Fast Adaptation to Super-Resolution Networks via Meta-Learning

Seobin Park[*1], Jinsu Yoo[*1], Donghyeon Cho[2], Jiwon Kim[3]
, and Tae Hyun Kim[**1]

[1] Hanyang University, Seoul, South Korea,
{seobinpark,jinsuyoo,taehyunkim}@hanyang.ac.kr
[2] Chungnam National University, Daejeon, South Korea, cdh12242@gmail.com
[3] SK T-Brain, Seoul, South Korea, jk@sktbrain.com

Abstract. Conventional supervised super-resolution (SR) approaches are trained with massive external SR datasets but fail to exploit desirable properties of the given test image. On the other hand, self-supervised SR approaches utilize the internal information within a test image but suffer from computational complexity in run-time. In this work, we observe the opportunity for further improvement of the performance of single-image super-resolution (SISR) without changing the architecture of conventional SR networks by practically exploiting additional information given from the input image. In the training stage, we train the network via meta-learning; thus, the network can quickly adapt to any input image at test time. Then, in the test stage, parameters of this meta-learned network are rapidly fine-tuned with only a few iterations by only using the given low-resolution image. The adaptation at the test time takes full advantage of patch-recurrence property observed in natural images. Our method effectively handles unknown SR kernels and can be applied to any existing model. We demonstrate that the proposed model-agnostic approach consistently improves the performance of conventional SR networks on various benchmark SR datasets.

Keywords: Deep learning, Meta-learning, Single-image super-resolution, Patch recurrence

1 Introduction

Super-resolution (SR) aims to increase the image size by recovering high-frequency details from a given low-resolution (LR) input image, and SR becomes a key feature in electrical goods, such as smartphone and TV; it has become popular as high-resolution (HR) screens are commonly available in our daily lives. The most basic methods utilize interpolation techniques (*e.g.,*nearest and bicubic resizing)

---

* Equal contribution. ** Corresponding author.

to fill in the missing pixels. These methods are efficient but produce blurry results. Moreover, dedicated hardware-equipped devices, such as jittering [2] and focal stack [23] cameras, allow the use of multiple images to solve the LR image problem. However, these specialized devices incur additional costs and cannot be used to restore images captured with conventional cameras in the past. To mitigate these problems, numerous single-image super-resolution (SISR) algorithms that restore high-quality images by using only a single LR image as input have been studied; in particular, optimization-based [9, 14] and learning-based [5, 21, 27, 18, 30, 33, 34] methods have been investigated intensively.

Since Dong *et al.* [5] demonstrated that a three-layered convolutional neural network could outperform the traditional optimization-based methods by a large margin, researchers have proposed numerous deep-learning-based methods for SISR. These methods aim to increase the performance of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) by allowing deeper networks to maximize the power of deep learning with large training datasets. In recent years, however, PSNR values have reached a certain limit, and more studies using perception metric [20, 22] have been introduced to focus on creating visually pleasing and human-friendly images.

Most of the current deep-supervised-learning approaches do not explicitly adapt their models during test time. Instead, fixed network parameters are used for all test images regardless of what we can learn more from the new test image. To fully utilize the additional information available from the given input test image (LR), we propose to extend this single fixed model approach by combining it with a dynamic parameter adaptation scheme. We find that the adaptive network results in better performance, especially for unseen type of images. In particular, we can utilize patch-recurrence property if available in the input image, which can be described as self-supervised learning. The notion of exploiting patch-recurrence has been introduced in prior works [9, 40]. Recently, Shocher *et al.* [30] proposed a zero-shot SR (ZSSR) method employing deep learning; this study is the most related work to our proposed method. ZSSR trains a relatively small convolutional neural network at test time from scratch, with training samples extracted only from the given input image itself. Therefore, ZSSR can naturally exploit the internal information of the input image. However, ZSSR has some limitations: (1) requirement of considerable inference time due to slow self-training step; (2) failure to take full advantage of using pre-trained networks learned by large amounts of external dataset;

Meta-learning can be a breakthrough for the above-mentioned problem. Meta-learning, *i.e.,* learning to learn, is gaining popularity in recent deep-learning studies [35, 31]. Meta-learning aims to learn quickly and efficiently from a small set of data available at test time. Several methods, such as recurrent architecture-based [10], and gradient-based methods [13, 26], have been proposed. In particular, model agnostic meta-learning (MAML) [7] is an example of a gradient-based method. We experimentally find that training a network with MAML results in the best initialization of the network parameter to perform well when fine-tuning with a small number of given input data. Consequently, as shown in Table 1, the

	External dataset?	Internal dataset?	Fast at run-time?
Self-supervision [30]	✗	✓	✗
Supervision [5, 27, 18]	✓	✗	✓
MLSR(ours)	✓	✓	✓

Table 1: Conventional supervised SR methods are trained with external SR datasets and run fast. Whereas, self-supervised SR methods typically exploit information using the given test image at run-time, which is time-consuming and impractical. Meta-learning for SR (MLSR) can efficiently utilize both external and internal information and take advantages of each approach.

proposed method can efficiently utilize both external and internal information and take advantages of each approach.

To this end, we introduce a method employing the meta-learning (fast adaptation) algorithm to solve the SISR problem. Using a large number of degraded images generated with various SR kernels, our SR networks are trained not only to generalize over large external data but also to adapt fast to any input image with real SR kernels.

Our contributions can be summarized as follows:

- To our knowledge, fully exploiting supervision signals available from both external and internal data with an effective meta-learning method is successful for the first time.
- Most state-of-the-art SR networks can be improved with our meta-learning scheme without changing the predefined network architectures.
- Our method achieves the state-of-the-art performance over benchmark datasets.

2   Related Works

In this section, we review the most relevant SISR works. Also, methods for handling unknown SR kernel (*i.e.,*blind SR) are briefly described.

An early example-based SISR method [8] learned the complicated relationships between LR and HR patches by learning how to use the external dataset. A locally linear embedding-based SISR method was introduced by Chang *et al.* [4]. Yang *et al.* [36] proposed a sparse coding-based algorithm assuming that a pair of LR and HR patches shares the same sparse coefficients with each distinct dictionary. Also, learning methodologies like random forest [17, 29, 28], hierarchical decision tree [16], and deep learning [5, 21, 27, 22, 18] have been proposed to boost the performance of SISR.

The self-similarity-based methods assume that a natural image contains repetitive patterns and structures within and across different image scales. Glasner *et al.* [9] proposed a unified framework that incorporates self-similarity-based approaches by exploiting patch-recurrence within and across different scales of a given LR input image. Huang *et al.* [14] handled transformed patches by estimating the transformations between the corresponding LR–HR patch pairs.

Dong *et al.* [6] proposed the non-local centralized sparse representation to exploit the non-local self-similarity of a given LR image. Huang *et al.* [15] combined the benefits from both external and internal databases for SISR by employing a hierarchical random forest.

Recently, a study dealing with the unknown SR kernel has begun to draw attention. Michaeli and Irani [25] exploit the nature of recurrence of small patches to handle unknown SR kernel. Yuan *et al.* [37] propose an unsupervised learning method with CycleGAN [39], and Gu *et al.* [11] estimate unknown SR kernel iteratively and additionally add a spatial feature transform (SFT) layers into the SR network for handling multiple blur kernels. Based on a simple convolutional neural network, ZSSR [30] deals with SR kernels given at test time by exploiting information from an input image itself.

In this work, we focus on overcoming the limitations of these conventional SISR methods. We observe that many existing deep learning-based methods fail to fully utilize the information provided in a given input image. Although ZSSR [30] utilizes both the power of deep learning and information from the input image at test time, it does not use pre-trained networks with large external dataset. Therefore, we start from training network to utilize the external examples. Then, we fine-tune the network with the input image to utilize the information captured by internal patch-recurrence and cover unknown SR kernels (given at test time) for the input. To obtain a well-trained network that can quickly adapt to the input image by using patch-recurrence, we integrate a meta-learning technique inspired by MAML [7] with conventional SISR networks without changing the network architectures. MAML aims to learn a meta-network that can swiftly adapt to new learning task and examples using a small number of iterations for fine-tuning. MAML is applicable in a variety of tasks, such as few-shot learning [19, 32] and reinforcement learning [12]. We apply the MAML method to fine-tune the pre-trained SR parameters to input images quickly and efficiently. We experimentally verify that our approach can boost the performance of state-of-the-art SISR by a large margin.

3   Meta-Learning for Super-Resolution

In this section, we introduce a new neural approach that integrates recent meta-learning techniques to solve the SISR problem by exploiting additional information available in the input LR image.

3.1   Exploiting patch-recurrence for deep SR

According to [9, 14, 30], small patches inside the natural image recur multiple times across the different scales of a given image. Therefore, unlike conventional learning-based methods that utilize large external datasets, we can find multiple HR patches corresponding to a given LR patch within a single-input image using the patch-recurrence. However, these previous methods have been developed separately and handle external and internal datasets differently. Thus, we develop a

Fig. 1: Increasing PSNR values of ENET [27], IDN [18], and RCAN [38] with fine-tuning process during the testing phase.

new method that facilitates SR networks by utilizing both (large) external and (small) internal datasets to further enhance the quality of the restored images.

First, we conduct a simple experiment to improve the performance of existing deep SR networks without changing their network architectures by using the patch-recurrence from a given LR test image. To achieve such goal, we retrain (fine-tune) the fully trained SR networks, such as ENET [27], IDN [18], and RCAN [38], with a new training set consisting of LR test image and its down-scaled version ($\times 0.5$). Note that, RCAN is currently state-of-the-art SR network. By updating the network parameters using the gradient descent, the PSNR values of SR networks increase. Also note that we can further increase PSNR values without using the ground truth HR image, because we utilize additional information obtained from the patch-recurrence of the new training set (Figure 1). The PSNR values in Figure 1 are obtained by calculating the average of the updated PSNR values on the Urban100 dataset [14]. The PSNR values tend to increase until 50 iterations, then decrease because the networks can be over-fitted with a small training set at test time. The performance of RCAN drops relatively quickly due to huge number of parameters used in the network.

This experiment demonstrates that there is still room to improve the performance of conventional SR networks while keeping their original network architectures, and patch-recurrence property is a key to boost the performance by adapting parameters of the fully pre-trained networks.

3.2   Handling unknown SR kernel for deep SR

SR in unknown degradation settings (*i.e.,*unknown SR kernel) is more challenging than conventional SR problem in ideal setting using the bicubic interpolation. According to [25], the performance of the conventional SR networks trained with only bicubic kernel deteriorates significantly when it comes to the non-bicubic and real SR kernels in real scenario [30]. That is, generalization capability of the networks which can handle newly seen SR kernel during test phase is restricted in real situation. However, many conventional SR networks still assume ideal and fixed bicubic SR kernel, and thus cannot handle real non-bicubic SR kernels.

In this section, we perform a simple experiment to see whether this problem can be also alleviated with patch-recurrence property. We first degrade input LR image with a non-bicubic SR kernel to generate a new training set consisting of LR image and its down-sized image ($\times 0.5$), and then evaluate the performance

Model	Set5	Set14	BSD100	Urban100
IDN [18]	28.24	26.29	26.28	23.49
IDN-Finetune	**31.52**	**28.91**	**28.47**	**25.93**

Table 2: Average PSNR for scale factor ×2 dealing with unknown SR kernel.

of the original IDN and its fine-tuned version with the down-sized training set. Note that the original IDN is initially trained with only bicubic SR kernel on the DIV2K [1] dataset, and our IDN with fine-tuning (IDN-Finetune) is further optimized for 1000 iterations with the gradient update. In Table 2, we observe that IDN-Finetune can handle the images degraded by non-bicubic SR kernel much better on numerous benchmark datasets compared to the original IDN trained with bicubic SR kernel. Thus, we can see that the patch-recurrence property still holds and can be also used to improve SR performance by handling unknown SR kernels.

3.3   Proposed Method

In previous sections, we have shown that the patch-recurrence property can be used not only to improve the performance of SR networks but also to deal with non-bicubic SR kernels. However, to update and adapt the pre-trained network parameters at test time to the specific input image, naive fine-tune-based update with stochastic gradient descent (SGD) requires large number of iterations and takes much time. To solve this problem, we integrate a meta-learning technique [7] with the SR networks to facilitate use of the patch-recurrence and boost the speed of the adaptation procedure at test time.

First, we define each task to employ meta-learning as super-resolving a single specific LR image by utilizing internal information available within the given LR input image. However, unlike conventional few-shot/k-shot problems which can be solved by meta-learning, our new SR task does not provide the ground-truth data (HR image) corresponding to the LR input image for adaptation at test time. Thus, it is difficult to directly apply the conventional meta-learning algorithms to our new learning task for SR.

Therefore, we develop a Meta-Learning for SR (MLSR) algorithm based on our observation that a pair of images composed of LR input and its down-scaled version (LR↓) can be used as a new training sample for our new SR task due to the patch-recurrence property of the natural image, which learns to adapt the pre-trained SR networks to the given test image. To be specific, we employ the recent model-agnostic meta-learning (MAML) approach. In particular, MAML allows fast adaptation to a new task with only a small number of gradient updates [7], so we can boost the speed of our test-time learning task which originally requires large number of gradient update steps without meta-learning scheme (*i.e.,* naive fine-tune).

In Figure 2, the overall flow of the proposed method is illustrated. First, we initialize the conventional SR networks with large external train datasets.

(a) Network initialization   **(b) Meta-learning**   **(c) Test**

Fig. 2: Overall flow of the proposed method (MLSR). (a) Initialization stage of MLSR. Conventional SR network is trained with large external dataset. (b) Meta-learning stage of MLSR. The SR network is meta-trained to allow fast adaptation to any input image at test time. (c) Test stage of MLSR. Meta-learned parameters are rapidly tuned to the given LR image.

Next, we start meta-learning using MAML which optimizes the initialized SR parameters to enable quick adaptation to the new LR test image. Finally, during the test phase, we adapt the meta-learned parameters with the given LR test image, and restore the HR image by using the adapted parameters.

Specifically, we formulate the proposed method more concretely. Our SR model f_θ which is initialized with parameter θ renders an HR image from a given LR image by minimizing the loss $\mathcal{L}$, and it yields,

$$\mathcal{L}(f_\theta(LR), HR) = ||f_\theta(LR) - HR||_2^2, \tag{1}$$

and our goal of meta-learning is to optimize the network parameter θ to be quickly adapted to θ_i at test time with the given input image LR_i and its down-

---

Algorithm 1: MLSR training algorithm

Require: $p(I)$: Distribution (*e.g.,* uniform) over images
Require: α, β: Hyper-parameters (step-size)

1 Initialize θ
2 **while** *not converged* **do**
3   Sample a batch of images $\{I_i\} \sim p(I)$
4   Generate $\{HR_i\}$, $\{LR_i\}$, $\{LR_i \downarrow\}$ from $\{I_i\}$
5   **foreach** i **do**
6   Evaluate $\nabla_\theta \mathcal{L}(f_\theta(LR_i \downarrow), LR_i)$ using $\mathcal{L}$
7   Compute adapted parameters with SGD:
        $\theta_i \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(f_\theta(LR_i \downarrow), LR_i)$
8   Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_i \mathcal{L}(f_{\theta_i}(LR_i), HR_i)$

Algorithm 2: MLSR inference algorithm

Require: I: Given image
Require: α: Hyper-parameter (step-size)
Require: n: Number of gradient updates

1 Initialize θ with meta-trained parameter
2 Generate LR, $LR \downarrow$ from I
3 $i \leftarrow 0$
4 **while** $i < n$ **do**
5 $\quad$ Compute adapted parameters with SGD: $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(f_\theta(LR \downarrow), LR)$
6 $\quad$ $i \leftarrow i + 1$
7 Compute $f_\theta(LR)$

---

scaled image $LR_i \downarrow$. Therefore, the adaptation formulation with gradient update is given as follows:

$$\theta_i = \theta - \alpha \nabla_\theta \mathcal{L}(f_\theta(LR_i \downarrow), LR_i), \tag{2}$$

where hyper-parameter α controls the learning rate of the inner update procedure. Notably, to generate the down-scaled image $LR \downarrow$ we can use any SR kernel if available. Then, we optimize the following objective function $w.r.t.$ θ:

$$\underset{\theta}{\operatorname{argmin}} \sum_i \mathcal{L}(f_{\theta_i}(LR_i), HR_i), \tag{3}$$

and the optimization is preformed by the gradient update as:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_i \mathcal{L}(f_{\theta_i}(LR_i), HR_i). \tag{4}$$

In general, we can use multiple iterations for the adaptation in (2), but it increases computational cost in calculation of high-order derivatives in (4). To alleviate this problem, we can simply employ the first-order approximation methods [7, 26], which is known to give competitive results with lower computational cost. In our experiments, we use the first-order MAML introduced in [7].

At test time, we first adapt the parameters (θ) of the meta-learned SR network with the input LR image (LR) and its down-sized image ($LR \downarrow$), then restore the HR image using the adapted SR parameters as elaborated in Algorithm 2.

4 Experimental Results

In this section, we perform extensive experiments to demonstrate the superiority of the proposed method, and show quantitative and qualitative comparison results. Our source code is publicly available.[1]

---

[1] https://github.com/parkseobin/MLSR

Model	Iteration	DIV2K		BSD100		Urban100	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
SRCNN [5]	-	34.11	0.9272	31.13	0.8852	29.39	0.8927
	5	34.14	0.9274	31.15	0.8855	29.42	0.8931
+ **MLSR (ours)**	20	34.18	0.9276	31.19	0.8857	29.48	0.8936
	100	**34.23**	**0.9281**	**31.22**	**0.8860**	**29.54**	**0.8945**
ENET [27]	-	34.59	0.9329	31.64	0.8935	30.38	0.9097
	5	34.62	0.9331	**31.69**	**0.8936**	30.46	0.9105
+ **MLSR (ours)**	20	34.64	0.9333	**31.69**	**0.8936**	30.49	0.9108
	100	**34.67**	**0.9335**	31.67	0.8934	**30.52**	**0.9112**
IDN [18]	-	35.24	0.9403	32.11	0.8994	31.95	0.9269
	5	35.36	0.9408	**32.17**	**0.8996**	32.06	0.9275
+ **MLSR (ours)**	20	35.38	0.9409	**32.17**	**0.8996**	32.17	0.9280
	100	**35.40**	**0.9413**	32.08	0.8988	**32.23**	**0.9286**
RCAN [38]	-	35.69	0.9451	32.38	**0.9023**	33.10	0.9369
	5	35.72	0.9454	**32.39**	**0.9023**	33.27	0.9373
+ **MLSR (ours)**	20	**35.75**	**0.9458**	32.37	0.9022	**33.32**	**0.9379**
	100	35.48	0.9444	32.04	0.8982	33.26	0.9373

Table 3: PSNR and SSIM results from different SR networks on different test dataset with scale ×2. Bicubic SR kernel is used. Baseline SR networks are SRCNN [5], ENET [27], IDN [18], and RCAN [38], and + **MLSR** indicates the meta-learned version of the baseline network.

4.1   Implementation details

For our experiments, we first pre-train conventional SR networks (SRCNN [5], ENET [27], IDN [18], and RCAN [38]) with DIV2K [1] dataset. We use publicly available pre-trained parameters for IDN and RCAN (TensorFlow versions), and use our own parameters trained from scratch for SRCNN and ENET. Next, we start meta-learning for these baseline SR networks in accordance with iterative steps in Algorithm 1. For meta-learning, we still use DIV2K dataset, and use 5 inner gradient update steps in (2) (line 7 in Algorithm 1). We set $\alpha = 10^{-5}$, $\beta = 10^{-6}$, train patch size to 512×512, and mini-batch size to 16.

4.2   MLSR with fixed bicubic SR kernel

First, we assume fixed bicubic SR kernel and compare PSNR and SSIM values of our SR networks on Urban100 and BSD100 [24] datasets. For the comparison on DIV2K, test set of DIV2K is used since our networks are trained with DIV2K train set. Results are shown in Table 3, and we can see that PSNR and SSIM values of SR networks with meta-learning are higher than the original ones. Notably, the performance gaps on Urban100 are significantly larger than on

Dataset	Model	Before/after adaptation
DIV2K	IDN [18]	32.19 / -
	IDN-ML	32.19 / **32.37**
Urban100*	IDN [18]	32.28 / -
	IDN-ML	32.13 / **32.43**

Table 4: PSNR results of IDN [18] and IDN-ML trained on different datasets. 40 images in Urban100 dataset are selected for the evaluation. * Another 50 images in the dataset are used for training.

Fig. 3: Performance changes of IDN-Finetune, and IDN-ML during adaptation on DIV2K.

other datasets, as urban scenes in the dataset mainly include structured scene with lots of patch-redundancy [14].

To further explore the patch-recurrence of natural images, we train the networks with the meta-learning scheme in Algorithm 1 on the Urban100 dataset which includes a large number of similar patches. For meta-learning with Urban100, we use 50 images for training, 10 images for validation, and the remaining 40 images for test. In Table 4, we evaluate differently trained IDNs, and IDN trained on Urban100 with meta-learning algorithm outperforms other models. Note that PSNR value of meta-learned IDN (IDN-ML) is relatively low before the adaptation, but improves dramatically with only 5 gradient updates (0.3dB gain). This proves that our MLSR method can learn better on images with rich patch-recurrence in urban scenes. More qualitative comparison results are shown in Figure 4, and the test images are particularly well restored with our network trained with meta-learning algorithm since specific patterns are repeated over the image itself. Moreover, we can see that the adapted parameters with more gradient update steps render visually much better results.

Moreover, in Figure 3, we show how PSNR value changes when the number of gradient steps in (2) increases during meta-learning and test phases. As shown, our meta-learned model (IDN-ML) can quickly adapt SR parameters at test time, and achieves competitive results with only few gradient updates. Indeed, only 5 gradient updates can produce results which can be obtainable with ∼15 iterations of IDN with naive fine-tuning (IDN-Finetune).

4.3   MLSR with unseen SR kernel

In this section, we further conduct experiments to see the capability of the proposed MLSR algorithm in dealing with new and unseen SR kernel during the test phase. We carry out meta-learning in Algorithm 1 with randomly generated 5×5 SR kernels on the DIV2K dataset and train for 30k iterations. Moreover, We generate 40k 5×5 SR kernels as in [3], and use 38k kernels for training, 1k for validation, and 1k for test.

IDN [18] IDN-ML IDN-ML IDN-ML   GT   IDN [18] IDN-ML IDN-ML IDN-ML   GT
(0)   (5)   (20)   (0)   (5)   (20)

Fig. 4: Qualitative comparison with differently trained IDN [18]. Number inside the bracket indicates the number of gradient update steps in run-time.

Model	Set5	Set14	BSD100	Urban100
ZSSR [30]	29.68	27.76	27.53	25.02
IDN-ML / 0	28.10	26.22	26.19	23.48
IDN-ML / 5	29.17	27.08	26.82	24.36
IDN-ML / 20	29.86	27.67	27.32	24.96
IDN-ML / 100	**30.41**	**28.12**	**27.75**	**25.42**

Table 5: Comparing ZSSR [30] and meta-trained IDN on non-bicubic SR kernel. Right side of the slash indicates the number of gradient update steps in run-time.

In Table 2, unlike fine-tuning with bicubic SR kernel, we need a large number of iterations ($\sim$1000) to achieve the highest PSNR value in dealing with non-bicubic SR kernel. However, our IDN-ML trained with many different SR kernels learnt the way to be quickly adapted to the new SR kernel given at test time, and it shows competitive results with only few gradient updates using the new kernel. Notably, we assume that the SR kernel is given or can be estimated with conventional methods as in [25, 30]. In Figure 5, we can see that results with only 5 gradient updates (IDN-ML) are similar to the results from 350 iterations using the naive fine-tune without our meta-learning (IDN-Finetune).

After meta-training, we compare our model on Set5, Set14, BSD100 and Urban100 datasets. In the inference stage, an SR kernel that has not been shown during training stage, and an LR image degraded with that SR kernel are provided. The results in Figure 5 show consistent improvements for various datasets as the number of gradient update steps increases. Specifically, the performances raise strikingly ($\sim$1dB) at around 5 iterations, and it verifies that the network can quickly adapt to the given input image and SR kernel at test time with the small number of updates. Notably, the result on Urban100 is slightly different from others. Performance of the adapted network on Urban100 improves more

Fig. 5: Performance curve of PSNR values on various test datasets. Random 5×5 SR kernels are used.

Fig. 6: PSNR of IDN [18] with scale ×3, ×4 on Urban100 dataset. Right and left sides of the y-axis indicate the PSNR values with respect to the upscaling factor ×3 and ×4, respectively.

rapidly than adapted networks on other datasets as rich patch-recurrence with urban scenes helps to handle newly seen SR kernels at test time.

In Table 5, we also compare ours with ZSSR [30] on numerous dataset with SR kernels used in ZSSR, and our proposed method with 20 gradient updates shows competitive results compared to ZSSR, and significantly outperforms ZSSR when adapted for 100 iterations. Notably, ours with 100 iterations takes only 6 minutes to restore 100 urban images, but ZSSR requires more than 3 hours with GeForce RTX 2080Ti.

In Figure 7 and Figure 8, we compare visual results by naive fine-tuning (IDN−Finetune), meta-learning (IDN−ML) and ZSSR. We see that the quality improves significantly within few iterations with our MLSR algorithm, and the boundaries are restored gradually as iteration goes. Moreover, artifacts near boundaries caused by ZSSR are not produced by our proposed method.

4.4   SR with large scaling factor

Finally, we study the validity of the patch-recurrence property with large SR scaling factor. Unfortunately, as shown in Figure 6, exploiting patch-recurrence on big scale factor is hard (*i.e.,*×3 or ×4). Maximal performance gained by fine-tuning with large SR factors are around 0.04dB which are negligible. Therefore, to produce large images with large scaling factors, we can employ multi-scale (coarse-to-fine) approaches embedded into the conventional SR methods with small scale factor (*e.g.,*×1.25) which also exploit self-similarity nature of the given test images [9, 14, 30].

5   Conclusion

In this work, we introduced a new SR method which utilizes both the power of deep learning with large external dataset and additional information avail-

(a) Ground truth	(b) IDN	(c) IDN - Finetune
(PSNR, Gradient update steps)	(22.80, -)	(24.69, 100)
(d) ZSSR	(e) IDN - ML	(f) IDN - ML
(21.64, -)	(23.50, 0)	(25.32, 100)

Fig. 7: The "butterfly" image from Set5 dataset with upscaling factor ×2. Input LR image is downscaled with a non-bicubic 5×5 SR kernel.

able from the input image at test time. To this end, we proposed a novel Meta-Learning for SR (MLSR) algorithm which enables quick adaptation of SR parameters using only input LR image during the test phase. MLSR can be combined with conventional SR networks without any architecture changes, and can utilize the patch-recurrence property of the natural image, which can further boost PSNR performance of various deep learning-based methods. In addition, MLSR can handle non-bicubic SR kernel that exists in real world because meta-learned networks can be adapted to the specific input image. In experiments, we show that our MLSR can greatly boost up the performance of existing SR networks, with a few gradient update steps. Moreover, we experimentally demonstrated that MLSR takes advantage of the patch-recurrence well, by showing the performance improvements on the Urban100 dataset, where patch-recurrence occurs frequently. Finally, the proposed MLSR algorithm was also validated with the unseen non-bicubic SR kernel and showed that MLSR required less gradient updates than naive fine-tuning. We believe that the proposed method can be applied not only to SR but also to various types of reconstruction and low-level vision tasks.

Fig. 8: The "102061" image from BSD100 dataset with upscaling factor ×2. LR image is generated using a non-bicubic kernel. Our method achieves better performance than naive fine-tuning with the same number of inner updates at run-time.

Acknowledgement

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017)
2. Ben-Ezra, M., Zomet, A., Nayar, S.: Jitter camera: high resolution video from a low resolution detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2004)
3. Chakrabarti, A.: A neural approach to blind motion deblurring. In: Proceedings of the European Conference on Computer Vision (ECCV) (2016)

4. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. vol. 1, pp. I–I. IEEE (2004)

5. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer (2014)

6. Dong, W., Zhang, L., Shi, G., Li, X.: Nonlocally centralized sparse representation for image restoration. IEEE Transactions on Image Processing **22**, 1620–1630 (2013)

7. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning (ICML) (2017)

8. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. IEEE Computer graphics and Applications (2), 56–65 (2002)

9. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2009)

10. Graves, A., Wayne, G., Danihelka, I.: Neural turing machines. arXiv preprint arXiv:1410.5401 (2014)

11. Gu, J., Lu, H., Zuo, W., Dong, C.: Blind super-resolution with iterative kernel correction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

12. Gupta, A., Mendonca, R., Liu, Y., Abbeel, P., Levine, S.: Meta-reinforcement learning of structured exploration strategies. In: Advances in Neural Information Processing Systems (NIPS) (2018)

13. Hochreiter, S., Younger, A.S., Conwell, P.R.: Learning to learn using gradient descent. In: International Conference on Artificial Neural Networks. pp. 87–94. Springer (2001)

14. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)

15. Huang, J.J., Liu, T., Luigi Dragotti, P., Stathaki, T.: Srhrf+: Self-example enhanced single image super-resolution using hierarchical random forests. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 71–79 (2017)

16. Huang, J.J., Siu, W.C.: Learning hierarchical decision trees for single-image super-resolution. IEEE Transactions on Circuits and Systems for Video Technology **27**, 937–950 (2017)

17. Huang, J.J., Siu, W.C., Liu, T.R.: Fast image interpolation via random forests. IEEE Transactions on Image Processing **24**(10), 3232–3245 (2015)

18. Hui, Z., Wang, X., Gao, X.: Fast and accurate single image super-resolution via information distillation network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

19. Jamal, M.A., Qi, G.J.: Task agnostic meta-learning for few-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11719–11727 (2019)

20. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Proceedings of the European Conference on Computer Vision (ECCV) (2016)

21. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)

22. Ledig, C., Theis, L., Huszar, F., Caballero, J., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
23. Lee, M., Tai, Y.W.: Robust all-in-focus super-resolution for focal stack photography. IEEE Transactions on Image Processing **25**, 1887–1897 (2016)
24. Martin, D., Fowlkes, C., Tal, D., Malik, J., et al.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Iccv Vancouver: (2001)
25. Michaeli, T., Irani, M.: Nonparametric blind super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2013)
26. Nichol, A., Achiam, J., Schulman, J.: On first-order meta-learning algorithms. arXiv preprint arXiv:1803.02999 (2018)
27. Sajjadi, M.S., Scholkopf, B., Hirsch, M.: Enhancenet: Single image super-resolution through automated texture synthesis. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017)
28. Salvador, J., Pérez-Pellitero, E.: Naive bayes super-resolution forest. Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2015)
29. Schulter, S., Leistner, C., Bischof, H.: Fast and accurate image upscaling with super-resolution forests. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3791–3799 (2015)
30. Shocher, A., Cohen, N., Irani, M.: "zero-shot" super-resolution using deep internal learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
31. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems (NIPS) (2017)
32. Sun, Q., Liu, Y., Chua, T.S., Schiele, B.: Meta-transfer learning for few-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 403–412 (2019)
33. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3147–3155 (2017)
34. Tai, Y., Yang, J., Liu, X., Xu, C.: Memnet: A persistent memory network for image restoration. In: Proceedings of the IEEE international conference on computer vision. pp. 4539–4547 (2017)
35. Thrun, S., Pratt, L.: Learning to learn: Introduction and overview. In: Learning to learn, pp. 3–17. Springer (1998)
36. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE Transactions on Image Processing **19**(11), 2861–2873 (2010)
37. Yuan, Y., Liu, S., Zhang, J., Zhang, Y., Dong, C., Lin, L.: Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) pp. 814–81409 (2018)
38. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 286–301 (2018)
39. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
40. Zontak, M., Irani, M.: Internal statistics of a single natural image. In: CVPR 2011. pp. 977–984. IEEE (2011)