

Stochastic Frequency Masking to Improve Super-Resolution and Denoising Networks

Majed El Helou*, Ruofan Zhou*, and Sabine Süsstrunk

School of Computer and Communication Sciences, EPFL, Switzerland
{majed.elhelou, ruofan.zhou, sabine.sustrunk}@epfl.ch

Abstract. Super-resolution and denoising are ill-posed yet fundamental image restoration tasks. In blind settings, the degradation kernel or the noise level are unknown. This makes restoration even more challenging, notably for learning-based methods, as they tend to overfit to the degradation seen during training. We present an analysis, in the frequency domain, of degradation-kernel overfitting in super-resolution and introduce a conditional learning perspective that extends to both super-resolution and denoising. Building on our formulation, we propose a stochastic frequency masking of images used in training to regularize the networks and address the overfitting problem. Our technique improves state-of-the-art methods on blind super-resolution with different synthetic kernels, real super-resolution, blind Gaussian denoising, and real-image denoising.

Keywords: Image Restoration, Super-Resolution, Denoising, Kernel Overfitting

1 Introduction

Image super-resolution (SR) and denoising are fundamental restoration tasks widely applied in imaging pipelines. They are crucial in various applications, such as medical imaging [33,38,45], low-light imaging [12], astronomy [7], satellite imaging [8,50], or face detection [24]. However, both are challenging ill-posed inverse problems. Recent learning methods based on convolutional neural networks (CNNs) achieve better restoration performance than classical approaches, both in SR and denoising. CNNs are trained on large datasets, sometimes real [65] but often synthetically generated with either one kernel or a limited set [54,63]. They learn to predict the restored image or the residual between the restored target and the input [27,56]. However, to be useful in practice, the networks should perform well on test images with unknown degradation kernels for SR, and unknown noise levels for denoising. Currently, they tend to overfit to the set of degradation models seen during training [16].

We investigate the SR degradation-kernel overfitting with an analysis in the frequency domain. Our analysis reveals that an implicit conditional learning is taking place in SR networks, namely, the learning of residual high-frequency

* The first two authors have similar contributions.

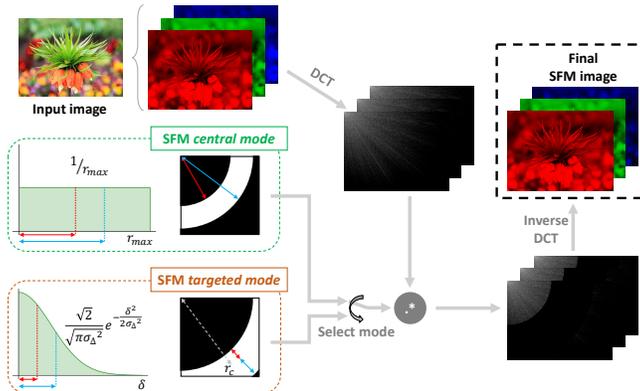


Fig. 1: Overview of Stochastic Frequency Masking (SFM). In the *central mode*, two radii values are sampled uniformly to delimit a masking area, and in the *targeted mode*, the sampled values delimit a quarter-annulus away from a target frequency. The obtained mask, shown with inverted color, is applied channel-wise to the discrete cosine transform of the image. We invert back to the spatial domain to obtain the SFM image that we use to train SR and denoising networks.

content given low frequencies. We additionally show that this result extends to denoising as well. Building on our insights, we present **Stochastic Frequency Masking** (SFM), which stochastically masks frequency components of the *images used in training*. Our SFM method (Fig. 1) is applied to a subset of the training images to regularize the network. It encourages the conditional learning to improve SR and denoising networks, notably when training under the challenging blind conditions. It can be applied during the training of *any* learning method, and has no additional cost at test time.

Our experimental results show that SFM improves the performance of state-of-the-art networks on blind SR and blind denoising. For SR, we conduct experiments on synthetic bicubic and Gaussian degradation kernels, and on real degraded images. For denoising, we conduct experiments on additive white Gaussian denoising and on real microscopy Poisson-Gaussian image denoising. SFM improves the performance of state-of-the-art networks on each of these tasks.

Our contributions are summarized as follows. We present a frequency-domain analysis of the degradation-kernel overfitting of SR networks, and highlight the implicit conditional learning that, as we also show, extends to denoising. We present a novel technique, SFM, that regularizes the learning of SR and denoising networks by only filtering the training data. It allows the networks to better restore frequency components and avoid overfitting. We empirically show that SFM improves the results of state-of-the-art learning methods on blind SR with different synthetic degradations, real-image SR, blind Gaussian denoising, and real-image denoising on high noise levels.

Code available at: <https://github.com/majedelhelou/SFM>

2 Related work

Super-resolution. Depending on their image prior, SR algorithms can be divided into prediction models [43], edge-based models [11], gradient-profile prior methods [48] and example-based methods [21]. Deep example-based SR networks hold the state-of-the-art performance. Zhang *et al.* propose a very deep architecture based on residual channel attention to further improve these networks [63]. It is also possible to train in the wavelet domain to improve the memory and time efficiency of the networks [64]. Perceptual loss [25] and GANs [30,54] are leveraged to mitigate blur and push the SR networks to produce more visually-pleasing results. However, these networks are trained using a limited set of kernels, and studies have shown that they have poor generalization to unseen degradation kernels [22,46]. To address blind SR, which is degradation-agnostic, recent methods propose to incorporate the degradation parameters including the blur kernel into the network [46,57,59,60]. However, these methods rely on blur-kernel estimation algorithms and thus have a limited ability to handle arbitrary blur kernels. The most recent methods, namely IKC [22] and KMSR [65], propose kernel estimation and modeling in their SR pipeline. However, it is hard to gather enough training kernels to cover the real-kernel manifold, while also ensuring effective learning and avoiding that these networks overfit to the chosen kernels. Recently, real-image datasets were proposed [10,61] to enable SR networks to be trained and tested on high- and low-resolution (HR-LR) pairs, which capture the same scene but at different focal lengths. These datasets are also limited to the degradations of only a few cameras and cannot guarantee that SR models trained on them would generalize to unseen degradations. Our SFM method, which builds on our degradation-kernel overfitting analysis and our conditional learning perspective, can be used to improve the performance of *all* the SR networks we evaluate, including ones that estimate and model degradation kernels.

Denoising. Classical denoisers such as PURE-LET [34], which is specifically aimed at Poisson-Gaussian denoising, KSVD [2], WNNM [23], BM3D [14], and EPLL [66], which are designed for Gaussian denoising, have the limitation that the noise level needs to be known at test time, or at least estimated [20]. Recent learning-based denoisers outperform the classical ones on Gaussian denoising [4,39,56], but require the noise level [58], or pre-train multiple models for different noise levels [31,57], or more recently attempt to predict the noise level internally [18]. For a model to work under blind settings and adapt to any noise level, a common approach is to train the denoiser network while varying the training noise level [4,39,56]. Other recent methods, aimed at real-image denoising such as microscopy imaging [62], learn image statistics without requiring ground-truth samples on which noise is synthesized. This is practical because ground-truth data can be extremely difficult and costly to acquire in, for instance, medical applications. Noise2Noise [32] learns to denoise from pairs of noisy images. The noise is assumed to be zero in expectation and decorrelated from the signal. Therefore, unless the network memorizes it, the noise would not be predicted by it, and thus gets removed [32,53]. Noise2Self [6], which is a similar but more general version of Noise2Void [29], also assumes the noise to

be decorrelated, conditioned on the signal. The network learns from single noisy images, by learning to predict an image subset from a separate subset, again with the assumption that the noise is zero in expectation. Although promising, these two methods do not yet reach the performance of Noise2Noise. By regularizing the conditional learning defined from our frequency-domain perspective, our SFM method improves the high noise level results of *all* tested denoising networks, notably under blind settings.

One example that uses frequency bands in restoration is the method in [5] that defines a prior based on a distance metric between a test image and a dataset of same-class images used for a deblurring optimization. The distance metric computes differences between image frequency bands. In contrast, we apply frequency masking on training images to regularize deep learning restoration networks, improving performance and generalization. Spectral dropout [26] regularizes network activations by dropping out components in the frequency domain to remove the least relevant, while SFM regularizes training by promoting the conditional prediction of different frequency components through masking the training images themselves. The most related work to ours is a recent method proposed in the field of speech recognition [37]. The authors augment speech data in three ways, one of which is in the frequency domain. It is a random separation of frequency bands, which splits different speech components to allow the network to learn them one by one. A clear distinction with our approach is that we do not aim to separate input components to be each individually learned. Rather, we mask targeted frequencies from the *training* input to strengthen the conditional frequency learning, and indirectly simulate the effect of a variety of kernels in SR and noise levels in denoising. The method we present is, to the best of our knowledge, the first frequency-based input *masking* method to regularize SR and denoising training.

3 Frequency perspective on SR and denoising

3.1 Super-resolution

Preliminaries Downsampling, a key element in modeling SR degradation, can be well explained in the frequency domain where it is represented by the sum of shifted and stretched versions of the frequency spectrum of a signal. Let x be a one-dimensional discrete signal, e.g., a pixel row in an image, and let z be a downsampled version of x with a sampling interval T . In the discrete-time Fourier transform domain, with frequencies $\omega \in [-\pi, \pi]$, the relation between the transforms X and Z of the signals x and z , respectively, is given by $Z(\omega) = \frac{1}{T} \sum_{k=0}^{T-1} X((\omega + 2\pi k)/T)$. The T replicas of X can overlap in the high frequencies and cause aliasing. Aside from complicating the inverse problem of restoring x from z , aliasing can create visual distortions. Before downsampling, low-pass filtering is therefore applied to attenuate if not completely remove the high-frequency components that would otherwise overlap.

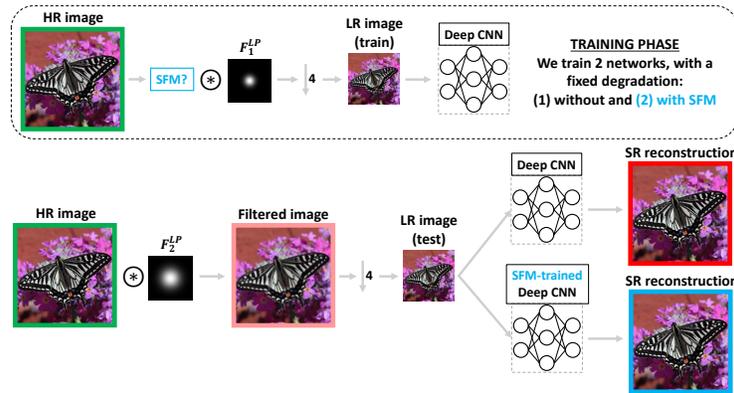
These low-pass filtering blur kernels are applied through a spatial convolution over the image. The set of real kernels spans only a subspace of all

mathematically-possible kernels. This subspace is, however, not well-defined analytically and, in the literature, is often limited to the non-comprehensive subspace spanned by 2D Gaussian kernels. Many SR methods thus model the anti-aliasing filter as a 2D Gaussian kernel, attempting to mimic the point spread function (PSF) of capturing devices [15,44,55]. In practice, even a single imaging device results in multiple kernels, depending on its settings [17]. For real images, the kernel can also be different from a Gaussian kernel [16,22]. The essential point is that the anti-aliasing filter causes the loss of high-frequency components, and that this filter can differ from image to image.

Frequency visualization of SR reconstructions SR networks tend to overfit to the blur kernels used in the degradation for obtaining the training images [59]. To understand that phenomenon, we analyze in this section the relation between the frequency-domain effect of a blur kernel and the reconstruction of SR networks. We carry out the following experiment with a network trained with a unique and known blur kernel. We use the DIV2K [1] dataset to train a 20-block RRDB [54] x4 SR network with images filtered by a Gaussian blur kernel called F_1^{LP} (standard deviation $\sigma = 4.1$), shown in the top row of Fig. 2(a). We then run an inference on 100 test images filtered with a different Gaussian blur kernel called F_2^{LP} ($\sigma = 7.4$), shown in the bottom row of Fig. 2(a), to analyze the potential network overfitting.

We present a frequency-domain visualization in Fig. 2(b). The power spectral density (PSD) is the distribution of frequency content in an image. The typical PSD of an image (green curve) is modeled as $1/f^\alpha$, where f is the spatial frequency, with $\alpha \in [1, 2]$ and varying depending on the scene (natural vs. man-made) [9,19,51,52]. The $1/f^\alpha$ trend is visible in the PSD of HR images (green fill). The degraded LR test images are obtained with a low-pass filter on the HR image, before downsampling, and their frequency components are mostly low frequencies (pink fill). The SR network outputs contain high-frequency components restored by the network (red fill). However, these frequencies are mainly above 0.2π . This is only the range that was filtered out by the kernel used in creating the *training* LR images. The low-pass kernel used in creating the test LR images filters out a larger range of frequencies; it has a lower cutoff than the training kernel (the reverse case is also problematic and is illustrated in *Supplementary Material*). This causes a gap of missing frequency components not obtained in the restored SR output, illustrated with a blue dashed circle in Fig. 2(b). The results suggest that an implicit conditional learning takes place in the SR network, on which we expand further in the following section. The results of the network trained with 50% SFM (masking applied to half of the training set) are shown in Fig. 2(c). A key observation is that the missing frequency components are predicted to a far better extent when the network is trained with SFM.

Implicit conditional learning As we explain in the Preliminaries of Sec. 3.1, the high-frequency components of the original HR images are removed by the anti-aliasing filter. If that filter is *ideal*, it means that the low-frequency com-



(a) Experimental setup

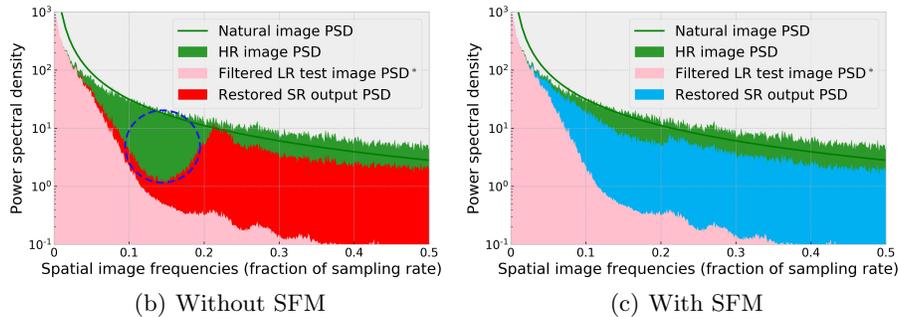


Fig. 2: (a) Overview of our experimental setup, with image border colors corresponding to the plot colors shown in (b,c). We train 2 versions of the same network on the same degradation kernel (F_1^{LP} anti-aliasing filter), one without and one with SFM, and test them using F_2^{LP} . (b) Average PSD (power spectral density) of HR images in green fill, with a green curve illustrating a typical natural-image PSD ($\alpha = 1.5$ [52]). The pink fill illustrates the average PSD of the low-pass filtered LR test images (*shown before downsampling for better visualization). In red fill is the average PSD of the restored SR output image. The blue dashed circle highlights the learning gap due to degradation-kernel overfitting. (c) The same as (b), except that the output is that of the network trained with SFM. Results are averaged over 100 random samples.

ponents are not affected and the high frequencies are perfectly removed. We propose that the SR networks in fact learn implicitly a conditional probability

$$P(I^{HR} \otimes F^{HP} \mid I^{HR} \otimes F^{LP}), \quad (1)$$

where F^{HP} and F^{LP} are ideal high-pass and low-pass filters, applied to the high-resolution image I^{HR} , and \otimes is the convolution operator. The low and high frequency ranges are theoretically defined as $[0, \pi/T]$ and $[\pi/T, \pi]$, which is the

minimum condition (largest possible cutoff) to avoid aliasing for a downsampling rate T . The components of I^{HR} that survive the low-pass filtering are the same frequencies contained in the LR image I^{LR} , when the filters F are ideal. In other words, the frequency components of $I^{HR} \circledast F^{LP}$ are those remaining in the LR image that is the network input.

The anti-aliasing filters are, in practice, not ideal, resulting in: (a) some low-frequency components of I^{HR} being attenuated, (b) some high frequencies surviving the filtering and causing aliasing. Typically, the main issue is the first issue (a), because filters are chosen in a way to remove the visually-disturbing aliasing at the expense of attenuating some low frequencies. We expand further on this in *Supplementary Material*, and derive that even with non-ideal filters, there is still conditional and residual learning components to predict a set of high-frequencies. These frequencies are, however, conditioned on a set of low-frequency components potentially attenuated by the non-ideal filter we call F_o^{LP} . This filter fully removes aliasing artifacts but can affect the low frequencies. The distribution can hence be defined by the components

$$P(I^{HR} \circledast F^{HP} | I^{HR} \circledast F_o^{LP}), \quad P(I^{HR} \circledast F^{LP} - I^{HR} \circledast F_o^{LP} | I^{HR} \circledast F_o^{LP}). \quad (2)$$

This is supported by our results in Fig. 2. The SR network trained with degradation kernel F_1^{LP} ($\sigma = 4.1$ in our experiment) restores the missing high frequencies of I^{HR} that would be erased by F_1^{LP} . However, that is the case even though the test image is degraded by $F_2^{LP} \neq F_1^{LP}$. As F_2^{LP} ($\sigma = 7.4$) removes a wider range of frequencies than F_1^{LP} , not predicted by the network, these frequencies remain missing. We observe a gap in the PSD of the output, highlighted by a blue dashed circle. This illustrates the degradation-kernel overfitting issue from a frequency-domain perspective. We also note that these missing frequency components are restored by the network trained with SFM.

3.2 Extension to denoising

We highlight a connection between our conditional learning proposition and denoising. As discussed in Sec. 3.1, the average PSD of an image can be approximated by $1/f^\alpha$. The Gaussian noise samples added across pixels are independent and identically distributed. The PSD of the additive white Gaussian noise is uniform. Fig. 3 shows the PSD of a natural image following a power law with $\alpha = 2$, that of white Gaussian noise (WGN), and the resulting signal-to-noise ratio (SNR) when the WGN is added to the image. The resulting SNR decreases proportionally to $1/f^\alpha$.

The relation between SNR and frequency shows that with increasing frequency, the SNR becomes exponentially small. In other words, high frequencies are almost completely overtaken by the noise, while low frequencies are much less affected by it. And, *the higher the noise level, the lower the starting frequency beyond which the SNR is significantly small*, as illustrated by Fig. 3. This draws a direct connection to our SR analysis. Indeed, in both applications there exists an implicit conditional learning to predict lost high-frequency components given low-frequency ones that are less affected.

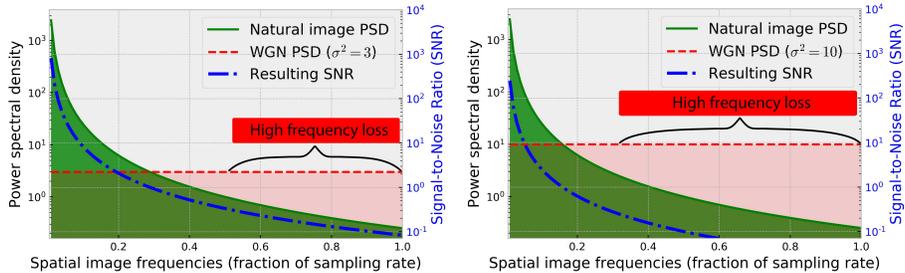


Fig. 3: Natural image PSD follows a power law as a function of spatial frequency. The plotted examples follow a power law with $\alpha = 2$ [52] and additive WGN ($\sigma^2 = 3$ on the left, and $\sigma^2 = 10$ on the right). The resulting SNR in the noisy image is exponentially smaller the higher the frequency, effectively causing a high frequency loss. The higher the noise level, the more frequency loss is incurred, and the more similar denoising becomes to our SR formulation.

4 Stochastic Frequency Masking (SFM)

4.1 Motivation and implementation

The objective of SFM is to improve the networks’ prediction of high frequencies given lower ones, whether for SR or denoising. We achieve this by stochastically masking high-frequency bands from some of the training images in the learning phase, to encourage the conditional learning of the network. Our masking is carried out by transforming an image to the frequency domain using the Discrete Cosine Transform (DCT) type II [3,47], multiplying channel-wise by our stochastic mask, and lastly transforming the image back (Fig. 1). See *Supplementary Material* for the implementation details of the DCT type we use. We define frequency bands in the DCT domain over quarter-annulus areas, to cluster together similar-magnitude frequency content. Therefore, the SFM mask is delimited with a quarter-annulus area by setting the values of its inner and outer radii. We define two masking modes, the *central mode* and the *targeted mode*.

In the *central mode*, the inner and outer radius limits r_I and r_O of the quarter-annulus are selected uniformly at random from $[0, r_M]$, where $r_M = \sqrt{a^2 + b^2}$ is the maximum radius, with (a, b) being the dimensions of the image. We ensure that $r_I < r_O$ by permuting the values if $r_I > r_O$. With this mode, the resulting probability of a given frequency band r_ω to be masked is

$$P(r_\omega = 0) = P(r_I < r_\omega < r_O) = 2 \left(\frac{r_\omega}{r_M} - \left(\frac{r_\omega}{r_M} \right)^2 \right), \quad (3)$$

meaning the central bands are the more likely ones to be masked, with the likelihood *slowly* decreasing for higher or lower frequencies. In the *targeted mode*, a target frequency r_C is selected, with a parameter σ_Δ . The quarter-annulus is delimited by $[r_C - \delta_I, r_C + \delta_O]$, where δ_I and δ_O are independently sampled from

the half-normal Δ distribution $f_{\Delta}(\delta) = \sqrt{2}/\sqrt{\pi\sigma_{\Delta}^2}e^{-\delta^2/(2\sigma_{\Delta}^2)}$, $\forall\delta \geq 0$. Therefore, with this mode, the frequency r_C is always masked, and the frequencies away from r_C are less and less likely to be masked, with a normal distribution decay.

We use the *central mode* for SR networks, and the *targeted mode* with a high target r_C for denoisers (Fig. 1). The former has a slow probability decay that covers wider bands, while the latter has an exponential decay adapted for targeting specific narrow bands. In both settings, the highest frequencies are most likely to be masked. The *central mode* masks the highest frequencies in SR, because central frequencies are the highest ones remaining after the anti-aliasing filter is applied. It is also worth noting that SFM thus simulates the effect of different blur kernels by stochastically masking different frequency bands.

4.2 Learning SR and denoising with SFM

We apply SFM *only* on the input training data. For the simulated-degradation data, SFM is applied in the process of generating the LR inputs. We apply SFM on HR images before applying the degradation model to generate the LR inputs (blur kernel and downsampling). The target output of the network remains the original HR images. For real images where the LR inputs are given and the degradation model is unknown, we apply SFM on the LR inputs and keep the original HR images as ground-truth targets. Therefore, the networks trained with SFM do not use any additional data relative to the baselines. We apply *the same* SFM settings for all deep learning experiments. During training, we apply SFM on 50% of the training images, using the *central mode* of SFM, as presented in Sec. 4.1. Ablation studies with other rates are in our *Supplementary Material*. We add SFM to the training of the original methods with no other modification.

When training for additive white Gaussian noise (AWGN) removal, we apply SFM on the clean image before the synthetic noise is added. When the training images are real and the noise cannot be separated from the signal, we apply SFM on the noisy image. Hence, we ensure that networks trained with SFM do not utilize any additional training data relative to the baselines. In all denoising experiments, and for all of the compared methods, we use *the same* SFM settings. We apply SFM on 50% of training images, and use the *targeted mode* of our SFM (ablation studies including other rates are in our *Supplementary Material*). We use a central band $r_C = 0.85 r_M$ and $\sigma_{\Delta} = 0.15 r_M$. As presented in Sec. 4.1, this means that the highest frequency bands are masked with high likelihood, and lower frequencies are exponentially less likely to be masked the smaller they are. We add SFM to the training of the original methods with no other modification.

5 Experiments

5.1 SR: bicubic and Gaussian degradations

Methods. We evaluate our proposed SFM method on state-of-the-art SR networks that can be divided into 3 categories. In the first category, we evaluate

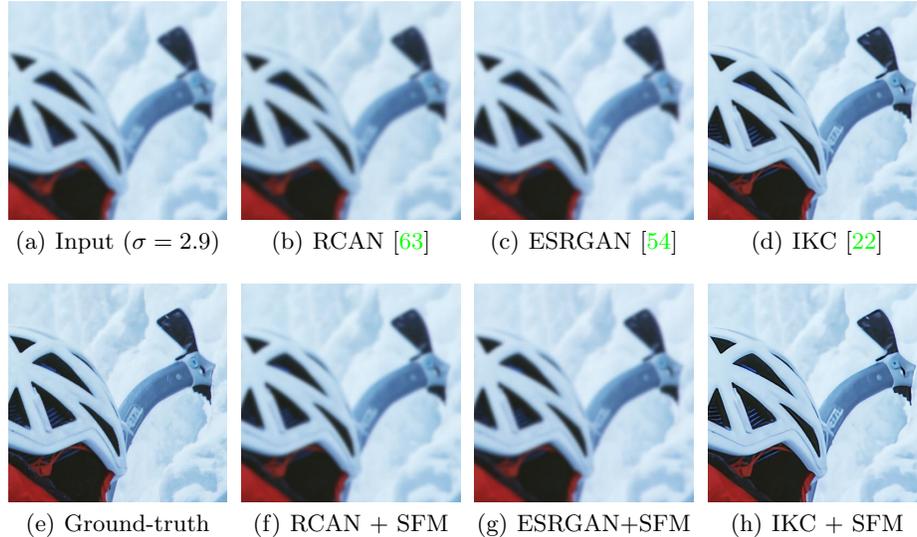


Fig. 4: Cropped SR results (x4) of different methods (top row), and with our SFM added (bottom row), for image 0844 of DIV2K. The visual quality improves for all methods when trained with SFM (images best viewed on screen).

	Test blur kernel (g_σ is a Gaussian kernel, standard deviation σ)									
	bicubic	$g_{1.7}$	$g_{2.3}$	$g_{2.9}$	$g_{3.5}$	$g_{4.1}$	$g_{4.7}$	$g_{5.3}$	$g_{5.9}$	$g_{6.5}$
RCAN [63]	29.18	23.80	24.08	23.76	23.35	22.98	22.38	22.16	21.86	21.72
RCAN+SFM	29.32	24.21	24.64	24.19	23.72	23.27	22.54	22.23	21.91	21.79
IKC [22]	27.81	26.07	26.15	25.48	25.03	24.41	23.39	22.78	22.41	22.08
IKC+SFM	27.78	26.09	26.18	25.52	25.11	24.52	23.54	22.97	22.62	22.35
RRDB [54]	28.79	23.66	23.72	23.68	23.29	22.75	22.32	22.08	21.83	21.40
RRDB+SFM	29.10	23.81	23.99	23.79	23.41	22.90	22.53	22.37	21.98	21.56
ESRGAN [54]	25.43	21.22	22.49	22.03	21.87	21.63	21.21	20.99	20.05	19.42
ESRGAN+SFM	25.50	21.37	22.78	22.26	22.08	21.80	21.33	21.10	20.13	19.77

Table 1: Single-image SR, with x4 upscaling factor, PSNR (dB) results on the DIV2K validation set. RCAN, RRDB and ESRGAN are trained using bicubic degradation, and IKC using Gaussian kernels ($\sigma \in [2.0, 4.0]$). Kernels seen in training are shaded gray. The training setups of the networks are presented in Sec. 5.1, and identical ones are used with SFM. We note that SFM improves the results of the various methods, even the IKC method that explicitly models kernels during its training improves by up to $0.27dB$ with SFM on unseen kernels.

RCAN [63] and RRDB [54], which are networks that target pixel-wise distortion for a single degradation kernel. RCAN leverages a residual-in-residual structure and channel attention for efficient non-blind SR learning. RRDB [54] employs a residual-in-residual dense block as its basic architecture unit. The second cate-

Method	Dataset and upscaling factor				
	RealSR [10]			SR-RAW [61]	
	x2	x3	x4	x4	x8
RCAN [‡] [63]	33.24	30.24	28.65	26.29	24.18
RCAN 50% SFM	33.32	30.29	28.75	26.42	24.50
KMSR [65]	32.98	30.05	28.27	25.91	24.00
KMSR 50% SFM	33.21	30.11	28.50	26.19	24.31
IKC [22]	33.07	30.03	28.29	25.87	24.19
IKC 50% SFM	33.12	30.25	28.42	25.93	24.25

Table 2: PSNR (dB) results of blind image super-resolution on two real SR datasets, for the different available upscaling factors. [‡]RCAN is trained on the paired dataset collected from the same sensor as the testing dataset.

gory covers perception-optimized methods for a single degradation kernel, and includes ESRGAN [54]. It is a version of the RRDB network using a GAN for better SR perceptual quality and obtains the state-of-the-art results in this category. The last category includes algorithms for blind SR, we experiment on IKC [22], which incorporates into the training of the SR network a blur-kernel estimation and modeling to explicitly address blind SR. **Setup.** We train all the models using the DIV2K [1] dataset, which is a high-quality dataset that is commonly used for single-image SR evaluation. RCAN, RRDB, and ESRGAN are trained with the bicubic degradation, and IKC with Gaussian kernels ($\sigma \in [0.2, 4.0]$ [22]). For all models, 16 LR patches of size 48×48 are extracted per training batch. All models are trained using the Adam optimizer [28] for 50 epochs. The initial learning rate is set to 10^{-4} and decreases by half every 10 epochs. Data augmentation is performed on the training images, which are randomly rotated by 90° , 180° , 270° , and flipped horizontally. **Results.** To generate test LR images, we apply bicubic and Gaussian blur kernels on the DIV2K [1] validation set. We also evaluate all methods trained with 50% SFM, following Sec. 4.2. Table 1 shows the PSNR results on x4 upscaling SR, with different blur kernels. Results show that the proposed SFM consistently improves the performance of the various SR networks on the different degradation kernels, even up to $0.27dB$ on an unseen test kernel for the recent IKC [22] that explicitly models kernels during training. We improve by up to $0.56dB$ for the other methods. With SFM, RRDB achieves comparable or better results than RCAN, which has double the parameters of RRDB. Sample visual results are shown in Fig. 4.

5.2 SR: real-image degradations

Methods. We train and evaluate the same SR models as the networks we use in Sec. 5.1, except for ESRGAN and RRDB, as ESRGAN is a perceptual-quality-driven method and does not achieve high PSNR, and RCAN outperforms RRDB according to our experiments in 5.1. We also evaluate on KMSR [65] for the real SR experiments. KMSR collects real blur kernels from real LR images to improve the generalization of the SR network on unseen kernels. **Setup.** We

	Test noise level (standard deviation of the stationary AWGN)									
	10	20	30	40	50	60	70	80	90	100
DnCNN-B [56]	33.33	29.71	27.66	26.13	24.88	23.69	22.06	19.86	17.88	16.35
DnCNN-B + SFM	33.35	29.78	27.73	26.27	25.09	24.02	22.80	21.24	19.46	17.87
Noise2Noise [32]	32.67	28.84	26.61	25.02	23.76	22.69	21.74	20.88	20.11	19.41
Noise2Noise + SFM	32.55	28.94	26.84	25.31	24.11	23.05	22.14	21.32	20.61	19.95
Blind [‡] N3Net [39]	33.53	30.01	27.84	26.30	25.04	23.93	22.87	21.84	20.87	19.98
N3Net + SFM	33.41	29.86	27.84	26.38	25.19	24.15	23.20	22.32	21.51	20.78
Blind [‡] MemNet [49]	33.51	29.75	27.61	26.06	24.87	23.83	22.67	21.00	18.92	17.16
MemNet + SFM	33.36	29.80	27.76	26.31	25.14	24.09	23.09	22.00	20.77	19.46
RIDNet [4]	33.65	29.87	27.65	26.04	24.79	23.65	22.25	20.05	18.15	17.09
RIDNet + SFM	33.43	29.81	27.76	26.30	25.12	24.08	23.11	22.08	20.74	19.17

Table 3: PSNR (dB) results on BSD68 for different methods and noise levels. SFM improves the various methods, and the improvement increases with higher noise levels, supporting our hypothesis. We clamp test images to $[0,255]$ as in camera pipelines. Denoisers are trained with levels up to 55 (shaded in gray), thus half the test range is not seen in training. [‡]Re-trained under blind settings.

train and evaluate the SR networks on two digital zoom datasets: the SR-RAW dataset [61] and the RealSR dataset [10]. The training setup of the SR networks is the same as in Sec. 5.1. Note that we follow the same training procedures for each method as in the original papers. IKC is trained with Gaussian kernels ($\sigma \in [0.2, 4.0]$) and KMSR with the blur kernels estimated from LR images in the dataset. RCAN is trained *on the degradation of the test data*; a starting advantage over other methods. **Results.** We evaluate the SR methods on the corresponding datasets and present the results in Table 2. Each method is also trained with 50% SFM, following Sec. 4.2. SFM consistently improves all methods on all upscaling factors, pushing the state-of-the-art results by up to $0.23dB$ on both of these challenging real-image SR datasets.

5.3 Denoising: AWGN

Methods. We evaluate different state-of-the-art AWGN denoisers. DnCNN-B [56] learns the noise residual rather than the final denoised image. Noise2Noise (N2N) [32] learns only from noisy image pairs, with no ground-truth data. N3Net [39] relies on learning nearest neighbors similarity, to make use of different similar patches in an image for denoising. MemNet [49] follows residual learning with memory transition blocks. Lastly, RIDNet [4] also does residual learning, but leverages feature attention blocks. **Setup.** We train all methods on the 400 Berkeley images [36], typically used to benchmark denoisers [13,42,56]. All methods use the Adam optimizer with a starting learning rate of 10^{-3} , except RIDNet that uses half that rate. We train for 50 epochs and synthesize noise instances per training batch. For blind denoising training, we follow the settings initially set in [56]: noise is sampled from a Gaussian distribution with standard deviation chosen at random in $[0, 55]$. This splits the range of test noise levels

into levels seen or not seen during training, which provides further insights on generalization. We also note that we use a U-Net [40] for the architecture of N2N as in the original paper. For N2N, we apply SFM on top of the added noise, to preserve the particularity that N2N can be trained without ground-truth data. **Results.** We evaluate all methods on the BSD68 [41] test set. Each method is also trained with 50% SFM as explained in Sec. 4.2 and the results are in Table 3. SFM improves the performance of a variety of different state-of-the-art denoising methods on high noise levels (seen during training, such as 40 and 50, or not even seen), and the results support our hypothesis presented in Sec. 3.2 that *the higher the noise level the more similar is denoising to SR and the more applicable is SFM*. Indeed, the higher the noise level the larger the improvement of SFM, and this trend is true across all methods. Fig. 5 presents sample results.

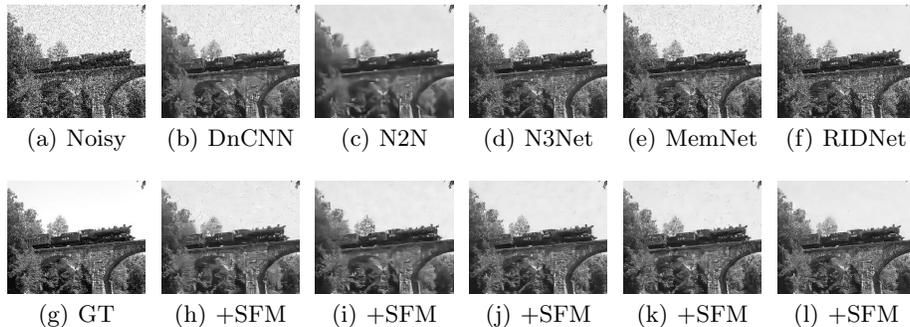


Fig. 5: Denoising ($\sigma = 50$) results with different methods (top row), and with our SFM added (bottom row), for the last image (#67) of the BSD68 benchmark.

5.4 Denoising: real Poisson-Gaussian images

Methods. Classical methods are often a good choice for denoising in the absence of ground-truth datasets. PURE-LET [34] is specifically aimed at Poisson-Gaussian denoising, and KSVD [2], WNNM [23], BM3D [14], and EPLL [66] are designed for Gaussian denoising. Recently, learning methods were presented such as N2S [6] (and the similar, but less general, N2V [29]) that can learn from a dataset of only noisy images, and N2N [32] that can learn from a dataset of only noisy image pairs. We incorporate SFM into the learning-based methods. **Setup.** We train the learning-based methods on the recent real fluorescence microscopy dataset [62]. The noise follows a Poisson-Gaussian distribution, and the image registration is of high quality due to the stability of the microscopes, thus yielding reliable ground-truth obtained by averaging 50 repeated captures. Noise parameters are estimated using the fitting approach in [20] for all classical denoisers. Additionally, the parameters are used for the variance-stabilization

Method	# raw images for averaging									
	Mixed test set [62]					Two-photon test set [62]				
	16	8	4	2	1	16	8	4	2	1
PURE-LET [34]	39.59	37.25	35.29	33.49	31.95	37.06	34.66	33.50	32.61	31.89
VST+KSVD [2]	40.36	37.79	35.84	33.69	32.02	38.01	35.31	34.02	32.95	31.91
VST+WNNM [23]	40.45	37.95	36.04	34.04	32.52	38.03	35.41	34.19	33.24	32.35
VST+BM3D [14]	40.61	38.01	36.05	34.09	32.71	38.24	35.49	34.25	33.33	32.48
VST+EPLL [66]	40.83	38.12	36.08	34.07	32.61	38.55	35.66	34.35	33.37	32.45
N2S [6]	36.67	35.47	34.66	33.15	31.87	34.88	33.48	32.66	31.81	30.51
N2S 50% SFM	36.60	35.62	34.59	33.44	32.40	34.39	33.14	32.48	31.84	30.92
N2N [32]	41.45	39.43	37.59	36.40	35.40	38.37	35.82	34.56	33.58	32.70
N2N 50% SFM	41.48	39.46	37.78	36.43	35.50	38.78	36.10	34.85	33.90	33.05

Table 4: PSNR (dB) results on microscopy images with Poisson-Gaussian noise. We train under blind settings and apply SFM on noisy input images to preserve the fact that N2S and N2N can be trained without clean images.

transform (VST) [35] for the Gaussian-oriented methods. In contrast, the learning methods can directly be applied under blind settings. We train N2S/N2N using a U-Net [40] architecture, for 100/400 epochs using the Adam optimizer with a starting learning rate of $10^{-5}/10^{-4}$ [62]. **Results.** We evaluate on the mixed and two-photon microscopy test sets [62]. We also train the learning methods with 50% SFM as explained in Sec. 4.2, and present the results in Table 4. A larger number of averaged raw images is equivalent to a lower noise level. N2N with SFM achieves the state-of-the-art performance on both benchmarks and for all noise levels, with an improvement of up to $0.42dB$. We also note that the improvements of SFM are larger on the more challenging two-photon test set where the noise levels are higher on average. SFM does not consistently improve N2S, however, this is expected. In fact, unlike other methods, N2S trains to predict a subset of an image given a surrounding subset. It applies spatial masking where the mask is made up of random pixels that interferes with the frequency components. For these reasons, N2S is not very compatible with SFM, which nonetheless improves results on the largest noise levels in both test sets.

6 Conclusion

We analyze the degradation-kernel overfitting of SR networks in the frequency domain. Our frequency-domain insights reveal an implicit conditional learning that also extends to denoising, especially on high noise levels. Building on our analysis, we present SFM, a technique to improve SR and denoising networks, without increasing the size of the training set or any cost at test time. We conduct extensive experiments on state-of-the-art networks for both restoration tasks. We evaluate SR with synthetic degradations, real-image SR, Gaussian denoising and real-image Poisson-Gaussian denoising, showing improved performance, notably on generalization, when using SFM.

References

1. Agustsson, E., Timofte, R.: NTIRE 2017 challenge on single image super-resolution: Dataset and study. In: CVPR Workshops (2017) [5](#), [11](#)
2. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing over-complete dictionaries for sparse representation. *IEEE Transactions on Signal Processing* **54**(11), 4311–4322 (2006) [3](#), [13](#), [14](#)
3. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. *IEEE Transactions on Computers* **100**(1), 90–93 (1974) [8](#)
4. Anwar, S., Barnes, N.: Real image denoising with feature attention. *ICCV* (2019) [3](#), [12](#)
5. Anwar, S., Huynh, C.P., Porikli, F.: Image deblurring with a class-specific prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **41**(9), 2112–2130 (2018) [4](#)
6. Batson, J., Royer, L.: Noise2Self: Blind denoising by self-supervision. In: *ICML* (2019) [3](#), [13](#), [14](#)
7. Beckouche, S., Starck, J.L., Fadili, J.: Astronomical image denoising using dictionary learning. *Astronomy & Astrophysics* **556**, A132 (2013) [1](#)
8. Benazza-Benyahia, A., Pesquet, J.C.: Building robust wavelet estimators for multi-component images using Stein’s principle. *IEEE Transactions on Image Processing* **14**(11), 1814–1830 (2005) [1](#)
9. Burton, G.J., Moorhead, I.R.: Color and spatial structure in natural scenes. *Applied Optics* **26**(1), 157–170 (1987) [5](#)
10. Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: A new benchmark and a new model. In: *ICCV* (2019) [3](#), [11](#), [12](#)
11. Chan, T.M., Zhang, J., Pu, J., Huang, H.: Neighbor embedding based super-resolution algorithm through edge detection and feature selection. *Pattern Recognition Letters* (2009) [3](#)
12. Chatterjee, P., Joshi, N., Kang, S.B., Matsushita, Y.: Noise suppression in low-light images through joint denoising and demosaicing. In: *CVPR* (2011) [1](#)
13. Chen, Y., Pock, T.: 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 (2016) [12](#)
14. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing* **16**(8), 2080–2095 (2007) [3](#), [13](#), [14](#)
15. Dong, W., Zhang, L., Shi, G., Li, X.: Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing* **22**(4), 1620–1630 (2012) [5](#)
16. Efrat, N., Glasner, D., Apartsin, A., Nadler, B., Levin, A.: Accurate blur models vs. image priors in single image super-resolution. In: *ICCV* (2013) [1](#), [5](#)
17. El Helou, M., Dümbgen, F., Süsstrunk, S.: AAM: An assessment metric of axial chromatic aberration. In: *ICIP* (2018) [5](#)
18. El Helou, M., Süsstrunk, S.: Blind universal Bayesian image denoising with Gaussian noise level learning. *IEEE Transactions on Image Processing* **29**, 4885–4897 (2020) [3](#)
19. Field, D.J.: Relations between the statistics of natural images and the response properties of cortical cells. *JOSA* **4**(12), 2379–2394 (1987) [5](#)
20. Foi, A., Trimeche, M., Katkovnik, V., Egiazarian, K.: Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing* **17**(10), 1737–1754 (2008) [3](#), [13](#)

21. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. *Computer Graphics and Applications* (2002) [3](#)
22. Gu, J., Lu, H., Zuo, W.Z., Dong, C.: Blind super-resolution with iterative kernel correction. In: *CVPR* (2019) [3](#), [5](#), [10](#), [11](#)
23. Gu, S., Zhang, L., Zuo, W., Feng, X.: Weighted nuclear norm minimization with application to image denoising. In: *CVPR* (2014) [3](#), [13](#), [14](#)
24. Gunturk, B.K., Batur, A.U., Altunbasak, Y., Hayes, M.H., Mersereau, R.M.: Eigenface-domain super-resolution for face recognition. *IEEE Transactions on Image Processing* **12**(5), 597–606 (2003) [1](#)
25. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: *ECCV* (2016) [3](#)
26. Khan, S.H., Hayat, M., Porikli, F.: Regularization of deep neural networks with spectral dropout. *Neural Networks* **110**, 82–90 (2019) [4](#)
27. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: *CVPR* (2016) [1](#)
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014) [11](#)
29. Krull, A., Buchholz, T.O., Jug, F.: Noise2Void-learning denoising from single noisy images. In: *CVPR* (2019) [3](#), [13](#)
30. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: *CVPR* (2017) [3](#)
31. Lefkimmiatis, S.: Universal denoising networks: A novel CNN architecture for image denoising. In: *CVPR* (2018) [3](#)
32. Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., Aila, T.: Noise2Noise: Learning image restoration without clean data. In: *ICML* (2018) [3](#), [12](#), [13](#), [14](#)
33. Li, S., Yin, H., Fang, L.: Group-sparse representation with dictionary learning for medical image denoising and fusion. *IEEE Transactions on Biomedical Engineering* **59**(12), 3450–3459 (2012) [1](#)
34. Luisier, F., Blu, T., Unser, M.: Image denoising in mixed Poisson-Gaussian noise. *IEEE Transactions on Image Processing* **20**(3), 696–708 (2011) [3](#), [13](#), [14](#)
35. Makitalo, M., Foi, A.: Optimal inversion of the generalized Anscombe transformation for Poisson-Gaussian noise. *IEEE Transactions on Image Processing* **22**(1), 91–103 (2012) [14](#)
36. 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. In: *ICCV* (2001) [12](#)
37. Park, D.S., Chan, W., Zhang, Y., Chiu, C.C., Zoph, B., Cubuk, E.D., Le, Q.V.: SpecAugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779* (2019) [4](#)
38. Peled, S., Yeshurun, Y.: Superresolution in MRI: application to human white matter fiber tract visualization by diffusion tensor imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* **45**(1), 29–35 (2001) [1](#)
39. Plötz, T., Roth, S.: Neural nearest neighbors networks. In: *NeurIPS* (2018) [3](#), [12](#)
40. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. pp. 234–241 (2015) [13](#), [14](#)
41. Roth, S., Black, M.J.: Fields of experts. *International Journal of Computer Vision* **82**(2), 205 (2009) [13](#)

42. Schmidt, U., Roth, S.: Shrinkage fields for effective image restoration. In: CVPR (2014) [12](#)
43. Schuler, S., Leistner, C., Bischof, H.: Fast and accurate image upscaling with super-resolution forests. In: CVPR (2015) [3](#)
44. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR (2016) [5](#)
45. Shi, W., Caballero, J., Ledig, C., Zhuang, X., Bai, W., Bhatia, K., de Marvao, A.M.S.M., Dawes, T., O'Regan, D., Rueckert, D.: Cardiac image super-resolution with global correspondence using multi-atlas patchmatch. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI). pp. 9–16 (2013) [1](#)
46. Shocher, A., Cohen, N., Irani, M.: "zero-shot" super-resolution using deep internal learning. In: CVPR (2018) [3](#)
47. Strang, G.: The discrete cosine transform. SIAM **41**(1), 135–147 (1999) [8](#)
48. Sun, J., Xu, Z., Shum, H.Y.: Image super-resolution using gradient profile prior. In: CVPR (2008) [3](#)
49. Tai, Y., Yang, J., Liu, X., Xu, C.: MemNet: A persistent memory network for image restoration. In: ICCV (2017) [12](#)
50. Thornton, M.W., Atkinson, P.M., Holland, D.: Sub-pixel mapping of rural land cover objects from fine spatial resolution satellite sensor imagery using super-resolution pixel-swapping. International Journal of Remote Sensing **27**(3), 473–491 (2006) [1](#)
51. Tolhurst, D., Tadmor, Y., Chao, T.: Amplitude spectra of natural images. Ophthalmic and Physiological Optics **12**(2), 229–232 (1992) [5](#)
52. Torralba, A., Oliva, A.: Statistics of natural image categories. Network: Computation in Neural Systems **14**(3), 391–412 (2003) [5](#), [6](#), [8](#)
53. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: CVPR (2018) [3](#)
54. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Loy, C.C.: ESRGAN: Enhanced super-resolution generative adversarial networks. In: ECCV Workshops (2018) [1](#), [3](#), [5](#), [10](#), [11](#)
55. Yang, C.Y., Ma, C., Yang, M.H.: Single-image super-resolution: A benchmark. In: ECCV (2014) [5](#)
56. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Transactions on Image Processing **26**(7), 3142–3155 (2017) [1](#), [3](#), [12](#)
57. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep CNN denoiser prior for image restoration. In: CVPR (2017) [3](#)
58. Zhang, K., Zuo, W., Zhang, L.: FFDNet: Toward a fast and flexible solution for CNN-based image denoising. IEEE Transactions on Image Processing **27**(9), 4608–4622 (2018) [3](#)
59. Zhang, K., Zuo, W., Zhang, L.: Learning a single convolutional super-resolution network for multiple degradations. In: CVPR (2018) [3](#), [5](#)
60. Zhang, K., Zuo, W., Zhang, L.: Deep plug-and-play super-resolution for arbitrary blur kernels. In: CVPR (2019) [3](#)
61. Zhang, X., Chen, Q., Ng, R., Koltu, V.: Zoom to learn, learn to zoom. In: ICCV (2019) [3](#), [11](#), [12](#)
62. Zhang, Y., Zhu, Y., Nichols, E., Wang, Q., Zhang, S., Smith, C., Howard, S.: A Poisson-Gaussian denoising dataset with real fluorescence microscopy images. In: CVPR (2019) [3](#), [13](#), [14](#)
63. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: ECCV (2018) [1](#), [3](#), [10](#), [11](#)

64. Zhou, R., Lahoud, F., El Helou, M., Süsstrunk, S.: A comparative study on wavelets and residuals in deep super resolution. In: Electronic Imaging (2019) **3**
65. Zhou, R., Süsstrunk, S.: Kernel modeling super-resolution on real low-resolution images. In: ICCV (2019) **1, 3, 11**
66. Zoran, D., Weiss, Y.: From learning models of natural image patches to whole image restoration. In: ICCV (2011) **3, 13, 14**