A Single Video Super-Resolution GAN for Multiple Downsampling Operators based on Pseudo-Inverse Image Formation Models

Santiago López-Tapia *, Alice Lucas†, Rafael Molina*, Aggelos K. Katsaggelos†
*Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
Email: {sltapia, rms}@decsai.ugr.es
†Dept. of Electrical Engineering and Computer Science Northwestern University Evanston, IL, USA
Email: alicelucas2015@u.northwestern.edu, aggk@eecs.northwestern.edu

Abstract

The popularity of high and ultra-high definition displays has led to the need for methods to improve the quality of videos already obtained at much lower resolutions. A large amount of current CNN-based Video Super-Resolution methods are designed and trained to handle a specific degradation operator (e.g., bicubic downsampling) and are not robust to mismatch between training and testing degradation models. This causes their performance to deteriorate in real-life applications. Furthermore, many of them use the Mean-Squared-Error as the only loss during learning, causing the resulting images to be too smooth. In this work we propose a new Convolutional Neural Network for video super resolution which is robust to multiple degradation models. During training, which is performed on a large dataset of scenes with slow and fast motions, it uses the pseudo-inverse image formation model as part of the network architecture in conjunction with perceptual losses and a smoothness constraint that eliminates the artifacts originating from these perceptual losses. The experimental validation shows that our approach outperforms current state-of-the-art methods and is robust to multiple degradations.

Index Terms

Video, Super-resolution, Convolutional Neuronal Networks, Generative Adversarial Networks, Perceptual Loss Functions

I. Introduction

The task of Super-Resolution (SR) consists of obtaining High-Resolution (HR) images from the corresponding Low-Resolution (LR) ones. This task has become one of the main problems in image and video processing because of the increasing demand for such methods from the industry. Due to the growing popularity of high-definition display devices, such as High-definition television (HDTV) and Ultra-high-definition television (UHDTV), there is an avid demand for HR videos. However, most of the content (especially, older videos) has been obtained at much lower resolution. Therefore, there is a high demand for methods able to transfer LR videos into HR ones so that they can be displayed on HR TV screens, void of artifacts and noise.

In the problem of image Super-Resolution (SR), the high-to-low image formation model can be written as:

$$y = D(x \otimes k) + \epsilon, \tag{1}$$

where y is the LR image, x is the HR image, ϵ is the noise, $x \otimes k$ represents the convolution of x with the blur kernel k and D is a downsampling operator (usually chosen to be bicubic downsampling). In the case of Video Super-Resolution (VSR), y, x, and ϵ are indexed by a time index t and additionally $\mathbf{y}_t$ is used to refer to the $2l+1$ LR frames in a time window around the HR center frame x_t, that is, $\mathbf{y}_t = \{y_{t-l}, \ldots, y_t, \ldots, y_{t+l}\}$. Due to the strongly ill-posed nature of the SR problem, the recovery of the original HR image or video sequence is a difficult task.

Current SR methods can be divided into two broad categories: model-based and learning-based approaches. Model-based approaches explicitly define and use the degradation process described by Eq. 1 by which LR image is obtained from the HR image or video sequence, see the reviews [1], [2]. With this explicit modeling,

an inverse problem is solved to obtain an estimate of the reconstructed HR frame. These methods rely on careful regularization to deal with the ill-posedness of the problem. To enforce image-specific features into the estimated HR, signal priors are used, such as those controlling the smoothness or the total variation of the reconstructed image. In the case of the work presented in [3] a new multichannel image prior model is used in conjunction with a state-of-the art image prior and observation models to produce the SR image using a MAP algorithm. In [4], an SR image is obtained from the LR observations through the simultaneous estimation of the SR and the motion between the LR observations using the Bayesian framework.

On the other hand, learning-based approaches do not explicitly make use of the image formation model and use instead a large training database of HR and LR image/sequence pairs to learn the solution to the SR problem, i.e. they learn a mapping between the LR observations and the HR estimation [5], [6]. Classic learning-based models mainly focused on how to build a dictionary or manifold space to relate LR and HR images and determine what representation schemes could be used in such spaces [5], [7], [8], [9]. Recently, methods based on Convolutional Neural Networks (CNNs) have been proposed for image SR and VSR, typically outperforming classic learning-based and model-based methods. These methods try to find a function $\mathrm{f}(\cdot)$ such that $x = \mathrm{f}(y)$ (or $x = \mathrm{f}(\mathbf{y})$ for VSR), which solves the mapping from LR images (or video sequences) to HR ones. The first use of these models for image SR was proposed in [6], where a three layered CNN was used to recover an HR image from its bicubically upsampled LR observation. Following works improved the architecture of the network by introducing layers that allow the network to learn the upsample operator [10] or increase the depth of the network through the use of residual blocks [11].

Although CNN-based models typically outperform model-based methods, most of them are not as flexible as model-based methods, in the sense that they are trained for a specific type of degradation operator. More specifically, an artificially synthesized dataset with LR and HR pairs is generated using only one degradation operator A (usually bicubic downsampling) for training. In addition, the LR sequences used for testing are assumed to have been subjected to the same degradation. This limits the trained model to only one type of degradation and its performance greatly deteriorates when a mismatch between training and testing degradation models occurs (see [12], [13]). This significantly limits their practical application. Recently, some works have proposed SR models that address this issue [12], [13] (see Section II).

An additional problem with most CNN-based approaches to SR and VSR problems is that they are trained using the Mean-Squared-Error (MSE) cost function between the estimated and original HR frames. Numerous works in the literature (e.g., [14], [11], [15], [16]) have shown that while the MSE-based approach provides reasonable SR solutions, its conservative nature does not fully exploit the potential of Deep Neural Networks (DNNs) and produces blurry images. As an alternative to the MSE cost function, recent CNN-based SR methods use (during training) features learned by pre-trained discriminative networks and compute the MSE between estimated and ground truth HR features. Using such feature-based and pixel-based losses has proven to boost the quality of the super-resolved images [17]. Unfortunately, this approach tends to introduce high-frequency artifacts, as was shown in [17].

The use of Generative Adversarial Networks (GANs) [18] was also proposed as a mechanism to increase the perceptual quality of the estimated images trying to avoid again the smoothing introduced by the MSE loss (e.g., [19], [11], [20]). GANs consist of two networks: a generator network that produces the SR image and a discriminator one that distinguishes between generated images and real ones. These networks can therefore constraint the generated HR images to satisfy the distribution of the real HR images. The produced images are sharper because blurry images do not belong to the distribution of HR images. In the case of SR and VSR, these models are trained incorporating additional terms to the loss function of the generator network (see [11], [15], [16], [17]).

In this work, we propose a new GAN model that adapts the approximation proposed in [21] to Multiple-Degradation Video Super-Resolution (MDVSR). It uses the pseudo-inverse image formation model not only in the image formation model (as proposed in [21]), but also as an input to the network. Our experiments show that this model trained with the MSE loss significantly outperforms current state-of-the-art methods for bicubic degradation in terms of PSNR and SSIM metrics and it is significantly more robust to multiple degradations than current approaches. To further increase the sharpness of the resulting frames, we propose the use of a new loss function that combines adversarial GAN loss and feature loss with a spatial smoothness constraint. This new loss allows for a significant increase in the perceptual quality of the estimated frames

without producing the high-frequency artifacts typically observed with the use of GANs.

For all described models, the use of an appropriate dataset is of paramount importance. While many of the current VSR learning-based models are trained using the Myanmar dataset, this dataset has limited variation of both scene types and motions. In this work we show that GAN-based VSR models significantly benefit from training with a dataset with more diverse scenes and motions. We obtain a significant increase in perceptual quality by training our best performing model on a dataset created from a subset of videos from the YouTube-8M dataset [22].

The rest of the paper is organized as follows. We provide a brief review of the current literature for learning-based VSR in Section II. In Section III, we present our baseline VSR model. In this section we detail the architecture used for our model. By additionally introducing our new spatial smoothness loss we obtain our proposed model trained to maximized perceptual quality. The training procedure, the new dataset, and experiments are described in detail in Section IV. In this section we also evaluate the performance of the proposed models by comparing them with current state-of-the-art VSR approaches for scale factors 2, 3, 4, 8 and different degradations. Our quantitative and qualitative results show that our proposed perceptual model sharpens the frames to a much greater extent than current VSR state-of-the-art DNNs without the introduction of artifacts. In addition, we show in this section that our resulting model is more robust to variations in the degradation model compared with the current state-of-the-art model. Finally, conclusions are drawn in Section V.

II. Related work

In recent years, different VSR CNN-based models have been proposed in the literature. Liao et al. [23] utilize a two-step procedure where an ensemble of SR solutions is first obtained through the use of an analytic approach. This ensemble then becomes the input to a CNN that calculates the final SR solution. Kappeler et al. [24] use a three layer CNN to learn a direct mapping between the bicubically interpolated and motion compensated LR frames in $\mathbf{y}_t$ and the corresponding HR central frame x_t. Other works have applied Recurrent Neural Networks (RNNs) to VSR. For example, in [25] the authors use a bidirectional RNN to learn from past and future frames in the input LR sequence. Although RNNs have the advantage of exploiting more effectively the temporal dependencies between frames, the challenges and difficulties associated with their training has led to CNN being the favored DNN for VSR. Li and Wang [26] exploit the benefits of residual learning with CNNs in VSR by predicting only the residuals between the high-frequency and low-frequency frame. Caballero et al. [27] jointly train a spatial transformer network and a CNN to warp video frames, so that they benefit from sub-pixel information and they avoid the use of motion compensation (MC). Similarly, Makansi et al. [28] and Tao et al. [29] found that jointly performing upsampling and MC increases the performance of the VSR model.

All these previous methods use the MSE loss as the cost function during the training phase. This is the most common practice in the literature for CNN-based models. However, the use of this loss during training causes the estimated frames to be blurred. In an attempt to solve this problem, recent works have used feature-based losses as additional cost functions, see [14]. This approach has significantly improved the sharpness and the perceptual quality of the estimations. Ledig et al. [11] proposed a combination of a GAN and feature loss for training, leading to the generation of images with superior photorealistic quality. In [17], Lucas et al. proposed an adaptation of this approach to VSR. They introduced a new loss based on a combination of perceptual features and the use of the GAN formulation. The model has led to a new state-of-the-art VSR approach in terms of perceptual quality. To improve the quality of the predicted images, Wang et al. [15] train a GAN for image SR conditioning the output of the network using semantic information extracted by a segmentation CNN. In [16] the authors propose the Residual-in-Residual Dense Block and use it to construct a very deep network that is trained for image SR using perceptual losses. More recently, Zareapoor et al. [30] proposed a dual generator and dual discriminator GAN. Each generator is specialized in different data distributions and the first discriminator distinguishes between real and fake data while the second one assigns examples to the correct generator to be re-synthesized in case of an initial mismatch (see [30] for a complete definition of the used terminology). In [31], Shamsolmoali et al. propose to control the model parameters and mitigate the training difficulties by using a densely connected residual network that is trained using a gradual learning process, from small upsampling factors to larger ones.

As previously stated in Section I, SR and VSR methods can be classified into two groups: model based and learning based. Recently new methods that blend the two approaches have emerged. Zhang et al. [32] use the Alternating Direction Method of Multipliers (ADMM) for image recovering problems with known linear degradation models, such as image deconvolution, blind image deconvolution, and SR. ADMM methods split the recovery problem into two subproblems: a regularized recovery one (subproblem A) and a denoising one (subproblem B). The authors of [32] propose to combine learning and analytical approaches by using a CNN for the denoising problem. This allows them to use the same network for multiple ill posed inverse imaging problems. At the same time, some works have been proposed to increase the performance and the flexibility of SR learning-based models by taking into account the image formation model when training their CNNs. More specifically, Sonderby et al. [21] proposed a new approach which estimates and explicitly uses the image formation model to learn the solution modeled by the network. The blurring and downsampling process to obtain LR frames from HR ones is estimated and the Maximum a Posteriori (MAP) HR image estimation procedure is approximated with the use of a GAN. We improve over this approach by generalizing it to multiple degradation operators for the VSR problem. Although we do not make use of more complex GAN training techniques such as the ones using dual generators and discriminators [30] and gradual learning [31], our method achieves state-of-art results due to the use of the LR image formation model in conjunction with our proposed smoothness constraint. To enforce robustness to multiple degradations in the case of single image SR, Zhang et al. [13] propose to input to their CNN not only the LR image but also the Principal Component Analysis (PCA) representation of the blur kernel used in the degradation process. We adopt a similar approach in our framework, as detailed in the next section. Preliminary results of our approach can be found in [33] for bicubic downsampling. In this work we extend it to multiple degradations and improve the loss function and the architecture used in [33].

III. Model description

In this section, we first introduce the problem of VSR with multiple degradations and explain how we can adapt the Amortised MAP approximation in [21] to solve it (see Section III-A). The loss used to train our GAN model is then introduced in Section III-B. Finally, in Section III-C we describe our proposed new architecture based on the VSRResNet architecture originally introduced in [17].

As previously stated in Section I, we use x to denote a HR frame in a video sequence and y its corresponding observed LR frame. Furthermore, we use $\mathbf{y}$ to refer to the LR frames in a time window around the HR center frame x, $\mathbf{y}$ contains $2l+1$ frames (we use $l=2$ in the experiments).

In the problem of VSR, the process of obtaining a LR image from the HR one is usually modeled using Eq. 1. In this paper, we assume that the image formation noise is negligible ($\epsilon = 0$) and that it has been absorbed by the downsampling process. Also, following previous works in the literature, e.g. [13], we assume that D represents bicubic downsampling and the blur k is known and has the form of an isotropic Gaussian kernel. Although more complex blurs, like motion blur, can also be considered, our downsampling and Gaussian blur model is frequently assumed to be a good representation of the high to low resolution degradation process [13].

We also assume that all the frames in the time window are degraded with the same operator. Since this operator depends mostly on the camera used, it is reasonable to assume that these conditions will not change drastically from one frame to another. However, we are not assuming that all cameras produce the same deterioration.

In summary, we assume that D (bicubic downsampling) and k (Gaussian blur) in Eq. 1 are known (or previously estimated) and that they are constant for all frames in $\mathbf{y}$. Notice that assuming that the Gaussian blur is known is not the same as assuming that it is the same for all video sequences. The use of this image forward model leads to a more challenging VSR problem than when only bicubic downsampling is considered, which is the modelling used in most previous works on VSR, see [17], [23], [26], [28], [29], [34], [24].

A. Robust VSR through the use of Amortised MAP

Let us now examine how we can approach the multiple degradations VSR problem. Most current VSR methods solve the problem by learning a function $f_\theta(.)$, which maps a low resolution image to the high

resolution space, using training data pairs x and $\mathbf{y}$ and optimizing the parameters θ using gradient descent over a *Mean Square Error* function. This model embeds the estimation of the downsampling process in the function $\mathrm{f}_\theta(\mathbf{y})$. We argue here that to obtain an SR network capable of dealing with multiple degradations, separating the learning of the HR video sequence from the learning of the degradation makes the whole process more manageable and increases the performance of the network, as shown by the experiments in Section IV. To achieve this, given D and k, we define $A = Dk$ and following [21] consider the function

$$\mathrm{g}_\theta(\mathbf{y}) = (I - A^+A)\mathrm{f}_\theta(\mathbf{y}) + A^+y, \tag{2}$$

where A^+ denotes the Moore-Penrose pseudoinverse of the degradation A. Since $AA^+A = A$ and $A^+AA^+ = A^+$, and because the rows of A are independent $AA^+ = I$, we have that

$$A\mathrm{g}_\theta(\mathbf{y}) = A(I - A^+A)\mathrm{f}_\theta(\mathbf{y}) + AA^+y = y \tag{3}$$

The resulting $\mathrm{g}_\theta(\mathbf{y})$ is an HR image which satisfies Eq. 1 with $\epsilon = 0$. With this formulation, the learning of the network is made easier by learning a residual only (A^+y can be considered as an initial approximation of the HR frame x and $\mathrm{f}_\theta(\cdot)$ is part of the added correction according to Eq. 2). This has been exploited in other works of SR where the networks learn to predict a residual over an initial estimation, usually chosen to be the bicubic interpolation [35]. However, these other approaches are not well suited for the Multiple Degrations setting, since the quality of the initial prediction may vary significantly from one degradation to another, showing different kind of artifacts. Figure 1 illustrates this problem for bicubic interpolation.

Notice that in order to use our approach, the estimation of the A^+ operator prior to training is required. In [21], this operator is modeled using a convolution operation followed by a subpixel shuffle layer [10]. The unknown parameters w are estimated by minimizing, via stochastic gradient descent, a loss function, that is,

$$\begin{aligned}\hat{\omega} =& \arg\min_\omega \mathbb{E}_x \| Ax - AA^+_\omega(Ax)\|_2^2 \\ &+ \mathbb{E}_y \| A^+_\omega(y) - A^+_\omega(AA^+_\omega(y))\|_2^2,\end{aligned} \tag{4}$$

where A^+_ω denotes the pseudo-inverse with ω network parameters.

An obvious disadvantage of this approach is that one needs to learn a specific $A^+_{\hat{w}}$ for each A. In order to have a single network robust to multiple A operators, we have implemented a network that, for any given A, it predicts the corresponding $\hat{\omega}$ of its pseudoinverse. We choose this network to be composed of three hidden layers with 512, 1024 and 512 hidden units. The input to this network is the PCA representation of the kernel k. The network is trained to predict the unknown $\hat{\omega}$ which solves Eq. 4 for a given A. Our experiments demonstrated that the performance obtained by this efficient approach was equivalent to calculating $\hat{\omega}$ for each A by individually solving for each operator according to Eq. 4.

Let us now see how $\mathrm{g}_\theta(\mathbf{y})$ is obtained using an enhanced formulation of a GAN model with increased perceptual quality.

B. A GAN model with increased perceptual quality

Taking into account that the transformation $\mathrm{g}_\theta(\mathbf{y})$ defines (from the distribution on $\mathbf{y}$) a probability distribution function $q_\theta(.)$ on the set of HR images, the Kullback-Leibler divergence between $q_\theta(.)$ and the distribution of real images $p_X(.)$, given by

$$\mathrm{KL}(q_\theta || p_X) = \int q_\theta(x) \log \frac{q_\theta(x)}{p_X(x)} dx, \tag{5}$$

is minimized using a GAN approach. This model has a maximum a posteriori approximation interpretation (see [21] for the details).

Together with the generative network, $\mathrm{g}_\theta(\mathbf{y})$, we learn a discriminative one, $\mathrm{d}_\phi(x)$, using the following two functions on θ and ϕ.

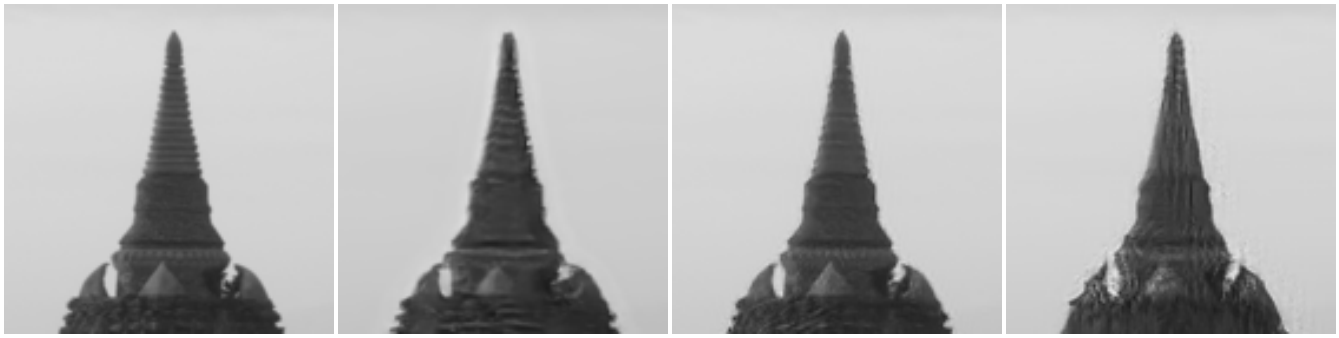
$$\mathrm{L_d}(\phi;\theta) = -\mathbb{E}_{x\sim \mathrm{X}}\left[\log \mathrm{d}_\phi(x)\right] - \mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}\left[\log(1 - \mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y})))\right] \tag{6}$$

$$\mathrm{L_g}(\theta;\phi) = -\mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}\left[\log \frac{\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}{1 - \mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}\right], \tag{7}$$

(a) HR (b) Bicubic (c) $\sigma = 2.0$ (d) $\sigma = 3.0$

Fig. 1: Example of artifacts introduced by the bicubic downsampling for a scaling factor of 3. (a) original image, (b) bicubically downsampled image in (a), (c) and (d) bicubically downsampled images which have previously been blurred with $\sigma = 2$ and $\sigma = 3$ Gaussian kernels respectively. The downsampled images have been enlarged to the size of the original one using bicubic upsampling.

where X and Y are the distribution of the HR and LR images, respectively. Iteratively, the algorithm updates ϕ by lowering $\mathrm{L_d}(\phi;\theta)$ while keeping θ fixed, and updates θ by lowering $\mathrm{L_g}(\theta;\phi)$ while keeping ϕ fixed.



(a) HR (b) VSRResGAN factor 2 (c) VSRResAffGAN factor 2 (d) VSRResAffGAN factor 4

Fig. 2: Example of the artifacts produced when only the adversarial loss is used for our GAN model (VSRResAffGAN) and VSRResGAN[17]. We can see how our model is able to recover the frame for factor 2 while VSRResGAN produces a lot of artifacts and a blurred frame. This issue is addressed in our GAN model by the addition of auxiliary losses to the standard GAN loss.

Notice that because the difference between $\mathbb{H}[q_\mathrm{G}, p_\mathrm{X}]$ (cross-entropy) and $\mathrm{KL}[q_\mathrm{G}|p_\mathrm{X}]$ is $\mathbb{H}[q_\mathrm{G}]$, this approximation is expected to lead to solutions with higher entropy and thus produce more diverse frames (see [21]). As can be seen in Fig. 2, this solution leads to a good results for factor 2, however, this is not the case for larger scale factors such as 3 and 4, where the GAN failed to converge. This is most likely caused by the discriminator's ability to easily distinguish between real and generated frames (furthermore, notice that the generator has to produce 16 pixels in the HR frame for each pixel in the input LR frame, which is a considerably more challenging task).

To address this issue, we regularize [36] the training of our GAN network following the approach described in [17] and use the Charbonnier loss between two images u and v (in a given space) defined as

$$\gamma(u, v) = \sum_k \sum_i \sum_j \sqrt{(u_{k,i,j} - v_{k,i,j})^2 + \epsilon^2}. \tag{8}$$

The Charbonnier loss is calculated in both pixel and feature spaces. We define our feature space to correspond to the activations provided by a CNN trained for discriminative tasks. For our model, we use the 3rd and 4th convolutional layers of VGG-16 [37] (denoted as VGG(.)).

The generator loss then becomes:

$$\begin{aligned} \mathrm{L_{g\,combined}}(\theta;\phi) &= \alpha \sum_{(x,\mathbf{y})\in T} \gamma(\mathrm{VGG(x)}, \mathrm{VGG}(\mathrm{g}_\theta(\mathbf{y}))) \\ &+ \beta \mathbb{E}_\mathbf{y}\left[\log \frac{1 - \mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}{\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}\right] + (1 - \alpha - \beta) \sum_{(x,\mathbf{y})\in T} \gamma(x, \mathrm{g}_\theta(\mathbf{y})), \end{aligned} \tag{9}$$

where $\alpha, \beta > 0$, $\alpha + \beta < 1$ and T is the dataset formed by pairs of LR sequences $\mathbf{y}$ and HR images x.

While the use of this loss successfully produces sharper frames, it also introduces high frequency artifacts, especially in smooth areas of the image. They are easily detectable and unpleasant to the human eye (see Fig. 3 for examples of such artifacts). While increasing the weight of the pixel-content loss ($\sum_{(x,\mathbf{y})\in T} \gamma(x, g_\theta(\mathbf{y}))$) significantly reduces these artifacts, it also smoothes the frame.

Because these artifacts are more prominent in the smooth regions of the frame, we propose the substitution of the pixel-content loss ($\sum_{(x,\mathbf{y})\in T} \gamma(x, g_\theta(\mathbf{y}))$) with a spatial smoothness constraint. This spatial smoothness constraint is calculated with a weight matrix $M(x)$ that assigns a larger weight to the pixel-content loss in the smooth areas of the real HR frame x during training. With the incorporation of this spatial smoothness constraint, the generator is penalized heavier during training when generating unwanted "noise" in smooth regions of the frame. We compute this weight matrix as $M(x) = 1 - S(x)$, where $S(x)$ is the Sobel operator applied to the image x. Therefore, the new proposed loss for the generator becomes:

$$\begin{aligned} L_{g\,\text{combined smooth}}(\theta;\phi) &= \alpha \sum_{(x,\mathbf{y})\in T} \gamma(\text{VGG}(x), \text{VGG}(g_\theta(\mathbf{y}))) \\ &+ \beta[\mathbb{E}_{\mathbf{y}}[\log \frac{1 - d_\phi(g_\theta(\mathbf{y}))}{d_\phi(g_\theta(\mathbf{y}))}]] + (1 - \alpha - \beta) \sum_{(x,\mathbf{y})\in T} M(x) \odot \gamma(x, g_\theta(\mathbf{y})), \end{aligned} \tag{10}$$

where $\alpha, \beta > 0$ and $\alpha + \beta < 1$ and $\odot$ denotes element-wise multiplication. $L_{g\,\text{combined smooth}}(\theta;\phi)$ is the generator loss that we use to train our GAN model.

In the next section, we describe in detail the CNN architecture used to approximate $f_\theta(\cdot)$.

(a) HR (b) VSRResFeatGAN[17]

Fig. 3: Examples of artifacts produced by VSRResFeatGAN[17] for scale factor 3. Notice the high frequency artifacts on smooth areas of the image. With the introduction in the loss function of the spatial smoothness constraint term these artifacts are considerably reduced.

C. Architecture

To implement $f_\theta(\cdot)$, from which we will obtain $g_\theta(y)$ using Eq. 2 which in turn will be used in Eq. 10, we adapt the VSRResNet model introduced in [17]. The authors of [17] found that the VSRResNet architecture results in state-of-the-art performance on the VideoSet4 dataset [38], the test dataset commonly used for evaluating VSR models. The VSRResNet model corresponds to a deep residual CNN that consists of three 3×3 convolutional layers each followed by a ReLU activation, 15 Residuals Blocks with no batch normalization and a final 3×3 convolutional layer. Padding is used at each convolution step in order to keep the spatial extent of the feature maps fixed across the network.

We note here that using as input the bicubically upsampled frames as in [17] is not well suited for the multiple degradation setting established in our work. Bicubic upsampling over-smoothes the images and introduces artifacts, making the learning process more difficult. Furthermore, as shown in Fig. 1, these artifacts are degradation operator dependent. Instead, we decided to input the LR video sequence to the network and use

the sub-pixel shuffle layer introduced in [10] in the network architecture to learn the up-scaling operation. This avoids the previously mentioned problem and increases training and inference speed.

Figure 4 shows the proposed architecture. Based on the VSRResNet[17], our model is a deep residual CNN with three 3×3 convolutional layers each followed by a ReLU activation, 15 Residuals Blocks with no batch normalization and a final 3×3 convolutional layer. However, instead of using a bicubically interpolated frame as an input, we upscale the feature maps using the sub-pixel shuffle layer. As can be seen, we do not add the sub-pixel shuffle layer at the end of the network as in [10], but rather we introduce it between the residual blocks of the network. This allows our network to extract features in the LR and HR spaces, increasing its performance. Furthermore, in the case of higher scaling factors like 4 and 8, we introduce several of these layers to allow us to perform progressive upsampling. As shown in Section IV-B the progressive upsampling results in a significant increase in performance. We use only one sub-pixel shuffle layer before the 10th Residual Block for factors 2 and 3. For factor 4 we use a sub-pixel shuffle layer with upscale factor 2 before the 5th and 10th residual blocks and for factor 8 we use an additional one before the last convolutional layer (see Section IV-A for details on model training).

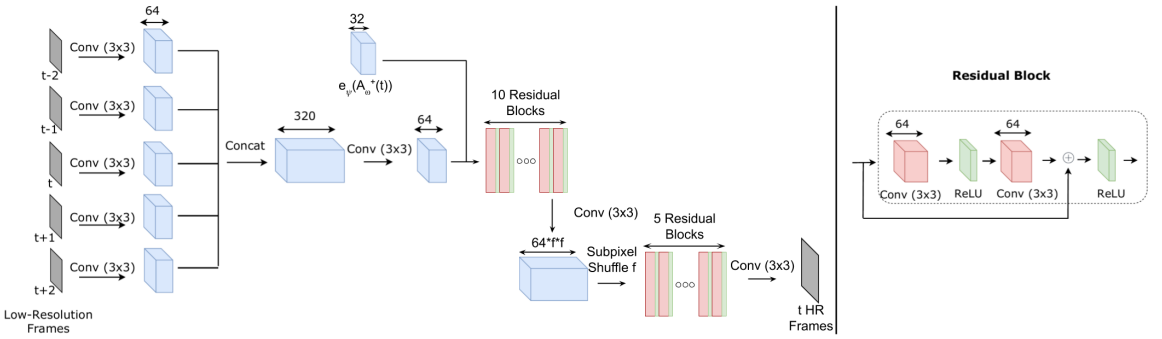


Fig. 4: The MD-AVSR architecture based on VSRResNet[17]. The network consists of a series of convolution operations with 64 kernels of size 3×3, applied to each input frame. The resulting feature maps are then concatenated to obtain 320 feature maps. This is followed by two convolution operations and 15 residual blocks. Each residual block consists of two convolutional operations with 64 kernels of size 3×3, each followed by a ReLU layer. Following the definition of a residual block, the inputted feature maps are added to the output feature maps to obtain the final output of the residual block. For scaling factors 2 and 3, before the 10th Residual block, we up-scale the feature maps by a factor f using a sub-pixel shuffle layer [10]. In the case of factor 4, a sub-pixel shuffle layer with upscale factor 2 is introduced before the 5th and 10th Residual Blocks and an additional one is used before the last convolution for scale factor 8.

The architecture defined above still suffers from a major problem: the parameters of the network θ depend on the choice of A. Although we ease the training procedure by only predicting the residual using $(I-A^+_{\hat{\omega}}A)\,f_\theta(\mathbf{y})$, the network parameters are dependent on $A^+_{\hat{\omega}}y$. In the MDVSR setting, it is necessary for the network parameters to be independent of A. This will allow any input video sequence to be provided to the trained network at test time. To this end, we modify the network architecture such that knowledge of A is provided. This will allow the network parameters to be learned for all As and to adapt to any given A at test time. More specifically, we encode $A^+_{\hat{\omega}}y$ using a network $\mathrm{e}_\psi(\cdot)$ (e_ψ in Fig. 4) and feed this compressed representation of $A^+_{\hat{\omega}}y$ to the CNN. This encoder network $\mathrm{e}_\psi(\cdot)$ seeks to extract the relevant and significant information from the degradation operator to guide the SR process. Notice that other approaches, such as utilizing the Principal Components of k (see [13]) may be considered. However, we argue that using an encoder network is more appropriate as more useful information can be extracted than by using a PCA representation. Our encoder $\mathrm{e}_\psi(A^+_{\hat{\omega}}y)$ consists of three convolutions of 3×3 and 32 filters with zero-padding, followed by the ReLU activation. Our best results were obtained by incorporating $\mathrm{e}_\psi(A^+_{\hat{\omega}}y)$ by concatenating the resulting feature maps before the first residual block of the VSRResNet architecture, as shown in Fig. 4. To ensure that the spatial size matches that of $\mathbf{y}$ we use a convolution stride equal to the scaling factor used for training. We jointly train the encoder $\mathrm{e}_\psi(\cdot)$ and the super-resolving $\mathrm{f}_\theta(\cdot)$ network.

IV. Experimental Results

In this section we detail the training process of the proposed model and show that our proposed approach significantly outperforms current state-of-the-art models for bicubic degradation in addition to being robust

to variation on the Gaussian blur deviation.

A. Training hyper-parameters

We use two datasets to train our models: The training sequences from the Myammar dataset and a second one extracted from a subset of the YouTube-8M dataset, which will be used to refine our best performing model.

The Myammar training dataset is formed by 10^6 patches of size 48×48 pixels. Patches with variance less than 0.0035 were determined to be uninformative and were hence removed from the dataset. For each HR patch at time t, we obtain the corresponding LR sequence of patches at time $t-2$, $t-1$, t, $t+1$, and $t+2$. These LR patches are obtained following Eq. 1, i.e., by first blurring the HR frames with a Gaussian kernel and then downsampling the images using bicubic downsampling.

We can distinguish two phases during training. During the first one, only the Myanmar training sequences are used. This phase consists of training the Generator using only the MSE loss ($\mathbb{E}_{x,\mathbf{y}}[\|x - \mathrm{g}_\theta(\mathbf{y})\|^2]$) for 100 epochs. We use the Adam optimizer [39] with the learning rate set to 10^{-3} for the first 50 epochs and then divided by 10 at the 50th and 75th epochs. The weight decay parameter was set to 10^{-5} for all our trained models.

Finally, the second phase uses the proposed GAN framework in Eq. 10. We fine-tuned the model already trained with MSE during the first phase for 30 epochs. The learning rate and weight decay for the discriminator are set to 10^{-4} and 10^{-3}, respectively, while for the generator they are both set to 10^{-4}. The learning rate of both the generator and discriminator are divided by 10 after 15 epochs. During this phase, the models are trained using 1,306,844 blocks of size 48×48 from 5 consecutive images extracted from a subset of YouTube-8M dataset. This subset was constructed by randomly selecting a total of 4358 videos. We consider all categories except those corresponding to video games and cartoons since these categories do not provide an accurate representation of natural scenes we are interested in recovering.

The complete training process takes roughly three days using a Nvidia Titan X GPU with 12 GB of RAM.

B. Ablation study of the proposed modifications of the architecture

To determine the contribution of each proposed component of the new architecture, we perform an ablation study by testing their effects adding one at a time. For ease of comparison, in these experiments, we will use neither the GAN framework nor the perceptual losses, i.e., we will focus on the first phase only (see Section IV-A). For training, we fix the degradation operator to correspond to the bicubic downsampling (no Gaussian blur). Because we only use one degradation for these experiments, we temporarily remove the use of $\mathrm{e}_\psi(A^+_{\hat{\omega}} y)$) from our architecture and instead use as input the LR $\mathbf{y}$ only.

We name the model that uses $\mathrm{g}_\theta(\cdot)$, i.e., the model that utilizes an affine projection, Affine VSR (AVSR) and the model that uses $\mathrm{f}_\theta(\cdot)$ No Affine VSR (NoAVSR) (i.e., it minimizes $\mathbb{E}_{x,\mathbf{y}}\|x - f_\theta(\mathbf{y})\|^2$). Both models include the sub-pixel shuffle layer. To determine the contribution of the sub-pixel shuffle layer in the affine network, we also train an architecture similar to AVSR but using the bicubic up-scaled frames at the input instead of using the subpixel shuffle layer to up-scale the frames. We name this model Bicubic-AVSR. Finally, to test the effects of using the subpixel shuffle layer in a non-affine network, we compare NoAVSR to the a non-affine architecture that uses the bicubic up-scaled frames at the input. We call this model Bicubic-NoAVSR. Notice Bicubic-NoAVSR is equivalent to VSRResNet[17].

Table I contains a quantitative comparison of these models for multiple degradations and up-scaling factors 2, 3 and 4 on the Myanmar video test sequences. We also include a study of the time complexity of each method using the number of Multiplication and Additions (MACs) as a metric. The smaller the number of MACs, the faster the algorithm is. From this table, it is clear that the proposed affine networks (Bicubic-AVSR and AVSR) significantly outperform the other models with a slight increase in MACs. Moreover, AVSR outperforms Bicubic-AVSR in all factors and degradations, showing that, in the case of affine networks, the use of the subpixel shuffle layer is always better than using bicubic interpolation, not only in terms of speed (as reflected by the significant reduction in MACs), but also in terms of fidelity. This experiment shows that even the best performing model, AVSR, is not robust to the use, during testing, of images which have been degraded with Gaussian blur when trained only with bicubic downsampling.

Notice that we did not perform experiments analyzing key components of the underlying residual architecture since it is based on VSRResNet and those elements were studied in [17].

TABLE I: Comparison of the proposed and state-of-the-art models on Myanmar video test sequences and factors 2, 3, and 4. The models were trained using only bicubic degradation. σ refers to the Gaussian blur deviation used during testing. The time complexity of each method is indicated by the number of Multiplications and Additions (MACs) performed for each frame (lower is better).

	Factor	PSNR/SSIM			MACs
		Bicubic	$\sigma = 1.3$	$\sigma = 2.6$	
Bicubic-NoAVSR (VSRResNet[17])	×2	40.58/0.9807	31.97/0.9058	27.85/0.7930	$6.73 * 10^{11}$
	×3	35.97/0.9481	31.77/0.8968	27.78/0.7918	$6.73 * 10^{11}$
	×4	32.85/0.9075	30.73/0.8700	27.64/0.7868	$6.73 * 10^{11}$
NoAVSR	×2	40.52/0.9792	31.97/0.9051	27.85/0.7932	$\mathbf{3.31 * 10^{11}}$
	×3	35.92/0.9474	31.92/0.8965	27.79/0.7924	$\mathbf{2.65 * 10^{11}}$
	×4	33.31/0.9077	31.48/0.8720	27.69/0.7881	$\mathbf{2.82 * 10^{11}}$
Bicubic-AVSR	×2	40.69/0.9811	32.59/0.9162	27.90/0.7935	$6.73 * 10^{11}$
	×3	36.09/0.9487	32.35/0.9065	27.80/0.7925	$6.73 * 10^{11}$
	×4	33.38/0.9077	31.95/0.8902	27.71/0.7883	$6.73 * 10^{11}$
AVSR	×2	**41.23/0.9833**	**32.75/0.9261**	**28.34/0.8180**	$\mathbf{3.31 * 10^{11}}$
	×3	**36.38/0.9527**	**32.53/0.9070**	**27.93/0.8043**	$\mathbf{2.65 * 10^{11}}$
	×4	**33.89/0.9170**	**32.18/0.8927**	**27.77/0.7965**	$\mathbf{2.82 * 10^{11}}$

C. Experiments with Multiple Degradations

Let us now address the multiple degradations scenario. As mentioned in Section I, one of the main factors that significantly lowers the robustness of CNN-based methods to different degradation is the use of only bicubic downsampling during training. Therefore, we will now train the following models with multiple degradations. The degradations considered here are a combination of Gaussian blurs with different kernels k and bicubic downsampling. We generated random Gaussian kernels with σ using a step of 0.1 in the range [0.2, 2.0] for factor 2, [0.2, 3.0] for factor 3, and [0.2, 4.0] for factor 4. The HR video sequences are blurred with these kernels and bicubic downsampling is applied to them to generate the LR samples. We retrain AVSR using these degradations and call this model Blind-MD-AVSR. As stated in Section III-C, we expect it to have a significant loss in performance compared to the one degradation case, since the parameters of the network θ depend on not only $\mathbf{y}$ but also A^+y and this information is not provided to the network. This issue is resolved with the introduction in the architecture of our encoding network $\mathrm{e}_\psi(A^+_{\hat{\omega}}y)$), as explained in Section III-C. We call this affine network ($\mathrm{g}_\theta(\cdot)$) that incorporates the encoded information $\mathrm{e}_\psi(A^+_{\hat{\omega}}y)$), MD-AVSR.

We compare our model with current state-of-the-art (SOTA) methods for SR with multiple degradations: IRCNN[32] and SRMDNF[13]. The authors of IRCNN[32] propose the use of ADMM to split the SR problem into two subproblems: a regularized recovery one(subproblem A) and a denoising one (subproblem B). To combine learning and analytical approaches they propose the use of a CNN for the denoising subproblem. SRMDNF[13] consists of a CNN for SR which takes as input the LR image and PCA(k) to provide the network information about the degradation, making it robust to multiple degradations. Notice that all previous SOTA methods are designed for Single Image Super-Resolution (SISR). As mentioned in Section I, this work is the first one to propose a VSR CNN-based method for multiple degradations. Therefore, to provide a fair comparison, we adapt SRMDNF[13] to VSR using our non-affine network ($\mathrm{f}_\theta(\cdot)$). We have experimentally determined that the optimal place to add PCA(k) is before the first residual block. We trained this network as we did with MD-AVSR. We call this model VSRMDNF.

Table II shows an experimental comparison with current SOTA models for multiple degradations and factors 2, 3, and 4 on the Myanmar video test sequences. Blind-MD-AVSR has a similar performance across the different degradations, showing that training with multiple degradations significantly increases robustness. However, we observe a sharp decrease in performance for bicubic degradation compared to AVSR (see Table I). This is not the case for the proposed MD-AVSR, which is able to outperform all the other models for all values of σ considered, see also Table I. This indicates the need to incorporate the knowledge about the degradation information if we intend to utilize the same network with multiple degradations (adding them to the training

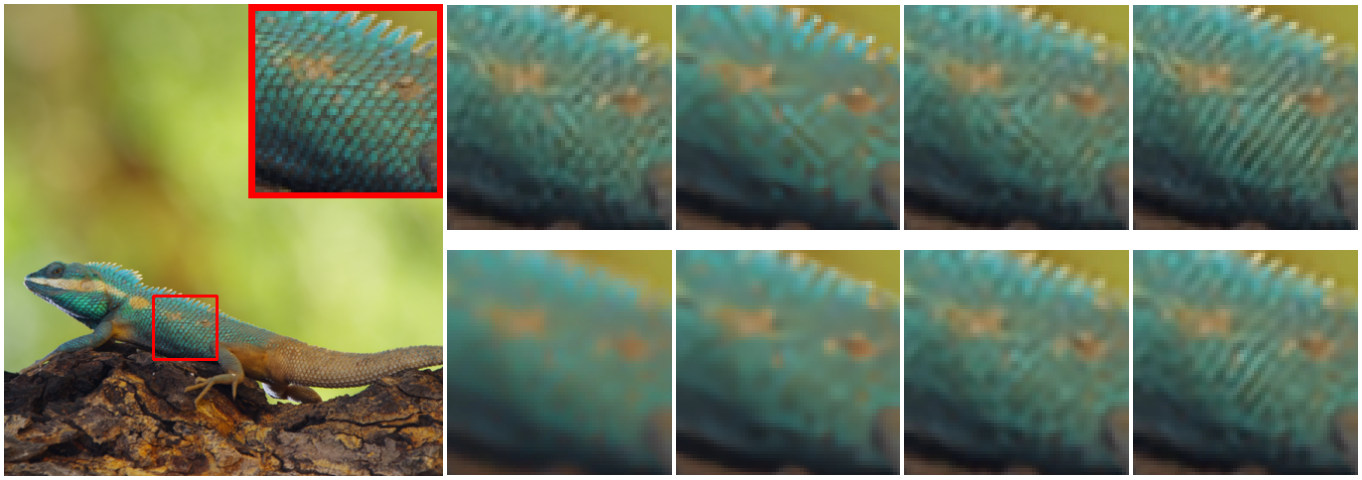


Fig. 5: Qualitative results of our MD-AVSR model compared to current state-of-the-art methods for factor 3 using bicubic downsampling (first row) and Gaussian blur with $\sigma = 2.0$ and bicubic downsampling (second row). Notice how MD-AVSR recovers more details compared to the rest.

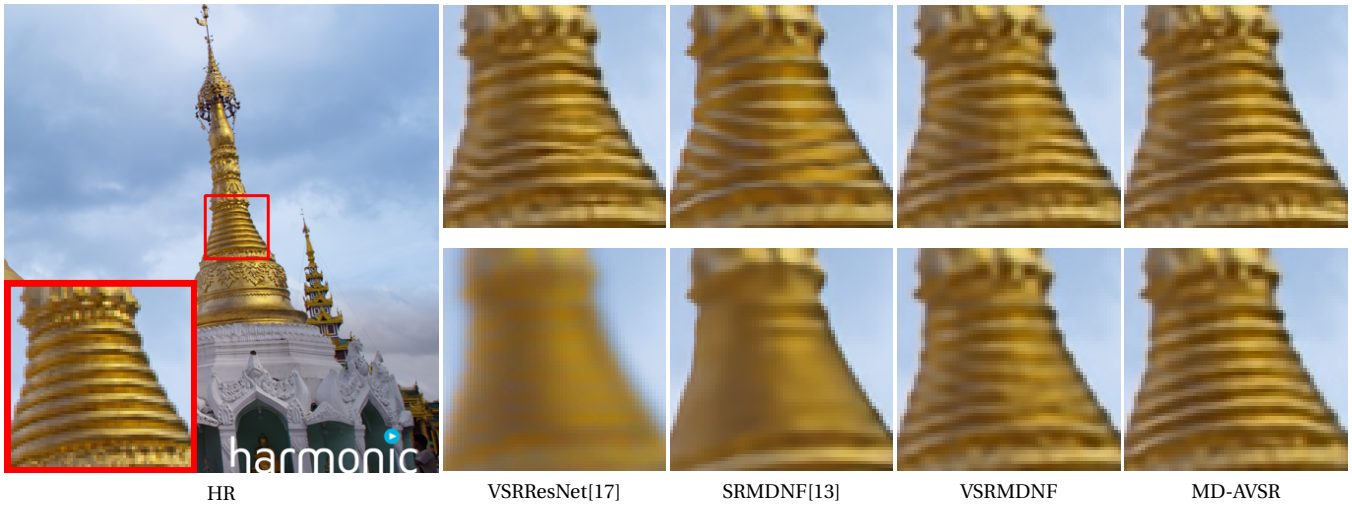


Fig. 6: Qualitative results of our MD-AVSR model compared to current state-of-the-art methods for factor 3 using bicubic downsampling (first row) and Gaussian blur with $\sigma = 3.0$ and bicubic downsampling (second row). The pattern highlighted in the HR image is difficult to recover when only bicubic downsampling is used because the lines either disappear or do not look straight (see Fig. 1). In this case, our MD-AVSR is able to almost fully recover the correct structure while the rest struggle. Furthermore, it is able to fully recover the correct pattern when a strong Gaussian blur of $\sigma = 3.0$ is used during acquisition, while the other methods fail to do so.

phase is not enough). See also the good performance of the other model which uses information of the degradation process (VSRMDNF). Notice also that MD-AVSR outperforms VSRMDNF in a similar manner as AVSR outperforms NoAVSR and VSRResNet. This indicates that the benefits of using the image formation model are carried over to the multiple degradation setting. It is important to observe in Tables I and II that AVSR and MD-AVSR are the best performing methods when compared to similar approaches. Furthermore, for bicubic downsampling MD-AVSR slightly outperforms AVSR which is expected to deal with this degradation only. Notice also that although the performance of MD-AVSR slightly deteriorates when blur is introduced, it is more robust than its AVSR counterpart, in other words, MD-AVSR can be safely implemented in systems that are expected to deal with video sequences degraded by different acquisition models. However, this comes with an increase in the number of MACs due to the inclusion of $e_\psi(A^+_\omega y)$).

A qualitative comparison of VSRResNet, SRMDNF[13], VSRMDNF and MD-AVSR can be seen in figures 5 and 6. This comparison reveals how our methods are more robust to the multiple degradations and produce HR images of higher quality.

TABLE II: Comparison of the proposed and state-of-the-art models on Myanmar video test sequences and factors 2, 3, and 4. Models were trained using multiple degradations. σ refers to the Gaussian blur deviation used. The time complexity of each method is indicated by the number of Multiplications and Additions (MACs) performed for each frame (the lower the better).

	Factor	PSNR/SSIM			MACs
		Bicubic	$\sigma = 1.3$	$\sigma = 2.6$	
IRCNN[32]	×2	37.35/0.9589	35.98/0.9304	26.00/0.4927	$2.94 * 10^{12}$
	×3	34.41/0.9240	34.38/0.9222	31.58/0.8304	$2.94 * 10^{12}$
	×4	31.56/0.8820	31.57/0.8814	31.06/0.8639	$2.94 * 10^{12}$
SRMDNF[13]	×2	39.01/0.9697	38.63/0.9656	35.17/0.9268	$\mathbf{1.96 * 10^{11}}$
	×3	35.08/0.9299	35.12/0.9289	34.03/0.9082	$\mathbf{8.81 * 10^{10}}$
	×4	32.98/0.8952	33.01/0.8949	32.80/0.8889	$\mathbf{5.03 * 10^{10}}$
Blind-MD-AVSR	×2	37.65/0.9523	37.31/0.9503	37.10/0.9458	$3.31 * 10^{11}$
	×3	34.97/0.9330	34.59/0.9275	34.27/0.9200	$2.65 * 10^{11}$
	×4	33.08/0.9002	32.96/0.8992	32.81/0.8972	$2.82 * 10^{11}$
VSRMDNF	×2	40.60/0.9809	39.78/0.9741	37.51/0.9596	$3.33 * 10^{11}$
	×3	35.95/0.9471	35.67/0.9415	34.99/0.9318	$2.65 * 10^{11}$
	×4	33.37/0.9077	33.05/0.9043	32.86/0.8992	$2.82 * 10^{11}$
MD-AVSR	×2	**41.25/0.9834**	**40.11/0.9778**	**37.79/0.9626**	$3.44 * 10^{11}$
	×3	**36.52/0.9525**	**36.17/0.9480**	**35.25/0.9355**	$2.76 * 10^{11}$
	×4	**34.01/0.9185**	**33.86/0.9153**	**33.30/0.9025**	$2.93 * 10^{11}$

D. Experiments using Perceptual losses

MD-AVSR MD-AVSR-FG MD-AVSR-P

Fig. 7: Qualitative comparison between MD-AVSR, MD-AVSR-FG and MD-AVSR-P for factor 4 with bicubic downsampling. We can see that MD-AVSR-P does not produce artifacts like MD-AVSR-FG does and produces frames that look much sharper than MD-AVSR.

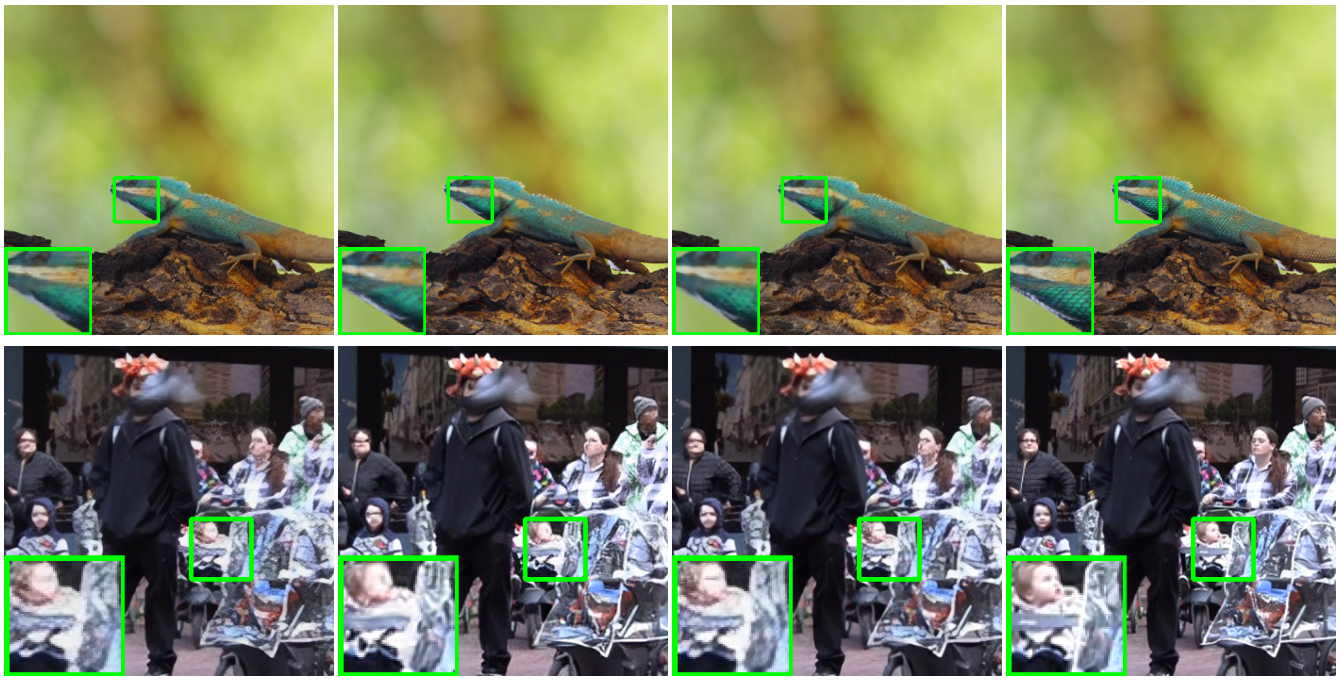
Let us now test the GAN framework (see Eq. 10) and analyze the performance of the new loss in comparison to the one proposed in [17] (see Eq. 9). This corresponds to the 2nd phase of the training described in Section IV-A. We first use during this 2nd phase the loss proposed in [17], to fine-tune the MD-AVSR model with

TABLE III: Comparison with state-of-the-art methods on the VideoSet4 [38] dataset on scale factors 2, 3, 4 and 8. The comparison is done in terms of PSNR, SSIM, Perceptual Distance as defined in [40] and the number of Multiplications and Additions (MACs) required for each frame. A smaller Perceptual Distance implies better perceptual quality. MACs indicates the time complexity of the method (lower is better).

	Factor	PSNR	SSIM	PercepDist	MACs
VDSR[35]	×2	31.61	0.9335	0.0541	$2.52 * 10^{11}$
	×3	26.65	0.8091	0.1355	$2.52 * 10^{11}$
	×4	25.05	0.7292	0.1860	$2.52 * 10^{11}$
RCAN[41]	×2	32.58	0.9414	0.0519	$1.45 * 10^{12}$
	×3	27.71	0.8404	0.1180	$6.52 * 10^{11}$
	×4	25.44	0.7405	0.1695	$3.77 * 10^{11}$
	×8	22.33	0.5053	0.3277	$1.07 * 10^{11}$
VESPCN[27]	×3	27.25	0.8253	0.1533	$\mathbf{5.74 * 10^{9}}$
	×4	25.35	0.7309	0.2022	$\mathbf{3.27 * 10^{9}}$
SPMC-SR[29]	×2	30.92	0.9235	0.0899	$\mathbf{4.97 * 10^{10}}$
	×3	27.49	0.84	-	$4.88 * 10^{10}$
	×4	25.63	0.7709	0.1908	$4.85 * 10^{10}$
TAN[34]	×4	25.53	0.7475	0.1798	-
VSRResNet[17]	×2	31.87	0.9426	0.0407	$4.91 * 10^{11}$
	×3	27.80	0.8571	0.1209	$4.91 * 10^{11}$
	×4	25.51	0.7530	0.1766	$4.91 * 10^{11}$
	×8	22.35	0.5072	0.3286	$4.91 * 10^{11}$
VSRResFeatGAN[17]	×2	30.90	0.9241	0.0283	$4.91 * 10^{11}$
	×3	26.53	0.8148	0.0668	$4.91 * 10^{11}$
	×4	24.50	0.7023	0.1043	$4.91 * 10^{11}$
	×8	22.12	0.5025	0.2773	$4.91 * 10^{11}$
ERSGAN[16]	×4	22.98	0.6336	0.0993	$4.24 * 10^{11}$
MD-AVSR	×2	**33.00**	**0.9496**	0.0292	$2.51 * 10^{11}$
	×3	**28.31**	**0.8751**	0.1081	$2.01 * 10^{11}$
	×4	**26.17**	**0.7895**	0.1655	$2.13 * 10^{11}$
	×8	**22.74**	**0.5126**	0.3243	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-FG	×2	31.54	0.9309	0.0229	$2.51 * 10^{11}$
	×3	26.73	0.8237	0.0588	$2.01 * 10^{11}$
	×4	25.10	0.7414	0.0939	$2.13 * 10^{11}$
	×8	22.28	0.5047	0.2715	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-P	×2	31.82	0.9345	0.0216	$2.51 * 10^{11}$
	×3	27.20	0.8383	0.0586	$2.01 * 10^{11}$
	×4	25.26	0.7501	0.0927	$2.13 * 10^{11}$
	×8	22.36	0.5056	0.2708	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-PY8	×2	31.81	0.9391	**0.0210**	$2.51 * 10^{11}$
	×3	27.09	0.8412	**0.0571**	$2.01 * 10^{11}$
	×4	25.18	0.7555	**0.0916**	$2.13 * 10^{11}$
	×8	22.31	0.5064	**0.2699**	$\mathbf{7.16 * 10^{10}}$

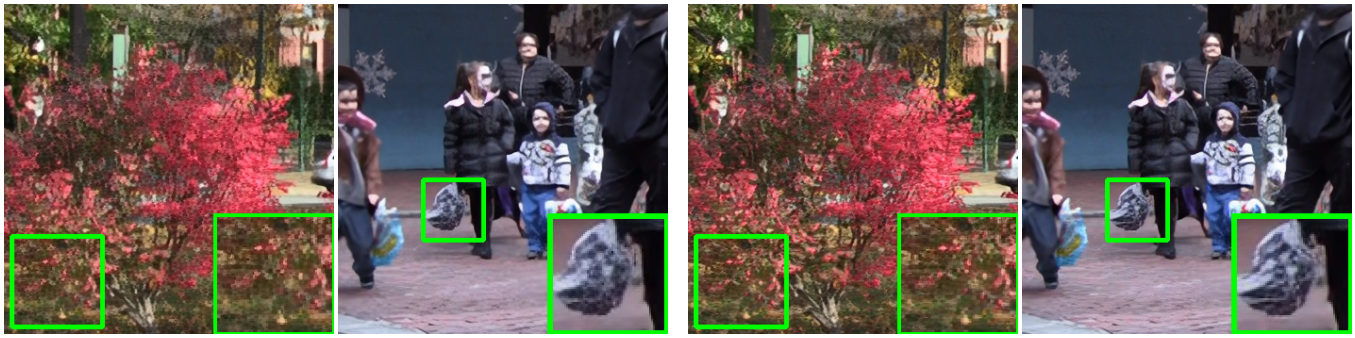
$\alpha = 0.998$ and $\beta = 0.001$ for the hyper-parameters of the loss function. We call this model, which incorporates features loss (F) and a GAN (G), MD-AVSR-FG. Then, we fine-tuned in the same fashion the MD-AVSR model with the new proposed perceptual (P) loss (see Eq. 10) and call it MD-AVSR-P. The values of the loss hyper-parameters are: $\alpha = 0.049$ and $\beta = 0.001$. Both models are trained using the Myammar dataset, instead of the YouTube-8M dataset. Finally, to demonstrate the influence of the diversity of the training dataset for GAN-based VSR models, we use the YouTube-8M dataset and our new loss in Eq. 10 to fine-tune MD-AVSR and obtain our final model MD-AVSR-PY8.

Table III contains a comparison of these models with multiple state-of-the-art methods in terms of PSNR, SSIM, Perceptual Distance [40] and the number MACs per frame for the VideoSet4 video sequences [38] and factors 2, 3, 4 and 8. These include SISR methods with Deep CNN like VDSR[35], RCAN[41] and ERSGAN[16].



VSRResFeatGAN[17] ERSGAN[16] MD-AVSR-P HR

Fig. 8: Comparison between VSRResFeatGAN[17], ERSGAN[16] and MD-AVSR-P for factor 4 with bicubic downsampling.



MD-AVSR-P MD-AVSR-PY8

Fig. 9: Comparison between MD-AVSR-P and MD-AVSR-PY8 for factor 4 with bicubic downsampling.

VDSR[35] uses a very deep convolutional neural network that predicts residuals between the upscaled low-resolution image and the high-resolution image. RCAN[41] consists of a very deep residual CNN that uses a new type of block that exploits channel attention to further increase the performance. ERSGAN[16] proposes the use of new residual-in-residual block to construct a very deep residual CNN and train it within a GAN framework to obtain photo-realistic SR images. Both methods use an architecture several times deeper than the one proposed here. We also include the following VSR methods: VESPCN[27], SPMC-SR[29], TAN[34] and VSRResNet and VSRResFeatGAN from [17]. VESPCN[27] incorporates time information in the network by using the motion compensated future and past frames in a time window and uses a sub-pixel convolution to upscale the output. SPMC-SR[29] proposes a convolution-LSTM neural network with efficient motion compensation learned jointly with the network. Lastly, TAN[34] uses a Temporal Adaptive Network that consists of a network with multiple SR branches, each responsible for super-resolving frames at a temporal scale. Then, a temporal modulation branch fuses the multiple VSR solutions into a single one. Notice that all these methods work only with bicubic downsampling, in contrast with SOTA methods in the previous section.

In Table III we observe how the proposed MD-AVSR-P outperforms all models trained with perceptual losses (VSRResFeatGAN[17], ERSGAN[16] and MD-AVSR-FG) for all figures of merits when trained on Myanmar

training sequences. Furthermore, a close examination of the generated frames and a comparison to the MD-AVSR-FG ones (see Fig. 7) shows that they are almost artifact-free. The model also shows a noticeable increase in sharpness compared to MD-AVSR. However, this increase in sharpness (reflected on the improvement on Perceptual Distance) comes with a decrease in PSNR and SSIM. This decrease aligns with the one shown in other models that use perceptual losses, like VSRResFeatGAN[17] or ERSGAN[16]. Figure 8 shows a qualitative comparison of VSRResFeatGAN[17] and ERSGAN[16] with MD-AVSR-P. We can see that the proposed model MD-AVSR-P outperforms current state-of-the-art methods. Notice that values for factors 2 and 3 for ERSGAN[16] could not be calculated since the authors did provide weights for those. Notice also how the proposed model outperforms, in terms of picture quality, models much more complex (with two times more MACs) like ERSGAN[16].

Table III also shows that MD-AVSR-PY8 outperforms MD-AVSR-P in terms of SSIM and Perceptual Distance significantly, although it suffers from a minor decrease in PSNR. This increase shows the importance of using a very diverse dataset during training for GAN models in contrast to experiments carried out with the non GAN model MD-AVSR, where the training with this new dataset did not produce results different enough to be significant for any factor. These results show that GAN-based VSR models require more data that other CNN. Figure 9 shows a qualitative comparison between MD-AVSR-P and MD-AVSR-PY8. It can be seen that MD-AVSR-PY8 is able to produce more detailed and realistic looking images.

V. Conclusions

In this work we have first introduced a multiple degradation Video Super-Resolution approach that explicitly utilizes the LR image formation model as an input to the network. The model, named MD-AVSR, has been trained with MSE only. The experiments show that MD-AVSR outperforms current state-of-the-art methods in terms of PSNR and SSIM for both multiple degradation and bicubic degradation only settings. We have then proposed a GAN-based approach that uses a new perceptual loss combining an adversarial loss, a feature loss, and a spatial smoothness constraint (MD-AVSR-P). This method improves the quality of the super resolved frames without the introduction of noticeable high frequency artifacts. The results show that it outperforms current state-of-the-art methods in terms of perceptual quality and in all metrics used by GAN methods trained without using the proposed perceptual loss. Finally, we use a much more diverse dataset created from a subset of the YouTube-8M dataset to train MD-AVSR-P and show that the new MD-AVSR-PY8 obtains significantly better results in terms of perceptual quality.

Acknowledgements

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

References

[1] C. Liu and D. Sun, "On Bayesian adaptive video super resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 346–360, 2014.

[2] A. K. Katsaggelos, R. Molina, and J. Mateos, "Super resolution of images and video," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 1, no. 1, pp. 1–134, 2007.

[3] S. P. Belekos, N. P. Galatsanos, and A. K. Katsaggelos, "Maximum a posteriori video super-resolution using a new multichannel image prior," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1451–1464, 2010.

[4] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 984–999, 2011.

[5] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.

[6] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, pp. 184–199, Springer, 2014.

[7] Hong Chang, Dit-Yan Yeung, and Yimin Xiong, "Super-resolution through neighbor embedding," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, pp. I–I, 2004.

[8] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[9] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, 2012.

[10] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874–1883, 2016.

[11] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint arXiv:1609.04802*, 2016.

[12] G. Riegler, S. Schulter, M. Rüther, and H. Bischof, "Conditioned regression models for non-blind single image super-resolution," in *IEEE International Conference on Computer Vision*, pp. 522–530, Dec 2015.

[13] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[14] 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.

[15] X. Wang, K. Yu, C. Dong, and C. Change Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018.

[16] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV Workshops*, 2018.

[17] A. Lucas, S. López-Tapia, R. Molina, and A. K. Katsaggelos, "Generative adversarial networks and perceptual losses for video super-resolution," *IEEE Transactions on Image Processing*, vol. 28, pp. 3312–3327, July 2019.

[18] 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.

[19] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, July 2017.

[20] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, "Compressed sensing mri reconstruction with cyclic loss in generative adversarial networks," *arXiv preprint arXiv:1709.00753*, 2017.

[21] C. Sonderby, J. Caballero, L. Theis, W. Shi, and F. Huszar, "Amortised MAP inference for image super-resolution," in *International Conference on Learning Representations*, 2017.

[22] S. Abu-El-Haija, N. Kothari, J. Lee, A. P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," in *arXiv:1609.08675*, 2016.

[23] R. Liao, X. Tao, R. Li, Z. Ma, and J. Jia, "Video super-resolution via deep draft-ensemble learning," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 531–539, 2015.

[24] 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.

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

[26] D. Li and Z. Wang, "Video super-resolution via motion compensation and deep residual learning," *IEEE Transactions on Computational Imaging*, 2017.

[27] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," *arXiv preprint arXiv:1611.05250*, 2016.

[28] O. Makansi, E. Ilg, and T. Brox, "End-to-end learning of video super-resolution with motion compensation," in *German Conference on Pattern Recognition*, pp. 203–214, Springer, 2017.

[29] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," *arXiv preprint arXiv:1704.02738*, 2017.

[30] M. Zareapoor, H. Zhou, and J. Yang, "Perceptual image quality using dual generative adversarial network," *Neural Computing and Applications*, pp. 1–11, 2019.

[31] P. Shamsolmoali, M. Zareapoor, R. Wang, D. K. Jain, and J. Yang, "G-ganisr: Gradual generative adversarial network for image super resolution," *Neurocomputing*, vol. 366, pp. 140 – 153, 2019.

[32] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," *CoRR*, vol. abs/1704.03264, 2017.

[33] S. Lopez-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Gan-based video super-resolution with direct regularized inversion of the low-resolution formation model," in *ICIP*, 2019.

[34] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2507–2515, 2017.

[35] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1646–1654, June 2016.

[36] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

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

[38] 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.

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

[40] 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.

[41] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV*, 2018.

Santiago López-Tapia received the bachelor's and master's degrees in computer science from the University of Granada in 2014 and 2015, respectively. He is currently pursuing the Ph.D. degree with the Visual Information Processing Group, Department of Computer Science and Artificial Intelligence, University of Granada. His research mainly focuses in the use of deep learning models for image restoration and classification.

Alice Lucas received the B.S. degree in applied math, engineering, and physics from the University of Wisconsin–Madison in 2015 and the M.S. degree in electrical engineering from Northwestern University in 2017, where she is currently pursuing the Ph.D. degree with the Image and Video Processing Laboratory (IVPL). Her research at IVPL is centered on the use of deep learning models for various image processing tasks, with focus on the task of video super-resolution (VSR). She received the Certificate in Computer Science from the University of Wisconsin–Madison in 2015.

Rafael Molina received the degree in mathematics and the Ph.D. degree in optimal design in linear models from the University of Granada, Granada, Spain, in 1979 and 1983, respectively. He was the Dean of the Computer Engineering School, University of Granada, from 1992 to 2002. In 2000, he joined the University of Granada, as a Professor of computer science and artificial intelligence. He was the Head of the Computer Science and Artificial Intelligence Department, University of Granada, from 2005 to 2007. His research focuses on using Bayesian modeling and inference in problems like image restoration, active learning, and machine learning.

Aggelos K. Katsaggelos received the Diploma degree in electrical and mechanical engineering from the Aristotelian University of Thessaloniki, Greece, in 1979, and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology in 1981 and 1985, respectively. In 1985, he joined the Department of Electrical Engineering and Computer Science, Northwestern University, where he is currently a professor. He was the Ameritech Chair of information technology and the AT&T Chair, and he is the Joseph Cummings Chair. He has published extensively in the areas of signal processing and communications, computational imaging, and machine learning (over 300 journal papers).