Signal Processing 156 (2019) 92–105

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Variational probabilistic generative framework for single image super-resolution

Zhengjue Wang[a,b], Bo Chen[a,b,*], Hao Zhang[a,b], Hongwei Liu[a,b]

[a] National Laboratory of Radar Signal Processing, Xidian University, Xi'an 710071, China
[b] Collaborative Innovation Center of Information Sensing and Understanding at Xidian University, Xi'an 710071, China

ARTICLE INFO

Article history:
Received 3 April 2018
Revised 2 October 2018
Accepted 4 October 2018
Available online 11 October 2018

Keywords:
Probabilistic generative model
Image super-resolution
Conditional prior
Recognition model

ABSTRACT

In this paper, a general variational probabilistic generative framework parameterized by deep networks is proposed for single image super-resolution, which assembles the advantages of coding-based methods and regression-based methods. We use probabilistic generative networks to model the joint full likelihood of a pair of low-resolution (LR) and high-resolution (HR) patches which are generated from a shared latent representation. An inference model is applied to infer the stochastic distribution of the latent representation. By jointly optimizing the generative and inference models, a regression process to the distribution of the HR patch is implied during the learning phase, which provides an efficient forward mapping to accomplish the super-resolution task. We use our framework as a guidance and develop a new model called PGM-CP, with the help of an informative conditional prior and a consistent recognition model. We likewise show how three existing popular example-based SR methods can be "reinvented" under our framework. The effectiveness and efficiency of the proposed method is examined based on three public datasets. Experimental results demonstrate that our model is competitive with state-of-the-art approaches, especially when the image is corrupted by noise.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Image super-resolution (SR) is desirable in overcoming the inherent resolution limitations from low-cost imaging sensors or other factors, and have shown promising results in many applications, such as medical diagnosis, remote sensing, and surveillance [1,2]. Given multiple low-resolution (LR) images from the same scene, the SR task is cast as the inverse problem of recovering the high-resolution (HR) image based on reasonable assumptions about the observation model that maps the HR image to LR ones, which is an ill-posed problem. Various deterministic and statistic regularization approaches, e.g., [3–6], have been proposed to stabilize the inversion of such ill-posed problem. However, the performances of those approaches degrade when the available LR images are insufficient [7,8], an extreme case as single LR image which is a practical scenario in real applications. In this paper, we focus on the task of single image SR, aiming at recovering an HR image from a single LR image.

A straight forward solution to enhance the resolution of the LR image is realized by interpolation, such as bicubic, Lanczos [9], and edge-guided methods [10,11]. In spite of their low computational cost, those methods are prone to producing blurred edges and unpleasant artifacts [12]. Recently, example-based strategy has been mostly adopted, leveraging machine learning techniques to restore the missing details of the LR image based on a set of prior examples. Among those methods, some approaches exploit the assumption that small patches often recur within and across scales of the same image, and conduct self-similarity examples from the testing image itself [13–16]. While, another line of works focus on learning the relationship between the LR and HR patches in pair based on external dataset, which this paper belongs to. Existing external example-based methods roughly concern two categories [17]: coding-based methods and regression-based methods.

The coding-based methods derive from modeling the decoding process from the latent codes to the observed patches, serving the SR task indirectly. Methods in [18] and [17] adopt the philosophy of locally linear embedding (LLE) [19] from manifold learning, and assume that the LR and HR patches form manifolds with similar local geometry, such that a pair of LR and HR patches can be linearly reconstructed by its neighbors with shared weights. However,

[*] Corresponding author.
E-mail addresses: zhengjuewang@163.com (Z. Wang), bchen@mail.xidian.edu.cn (B. Chen).

https://doi.org/10.1016/j.sigpro.2018.10.004
0165-1684/© 2018 Elsevier B.V. All rights reserved.

the manipulation of neighbor search on large training dataset is often computational expensive. Yang et al. [7,8,20] apply linear dictionary learning and constrain that the sparse representation of an LR patch over the LR dictionary is identical to that of its HR counterpart over the HR dictionary, forming a SR path from the LR patch to its sparse code and HR patch successively. Considering those methods have the requirement of specifying the parameters, e.g., the number of dictionary atoms, the variance of the noise, Polatkan et al. [21] develop a sparse Bayesian nonparametric method for SR, where the sparse prior of the latent variable is provided by a beta-Bernoulli process. Thanks to the sparsity reconstruction scheme that models the distribution of the noise, methods in [7,8], and [21] are robust to noise to some degree. However, the performances of those methods are limited by the linear nature, and the testing stage is time-consuming, since they need to infer the latent codes iteratively.

The regression-based methods directly learn mapping functions from the LR patch to its HR counterpart. In [22] and [23], tensor regression and kernel ridge regression are adopted for this purpose, respectively. Considering that deep architectures are more powerful in data expression than shallow models in various applications [24–29], various methods build deep networks to solve the SR regression problem. In [30–33], and [34], a coupled deep autoencoder, a deep convolutional neural network, a network based on LISTA [35], a deeply-recursive convolutional network, and a generative adversarial network, are respectively designed to learn an end-to-end non-linear mapping between the LR and HR patches. During the testing phase, in contrast to the coding-based methods iteratively inferring the latent codes, the regression-based methods are more efficient thanks to the forward mapping. Though the SR performances are appealing, the regression-based methods lack of robustness when noise or missing values are present in the testing images. This is partially due to the fact that the aforementioned deep models use deterministic forward mappings to realize point estimate, which highly relies on the point similarity between the training and testing datasets [36]. Therefore, any anomaly in the testing image, e.g., noise, would be treated as normal pixels to be mapped into the output image. Although [32] and [31] claim that the sparse coding is implied in their networks, they do not apply potential uncertainty of the latent representations nor constrain the reconstruction of a clean input, therefore the robustness of the latent codes to noise is not guaranteed [37].

Thus, in this paper we try to formulate a general framework to fuse the strategies of coding and regression for the SR task, in order to realize good SR performance with low computational cost and robustness to noise. It is known that probabilistic generative models aim to reveal the entire distribution profile of the underlying structure of the data, which possess robustness and flexibility in modeling noise characteristics and priori knowledge [1]. Taking advantages of both probabilistic generative models and deep networks becomes popular and has achieved state-of-the-art performance in other tasks, such as supervised learning [38], and recommender system [37]. The main contributions of the this paper are summarized as follows:

Firstly, a general variational probabilistic generative framework parameterized by deep network is proposed for single image SR. Specifically, the probabilistic generative network models the joint full likelihood of a pair of LR and HR patches that are generated from a shared latent variable, acting as a decoder; an inference network is applied to infer the stochastic distribution of the latent variable, acting as an encoder. Since we jointly optimize the generative and inference models, a regression process to the distribution of the HR patch is implied during the learning phase, which provides an efficient forward mapping to accomplish the super-resolution task.

Besides, since the proposed framework is flexible that can be implemented with different choices of prior, likelihoods, and neural networks, we use our framework as a guidance and develop a new model, probabilistic generative model with conditional prior (PGM-CP), where an informative conditional prior and a consistent recognition model are proposed with low computational cost and robustness to noise.

In addition, we explicitly formulate how the existing popular coding-based methods ScSR [8] and BPFASR [21], and the regression-based method SRCNN [39] can be "reinvented" under the propose framework, in order to make the comparisons more intuitive.

The rest of this paper is organized as follows. In Section 2, we formulate the proposed framework. In Section 3, we implement the framework and show a novel SR model called PGM-CP. Section 4 detailed compares the PGM-CP with three existing models under the proposed framework. Experimental results are presented in Section 5. Section 6 concludes this paper.

2. Variational probabilistic generative framework

Consider a data set $\{\mathbf{X}_l, \mathbf{X}_h\} = \{\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)}\}_{i=1}^{N}$ consisting of N data pairs, where $\boldsymbol{x}_i^{(l)}$ and $\boldsymbol{x}_i^{(h)}$ are two column vectors representing an LR patch and an HR patch at the corresponding locations in an LR and HR image pair, respectively.

Firstly, a probabilistic generative model (PGM) is proposed aiming at modeling the generative processes and distributions of the LR and HR patches, and revealing the inherent relations between patches in pair in order to serve the task of single-image SR. It is assumed that $\boldsymbol{x}_i^{(l)}$ and $\boldsymbol{x}_i^{(h)}$ are generated with a shared latent representation, and that all the data pairs are independently and identically distributed. More detailed formulations are as follows:

- ***Prior:*** the latent variable $\boldsymbol{z}_i$ corresponding to the ith data pair is generated from some prior distribution $p(\boldsymbol{z}_i)$.
- ***Likelihood:*** the ith data pair is generated from the following conditional distributions:

$$p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i) = \mathcal{F}(\ldots, f_m(\boldsymbol{z}_i), \ldots), \quad m = 1, 2, \ldots \tag{1}$$

$$p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) = \mathcal{G}(\ldots, g_n(\boldsymbol{z}_i), \ldots), \quad n = 1, 2, \ldots \tag{2}$$

$\boldsymbol{x}_i^{(l)}$ and $\boldsymbol{x}_i^{(h)}$ are conditional independent:

$$p_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) = p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i)\, p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) \tag{3}$$

where, $\mathcal{F}$ and $\mathcal{G}$ are some distributions, $\{f_m\}$ and $\{g_n\}$ are the corresponding sufficient statistics which are deterministic functions w.r.t. the latent variable $\boldsymbol{z}_i$ with parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, respectively. In terms of their highly expressive ability, deep networks are used to realize these functions, usually non-linear, so that the generative process describes the nonlinear directed-relationship from the distribution of the latent representation to the distributions of the observed spaces, which should be more powerful than the linear coding-based methods in [8] and [21].

The objective function is to maximize the joint full likelihood of the observed $\mathbf{X}_l$ and $\mathbf{X}_h$ as follows:

$$\log p(\mathbf{X}_l, \mathbf{X}_h) \tag{4}$$

$$= \sum\nolimits_{i=1}^{N} \log p(\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)}) \tag{5}$$

$$= \mathcal{L}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\phi}; \mathbf{X}_l, \mathbf{X}_h) + \sum_{i=1}^{N} \mathcal{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)\middle|\middle| p(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)})\right] \tag{6}$$

where, a parameterized variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)$, also called recognition model, is applied to approximate the intractable true posterior $p(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)})$ in Kullback–Leibler (KL) divergence, denoted as $\mathcal{KL}[\cdot||\cdot]$. Since the KL divergence in the second term is non-negative, optimizing (6) is equivalent to maximizing the first term, i.e., the evidence lower bound:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\phi}; \mathbf{X}_l, \mathbf{X}_h) = &\sum\nolimits_{i=1}^{N} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)}[\ln p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i)] \\ &+ \sum\nolimits_{i=1}^{N} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)}[\ln p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i)] \\ &- \sum\nolimits_{i=1}^{N} \mathcal{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)||p(\boldsymbol{z}_i)] \qquad (7)\end{aligned}$$

where, the first and second terms respectively represent the expected negative reconstruction errors of the LR and HR training patches with the latent representations drawn from the optimal variational distributions; the third term measures the KL divergence between the variational distribution and the prior distribution of the latent representation, acting as a regularizer to affect the model generalization. It is interesting to notice that the second term plays the role gathering the recognition model and the HR generative model, describing the regression process to the distribution of the HR patch, i.e., the SR process. The robustness of the model to noise is addressed from two aspects: first, the model infers the stochastic distribution of the latent variable as well as the observed space, instead of point estimate; second, the inferred latent representation tries to restore a clean LR patch via the first term, since, as claimed in [37], the variational distribution can be viewed as corrupting the latent representation with Gaussian noise, and the noise level of the latent representation is automatically learned through the recognition model, which leads to more robust and systematic learning of latent representation regardless of the data corruption scheme. Thanks to the third term, the prior information is easily brought in to affect the SR process.

By applying the stochastic gradient variational Bayes (SGVB) [40], inferring the latent representation $\boldsymbol{z}_i$ and learning the parameters $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, and $\boldsymbol{\phi}$ are simultaneously realized.

The proposed PGM is a general and flexible framework: 1) the choice of the conditional distributions is flexible, depending on the type of data, e.g., Gaussian for continuous-valued data, and Bernoulli for binary-valued data; 2) the choice of the variational distribution is flexible based on different observations; 3) the prior information w.r.t. the latent representation can be delivered via a suitable $p(\boldsymbol{z}_i)$; 4) different networks can be embedded into the PGM, such as multilayer perceptron (MLP) and convolutional neural network (CNN).

3. Probabilistic generative model with conditional prior (PGM-CP)

In this section, we use our framework to guide the development of a novel SR model based on specific choices of the conditional distributions $p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i)$, $p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i)$, the prior distribution $p(\boldsymbol{z}_i)$, and the variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)$.

3.1. Expressive likelihood

Considering the image patches are continuous-valued, both of the conditional distributions in (1) and (2) are designed as Gaussian distributions:

$$p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i) = \mathcal{N}(\boldsymbol{\mu}_{p,(l)}(\boldsymbol{z}_i), \boldsymbol{\sigma}^2_{p,(l)}(\boldsymbol{z}_i)\mathbf{I}), \qquad (8)$$

$$p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) = \mathcal{N}(\boldsymbol{\mu}_{p,(h)}(\boldsymbol{z}_i), \boldsymbol{\sigma}^2_{p,(h)}(\boldsymbol{z}_i)\mathbf{I}), \qquad (9)$$

where, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2\mathbf{I})$ denotes a Gaussian distribution with mean vector $\boldsymbol{\mu}$ and diagonal covariance matrix $\boldsymbol{\sigma}^2\mathbf{I}$ with $\boldsymbol{\sigma}^2$ being its diagonal elements, $\mathbf{I}$ is the unit matrix. Both $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ are non-linear functions with respect to the latent variable $\boldsymbol{z}_i$, where the subscript p (or q in (11)) is to highlight that the $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ belong to the generative (or recognition) model, while the subscripts (l) and (h) highlight the $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ belong to the LR or the HR generative path, respectively. Therefore, $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ contain all the parameters of the networks to be learned.

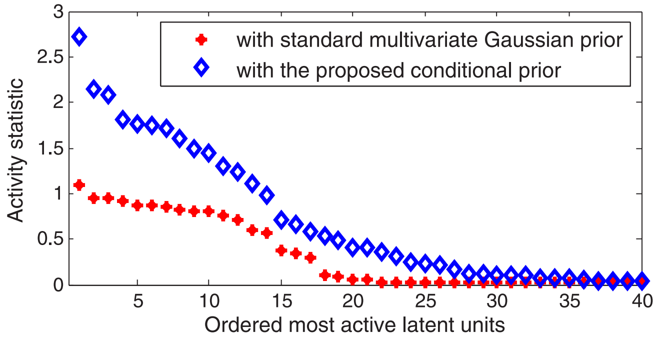


Fig. 1. Ordered activity values of the latent units. The activity statistic is defined as $A_u = Cov(\mathbb{E}_{u\sim q_{\boldsymbol{\phi}}(u|\cdot)}[u])$ following [41], and measured on the Set14 dataset [42] with a hidden layer of 200 units. As opposed to only 20 active units with the prior $\mathcal{N}(\mathbf{0}, \mathbf{I})$, the informative conditional prior helps to produce more active latent units and also promote the activity values of those units.

3.2. Conditional prior

According to the objective function (7), the prior distribution $p(\boldsymbol{z}_i)$ plays an important role in regularizing the variational distribution which affects the inference of the latent representation indirectly.

A naive choice is $p(\boldsymbol{z}_i) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ as used in the VAE model [40], but have several drawbacks. A commonly known problem is self-prune [43], namely only limited degrees of freedom are used, such that the model's capacity is weakened. As shown in Fig. 1, most of the latent units drawn from the variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)$ are inactive, which is attributed to the facts that $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)$ is driven towards the prior $\mathcal{N}(\mathbf{0}, \mathbf{I})$ during the early training and unable to be resurrected [44]. More importantly, uni-modal prior is insufficient in describing the real-world data distributions which are often complex and multi-modal, and limits the recognition model to capture more complex distributions.

Since conditional prior is believed to have the ability of modelling multiple modes [45], we propose an informative conditional prior to solve the above two issues. We assume that the prior of the latent representation is conditioned on the HR observations, whose sufficient statistics are parameterized by neural networks, formally as:

$$p_{\boldsymbol{\omega}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(h)}) = \mathcal{N}(\boldsymbol{\mu}_{prior}(\boldsymbol{x}_i^{(h)}), \boldsymbol{\sigma}^2_{prior}(\boldsymbol{x}_i^{(h)})\mathbf{I}), \qquad (10)$$

which establish a channel for the information transforming from the HR patch to its latent representation. Since both the prior and the recognition model embrace the mapping from the observed space to the latent space and are constrained in KL divergence, more unique details contained in the HR patches can influence the recognition model to reveal a more meaningful latent space. Moreover, the conditional prior in (10) and the LR generative model in (8) implies the degradation process from the HR patch to the corresponding LR patch.

Although we assume that every point in the latent space follows its own distribution, the parameters in $\boldsymbol{\omega}$ are global that can be efficiently inferred.

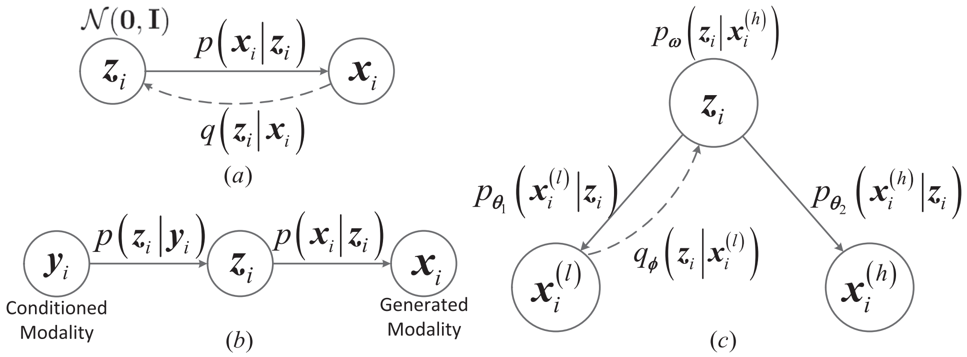

$\mathcal{N}(\mathbf{0},\mathbf{I})$
$p(\boldsymbol{x}_i|\boldsymbol{z}_i)$
$\boldsymbol{z}_i$
$\boldsymbol{x}_i$
$q(\boldsymbol{z}_i|\boldsymbol{x}_i)$
(a)
$\boldsymbol{y}_i$
$p(\boldsymbol{z}_i|\boldsymbol{y}_i)$
$\boldsymbol{z}_i$
$p(\boldsymbol{x}_i|\boldsymbol{z}_i)$
$\boldsymbol{x}_i$
Conditioned Modality
Generated Modality
(b)
$p_{\omega}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(h)})$
$\boldsymbol{z}_i$
$p_{\theta_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i)$
$p_{\theta_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i)$
$q_{\phi}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})$
$\boldsymbol{x}_i^{(l)}$
$\boldsymbol{x}_i^{(h)}$
(c)

Fig. 2. The graphical illustration of (a) the VAE model in [40], (b) the conditional modality learning in [46], and (c) the proposed PGM-CP, where the generative and recognition models are represented by solid lines and dashed lines, respectively.

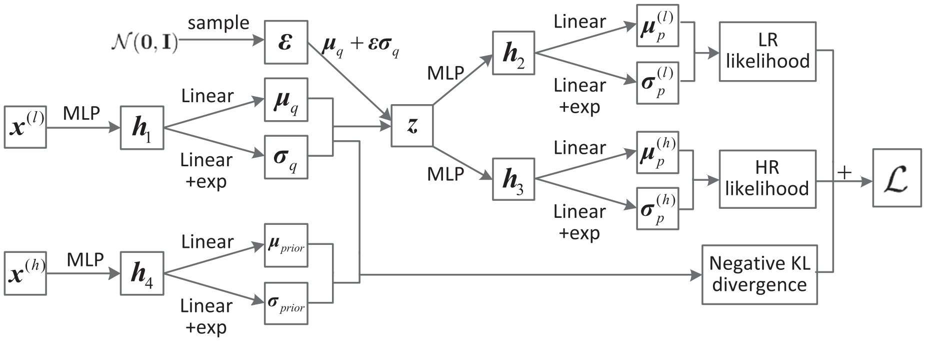


Fig. 3. Illustration of the networks learned in PGM-CP. The subscript “i” is omitted.

3.3. Consistent recognition model

In both training and testing, one crucial step is to infer the latent representation via the recognition model $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\cdot)$, which is an approximation to the true posterior $p(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)},\boldsymbol{x}_i^{(h)})$. Such posterior is different from the posterior $p(\boldsymbol{z}_i|\boldsymbol{x}_i)$ in the VAE model in [40], because it is conditioned on binary modalities in our model, but on the single modality in the VAE model. Since the HR modality is unobserved during testing, training the recognition model via binary-modality fusion, i.e., $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)},\boldsymbol{x}_i^{(h)})$, might be a good approximation to the true posterior on the training dataset, but it is clumsy to be used on the testing dataset with the HR patches unobserved.

In order to maintain the consistency in both training and testing, we employ an uniform recognition model which enforces the LR patches as inputs. Formally, the variational distribution is written as:

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})=\mathcal{N}(\boldsymbol{\mu}_q(\boldsymbol{x}_i^{(l)}),\boldsymbol{\sigma}_q^2(\boldsymbol{x}_i^{(l)})\mathbf{I}), \tag{11}$$

where, a Gaussian is chosen for the purpose of reparameterization [40]. A major benefit of the consistent recognition model is that the recognition network can be directly used for testing data without running any optimization.

3.4. Graphical model and network implementation

According to the above analysis, the graphical model of the proposed PGM-CP is illustrated in Fig. 2, where two other existing models are also illustrated for better comparison. It can be seen that the shared latent variable establishes a bridge between an LR patch and its corresponding HR patch. If we consider the HR patch as our target, the PGM-CP is a supervised generative model aiming at image SR, more than representing the visible data like the unsupervised generative model, VAE in Fig. 2(a). Besides, compared with conditional modality learning in Fig. 2(b) modelling the conditional likelihood of one modality given the other, we are interested in the joint full likelihood of both modalities. Based on the probabilistic modelling analyzed before, commonly used basic networks are embedded in our model to realize the nonlinear mean and covariance functions in (8)–(11).

Here, we use the following equations as an example to illustrate the network implementation of the distribution in (11). Firstly, a hidden layer $\boldsymbol{h}_1$ is obtained via a MLP.

$$\boldsymbol{h}_1=\mathrm{MLP}(\boldsymbol{x}^{(l)})=\tanh(\mathbf{W}_1\boldsymbol{x}^{(l)}+\boldsymbol{b}_1), \tag{12}$$

which is a single-layer MLP and can be extended into multiple layers. The activation function tanh is chosen following [40]. And then, the sufficient statistics are realized by

$$\boldsymbol{\mu}_q=\mathrm{Linear}(\boldsymbol{h}_1)=\mathbf{W}_2\boldsymbol{h}_1+\boldsymbol{b}_2, \tag{13}$$

$$\boldsymbol{\sigma}_q=\exp(\mathbf{W}_3\boldsymbol{h}_1+\boldsymbol{b}_3). \tag{14}$$

Thus, the proposed probabilistic generative model appears as a complex network as shown in Fig. 3, which is distinct from the regression-based methods [30–34]. On the one hand, compared with them applying point estimates, the PGM-CP prefer distribution estimates, modeling the sufficient statistics. On the other hand, compared with them only considering the regression ability from the LR patch to the HR patch, the PGM-CP also take the robustness and regularization into account, namely the LR generative term and the KL divergence term.

As a result, the proposed probabilistic generative model can be efficiently optimized via backpropagation (BP) [47] and stochastic optimization methods such as Adam [48].

3.5. Image super-resolution

During the testing stage, the recognition model and the HR generative model can be concatenated as a whole path to serve the SR task, as shown in Figs. 2 and 3. The ith reconstructed high-resolution patch $\hat{\boldsymbol{y}}_i$ is obtained via a simple and efficient forward mapping:

$$\hat{\boldsymbol{y}}_i = \boldsymbol{\mu}_{p,(h)}(\boldsymbol{\mu}_q(\boldsymbol{y}_i^{(l)})). \tag{15}$$

which is expressive enough owing to the non-linearity, such that there is no need to perform feature extraction in advance like [8]. Actually, all the patches are expressed by stacking the original pixels as column vectors.

After patch-wise super-resolution, we put the patches $\{\hat{\boldsymbol{y}}_i\}$ into the corresponding locations. The overlapping patches are averaged to produce the reconstructed image $\hat{\mathbf{Y}}_h$. Following the strategy used in [8,21], a global fine-tuning is applied by computing:

$$\mathbf{Y}_h^* = \arg\min_{\mathbf{Y}_h} \|SH\mathbf{Y}_h - \mathbf{Y}_l\| + c\|\mathbf{Y}_h - \hat{\mathbf{Y}}_h\|_2^2 \tag{16}$$

where, S represents the downsampling operator, H a blurring filter, c is a constant, set as 0.01 in our experiment. All of them are set according to [8]. The whole process of the super-resolution realization is summarized in Algorithm 1.

Algorithm 1 PGM-CP for super-resolution.

```
Input: A set of N training data points $\{\mathbf{X}_l, \mathbf{X}_h\} = \{\boldsymbol{x}_i^{(l)}, \boldsymbol{x}_i^{(h)}\}_{i=1}^N$, a low-resolution image $\mathbf{Y}_l$.
Training stage:
  • Initialize $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, $\boldsymbol{\phi}$, and $\boldsymbol{\omega}$
  • Repeat until convergence
      Gradients estimation w.r.t. $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, $\boldsymbol{\phi}$, and $\boldsymbol{\omega}$ based on mini-batches using BP;
      Parameters update using the Adam algorithm.
Testing stage:
  • patch-wise super-resolution
      $\hat{\boldsymbol{y}}_i = \boldsymbol{\mu}_{p,(h)}(\boldsymbol{\mu}_q(\boldsymbol{y}_i^{(l)}))$.
  • fine tuning
      $\mathbf{Y}_h^* = \arg\min_{\mathbf{Y}_h} \|SH\mathbf{Y}_h - \mathbf{Y}_l\| + c\|\mathbf{Y}_h - \hat{\mathbf{Y}}_h\|_2^2$
Output: super-resolution image $\mathbf{Y}_h^*$
```

4. Examples under the proposed framework

For better understanding the inclusivity and flexibility of our proposed framework PGM and make concrete comparisons with some coding-based and regression-based methods, we explicitly state how recently developed SR methods ScSR [8], BPFASR [21], and SRCNN [39] partially fall within the proposed framework.

4.1. Coding-based examples

Specify that the conditional distributions in (1) and (2) are Gaussian distributions as follows:

$$p_{\boldsymbol{\theta}_1}(\boldsymbol{x}_i^{(l)}|\boldsymbol{z}_i) = \mathcal{N}(\mathbf{D}_l\boldsymbol{z}_i, \alpha_l^{-1}\mathbf{I}), \tag{17}$$

$$p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) = \mathcal{N}(\mathbf{D}_h\boldsymbol{z}_i, \alpha_h^{-1}\mathbf{I}), \tag{18}$$

where the mean vectors are linear functions, and the covariance matrices as constant matrices.

As the elements in the latent variable $\boldsymbol{z}_i$ are i.i.d. drawn from a zero-mean Laplace distribution, i.e., $z_{i,k} \sim Laplace(0, 2\beta^{-1})$, where $z_{i,\,k}$ is the k-th element of $\boldsymbol{z}_i$, the ScSR [8] can be derived by minimizing the negative logarithm of the posterior density function as follows:

$$-\ln p(\mathbf{D}_l, \mathbf{D}_h, \mathbf{Z}|\mathbf{X}_l, \mathbf{X}_h) \tag{19}$$

$$\propto \sum_{i=1}^{N}\left\{\alpha_l\|\boldsymbol{x}_i^{(l)} - \mathbf{D}_l\boldsymbol{z}_i\|_2^2 + \alpha_h\|\boldsymbol{x}_i^{(h)} - \mathbf{D}_h\boldsymbol{z}_i\|_2^2 + \beta\sum_{k=1}^{K}|\boldsymbol{z}_{i,k}|\right\} \tag{20}$$

$$= \alpha_l\|\mathbf{X}_l - \mathbf{D}_l\mathbf{Z}\|_2^2 + \alpha_h\|\mathbf{X}_h - \mathbf{D}_h\mathbf{Z}\|_2^2 + \beta\|\mathbf{Z}\|_1 \tag{21}$$

where, $\mathbf{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N]$.

As the prior distribution of the latent variable is chosen as a beta-Bernoulli process as stated in [21,49]:

$$\boldsymbol{z}_i = \boldsymbol{v}_i \odot \boldsymbol{s}_i, \tag{22}$$

$$v_{i,k} \sim Bernoulli(\pi_k), \tag{23}$$

$$\pi_k \sim Beta(c\eta, c(1-\eta)), \tag{24}$$

$$\boldsymbol{s}_i \sim \mathcal{N}(\mathbf{0}, \alpha_s^{-1}), \tag{25}$$

the BPFASR [21] model is aroused.

According to the above analysis, both ScSR and BPFASR are evolved from the same likelihood setting but different uni-modal priors. By comparing (17) and (18) with (8) and (9), it is clear that our model can better express the relationship between the latent representation and the LR and HR patches owing to the non-linear functions.

Although the models in [8] and [21] can be reinvented under the proposed framework, the inferences and learnings are quite different from the PGM. In [8], the ScSR is optimized as an l_1-regularization problem which is solved by iteratively optimizing the sparse codes $\{\boldsymbol{z}_i\}$ and the dictionaries $\mathbf{D}_l$ and $\mathbf{D}_h$. In [21], the BPFASR can be inferred using Gibbs sampling, variational inference (VI), or online VI. All those inference methods are time-consuming for test, since they need to infer the latent representation iteratively. Whereas, the generative and recognition models are jointly optimized in our framework, leading an efficient nonlinear feedforward mapping for test.

4.2. Regression-based example

Specify that the conditional distributions in (2) and the variational distribution in (11) are Gaussian distributions as follows:

$$p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i) = \mathcal{N}(\mathrm{CNN}_1(\boldsymbol{z}_i), \mathbf{I}), \tag{26}$$

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)}) = \mathcal{N}(\mathrm{CNN}_2(\boldsymbol{x}_i^{(l)}), \gamma^2\mathbf{I}), \tag{27}$$

where, the $\mathrm{CNN}_1(\cdot)$ and $\mathrm{CNN}_2(\cdot)$ are nonlinear functions realized by convolutional neural networks.

As the γ^2 approaches to zero, the limit of the Normal distribution in (27) is a dirac delta distribution $\delta\big(\boldsymbol{z}_i; \mathrm{CNN}_2(\boldsymbol{x}_i^{(l)})\big)$ that satisfies

$$\int F(\boldsymbol{z}_i)\delta(\boldsymbol{z}_i)d\boldsymbol{z}_i = F(\mathrm{CNN}_2(\boldsymbol{x}_i^{(l)}))\big) \tag{28}$$

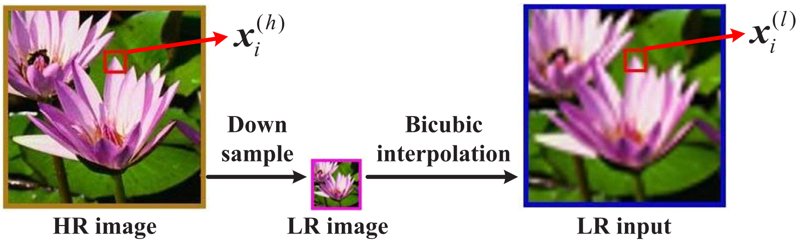


Fig. 4. The procedure of constructing a pair of LR-HR images and a pair of LR-HR patches.

for all continuous function F. As the function F is identified as $\|\boldsymbol{x}_i^{(h)} - \mathrm{CNN}_1(\boldsymbol{z}_i)\|^2$, we have:

$$-\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})}\left[\ln p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i)\right] \tag{29}$$

$$= -\int \left[\ln p_{\boldsymbol{\theta}_2}(\boldsymbol{x}_i^{(h)}|\boldsymbol{z}_i)\right] q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)}) d\boldsymbol{z}_i \tag{30}$$

$$\propto \int \left\|\boldsymbol{x}_i^{(h)} - \mathrm{CNN}_1(\boldsymbol{z}_i)\right\|^2 \delta\left(\boldsymbol{z}_i; \mathrm{CNN}_2(\boldsymbol{x}_i^{(l)})\right) d\boldsymbol{z}_i \tag{31}$$

$$= \left\|\boldsymbol{x}_i^{(h)} - \mathrm{CNN}_1\left(\mathrm{CNN}_2(\boldsymbol{x}_i^{(l)})\right)\right\|^2 \tag{32}$$

which describe the mean squared error of the CNN regression problem for super-resolution, coincident with the loss function in [39].

In other words, under the above assumptions, the SRCNN model can be derived from the second term in (7) via discarding the randomness of the latent representation. In contrast to the SRCNN only considering the regressive ability to the training data, our framework also take the model's robustness and generalization into account which are reflected by the other two terms in (7).

Clearly, the PGM is a flexible probabilistic model that implies more constraints than the purely network-based method, e.g., SRCNN. Thus, our model has the potential that using relatively less data but achieving comparable results as the SRCNN.

5. Experiments

In this section, we present experimental results on three commonly used datasets to illustrate the effectiveness and efficiency of the proposed PGM-CP for the single image super-resolution task. We compare our approach with Bicubic interpolation, two coding-based methods ScSR [8] and BPFASR [21], and four regression-based methods SRCNN [39], SRCDA [30], SRGAN [34], and LapSRN [50].

5.1. Data and setting

We adopt a dataset consisting of 91 images [8] for training, and perform evaluations on three widely used benchmark datasets Set5 [51], Set14 [42] and BSD100 [52] with scaling factors 2, 3, and 4. Following the protocols in [21,39], the LR images are synthesized via down-sampling the ground-truth HR images, which are subsequently up-scaled with equal scaling factor via bicubic interpolation to form the LR inputs of our model, as shown in Fig. 4. The same-sized patches extracted from the same locations in both the LR input and the HR ground truth are treated as LR-HR patch pairs.

We set the patch size as 8 × 8, 10 × 10, and 12 × 12 when using scaling factors 2, 3, and 4, respectively. After discarding some smooth and similar ones [8], we obtain about $100,000$ pairs of training patches. For fair comparison, following [8,21,34,39], we evaluate all algorithms only on the luminance channel (Y channel in YCbCr color space) and apply Bicubic interpolation on the other channels (Cb and Cr) just for the purpose of displaying.

5.2. Analysis on network structures

As shown in Fig. 3, the model consists of four basic networks respectively for prior distribution, variational distribution, LR and HR conditional distributions. We restrict that the recognition network mirrors the generative network, and that the prior network have the same structure with the recognition network[1]. The parameters of these networks are not shared. Thus, the model architecture is relevant with the network structure from three aspects: the dimension of the latent representation $\boldsymbol{z}$, the dimension of the hidden layers $\boldsymbol{h}$, and the depth, which are reflected by three groups of experiments. What calls for special attention is that the hidden layers are different from the latent representation, since $\boldsymbol{h}$ is deterministic, while $\boldsymbol{z}$ is a stochastic variable.

All the parameters are initialized by random sampling from $\mathcal{N}(0, 0.01)$, and we optimize them via Adam algorithm [48]. The model is trained with 2×10^6 backpropagations, about 2 h.

The comparisons are made on the Set14 dataset according to the SR performance on mean peak signal-to-noise ratio (PSNR) when scaling factor is 2, as shown in Table 1.

It can be seen that, enlarging the dimension of the latent representation or the dimensions of the hidden layers can moderately improve the PSNR performance. Three controlled experiments are conducted with different numbers of hidden layers. Among them, the 300-300-200 performs the best on PSNR, which illustrates that the model could benefit from increasing the depth of the network to a certain degree thanks to the expressive ability, and that a deeper model does not always result in better performance. Because, the deeper, the more parameters the model has to estimate.

Based on these analysis, the structure 400-400-200 is used for further comparisons in the following subsections.

5.3. Analysis on the prior distribution

Since the prior distribution could affect inferring the latent space through the KL divergence $\mathcal{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})||p(\boldsymbol{z}_i|\boldsymbol{x}_i^{(h)})]$, as shown in (7), we look in sight into the latent space obtained by the recognition model to explore the effectiveness of the proposed conditional prior indirectly. After the model is well trained, we choose two different images corresponding to a girl and a zebra, respectively, and use the recognition model $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})$ to infer the latent representations of the patches extracted from these images.

[1] The network structure is simplified notated. Take 300-100-200 as an example: the inference network and the prior network respectively has two hidden layers with 300 and 100 units successively; the dimension of latent representation $\boldsymbol{z}$ is 200; every generative network has two hidden layers with 100 and 300 units successively.

Table 1
Average PSNR (dB) with Different Model Structures on the Set14 Dataset with Scaling Factor 2.

	Different dimensions of $\boldsymbol{z}$		Different network depths		Different dimensions of $\boldsymbol{h}$	
Methods	Structure	PSNR	Structure	PSNR	Structure	PSNR
PGM-UP	300-100	31.53	300-200	31.56	200-200-200	31.71
	300-200	31.56	300-300-200	31.84	300-300-200	31.84
	300-300	31.60	300-300-300-200	31.69	400-400-200	**31.91**
PGM-CP	300-100	31.98	300-200	31.99	200-200-200	32.15
	300-200	31.99	300-300-200	32.26	300-300-200	32.26
	300-300	32.07	300-300-300-200	32.10	400-400-200	**32.32**

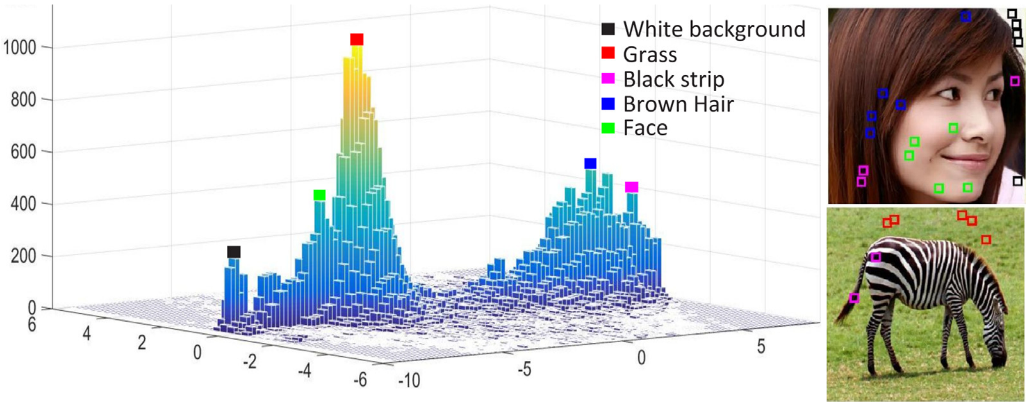


Fig. 5. Histogram illustration of the 2D principle components of the latent representations drawn from the recognition model $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i^{(l)})$ corresponding to the patches extracted from the images on the right after training the model of PGM-CP. For each mode, the nearest 5 points are marked on the images using the same color.

Table 2
Average Test Time (Sec.) Comparison on the Set14 Dataset with Scaling Factor 2.

Coding-based methods		Regression-based methods				Hybrid methods	
ScSR [8]	BPFASR [21]	SRCNN [39]	SRCDA [30]	SRGAN [34]	LapSRN [50]	PGM-UP	PGM-CP
130	796	0.24	0.20	2.64	1.08	0.38	0.38

We adopt 2D principle component analysis to observe the distributions of the inferred latent representations, which are illustrated in Fig. 5. It is clear that the conditional prior helps the recognition model to reveal a meaningful latent space, where patches with different texture are distributed apart, while patches with similar texture are distributed closely.

In order to further demonstrate the effect of the conditional prior on the SR performance more directly, we conduct a contrast experiment via replacing the conditional prior with a uni-prior $\mathcal{N}(\mathbf{0}, \mathbf{I})$, term as PGM-UP. The results listed in Tables 1 and 2 are obtained via training the PGM-UP with the same experimental setting as the PGM-CP. It can be seen that the test running time is not influenced by the complexity of the prior distribution, because the prior model is not utilized during the testing phase. However, the model indeed benefits from the proposed conditional prior, thanks to the influence of the prior model during the training phase.

5.4. Analysis on the learned dictionary

As a generative model, the diversity of the learned dictionary atoms are vital to the final performance [53], so we look insight into the weight matrix in the last layer of the HR generative network, which is equivalent to a dictionary to some extent. Fig. 6 illustrates part of the learned atoms, which show diverse meaningful structures, like arc, corner, edge, point, center-surrounding, etc. For quantitative comparison, we make histogram statistics based on the entropy of every atom. According to Fig. 7, the entropy values of PGM-CP have a wider range and distribute more balance, which further validate its high diversity.

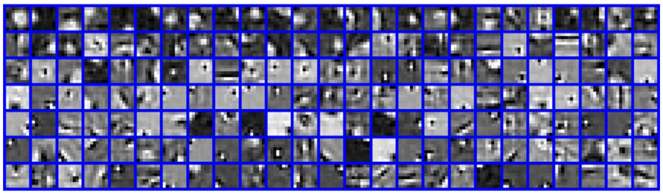


Fig. 6. Top 175 informative atoms learned with scaling factor 2 arranged according to the entropy in descending order.

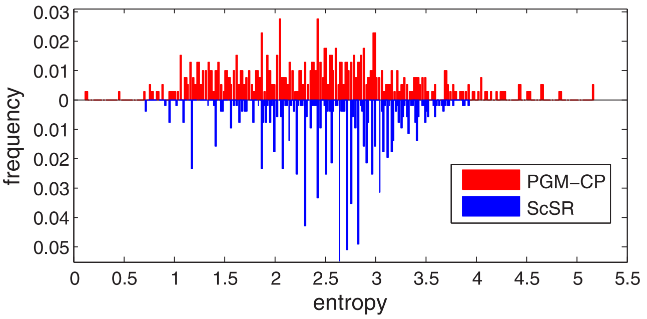
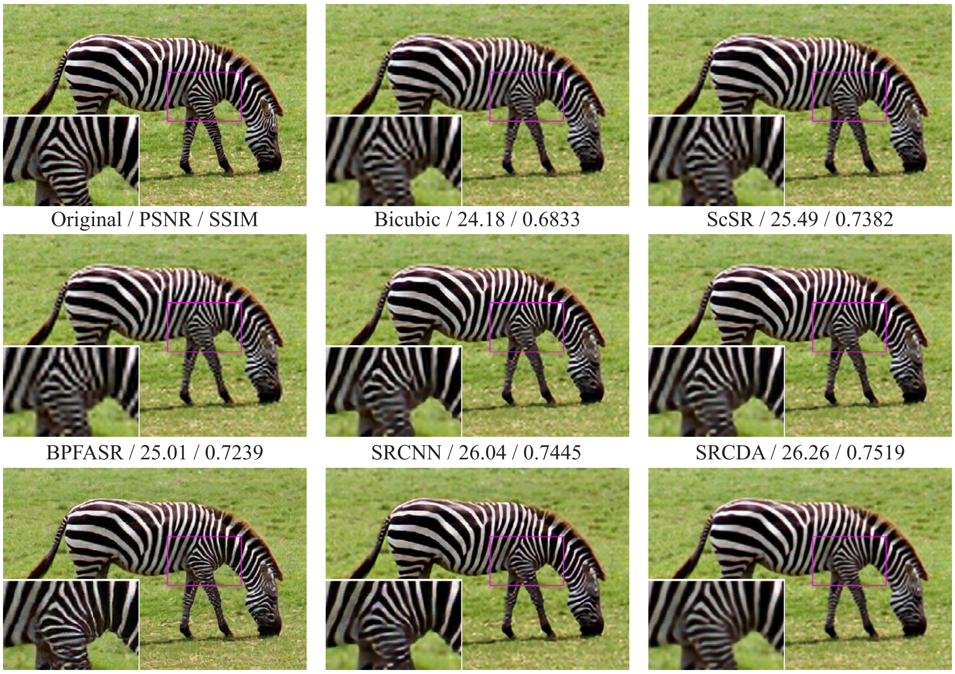


Fig. 7. Texture diversity comparison between all the 400 atoms of PGM-CP and all the 512 atoms of ScSR with scaling factor 2.



Original / PSNR / SSIM

Bicubic / 24.18 / 0.6833

ScSR / 25.49 / 0.7382

BPFASR / 25.01 / 0.7239

SRCNN / 26.04 / 0.7445

SRCDA / 26.26 / 0.7519

SRGAN / 24.95 / 0.6831

LapSRN / 26.31 / 0.7511

PGM-CP / **26.60** / **0.7631**

Fig. 8. The "zebra" image from Set14 with scaling factor 4.

5.5. Comparison with related works

We compare the PGM-CP with structure 400-400-200 with some state-of-the-art approaches, including Bicubic interpolation, two coding-based methods ScSR [8] and BPFASR [21], and four regression-based methods SRCNN [39], SRCDA [30], SRGAN [34], and LapSRN [50]. It should be noted that the SRCDA is a model that initialized as two coupled autoencoders and further fine-tuned as a regression problem, which seems to be a combination of the coding and regression. That is the reason why we choose the SRCDA for comparison. The list results are cited from the published papers, got via running the publicly available codes and from the online resources. For fair comparison, we employ the training dataset without augmentation [50] nor a larger dataset like ImageNet [32]; moreover, the reconstructed images are evaluated without shaved, since their is no border effect produced by the convolutional operators like [39] and [34].

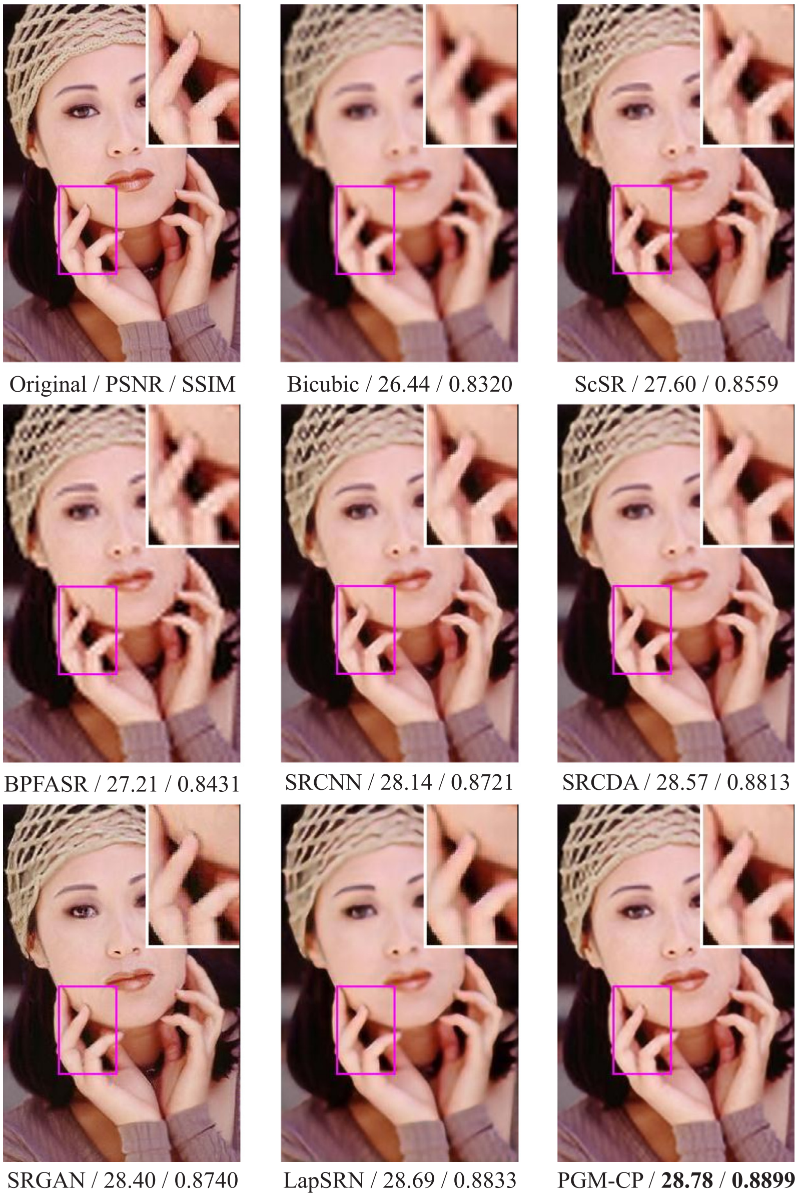
Firstly, the SR performances under the common scenario are evaluated based on the criteria of the PSNR, the structural similarity (SSIM) [54], and the average testing time.

As the quantitative results shown in Table 3, the PGM-CP yields the competitive scores on all the three datasets. Compared with the coding-based methods ScSR and BPFASR, the superior performance validate the effectiveness of the nonlinear expression of the proposed PGM-CP. Besides, although the PGM-CP is implemented using relatively simple networks, i.e., MLPs, have not considering convolutional operators, the PGM-CP achieves better scores than the SRCNN, SRGAN, and the LapSRN, which might be attributed to the KL divergence that can utilizing prior information. Except for SRGAN and LapSRN that are pretrained or trained on large datasets, all the other methods use the same training dataset as ours. As analyzed in Section 4, the SRCNN can be reinvented under the proposed framework, but with less constrains than the PGM-CP. As the comparison shown in Table 4, unlike SRCNN that need large dataset in exchange for better performance, the PGM-CP can achieve comparable performance with relatively smaller set of training data, because the PGM-CP is a probabilistic model where the priors bring more constraints than those deterministic models.

As the visual comparisons shown in Figs. 8 and 9, the PGM-CP produces images with sharp edges and without any obvious artifacts. Note that although the reconstructed zebra of SRGAN looks sharper than ours, the SRGAN is more likely to produce artifacts, as shown in Fig. 8. More visual results can be found in Appendix I.

The average testing time of each image on the whole dataset with scaling factor 2 is shown in Table 2. Except that the ScSR and BPFASR are evaluated with 3.6GHz CPU, all the other methods are evaluated with Tesla K40 GPU, since both of ScSR and BPFASR need iterative inference during the test, which barriers their testing stage to be parallelized. In contrast, with the combination of the recognition model and the generative model, the proposed PGM-CP realizes the SR process through an efficient forward mapping. Besides, further acceleration is realized via processing patches in parallel. Thus, the proposed PGM-CP accelerates the testing speed, compared with those coding-based methods. In addition, the PGM-CP is comparable with regression-based methods and superior than the SRGAN and the LapSRN which are built upon a much deeper network.

In order to assess the robustness of these methods, a more challenging scenario is used, where the test images are corrupted by zero-mean Gaussian noise with different standard deviations. Given the noisy images, we iteratively perform the following steps before image super-resolution: (1) using the recognition model and LR generative model to obtain a reconstructed LR image; (2) using the reconstructed LR image as the input for the next round. Since their is no available well-learned model provided by the authors in [34], and the generative adversarial network is hard to train, for fair comparisons, here we do not evaluate the robustness of the SRGAN. Basically, according to the results in Figs. 10 and 11, the coding-based methods ScSR and BPFASR show superior performance than the regression-based methods, SRCNN, SRCDA, and

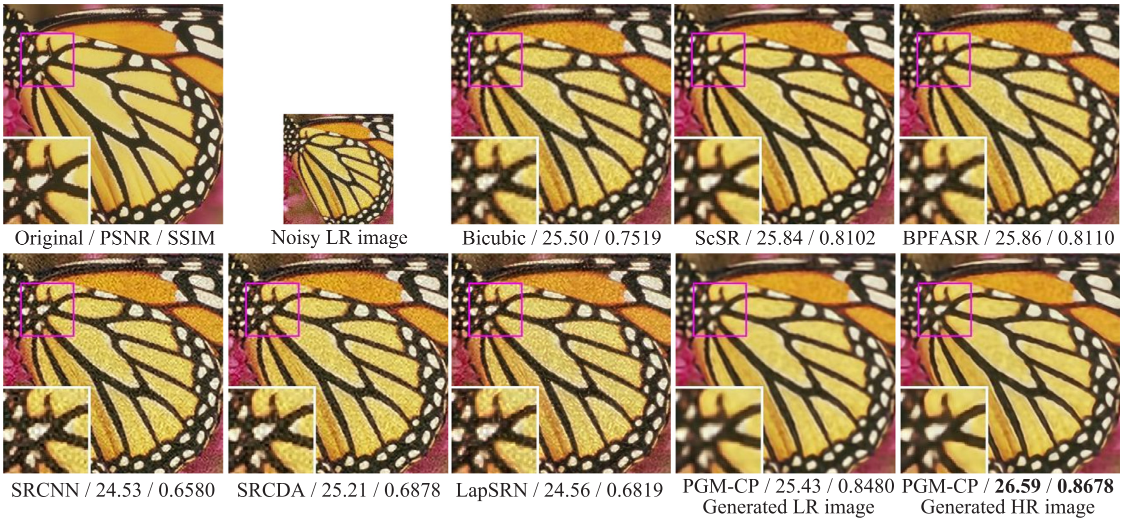


Original / PSNR / SSIM Bicubic / 26.44 / 0.8320 ScSR / 27.60 / 0.8559

BPFASR / 27.21 / 0.8431 SRCNN / 28.14 / 0.8721 SRCDA / 28.57 / 0.8813

SRGAN / 28.40 / 0.8740 LapSRN / 28.69 / 0.8833 PGM-CP / **28.78** / **0.8899**

Fig. 9. The “woman” image from Set5 with scaling factor 4.



Original / PSNR / SSIM Noisy LR image Bicubic / 25.50 / 0.7519 ScSR / 25.84 / 0.8102 BPFASR / 25.86 / 0.8110

SRCNN / 24.53 / 0.6580 SRCDA / 25.21 / 0.6878 LapSRN / 24.56 / 0.6819 PGM-CP / 25.43 / 0.8480 Generated LR image PGM-CP / **26.59** / **0.8678** Generated HR image

Fig. 10. The noisy “butterfly” from Set5 with scaling factor 2. The standard deviation of the Gaussian noise is 0.04.

Table 3
Average PSNR (dB) and SSIM on the Set5, the Set14, and the BSD100.

	Scale		Bicubic	ScSR [8]	BPFASR [21]	SRCNN[a] [39]	SRCDA [30]	SRGAN[b] [34]	LapSRN [50]	PGM-CP
Set5	2	PSNR	33.68	35.27	35.23	36.31	36.40	–	34.72	**36.75**
		SSIM	0.9305	0.9450	0.9437	0.9527	0.9531	–	0.9390	**0.9553**
	3	PSNR	30.40	31.42	31.38	32.34	32.61	–	–	**32.86**
		SSIM	0.8685	0.8821	0.8822	0.9035	0.9083	–	–	**0.9103**
	4	PSNR	28.43	29.51	29.24	30.03	30.32	29.40	29.83	**30.51**
		SSIM	0.8107	0.8383	0.8241	0.8530	0.8613	0.8472	0.8559	**0.8632**
Set14	2	PSNR	30.00	31.34	31.25	31.81	31.96	–	31.29	**32.32**
		SSIM	0.8693	0.8963	0.8967	0.9044	0.9039	–	0.8948	**0.9074**
	3	PSNR	27.32	28.30	28.09	28.65	28.82	–	–	**29.15**
		SSIM	0.7746	0.8104	0.8015	0.8152	0.8194	–	–	**0.8205**
	4	PSNR	25.78	26.49	26.38	26.86	26.90	26.02	26.55	**27.23**
		SSIM	0.7031	0.7348	0.7256	0.7421	0.7450	0.7397	0.7471	**0.7486**
BSD100	2	PSNR	29.57	30.77	30.56	31.11	30.43	–	–	**31.39**
		SSIM	0.8440	0.8744	0.8687	0.8835	0.8668	–	–	**0.8868**
	3	PSNR	27.22	27.72	27.53	28.20	28.04	–	–	**28.46**
		SSIM	0.7399	0.7647	0.7531	0.7794	0.7749	–	–	**0.7844**
	4	PSNR	25.99	26.61	26.49	26.70	26.68	25.16	–	**27.01**
		SSIM	0.6695	0.6983	0.6872	0.7018	0.7042	0.6688	–	**0.7105**

[a] The well-learned models of SRCNN are available at http://mmlab.ie.cuhk.edu.hk/projects/SRCNN.html.
[b] The reconstructed images of SRGAN are available at https://twitter.box.com/s/lcue6vlrd01ljkdtdkhmfvk7vtjhetog.

Table 4
Results on different training datasets with scaling factor 2.

Methods Training datasets	SRCNN ImageNet		SRCNN 91 Images		PGM-CP 91 Images	
Criteria	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Set5	36.66	0.9542	36.31	0.9527	**36.75**	**0.9553**
Set14	**32.45**	0.9067	31.81	0.9044	32.32	**0.9074**
BSD100	31.32	0.8851	31.11	0.8835	**31.39**	**0.8868**

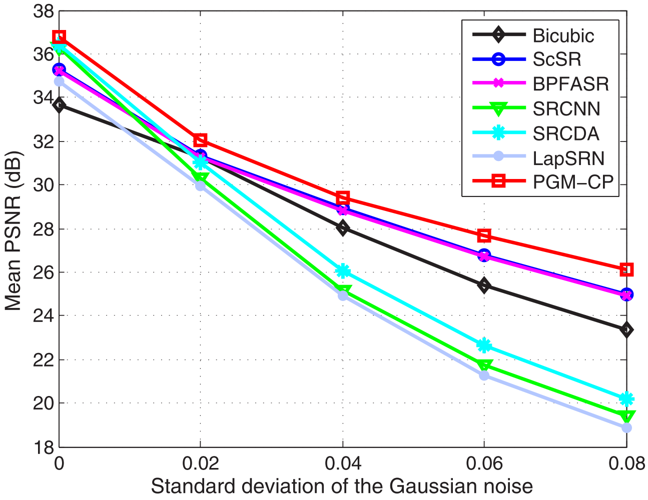


Fig. 11. Mean PSNR comparison on Set5 with scaling factor 2 in the presence of Gaussian noise.

LapSRN. Although the SRCDA is pre-trained as two coupled autoencoders, the model is further fine-tuned as a regression model that destroys the expression of the latent space to well reconstruct the LR images. In addition, the latent representations in SRCNN, SRCDA, and LapSRN are deterministic, which lack of randomness to model the noise. Thus, with noisy input for the forward mapping, the noise are aggravated and results in poor performance even worse than Bicubic interpolation. On the contrary, the coding-based methods ScSR and BPFASR can moderately eliminate the noise due to the sparse reconstruction of the LR images. As shown in Fig. 10, the PGM-CP has the ability to reconstruct an LR image and get rid of noise. Based on such clean input, the HR generative model reconstruct a clean HR image with shaper edges, which performs better than ScSR and BPFASR. As summarized in Fig. 11, as the distortion level increases, the performance differences between the coding-based methods and the regression-based methods become more apparent. More visual comparisons can be found in Appendix II.

6. Conclusion

In this paper, a general variational probabilistic generative framework parameterized by deep network is proposed for single image SR, which combines the strengths of coding-based methods and regression-based methods. We use our framework as a guidance to develop a new model called PGM-CP, with the help of an informative conditional prior and a consistent recognition model. The PGM-CP has shown superior SR performance than state-of-the-art methods, faster inference than previous coding-based methods, and more robust to noise than regression-based methods.

Acknowledgement

This work is partially supported by the Thousand Young Talent Program of China, NSFC (61771361, 61671354), the 111 Project (B18039) and the National Science Fund for Distinguished Young Scholars of China (61525105).

Appendix I

This section presents several visual comparison results with different scaling factors.

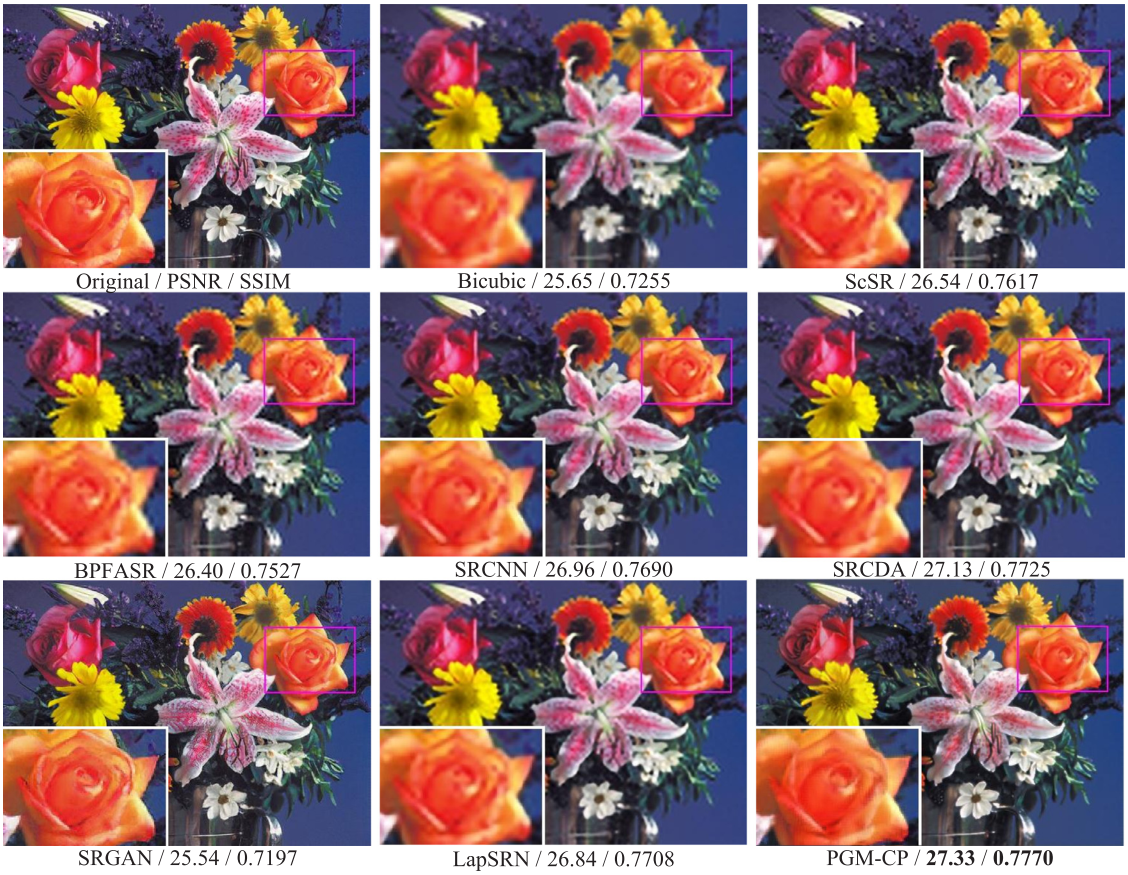


Fig. 12. The “flowers” image from Set14 with scaling factor 4.

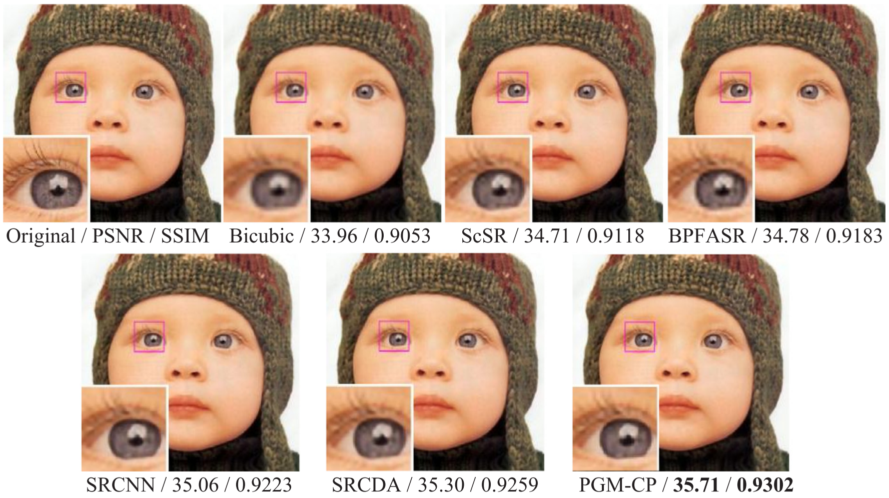
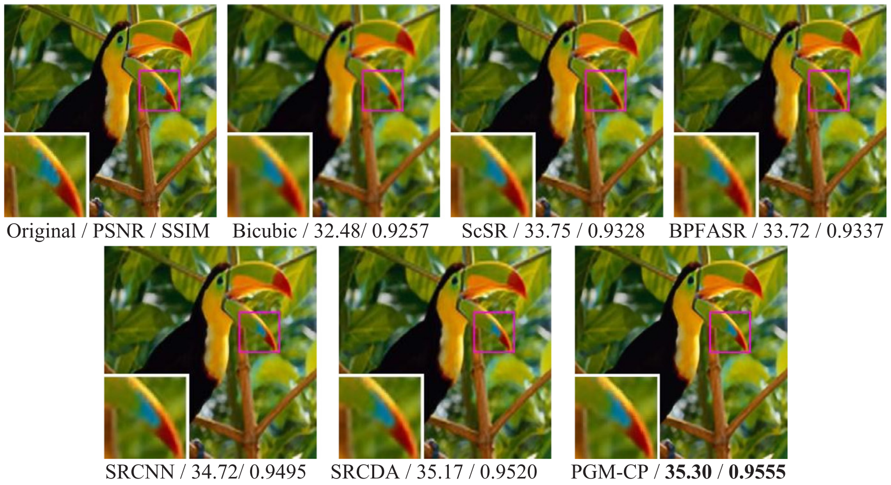


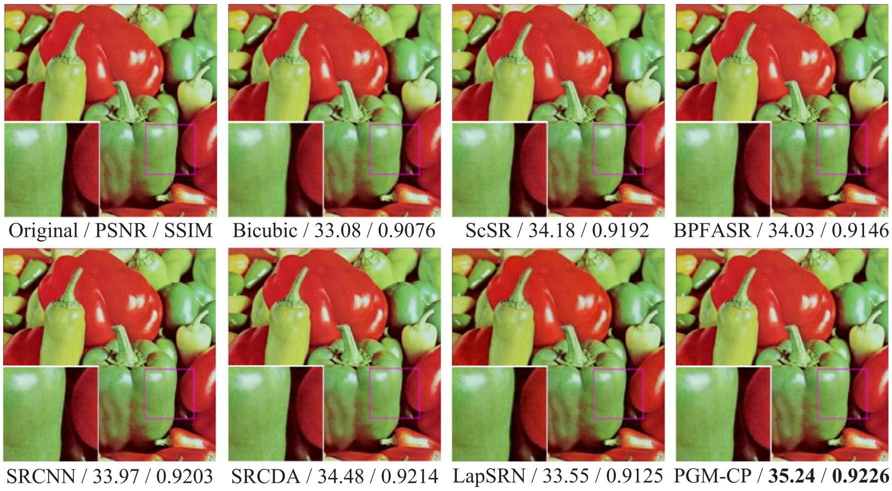
Fig. 13. The “baby” image from Set5 with scaling factor 3.



Original / PSNR / SSIM Bicubic / 32.48/ 0.9257 ScSR / 33.75 / 0.9328 BPFASR / 33.72 / 0.9337

SRCNN / 34.72/ 0.9495 SRCDA / 35.17 / 0.9520 PGM-CP / **35.30** / **0.9555**

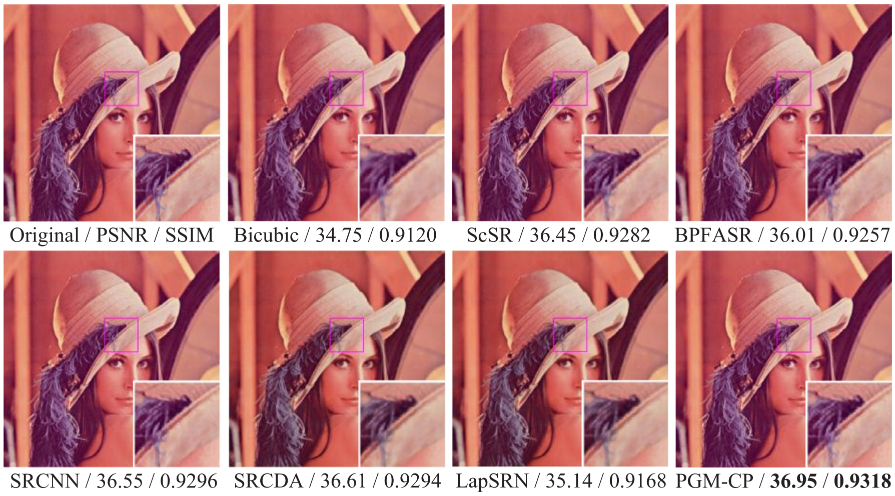
Fig. 14. The "bird" image from Set5 with scaling factor 3.



Original / PSNR / SSIM Bicubic / 33.08 / 0.9076 ScSR / 34.18 / 0.9192 BPFASR / 34.03 / 0.9146

SRCNN / 33.97 / 0.9203 SRCDA / 34.48 / 0.9214 LapSRN / 33.55 / 0.9125 PGM-CP / **35.24** / **0.9226**

Fig. 15. The "pepper" image from Set14 with scaling factor 2.



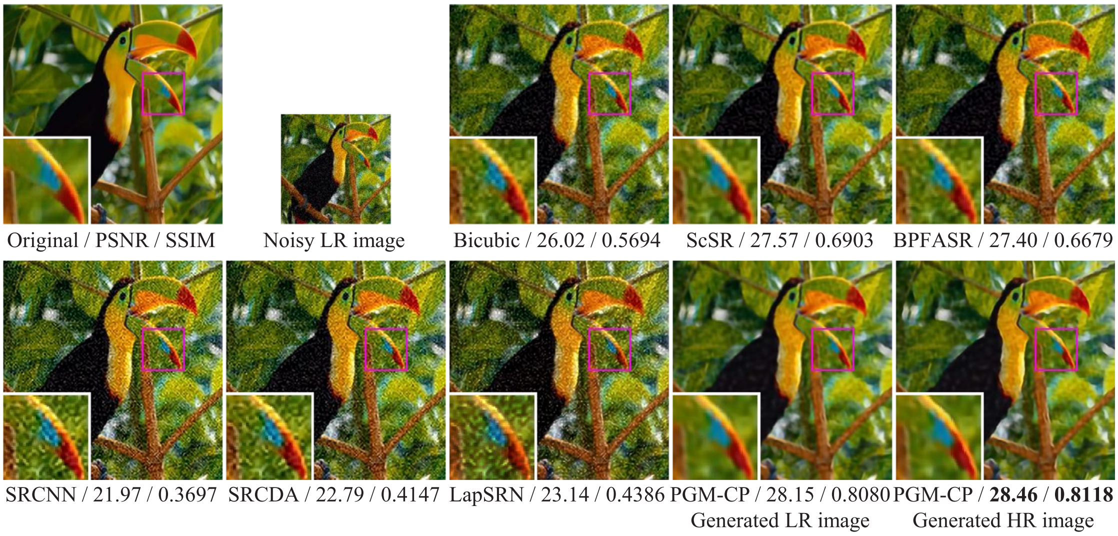
Original / PSNR / SSIM Bicubic / 34.75 / 0.9120 ScSR / 36.45 / 0.9282 BPFASR / 36.01 / 0.9257

SRCNN / 36.55 / 0.9296 SRCDA / 36.61 / 0.9294 LapSRN / 35.14 / 0.9168 PGM-CP / **36.95** / **0.9318**

Fig. 16. The "lenna" image from Set14 with scaling factor 2.

Appendix II

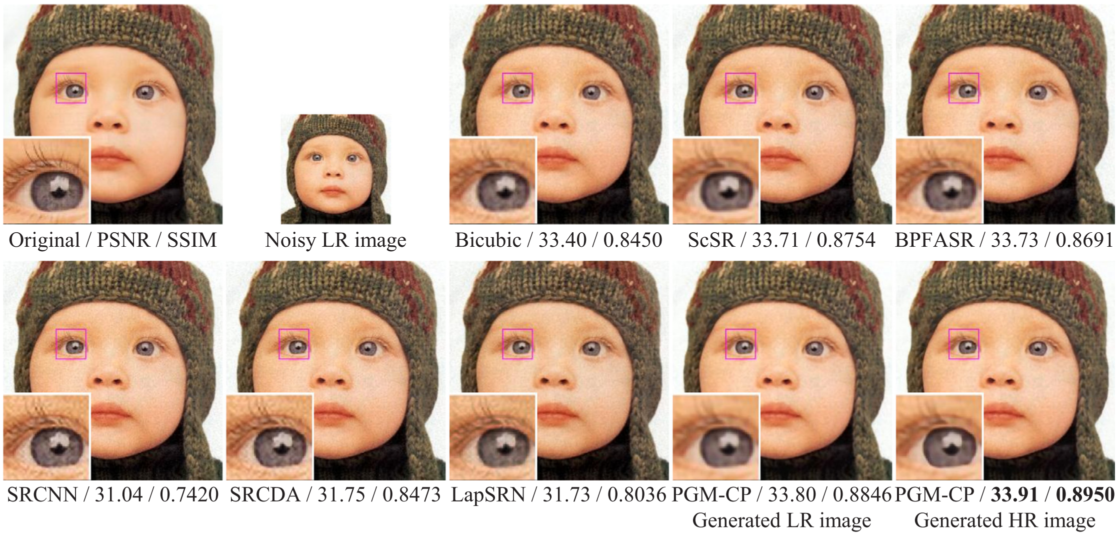
This section presents several visual comparison results with different noise level.



Original / PSNR / SSIM Noisy LR image Bicubic / 26.02 / 0.5694 ScSR / 27.57 / 0.6903 BPFASR / 27.40 / 0.6679

SRCNN / 21.97 / 0.3697 SRCDA / 22.79 / 0.4147 LapSRN / 23.14 / 0.4386 PGM-CP / 28.15 / 0.8080 Generated LR image PGM-CP / **28.46** / **0.8118** Generated HR image

Fig. 17. The noisy "bird" from Set5 with scaling factor 2. The standard deviation of the Gaussian noise is 0.06.



Original / PSNR / SSIM Noisy LR image Bicubic / 33.40 / 0.8450 ScSR / 33.71 / 0.8754 BPFASR / 33.73 / 0.8691

SRCNN / 31.04 / 0.7420 SRCDA / 31.75 / 0.8473 LapSRN / 31.73 / 0.8036 PGM-CP / 33.80 / 0.8846 Generated LR image PGM-CP / **33.91** / **0.8950** Generated HR image

Fig. 18. The noisy "baby" from Set5 with scaling factor 2. The standard deviation of the Gaussian noise is 0.02.

References

[1] S.C. Park, M.K. Park, M.G. Kang, Super-resolution image reconstruction: a technical overview, IEEE Signal Process. Mag. 20 (3) (2003) 21–36.

[2] K. Nasrollahi, T.B. Moeslund, Super-resolution: a comprehensive survey, Mach. Vis. Appl. 25 (6) (2014) 1423–1468.

[3] M.C. Hong, M.G. Kang, A.K. Katsaggelos, An iterative weighted regularized algorithm for improving the resolution of video sequences, in: Proceedings of the International Conference on Image Processing, 1997, 1997, p. 474.

[4] R.C. Hardie, K.J. Barnard, J.G. Bognar, E.E. Armstrong, E.A. Watson, High-resolution image reconstruction from a sequence of rotated and translated frames and its application to an infrared imaging system, Opt. Eng. 37 (1) (1998) 247–260.

[5] M.E. Tipping, C.M. Bishop, Bayesian image super-resolution, Adv. Neural Inf. Process. Syst. (2003) 1303–1310.

[6] L.C. Pickup, D.P. Capel, S.J.R.A. Zisserman, Bayesian image super-resolution, continued, in: Proceedings of the Conference on Advances in Neural Information Processing Systems, 2006, pp. 1089–1096.

[7] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution as sparse representation of raw image patches, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[8] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution via sparse representation, IEEE Trans. Image Process. 19 (11) (2010) 2861–2873.

[9] C.E. Duchon, Lanczos filtering in one and two dimensions, J. Appl. Meteorol. 18 (8) (1979) 1016–1022.

[10] X. Li, M.T. Orchard, New edge-directed interpolation, IEEE Trans. Image Process. 10 (10) (2001) 1521–1527.

[11] L. Zhang, X. Wu, An edge-guided image interpolation algorithm via directional filering and data fusion, IEEE Trans. Image Process. 15 (8) (2006) 2226–2238.

[12] J.D.V. Ouwerkerek, Image super-resolution survey, Image Vis. Comput. 24 (10) (2006) 1039–1052.

[13] G. Freedman, R. Fattal, Image and video upscaling from local self-examples, ACM Trans. Graph. 30 (2) (2011) 1–11.

[14] J. Yang, Z. Lin, S. Cohen, Fast image super-resolution based on in-place example regression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1059–1066.

[15] Z. Cui, H. Chang, S. Shan, B. Zhong, X. Chen, Deep network cascade for image super-resolution, in: Proceedings of the European Conference on Computer Vision, 2014, pp. 49–64.

[16] J. Huang, A. Singh, N. Ahuja, Single image super-resolution from transformed self-exemplars, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5197–5206.

[17] X. Gao, K. Zhang, D. Tao, X. Li, Image super-resolution with sparse neighbor embedding, IEEE Trans. Image Process. 21 (7) (2012) 3194–3205.

[18] H. Chang, D.-Y. Yeung, Y. Xiong, Super-resolution through neighbor embedding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2004, pp. 275–282.

[19] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[20] J. Yang, Z. Wang, Z. Lin, S. Cohen, T. Huang, Coupled dictionary training for image super-resolution, IEEE Trans. Image Process. 21 (11) (2012) 3467–3478.

[21] G. Polatkan, M. Zhou, L. Carin, D. Blei, I. Daubechies, A bayesian nonparametric approach to image super-resolution, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2) (2015) 346–358.

[22] M. Yin, J. Gao, S. Cai, Image super-resolution via 2d tensor regression learning, Comput. Vis. Image Underst. 132 (4) (2015) 12–23.

[23] K.I. Kim, Y. Kwon, Single-image super-resolution using sparse regression and natural image prior, IEEE Trans. Pattern Anal. Mach. Intell. 32 (6) (2010) 1127–1133.

[24] Y. Bengio, Learning deep architectures for AI, Found. Trends Mach. Learn. 2 (1) (2009) 1–127.

[25] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: International Conference on Neural Information Processing Systems, 2012, pp. 1097–1105.

[26] C. Hong, J. Yu, D. Tao, M. Wang, Image-based three-dimensional human pose recovery by multiview locality-sensitive sparse retrieval, IEEE Trans. Ind. Electron. 62 (6) (2015) 3742–3751.

[27] J. Zhang, J. Yu, D. Tao, Local deep-feature alignment for unsupervised dimension reduction, IEEE Trans. Image Process. 27 (5) (2018) 2420–2432.

[28] W. Huang, H. Ding, G. Chen, A novel deep multi-channel residual networks-based metric learning method for moving human localizationin video surveillance, Signal Process. 142 (2018) 104–113.

[29] N.M. Rad, S.M. Kia, C. Zarbo, T.V. Laarhoven, G. Jurman, P. Venuti, E. Marchiori, C. Furlanello, Deep learning for automatic stereotypical motor movement detection using wearable sensors in autism spectrum disorders, Signal Process. 144 (2018) 180–191.

[30] K. Zeng, J. Yu, R. Wang, C. Li, D. Tao, Coupled deep autoencoder for single image super-resolution., IEEE Trans. Cybern. 47 (1) (2017) 27–37.

[31] Z. Wang, D. Liu, J. Yang, W. Han, T. Huang, Deep networks for image super-resolution with sparse prior, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 370–378.

[32] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks., IEEE Trans. Pattern Anal. Mach. Intell. 38 (2) (2015) 295–307.

[33] J. Kim, K. Lee, K.M. Lee, Deeply-recursive convolutional network for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1637–1645.

[34] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4681–4690.

[35] K. Gregor, Y. LeCun, Learning fast approximations of sparse coding, in: International Conference on Machine Learning, 2010, pp. 399–406.

[36] J. Sun, Z. Xu, H.-Y. Shum, Image super-resolution using gradient profile prior, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[37] X. Li, J. She, Collaborative variational autoencoder for recommender systems, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 305–314.

[38] C. Li, J. Zhu, T. Shi, B. Zhang, Max-margin deep generative models, in: Proceedings of the International Conference on Neural Information Processing Systems, pp. 1837–1845.

[39] C. Dong, C.L. Chen, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, in: Proceedings of the European Conference on Computer Vision, 2014, pp. 184–199.

[40] D.P. Kingma, M. Welling, Auto-encoding variational bayes, in: Proceedings of the International Conference on Learning Representations, 2014.

[41] Y. Burda, R. Grosse, R. Salakhutdinov, Importance weighted autoencoders, in: Proceedings of the International Conference on Learning Representations, 2016.

[42] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: Proceedings of the International Conference on Curves and Surfaces, 2010, pp. 711–730.

[43] D.J. Mackay, Local minima, symmetry-breaking, and model pruning in variational free energy minimization, Inference Group, Cavendish Laboratory, Cambridge, UK, 2001.

[44] C.K. Sønderby, T. Raiko, L. Maaløe, S.K. Sønderby, O. Winther, How to train deep variational autoencoders and probabilistic ladder netwroks, in: Proceedings of the International Conference on Machine Learning, 2016.

[45] K. Sohn, X. Yan, H. Lee, Learning structured output representation using deep conditional generative models, in: Proceedings of the International Conference on Neural Information Processing Systems, 2015, pp. 3483–3491.

[46] G. Pandey, A. Dukkipati, Variational methods for conditional multimodal deep learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[47] C.M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer, pp. 241–249.

[48] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Proceedings of the International Conference for Learning Representations, 2015, pp. 1–13.

[49] T. Griffiths, Z. Ghahramani, Infinite latent feature models and the indian buffet process, in: Proceedings of the International Conference on Neural Information Processing Systems, 2006.

[50] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep laplacian pyramid networks for fast and accurate super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 624–632.

[51] M. Bevilacqua, A. Roumy, C. Guillemot, M.L.A. Morel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding, in: Proceedings of the British Machine Vision Conference, 2012.

[52] D.R. Martin, C. Fowlkes, D. Tal, J. Malik, A dataset of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings of the IEEE International Conference on Computer Vision, 2, 2001, pp. 416–423.

[53] K. Kavukcuoglu, P. Sermanet, Y.L. Boureau, K. Gregor, M. Mathieu, Y. Lecun, Learning convolutional feature hierarchies for visual recognition, in: Proceedings of the International Conference on Neural Information Processing Systems, 2010, pp. 1090–1098.

[54] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity., IEEE Trans. Image Process. 13 (4) (2004) 600–612.