# CFSNet: Toward a Controllable Feature Space for Image Restoration

Wei Wang [*1], Ruiming Guo [*2], Yapeng Tian[3] and Wenming Yang [†1]

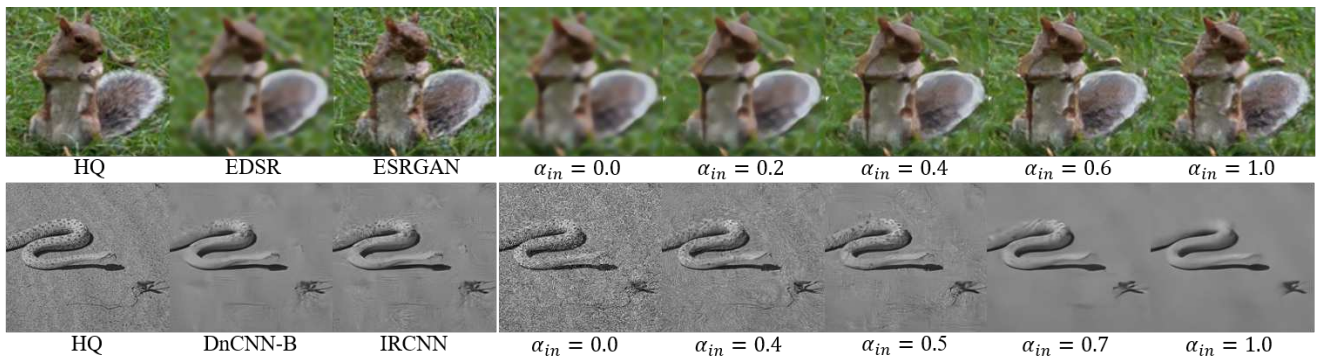[1]Tsinghua University, [2]Chinese University of Hong Kong, [3]University of Rochester

Figure 1. The perception-distortion trade-off of image super-resolution (1st row) and the trade-off between noise reduction and detail preservation (2nd row). At test-time, when using CFSNet, users can easily adjust an input control variable $\alpha_{in}$ to attain the most satisfactory result according to personal preferences. In contrast, the fixed methods (*e.g.*, EDSR, DnCNN-B) can not always guarantee the optimal visual quality.

## Abstract

*Deep learning methods have witnessed the great progress in image restoration with specific metrics (e.g., PSNR, SSIM). However, the perceptual quality of the restored image is relatively subjective, and it is necessary for users to control the reconstruction result according to personal preferences or image characteristics, which cannot be done using existing deterministic networks. This motivates us to exquisitely design a unified interactive framework for general image restoration tasks. Under this framework, users can control continuous transition of different objectives, e.g., the perception-distortion trade-off of image super-resolution, the trade-off between noise reduction and detail preservation. We achieve this goal by controlling the latent features of the designed network. To be specific, our proposed framework, named Controllable Feature Space Network (CFSNet), is entangled by two branches based on different objectives. Our framework can adaptively learn the coupling coefficients of different layers and channels, which provides finer control of the restored im-*
*age quality. Experiments on several typical image restoration tasks fully validate the effective benefits of the proposed method. Code is available at* `https://github.com/qibao77/CFSNet`.

## 1. Introduction

Image restoration is a classic ill-posed inverse problem that aims to recover high-quality images from damaged images affected by various kinds of degradations. According to the types of degradation, it can be categorized into different subtasks such as image super-resolution, image denoising, JPEG image deblocking, etc.

The rise of deep learning has greatly facilitated the development of these subtasks. But these methods are often goal-specific, and we need to retrain the network when we deal with images different from the training dataset. Furthermore, most methods usually aim to pursue high reconstruction accuracy in terms of PSNR or SSIM. However, image quality assessment from personal opinion is relatively subjective, and low reconstruction distortion is not always consistent with high visual quality [4]. In addition, in many

---
*Denotes equal contribution.
†Denotes corresponding author.

practical applications (*e.g.*, mobile), it is often challenging to obtain user's preference and the real degradation level of the corrupted images. All of these appeal to an interactive image restoration framework which can be applied to a wide variety of subtasks. However, to the best of our knowledge, currently there are few available networks which can satisfy both interactivity and generality requirements.

Some designs have been proposed to improve the flexibility of deep methods. Take image denoising for example, data augmentation is widely used to improve the generalization of a model. Training with the dataset which contains a series of noise levels, a single model can be applied to blind denoising task [37]. However, this method still produces a fixed reconstruction result of the input, which does not necessarily guarantee satisfactory perceptual quality (as shown in Fig. 1). An alternative choice, Zhang *et al.* [39] concatenated a tunable noise level map with degraded images as input to handle blind image denoising task. Though this scheme is also user-friendly, it can not be generalized to other tasks. In image super-resolution, [24] added noise to the input to control the compromise between perceptual quality and distortion. However, this scheme is specific for image super-resolution and cannot guarantee smooth and continuous control.

In this paper, to rectify these weaknesses, we propose a novel framework equipped with controllability for human perception-oriented interactive image restoration. To be more specific, we realize the interactive control of the reconstruction result by tuning the features of each unit block, called coupling module. Each coupling module consists of a main block and a tuning block. The parameters of two blocks are obtained under two endpoint optimization objectives. Taking image super-resolution as an example, the main block is optimized for low distortion while the tuning block is optimized for high perceptual quality. Besides, as a key to achieving fine feature control, we assign the high-degree-of-freedom coupling coefficients adaptively learned from a control scalar to each coupling module.

Our main contributions can be summarized as follows:

▶ We propose a novel controllable end-to-end framework for interactive image restoration in a fine-grained way.

▶ We propose a coupling module and an adaptive learning strategy of coupling coefficients to improve reconstruction performance.

▶ Our CFSNet outperforms the state-of-the-art methods on super-resolution, JPEG image deblocking and image denoising in terms of flexibility and visual quality.

## 2. Related Work

**Image Restoration.** Deep learning methods have been widely used in image restoration. [15, 18, 29, 30, 35, 40, 41]

continuously deepen, widen or lighten the network structure, aiming at improving the super-resolution accuracy as much as possible. While [13, 17, 23, 26] paid more attention to the design of loss function to improve visual quality. Besides, [4, 20, 33] explored the perception-distortion trade-off. In [8], Dong *et al.* adopted ARCNN built with several stacked convolutional layers for JPEG image deblocking. Zhang *et al.* [39] proposed FFDNet to make image denoising more flexible and effective. Guo *et al.* [11] designed CBDNet to handle blind denoising of real images. Different from these task-specific methods, [19, 20, 37, 38] proposed some unified schemes that can be employed to different image restoration tasks. However, these fixed networks are not flexible enough to deal with volatile user needs and application requirements.

**Controllable Image Transformation.** In high-level vision task, many technologies have been explored to implement controllable image transformation. [21] and [36] incorporated facial attribute vector into network to control facial appearance (*e.g.*, gender, age, beard). In [31], deep feature interpolation was adopt to implement automatic high-resolution image transformation. [14] also proposed a scheme that controls adaptive instance normalization (AdaIN) in feature space to adjust high-level attributes. Shoshan *et al.* [28] inserted some tuning blocks in the main network to allow modification of the network. However, all of these methods are designed for high-level vision tasks and can not be directly applied to image restoration. To apply controllable image transformation to low-level vision tasks, Wang *et al.* [32] performed interpolation in the parameter space, but this method can not guarantee the optimality of the outputs, which inspires us to further explore fine-grain control of image restoration.

## 3. Proposed Method

In this section, we first provide an overview of the proposed framework, called CFSNet and then, present the modeling process inspired by the image super-resolution problem. Instead of specializing for the specific super-resolution task, we finally generalize our CFSNet to multiple image restoration tasks, including denoising and deblocking. Moreover, we give the explicit model interpretation based on the manifold learning to show the intrinsic rationality of the proposed network. At the end of this section, we show the superiority and improvements of the proposed CFSNet through the detailed comparison with the current typical related methods.

### 3.1. Basic Network Architecture

As shown in Fig. 2, our CFSNet consists of a main branch and a tuning branch. The main branch contains $M$ main blocks (residual blocks [18]) while the tuning branch contains $M$ tuning blocks with additional $2 * M + 3$ fully
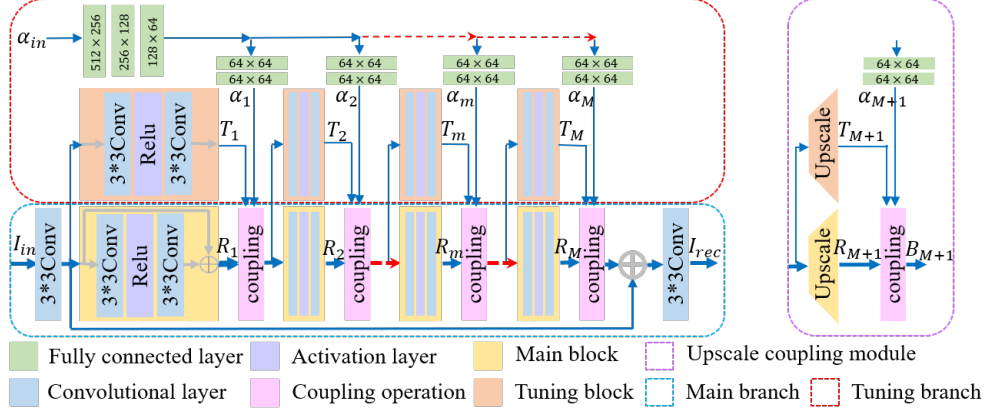
Figure 2. The framework of our proposed controllable feature space network (CFSNet).

connected layers. A pair of main block and tuning block constitute a coupling module by using a coupling operation to combine the features of the two branches effectively. We take the original degraded image $I_{in}$ and the control scalar $\alpha_{in}$ as the input and output the restored image $I_{rec}$ as the final result.

In the beginning, we first use a $3 \times 3$ convolutional layer to extract features from the degraded image $I_{in}$,

$$B_0 = F_{in}(I_{in}), \qquad (1)$$

where $F_{in}(\cdot)$ represents the feature extraction function and $B_0$ serves as the input of next stage. Here, together with the input image $I_{in}$, we introduce a control scalar $\alpha_{in}$ to balance the different optimization goals. To be more specific, there are 3 shared fully connected layers to transform the input scalar $\alpha_{in}$ into multi-channels vectors and 2 independent fully connected layers to learn the optimal coupling coefficient for each coupling module:

$$\alpha_m = F_m^{ind}(F_m^{sha}(\alpha_{in})), \qquad (2)$$

where both $F_m^{sha}(\cdot)$ and $F_m^{ind}(\cdot)$ denote the function of the shared and independent fully connected layers, and $\alpha_m$ is the coupling coefficient vector of the $m$-th coupling module. Each coupling module couples the output of a main block and a tuning block as follows:

$$\begin{aligned} B_m &= F_m(R_m, T_m) \\ &= F_m(F_m^{main}(B_{m-1}), F_m^{tun}(B_{m-1})), \end{aligned} \qquad (3)$$

where $F_m(\cdot)$ represents the $m$-th coupling operation, $R_m$ and $T_m$ denote the output features of $m$-th main block and $m$-th tuning block respectively, $F_m^{main}(\cdot)$ and $F_m^{tun}(\cdot)$ are the $m$-th main block function and $m$-th tuning block function respectively. To address the image super-resolution task, we add an extra coupling module consisting of the upscaling block before the last convolutional layer, as shown

in Fig. 2. Specifically, we utilize sub-pixel convolutional operation (convolution + pixel shuffle) [27] to upscale feature maps. Finally, we use a $3 \times 3$ convolutional layer to get the reconstructed image,

$$I_{rec} = F_{out}(B_M + B_0) \ or \ I_{rec} = F_{out}(B_{M+1}), \qquad (4)$$

where $F_{out}(\cdot)$ denotes convolution operation. The overall reconstruction process can be expressed as

$$I_{rec} = F_{CFSN}(I_{in}, \alpha_{in}; \theta_{main}, \theta_{tun}, \theta_{\alpha}), \qquad (5)$$

where $F_{CFSN}(\cdot)$ represents the function of our proposed CFSNet. $\theta_{main}$, $\theta_{tun}$ and $\theta_{\alpha}$ represent the parameters of main branch, all tuning blocks and all fully connected layers respectively.

Since the two branches of our framework are based on different optimization objectives, in further detail, our training process can be divided into two steps:

**Step 1** Set the control variable $\alpha_{in}$ as 0. Train the main branch with the loss function $L_1(I_{rec}, I_g; \theta_{main})$, where $I_g$ is the corresponding ground truth image.

**Step 2** Set the control variable $\alpha_{in}$ as 1. Map the control variable $\alpha_{in}$ to different coupling coefficients $\{\alpha_m\}$, fix parameters of the main branch and train the tuning branch with another loss function $L_2(I_{rec}, I_g; \theta_{tun}, \theta_{\alpha})$.

### 3.2. Coupling module

We now present the details of our coupling module. We mainly introduce our design from the perspective of image super-resolution. In order to balance the trade-off between perceptual quality and distortion, we usually realize it by modifying the penality parameter $\lambda$ of the loss terms [4],

$$L_{gen} = L_{distortion} + \lambda L_{adv}, \qquad (6)$$

where $L_{distortion}$ denotes distortion loss (*e.g.*, MSE and MAE), $L_{adv}$ contains GAN loss [10, 33] and perceptual loss [13], $\lambda$ is a scalar. We usually pre-train the network with $L_{distortion}$ loss, then we fine-tune the network with combined loss $L_{gen}$ to reach a different working point for the trade-off determined by the value of $\lambda$. That is to say, if we regard pre-trained results as a reference point, then we can start from the reference point and gradually convert it to the result of another optimization goal.

However, it is not efficient to train a network for each different value of $\lambda$. In order to address this issue, we convert the control scalar $\lambda$ to the input and directly control the offset of the reference point in latent feature space. For this purpose, we implement a controllable coupling module to couple reference features learned with $L_{distortion}$ and new features learned with $L_{gen}$ together. We set the feature based on distortion optimization as the reference point which is denoted as $R_m$. In the process of optimization based on perceptual quality, we keep the reference point unchanged and set $T_m - R_m$ as direction of change. In other words, in a coupling module, part of features are provided by reference information, and the other part are obtained from new exploration:

$$B_m = R_m + \alpha_m(T_m - R_m) = (1-\alpha_m)R_m + \alpha_m T_m, \quad (7)$$

where $T_m \in \mathbb{R}^{W \times H \times C}$, $R_m \in \mathbb{R}^{W \times H \times C}$, $\alpha_m \in \mathbb{R}^C$ denotes the $m$-th coefficient, and $C$ is the number of channels.

It is worth noting that different main blocks provide different reference information, so we should treat them differently. We expect to endow each coupling module a different coupling coefficient to make full use of reference information. Therefore, our control coefficients $\{\alpha_m\}$ are learned from optimization process. To be more specific, we use some fully connected layers to map a single input control scalar $\alpha_{in}$ into different coupling coefficients (Eq. 2). The proposed network will find the optimal coupling mode since our control coefficients $\{\alpha_m\}$ are adaptive not fixed.

Thanks to the coupling module and adaptive learning strategy, we can realize continuous and smooth transition by a single control variable $\alpha_{in}$. Moreover, if this framework achieves an excellent trade-off between perceptual quality and distortion for super-resolution, then can we generalize this model to other restoration tasks? After the theoretical analysis and experimental tests, we find that this framework is applicable to a wide variety of image restoration tasks. In the next section, we will provide a more general theoretical explanation of our model.

### 3.3. Theoretical analysis

Suppose there is a high-dimensional space containing all natural images. The degradation process of a natural image can be regarded as continuous in the space. So approximately, these degraded images are adjacent in the space. It
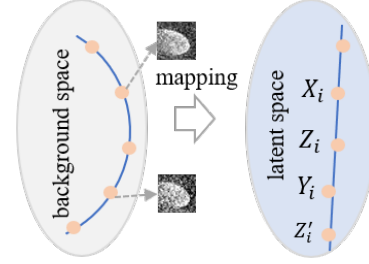


Figure 3. Neural network mapping gradually disentangles data manifolds. We can represent unknown point with known point in latent space.

is possible to approximate the reconstruction result of unknown degradation level with the results of known degradation levels. Unfortunately, natural images lie on an approximate non-linear manifold [34]. As a result, simple image interpolation tends to introduce some ghosting artifacts or other unexpected details to final results.

Instead of operating in the pixel space, we naturally turn our attention to the feature space. Some literatures indicate that the data manifold can be flattened by neural network mapping and we can approximate the mapped manifold as a Euclidean space [2, 5, 28]. Based on this hypothesis, as shown in Fig. 3, we denote $X_i$ and $Y_i$ in the latent space as two endpoints respectively, and we can represent unknown point $Z_i$ as a affine combination of $X_i$ and $Y_i$:

$$Z_i \approx \alpha_i X_i + (1 - \alpha_i)Y_i, \quad (8)$$

where $\alpha_i$ is the $i$-th combination coefficient. This is exactly the formula of the controllable coupling module Eq. 7. However, we should also note that this hypothesis is influenced by the depth and width of CNN. In other words, we do not know the degree of flattening that can be achieved by different channels and different layers. Thus the combination coefficients of different channels and different layers should be discrepant. Besides, we hope to find the optimal combination coefficients through the optimization process:

$$\alpha^* = \arg\min_{\alpha_{ij}} \sum_{i=1}^{M} \sum_{j=1}^{C} \left[ Z_{ij} - (\alpha_{ij} X_{ij} + (1 - \alpha_{ij}) Y_{ij}) \right], \quad (9)$$

where $\alpha^* = \{\alpha_{i,j} | i = 1 \cdots M, j = 1 \cdots C\}$ represents the optimal solution. However, it is difficult to directly obtain the optimal $\alpha^*$, because the unknown working point $Z_{ij}$ cannot in general be computed tractably. So we solve Eq. 9 in an implicit way. Specifically, we map the input control variable $\alpha_{in}$ to different combination coefficients with some stacked fully connected layers, and then, we can approximate the above process into optimizing the parameters of the linear mapping network:

$$\alpha^* \approx \hat{\alpha} = F_{alpha}(\alpha_{in}; \theta_\alpha), \quad (10)$$

where $F_{alpha}$ denotes mapping function of $\alpha_{in}$, $\hat{\alpha}$ is the approximated solution of the optimal $\alpha^*$. Fortunately, this network can be embedded into our framework. Therefore, we can optimize the parameters of the linear mapping network and the tuning blocks in one shot. The entire optimization process (corresponding to Step 2) can be expressed as

$$\theta_{tun}, \theta_\alpha = \underset{\theta_{tun}, \theta_\alpha}{\arg\min} \ L_2(F_{CFSN}(I_{in}, \alpha_{in}; \theta_{tun}, \theta_\alpha), I_g), \tag{11}$$

### 3.4. Discussions

**Difference to Dynamic-Net** Recently, Dynamic-Net [28] realized interactive control of continuous image conversion. In contrast, there are two main differences between Dynamic-Net and our CFSNet. First of all, Dynamic-Net is mainly designed for image transformation tasks, like style transfer. It's difficult to achieve the desirable results when we use Dynamic-Net directly for image restoration tasks. While motivated by super-resolution, we design the proposed CFSNet for low-level image restoration. Secondly, in Dynamic-Net, as shown in Fig. 4, they directly tune multiple scalars $\{\alpha_m\}$ to get different outputs. While, in our framework coupling coefficient $\alpha_m$ is a vector and is learned adaptively from the input scalar $\alpha_{in}$ through the optimization process. This is more user-friendly and we explain the reasonability of this design in Sec. 3.3.
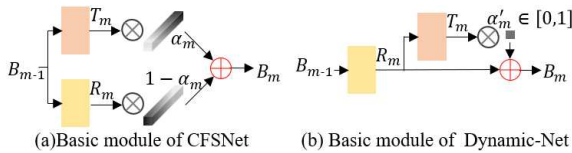


Figure 4. Basic module. Yellow and orange bar represent main block and tuning block respectively.

**Difference to Deep Network Interpolation** Deep Network Interpolation (DNI) is another choice to control the compromise between perceptual quality and distortion [32, 33]. DNI also can be applied to many low-level vision tasks [32]. However, this method needs to train two networks with the same architecture but different losses and will generate a third network to control. In contrast, our framework can achieve better interactive control with a unified end-to-end network. Moreover, our framework makes better use of reference information using coupling module. DNI performs interpolation in the parameter space to generate continuous transition effects, and interpolation coefficients are kept the same in the whole parameter space. However, this simple strategy can not guarantee the optimality of the outputs. While in our CFSNet, we perform the interpolation in the feature space and the continuous transition of the reconstruction effect is consistent with the variation of

the control variable $\alpha_{in}$. We can produce a better approximation of the unknown working point. See Sec. 4.3 for more experimental comparisons.

## 4. Experiments

In this section, we first demonstrate implementation details of our framework. Then we validate the control mechanism of our CFSNet. Finally, we apply our CFSNet to three classic tasks: image super-resolution, image denoising, JPEG image deblocking. All experiments validate the effectiveness of our model. Due to space limitations, more examples and analyses are provided in the appendix.

### 4.1. Implement Details

For image super-resolution task, our framework contains 30 main blocks and 30 tuning blocks (i.e., $M = 30$). As for the other two tasks, the main branch parameters of our CFSNet are kept similar to that of the compared method [37] (i.e., $M = 10$) for a fair comparison. Furthermore, we first generate a 512-dimensional vector with values of all 1. Then we multiply it by the control scalar $\alpha_{in}$ to produce a control input. All convolutional layers have 64 filters and the kernel size of each convolutional layer is $3 \times 3$. We use the method in [12] to perform weight initialization. For both training stage of all tasks, we use the ADAM optimizer [16] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$ with the initial learning rate 1e-4. We adopt 128 as the minibatch size in image denoising task and set it as 16 in the other three tasks. We use PyTorch to implement our network and perform all experiments on GTX 1080Ti GPU.

For image denoising and JPEG image deblocking, we follow the settings as in [37] and [8] respectively. The training loss function in Step 1 and Step 2 remains unchanged: $L_1(I_{rec}, I_g) = L_2(I_{rec}, I_g)$. In particular, for image denoising, we input the degraded images of noise level 25 when we train the main branch in Step 1 and we input the degraded images of noise level 50 when we train the tuning branch in Step 2. Training images are cut into $40 \times 40$ patches with a stride of 10. And the learning rate is reduced by 10 times every 50000 steps. For JPEG deblocking, we set quality factor as 10 in the first training stage and change it to 40 in the second training stage. Besides, we choose $48 \times 48$ as patch size and the learning rate is divided by 10 every 100000 steps. For image super-resolution, we first train the main branch with objective MAE loss, then we train the tuning branch with objective $L_2 = L_{mae} + 0.01L_{gan} + 0.01L_{per}$, where $L_{mae}$ denotes mean absolute error (MAE), $L_{gan}$ represents wgan-gp loss [10] and $L_{per}$ is a variant of perceptual loss [33]. We set HR patch size as $128 \times 128$ and we multiply the learning rate by 0.6 every 400000 steps.

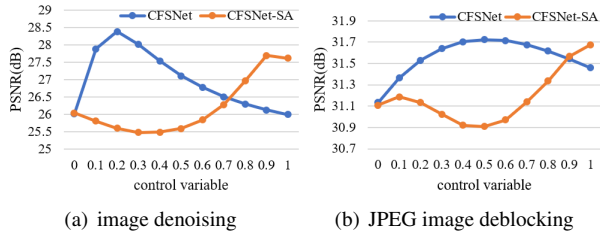(a) image denoising     (b) JPEG image deblocking

Figure 5. Average PSNR curve for noise level 30 on the BSD68 dataset and same curve for quality factor 20 on the LIVE1 dataset.

## 4.2. Ablation Study

Fig. 5 presents the ablation study on the effects of adaptive learning coupling coefficients strategy. We directly set the coupling coefficients of different channels and different layers as the same $\alpha_{in}$ in CFSNet-SA. That is, compared with CFSNet, CFSNet-SA removes the linear mapping network of control variable $\alpha_{in}$. Otherwise, we keep the training process of CFSNet-SA consistent with CFSNet. We can find that, no matter in denoising task or in deblocking task, the best restored result of CFSNet is better than that of CFSNet-SA for unseen degradation level. In particular, the curve of CFSNet is concave-shaped, which means that there is a bijective relationship between the reconstruction effect and the control variable. In contrast, there is no obvious change law in the curve of CFSNet-SA. The reason is that adaptive coupling coefficients help to produce better intermediate features. This merit provides more friendly interaction control. What's more, JPEG deblocking task is more robust to control variable than image denoising task, we speculate that this is because JPEG images of different degradation levels are closer in the latent space.

## 4.3. Image Super-resolution

For image super-resolution, we adopt a widely used DIV2K training dataset [1] that contains 800 images. We down-sample the high resolution image using MATLAB bicubic kernel with a scaling factor of 4. Following [24, 33], we evaluate our models on PIRM test dataset provided in the PIRM-SR Challenge [3]. We use the perception index (PI) to measure perceptual quality and use RMSE to measure distortion. Similar to the PIRM-SR challenge, we choose EDSR [18], CX [23] and EnhanceNet [26] as baseline methods. Furthermore, we also compare our CFSNet with another popular trade-off method, deep network interpolation [32, 33]. We directly use source code from ESR-GAN [33] to produce SR results with different perceptual quality, namely ESRGAN-I.

Fig. 7 and Fig. 1 show the visual comparison between our results and the baselines. We can observe that CFSNet can achieve a mild transition from low distortion results to high perceptual quality results without unpleasant artifacts.
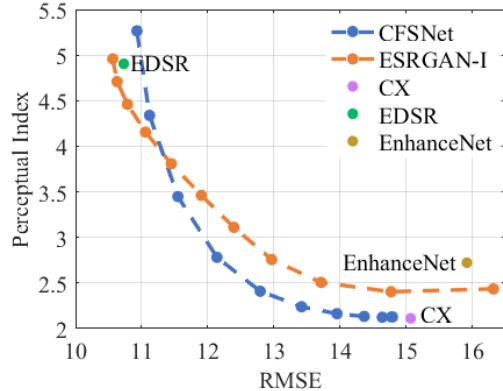


Figure 6. Perception-distortion plane on PIRM test dataset. We gradually increase $\alpha_{in}$ from 0 to 1 to generate different results from distortion point to perception point.

In addition, it can be found that our CFSNet outperforms the baselines on edges and shapes. Due to different user preferences, it is necessary to allow users to adjust the reconstruction results freely.

We also provide quantitative comparisons on PIRM test dataset. Fig. 6 shows the perception-distortion plane. As we can see, CFSNet improves the baseline (EnhanceNet) in both perceptual quality and reconstruction accuracy. The blue curve shows that our perception-distortion function is steeper than ESRGAN-I (orange curve). Meanwhile, CFSNet performs better than ESRGAN-I in most regions, although our network is lighter than ESRGAN. This means that our result is closer to the theoretical bound of perception-distortion.

## 4.4. Image Denoising

In image denoising experiments, we follow [37] to use 400 images from the Berkeley Segmentation Dataset (BSD) [22] as the training set. We test our model on BSD68 [22] using the mean PSNR as the quantitative metric. Both training set and test set are converted to gray images. We generate the degraded images by adding Gaussian noise of different levels (*e.g.*, 15, 25, 30, 40, and 50) to clean images.

We provide visual comparison in Fig. 8 and Fig. 1. As we can see, users can easily control $\alpha_{in}$ to balance noise reduction and detail preservation. It is worth noting that, our highest PSNR results ($\alpha_{in} = 0.5$) have similar visual quality with other methods, but it does not necessarily mean the best visual effects, for example, the sky patch of 'test017" enjoys a smoother result when $\alpha_{in} = 0.6$. Users can personalize each picture and choose their favorite results by controlling $\alpha_{in}$ at test-time.

In addition to perceptual comparisons, we also provide objective quantitative comparisons. We change $\alpha_{in}$ from 0 to 1 with an interval of 0.1 for preset noise range ($\sigma \in [25, 30, 40, 50]$). Then we choose the final result according
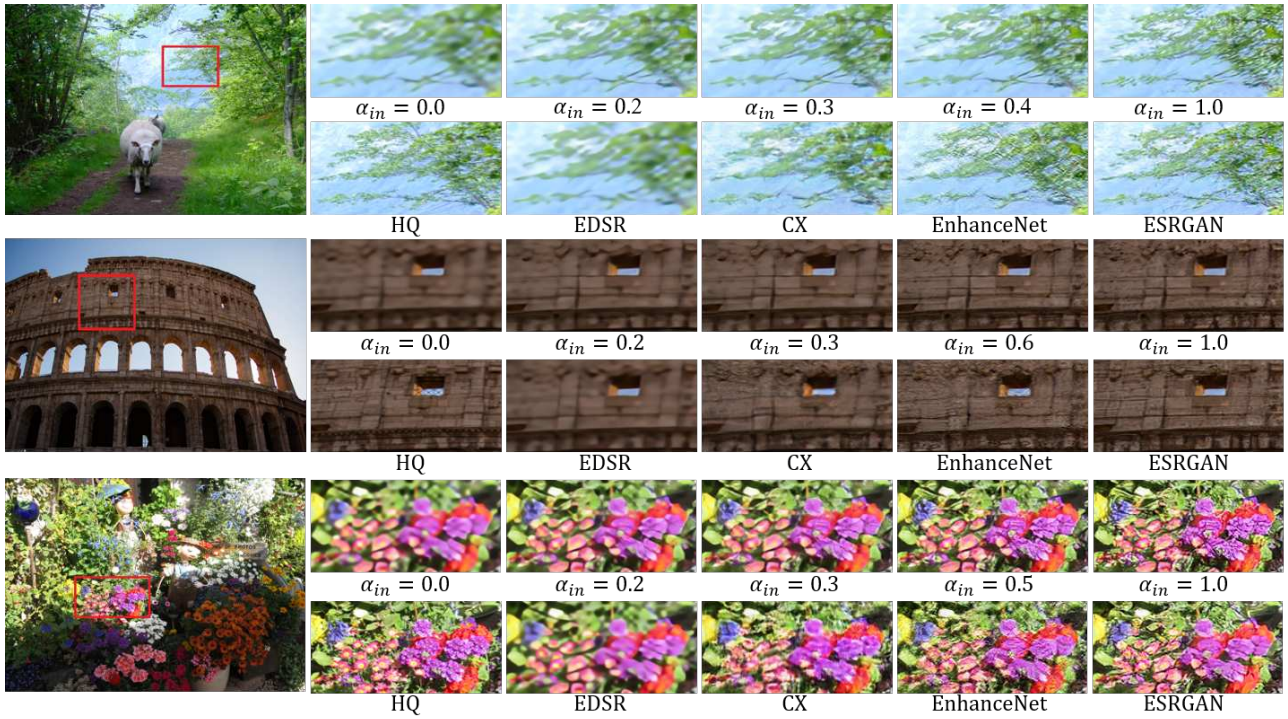
Figure 7. Perceptual and distortion balance of "215", "211" and "268" (PIRM test dataset) for $4\times$ image super-resolution.
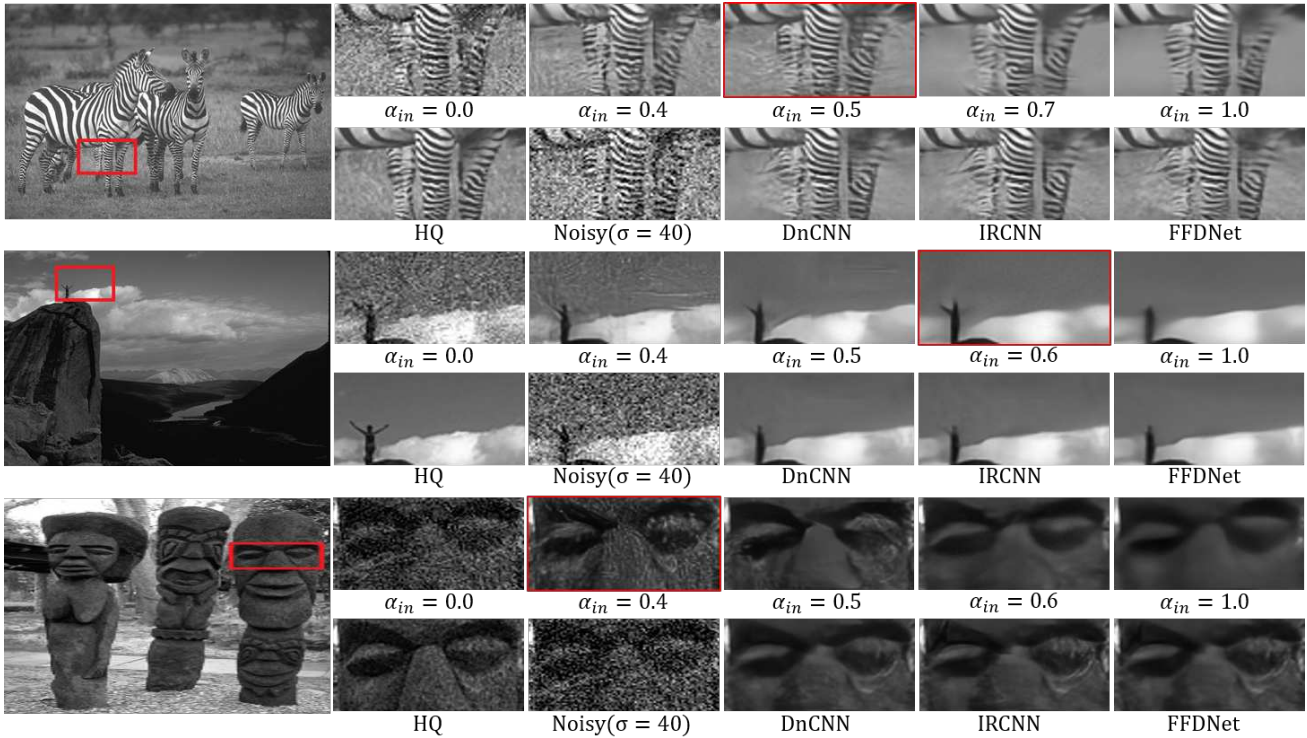


Figure 8. Gray image denoising results of "test051" "test017" and "test001" (BSD68) with unknown noise level $\sigma = 40$. $\alpha_{in} = 0.5$ corresponds to the highest PSNR results, and the best visual results are marked with red boxes.
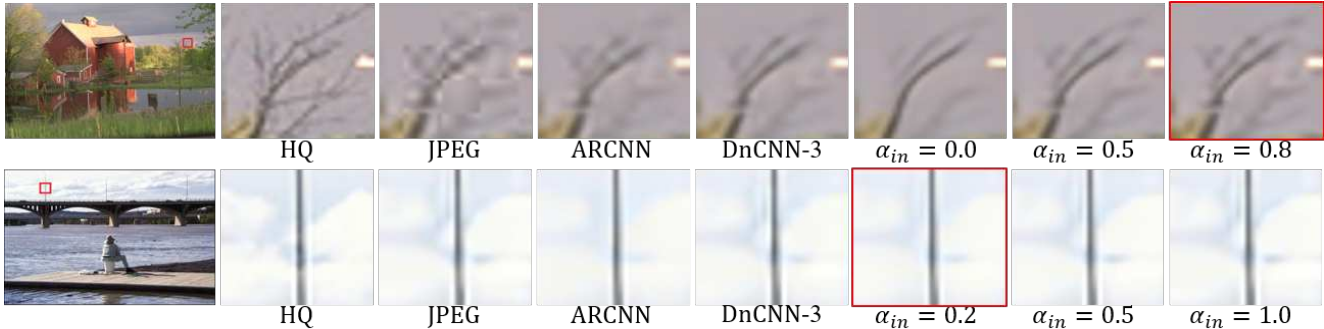
Figure 9. JPEG image artifacts removal results of "house" and "ocean" (LIVE1) with unknown quality factor 20. $\alpha_{in} = 0.5$ corresponds to the highest PSNR results, and the best visual results are marked with red boxes.

Table 1. Benchmark image denoising results. The average PSNR(dB) for various noise levels on (gray) BSD68. *denotes unseen noise levels for our CFSNet in the training stage.

| methods | $\sigma = 15^*$ | $\sigma = 25$ | $\sigma = 30^*$ | $\sigma = 50$ |
|---------|------|------|------|------|
| BM3D | 31.08 | 28.57 | 27.76 | 25.62 |
| TNRD | 31.42 | 28.92 | 27.66 | 25.97 |
| DnCNN-B | 31.61 | 29.16 | 28.36 | 26.23 |
| IRCNN | 31.63 | 29.15 | 28.26 | 26.19 |
| FFDNet | 31.63 | 29.19 | 28.39 | 26.29 |
| CFSNet | 31.29 | 29.24 | 28.39 | 26.28 |

Table 2. Benchmark JPEG deblocking results. The average PSNR(dB) on the LIVE1 dataset. * denotes unseen quality factors for our CFSNet in the training stage.

| methods | $q = 10$ | $q = 20^*$ | $q = 30^*$ | $q = 40$ |
|---------|------|------|------|------|
| JPEG | 27.77 | 30.07 | 31.41 | 32.35 |
| SA-DCT | 28.65 | 30.81 | 32.08 | 32.99 |
| ARCNN | 28.98 | 31.29 | 32.69 | 33.63 |
| TNRD | 29.15 | 31.46 | 32.84 | N/A |
| DnCNN-3 | 29.19 | 31.59 | 32.98 | 33.96 |
| CFSNet | 29.36 | 31.71 | 33.16 | 34.16 |

to the highest PSNR. We compare our CFSNet with several state-of-the-art denoising methods: BM3D [7], TNRD [6], DnCNN [37], IRCNN [38], FFDNet [39]. More interestingly, as shown in Tab. 1, our CFSNet is comparable with FFDNet on the endpoint ($\sigma = 25$ and $\sigma = 50$), but our CFSNet still achieves the best performance on the noise level 30 which is not contained in the training process. Moreover, our CFSNet can even deal with unseen outlier ($\sigma = 15$). This further verifies that we can obtain a good approximation of the unknown working point.

### 4.5. JPEG Image Deblocking

We also apply our framework to reduce image compression artifacts. As in [8, 20, 37], we adopt LIVE1 [25] as the test dataset and use the BSDS500 dataset [22] as base training set. For a fair comparison, we perform training and evaluating both on the luminance component of the YCbCr color space. We use the MATLAB JPEG encoder to generate JPEG deblocking input with four JPEG quality settings q = 10, 20, 30, 40.

We select the deblocking result in the same way as the image denoising task. We select SA-DCT [9], ARCNN [8], TNRD [6] and DnCNN [37] for comparisons. Tab. 2 shows the JPEG deblocking results on LIVE1. Our CFSNet achieves the best PSNR results on all compression quality factors. Especially, our CFSNet does not degrade too much and still achieves 0.12 dB and 0.18 dB improve-

ments over DnCNN-3 on quality 20 and 30 respectively, although JPEG images of quality 20 and 30 never appear in training process. Fig. 9 shows visual results of different methods on LIVE1. Too small $\alpha_{in}$ produces too smooth results, while too large $\alpha_{in}$ leads to incomplete artifacts elimination. Compared to ARCNN [8] and DnCNN [37], our CFSNet can make a better compromise between artifacts removal and details preservation.

## 5. Conclusion

In this paper, we introduce a new well-designed framework which equipped with flexible controllability for image restoration. The reconstruction results can be finely controlled using a single input variable with an adaptive learning strategy of coupling coefficients. Besides that, it is capable of producing high quality images on image restoration tasks such as image super-resolution, image blind denoising and image blind deblocking, and it outperforms the existing state-of-the-art methods in terms of user-control flexibility and visual quality. Future works will focus on the expansion of the multiple degraded image restoration tasks.

# References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017.

[2] Yoshua Bengio, Grégoire Mesnil, Yann Dauphin, and Salah Rifai. Better mixing via deep representations. In *International conference on machine learning*, pages 552–560, 2013.

[3] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *European Conference on Computer Vision*, pages 334–355. Springer, 2018.

[4] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018.

[5] Pratik Prabhanjan Brahma, Dapeng Wu, and Yiyuan She. Why deep learning works: A manifold disentanglement perspective. *IEEE transactions on neural networks and learning systems*, 27(10):1997–2008, 2016.

[6] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1256–1272, 2017.

[7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, Aug 2007.

[8] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 576–584, 2015.

[9] Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. *IEEE transactions on image processing*, 16(5):1395–1411, 2007.

[10] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

[11] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1712–1722, 2019.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[13] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[14] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[15] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[18] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017.

[19] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems*, pages 1680–1689, 2018.

[20] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 773–782, 2018.

[21] Yongyi Lu, Yu-Wing Tai, and Chi-Keung Tang. Attribute-guided face generation using conditional cyclegan. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 282–297, 2018.

[22] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *null*, page 416. IEEE, 2001.

[23] Roey Mechrez, Itamar Talmi, Firas Shama, and Lihi Zelnik-Manor. Maintaining natural image statistics with the contextual loss. In *Asian Conference on Computer Vision*, pages 427–443. Springer, 2018.

[24] Pablo Navarrete Michelini, Dan Zhu, and Hanwen Liu. Multi–scale recursive and perception–distortion controllable image super–resolution. In *European Conference on Computer Vision*, pages 3–19. Springer, 2018.

[25] Anush Krishna Moorthy and Alan Conrad Bovik. Visual importance pooling for image quality assessment. *IEEE journal of selected topics in signal processing*, 3(2):193–201, 2009.

[26] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4491–4500, 2017.

[27] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan

Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.

[28] Alon Shoshan, Roey Mechrez, and Lihi Zelnik-Manor. Dynamic-net: Tuning the objective without re-training. *arXiv preprint arXiv:1811.08760*, 2018.

[29] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE Conference on Computer vision and Pattern Recognition*, pages 3147–3155, 2017.

[30] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4539–4547, 2017.

[31] Paul Upchurch, Jacob Gardner, Geoff Pleiss, Robert Pless, Noah Snavely, Kavita Bala, and Kilian Weinberger. Deep feature interpolation for image content changes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7064–7073, 2017.

[32] Xintao Wang, Ke Yu, Chao Dong, Xiaoou Tang, and Chen Change Loy. Deep network interpolation for continuous imagery effect transition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1701, 2019.

[33] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *European Conference on Computer Vision*, pages 63–79. Springer, 2018.

[34] Kilian Q Weinberger and Lawrence K Saul. Unsupervised learning of image manifolds by semidefinite programming. *International journal of computer vision*, 70(1):77–90, 2006.

[35] Wenming Yang, Wei Wang, Xuechen Zhang, Shuifa Sun, and Qingmin Liao. Lightweight feature fusion network for single image super-resolution. *IEEE Signal Processing Letters*, 2019.

[36] Xin Yu, Basura Fernando, Richard Hartley, and Fatih Porikli. Super-resolving very low-resolution face images with supplementary attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 908–917, 2018.

[37] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[38] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3929–3938, 2017.

[39] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018.

[40] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the*

*European Conference on Computer Vision (ECCV)*, pages 286–301, 2018.

[41] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2472–2481, 2018.