Recovering Realistic Texture in Image Super-resolution by Deep Spatial Feature Transform

Xintao Wang [1] Ke Yu[1] Chao Dong[2] Chen Change Loy[1]
[1]CUHK - SenseTime Joint Lab, The Chinese University of Hong Kong, [2]SenseTime Research
{wx016, yk017, ccloy}@ie.cuhk.edu.hk dongchao@sensetime.com

Abstract

Despite that convolutional neural networks (CNN) have recently demonstrated high-quality reconstruction for single-image super-resolution (SR), recovering natural and realistic texture remains a challenging problem. In this paper, we show that it is possible to recover textures faithful to semantic classes. In particular, we only need to modulate features of a few intermediate layers in a single network conditioned on semantic segmentation probability maps. This is made possible through a novel Spatial Feature Transform (SFT) layer that generates affine transformation parameters for spatial-wise feature modulation. SFT layers can be trained end-to-end together with the SR network using the same loss function. During testing, it accepts an input image of arbitrary size and generates a high-resolution image with just a single forward pass conditioned on the categorical priors. Our final results show that an SR network equipped with SFT can generate more realistic and visually pleasing textures in comparison to state-of-the-art SRGAN [27] and EnhanceNet [38].

1. Introduction

Single image super-resolution aims at recovering a high-resolution (HR) image from a single low-resolution (LR) one. The problem is ill-posed since a multiplicity of solutions exist for any given low-resolution pixel. To overcome this problem, contemporary methods such as those based on deep convolutional neural networks [7, 8, 9, 22, 26, 27, 23, 43, 44] constrain the solution space through learning mapping functions from external low- and high-resolution exemplar pairs. To push the solution closer to the natural manifold, new losses are proposed to replace the conventional pixel-wise mean squared error (MSE) loss [7] that tends to encourage blurry and overly-smoothed results. Specifically, perceptual loss [21, 3] is introduced to optimize a super-resolution model in a feature space instead of pixel space. Ledig *et al*. [27] and Sajjadi *et al*. [38] further propose adversarial loss to encourage the network to favor solutions that look more like natural images. With these loss functions the overall visual quality of reconstruction is significantly improved.

Figure 1. The extracted building and plant patches from two low-resolution images look very similar. Using adversarial loss and perceptual loss without prior could add details that are not faithful to the underlying class. More realistic results can be obtained with the correct categorical prior. (**Zoom in for best view**).

Though great strides have been made, texture recovery in SR remains an open problem. Examples are shown in Fig. 1. A variety of different HR patches could have very similar LR counterparts, as shown by the building and plant examples. Generating realistic textures faithful to the inherent class is non-trivial. The results obtained by using perceptual and adversarial losses (without prior) do add fine details to the reconstructed HR image. But if we examine closely, these details are not reminiscent of the textures one would usually observe. Without stronger prior information, existing methods struggle in distinguishing these LR patches and restoring natural and realistic textures thereon.

We believe that the categorical prior, which characterizes the semantic class of a region in an image (*e.g*., sky, building, plant), is crucial for constraining the plausible solution space in SR. We demonstrate the effectiveness of categorical prior using the same example in Fig. 1. Specifically, we try to restore the visually ambiguous plant and building pairs using two different CNN models, each of which is specially trained on a plant dataset and a building dataset. It is observed that generating realistic textures faithful to

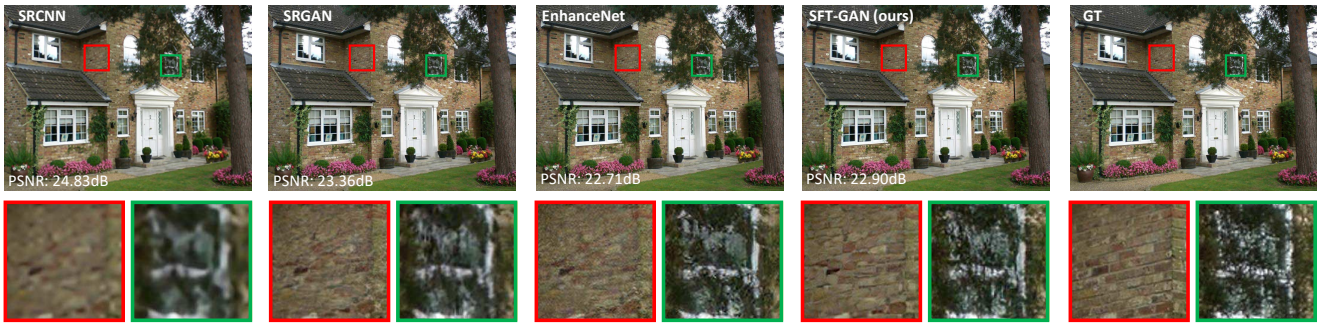

Figure 2. Comparing different SR approaches with downsampling factor $\times 4$: SRCNN [7], SRGAN [27], EnhanceNet [38], our proposed SFT-GAN and the original HR image. SRGAN, EnhanceNet, and SFT-GAN clearly outperform SRCNN in terms of perceptual quality, although they yield lower peak signal-to-noise ratio (PSNR) values. SRGAN and EnhanceNet result in more monotonous textures across different patches while SFT-GAN is capable of generating richer and visually pleasing textures. (**Zoom in for best view**).

the inherent class can be better achieved by selecting the correct class-dedicated model. This phenomenon is previously documented by Timofte *et al*. [47]. They train specialized models for each semantic category on exemplar-based methods [50, 46] and show that SR results can be improved by semantic priors.

In this study, we wish to investigate class-conditional image super-resolution with CNN. This problem is challenging especially when multiple segments of different classes and sizes co-exist in a single image. No previous work has investigated how categorical priors can be obtained and further incorporated into the reconstruction process. We make this attempt by exploring the possibility of using semantic segmentation maps as the categorical prior. Our experiments embrace this choice and show that segmentation maps encapsulate rich categorical prior up to pixel level. In addition, semantic segmentation results on LR images are satisfactory given a contemporary CNN [32, 31, 28] that is fine-tuned on LR images. The remaining question is to find a formulation that allows factorized texture generation in an SR network conditioned on segmentation maps. This is a non-trivial problem. Training a separate SR model for each semantic class is neither scalable nor computationally efficient. Combining LR images with segmentation maps as inputs, or concatenating segmentation maps with intermediate deep features cannot make an effective use of segmentation.

In this work, we present a novel approach known as Spatial Feature Transform (SFT) that is capable of altering the behavior of an SR network through just transforming the features of some intermediate layers of the network. Specifically, an SFT layer is conditioned on semantic segmentation probability maps, based on which it generates a pair of modulation parameters to apply affine transformation spatially on feature maps of the network. The advantages of SFT are three-fold: (1) It is parameter-efficient. Reconstruction of an HR image with rich semantic regions can be achieved with just a single forward pass through transforming the intermediate features of a single network. (2) SFT layers can be easily introduced to existing SR network structures. The layers can be trained end-to-end together with the SR network using conventional loss functions. (3) It is extensible. While we consider categorical prior in our study, other priors such as depth maps can also be applied using the proposed SFT layer. We demonstrate the effectiveness of our approach, named as SFT-GAN, in Fig. 2. More results, a user study, and ablation experiments are provided in Sec. 4.

2. Related Work

Single image super-resolution. Many studies have introduced prior information to help address the ill-posed SR problem. Early methods explore a smoothing prior such as bicubic interpolation and Lanczos resampling [11]. Image priors such as edge features [13, 41], statistics [24, 1] and internal patch recurrence [16] are employed to improve performance. Dong *et al*. [10] train domain specific dictionaries to better recover local structures in a sparse representation framework. Sun *et al*. [42] propose context-constrained super-resolution by learning from texturally similar training segments. Timofte *et al*. [47] investigate semantic priors by training specialized models separately for each semantic category on exemplar-based methods [50, 46]. In contrast to these studies, we explore categorical priors in the form of segmentation probability maps in a CNN framework.

Contemporary SR algorithms are mostly learning-based methods, including neighbor embedding [4], sparse coding [49, 50, 45, 46] and random forest [39]. As an instantiation of learning-based methods, Dong *et al*. [7] propose SRCNN for learning the mapping of LR and HR images in an end-to-end manner. Later on, the field has witnessed a variety of network architectures, such as a deeper network with residual learning [22], Laplacian pyramid structure [26], residual blocks [27], recursive learning [23, 43], and densely connected network [44]. Multi-scale guidance structure has also been proposed for depth map super-resolution [18]. Different losses have also been

proposed. Pixel-wise loss functions, like MSE and Charbonnier penalty [26], encourage the network to find an average of many plausible solutions and lead to overly-smooth results. Perceptual losses [21, 3] are proposed to enhance the visual quality by minimizing the error in a feature space. Ledig *et al*. [27] introduce an adversarial loss, generating images with more natural details. Sajjadi *et al*. [38] develop a similar approach and further explore the local texture matching loss, partly reducing visually unpleasant artifacts. We use the same losses but encourage the network to find solutions under the categorical priors. Enforcing category-specific priors in CNN has been attempted in Xu *et al*. [48] but they only focus on two classes of images, *i.e*., faces and text. Prior is assumed at image-level rather than pixel-level. We take a further step to assume multiple categorical classes to co-exist in an image, and propose an effective layer that enables an SR network to generate rich and realistic textures in a single forward pass conditioned on the prior provided up to the pixel level.

Network conditioning. Our work is inspired by previous studies on feature normalization. Batch normalization (BN) is a widely used technique to ease network training by normalizing feature statistics [19]. Conditional Normalization (CN) applies a learned function of some conditions to replace parameters for feature-wise affine transformation in BN. Some variants of CN have proven highly effective in image style transfer [12, 17, 15], visual question answering [6] and visual reasoning [35]. Perez *et al*. [36] develop a feature-wise linear modulation layer (FiLM), to exploit linguistic information for visual reasoning. This layer can be viewed as a generalization of CN. Perez *et al*. show that the affine transformation in CN needs not be placed after normalization. Features can be directly modulated. FiLM shows promising results in visual reasoning. Nonetheless, the formulation cannot handle conditions with spatial information (*e.g*., semantic segmentation maps) since FiLM accepts a single linguistic input and outputs one scaling and one shifting parameter for each feature map, agnostic to spatial location. Preserving spatial information is crucial for low-level tasks, *e.g*., SR, since these tasks usually require adaptive processing at different spatial locations of an image. Applying FiLM in SR will result in homogeneous spatial feature modulation, hurting SR quality. The proposed SFT layer addresses this shortcoming. It is capable of converting spatial conditions for not only feature-wise manipulation but also spatial-wise transformation.

Semantic guidance. In image generation [20, 5], semantic segments are used as input conditions to generate natural images. Gatys *et al*. [14] use semantic maps to control perceptual factors in neural style transfer. [37] uses semantic segmentation for video deblurring. Zhu *et al*. [52] propose an approach to generate new clothing on a wearer. It first generates human segmentation maps and then uses them to render plausible textures by enforcing region-specific texture rendering. Our work differs from these works mainly in two aspects. First, we use semantic maps to guide texture recovery for different regions in SR domain. Second, we utilize probability maps to capture delicate texture distinction instead of simple image segments.

3. Methodology

Given a single low-resolution image $\boldsymbol{x}$, super-resolution aims at estimating a high-resolution one $\hat{\boldsymbol{y}}$, which is as similar as possible to the ground truth image $\boldsymbol{y}$. Current CNN-based methods use feed-forward networks to directly learn a mapping function $G_{\boldsymbol{\theta}}$ parametrized by $\boldsymbol{\theta}$ as

$$\hat{\boldsymbol{y}} = G_{\boldsymbol{\theta}}(\boldsymbol{x}). \tag{1}$$

In order to estimate $\hat{\boldsymbol{y}}$, a specific loss function $\mathcal{L}$ is designed for SR to optimize $G_{\boldsymbol{\theta}}$ on the training samples,

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \sum\nolimits_i \mathcal{L}(\hat{\boldsymbol{y}}_i, \boldsymbol{y}_i), \tag{2}$$

where $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ are training pairs. Perceptual loss [21, 3] and adversarial loss [27, 38] are introduced to solve the regression-to-the-mean problem that is usually caused by conventional MSE-oriented loss functions. These new losses greatly improve the perceptual quality of reconstructed images. However, the generated textures tend to be monotonous and unnatural, as seen in Fig. 1.

We argue that the semantic categorical prior, *i.e*., knowing which region belongs to the sky, water, or grass, is beneficial for generating richer and more realistic textures. The categorical prior Ψ can be conveniently represented by semantic segmentation probability maps $\mathbf{P}$,

$$\Psi = \mathbf{P} = (P_1, P_2, \ldots, P_k, \ldots, P_K), \tag{3}$$

where P_k is the probability map of k^{th} category and K is the total number of considered categories. To introduce priors in SR, we reformulate Eqn. (1) as

$$\hat{\boldsymbol{y}} = G_{\boldsymbol{\theta}}(\boldsymbol{x}|\Psi), \tag{4}$$

where Ψ defines the prior upon which the mapping function can condition. Note that apart from categorical priors, the proposed formulation is also applicable to other priors such as depth information, which could be helpful to the recovery of texture granularity in SR. And one could easily extend the formulation to consider multiple priors simultaneously. In the following section, we focus on categorical priors and the way we use them to influence the behavior of an SR network.

3.1. Spatial Feature Transform

A Spatial Feature Transform (SFT) layer learns a mapping function $\mathcal{M}$ that outputs a modulation parameter pair

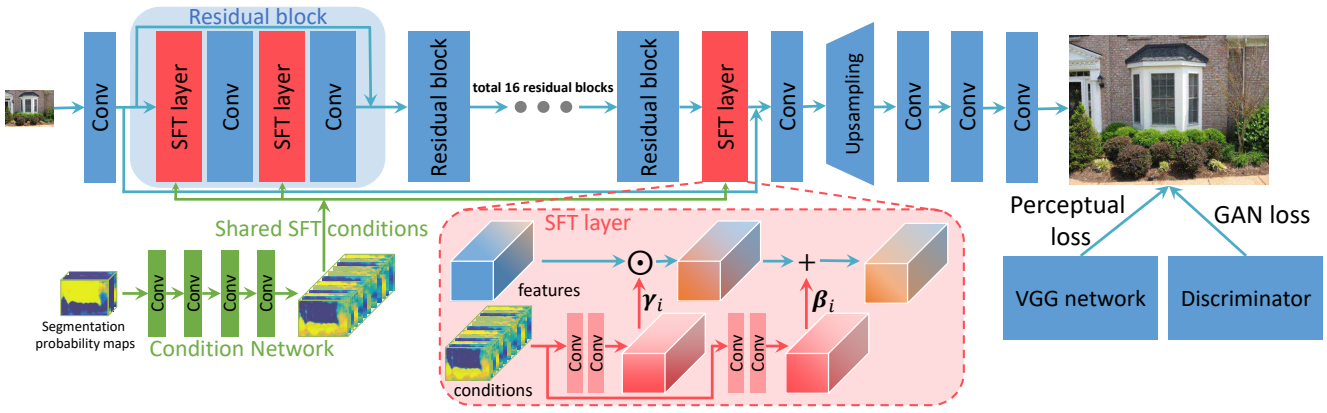

Figure 3. The proposed SFT layers can be conveniently applied to existing SR networks. All SFT layers share a condition network. The role of the condition network is to generate intermediate conditions from the prior, and broadcast the conditions to all SFT layers for further generation of modulation parameters.

$(\boldsymbol{\gamma}, \boldsymbol{\beta})$ based on some prior condition Ψ. The learned parameter pair adaptively influences the outputs by applying an affine transformation spatially to each intermediate feature maps in an SR network. During testing, only a single forward pass is needed to generate the HR image given the LR input and segmentation probability maps.

More precisely, the prior Ψ is modeled by a pair of affine transformation parameters $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ through a mapping function $\mathcal{M} : \Psi \mapsto (\boldsymbol{\gamma}, \boldsymbol{\beta})$. Consequently,

$$\hat{\boldsymbol{y}} = G_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{\gamma}, \boldsymbol{\beta}), \quad (\boldsymbol{\gamma}, \boldsymbol{\beta}) = \mathcal{M}(\Psi). \tag{5}$$

After obtaining $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ from conditions, the transformation is carried out by scaling and shifting feature maps of a specific layer:

$$\mathrm{SFT}(\boldsymbol{F}|\boldsymbol{\gamma}, \boldsymbol{\beta}) = \boldsymbol{\gamma} \odot \boldsymbol{F} + \boldsymbol{\beta}, \tag{6}$$

where $\boldsymbol{F}$ denotes the feature maps, whose dimension is the same as $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, and $\odot$ is referred to element-wise multiplication, *i.e.*, Hadamard product. Since the spatial dimensions are preserved, the SFT layer not only performs feature-wise manipulation but also spatial-wise transformation.

Figure 3 shows an example of implementing SFT layers in an SR network. We provide more details of the SR branch in Sec. 3.2. Here we focus on the conditioning part. The mapping function $\mathcal{M}$ can be arbitrary functions. In this study, we use a neural network for $\mathcal{M}$ so that it can be optimized end-to-end with the SR branch. To further share parameters among multiple SFT layers for efficiency, we use a small condition network to generate shared intermediate conditions that can be broadcasted to all the SFT layers. Meanwhile, we still keep few parameters inside each SFT layer to further adapt the shared conditions to the specific parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, providing fine-grained control to the features.

Segmentation probability maps as prior. We provide a brief discussion on the segmentation network we used. The details are provided in the *supplementary material*. The LR image is first upsampled to the desired HR size with bicubic interpolation. It is then fed into a segmentation network [31] as the input. The network is pretrained on the COCO dataset [30] and then fine-tuned on the ADE dataset [51] with additional animal and mountain images. We train the network separately from the main SR network.

For a sanity check, we study the accuracy of segmentation maps obtained from LR image. In a typical setting of SR studies, LR images are downsampled with a scaling factor of $\times 4$ from HR images. We find that under this resolution, satisfactory segmentation results can still be obtained even on LR images given a modern CNN-based segmentation model [32, 31]. Some LR images and the corresponding segmentation results are depicted in Fig. 4. As can be observed in Fig. 4, LR segmentation is close to that of HR. We have not yet tried segmentation on small objects as this remains a challenging problem in the image segmentation community. During testing, classes that fall outside the pre-defined K segmentation classes will be categorized as 'background' class. In this case, our method would still generate a set of default $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, degenerating itself as SRGAN, *i.e.*, treating all classes equally.

Discussion. There are alternative ways to introduce categorical priors to an SR network. For instance, one can concatenate the segmentation probability maps with the input LR image as a joint input to the network. We find that this method is ineffective in altering the behavior of CNN (see Sec. 4.3). Another method is to directly concatenate the probability maps with feature maps in the SR branch. This approach resembles the multi-texture synthesis network [29][1]. This method, though not as parameter efficient as SFT, amounts to simply adding a post-layer for feature-wise conditional bias. It is thus a special case of

[1]The approach in [29] is not applicable in our work since it can only generate outputs with a fixed size due to the upsampling operation from one-hot vector.

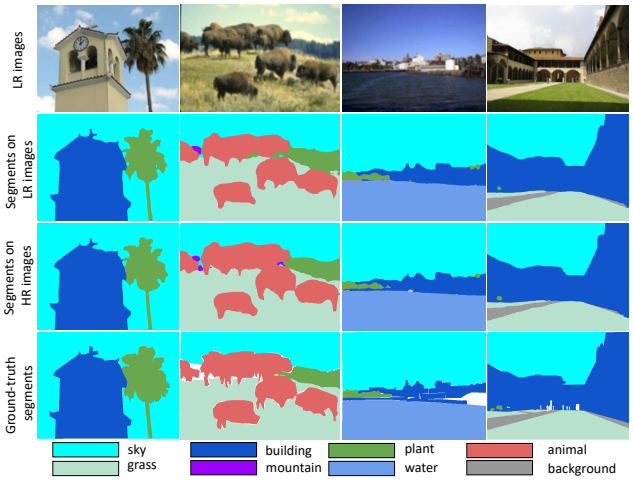

Figure 4. Some examples on segmentation. First row: LR images. Second row: segmentation results on LR images. Third row: segmentation results on HR images. Forth row: Ground-truth segmentation.

SFT. Another more brute-force approach is to first decompose the LR image based on the predicted semantic class and process each region separately using a model trained specifically for that class. These models may share features to save computation. The final output is generated by combining the output of each model class-wise. This method is computationally inefficient as we need to perform forward passes with several CNN models for a single input image.

3.2. Architecture

Our framework is based on adversarial learning, inspired by [27, 38]. Specifically, it consists of one generator $G_{\boldsymbol{\theta}}$ and one discriminator $D_{\boldsymbol{\eta}}$, parametrized by $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ respectively. They are jointly trained with a learning objective given below:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\eta}} \mathbb{E}_{\boldsymbol{y}\sim p_{\text{HR}}} \log D_{\boldsymbol{\eta}}(\boldsymbol{y}) + \mathbb{E}_{\boldsymbol{x}\sim p_{\text{LR}}} \log(1 - D_{\boldsymbol{\eta}}(G_{\boldsymbol{\theta}}(\boldsymbol{x}))),$$

where p_{HR} and p_{LR} are the empirical distributions of HR and LR training samples, respectively.

The architecture of $G_{\boldsymbol{\theta}}$ is shown in Fig. 3. It consists of two streams: a condition network and an SR network. The condition network takes segmentation probability maps as input, which are then processed by four convolution layers. It generates intermediate conditions shared by all the SFT layers. To avoid interference of different categorical regions in one image, we restrict the receptive field of the condition network by using 1×1 kernels for all the convolution layers.

The SR network is built with 16 residual blocks with the proposed SFT layers, which take the shared conditions as input and learn $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ to modulate the feature maps by applying affine transformation. Skip connection [27] is used to ease the training of deep CNN. We upsample features by using nearest-neighbor upsampling followed by a convolution layer. The upsampling operation is performed in the latter part of the network and thus most computation is done in the LR space. Although we have not tried other architectures for the SR network, we believe many contemporary models such as DRRN [43] and MemNet [44] are applicable and can equally be benefited from the SFT layer.

For discriminator $D_{\boldsymbol{\eta}}$, we apply a VGG-style [40] network of strided convolutions to gradually decrease the spatial dimensions. The full architecture and details are provided in the *supplementary material*.

3.3. Loss Function

We draw inspiration from [27, 38] and apply perceptual loss and adversarial loss in our model. The perceptual loss measures the distance in a feature space. To obtain the feature maps, we use a pre-trained 19-layer VGG network [40], denoted as ϕ,

$$\mathcal{L}_P = \sum_i \|\phi(\hat{\boldsymbol{y}}_i) - \phi(\boldsymbol{y}_i)\|_2^2. \tag{7}$$

Similar to [27], we use the feature maps obtained by the forth convolution before the fifth max-pooling layer and compute the MSE on their feature activations.

The adversarial loss $\mathcal{L}_D$ from GAN is also used to encourage the generator to favor the solutions in the manifold of natural images,

$$\mathcal{L}_D = \sum_i \log(1 - D_{\boldsymbol{\eta}}(G_{\boldsymbol{\theta}}(\boldsymbol{x}_i))). \tag{8}$$

4. Experiments

Implementation details. In this work, we focus on outdoor scenes since their textures are rich and well-suited for our study. For example, the sky is smooth and lacks sharp edges, while the building is rich of geometric patterns. The water presents smooth surface with waves, while the grass has matted textures. We assume seven categories, *i.e.*, sky, mountain, plant, grass, water, animal and building. A 'background' category is used to encompass regions that do not appear in the aforementioned categories.

Following [27], all experiments were performed with a scaling factor of $\times 4$ between LR and HR images. We initialized the SR network by parameters pre-trained with perceptual loss and GAN loss on ImageNet dataset. After removing low resolution images of size below 30kB, we obtained roughly 450k training images. During training, we followed existing studies [7, 27] to obtain LR images by downsampling HR images using MATLAB bicubic kernel. The mini-batch size was set to 16. The spatial size of cropped HR and LR sub-images were 96×96 and 24×24, respectively.

After initialization, with the same training setting, we fine-tuned our full network on outdoor scenes condition-

Figure 5. GAN-based methods (SRGAN [27], EnhanceNet [38] and ours) clearly outperform PSNR-oriented approaches in term of perceptual quality. Our proposed SFT-GAN is capable of generating richer and more realistic textures among different categories. The first and second restored images show that our method captures the characteristics of animal fur and building brick. In the third image, SRGAN tends to produce unpleasant water waves. (**Zoom in for best view**).

ally on the input segmentation probability maps. In particular, we collected a new outdoor dataset by querying images from search engines using the defined categories as keywords. The dataset was divided into training and test partitions (OutdoorSceneTrain and OutdoorSceneTest)[2]. For OutdoorSceneTrain, we cropped each image so that only one category exists, resulting in 1k to 2k images for each category. Background images were randomly sampled from ImageNet. The total number of training images is 10,324. The OutdoorSceneTest partition consists of 300 images and they are not pre-processed. Segmentation probability maps were generated by the segmentation network (Sec. 3.1).

For optimization, we used Adam [25] with $\beta_1 = 0.9$. The learning rate was set to 1×10^{-4} and then decayed by a factor of 2 every 100k iterations. We alternately optimized the generator and discriminator until the model converged at about 5×10^5 iterations. Inspired by [34], our discriminator not only distinguishes whether the input is real or fake, but also predicts which category the input belongs to. This is possible since our training images were cropped to contain only one category. This restriction was not applied on test images. We find this strategy facilitates the generation of images with more realistic textures.

[2]All data, codes and models can be downloaded from http://mmlab.ie.cuhk.edu.hk/projects/SFTGAN/.

We did not conduct our main evaluation on standard benchmarks such as Set5 [2], Set14 [50] and BSD100 [33] since these datasets are lack of regions with well-defined categories. Nevertheless, we will show that our method still perform satisfactorily on out-of-category examples in Sec. 4.3. Results (PSNR, SSIM) on standard benchmarks are provided in the *supplementary material*.

4.1. Qualitative Evaluation

Figure 5 shows the qualitative results of different models including PSNR-oriented methods, such as SRCNN [7], VDSR [22], LapSRN [26], DRRN [43], MemNet [44], and GAN-based methods, such as SRGAN [27] and EnhanceNet [38]. More results are provided in the *supplementary material*. For SRGAN, we re-implemented their method and fine-tuned the model with the same setting as ours. We directly used the released test code of EnhanceNet since no training code is available. Despite of preserving sharp edges, PSNR-oriented methods always produce blurry textures. SRGAN and EnhanceNet largely improve the high-frequency details, however, they tend to generate monotonous and unnatural textures, like water waves in Fig. 5. Our method employs categorical priors to help capture the characteristics of each category, leading to more natural and realistic textures.

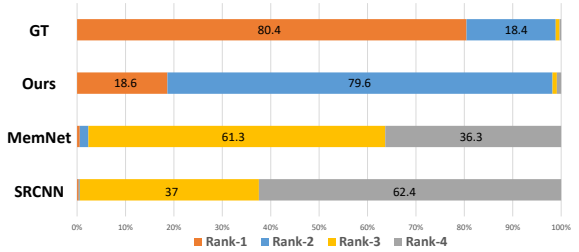

Figure 6. User study results of ranking SRCNN [7], MemNet [44], SFT-GAN (ours), and the original HR image. Our method outperforms PSNR-oriented methods by a large margin.

4.2. User Study

We performed a user study to quantify the ability of different approaches to reconstruct perceptually convincing images. To better compare our method against PSNR-oriented baselines and GAN-based approaches, we divided the evaluations into two sessions. In the first session, we focused on PSNR-oriented baselines. The users were requested to rank 4 versions of each image: SRCNN [7], MemNet [44] (the state-of-the-art PSNR-oriented method), our SFT-GAN, and the original HR image according to their visual quality. We used 30 random images chosen from OutdoorSceneTest and all images were presented in a randomized fashion. In the second session, we focused on GAN-based methods so that the user can concentrate on the texture quality. The subjects were shown the super-resolved image pairs (enlarged texture patches were depicted to facilitate the comparison). Each pair consists of an image of the proposed SFT-GAN and the counterpart generated by SRGAN [27] or EnhanceNet [38]. The users were asked to pick the image with more natural and realistic textures. This session involved 96 randomly selected images in total.

We asked 30 users to finish our user study. The results of the first and second sessions are presented in Fig. 6 and Fig. 7, respectively. The results of the first session show that SFT-GAN outperforms the PSNR-oriented methods by a large margin. This is not surprising since PSNR-oriented methods always produce blurry results especially in texture regions. Our method sometimes generates good-quality images comparable to HR causing confusion within the users. In the second session, our method is ranked higher than SRGAN [27] and EnhanceNet [38], especially in building, animal, and grass categories. Comparable performance is found on sky and plant categories.

4.3. Ablation Study

From segmentation probability maps to feature modulation parameters. Our method modulates intermediate features based on segmentation probability maps. We investigate the relationship between the probability and feature modulation parameters, as depicted in Fig. 8. All SFT layers exhibit similar behavior so that we only present its

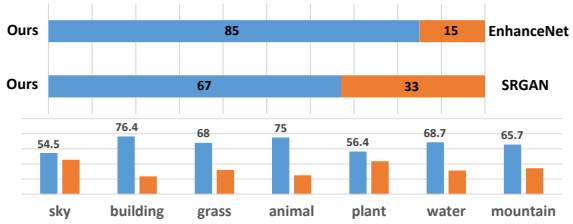

Figure 7. First row: the results of user studies, comparing our method with SRGAN [27] and EnhanceNet [38]. Second row: our methods produce visual results that are ranked higher in all categories in comparison with SRGAN [27].

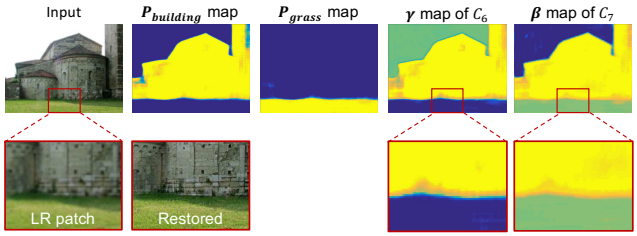

Figure 8. The modulation parameters γ and β have a close relationship with probability maps P and contain spatial information. The boundary of different categories is clear without interference. C_i denotes the i^{th} channel of the first SFT layer. (**Zoom in for best view**)

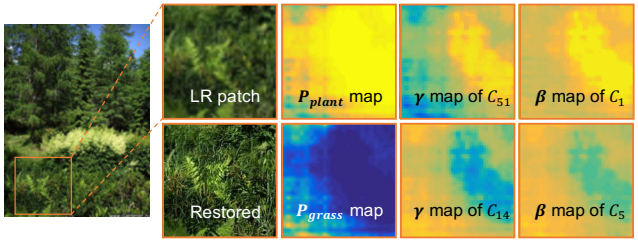

Figure 9. The SFT layer can provide delicate feature transform under the segmentation probability. C_i denotes the i^{th} channel of the first SFT layer. (**Zoom in for best view**)

first layer for this analysis. In the top row, we show an image where building and grass co-exist. It is observed that modulation parameters γ and β are different for various categorical regions to exert meaningful spatial-wise transformation. From the heat map of γ and β, we can see that the modulation parameter maps are closely related to the segmentation probability maps. The boundary of building and grass is clear with a sharp transition. With the guidance of the probability map, our model is able to generate building and grass textures simultaneously without interference of different categories.

Some classes, like plant and grass, may not be clearly distinguishable by their visual appearance. They interlace with each other without a clear boundary. Despite the ambiguity, the probability maps are still capable of capturing the semantics to certain extent and the SFT layers reflect the subtle differences between categories in its spatial transformation. In Fig. 9, the upper row shows the probability

map and modulation parameters activated for plant while the lower row shows those for grass. Distinct activations with smooth transition can be observed. As a result, textures generated by SFT-GAN become more realistic.

Robustness to out-of-category examples. Our model mainly focuses on outdoor scenes and it is effective given segmentation maps of the pre-defined K classes. Despite the assumption, it is also robust to other scenes where segmentation results are not available. As shown in Fig. 10, the SFT-GAN can still produce comparative results with SRGAN when all the regions are deemed as 'background'. More results are provided in the *supplementary material*.

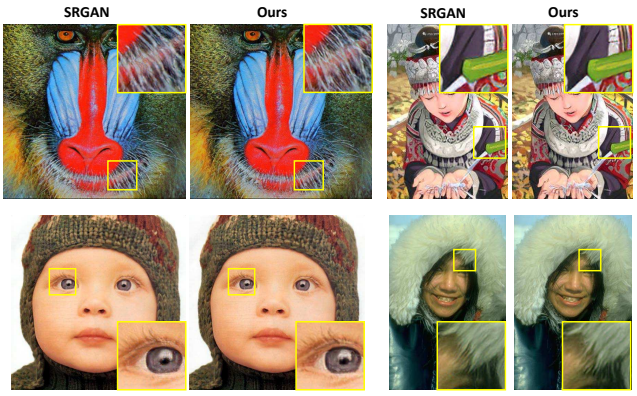

Figure 10. When facing with other scenes or the absence of segmentation probability maps, our model degenerates itself as SRGAN and produces comparative results with SRGAN. (**Zoom in for best view**)

Comparison with other conditioning methods. We qualitatively compare with several alternatives for conditioning SR network, which are already discussed in Sec. 3.1.
1) *Input concatenation* – This method concatenates the segmentation probability maps with the LR image as a joint input to the network. This is equivalent to adding SFT conditional bias at the input layer.
2) *Compositional mapping* – This method is identical to Zhu *et al*. [52]. It decomposes an LR image based on the predicted semantic classes and processes each region separately using a specific model for that class. Different models share parameters at lower layers.
3) *FiLM* [36] – This method predicts one parameter for each feature map without spatial information and then uses these parameters to modulate the feature maps.

As can be observed from Fig. 11, the proposed SFT-GAN yields outputs that are perceptually more convincing. Naive input concatenation is not sufficient to exert the necessary condition for class-specific texture generation. Compositional mapping produces good results but it is not parameter efficient ($\times 2.5$ parameters as ours). It is also computationally inefficient as we need to forward several times for a single input image. FiLM [36] cannot handle the situations where multiple categorical classes co-exist in an image

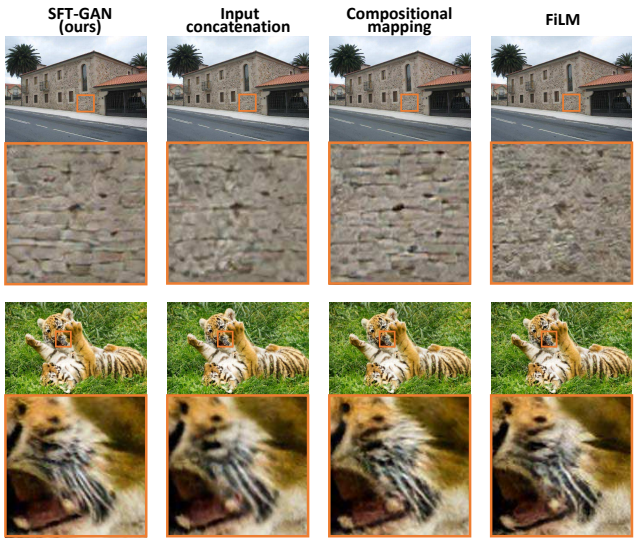

Figure 11. Comparison with other conditioning methods - input concatenation, compositional mapping and FiLM [36].

since it predicts one parameter for each feature map, agnostic to spatial information. For example, in the first image of Fig. 11, the road and sky interfere with the building's structure and thus noisy bricks are generated. Similarly in the second image, the animal's fine texture is severely affected by the grass.

5. Discussion and Conclusion

We have explored the use of semantic segmentation maps as categorical prior for constraining the plausible solution space in SR. A novel Spatial Feature Transform (SFT) layer has been proposed to efficiently incorporate the categorical conditions into a CNN-based SR network. Thanks to the SFT layers, our SFT-GAN is capable of generating distinct and rich textures for multiple semantic regions in a super-resolved image in just a single forward pass. Extensive comparisons and a user study demonstrate the capability of SFT-GAN in generating realistic and visually pleasing textures, outperforming previous GAN-based methods [27, 38].

Our work currently focuses on SR of outdoor scenes. Despite robust to out-of-category images, it does not consider priors of finer categories, especially for indoor scenes, *e.g*., furniture, appliance and silk. In such a case, it puts forward challenging requirements for segmentation tasks from an LR image. Future work aims at addressing these shortcomings. Furthermore, segmentation and SR may benefit from each other and jointly improve the performance.

Acknowledgement. This work is supported by SenseTime Group Limited and the General Research Fund sponsored by the Research Grants Council of the Hong Kong SAR (CUHK 14241716, 14224316. 14209217).

References

[1] H. A. Aly and E. Dubois. Image up-sampling using total-variation regularization with a new observation model. *TIP*, 14(10):1647–1659, 2005. 2

[2] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 6

[3] J. Bruna, P. Sprechmann, and Y. LeCun. Super-resolution with deep convolutional sufficient statistics. In *ICLR*, 2015. 1, 3

[4] H. Chang, D.-Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *CVPR*, 2004. 2

[5] Q. Chen and V. Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 3

[6] H. de Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. Courville. Modulating early visual processing by language. *arXiv preprint arXiv:1707.00683*, 2017. 3

[7] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, 2014. 1, 2, 5, 6, 7

[8] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *TPAMI*, 38(2):295–307, 2016. 1

[9] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, 2016. 1

[10] W. Dong, L. Zhang, G. Shi, and X. Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *TIP*, 20(7):1838–1857, 2011. 2

[11] C. E. Duchon. Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology*, 18(8):1016–1022, 1979. 2

[12] V. Dumoulin, J. Shlens, M. Kudlur, A. Behboodi, F. Lemic, A. Wolisz, M. Molinaro, C. Hirche, M. Hayashi, E. Bagan, et al. A learned representation for artistic style. In *ICLR*, 2016. 3

[13] R. Fattal. Image upsampling via imposed edge statistics. In *TOG*. ACM, 2007. 2

[14] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann, and E. Shechtman. Controlling perceptual factors in neural style transfer. In *CVPR*, 2017. 3

[15] G. Ghiasi, H. Lee, M. Kudlur, V. Dumoulin, and J. Shlens. Exploring the structure of a real-time, arbitrary neural artistic stylization network. In *BMVC*, 2017. 3

[16] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In *ICCV*, 2009. 2

[17] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017. 3

[18] T.-W. Hui, C. C. Loy, and X. Tang. Depth map super-resolution by deep multi-scale guidance. In *ECCV*, 2016. 2

[19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICMR*, 2015. 3

[20] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 3

[21] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 1, 3

[22] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, 2016. 1, 2, 6

[23] J. Kim, J. Kwon Lee, and K. Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR*, 2016. 1, 2

[24] K. I. Kim and Y. Kwon. Single-image super-resolution using sparse regression and natural image prior. *TPAMI*, 32(6):1127–1133, 2010. 2

[25] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[26] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, 2017. 1, 2, 3, 6

[27] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7, 8

[28] X. Li, Z. Liu, P. Luo, C. C. Loy, and X. Tang. Not all pixels are equal: difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, 2017. 2

[29] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang. Diversified texture synthesis with feed-forward networks. In *CVPR*, 2017. 4

[30] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 4

[31] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang. Deep learning markov random field for semantic segmentation. *TPAMI*, 2017. 2, 4

[32] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2, 4

[33] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001. 6

[34] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier GANs. *arXiv preprint arXiv:1610.09585*, 2016. 6

[35] E. Perez, H. de Vries, F. Strub, V. Dumoulin, and A. Courville. Learning visual reasoning without strong priors. *arXiv preprint arXiv:1707.03017*, 2017. 3

[36] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville. FiLM: Visual reasoning with a general conditioning layer. *arXiv preprint arXiv:1709.07871*, 2017. 3, 8

[37] W. Ren, J. Pan, X. Cao, and M.-H. Yang. Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In *ICCV*, 2017. 3

[38] M. S. Sajjadi, B. Schölkopf, and M. Hirsch. EnhanceNet: Single image super-resolution through automated texture synthesis. In *ICCV*, 2017. 1, 2, 3, 5, 6, 7, 8

[39] S. Schulter, C. Leistner, and H. Bischof. Fast and accurate image upscaling with super-resolution forests. In *CVPR*, 2015. 2

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[41] J. Sun, Z. Xu, and H.-Y. Shum. Image super-resolution using gradient profile prior. In *CVPR*, 2008. 2

[42] J. Sun, J. Zhu, and M. F. Tappen. Context-constrained hallucination for image super-resolution. In *CVPR*, 2010. 2

[43] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In *CVPR*, 2017. 1, 2, 5, 6

[44] Y. Tai, J. Yang, X. Liu, and C. Xu. MemNet: A persistent memory network for image restoration. In *ICCV*, 2017. 1, 2, 5, 6, 7

[45] R. Timofte, V. De Smet, and L. Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, 2013. 2

[46] R. Timofte, V. De Smet, and L. Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV*, 2014. 2

[47] R. Timofte, V. De Smet, and L. Van Gool. Semantic super-resolution: When and where is it useful? *CVIU*, 142:1–12, 2016. 2

[48] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang. Learning to super-resolve blurry face and text images. In *CVPR*, 2017. 3

[49] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *CVPR*, 2008. 2

[50] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *International Conference on Curves and Surfaces*, 2010. 2, 6

[51] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017. 4

[52] S. Zhu, S. Fidler, R. Urtasun, D. Lin, and C. C. Loy. Be your own prada: Fashion synthesis with structural coherence. In *ICCV*, 2017. 3, 8