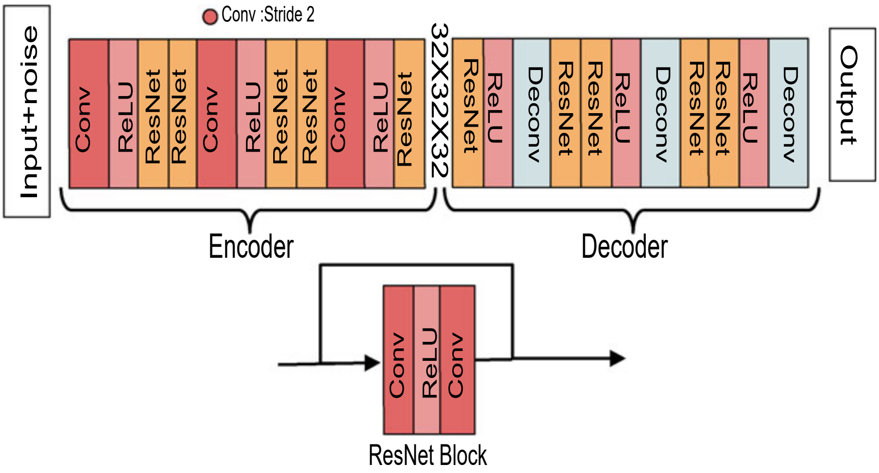

Fig. 6.2 Autoencoder architecture with residual networks

6.4.1 Encoder-Decoder

Autoencoders were proposed for the purpose of unsupervised learning [15] and have since been extended to a variety of applications. AE projects the input data into a low-dimensional space and recovers the input from this representation. When not modeled properly, it is likely that the autoencoder learns to just compress the data without learning any useful representation. Regularization using denoising encoders [42] overcomes this issue by corrupting the data with noise and letting the network undo this effect and get back a clean output. This ensures that the AE learns to correctly represent clean data. Deepak et al. [34] extended this idea from mere data representation to context representation for the task of inpainting. In effect, it learns a meaningful representation that can capture domain information of the data.

We investigated different architectures for AE and observed that including residual blocks (ResNet) [14] helped in achieving faster convergence and in improving the reconstructed output. Residual blocks help by by-passing the higher-level features to the output while avoiding the vanishing gradient problem. The training data was corrupted by noise (30% of the time) to ensure encoder reliability and to avoid learning an identity map. The architecture used in our work along with the ResNet block is shown in Fig. 6.2. A detailed description of the filter and feature map sizes along with the stride values used are as given below.

Encoder: $C^5_{3\to 8} \downarrow 2 \to R^{5(2)}_{8} \to C^5_{8\to 16} \downarrow 2 \to R^{5(2)}_{16} \to C^3_{16\to 32} \downarrow 2 \to R^3_{32}$

Decoder: $R^3_{32} \to C^2_{32\to 16} \uparrow 2 \to R^{5(2)}_{16} \to C^4_{16\to 8} \uparrow 2 \to R^{5(2)}_{8} \to C^4_{8\to 3} \uparrow 2$

where $C^c_{a\to b} \downarrow d$ represents convolution mapping from a feature dimension of a to b with a stride of d and filter size of c, $\downarrow$ represents down-convolution, $\uparrow$ stands for

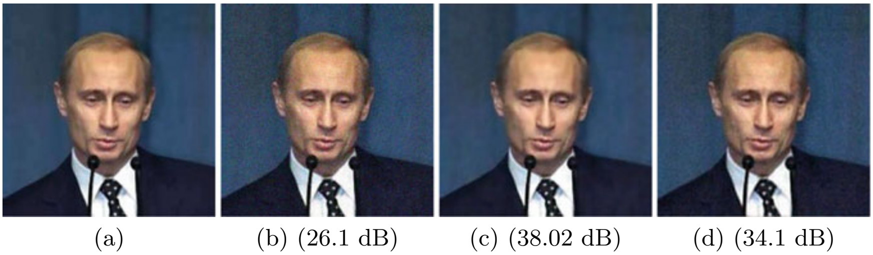

Fig. 6.3 Effect of ResNet on reconstruction. **a** The target image. **b** Noisy input to the encoder-decoder module. **c** Result of encoder-decoder module of Fig. 6.2. **d** Result obtained by removing ResNet for the same number of iterations. PSNR values are given under the respective figures. (*Enlarge for better viewing*)

up-convolution. $R_a^{b(c)}$ represents the residual block which consists of a convolution and a ReLU block with output feature size a, filter size b, while c represents the number of repetitions of residual blocks.

Figure 6.3 shows the advantage of the ResNet block. Figure 6.3a is the target image and Fig. 6.3c, d are the output of autoencoders with and without ResNet block for the same number of iterations for the input noisy image in Fig. 6.3b. Note that the one with ResNet converges faster and preserves the edges due to skip connections that pass on the information to deeper layers.

6.4.2 GAN for Feature Mapping

The second stage of training constitutes learning a generator that can map from the blurred LR image to clean HR features. For this purpose, we used a generative adversarial network (introduced by Goodfellow [13] in 2014). GANs have since been widely used for various image related tasks. It consists of two models: a Generator ($\mathcal{G}$) and a Discriminator ($\mathcal{D}$) which play a two-player mini-max game. $\mathcal{D}$ tries to discriminate between the samples generated by $\mathcal{G}$ and training data samples, while $\mathcal{G}$ attempts to fool the discriminator by generating samples close to the actual data distribution. The mini-max cost function [13] for training GANs is given by

$$\min_{\mathcal{G}} \max_{\mathcal{D}} C(\mathcal{G}, \mathcal{D}) = E_{x \sim P_{\text{data}}(x)}[\log \mathcal{D}(x)] + E_{z \sim P_z(z)}[\log(1 - \mathcal{D}(\mathcal{G}(z))]$$

where $\mathcal{D}(x)$ is the probability assigned by the discriminator to the input x for discriminating x as a real sample. P_{data} and P_z are the respective probability distributions of data x and the input random vector z. The main goal of [13] is to generate a class of natural images from z.

Theoretically, GANs are well-defined, but many a time it is difficult to train them. Often there are instability issues that results in artifacts in the generated image. Works exsist that specifically address this issue [37, 38] and try to stabilize the training by introducing new distance metrics [2]. One such work uses conditional GAN (Mirza et al. [29]) which enables GANs to accommodate extra information in the form of conditional input. Training conditional GANs is a lot more stable than unconditional GANs due to the additional guiding input. The inclusion of adversarial cost in the loss function has shown great promise [18, 34]. The modified cost function [18] is given by

$$\min_{\mathcal{G}} \max_{\mathcal{D}} C_{cond}(\mathcal{G}, \mathcal{D}) = E_{x,y\sim P_{\text{data}}(x,y)}[\log \mathcal{D}(x, y)] \\ + E_{x\sim P_{\text{data}}(x), z\sim P_z(z)}[\log(1 - \mathcal{D}(x, \mathcal{G}(x, z))] \quad (6.1)$$

where y is the clean target feature, x is the conditional image (the blurred input), and z is the input random vector. In conditional GANs, the generator tries to model the distribution of data over the joint probability distribution of x and z. When trained without z for our task, the network learns a mapping for x to a deterministic output y which is the corresponding clean feature.

Following [18] that uses an end-to-end network with generative model to perform image-to-image translation, we initially attempted regressing directly to the clear pixels using off-the-shelf generative networks. However, we observed that this lead to erroneous results. One reason for this could be due to the high dimensionality of data. Hence, we used the apriori-learned features (which are of a lower dimension as compared to image space) of the autoencoder for training GAN. Training a perfect discriminator requires its weights to be updated simultaneously along with the generator such that it is able to discriminate between the generated samples and data samples. This task becomes easy and viable for the discriminator in the feature space for two reasons:

(i) In this space, the distance between blurred LR features and its equivalent clean HR features is higher as compared to the image space. This helps in faster training in the initial stage.

(ii) The dimensionality of the feature space is much lower as compared to that of image space. GANs are known to be quite effective in matching distributions in lower-dimensional spaces [8].

We train the GAN using the normal procedure but instead of asking the discriminator to discern between generated images and clean images, we ask it to discriminate between their corresponding features. The generator (4 ×) and the discriminator architectures are as given below.

Generator: $C^5_{3\to8} \uparrow 2 \to C^5_{8\to8} \to C^5_{8\to16} \uparrow 2 \to C^5_{16\to16} \downarrow 2 \to R^{5(2)}_{16} \to C^5_{16\to32} \downarrow 2 \to R^{5(2)}_{32} \to \hat{C}^3_{32\to32} \downarrow 2 \to R^{5(2)}_{32} \to C^3_{32\to128} \downarrow 2 \to R^{3(2)}_{128} \to \hat{C}^3_{128\to32} \uparrow 2$

Discriminator: $C^5_{32\to32} \to C^5_{32\to32} \downarrow 2 \to C^5_{32\to16} \to C^5_{16\to16} \downarrow 2 \to C^5_{16\to8} \to C^3_{8\to8} \downarrow 2 \to C^3_{8\to1}$

Each convolution is followed by a Leaky ReLU and batch-normalization in the discriminator, and ReLU in the generator. The input stage of the generator is a stack of learnable upsampling filters (Deconv layers) and the number of such layers depends on the upsampling factor. Above, we have shown a generator module for 4 × SR factor. $\hat{C}$ indicates a skip connection from that convolution layer till the next $\hat{C}$. Using skip connections help in preserving the finite feature from the lower layers while going deeper helps in reducing the blur. We also tried other models where the generator architecture was similar to that of encoder. Such an architecture helps to preserve details in the final output but residual blur still remains in the output. We observed that going deeper helps in reducing blur at the expense of missing finite details. Hence, we used a generator which goes deeper but at the same time preserves features using skip connections.

Once the second stage is trained, we have a generator module to which we pass the blurred LR input during the test phase. The generator produces features which correspond to clean image features which when passed through the decoder deliver the final deblurred HR result.

6.4.3 Loss Function

Our network is trained in two stages. In the initial phase, the encoder is trained to learn the HR clean feature representation. For this training, we used the widely preferred reconstruction cost. The reconstruction (MSE loss) cost is defined as the l_2 distance between the expected and observed image and is given as

$$\mathcal{L}_{\text{mse}} = \|\mathcal{D}e(\mathcal{E}(I + N)) - I\|_2^2 \tag{6.2}$$

where $\mathcal{D}e$ is the decoder, $\mathcal{E}$ the encoder, N is noise and I is the target (clean) image. The MSE error captures overall image content but tends to prefer a blurry solution. Hence, training only with MSE loss results in loss of edge details. To overcome this, we used gradient loss ($\mathcal{L}_{\text{grad}}$) as it favors edges as discussed in [28] for video-prediction.

$$\mathcal{L}_{\text{grad}} = \|\nabla\mathcal{D}e(\mathcal{E}(I + N)) - \nabla I\|_2^2 \tag{6.3}$$

where ∇ is the gradient operator. Adding the gradient loss helps in preserving edges and recovering sharp images as compared to $\mathcal{L}_{\text{mse}}$ alone.

The second phase of training learns the invariant representation using GANs. For training GAN we tried different combinations of cost functions and found that a combined cost function given by $\lambda_{\text{adv}}\mathcal{L}_{\text{adv}} + \lambda_1\mathcal{L}_{\text{abs}} + \lambda_2\mathcal{L}_{\text{mse}}$ in the image and feature space worked for us. Even though l_2 loss is simple and easy to back-propagate, it under-performs on sparse data. Hence, we used l_1 loss for feature back-propagation, i.e.

$$\mathcal{L}_{\text{abs}} = \|\mathcal{G}(B) - \mathcal{E}(I)\|_1 \tag{6.4}$$

where B is the blurred LR image. The adversarial loss function $\mathcal{L}_{\text{adv}}$ (given in Eq. (6.1)) requires that the samples output by the generator should be indistinguish-

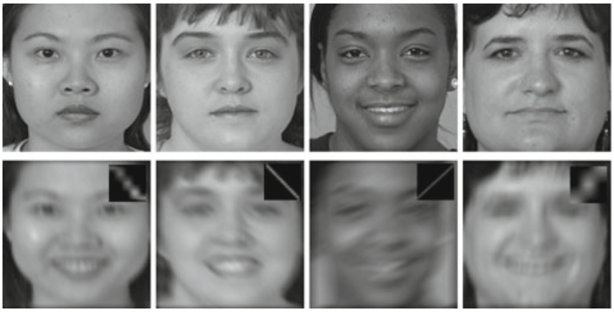

Fig. 6.4 Some example images from gallery (first row) and probe (second row). The kernels used to synthesize the probe images are shown in the inset

able to the discriminator. This is a strong condition and forces the generator to produce samples that are close to the underlying data distribution. As a result, the generator outputs features that are close to the clean HR feature samples. Another advantage of this loss is that it helps in faster training (especially during the initial stages) as it provides strong gradients. Apart from adversarial and l_1 cost on the feature space, we also used MSE cost on the recovered clean image after passing the generated features through the decoder. This helps in fine-tuning the generator to match with the decoder.

6.4.4 Training

We trained the autoencoder using images from the CelebA dataset [25] which consists of around 202,599 face images by resizing them to 256 × 256. We randomly picked 200K data as training set and rest as test and validation set. The inputs were randomly corrupted with Gaussian noise (standard deviation = 0.2) 30% of the time to ensure learning of useful data representation. We used Adam [20] with an initial learning rate of 0.0002 and momentum 0.9 with batch-size of 16. The training took around 3×10^5 iterations to converge. The gradient cost was scaled by $\lambda = 0.1$ to ensure that the final results are not over-sharpened.

The second stage of training involved learning a blur and resolution invariant representation from blurred LR data. We created blurred face data by synthetically blurring the CelebA dataset with space-invariant parametric blur kernels. We used $\{l, \theta\}$ (l stands for length and θ is the angle) parametrization of the blur and produced blur in the range $l \in \{0, 40\}$ pixels and $\theta \in \{0, 180\}$ degrees. The input clean images were blurred by the parametrized kernels and downsampled by factors of 2, 4, and 8 to generate the training sets for different SR factors. Each set consisted of 4 lakh blurred LR training data. The first stage of the generator was a set of up-convolution learnable filters that scale up the input data to 256. To improve GAN stability, we

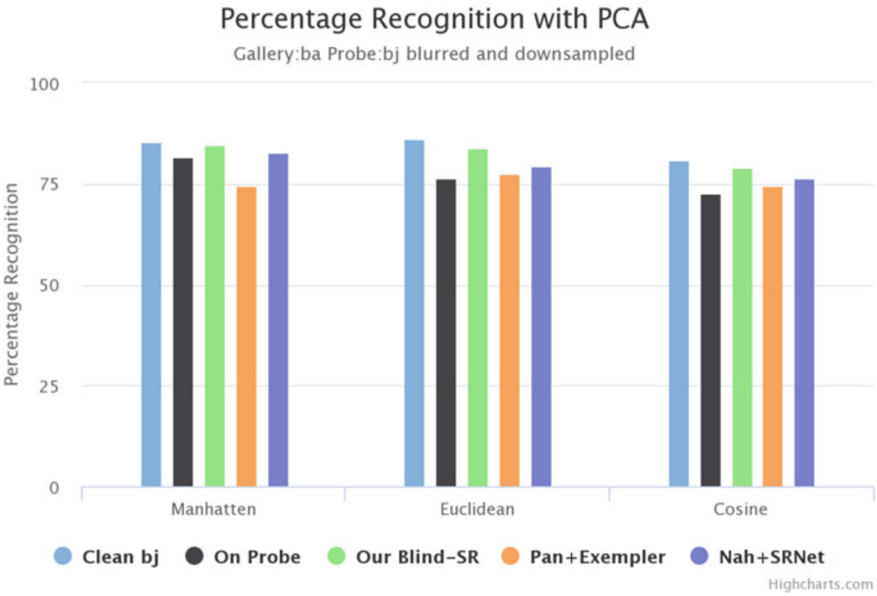

Fig. 6.5 Percentage recognition with a simple PCA FR system. The improvement in accuracy with our blind SR network over other comparative methods can be clearly observed from the figure. Our method performs well with respect to all the matching distance metrics

also used smooth labeling of blur and clean features as discussed in [1]. For around 10^5 iterations, the training was done with feature costs alone with $\lambda_{adv} = 0.001$ and $\lambda_1 = 1$. Fine-tuning of the generator was subsequently done by adding the MSE cost and weighing down the adversarial cost ($\lambda_2 = 1$, $\lambda_1 = 1$ and $\lambda_{adv} = 0.0001$).

6.5 Experiments

We demonstrate the effectiveness of our proposed blind SR network on synthetic as well as real images. We have subdivided the experiments into two sections. In the first section, we quantify performance by analyzing the recognition accuracy of a baseline FR system on the input blurred LR images prior to and after passing through our network. We observed an improvement in accuracy after using our network. The experimental setup for this is as follows. We took the ba and bj folders from the FERET dataset both of which contain 200 subjects (256 × 256 dimension) with one image per subject. We used ba as our gallery and used bj to produce the probe. The images from bj were subjected to parametric blur and downsampled to get 64×64 probe data. A few examples from the gallery and the probe along with the kernels used to create them are shown in Fig. 6.4. Following this, a basic FR system using PCA was used as the baseline to calculate the percentage recognition rate.

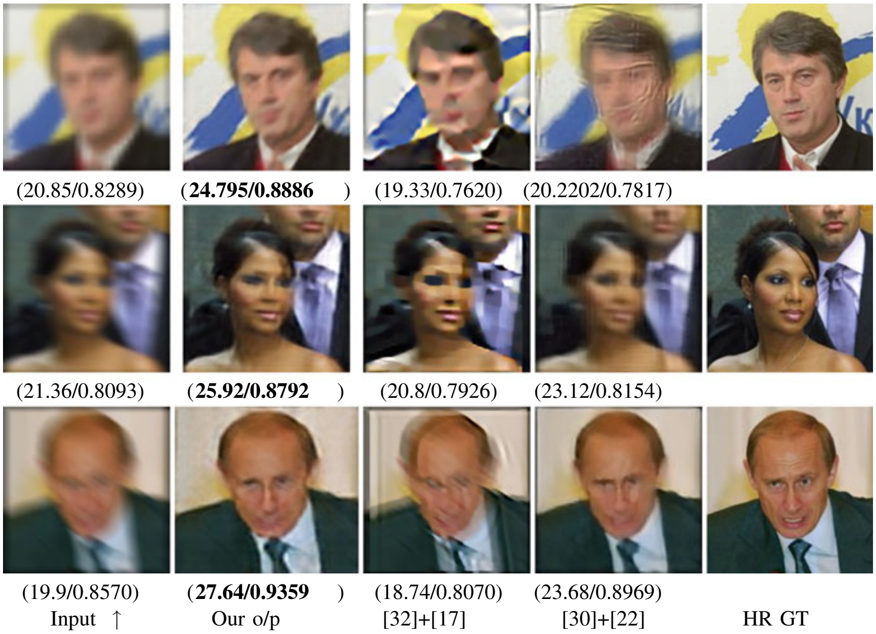

Fig. 6.6 Results on LFW dataset [16]: The input images were upsampled to [256 × 256]. Results obtained by our blind SR network given in column 2. Results obtained by separately performing deblurring and SR by conventional methods [32]+[17], and deep methods [30]+[22], are given in column 3 and 4, respectively. The ground truth HR image is shown in the last column

The system works by first estimating the PCA basis from the clean HR gallery images. It then projects the probe using the estimated basis and recognizes the subject by matching the features to that of the gallery. We used three distance metrics for matching: Euclidean, Manhattan, and Cosine. Initially, we estimated the recognition rate on the clean probe and found that the recognition was on an average 84.16% only. This was because the probe images had small expression changes from the gallery and our FR system is a simple PCA-based model. Next, we checked the accuracy on the blurred LR probe data and noticed that the accuracy went down from 84.16to 76.8% after blurring and downsampling. We passed these LR probes through our trained network to get a 4 × SR and estimated the accuracy of FR on the output, the accuracy improved to 82.5% using our blind SR model.

Since there are no works on single blind SR for this type of a setting, we performed comparison by independently deblurring the LR frames followed by a single image SR framework. This we did using both conventional methods and deep learning methods. For conventional method, we use a single image deblurring framework of Pan et al. [32] to deblur the LR frames. This is followed by exemplar-based SR as proposed in [17]. The accuracy obtained in this case was quite less (75.5%). The main reason for the reduced accuracy could be due to the artifacts induced by deblurring

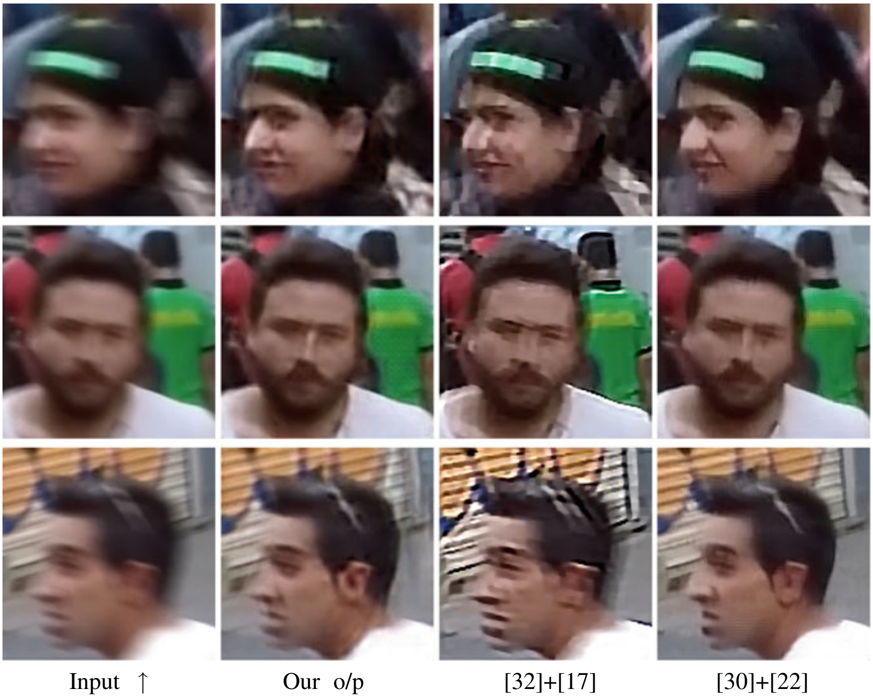

Fig. 6.7 Results on Gopro dataset [30]. Faces were cropped from the blurred images provided in the test set. Even though our network was not trained for such a real dataset, it was able to produce comparable results to the work in [30] that was specifically trained on Gopro

which can be attributed to improper selection of prior. The second comparison was with deep learning networks. For this, we used the network in [30] to deblur the probes and these deblurred results were subjected to the SRResnet (proposed in [22]) for 4× upsampling. The accuracy improved to 79.56%. From this experiment, we can conclude that our blind SR network that performs end-to-end simultaneous deblurring and SR can help in improving the recognition accuracy of FR systems. The obtained accuracy using each of the matching methods along with comparisons are provided in Fig. 6.5.

In the second section of our experiments, we show quantitative results on synthetically blurred LR dataset from the LFW dataset [16]. We provide quantification in terms of PSNR (Peak Signal to Noise Ratio) and SSIM (Structure Similarity Index). We also provide qualitative results on a few examples from the real blurred dataset of [36] and Gopro dataset in [30].

For the quantitative experiment in Fig. 6.6, we synthetically blurred the LFW dataset and downsampled it to different scales. For comparison, we tried the existing conventional and deep learning methods as before. Results in the third column of Fig. 6.6 were obtained by deblurring the LR image with the conventional deblurring

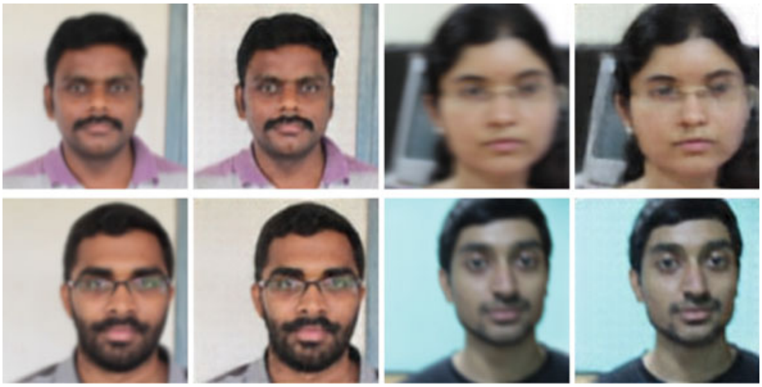

Fig. 6.8 Results on real blurred dataset of [36]. Input blurred face and the corresponding result obtained by our network are shown side-by-side

work of Pan et al. [32] which was followed by exemplar-based SR method of [17] for the specified SR factors. Similarly, the results in column four were obtained by the deep learning-based deblurring work of [30] followed by the single image deep learning-based SR work of [22]. The obtained PSNR and SSIM values for each of these examples are provided under each image. Each row corresponds to a different upsampling factor: first row (8 ×), second row (4 ×) and third row (2 ×). Note that the training for each SR factor was done separately in our network, but the encoder training was done only once. From the results, it is evident that a joint approach for deblurring and SR performs much better than individually performing deblurring and SR.

Next, we tested our network on two real blurred datasets provided in [30, 36]. Gopro dataset introduced in [30] was produced by capturing videos using a high frame rate camera and then averaging the frames to produce realistic blurred dataset. We manually cropped faces from their test sets and fed them to our network. Our network was trained with synthetic parametric blur kernels as discussed in Sect. 6.4.4. Even with this training, we obtained results (second column) comparable to that of [30] (fourth column), which was specifically trained on Gopro. The obtained results for visual comparison are provided in Fig. 6.7. A comparison with the traditional method is also provided in the third column of Fig. 6.7. A qualitative result of our method on the real blurred dataset captured by Punnapurath et al. [36] is also provided in Fig. 6.8. Although the blur encountered in the inputs was not high, one can observe an improvement in quality with our network.

It must be mentioned that the work in [47] also addresses blind SR problem for face images. They achieve this by using a direct regression for the HR image from the blurred LR using a generative loss. Our method differs from them in the network architecture. We learn a feature representation with our network that is invariant to the blur and resolution by making use of the generative framework. To compare with the method in [47], we retrained our encoder and generator module on celebA dataset

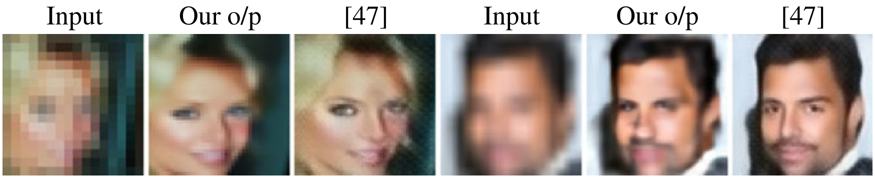

Fig. 6.9 Experimental setup similar to [47]. The input images were of size 64 × 64. These were subjected to blur and downsampling to get LR inputs of size 16 × 16. They trained by cropping faces alone but our training was by resizing. Hence, we had to crop out the face after passing through our network to match their result. The reduction is quality in our result is due to this cropping

for the specified input–output resolution as mentioned in their paper. The inputs (HR) were of 64 × 64 and the LR blurred data were 16 × 16. We modified our architecture to accommodate this input size and learned the invariant features. The input image, the result obtained by our method, and the output of [47] are shown in Fig. 6.9. Our results are comparable to that of [47].

6.6 Conclusions

In this chapter, we proposed an end-to-end deep network for single image blind SR using autoencoder and GAN. Instead of directly regressing for clean pixels, we performed regression over encoder-features to arrive at an invariant representation, which when passed through the decoder produces the desired clean HR output. Our network is kernel-free and does not require any prior modeling. The method shows improvement in FR accuracy even with a baseline FR system. When tested on real datasets, our method showed improved quality when compared to decoupled deblurring and SR.

References

1. Arjovsky M, Bottou L (2017) Towards principled methods for training generative adversarial networks. In: NIPS 2016 workshop on adversarial training, vol 2016
2. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: International conference on machine learning, pp 214–223
3. Capel DP (2004) Image mosaicing and superresolution
4. Chakrabarti A (2016) A neural approach to blind motion deblurring. In: ECCV. Springer, pp 221–235
5. Chan TF, Wong C-K (1998) Total variation blind deconvolution. TIP 7(3):370–375
6. Cho S, Lee S (2009) Fast motion deblurring. In: TOG, vol 28. ACM, p 145
7. Delbracio M, Sapiro G (2015) Burst deblurring: removing camera shake through fourier burst accumulation. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 2385–2393

8. Donahue J, Krähenbühl P, Darrell T (2017) Adversarial feature learning. In: ICLR
9. Dong C, Loy CC, He K, Tang X (2014) Learning a deep convolutional network for image super-resolution. In: European conference on computer vision. Springer, pp 184–199
10. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. TPAMI 38(2):295–307
11. Farsiu S, Robinson MD, Elad M, Milanfar P (2004) Fast and robust multiframe super resolution. IEEE Trans Image Process 13(10):1327–1344
12. Glasner D, Bagon S, Irani M (2009) Super-resolution from a single image. In: 2009 IEEE 12th international conference on computer vision. IEEE, pp 349–356
13. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: NIPS, pp 2672–2680
14. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: CVPR, pp 770–778
15. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
16. Huang GB, Ramesh M, Berg T, Learned-Miller E (2007) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst
17. Huang J-B, Singh A, Ahuja N (2015) Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5197–5206
18. Isola P, Zhu J-Y, Zhou T, Efros AA (2016) Image-to-image translation with conditional adversarial networks. arXiv:1611.07004
19. Ito A, Sankaranarayanan AC, Veeraraghavan A, Baraniuk RG (2014) Blurburst: removing blur due to camera shake using multiple images. ACM Trans Graph Submitt 3(1)
20. Kingma D, Adam JB (2015) A method for stochastic optimisation. In: ICLR. ICLR
21. Krishnan D, Tay T, Fergus R (2011) Blind deconvolution using a normalized sparsity measure. In: CVPR. IEEE, pp 233–240
22. Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: The IEEE conference on computer vision and pattern recognition (CVPR)
23. Levin A, Weiss Y, Durand F, Freeman WT (2011) Understanding blind deconvolution algorithms. TPAMI 33(12):2354–2367
24. Lim B, Son S, Kim H, Nah S, Lee KM (2017) Enhanced deep residual networks for single image super-resolution. In: The IEEE conference on computer vision and pattern recognition (CVPR) workshops
25. Liu Z, Luo P, Wang X, Tang X (2015) Deep learning face attributes in the wild
26. Ma Z, Liao R, Tao X, Xu L, Jia J, Wu E (2015) Handling motion blur in multi-frame super-resolution. In: Proceedings of the CVPR, pp 5224–5232
27. Mairal J, Ponce J, Sapiro G, Zisserman A, Bach FR (2009) Supervised dictionary learning. In: NIPS, pp 1033–1040
28. Mathieu M, Couprie C, LeCun Y (2015) Deep multi-scale video prediction beyond mean square error. arXiv:1511.05440
29. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv:1411.1784
30. Nah S, Kim TH, Lee KM (2017) Deep multi-scale convolutional neural network for dynamic scene deblurring. In: The IEEE conference on computer vision and pattern recognition (CVPR)
31. Nimisha TM, Singh AK, Rajagopalan AN (2017) Blur-invariant deep learning for blind-deblurring. In: The IEEE international conference on computer vision (ICCV)
32. Pan J, Sun D, Pfister H, Yang M-H (2016) Blind image deblurring using dark channel prior. In: CVPR, pp 1628–1636
33. Park H, Lee KM (2017) Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence. In: The IEEE international conference on computer vision (ICCV)

34. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: Feature learning by inpainting. In: CVPR, pp 2536–2544
35. Perrone D, Favaro P (2014) Total variation blind deconvolution: the devil is in the details. In: CVPR, pp 2909–2916
36. Punnappurath A, Rajagopalan AN, Taheri S, Chellappa R, Seetharaman G (2015) Face recognition across non-uniform motion blur, illumination, and pose. IEEE Trans Image Process 24(7):2067–2082
37. Roth K, Lucchi A, Nowozin S, Hofmann T (2017) Stabilizing training of generative adversarial networks through regularization. arXiv:1705.09367
38. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training GANS. In: NIPS, pp 2226–2234
39. Schuler CJ, Hirsch M, Harmeling S, Schölkopf B (2014) Learning to deblur. In: NIPS
40. Shan Q, Jia J, Agarwala A (2008) High-quality motion deblurring from a single image. In: TOG, vol 27. ACM, p 73
41. Sroubek F, Cristobal G, Flusser J (2007) A unified approach to superresolution and multichannel blind deconvolution. IEEE Trans Image Process 16(9):2322–2332
42. Vincent P, Larochelle H, Bengio Y, Manzagol P-A (2008) Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th iICML. ACM, pp 1096–1103
43. Xiang S, Meng G, Wang Y, Pan C, Zhang C (2015) Image deblurring with coupled dictionary learning. IJCV 114(2–3):248–271
44. Xie J, Xu L, Chen E (2012) Image denoising and inpainting with deep neural networks. In: NIPS, pp 341–349
45. Xu L, Ren JS, Liu C, Jia J (2014) Deep convolutional neural network for image deconvolution. In: NIPS, pp 1790–1798
46. Xu L, Zheng S, Jia J (2013) Unnatural l0 sparse representation for natural image deblurring. In: CVPR, pp 1107–1114
47. Xu X, Sun D, Pan J, Zhang Y, Pfister H, Yang M-H (2017) Learning to super-resolve blurry face and text images. In: The IEEE international conference on computer vision (ICCV)
48. Yang J, Wright J, Huang TS, Ma Y (2010) Image super-resolution via sparse representation. IEEE Trans Image Process 19(11):2861–2873
49. Zhang H, Carin L (2014) Multi-shot imaging: joint alignment, deblurring and resolution-enhancement. In: CVPR, pp 2925–2932
50. Zhu Y, Zhang Y, Yuille AL (2014) Single image super-resolution using deformable patches. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2917–2924