SPATIALLY ADAPTIVE LOSSES FOR VIDEO SUPER-RESOLUTION WITH GANS

Xijun Wang[a], *Alice Lucas*[a], *Santiago Lopez-Tapia*[b],
Xinyi Wu[a], *Rafael Molina*[b], *Aggelos K. Katsaggelos*[a]

a) Dept. of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, USA
b) Dpto. de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada, Spain

ABSTRACT

Deep Learning techniques and more specifically Generative Adversarial Networks (GANs) have recently been used for solving the video super-resolution (VSR) problem. In some of the published works, feature-based perceptual losses have also been used, resulting in promising results. While there has been work in the literature incorporating temporal information into the loss function, studies which make use of the spatial activity to improve GAN models are still lacking. Towards this end, this paper aims to train a GAN guided by a spatially adaptive loss function. Experimental results demonstrate that the learned model achieves improved results with sharper images, fewer artifacts and less noise.

Index Terms— Video Super-Resolution, Generative Adversarial Networks, Perceptual Loss, Spatial Adaptivity

1. INTRODUCTION

The first Deep Neural Network (DNN)-based approach for Video Super-Resolution was proposed by Kappeler et al. [1], who used an end-to-end approach to train a three-layer Convolutional Neural Network (CNN) for super-resolving a sequence of low-resolution (LR) frames to the corresponding high-resolution (HR) center frame. While some authors experimented with the use of Recurrent Neural Networks (RNNs) for VSR [2], the challenges and difficulties associated with RNN-based training has led CNN-based approaches to be the favored ones. More recently, the use of GANs was introduced to learn complex distributions of various datasets [3]. Due to this, the use of a GAN-based training instead of an mean squared error (MSE)-based training enables the model to generate frames of much higher perceptual quality [4, 5, 6].

A promising new trend in VSR has also emerged; instead of focusing on the previously ubiquitous optimization of the mean squared error, state of the art approaches are investigating the use of feature-based cost functions. Many works for image SR have successfully incorporated feature-based perceptual losses which resulted in near-photorealistic images [4, 7, 8]. These feature losses compute differences between high-level image feature representations extracted from pre-trained convolutional neural networks. In most recent SR works, a weighted combination of distance-based losses in both feature and pixel spaces have been proposed to improve the GAN model and obtained very promising results applied for still images in [5] and videos in [9].

It is well-known that adapting the processing to the spatial activity of an image frame in general produces improved visual results in many image processing problems [10, 11]. It is therefore reasonable to include spatial information into the training objective functions. However, to the best of our knowledge, no published results have proposed to incorporate spatial information into the distance-based losses used in super-resolution tasks. Related to this in [6, 9], the authors make use of the motion information from past and future frames in defining loss functions, but do not take spatial activity information into account. In the high-resolution frames generated by our VSRResFeatGAN model in [9, 12], we could still observe noise and also blurred edges of objects when zoomed in. We therefore propose in this paper the use of spatially adaptive losses both in the native spatial domain as well as in the feature domain when training a GAN-based VSR neural network.

The rest of this paper is organized as follows. In Section 2 we first describe our framework for extracting the spatial information from the high-resolution frames in our training data. We then combine this spatial information with the Charbonnier loss to convert our pixel and feature-based perceptual losses to spatially adaptive ones. Finally, we combine these losses with the adversarial loss and show that we are successful at generating sharper edges and reducing the noise generated by models which do not use spatial adaptivity in Section 3. We draw our conclusions in Section 4.

2. PROPOSED METHOD

The model we proposed in [9, 12], while outperforming the current state-of-the-art VSR networks, still produces high-resolution frames with blurry edges and artifacts, especially in the high frequency regions. A limitation of this work is that spatial activity is not taken into account when defining the distance losses in both the image and feature spaces. It is clear that edge regions in the frames are more difficult to super-resolve than flat regions, hence a more suitable loss function for training a VSR model should penalize edge regions more than flat areas. We thus propose to improve on the model introduced in [9, 12] and define spatially adaptive losses, in other words, not treat all pixels equally but add more weight to regions with high spatial activity, such as edges. These specialized losses restrict the solution learned by the GAN network so that sharper and void of artifacts (and therefore more pleasing to the human viewer) solutions are generated.

2.1. Extracting spatial information

There are a number of ways to determine the spatial activity in an image. A rather straight forward one is with the use of the local

This work was supported in part by the Sony 2016 Research Award Program Research Project. The work of SLT and RM was supported by the Spanish Ministry of Economy and Competitiveness through project DPI2016-77869-C2-2-R and the Visiting Scholar program at the University of Granada. SLT received financial support through the Spanish FPU program.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

variance. In general, for an image x with elements $x(i,j)$, the local variance $\mu_{i,j}(x)$ at pixel location (i,j) is calculated according to

$$\mu_{i,j}\,(x) = \sum_{(l_1,\ l_2\,)\in\Gamma_{i,j}} \frac{1}{|\Gamma_{i,j}|}(x(i+l_1, j+l_2) - m_{i,j}(x))^2 \qquad (1)$$

where

$$m_{i,j}(x) = \sum_{(l_1,\ l_2\,)\in\Gamma_{i,j}} \frac{1}{|\Gamma_{i,j}|} x\left(i+l_1,\ j+l_2\,\right) \qquad (2)$$

represents the local mean, $\Gamma_{i,j}$ is the analysis window which in general changes support at each pixel (i,j) and $|\Gamma_{i,j}|$ denotes the number of elements in the analysis window.

The local variance clearly takes large values at the edges and highly textured areas of an image and small values in the flat regions of an image. There are various ways it can be used to control loss functions. One step usually taken before its use is to normalize its values to belong to the range $[0,1]$. While there are various ways to do so [11], one such way is to define a weight image $W(x)$, with elements

$$w_{i,j}(x) = \frac{\mu_{i,j}(x)}{\mu_{i,j}(x)+\delta} \qquad (3)$$

with $\delta > 0$ a tuning parameter determined experimentally. Clearly, in flat regions $w_{i,j}(x) \approx 0$, while in areas of high spatial activity $w_{i,j}(x) \approx 1$, since $\mu_{i,j}(x) \gg \delta$. In [11] the determination of $w_{i,j}(x)$ is motivated through the use of the noise visibility function, which is defined as $1 - w_{i,j}(x)$. It expressed the masking property of the human visual system according to which noise is visible in the flat regions but not visible at the edges. The values of the visibility function for patches in the training set are shown in Figure 1.

In this work we consider multi-channel images (e.g., color and multi/hyper-spectral images), as well as, the representation of an image (single channel or multi-channel) in the feature domain, which is typically a multi-channel domain.

For both of these cases, i.e., multi-channel images in the pixel domain and multi-channel images in the feature domain, we extend the definition in Equation 3. For the former case we define

$$w_{k,i,j}(x) = \frac{\mu_{k,i,j}(x)}{\mu_{k,i,j}(x)+\delta_k} \qquad (4)$$

where x represents the multi-channel image and k the channel index.

In the latter case

$$w_{k,i,j}(x) = feat(\psi(w_{k',i,j}(x))) \qquad (5)$$

where k' provides the index to the number of channels in the pixel domain and k the number of channels in the feature domain represented by the function $feat()$. The number of channels in the two domains are typically not the same. The function $\psi()$ represents a fusion of $w_{k',i,j}(x)$. Regarding the $feat()$ function, we will be using the activations provided by the 3^{rd} and 4^{th} convolutional layers of the VGG network, defined in [13].

2.2. Spatially adaptive losses

In this paper, we use two distance-based losses to regularize the GAN training: one defined in pixel space and another in the feature space. As described in [9, 12], we have been using the Charbonnier loss defined as

$$\gamma(u,v) = \sum_k \sum_i \sum_j \sqrt{(u_{k,i,j} - v_{k,i,j})^2 + \epsilon^2}, \qquad (6)$$

for the general case of two multi-channel images u and v, with elements $u_{k,i,j}$ and $v_{k,i,j}$, respectively. Index k is used to describe the channel number, e.g., $k = 1$ for a gray-scale image and $k = 1,2,3$ for a color image.

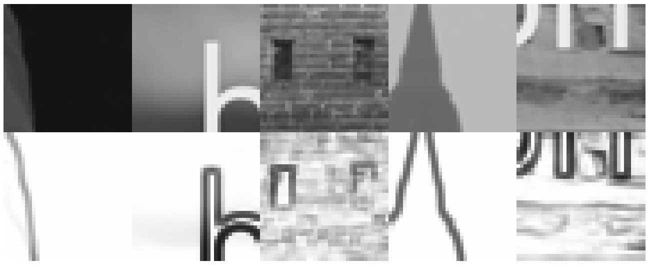

Fig. 1: Top row: image patches; Bottom row: corresponding values of the visibility function.

The Charbonnier loss has shown to be successful in stabilizing the GAN training and produces more robust solutions [9, 12]. One limitation of the definition of the Charbonnier loss above is that all pixel in the frame are weighted equally. However, a typical requirement in SR is to produce HR images which preserve as much as possible the edge information. Towards this end, during the training of our generator, regions of high spatial activity should be weighted heavier than smooth regions. Therefore, we propose the following modification of the Charbonnier loss for our distance-based losses:

$$\gamma_W\,(u,v,W(u)) = \sum_k \sum_i \sum_j w_{k,i,j}\,(u)\sqrt{(u_{k,i,j} - v_{k,i,j})^2 + \epsilon^2}, \qquad (7)$$

where $W(u)$ includes the set of weights $w_{k,i,j}\,(u)$ and the interpretation of these weights is either according to Equation 4 or 5, depending on whether we are defining the loss in the pixel or feature spaces, respectively. In the following, we describe how to compute the spatial activity information in both in pixel and feature spaces, and incorporate it into the low-level pixel loss and the high-level feature loss. In this way, the regions of high spatial activity will be heavier weighted both in the native pixel and feature domains.

Spatially adaptive pixel-wise loss in pixel space. The idea of incorporating spatial information into the pixel-wise loss is quite simple, because it only depends on low-level pixel information. We thus propose to use the weighted Charbonnier loss in pixel space:

$$L_{pixel} = \sum_{(x,Y)} \gamma_W\left(x, G_\theta\,(Y)\,, \alpha + \beta W(x)\right), \qquad (8)$$

where x is the center high-resolution frame at time t, Y is the sequence of low-resolution frames defined at times $t-M, ..., t-1,\ t,\ t+1, ..., t+M$, for a predetermined M, and $G_\theta\,(Y)$ is the high-resolution estimate of x provided by the generator network. The elements of weight matrix W(x) are defined by Equation 4. The weight α is a hyper-parameter which controls the contribution of the equally weighted pixel-wise loss (thus defining an unweighted Charbonnier loss term). The weight β is a hyper-parameter which controls the weighted Charbonnier loss in pixel space. The larger the value of $w_{k,i,j}(x)$, the more important the corresponding pixel becomes in the function to be optimized. In other words, during the backward pass, larger weight updates will be given to those pixels responsible for super-resolving these edge-like regions. The reason for including the α weight is to ensure that the training loss does not ignore smooth regions in the video frames.

Spatially adaptive perceptual loss in feature space. In addition to imposing spatial activity constraints at the pixel level, we also impose restrictions on the feature loss that measures high-level perceptual differences between the predicted and ground-truth frames in a precomputed feature space, leveraging the deep compressed representations learned by deep discriminative classifiers. Our feature space is computed from the activations provided by the 3^{rd} and 4^{th} convolution layers of the VGG network [13], denoted as $VGG(\cdot)$. Thus, our spatially adaptive feature-based perceptual loss (or feature loss) becomes:

$$L_{feature} = \sum_{(x,Y)} \gamma_w (VGG(x), VGG(G_\theta(Y)), \alpha + \beta VGG(W(x))) \tag{9}$$

which, similarly to the spatially adaptive loss in pixel space, corresponds to a weighted Charbonnier loss. Note that because the difference is computed in a high-level feature space, using the weight matrix $W(x)$ with elements directly computed from Equation 4 is not appropriate in this case. Therefore, it is necessary to compute the equivalent of the weight matrix $W(x)$ (computed in pixel space) in the corresponding feature space. Therefore, we propose to convert $W(x)$ to its equivalent in feature-space by feeding it into the VGG network as well, i.e., computing $VGG(W(x))$ and using its output as the new weight matrix (see Equation 5). Therefore, the resulting matrix is used to assign more weight to the regions of high spatial activity, as represented in the feature space. As in the previous section, α and β control the contribution of equally weighted feature and unequally weighted feature loss, respectively.

2.3. GAN loss

Following the state-of-the-art methods in super-resolution for both classical and perceptual loss functions [4, 6, 9] we use a GAN-based training to produce frames of high perceptual quality. We adopt the VSRResNet architecture proposed in [9, 12] as our generator. The architecture is shown in Figure 2. It is composed of 15 residual blocks, each block containing two convolutional layers with kernels of size 3 by 3, with a Rectified Linear Unit (ReLU) activation function following each convolution step.

The discriminator used in our work is borrowed from [9, 12] and is composed of three convolutional layers followed by a fully connected layer and a sigmoid operation, providing the probability of a real patch. The discriminator architecture is shown in Figure 3.

Adapting the GAN formulation first introduced in [3] to VSR results in solving the adversarial min-max problem

$$\min_{\theta} \max_{\phi} L_{GAN}(\phi, \theta) = \mathbb{E}_x \left[\log D_\phi(x)\right] + \mathbb{E}_Y \left[\log\left(1 - D_\phi\left(G_\theta(Y)\right)\right)\right], \tag{10}$$

where D_ϕ is the discriminator with trainable parameters ϕ and G_θ is the generator network with trainable parameters θ.

The generator network minimizes the following loss with respect to θ

$$L_{gen} = \mathbb{E}_Y \left[-\log D_\phi\left(G_\theta(Y)\right)\right], \tag{11}$$

While the discriminator network minimizes the following loss with respect to ϕ

$$L_{dis} = \mathbb{E}_x \left[-\log D_\phi(x)\right] + \mathbb{E}_Y \left[-\log\left(1 - D_\phi\left(G_\theta(Y)\right)\right)\right] \tag{12}$$

In the next section, we introduce our proposed spatially adaptive loss for training the GAN model.

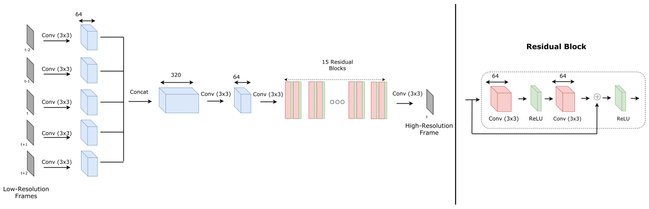

Fig. 2: The proposed generator architecture [9, 12]

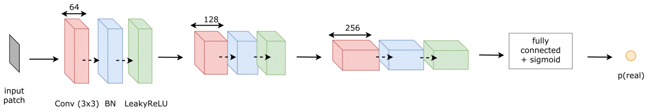

Fig. 3: The proposed discriminator architecture [9, 12]

2.4. The spatially adaptive loss

Our final model incorporates spatial adaptivity into the distance-based losses in pixel and feature space during the adversarial training. We thus combine the GAN loss with the spatially adaptive pixel and feature loss, resulting in the following spatially adaptive loss

$$\begin{aligned} L_{final} = {} & \alpha_1 \left[\mathbb{E}_Y \left[-\log D_\phi\left(G_\theta(Y)\right)\right]\right] \\ & + \sum_{(x,\ Y)} \gamma_w\left(x,\ G_\theta(Y), \alpha_2 + \beta_2 W(x)\right) \\ & + \sum_{(x,\ Y)} \gamma_w\left(VGG(x),\ VGG\left(G_\theta(Y)\right),\ \alpha_3 + \beta_3 VGG(W(x))\right), \end{aligned} \tag{13}$$

where α_1 controls the contribution of the GAN loss, α_2 and α_3 control the contribution of the equally weighted pixel-wise and feature losses, and β_2 and β_3 the contribution of the weighted pixel-wise and feature loss, respectively. We name our final model trained with this spatially adaptive loss the Spatially Adaptive VS-RGAN (Spatially adaptive video super-resolution GAN).In the next sections, we show that the use of our final spatially adaptive loss greatly improves the performance of generator networks.

3. EXPERIMENTS

3.1. Training and parameters

Our training dataset is extracted from the Myanmar video sequence. Each sample in the training dataset is composed of five extracted 36 × 36 low-resolution patches ($M = 2$) at times $\{t-2, t-1, t, t+1, t+2\}$, and their corresponding 36 × 36 HR patch at time t. The LR frames are computed using bicubic downsampling followed by bicubic interpolation in order to bring them to the same spatial extent as the HR patch.

Pre-training the generator with a pixel-wise loss helps ensuring a subsequent stable GAN training process. Therefore, we pre-trained the generator for 100 epochs with the traditional MSE loss in pixel-level using the ADAM [14] optimizer and a batch size of 64. For this pre-training experiment, the initial learning rate is set to $10^{-3}$and is then further divided by a factor of 10 at the 50th and 75th epoch of the training. We train a separate generator for each of the SR scale factors of 2, 3 and 4.

Using the weights of this pre-trained generator as initial weights, we trained our GAN model with the spatially adaptive perceptual loss defined in Equation 13 for 30 epochs, setting the learning rate to 10^{-4} for both the discriminator and generator networks. The weight decay was set to 10^{-3} for the discriminator and 10^{-4} for the generator. We use the ADAM [14] optimizer and a batch size of 64. The values for α and β parameters are determined experimentally with the constraint that their sum adds up to 1. We find their optimal values to be: $\alpha_1 = 0.001$, $\alpha_2 = 0.001$, $\beta_2 = 0.1$, $\alpha_3 = 0.798$ and $\beta_3 = 0.1$. The ϵ parameter in the Charbonnier loss is set to 0.001 and the parameter δ in Equation 4 is set to 0.01 in our experiments. We found out that 30 epochs is an appropriate number for our model to converge.

While the elements of the spatially weight matrix W initially range from 0 to 1, we choose to scale them by factor 10, which we found resulted in sharper edges and less artificial noise in the resulting frames.

3.2. Evaluation results

We trained our model on the Myanmar dataset. In order to check whether our model could also work well in different datasets, we test our model on the VidSet4 dataset [15], a commonly used dataset for testing video super-resolution models.

Recent works in super-resolution have shown that the PSNR metric does not always provide an accurate assessment of the perceptual quality of the HR images. During experiments, we also found that PSNR and SSIM values sometimes do not agree with the subjective evaluation of the quality of a frame. Recently, Zhang et al. [16] have proposed a new standard to compare the perceptual similarity between a reference image and a distorted one [17], with a convolutional neural network. Given a reference (ground-truth) and example image (prediction), the CNN outputs distance values which quantify the perceptual similarity between the two images. The author found that this Perceptual Distance predicted by these networks provides results consistent with the human judgement. Similarly, in [12] we found that this perceptual distance was consistent with the human's opinions regarding the sharpness of the produced super-resolved video frames.

Using the PSNR, SSIM, and Perceptual Distance (which we refer to as the PercepDist metric), we compare our Spatially Adaptive VSRGAN with the current state-of-the-art video super-resolution model, VSRResFeatGAN, proposed in [9, 12] for each scale factor. The results of our computation are shown in table 1.

	VSRResFeatGAN PSNR/SSIM/PercepDist	Spatially Adaptive VSRGAN PSNR/SSIM/PercepDist
2	30.90/0.9241/0.0283	**31.64/0.9327/0.0257**
3	26.53/0.8148/0.0668	**26.80/0.8256/0.0641**
4	24.50/0.7023/0.1043	**24.72/0.7233/0.1010**

Table 1: Comparison with state-of-the-art for VidSet4 dataset for scale factors 2,3, and 4. For the PercepDist metric, smaller is better.

Table 1 shows that our Spatially Adaptive VSRGAN model surpasses the state-of-the-art VSRResFeatGAN model [9, 12] with respect to all three metrics. A qualitative comparison is shown in Figures 4-5. Considering the zoomed in regions in the frames we see that our Spatially Adaptive VSRGAN model more accurately super-resolves edges and fine details with less noise compared with the baseline model.

We conclude from these quantitative and qualitative results that the use of spatial information into the losses for training GANs has a significant improvement on the perceptual quality of the resulting frame.

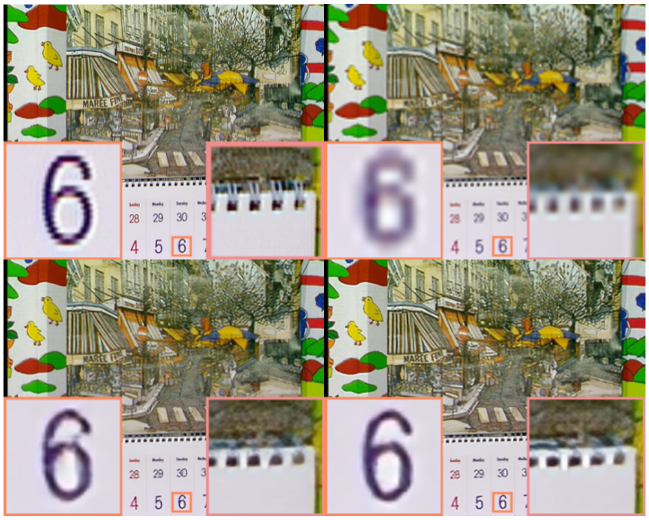

Fig. 4: Qualitative comparison of VSRResFeatGAN (2nd row, left) vs. Spatially Adaptive VSRGAN (2nd row, right) on scale factor 4, with ground Truth (1st row, left) and input low-resolution (1st row, right).

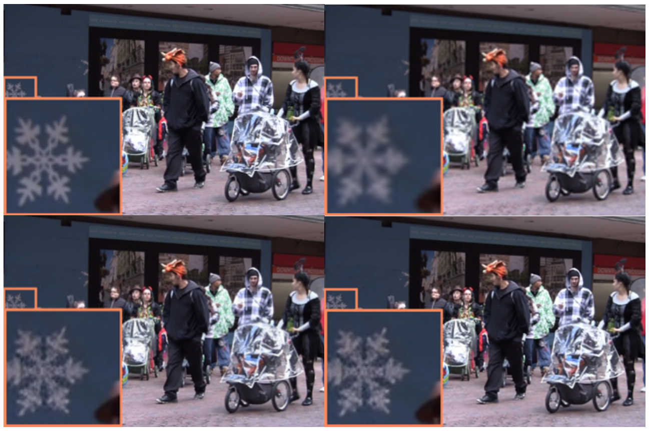

Fig. 5: Qualitative comparison of VSRResFeatGAN [12] (2nd row, left) vs. Spatially Adaptive VSRGAN (2nd row, right) on scale factor 3, with ground Truth (1st row, left) and input low-resolution (1st row, right)

4. CONCLUSION

In this paper, we have described our approach for using a weighted loss in pixel and feature-spaces. We showed that the use of such a spatially adaptive loss during the GAN training results in sharper edges, better reconstruction of fine details, and a significant decrease in the noise resulting from a GAN-based training. Future work will involve exploring the use of spatial information into our adversarial loss as well, in order to provide better guidance to the GAN and improve its solution. Furthermore, we will investigate the use of additional means to incorporate spatial information into the loss functions, in order to further encourage our model to generate more naturalistic frames.

5. REFERENCES

[1] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, “Video super-resolution with convolutional neural networks,” *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[2] Y. Huang, W. Wang, and L. Wang, “Bidirectional recurrent convolutional networks for multi-frame super-resolution,” in *Advances in Neural Information Processing Systems 28*, pp. 235–243, 2015.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[4] M. S. Sajjadi, B. Schölkopf, and M. Hirsch, “Enhancenet: Single image super-resolution through automated texture synthesis,” in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 4501–4510, IEEE, 2017.

[5] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, “Photo-realistic single image super-resolution using a generative adversarial network.,” in *CVPR*, vol. 2, p. 4, 2017.

[6] E. Pérez-Pellitero, M. S. Sajjadi, M. Hirsch, and B. Schölkopf, “Photorealistic video super resolution,” *arXiv preprint arXiv:1807.07930*, 2018.

[7] X. Wang, K. Yu, C. Dong, and C. C. Loy, “Recovering realistic texture in image super-resolution by deep spatial feature transform,” *CoRR*, vol. abs/1804.02815, 2018.

[8] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *European Conference on Computer Vision*, pp. 694–711, Springer, 2016.

[9] A. Lucas, A. K. Katsaggelos, S. L. Tapia, and R. Molina, “Generative adversarial networks and perceptual losses for video super-resolution,” in *ICIP*, 2018.

[10] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, “Svm active learning approach for image classification using spatial information,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, pp. 2217–2233, April 2014.

[11] S. N. Efstratiadis and A. K. Katsaggelos, “Adaptive iterative image restoration with reduced computational load,” *Optical engineering*, vol. 29, no. 12, pp. 1458–1469, 1990.

[12] A. Lucas, S. Lopez Tapia, R. Molina, and A. K. Katsaggelos, “Generative adversarial networks and perceptual losses for video super-resolution,” *ArXiv e-prints*, June 2018.

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

[14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014.

[15] C. Liu and D. Sun, “A bayesian approach to adaptive video super resolution,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 209–216, IEEE, 2011.

[16] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep networks as a perceptual metric,” in *CVPR*, 2018.

[17] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” *arXiv preprint arXiv:1704.03264*, 2017.