SiamTrans: Zero-Shot Multi-Frame Image Restoration with Pre-trained Siamese Transformers

Lin Liu[1], Shanxin Yuan[2*], Jianzhuang Liu[2], Xin Guo[1], Youliang Yan[2], Qi Tian[3]

[1]EEIS Department, University of Science and Technology of China
[2]Huawei Noah's Ark Lab
[3]Huawei Cloud BU
{ll0825,willing}mail.ustc.edu.cn {shanxin.yuan, liu.jianzhuang, yanyouliang, tian.qi1}@huawei.com

Abstract

We propose a novel zero-shot multi-frame image restoration method for removing unwanted obstruction elements (such as rains, snow, and moiré patterns) that vary in successive frames. It has three stages: transformer pre-training, zero-shot restoration, and hard patch refinement. Using the pre-trained transformers, our model is able to tell the motion difference between the true image information and the obstructing elements. For zero-shot image restoration, we design a novel model, termed SiamTrans, which is constructed by Siamese transformers, encoders, and decoders. Each transformer has a temporal attention layer and several self-attention layers, to capture both temporal and spatial information of multiple frames. Only pre-trained (self-supervised) on the denoising task, SiamTrans is tested on three different low-level vision tasks (deraining, demoiréing, and desnowing). Compared with related methods, ours achieves the best performances, even outperforming those with supervised learning.

Introduction

Taking clean photographs under bad weather (*e.g.*, snow and rain) or recovering clean images from occluding elements (*e.g.*, moiré patterns), is challenging as the scene information is corrupted by these occluding elements in the captured images. These occluding elements can change quickly in a very short time (*e.g.*, snow and rain) or due to the small movement of the camera (*e.g.*, moiré patterns), making it difficult to do multi-frame image restoration.

Recovering the underlying clean image from a single degraded image is an ill-posed problem due to occlusions. Most of existing single image restoration methods that focus on dealing with these types of problem often mine high-level semantic information of the scene or the properties of degrading elements (*e.g.*, noise and rain streak). But these single image restoration methods have difficulty in handling challenging and complex cases. To tackle these problems, multi-frame-based approaches, which use several images as input, are proposed to exploit additional information from supporting frames to the reference frame. Task-specific multi-frame-based methods include denoising (Liang et al. 2020; Mildenhall et al. 2018; Godard, Matzen, and Uyttendaele 2018), demosaicing (Ehret et al. 2019; Kokkinos and Lefkimmiatis 2019), super-resolution (Farsiu et al. 2014; El Mourabit et al. 2017; Wronski et al. 2019; Isobe et al. 2020), reflection removal (Li and Brown 2013; Guo, Cao, and Ma 2014), HDR imaging (Dai et al. 2021; Yan et al. 2019), *etc*. In addition, a few general frameworks have been proposed for multiple low-level vision tasks (Xue et al. 2015; Alayrac 2019; Liu et al. 2020d; Fan et al. 2020). The work in (Alayrac 2019) proposes a generic 3D CNN to estimate the foreground layer and (Liu et al. 2020d) estimates the optical flows of obstruction elements. Both methods cannot obtain satisfactory results because it is hard to estimate either obstruction elements or their optical flows that vary dramatically in successive frames. These supervised methods require a large amount of annotated training data and often have problems when applied to new tasks.

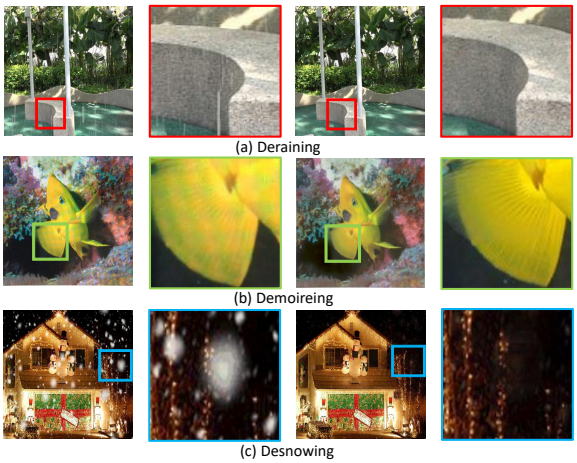

Figure 1: Example results of our zero-shot method for multi-frame deraining (top), demoiréing (middle), and desnowing (bottom). Pre-trained on denoising only, our SiamTrans model removes unwanted elements while retaining the image details on multiple restoration tasks.

The pre-training and fine-tuning strategy is effective to obtain the natural image prior and adapt to new tasks. This strategy is often used on high-level vision tasks (Chen et al. 2020b; Grill et al. 2020; He et al. 2020b) showing great per-

*Corresponding author

formances, but it has not been widely used in low-level vision tasks yet. Recently, some studies (Gu, Shen, and Zhou 2020; Pan et al. 2020; Chan et al. 2012; Bau et al. 2020) use pre-trained GANs to do image restoration, where the GAN models are trained on large-scale natural images and can capture rich texture and shape priors. But the spatial information may not be faithfully kept due to low dimensionality of the latent code. Transformer (Vaswani et al. 2017) has been used in some low-level vision tasks very recently, such as image super-resolution (Yang et al. 2020a), video synthesis (Liu et al. 2020e) and video inpainting (Zeng, Fu, and Chao 2020). Image processing transformer (IPT) (Chen et al. 2020a) is pre-trained on three low-level vision tasks and outperforms state-of-the-art methods. It shows that the transformer is more advantageous than convolutional neural networks (CNNs) in large-scale data pre-training of low-level vision. However, IPT requires that the pre-training and fine-tuning are conducted on the same tasks, having difficulty in generalizing to completely unseen tasks, *e.g.*, desnowing. Our model is a zero-shot learning setting, where the pre-training is only conducted on image denoising and the testing (without fine-tuning) is conducted on new tasks.

In this paper, inspired by the advantages of the multi-frame methods and pre-trained transformers, we propose a three-stage pipeline for multi-frame image restoration. This pipeline contains transformer pre-training, zero-shot restoration, and hard patch refinement. In the first stage, the transformer is self-supervisedly pre-trained on the denoising task on a large scale dataset. The pre-training enables the transformer to learn natural image prior information between different frames and to converge fast in downstream iterations. In the second stage, we design a model with Siamese Transformers (SiamTrans) for multiple downstream low-level tasks through zero-shot restoration. Note that the downstream tasks are unknown to the pre-training. SiamTrans consists of encoders, decoders, temporal attention modules, and self-attention modules. The aim of the third stage is to locate and refine hard-case patches.

In summary, we make the following contributions:

- A three-stage pipeline for multi-frame image restoration is proposed. It consists of transformer pre-training, zero-shot restoration, and hard patch refinement.
- We design a novel model with Siamese Transformers (SiamTrans) for zero-shot image restoration. Using pre-trained transformers with temporal and spatial attentions, our model is able to tell the motion difference between the nature image information and the obstructing elements.
- When tested on three different low-level vision tasks (deraining, demoiréing, and desnowing; see Fig. 1), our model achieves the best performances, even outperforming supervised learning methods.

Related Work

In this section, we present the most related works, including multi-frame image restoration, transformers and pre-training for low-level vision, and related tasks.

Multi-frame image restoration. Image restoration is an ill-posed problem and most of single image restoration methods (Wang et al. 2018; Liu et al. 2019; Pan et al. 2020; Zheng et al. 2020; Chen, Liu, and Wang 2020) often resort to high-level semantic information of the scenes or the degrading properties (*e.g.*, noise level and rain streak). But these single image restoration methods have difficulty in handling challenging and complex cases. To deal with these issues, multi-frame-based methods (Godard, Matzen, and Uyttendaele 2018; Ehret et al. 2019; Farsiu et al. 2014) have been proposed for low-level vision tasks, such as denoising (Liang et al. 2020; Mildenhall et al. 2018; Godard, Matzen, and Uyttendaele 2018), demosaicing (Ehret et al. 2019; Kokkinos and Lefkimmiatis 2019), super-resolution (Farsiu et al. 2014; El Mourabit et al. 2017; Wronski et al. 2019) and reflection removal (Li and Brown 2013; Guo, Cao, and Ma 2014). In addition to these task-specific multi-frame methods, a few general frameworks have been proposed for multiple tasks (Xue et al. 2015; Alayrac 2019; Liu et al. 2020d). Xue *et al.* (Xue et al. 2015) present a computational approach for obstruction removal, which is applicable to multiple tasks, *e.g.*, reflection removal and fence removal. Liu *et al.* (Liu et al. 2020d) estimate dense optical flow fields of the background and degrading element layers and then reconstruct them. Our work is an unsupervised zero-shot multi-frame image restoration method that can be applied to multiple low-level vision tasks.

Transformers for low-level vision tasks. Transformers (Vaswani et al. 2017) are a neural network framework using the self-attention mechanism. They are originally used in natural language processing, and then used in computer vision tasks including low-level vision very recently (Yang et al. 2020a; Chen et al. 2020a; Zeng, Fu, and Chao 2020). Yang *et al.* (Yang et al. 2020a) propose a texture transformer network for image super-resolution. It transfers relevant textures from reference images to low-resolution images. Chen *et al.* (Chen et al. 2020a) develop a pre-trained transformer called IPT for three low-level vision tasks, which outperforms state-of-the-art methods. In low-level video processing, Liu *et al.* (Liu et al. 2020e) propose ConvTransformer to synthesize video frames. Zeng *et al.* (Zeng, Fu, and Chao 2020) propose a spatial-temporal transformer network for video inpainting, where frames with holes are taken as input and the holes are filled simultaneously.

Pre-training for low-level vision tasks. The pre-training and fine-tuning strategy is often used on high-level vision tasks, showing good performances (Chen et al. 2020b; Grill et al. 2020; He et al. 2020b). For low-level vision tasks, the random initialization and end-to-end training strategy is usually adopted. More recently, some studies (Gu, Shen, and Zhou 2020; Pan et al. 2020; Chan et al. 2012; Bau et al. 2020) use pre-trained GANs to do image restoration. A GAN model trained on a large-scale set of natural images can capture rich texture and shape priors. The problem of using pre-trained GANs for image restoration is that some spatial details may not be faithfully recovered due to the low dimensionality of the latent code, resulting in artifacts compared with ground-truth. Recently, IPT (Chen et al. 2020a) shows that transformers are more advantageous than

CNNs in large-scale data pre-training for low-level vision. IPT uses task-specific embeddings as an additional input for the decoder, where the task-specific embeddings are learned to decode features for different tasks. Different from IPT, our SiamTrans is pre-trained for zero-shot image restoration. We only need to pre-train it on the denoising task and then apply it to multiple downstream low-level tasks which are unknown to the pre-training.

Deraining, desnowing, and demoiréing. Deraining methods can be grouped into single image derainging and video deraining. The former group focuses on mining the intrinsic prior of the rain signal (Fu et al. 2017; Yang et al. 2017; Deng et al. 2018; Li et al. 2018; Guo et al. 2020). Compared with single-image rain removal, video deraining utilizes temporal information to detect and remove rains (Jiang et al. 2019; Li et al. 2019; Yang et al. 2020c; Li et al. 2021). We take advantage of transformer's ability to acquire natural image prior and temporal information after training on a large scale dataset.

Compared with rain, snow is more complicated due to its large variations of size and shape, and its transparency property. Snow removal methods also include single image desnowing (Chen et al. 2020c; Jaw, Huang, and Kuo 2020; Liu et al. 2018) and video desnowing (Ren et al. 2017; Li et al. 2019). For single image desnowing, Liu *et al.* (Liu et al. 2018) propose a learning based model. Chen *et al.* (Chen et al. 2020c) design a desnowing network which contains three parts: snow removal, veiling effect removal, and clean image discriminator. For video desnowing, Ren *et al.* (Ren et al. 2017) use matrix decomposition to desnow.

Moiré artifacts are not unusual in digital photography, especially when photos are taken of digital screens. Moiré patterns are mainly caused by the interference between the screen's subpixel layout and the camera's color filter array. Recently, some deep learning models (He et al. 2020a; Zheng et al. 2020; Yang et al. 2020b; He et al. 2019; Liu et al. 2020a,b; Zheng et al. 2021; Yuan et al. 2019b,a, 2020) are proposed for single image demoiréing. For multi-frame demoiréing, Liu *et al.* (Liu et al. 2020c) use multiple images as inputs and design multi-scale feature encoding modules to enhance low-frequency information. Unlike their approach, our method is unsupervised and does not need to train with a large number of moiré and moiré-free image pairs. We only need pre-training on denoising and then do zero-shot restoration with multiple moiré frames.

Proposed Method

In this section, we first introduce our basic network architecture and then present the three stages of our method, including transformer pre-training, zero-shot restoration with SiamTrans, and hard patch refinement. This basic network is the building block of our SiamTrans.

Basic Network Architecture

As shown in Fig. 2, our basic network includes two weight-sharing CNN encoders each corresponding to an input, and a CNN decoder to generate the final output. Between the encoders and the decoder, we construct a transformer that has a temporal attention module and six spatial self-attention modules.

Temporal attention module. In multi-frame image restoration, information from supporting frames can help to restore the corrupted reference frame. Given the input images x_1 and x_k ($k \in \{2, \ldots, N\}$), we present the process of the encoding as:

$$y_1 = E(x_1), \;\; y_2 = E(x_k), \tag{1}$$

where $y_1, \; y_2 \in \mathbb{R}^{C \times H \times W}$ denote the output feature maps of the encoders, C is the number of feature channels, H and W are the height and width of the feature map, respectively.

After feature extraction, the obtained feature maps, y_j, $j \in \{1, 2\}$ is reshaped into a sequence of flattened patches (vectors), $y_p^j = \{y_{p_1}^j, y_{p_2}^j, \ldots, y_{p_m}^j\}$, where $y_{p_i}^j \in \mathbb{R}^{CP^2}, i \in \{1, \ldots, m\}$; $m = \frac{HW}{P^2}$ is the total number of patches and $P \times P$ is the patch size. The process of the temporal attention module is formulated as:

$$z_0 = (\mathrm{MHA}\left(\mathrm{NL}(y_p^1), \mathrm{NL}(y_p^2), \mathrm{NL}(y_p^2)\right) + y_p^1, \tag{2}$$

$$z_1 = \mathrm{FF}\left(\mathrm{NL}(z_0)\right) + z_0, \tag{3}$$

where $\mathrm{MHA}(Q, K, V)$ denotes the multi-head attention module with $Q = \mathrm{NL}(y_p^1)$, $K = \mathrm{NL}(y_p^2)$, and $V = \mathrm{NL}(y_p^2)$ corresponding to the three basic transformer elements Query, Key, and Value, respectively, NL denotes the operation of the normalization layer, and FF is a feed forward network (Dosovitskiy et al. 2021; Vaswani et al. 2017).

Self-attention module & decoder. After the fusion of two corresponding frames by the temporal attention module, the self-attention module uses the self-attention mechanism to extract useful spatial information from z_i. In our work, six self-attention modules are employed, each with a multi-head self-attention layer and a feed forward network. The process is represented as:

$$z_i' = (\mathrm{MHA}\left(\mathrm{NL}(z_i), \mathrm{NL}(z_i), \mathrm{NL}(z_i)\right) + z_i, i = 1, 2, ..., 6, \tag{4}$$

$$z_{i+1} = \mathrm{FF}\left(\mathrm{NL}(z_i')\right) + z_i, i = 1, 2, ..., 6. \tag{5}$$

Finally the output z_7 is reshaped to $g \in \mathbb{R}^{C \times H \times W}$. The output of the decoder is $o = D(g) \in \mathbb{R}^{3 \times H \times W}$.

Transformer Pre-Training

We pre-train the basic network in Fig. 2 such that it can capture the intrinsic properties and transformations of various images, *i.e.*, image prior. The pre-training task is denoising with the Place365 dataset (Zhou et al. 2017). We choose 328,000 images for the pre-training. In each iteration, we randomly choose an image I and a noise level $\sigma \in [1, 50]$ and synthesize two degraded images by:

$$x_1 = I + \mathcal{N}_1, \;\; x_2 = I + \mathcal{N}_2, \tag{6}$$

where $\mathcal{N}_1$ and $\mathcal{N}_2$ are two different samples from the Gaussian noise distribution $\mathcal{N}(0, \sigma)$. The loss function in the pre-training stage is,

$$L_{\text{pre-train}} = \left\| M(x_1, x_2) - I \right\|_1, \tag{7}$$

where M denotes the basic network. After the pre-training, the network learns the feature correlation of different frames and the natural image prior. The ablation study in Sec. shows that the pre-training cannot only make SiamTrans converge faster but also improve its performance.

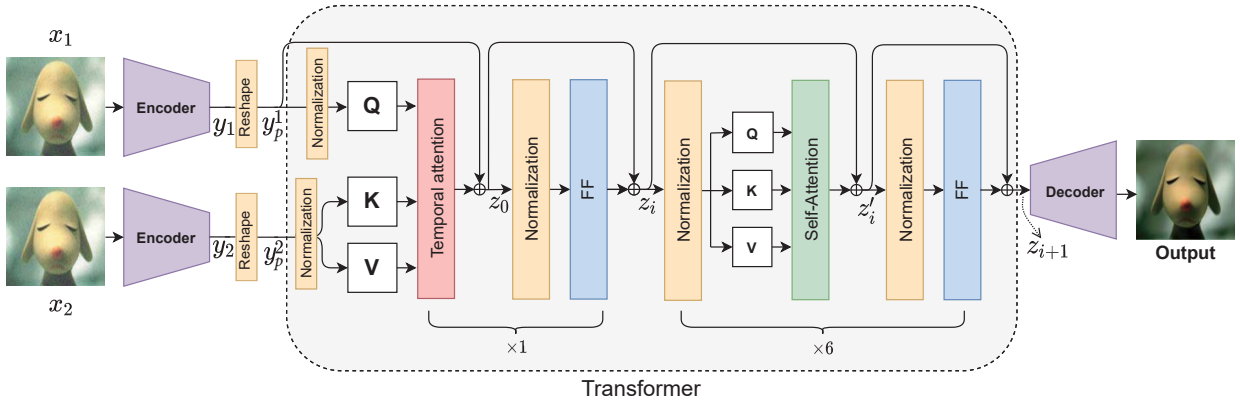

Figure 2: The architecture of our basic network, which consists of three parts: two CNN encoders, a transformer with both temporal and spatial attention modules, and a CNN decoder. Note that we also add learnable position embeddings (Dosovitskiy et al. 2021) to the input sequence of the transformer, which are not displayed here.

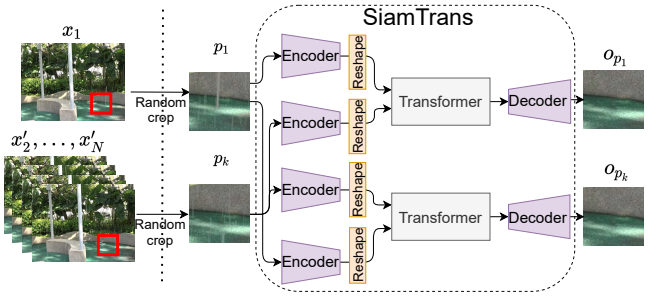

Figure 3: The architecture of our SiamTrans for zero-shot restoration. It includes Siamese transformers, four encoders and two decoders. $x_2^{'}, ..., x_N^{'}$ are obtained by warping $x_2, ..., x_N$ towards x_1. Two same-location patches p_1 and $p_k,\ k \in \{2, 3, ..., N\}$, are randomly cropped and served as the inputs to SiamTrans.

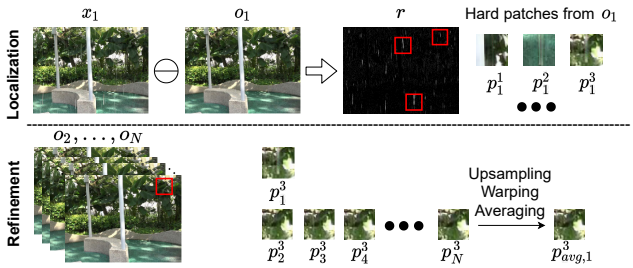

Figure 4: Procedure of our hard patch refinement, where one example with patch p_1^3 is given.

SiamTrans for Zero-Shot Restoration

The SiamTrans model is shown in Fig. 3, which is formed by two basic networks, with four weight-sharing CNN encoders, two weight-sharing transformers and two weight-sharing CNN decoders. Suppose we have a short sequence of images $\{x_1, x_2, ..., x_N\}$, where x_1 is the reference frame. The task of multi-frame image restoration is to recover a clean image o_1 corresponding to x_1. The images $\{x_2, ..., x_N\}$ are first warped to x_1 by FlowNet (Ilg et al. 2017), resulting in the aligned images $\{x_2', ..., x_N'\}$. In each iteration, we randomly crop the patches p_1 and p_k ($k \in \{2, 3, ..., N\}$) of the same location from x_1 and x_k' respectively. p_1 and p_k are then sent to SiamTrans. We define the loss function for the restoration as,

$$\begin{aligned} L_{\text{zero-shot}} &= \|M_1(p_1, p_k) - M_2(p_k, p_1)\|_1 \\ &+ \lambda \left(\|M_1(p_1, p_k) - p_1\|_1 + \|M_2(p_k, p_1) - p_k\|_1 \right), \end{aligned} \tag{8}$$

where the first term is the consistency loss, and the second term is the fidelity loss, and $o_{p_1} = M_1(p_1, p_k)$ and $o_{p_k} = M_2(p_k, p_1)$ are the outputs of the two basic networks M_1 and M_2, respectively.

After a number of iterations with different random patch pairs, SiamTrans has learnt how to restore the clean images from $x_1, x_2', ..., x_N'$. Then, we use M_1 or M_2 to obtain the restored images o_1, o_2, ..., and o_N from x_1, x_2', ..., and x_N', respectively.

Hard Patch Refinement

After the initial restoration described in Sec. , SiamTrans with learned image prior can recover a good result for a specific scene. However, due to the variety of degradation, it is difficult to get a completely clean image in the zero-shot setting. So we design a hard patch refinement to locate and recover the patches where the degradation has not been well tackled.

Localization. As shown in Fig. 4, to localize the hard patches in the frame x_1, we generate a residual map $r = \|o_1 - x_1\|_1$. On the residual map r, we select n points with the highest values, where the patches of size $s \times s$ centered at these points are non-overlapping.

Refinement. After locating the hard patches that are not well restored, we extract n $s \times s$ patches $p_j^1, p_j^2, ..., p_j^n$ at the n centers of $o_j,\ j = 1, 2, ..., N$. We update each patch in each frame iteratively. For every patch p_1^k in o_1, we update

p_1^k as follows:

$$p_1^k \leftarrow \alpha \times p_1^k + \frac{1-\alpha}{N-1}\left(\sum_{m=2}^{N} W_1(p_m^k \uparrow)\right)\downarrow, \quad (9)$$

where α, W_1, $\uparrow$, and $\downarrow$ denote the balancing parameter, warping towards o_1, upsampling, and downsampling, respectively. We use the pre-trained LIIF model (Chen, Liu, and Wang 2020) to perform upsampling and downsampling. The upsampling is for better alignment of the frames. Then, for every patch p_j^k in $o_j,\ j = 2, 3, ..., N$, we update p_j^k as follows:

$$p_j^k \leftarrow \alpha \times p_j^k + \frac{1-\alpha}{N-2}\left(\sum_{m=2, m\neq j}^{N} W_j(p_m^k \uparrow)\right)\downarrow, \quad (10)$$

where W_j denotes warping towards o_j.

Experiments and Analysis

In this section, we show ablation study and comparison with state-of-the-art methods. Our algorithm is implemented on a NVIDIA Tesla V100 GPU in PyTorch. The network is optimized with the Adam (Kingma and Ba 2015) optimizer. In both the pre-training and zero-shot restoration, the batch size is set to 1 and the initial learning rate is 1×10^{-5}. The algorithm runs for 20 epochs and 20 iterations for the pre-training and the hard patch refinement, respectively. For zero-shot restoration, it takes 200, 500, and 1000 iterations for demoiréing, desnowing, and deraining, respectively. The λ and α in Eqn. 8 and Eqn. 9/Eqn. 10 are empirically set to 5 and 0.9 respectively. Besides, the feature map size $H \times W$ in Sec. is 128×128, the patch size $p \times p$ in Sec. is 32×32, the patch size $s \times s$ in Sec. is also 32×32, and the patch number n in Sec. is 50. The structures of the CNN encoders and decoders can be found from the supplementary materials.

Datasets and State-of-the-Arts

Datasets. 1) Deraining. Since there is no existing short-sequence deraining dataset, we build our multi-frame deraining test set through extracting adjacent frames from the NTURain dataset (Chen et al. 2018), where the images are taken from an unstable panning camera with slow movements. In total, we extract 40 synthetic rain sequences (denoted as *NTURainSyn*) and 12 real rain sequences (denoted as *NTURainReal*), where each sequence contains 8 consecutive frames. The training set for the compared supervised methods is Rain100L (Yang et al. 2017), which contains 1800 scenes and rain maps[1].

2) Demoiréing. We create a multi-frame demoiréing dataset (*MFMoiré*) to evaluate our method quantitatively and qualitatively. We collect 146 high quality images from the Internet as ground truth. To get pre-aligned moiré sequences, we adopt the method in (Sun, Yu, and Wang 2018) to align the ground truth and corresponding moiré images. Each moiré sequence contains 10 moiré images. Note that the moiré patterns vary a lot within a sequence and across different sequences. We split the 146 sequences into the training set with 116 sequences (for compared supervised methods) and the testing set with 30 sequences.

3) Desnowing. We create a multi-frame desnowing dataset (*MFSnow*) to evaluate our method. We use the ground-truth images in *NTURainSyn* as snow-free images and synthesize corresponding snow frames using the method in (Liu et al. 2018). Finally, 3000 training snow sequences (for compared supervised methods) and 30 testing snow sequences are collected.

State-of-the-art methods. 1) For multi-frame deraining, we compare with nine state-of-the-art methods, including three supervised single-image deraining methods (RESCAN (Li et al. 2018), MSPFN (Jiang et al. 2020), and DIDMDN (Zhang and Patel 2018)), one unsupervised image restoration method (DGP (Pan et al. 2020)), three unsupervised video deraining methods (MS-CSC (Li et al. 2019), FastDerain (Jiang et al. 2019), and SelfDerain (Yang et al. 2020c)), and two supervised multi-frame image restoration methods (MLVR (Alayrac 2019) and LSTO (Liu et al. 2020d)). 2) For multi-frame demoiréing, we compare with five state-of-the-art methods, including two supervised single-image demoiréing methods (MopNet (He et al. 2019) and HRDN (Yang et al. 2020b)) and three multi-frame demoiréing methods (MMDM (Liu et al. 2020c), MLVR, and LSTO). 3) For multi-frame desnowing, we compare with one single-frame desnowing method (JSTASR (Chen et al. 2020c)) and three video desnowing methods (MS-CSC, MLVR, and LSTO). Except MS-CSC, the other three

[1]The detailed information about how to synthesize rain sequences is described in the supplementary materials.

	Method	DGP	MSPFN	MS-CSC	FastDeRain
(a)	PSNR↑	20.67	25.16	24.78	25.75
	SSIM↑	0.5291	0.8497	0.7344	0.8991
(b)	NIQE↓	4.051	3.462	3.368	3.627
	Method	SelfDeRain	MLVR	LSTO	SiamTrans
(a)	PSNR↑	26.81	26.78	24.72	**27.02**
	SSIM↑	0.8935	0.8678	0.8853	**0.9024**
(b)	NIQE↓	3.322	3.316	3.354	**3.302**

Table 1: Quantitative deraining comparison. The best results are in **bold**. (a): *NTURainSyn*; (b): *NTURainReal*.

	Method	PSNR↑	SSIM↑	LPIPS↓
MFMoiré	LSTO	21.51	0.6504	0.3573
	MLVR	19.87	0.5904	0.5022
	MMDM	21.61	0.6476	0.3710
	SiamTrans (Ours)	**22.26**	**0.6642**	**0.3197**
MFSnow	LSTO	23.41	0.8228	0.2056
	MLVR	21.88	0.8064	0.2233
	MS-CSC	23.16	0.8201	0.2137
	OTMSCSC	24.21	0.8332	–
	SiamTrans (Ours)	**26.05**	**0.8605**	**0.1323**

Table 2: Quantitative demoiréing and desnowing comparison. The best results are in **bold**.

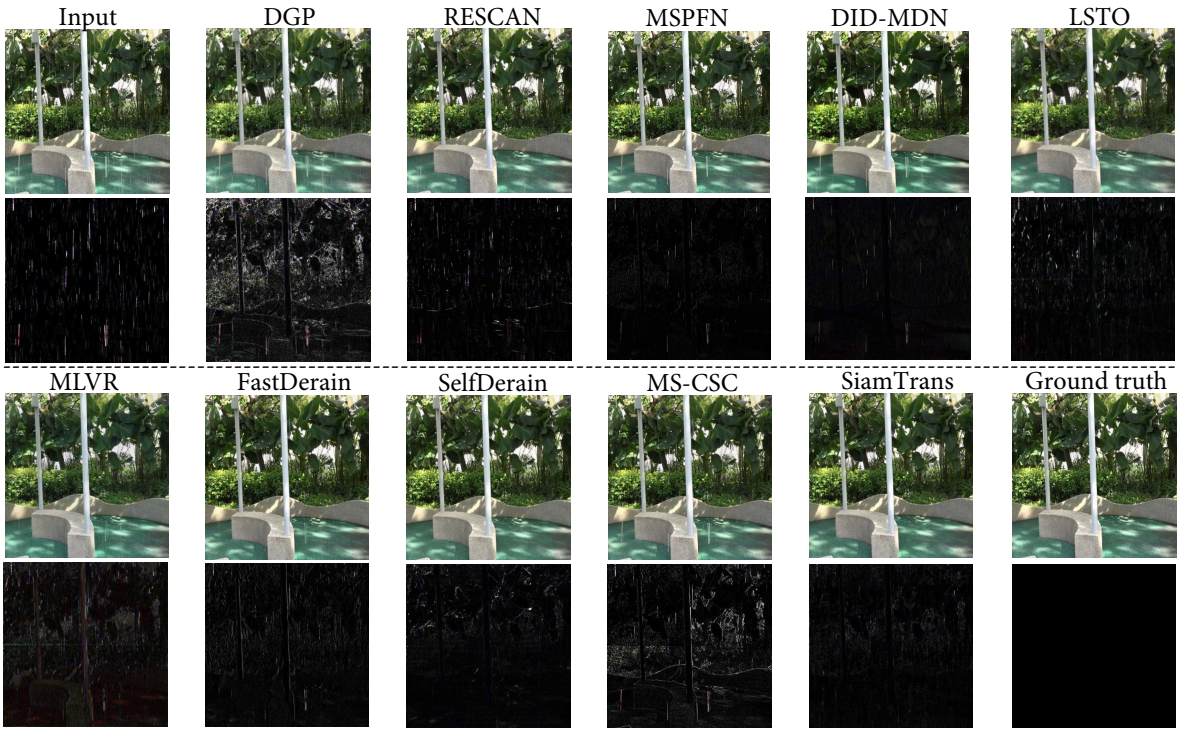

Figure 5: Visual deraining comparison among our method and other methods including deraining-specific methods (RESCAN, MSPFN, DID-MDN, FastDerain, SelfDerain, and MS-CSC) and general restoration methods (DGP, LSTO, and MLVR), evaluated on *NTURainSyn*. The second and the last rows are the differences between the predicted images and the ground truth.

Model	PSNR↑	SSIM↑	LPIPS↓
W/o pre-training	18.92	0.535	0.6248
W/o Hard Patch Refinement	22.23	0.663	0.3244
W/o TA Modules	21.68	0.653	0.3402
two SA Modules	21.87	0.658	0.3387
four SA Modules	22.01	0.661	0.3270
UNet	21.56	0.649	0.3570
SiamTrans	**22.26**	**0.664**	**0.3197**

Table 3: Ablation study on *MFMoiré*.

desnowing methods are supervised. For all the methods, we use their default parameters to generate the results.

Comparison with State-of-the-Arts

Quantitative results. On the datasets with ground truth (*e.g.*, *NTURainSyn* and *MFMoiré*), we use PSNR, SSIM (Wang et al. 2004), and Learned Perceptual Image Patch Similar (LPIPS) (Zhang et al. 2018) to compare the restored images. LPIPS measures perceptual image similarity using a pre-trained deep network. On the dataset without ground truth (*NTURainReal*), we evaluate all generated images using a no-reference quality metric, NIQE (Mittal et al. 2012). As shown in Table 1 and Table 2, the proposed method obtains the best scores on all the evaluation metrics and on all the datasets. More details about the comparison can be found from the supplementary materials.

Qualitative results. As shown in Fig. 5, the first and third rows are the predicted results except the input and the ground truth. The second and the last rows are the differences between the predictions and the ground truth. Through a pre-trained model, the output of DGP losses many image details. The single-image deraining methods, RESCAN, MSPFN and DID-MDN, cannot remove the rain streaks thoroughly because of the limitation of their generalization ability. Our method removes the rain streaks and retains the image details at the same time.

For image demoiréing, as shown in Fig. 6(a), the output results of the single-image demoiréing methods (MopNet and HRDN) may keep some noise or moiré artifacts. Fig. 6(b) shows that the multi-frame methods, MLVR and MMDM, face the color-shift and color artifact problems, respectively. The visual results of desnowing are shown in the supplementary materials.

Ablation Study

To illustrate the contributions of the three stages in our pipeline, we conduct ablation study on *MFMoiré*. The quantitative comparisons are shown in Table 3, where the first row are different variants of our model described below.

Importance of the stages. Although the second stage (zero-shot restoration) is the core of our pipeline, the first and third stages are also important.

Pre-training. The model of 'W/o pre-training' means the model is randomly initialized without going through the pre-training stage. Compared with the final full model 'SiamTrans', it shows that removing the pre-training stage

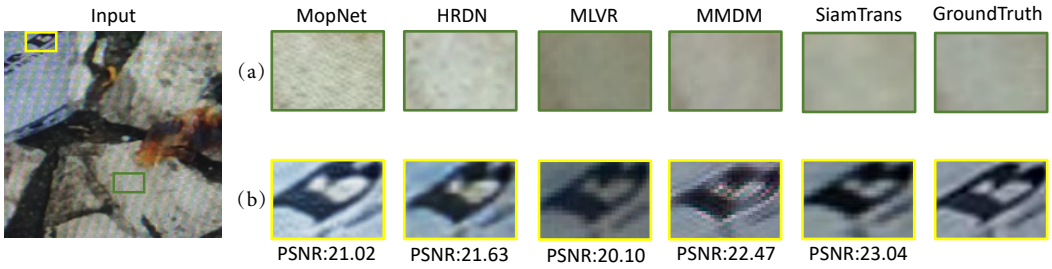

Figure 6: Visual demoiréing comparison among our method and other algorithms including three demoiréing-specific methods (MopNet, HRDN and MMDM) and a general image restoration method (MLVR), evaluated on *MFMoiré*.

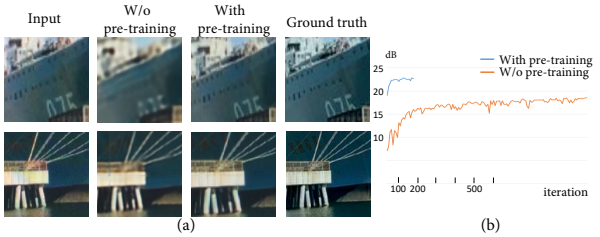

Figure 7: Demoiréing comparison between our two models with and without the pre-training stage.

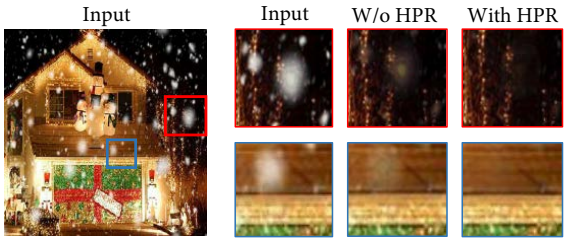

Figure 8: One example of the desnowing results of our method with HPR and without HPR.

leads to significant performance drop of 3.34dB on PNSR. As shown in Fig. 7(a), the results of 'W/o pre-training' are blurry and lose some details. These results indicate that the pre-training is important for the later stages. Without the pre-training, even if we train the later stages longer (*e.g.*, increasing the number of iterations for zero-shot restoration ten times to 2000), the network still cannot capture the details. Fig. 7(b) illustrates the numbers of iterations for both models (with and without the pre-training) needed to reach convergence for the zero-shot restoration stage. It clearly shows that the pre-training enables the network to converge much faster and perform much better in this stage.

Hard patch refinement (HPR). To verify the contribution of the HPR, we remove the 3rd stage from the whole pipeline (indicated as 'W/o HPR' in Table 3). Although its performance drop is not as serious as 'W/o pre-training' in terms of the metrics in Table 3, its visual quality is degraded obviously in many cases. As shown in Fig. 8, SiamTrans can better recover the image occluded by the snow.

The Network Architecture. We also conduct a study to analyze the effectiveness of the network architecture.

Temporal attention module & self-attention module. In Table 3, 'W/o TA' means that in the basic network, the temporal-attention module and one encoder are removed and the input is a single frame. The performance drop in Table 3 verifies the necessity of the temporal-attention module. The self-attention module in the transformer can be stacked to enhance the learning ability. We verify the effectiveness of using multiple self-attention modules. In Table 3, 'n SA Modules' means the transformer of our basic network has n self-attention modules. Comparing '2 SA Modules', '4 SA Modules', and SiamTrans that uses 6 modules, we find that more self-attention modules can get better results.

Transformer or CNN. We replace the transformer with the UNet (Ronneberger et al. 2015) which has the same model size with the transformer in the basic network (denoted as 'UNet' in Table 3). Two feature maps outputed from the encoders are concatenated and served as the input of the UNet. The result shows that using the transformer is better than using the UNet in the basic network.

Practical Applications of our Method

From the above experiments, our multi-frame method performs better than previous single- or multi-frame methods. It requires to use multiple consecutive frames. In practice, it is easy to obtain multiple frames from modern cameras or mobile phones. These equipments have the burst mode and we can get multiple images from one scene in a short period. These multiple frames can also be extracted from videos, like the *NTURainReal* dataset we obtain in Sec. .

Conclusions

In this paper, we have proposed a zero-shot multi-frame image restoration method for removing unwanted obstruction elements that vary in successive frames. Our method contains three stages. After self-supervisedly pre-trained on the denoising task, our SiamTrans model is tested on three tasks unseen during the pre-training. The results of SiamTrans are further improved by the hard patch refinement. Compared with a number of supervised or unsupervised, single-frame or multi-frame state-of-the-arts, our method achieves the best performance on all the tasks and on all the datasets. The future work includes applying SiamTrans to other tasks to explore more of its capacity.

References

Alayrac, J.-B. 2019. The visual centrifuge: Model-free layered video representations. In *CVPR*.

Bau, D.; Strobelt, H.; Peebles, W.; Zhou, B.; Zhu, J.-Y.; Torralba, A.; et al. 2020. Semantic photo manipulation with a generative image prior. *TOG*.

Chan, K. C.; Wang, X.; Xu, X.; Gu, J.; and Loy, C. C. 2012. GLEAN: Generative Latent Bank for Large-Factor Image Super-Resolution. *arXiv preprint arXiv:2012.00739*.

Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; and Gao, W. 2020a. Pre-trained image processing transformer. *arXiv preprint arXiv:2012.00364*.

Chen, J.; Tan, C.-H.; Hou, J.; Chau, L.-P.; and Li, H. 2018. Robust video content alignment and compensation for rain removal in a cnn framework. In *CVPR*.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020b. A simple framework for contrastive learning of visual representations. In *ICML*.

Chen, W.-T.; Fang, H.-Y.; Ding, J.-J.; Tsai, C.-C.; and Kuo, S.-Y. 2020c. JSTASR: Joint Size and Transparency-Aware Snow Removal Algorithm Based on Modified Partial Convolution and Veiling Effect Removal. In *ECCV*.

Chen, Y.; Liu, S.; and Wang, X. 2020. Learning Continuous Image Representation with Local Implicit Image Function. *arXiv preprint arXiv:2012.09161*.

Dai, T.; Li, W.; Cao, X.; Liu, J.; Jia, X.; Leonardis, A.; Yan, Y.; and Yuan, S. 2021. Wavelet-Based Network For High Dynamic Range Imaging. *arXiv*.

Deng, L.-J.; Huang, T.-Z.; Zhao, X.-L.; and Jiang, T.-X. 2018. A directional global sparse model for single image rain removal. *AMM*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*.

Ehret, T.; Davy, A.; Arias, P.; and Facciolo, G. 2019. Joint Demosaicking and Denoising by Fine-Tuning of Bursts of Raw Images. In *ICCV*.

El Mourabit, I.; El Rhabi, M.; Hakim, A.; Laghrib, A.; and Moreau, E. 2017. A new denoising model for multi-frame super-resolution image reconstruction. *Signal Processing*.

Fan, Y.; Yu, J.; Liu, D.; and Huang, T. S. 2020. Scale-wise Convolution for Image Restoration. In *AAAI*.

Farsiu, S.; Robinson, M. D.; Elad, M.; and Milanfar, P. 2014. Fast and robust multiframe super resolution. *TIP*.

Fu, X.; Huang, J.; Zeng, D.; Huang, Y.; Ding, X.; and Paisley, J. 2017. Removing rain from single images via a deep detail network. In *CVPR*.

Godard, C.; Matzen, K.; and Uyttendaele, M. 2018. Deep burst denoising. In *ECCV*.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; Azar, M. G.; et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. *NeurIPS*.

Gu, J.; Shen, Y.; and Zhou, B. 2020. Image processing using multi-code gan prior. In *CVPR*.

Guo, Q.; Sun, J.; Juefei-Xu, F.; Ma, L.; Xie, X.; Feng, W.; and Liu, Y. 2020. EfficientDeRain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining. In *AAAI*.

Guo, X.; Cao, X.; and Ma, Y. 2014. Robust separation of reflection from multiple images. In *CVPR*.

He, B.; Wang, C.; Shi, B.; and Duan, L.-Y. 2019. Mop Moire Patterns Using MopNet. In *ICCV*.

He, B.; Wang, C.; Shi, B.; and Duan, L.-Y. 2020a. FHDe2Net: Full High Definition Demoireing Network. In *ECCV*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020b. Momentum contrast for unsupervised visual representation learning. In *CVPR*.

Ilg, E.; Mayer, N.; Saikia, T.; Keuper, M.; Dosovitskiy, A.; and Brox, T. 2017. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*.

Isobe, T.; Li, S.; Jia, X.; Yuan, S.; Slabaugh, G.; Xu, C.; Li, Y.-L.; Wang, S.; and Tian, Q. 2020. Video super-resolution with temporal group attention. In *CVPR*.

Jaw, D.-W.; Huang, S.-C.; and Kuo, S.-Y. 2020. Desnow-GAN: An Efficient Single Image Snow Removal Framework using Cross-resolution Lateral Connection and GANs. *TCSVT*.

Jiang, K.; Wang, Z.; Yi, P.; Chen, C.; Huang, B.; Luo, Y.; Ma, J.; and Jiang, J. 2020. Multi-Scale Progressive Fusion Network for Single Image Deraining. In *CVPR*.

Jiang, T.-X.; Huang, T.-Z.; Zhao, X.-L.; Deng, L.-J.; and Wang, Y. 2019. FastDeRain: A Novel Video Rain Streak Removal Method Using Directional Gradient Priors. *TIP*.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. *ICLR*.

Kokkinos, F.; and Lefkimmiatis, S. 2019. Iterative residual cnns for burst photography applications. In *CVPR*.

Li, M.; Cao, X.; Zhao, Q.; Zhang, L.; Gao, C.; and Meng, D. 2019. Video Rain/Snow Removal by Transformed Online Multiscale Convolutional Sparse Coding. *arXiv preprint arXiv:1909.06148*.

Li, M.; Cao, X.; Zhao, Q.; Zhang, L.; and Meng, D. 2021. Online Rain/Snow Removal From Surveillance Videos. *TIP*.

Li, X.; Wu, J.; Lin, Z.; Liu, H.; and Zha, H. 2018. Recurrent Squeeze-and-Excitation Context Aggregation Net for Single Image Deraining. In *ECCV*.

Li, Y.; and Brown, M. S. 2013. Exploiting reflection change for automatic reflection removal. In *ICCV*.

Liang, Z.; Guo, S.; Gu, H.; Zhang, H.; and Zhang, L. 2020. A Decoupled Learning Scheme for Real-World Burst Denoising from Raw Images. In *ECCV*.

Liu, H.; Jiang, B.; Xiao, Y.; and Yang, C. 2019. Coherent semantic attention for image inpainting. In *ICCV*.

Liu, L.; Liu, J.; Yuan, S.; Slabaugh, G.; Leonardis, A.; Zhou, W.; and Tian, Q. 2020a. Wavelet-based dual-branch network for image demoiréing. *ECCV*.

Liu, L.; Yuan, S.; Liu, J.; Bao, L.; Slabaugh, G.; and Tian, Q. 2020b. Self-Adaptively Learning to Demoire from Focused and Defocused Image Pairs. *NeurIPS*.

Liu, S.; Li, C.; Nan, N.; Zong, Z.; and Song, R. 2020c. MMDM: Multi-frame and multi-scale for image demoiréing. In *CVPRW*.

Liu, Y.-F.; Jaw, D.-W.; Huang, S.-C.; and Hwang, J.-N. 2018. DesnowNet: Context-aware deep network for snow removal. *TIP*.

Liu, Y.-L.; Lai, W.-S.; Yang, M.-H.; Chuang, Y.-Y.; and Huang, J.-B. 2020d. Learning to See Through Obstructions. In *CVPR*.

Liu, Z.; Luo, S.; Li, W.; Lu, J.; Wu, Y.; Li, C.; and Yang, L. 2020e. ConvTransformer: A Convolutional Transformer Network for Video Frame Synthesis. *arXiv preprint arXiv:2011.10185*.

Mildenhall, B.; Barron, J. T.; Chen, J.; Sharlet, D.; Ng, R.; and Carroll, R. 2018. Burst denoising with kernel prediction networks. In *CVPR*.

Mittal, A.; Soundararajan, R.; Bovik, A. C.; and et al. 2012. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*.

Pan, X.; Zhan, X.; Dai, B.; Lin, D.; Loy, C. C.; and Luo, P. 2020. Exploiting deep generative prior for versatile image restoration and manipulation. In *ECCV*.

Ren, W.; Tian, J.; Han, Z.; Chan, A.; and Tang, Y. 2017. Video desnowing and deraining based on matrix decomposition. In *CVPR*.

Ronneberger, O.; Fischer, P.; Brox, T.; and et al. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Sun, Y.; Yu, Y.; and Wang, W. 2018. Moiré Photo Restoration Using Multiresolution Convolutional Neural Networks. *TIP*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wang, X.; Yu, K.; Dong, C.; and Loy, C. C. 2018. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *TIP*.

Wronski, B.; Garcia-Dorado, I.; Ernst, M.; Kelly, D.; Krainin, M.; Liang, C.-K.; Levoy, M.; and Milanfar, P. 2019. Handheld multi-frame super-resolution. *TOG*.

Xue, T.; Rubinstein, M.; Liu, C.; and Freeman, W. T. 2015. A computational approach for obstruction-free photography. *TOG*.

Yan, Q.; Gong, D.; Shi, Q.; Hengel, A. v. d.; Shen, C.; Reid, I.; and Zhang, Y. 2019. Attention-guided network for ghost-free high dynamic range imaging. In *CVPR*.

Yang, F.; Yang, H.; Fu, J.; Lu, H.; and Guo, B. 2020a. Learning Texture Transformer Network for Image Super-Resolution. In *CVPR*.

Yang, S.; Lei, Y.; Xiong, S.; and Wang, W. 2020b. High Resolution Demoire Network. In *ICIP*.

Yang, W.; Tan, R. T.; Feng, J.; Liu, J.; Guo, Z.; and Yan, S. 2017. Joint Rain Detection and Removal from a Single Image. *CVPR*.

Yang, W.; Tan, R. T.; Wang, S.; and Liu, J. 2020c. Self-Learning Video Rain Streak Removal: When Cyclic Consistency Meets Temporal Correspondence. In *CVPR*.

Yuan, S.; Timofte, R.; Leonardis, A.; and Slabaugh, G. 2020. Ntire 2020 challenge on image demoireing: Methods and results. In *CVPR Workshops*.

Yuan, S.; Timofte, R.; Slabaugh, G.; and Leonardis, A. 2019a. Aim 2019 challenge on image demoireing: Dataset and study. In *ICCV Workshops*.

Yuan, S.; Timofte, R.; Slabaugh, G.; Leonardis, A.; Zheng, B.; Ye, X.; Tian, X.; Chen, Y.; Cheng, X.; Fu, Z.; et al. 2019b. Aim 2019 challenge on image demoireing: Methods and results. In *ICCV Workshops*.

Zeng, Y.; Fu, J.; and Chao, H. 2020. Learning Joint Spatial-Temporal Transformations for Video Inpainting. In *ECCV*.

Zhang, H.; and Patel, V. M. 2018. Density-aware Single Image De-raining using a Multi-stream Dense Network. In *CVPR*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*.

Zheng, B.; Yuan, S.; Slabaugh, G.; and Leonardis, A. 2020. Image demoireing with learnable bandpass filters. In *CVPR*.

Zheng, B.; Yuan, S.; Yan, C.; Tian, X.; Zhang, J.; Sun, Y.; Liu, L.; Leonardis, A.; and Slabaugh, G. 2021. Learning Frequency Domain Priors for Image Demoireing. *TPAMI*.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million Image Database for Scene Recognition. *TPAMI*.