electronics

MDPI

Article

Video Super-Resolution Based on Generative Adversarial Network and Edge Enhancement

Jialu Wang, Guowei Teng * and Ping An

Department of Signal and Information Processing, School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China; wangjialu1005@163.com (J.W.); anping@shu.edu.cn (P.A.)
* Correspondence: tenggw@shu.edu.cn; Tel.: +8-6021-6613-5051

Abstract: With the help of deep neural networks, video super-resolution (VSR) has made a huge breakthrough. However, these deep learning-based methods are rarely used in specific situations. In addition, training sets may not be suitable because many methods only assume that under ideal circumstances, low-resolution (LR) datasets are downgraded from high-resolution (HR) datasets in a fixed manner. In this paper, we proposed a model based on Generative Adversarial Network (GAN) and edge enhancement to perform super-resolution (SR) reconstruction for LR and blur videos, such as closed-circuit television (CCTV). The adversarial loss allows discriminators to be trained to distinguish between SR frames and ground truth (GT) frames, which is helpful to produce realistic and highly detailed results. The edge enhancement function uses the Laplacian edge module to perform edge enhancement on the intermediate result, which helps further improve the final results. In addition, we add the perceptual loss to the loss function to obtain a higher visual experience. At the same time, we also tried training network on different datasets. A large number of experiments show that our method has advantages in the Vid4 dataset and other LR videos.

Keywords: video super-resolution; generative adversarial networks; edge enhancement

Citation: Wang, J.; Teng, G.; An, P. Video Super-resolution Based on Generative Adversarial Network and Edge Enhancement. *Electronics* **2021**, *10*, 459. https://doi.org/10.3390/electronics10040459

Academic Editors: Chun Sing Lai, Kim-Fung Tsang and Yinhai Wang
Received: 1 January 2021
Accepted: 5 February 2021
Published: 13 February 2021

1. Introduction

Super-resolution (SR) aims to reconstructing high-resolution (HR) images or videos from their low-resolution (LR) versions, which is a classic problem in computer vision. It not only pursues the enlargement of the physical size but also recovers high-frequency details to ensure clarity. Classical algorithms have existed for decades and can be divided into the following categories, methods based on patch [1], edge [2], sparse coding [3], prediction [4], and statistics [5]. These methods have lower computational cost than deep learning methods, but their recovery performance is also very limited. With the popularity of deep learning, convolutional neural networks have been widely applied and led to a dramatic leap in SR.

This field can be divided into two parts, single image super-resolution (SISR) and video super-resolution (VSR). The former exploits the spatial correlation in a single frame, while the latter additionally uses inter-frame temporal correlation. Digital video processing technology includes many fields, such as passive video forgery detection techniques [6–8]. In this article, we will focus on videos with lower resolution and blurry quality. To obtain HR data, the most direct way is to use HR cameras. However, due to the production process and engineering cost considerations, high-resolution cameras will not use for shooting in many cases, such as CCTV. Urban CCTV is helpful to security. However, in order to ensure the long-term stable operation of recording equipment and the appropriate frame rate of dynamic scenes, this product often sacrifices resolution to some extent. 1G of a 1080p video file can only record for less than half an hour at most. If it can only record for a short time, it loses the meaning of monitoring. However, we can improve the quality of CCTV through SR to obtain more information that is useful. In addition,

video SR is also used in the HR reconstruction of old movies and TV shows, such as Farewell My Concubine. Similar applications exist in the field of remote sensing and medical imaging. Moreover, SR also helps to improve the performance of other computer vision tasks, such as semantic segmentation [9]. Therefore, obtaining HR data through super-resolution (SR) technology has many practical applications and demands.

On the one hand, choosing the proper VSR algorithm is crucial. VSR was once divided into a large number of single multi-frame SR subtasks [10,11], which resulted in inevitable flicker artifacts and expensive calculations. In our work, as with mainstream algorithms, we use the previously reconstructed high-resolution results to SR the subsequent frames. Since the above methods ignore people's perception, some SR reconstruction results are still unsatisfactory. Therefore, Generative Adversarial Network (GAN) was introduced into the field of SR. GAN, which contains a generator (G) and a discriminator (D), is a popular deep learning-based model. G and D compete with each other during the training process so that the generated data obtained from the generator are as similar to the real data as possible. Goodfellow et al. [12] proposed GAN in 2014. After that, GAN has been applied to various computer vision problems, including SR. For example, a GAN for image SR (SRGAN) [13] uses adversarial loss and perceptual loss to recover photo-realistic textures from LR images. This type of network has excellent performance in reconstructing high-frequency details and can restore textures that are more realistic. However, it also has limitations. GAN will introduce noise and cause some details of the dislocation. Later, the comprehensive consideration of SR combined with other image enhancement methods [14,15] attracted people's attention. SR belongs to the big field of image enhancement. Both of their purpose is to improve people's perception. When SR increases the physical size, it will inevitably cause some discomfort such as blur, which can be improved by combining with other image enhancement methods. After the initial SR, the edge enhancement module is added, which will greatly help the image quality improvement.

On the other hand, methods based on deep learning are data-driven. Specifically, training requires a large amount of paired LR–HR data, which determines the reconstruction ability of the network to a certain extent. Generally, LR frames are degraded from a continuous set of HR frames by linear down-sampling (for example, bi-cubic degradation) or adding other noise on this basis, and formalized as (1) or (2):

$$y = (x \otimes k) \downarrow_s + n, \tag{1}$$

$$y = ((x \downarrow_s) \otimes k) + n, \tag{2}$$

where $\otimes$ represents the convolution operation, k represents the blur kernel, $\downarrow_s$ represents down-sampled operation, and n represents additive noise [16,17]. Then, the network is used to learn the mapping between low-resolution image y and high-resolution image x. However, the degradation process is more complicated or even unknown in the real world. Recently, many studies have been conducted on this issue [18–22]. In addition, the dataset may not match the actual LR scene. For example, the dataset is about landscapes, and the characters need to be reconstructed. In this article, we try to train the network on different datasets and test on different testing datasets.

Our main contributions in this paper can be summarized as follows:

1. We proposed an end-to-end GAN-based network for VSR, which focuses on videos with lower resolution and blurry quality.

2. The Laplacian edge module, which can enhance edges while suppressing noise, is added in the generator after SR to meet the needs of people's perception.

3. We trained and tested our method on different datasets.

Extensive experiments demonstrate the superiority of our method.

2. Related Works

While SR is a classical task, our review in this section focuses on deep learning-based methods for SISR and VSR.

2.1. Single Image Super-Resolution (SISR)

Given that Y is the low-resolution image, $F(Y)$ is the reconstructed image, and X is the corresponding ground truth HR image, the goal of SISR is to ensure that $F(Y)$ and X are as similar as possible.

Dong et al. [23] proposed a deep convolutional network for image SR (SRCNN), which introduced the convolutional neural network into the SR field for the first time. Subsequently, to accelerate the speed, the same team proposed the fast SR convolutional neural network (FSRCNN) [24], which is a compact hourglass-shape structure. Shi et al. [25] proposed a novel sub-pixel convolutional layer to replace the deconvolutional layer. By doing so, the training complexity is significantly reduced. The above approaches are based on linear networks, and the structure is relatively simple. However, as the depth of networks increased, over-parameterization appeared. To address these difficulties, recursive networks [26,27] behaved well by using weights repeatedly. On the one hand, the network is deeper; thus, the performance is better. On the other hand, deeper networks are also more likely to cause an exploding gradient. To deal with this contradiction, Kim et al. [28] proposed learning residuals only, since the low-frequency information carried by the LR image is similar to the HR images. A very deep residual channel attention network (RCAN) [29] is proposed for high-precision image SR. As a result of the sparsity of residual images, the convergence speed is accelerated. Afterwards, based on residual learning, many frameworks were proposed [30,31].

With the development of deep neural network, excellent networks are constantly being introduced into this field. [32,33] are based on the densely connected convolutional network (DenseNet) [34]. They make full use of low-level features by introducing dense skip connections. GANs are also adapted for SISR in SRGAN [13]. These kinds of methods propose a perceptual loss function in order to recover photo-realistic textures from LR images. Perceptually satisfying in the sense is their main target.

Recently, more categories of SR appeared, such as blind SR [20–22] and unsupervised SR [35]. Moreover, it has been found that the development of SISR tends to be practical. Google announced the Super Res Zoom technology [36], which focused on solving the problem that the images taken by handheld devices are not clear enough. Dong et al. proposed [37,38], which combine the SR with mersisters. Qian et al. proposed the Trinity Enhancement Network (TENet) [15], which can solve multiple problems at the same time. Deng proposed an algorithm, named SR by Neural Texture Transfer (SRNTT) [39], which implemented SR in a referential way. This year, a large number of SR methods for specific objects have emerged, such as hyperspectral SISR [40], face SR [41], and so on.

2.2. Video Super-Resolution (VSR)

In addition to information in a single frame, VSR has inter-frame temporal correlation. Therefore, both accuracy and consistency need to be considered at the same time. For this purpose, VSR usually has two unavoidable steps: motion compensation and SR restoration.

At the very beginning, VSR was divided into a large number of independent multi-frame SR subtasks [10,11]. They focused on obtaining high-quality reconstruction results for each single frame, while the individually generated high-resolution frames lack coherency temporally, resulting in unpleasant flickering artifacts. The above methods did not make full use of time domain information.

Afterwards, adding optical flow networks to the VSR for motion estimation became popular. Taking efficient sub-pixel convolutional neural network (ESPCN) [25] as a reference, Caballero et al. [42] proposed video ESPCN (VESPCN), which consisted of spatio-

temporal sub-pixel convolution networks and optical flow networks. Specifically, VESPCN learned the motion compensation by the former and improved the accuracy in real time by the latter. Sajjadi et al. [43] proposed frame-recurrent video super-resolution (FRVSR), which repeatedly using previously estimated SR frames to recover subsequent frames. In addition to reusing the reconstructed HR frames, frame and feature-context video super-resolution (FFCVSR) [44] was proposed to exploit the features of the previous frame repeatedly. Likewise, Wang et al. [45] proposed learning for video super-resolution through HR optical flow estimation (SOF-VSR), which innovatively reconstructed high-resolution optical flow instead of estimating the optical flow among low-resolution frames to improve the accuracy of motion compensation. Chu et al. proposed Temporally Coherent GAN (TecoGAN) [46], of which the architecture is based on GAN. It not only used optical flow networks, but also suggested novel loss functions to improve time consistency. Furthermore, due to its feature space losses, the proposed approach improved perceptual quality in VSR.

The addition of the optical flow network does improve the experimental results, but it also increases the computational and memory cost as well. Moreover, the final performance heavily depends on the accuracy of the optical flow prediction. Inaccurate optical flow will cause artifacts, which will also propagate to the reconstructed HR video frame. Therefore, several studies have been done to remove explicit motion compensation. Unlike the previous works, video super-resolution via residual learning (EVSR) [47] estimated motion compensation between frames automatically without explicit motion compensation modules. Ganet [48] integrated motion estimation and the frame recovery into one step by utilizing the self-attention network to merge local features into global features. Younghyun et al. [49] introduce a novel framework dynamic upsampling filters (DUF). Instead of explicitly estimating the motion compensation between LR frames, DUF implicitly utilized the motion information to generate suitable up-sampled filters. In [50], a new method to ensure temporal consistency is proposed. Instead of using optical flow, it uses deformable convolution to track the traceable points by a pyramid, cascading and deformable (PCD) module. Tian et al. [51] proposed a time deformable alignment network (TDAN), which aligned adaptively at the feature level.

3. Methods

In this paper, we aimed at learning non-linear mapping between the input LR frames and the final HR frames. Our main framework is based on GAN, and the main work is to improve the generator. As illustrated in Figure 1, the generator mainly consists of two parts: one for intermediate SR results [46] and the other for edge enhancement [14], which makes the final results clearer. LR videos are usually blurry and accompanied by noise. In addition, GAN will inevitably introduce noise. Therefore, edge enhancement while suppressing noise will greatly improve people's perception. Instead of discriminating the realism of spatial detail only, the generator discriminating temporal changes as well. Moreover, in order to obtain good objective indicators while ensuring people's perception, we added a trained Visual Geometry Group (VGG) to compare the difference between the final results and the GT on several specific feature layers.

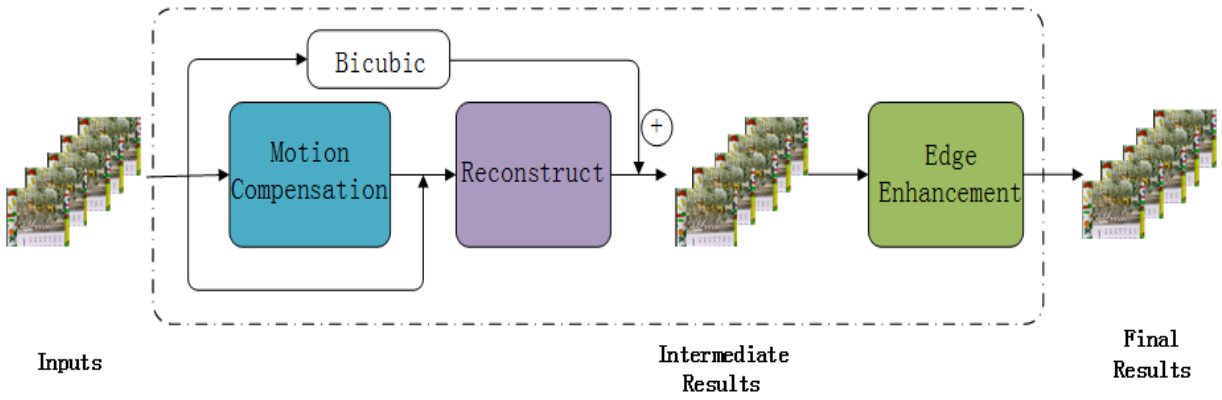


Figure 1. Outline of the generator part in our proposed network.

3.1. TecoGAN

The network is based on the GAN. The generator G is divided into two parts. The first part is the optical flow network F, which obtains the motion compensation v_t from two adjacent low-resolution input frames x_{t-1}^{LR} and x_t^{LR}. Then, v_t is linearly up-sampled four times to obtain V_t. Afterwards, the previous SR frame X_{t-1}^{SR} warps with the inferred motion V_t to obtain $W(V_t, X_{t-1}^{SR})$. The second part is the SR reconstruction network. x_{t-1}^{LR} and $W(V_t, X_{t-1}^{SR})$ are put into this part together for SR reconstruction. The result obtained at this stage is $X_t^{SR_0}$. The network only learns the residual part to stabilize the network training; therefore, we add $X_t^{SR_0}$ to the linear up-sampled result X_t^{LR} from x_t^{LR} to obtain the final result X_t^{SR}.

The following formula can be used to summarize the above steps:

$$V_t = UpSample(F(x_{t-1}^{LR}, x_t^{LR})), \tag{3}$$

$$X_t^{SR} = G(x_t^{LR}, W(V_t, X_{t-1}^{SR})) + UpSample(x_t^{LR}), \tag{4}$$

The design of the adversarial network and loss function is the main innovation. The adversarial network, which called a spatio-temporal discriminator, not only discriminates spatial details but also includes information in the temporal. It receives two sets of inputs, which consists of the generated results and the GT. In each set of inputs, in addition to spatial details, it also includes temporal information. In this way, the discriminator can automatically balance space and time information to avoid inconsistent clarity or excessive smooth result. TecoGAN has a novel loss function named ping-pong loss as well. The input of the optical flow network is two low-resolution groups. The first group has n continuous frames, and the second group is the reverse sequence of the first group. Therefore, it is possible to get the motion compensation v_t between x_{t-1}^{LR} and x_t^{LR} as well as the motion compensation $v_t^{'}$ between x_t^{LR} and x_{t-1}^{LR}, which are used to generate the forward result X_t^{SR} and the reversed $X_t^{SR'}$. Theoretically, the two are the same. Therefore, the ping-pong loss is as follows:

$$L_{pp} = \sum_{i=1}^{n-1} \left\| X_t^{SR} - X_t^{SR'} \right\|_2, \tag{5}$$

3.2. EEGAN

The network is based on the GAN for SISR. The main innovation of this method is in its generator, which divides the results into intermediate result I_{base} and final result I_{edge}^{*}

. Intermediate result I_{base} is generated by a topologically shaped network. This dense block D in the topological structure is regarded as the basic module of feature extraction and fusion. Unlike traditional dense blocks, they can share and fuse feature maps extracted from multiple previous convolutional layers in both horizontal and vertical directions. Therefore, the number of link nodes is approximately twice that of the original dense block, thereby achieving a variety of fine feature expressions.

The final result is the edge enhancement of the intermediate result. Taking into account that edge enhancement will also amplify noise, the mask branch is performed to learn the image mask to detect and remove isolated noise, which are false edges generated in edge extraction. Subsequently, the enhanced edge map is projected onto the HR space through a sub-pixel convolution operation. According to [14], the mathematical expression of the edge enhancement can be written as follows:

$$I_{edge}^{*} = PS(F(D(I_{edge})) \otimes M(D(I_{\text{edge}}))) . \tag{6}$$

Among them:

1. I_{edge} means the extracted edge from intermediate super-resolution result I_{base} by the Laplacian operator.
2. $D(\cdot)$ is the down-sampled operation by the strided convolution, which transforms I_{edge} into LR space.
3. $F(\cdot)$ denotes the dense block above using feature extraction and fusion.
4. $M(\cdot)$ represents the mask branch, which is used for removing false edges caused by noise.
5. $PS(\cdot)$ denotes sub-pixel convolutional, which up-samples the edge maps into HR space.

3.3. Our Method

Referring to the generator of TecoGAN, we constructed the intermediate SR result $X_{t,base}^{SR}$, which is the final result of the generator in TecoGAN. As we all know, the picture quality of videos with lower resolution is always blurry. In view of the characteristic above, we perform edge enhancement after $X_{t,base}^{SR}$, which will significantly improve the edge of the subtitles and the outline of the things, thereby improving the overall picture quality. In the subsequent edge enhancement part, we refer to edge-enhanced GAN (EEGAN) [14]. First, the edge of $X_{t,base}^{SR}$ is extracted with Laplacian operator. The Laplacian operation of the image $X_{t,base}^{SR}$ can be defined as its second derivative. In this article, we used ([-1,-1,-1],[-1,8,-1],[-1,-1,-1]) as the discrete convolution mask to extract the image edge $X_{t,egde}^{SR}$, and its formula is as follows:

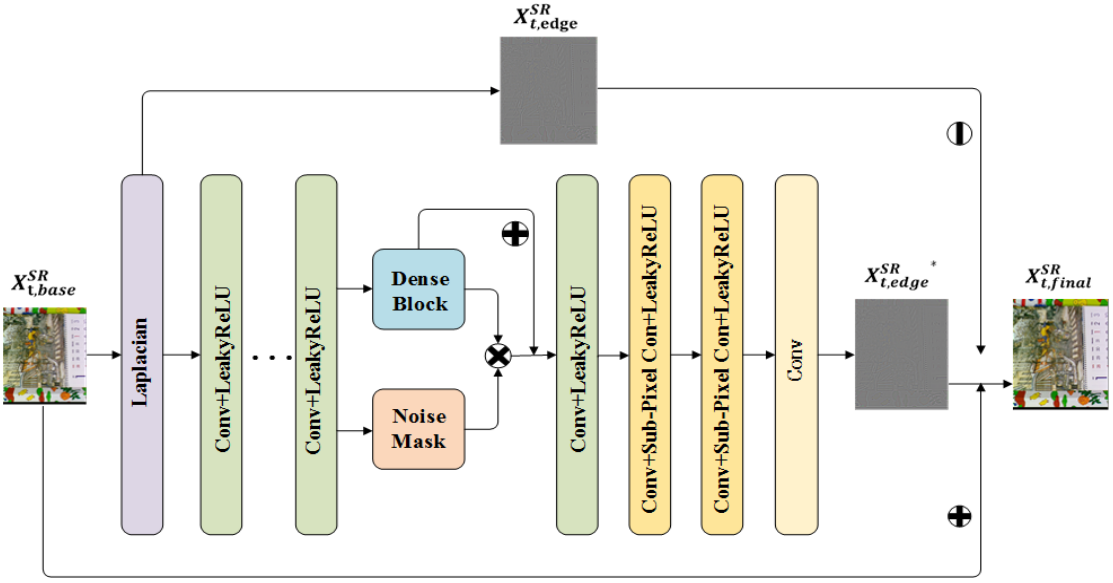
$$X_{t,edge}^{SR} = L \otimes X_{t,base}^{SR}, \tag{7}$$

where $\otimes$ is the convolution operation, and $X_{t,egde}^{SR}$ represents the extracted edge from $X_{t,base}^{SR}$.

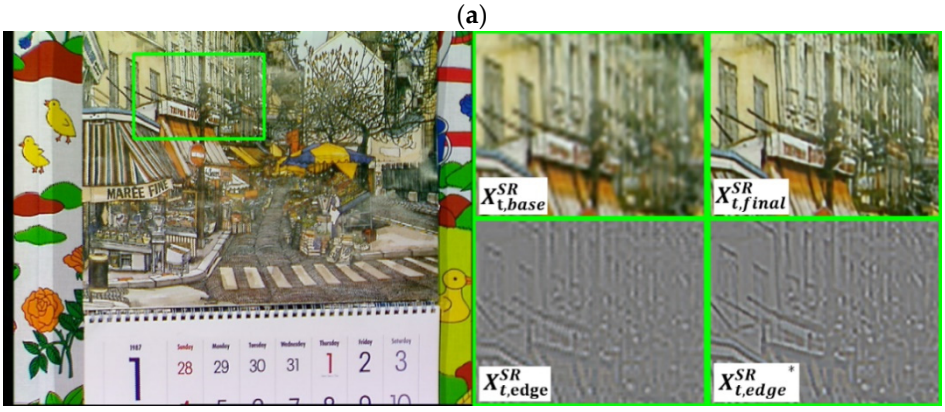
However, videos with lower resolution, such as CCTV, are accompanied by inevitable noise due to the limitations of shooting and production technology. Therefore, the edge obtained at this stage contains a part of false edges caused by noise. GAN will inevitably introduce noise. In order to extract more pure and effective edges, we learn from EEGAN to refine and strengthen $X_{t,edge}^{SR}$. The specific structure is shown in Figure 2. $X_{t,edge}^{SR}$ is firstly converted to low-resolution space in order to reduce the computational cost. After a few convolutional layers, the dense block in EEGAN [14] is used for feature extraction to obtain edges that are more refined. Meanwhile, we learn the noise mask

through a mask branch to achieve the purpose of eliminating noise and artifacts and obtain refined and enhanced edge $X_{t,edge}^{SR\ *}$. We choose leaky rectified linear unit (LeakyReLU) for the activation function of this part. As a variant of rectified linear unit (ReLU), the response of LeakyReLU to the input less than zero is linearly varying, which reduces the sparsity of ReLU. The final result of our SR is $X_{t,final}^{SR}$. It can be expressed as:

$$X_{t,final}^{SR} = X_{t,base}^{SR} + X_{t,\mathrm{edge}}^{SR\ *} - X_{t,edge}^{SR}. \tag{8}$$



(**a**)



(**b**)

Figure 2. This is our edge enhancement module and partial results. (**a**) This is our edge enhancement module. We take an image as an example to show its process. The Dense Block and Noise Mask inside refer to the design in edge-enhanced GAN (EEGAN) [14]; (**b**) On the left is ground truth, and on the right is a partial enlarged view of $X_{t,base}^{SR}$, $X_{t,edge}^{SR}$, $X_{t,edge}^{SR\ *}$, and $X_{t,final}^{SR}$ of the city clip for $4\times$ video super-resolution, where $X_{t,base}^{SR}$ is the intermediate result, $X_{t,edge}^{SR}$ is the edge extraction of $X_{t,base}^{SR}$, $X_{t,edge}^{SR\ *}$ is the enhancement and noise purification of $X_{t,edge}^{SR}$, and $X_{t,final}^{SR}$ is the final result.

In order to ensure the continuity of the reconstructed video in the temporal, we add ping-pong loss from TecoGAN [46] in our framework. We input two groups of consecutive video frames, each of n frames. The second group is the reverse sequence of the first group. In this way, we obtain the forward result $X_{t,final}^{SR}$ and the reversed result $X_{t,final}^{SR\ '}$, and the ping-pong loss is:

$$L_{pp} = \sum_{i=1}^{n-1} \left\| X_{t,final}^{SR} - X_{t,final}^{SR\ '} \right\|_2. \tag{9}$$

During network training, the generator returns two results $X_{t,base}^{SR}$ and $X_{t,final}^{SR}$, which are the intermediate result and the final result. To make the generator robust, we assign different loss weights to these two results when designing the content loss function $L_{content}$.

$$L_{content} = \sum_{i-1}^{n-1}(\left\| X_{base}^{SR} - X_t^{HR} \right\|_2 + \alpha \left\| X_{t,final}^{SR} - X_t^{HR} \right\|_2), \quad (10)$$

where X_t^{HR} is the GT, and α is the weight. Specifically, α changes according to a certain rule during the training. As the training step increased, the model becomes more and more accurate. Simultaneously, the difference between the intermediate result and the final result is getting bigger and bigger. Based on this, the α is set to 10 at the beginning and is increased with the training step. See the experimental part for specific parameters. In addition to the above loss functions, we retain the other loss functions in TecoGAN.

In addition, we train the model in two steps. In a word, we firstly train the simplified network and then train the complete network on the basis of the simplified network. In the intermediate model, we only train the generator. In addition, the loss function is simplified. In this step, only the content loss is retained, and the weight remains unchanged. This step is equivalent to an initialization parameter training of the subsequent mode. Since the framework and loss functions here are more complicated, if we train the complete network directly, it is difficult to find accurate network parameters or it takes a long time. A simplified network that is pre-trained helps find the approximate range of the final parameters of the network. Then, we initialize the complete network with the parameters of the simplified network. Next, we fine-tune the framework. In this step, α increases with the training steps as well as the learning rate decays with the training steps.

4. Experiments

In this chapter, we first give training details. Secondly, we perform a comparative experiment study. Then, the evaluation metrics will be illustrated. Finally, we will provide qualitative analysis and quantitative evaluation of the experimental results.

4.1. Train Details

We perform the experiment using Python3.6 and Tensorflow-gpu1.10.0 on PyCharm2019.1.3 (Community Edition). The computer used for the experiment is of 3.6 GHz CPU and NVIDIA GeForce GTX 1080Ti GPU. See Table 1 for more details.

Table 1. Components and information of the system used for implementation.

Components	Information
Operating System	Ubuntu 16.04 Long Term Support
Memory	32 G
Graphic Processing Unit (GPU)	NVIDIA GeForce GTX 1080Ti
Central Processing Unit (CPU)	Inter® Xeon(R) W-2123 CPU @ 3.60 GHz
Integrated Development Environment (IDE)	PyCharm2019.1.3 (Community Edition)
Language	Python 3

The dataset used for training was downloaded from Vimeo. We got the video download link from TecoGAN [46]. Vimeo Terms of Service are followed, and all used videos are available on Vimeo with the download option. Specifically, we download 25 high-resolution videos. In order to learn fine motion compensation, we selected 276 scenes, each of which contains 120 frames without lens switching. The resolution size of each scene is not uniformly specified, but the length or height must be larger than 400. Imitating the characteristics of videos with lower resolution, fuzziness, and noise, we use Gaussian blur kernel for four times down-sampled. See Table 2 for more details.

Table 2. Specific parameters of the training dataset.

Items	Parameters
Video Source	Vimeo
Number of Scenes	276
Number of Frames per Scene	120 frames

Training the model is divided into two steps. When training the intermediate model, the batch size is 4, the input LR patch size is 32×32, the learning rate is fixed at 5×10^{-5}, and the α is fixed at 10. When training the final model, the batch size is 1, the size of the input LR patch is 32×32, the initial learning rate is 5×10^{-5}, and the initial α is fixed at 10. Moreover, we use the decay function provided in the Tensorflow to dynamically decay the learning rate and α. The formula is as follows:

$$decayed = initial \times decay\_rate\hat{}(\frac{global\_step}{decay\_step}), \tag{11}$$

For learning rate, the decay_rate is 0.9 and the decay_step is 28 K. For α, the decay_rate is 1.1 and the decay_step is 50 K. The intermediate model performs 600 K iterations, while the intermediate model performs 1200 K. We use Adam with a momentum of 0.9 and a weight decay of the same as the learning rate for optimization. We also recorded the performance of the model on peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as the interaction changes when training the final model. See Table 3, Figure 3 for more details.

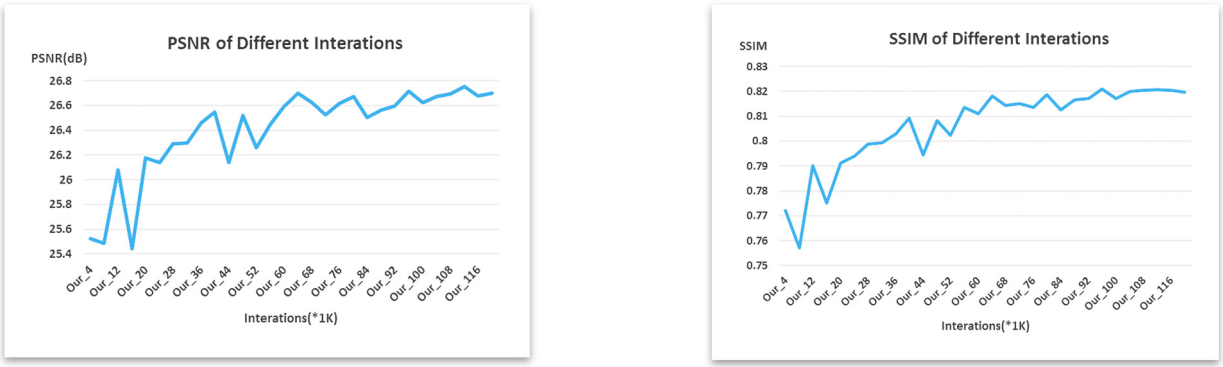
Table 4 shows the details of the Vid4. During the test, we removed the first and last two frames. Specifically, the final data are the average of 155 frames of images, including 37 frames of calendar, 30 frames of city, 45 frames of foliage, and 43 frames of walk.

Table 3. Training parameters.

Items of step1	Parameters	Items of step2	Parameters
Batch Size	4	Batch Size	1
Patch Size	32×32	Patch Size	32×32
Learning Rate	5×10^{-5}	Learning Rate	5×10^{-5}
Decay Rate		Decay Rate	0.9
Decay Step		Decay step	28 K
α	10	α	10
Decay Rate		Decay Rate	1.1
Decay Step		Decay Step	50 K
Iteration	600K	Iteration	1200 K

Table 4. The details of the Vid4.

Scenes	Low-Resolution	High-Resolution	Frames
calendar	180×144	720×576	41
city	176×144	704×576	34
foliage	180×120	720×480	49
walk	180×120	720×480	47



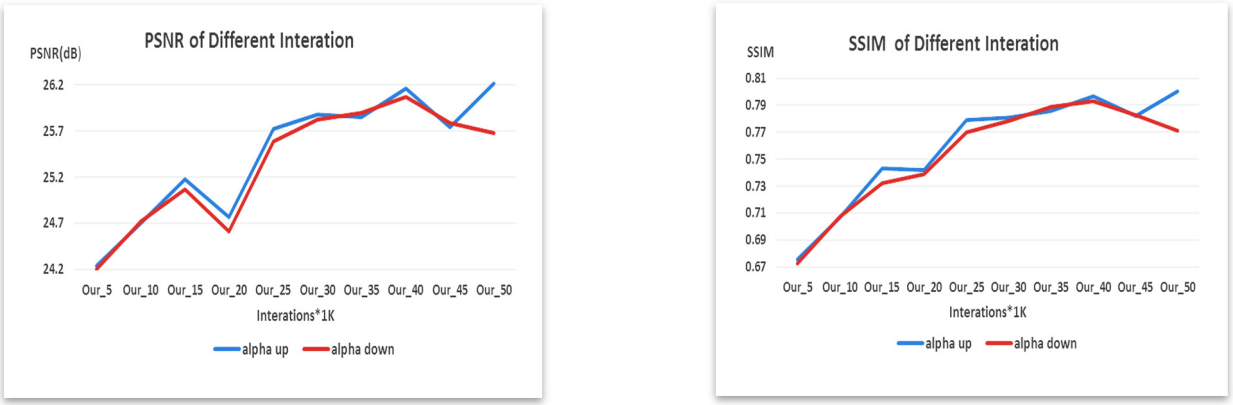
(**a**) (**b**)

Figure 3. (**a**) Peak Signal to Noise Ratio (PSNR) changes with iterations; (**b**) Structural SIMilarity SSIM changes with iterations.

4.2. Comparative Study

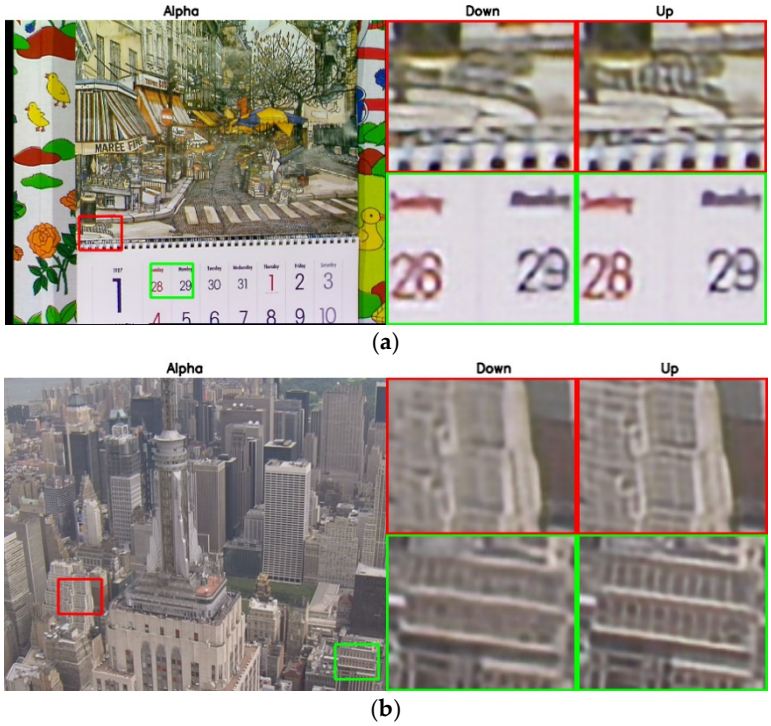
We tried different decay methods, different loss functions, and different datasets to compare the final results and different edge enhancement modules.

For different α decay methods, we compared two patents. Both of them start from 10; one is exponentially decreasing at a rate of 0.9, while the other is exponentially increasing at a rate of 1.1. Other factors remain the same. The experimental results in Figures 4 and 5 show that the incremental approach is better. The testing samples are the same as above, including 155 frames. As the number of iterations increased, the model becomes more and more accurate, and the gap between $X_{t,base}^{SR}$ and $X_{t,final}^{SR}$ becomes larger and larger. Therefore, the larger and larger α conforms to this trend.



(**a**) (**b**)

Figure 4. The experimental results of different decay mode: (**a**) Peak Signal to Noise Ratio (PSNR) changes with iterations; (**b**) Structural SIMilarity (SSIM) changes with iterations.



(**a**)

(**b**)

Figure 5. Results of different α decay methods, where 'Down' means exponentially decreasing at a rate of 0.9, 'Up' means exponentially increasing at a rate of 1.1: (**a**) Qualitative comparison on the calendar clip for $4\times$ video super-resolution; (**b**) Qualitative comparison on the city clip for $4\times$ video super-resolution.

For the loss function, we tried to calculate the loss in proportion to the two outputs $X_{t,base}^{SR}$ and $X_{t,final}^{SR}$ of the generator for all loss functions or to calculate the loss in proportion to the content loss only. The former is loss function A, the latter is loss function B. The latter performs better. In Figure 6, we can find that loss function A will cause a more obvious mosaic phenomenon. Using the two layers of the generator on the content loss and assigning different loss weights helps lock in the final result in a more accurate range at the beginning and keep a relatively reasonable range later. Other loss functions only need to use the final result $X_{t,final}^{SR}$.

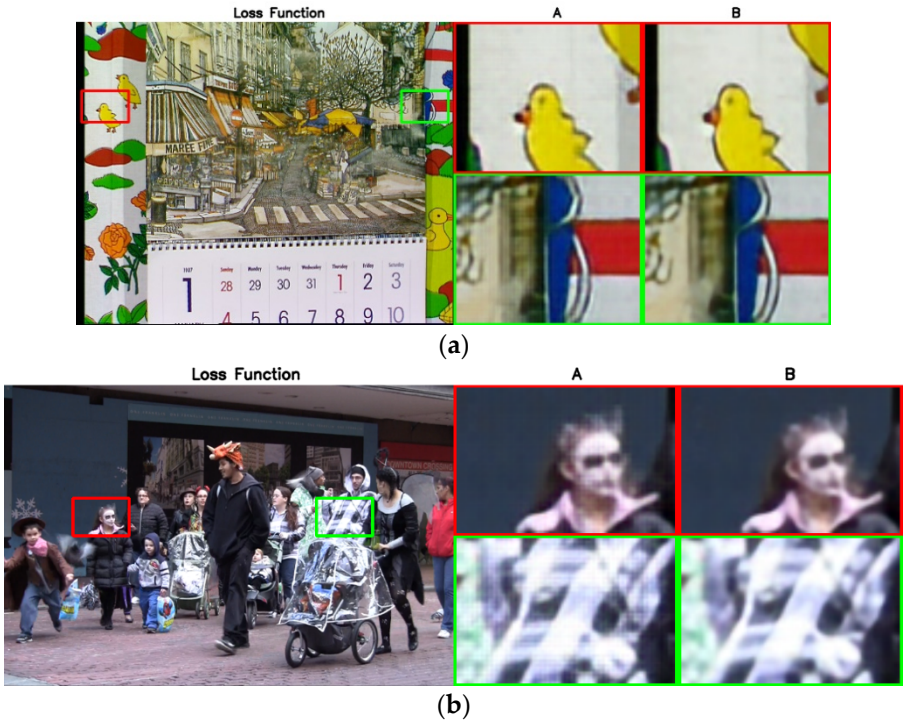


Figure 6. Results of different loss functions: (**a**) Qualitative comparison on the calendar clip for 4× video super-resolution; (**b**) Qualitative comparison on the walk clip for 4× video super-resolution.

For the datasets, we tried down-sampling from high-resolution videos downloaded randomly on vimeo or down-sampling from high-resolution repaired versions of film and television dramas around 2000. Models train on different datasets perform differently in different scenes. The former performed better on Vid4, while the latter performed better on the film and television scene, which you can see in Figure 7. The experiment shows that the models trained on different training datasets adapt to different scenarios.

Figure 7. Qualitative comparison on the Secret History of Xiaozhuang clip for 4× video super-resolution results of models trained on different training datasets.

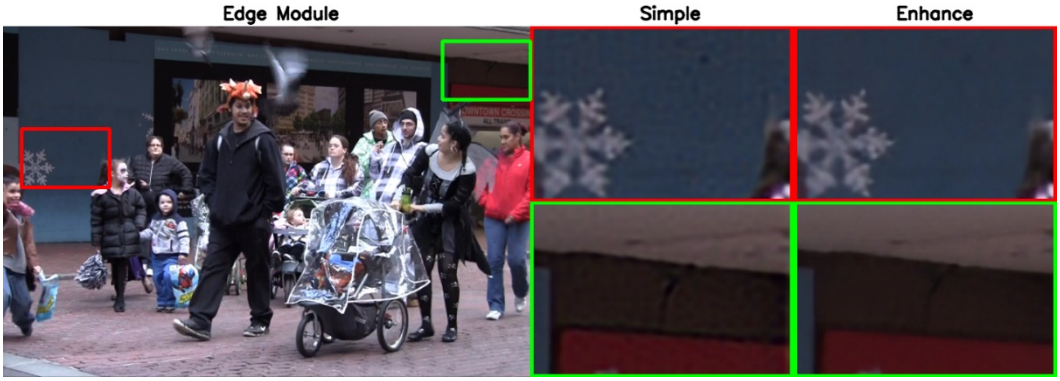
For edge enhancement modules, we tried simple edge enhancement and complex edge enhancement. The final result of the former is only the sum of the intermediate result

$X_{t,base}^{SR}$ and the Laplacian edge enhancement $X_{t,edge}^{SR}$ of the intermediate result. It can be expressed as:

$$(X_{t,final}^{SR})_{simple} = X_{t,base}^{SR} + X_{t,edge}^{SR}, \quad (12)$$

The latter is as described in Section 3.3, where experiments proved our theory. As you can see in Figure 8, when performing edge enhancement, if both denoising and strengthening are considered, the result is better. The simple edge enhancement will lead to some noise and blurred edges.

(**a**)



(**b**)

Figure 8. Results of different edge enhancement modules: (**a**) Qualitative comparison on the calendar clip for $4\times$ video super-resolution; (**b**) Qualitative comparison on the walk clip for $4\times$ video super-resolution.

4.3. Evaluation

According to the mainstream of the SR field, we calculate Peak Signal to Noise Ratio (PSNR) and Structural SIMilarity (SSIM) on the Y channel of YCbCr space, where Y refers to the luminance component, Cb refers to the blue chrominance component and Cr refers to the red chrominance component

Peak signal-to-noise ratio (PSNR) is an objective standard for evaluating images. The mathematical formula is as follows:

$$PSNR = 10 \times \log_{10}(\frac{MAX^2}{MSE}) = 20 \times \log_{10}(\frac{MAX}{\sqrt{MSE}}), \quad (13)$$

where MSE is the mean square error between the original image and the SR frame, and MAX indicates the maximum value of the image color. For example, the 8-bit sampling point is expressed as 255.

Structural similarity (SSIM) is an index to measure the similarity of two images. The mathematical formula is as follows:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x{}^2 + \mu_y{}^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (14)$$

where x is the SR frame, y is the GT, μ_x and μ_y are the mean values; σ_x and σ_y are the standard deviations, and σ_{xy} is the covariance of x and y. We use the built-in compare_ssim function of the skimage module to calculate. SSIM is a number between 0 and 1. The larger it is, the smaller the gap between the result frame and the GT; that is, the image quality is better. When the two images are exactly the same, SSIM is 1.

We compare the proposed method on the Vid4 dataset with some other SR algorithms: video super-resolution with convolutional neural network (VSRNet) [52], VESPCN [42], SOF-VSR [45], FRVSR [43], and TecoGAN [46]. Table 4 shows the details of the Vid4. During the test, we removed the first and last two frames. Table 5 shows that our network has the best average results on PSNR and SSIM on the Vid4 dataset. Figures 9 and 10 also show the superiority of our method in qualitative results. Compared with TecoGAN [46], the results of our method are closer to GT. The results of TecoGAN contain more noise. Meanwhile, distortion is more obvious in some details.

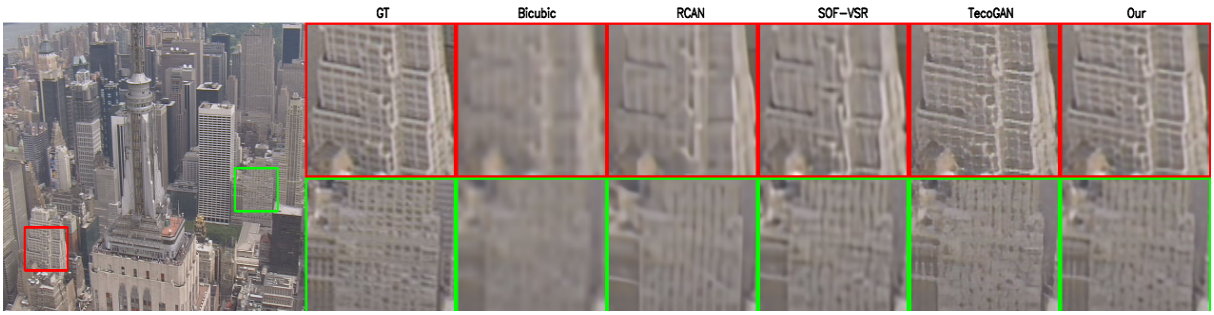


Figure 9. Qualitative comparison on the city clip for $4\times$ video super-resolution results.

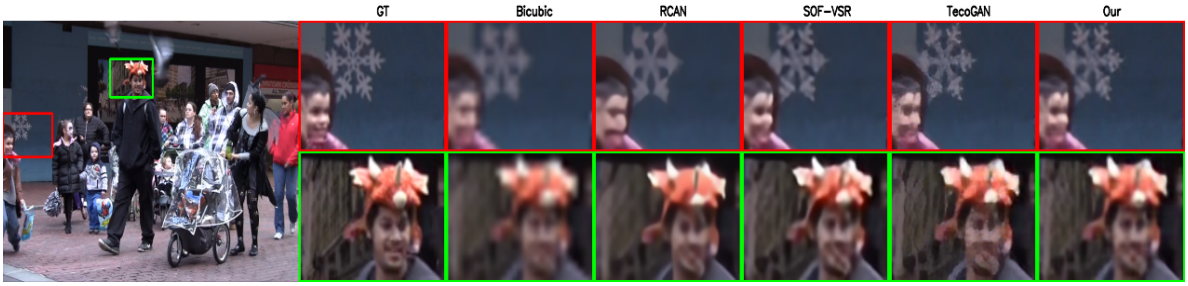


Figure 10. Qualitative comparison on the walk clip for $4\times$ super-resolution results.

Table 5. Peak Signal to Noise Ratio (PSNR) and Structural SIMilarity (SSIM) of different methods on the Vid4 dataset.

Scale	Evaluation		Bicubic	VSRNet	VESPCN	SOF-VSR	FRVSR	TecoGAN	Our
4	PSNR	calendar	20.34			22.89		23.22	23.79
		city	24.88			26.88		26.79	27.63
		foliage	23.36			25.58		24.30	26.03
		walk	25.52			28.93		28.12	29.45
		average	23.53	24.84	25.35	26.12	26.69	25.58	26.75
	SSIM	calendar	0.55			0.78		0.79	0.81

	city	0.50			0.76		0.77	0.80
	foliage	0.56			0.76		0.71	0.78
	walk	0.79			0.89		0.88	0.90
	average	0.61	0.70	0.76	0.80	0.82	0.79	0.82

VESPCN means video efficient sub-pixel convolutional neural network [42], VSRNet means video super-resolution with convolutional neural network [52], SOF-VSR means learning for video super-resolution through HR optical flow estimation [45], FRVSR means frame-recurrent video super-resolution [43] and TecoGAN means temporally coherent generative adversarial network [46]. The data of VSRNet, VESPCN, and FRVSR are quoted from their paper directly.

We also tested our method in other low-resolution scenes in our lives. Table 6 shows the details of the data. During the test, we removed the first and last two frames. Table 7 shows that our network has the best average results on PSNR and SSIM on film and television scenes. Figure 11 also shows the superiority of our method in qualitative results.

Table 6. The details of film and television scenes.

Scenes	Low-Resolution	High-Resolution	Frames
Create State	320×180	1280×720	149
Secret History of Xiaozhuang	320×242	1280×968	250

Table 7. Peak Signal to Noise Ratio (PSNR) and Structural SIMilarity (SSIM) of different methods on film and television scenes.

Scale		Evaluation	Bicubic	TecoGAN	Our
4	PSNR	Create State	28.00	31.74	32.72
		Secret History of Xiaozhuang	34.31	37.13	39.05
		average	31.96	35.12	36.69
	SSIM	Create State	0.94	0.97	0.98
		Secret History of Xiaozhuang	0.95	0.98	0.98
		average	0.95	0.98	0.98

TecoGAN means temporally coherent generative adversarial network [46].

Figure 11. Qualitative comparison on Create State clip for 4× video super-resolution results.

5. Conclusion

In this article, we proposed an end-to-end SR method for LR video, which can be used to improve the image quality of urban CCTV. A large number of experiments have shown that our method can improve the resolution of the video and meet people's perception. These LR videos are usually blurry and inevitably accompanied by noise. The edge enhancement module we added can successfully enhance the edge but does not amplify the noise. At the same time, we have also done many comparative experiments. These experiments show that models trained on different training datasets perform significantly differently in different scenarios. We have proved that this method is superior to other methods on different test datasets.

In the future, we will consider optimization based on the training dataset. In this article, we down-sample HR frames to obtain the dataset. The down-sampling process

simulates the degradation process of LR data as much as possible, but the same effect cannot be guaranteed. Therefore, the trained model is only most suitable for LR scenarios that meet specific degradation conditions. Based on this, we will try to eliminate the process of manually down-sampling HR frames to obtain LR frames. Specifically, we will directly use continuous frames of the original video as inputs and the corresponding continuous frames of the HR repair version as targets.

Author Contributions: Investigation, (G.T.) Methodology, J.W.; Project administration, P.A.; Software, J.W.; Supervision, P.A.; Writing—original draft, J.W.; Writing—review and editing, J.W. and G.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China Project (grant number 62020106011) and the Shanghai Science and Technology Commission Project (grant number 20DZ2290100).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Freeman, W.T.; Jones, T.R.; Pasztor, E.C. Example-based super-resolution. *IEEE Eng. Med. Boil. Mag.* **2002**, *22*, 56–65, doi:10.1109/38.988747.
2. Tai Y.W.; Liu S.; Brown M.S.; Lin, S. Super resolution using edge prior and single image detail synthesis. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 13-18 June 2010.
3. Yang, J.; Wright, J.; Huang, T.S.; Ma, Y. Image Super-Resolution Via Sparse Representation. *IEEE Trans. Image Process.* **2010**, *19*, 2861–2873, doi:10.1109/tip.2010.2050625.
4. Chang, H.; Yeung, D.-Y.; Xiong, Y. Super-resolution through neighbor embedding. In Proceedings of the Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.; IEEE, 2004; Vol. 1.
5. Lidke, K.A.; Rieger, B.; Jovin, T.M.; Heintzmann, R. Superresolution by localization of quantum dots using blinking statistics. *Opt. Express* **2005**, *13*, 7052–7062, doi:10.1364/opex.13.007052.
6. Wahab, A.W.A.; Bagiwa, M.A.; Idris, M.Y.I.; Khan, S.; Razak, Z.; Ariffin, M.R.K. Passive video forgery detection techniques: A survey. In Proceedings of the 2014 10th International Conference on Information Assurance and Security; IEEE, 2014; pp. 29–34.
7. Bagiwa, M.A.; Wahab, A.W.A.; Idris, M.Y.I.; Khan, S.; Choo, K.-K.R. Chroma key background detection for digital video using statistical correlation of blurring artifact. *Digit. Investig.* **2016**, *19*, 29–43, doi:10.1016/j.diin.2016.09.001.
8. Bagiwa, M.A.; Wahab, A.W.A.; Idris, M.Y.I.; Khan, S. Digital Video Inpainting Detection Using Correlation Of Hessian Matrix. *Malays. J. Comput. Sci.* **2016**, *29*, 179–195, doi:10.22452/mjcs.vol29no3.2.
9. Wang, L.; Li, D.; Zhu, Y.; Tian, L.; Shan, Y. Dual Super-Resolution Learning for Semantic Segmentation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2020; pp. 3773–3782.
10. Liu, D.; Wang, Z.; Fan, Y.; Liu, X.; Wang, Z.; Chang, S.; Huang, T. Robust Video Super-Resolution with Learned Temporal Dynamics. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); IEEE, 2017; pp. 2526–2534.
11. Tao, X.; Gao, H.; Liao, R.; Wang, J.; Jia, J. Detail-Revealing Deep Video Super-Resolution. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); IEEE, 2017; pp. 4482–4490.
12. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.
13. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21--26 July 2017; pp. 5892–5900.
14. Jiang, K.; Wang, Z.; Yi, P.; Wang, G.; Lu, T.; Jiang, J. Edge-Enhanced GAN for Remote Sensing Image Superresolution. *IEEE Trans. Geosci. Remote. Sens.* **2019**, *57*, 5799–5812, doi:10.1109/tgrs.2019.2902431.
15. Qian, G.; Gu, J.; Ren, J.S.; Dong, C.; Zhao, F.; Lin, J. Trinity of Pixel Enhancement: a Joint Solution for Demosaicking, Denoising and Super-Resolution. *arXiv***2019** arXiv:1905.02538.
16. Dong, W.S.; Zhang, L.; Shi, G.M.; Li, L. Nonlocally centralized sparse representation for image restoration. IEEE Transactions on Image Processing. *IEEE Transactions on Image Processing* **2013**, *22*, 1620–1630.
17. Chan, S.H.; Wang, X.; Elgendy, O.A. Plug-and-Play ADMM for Image Restoration: Fixed-Point Convergence and Applications. *IEEE Trans. Comput. Imaging* **2016**, *3*, 84–98, doi:10.1109/tci.2016.2629286.
18. Guo, Y.; Chen, J.; Wang, J.; Chen, Q.; Cao, J.; Deng, Z. Closed-Loop Matters: Dual Regression Networks for Single Image Super-Resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 5406-5415.
19. Maeda, S. Unpaired Image Super-Resolution Using Pseudo-Supervision. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2020; pp. 288–297.

20. Zhang, K.; Zuo, W.; Zhang, L. Deep Plug-And-Play Super-Resolution for Arbitrary Blur Kernels. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2019; pp. 1671–1681.
21. Gu, J.; Lu, H.; Zuo, W.; Dong, C. Blind Super-Resolution With Iterative Kernel Correction. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2019; pp. 1604–1613.
22. Zhang, K.; Zuo, W.; Zhang, L. Learning a Single Convolutional Super-Resolution Network for Multiple Degradations. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; IEEE, 2018; pp. 3262–3271.
23. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a deep convolutional network for image super-resolution. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 184–199.
24. Dong, C.; Loy, C.C.; Tang, X. Accelerating the Super-Resolution Convolutional Neural Network. In Proceedings of the Lecture Notes in Computer Science; Springer Science and Business Media LLC, 2016; pp. 391–407.
25. Shi, W.; Caballero, J.; Huszar, F.; Totz, J.; Aitken, A.P.; Bishop, R.; Rueckert, D.; Wang, Z. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Institute of Electrical and Electronics Engineers (IEEE), 2016; pp. 1874–1883.
26. Kim, J.; Lee, J.K.; Lee, K.M. Deeply-Recursive Convolutional Network for Image Super-Resolution. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2016; pp. 1637–1645.
27. Tai, Y.; Yang, J.; Liu, X. *Image Super-Resolution via Deep Recursive Residual Network*; IEEE Computer Vision and Pattern Recognition: Honolulu, HI, USA, 2017.
28. Kim, J.; Lee, J.K.; Lee, K.M. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
29. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
30. Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M. Enhanced Deep Residual Networks for Single Image Super-Resolution. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); IEEE, 2017; pp. 1132–1140.
31. Yu, J.; Fan, Y.; Yang, J.; Xu, N.; Wang, Z.; Wang, X.; Huang, T. Wide activation for efficient and accurate image super-resolution. *arXiv* **2018**, arXiv:1808.08718.
32. Tong, T.; Li, G.; Liu, X.; Gao, Q. Image Super-Resolution Using Dense Skip Connections. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); IEEE, 2017; pp. 4809–4817.
33. Hu, X.; Mu, H.; Zhang, X.; Wang, Z.; Tan, T.; Sun, J. Meta-SR: A Magnification-Arbitrary Network for Super-Resolution. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2019; pp. 1575–1584.
34. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In *CVPR 2017, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017*; IEEE: New York, NY. USA, 2017; pp. 4700–4708.
35. Shocher, A.; Cohen, N.; Irani, M. Zero-Shot Super-Resolution Using Deep Internal Learning. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; IEEE, 2018; pp. 3118–3126.
36. Wronski, B.; Garcia-Dorado, I.; Ernst, M.; Kelly, D.; Krainin, M.; Liang, C.K.; Milanfar, P. Handheld Multi-Frame Super-Resolution. *ACM Transactions on Graphics (TOG)*, **2019**, *38(4)*, 1–18.
37. Dong, Z.; Lai, C.S.; He, Y.; Qi, D.; Duan, S. Hybrid dual-complementary metal–oxide–semiconductor/memristor synapse-based neural network with its applications in image super-resolution. *IET Circuits, Devices & Systems* **2019**, *13(8)*, 1241–1248.
38. Dong, Z.; Lai, C.S.; Qi, D.; Xu, Z.; Li, C.; Duan, S. A general memristor-based pulse coupled neural network with variable linking coefficient for multi-focus image fusion. *Neurocomputing* **2018**, *308*, 172–183, doi:10.1016/j.neucom.2018.04.066.
39. Zhang, Z.; Wang, Z.; Lin, Z.; Qi, H. Image Super-Resolution by Neural Texture Transfer. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2019; pp. 7974–7983.
40. Zhang, L.; Nie, J.; Wei, W.; Zhang, Y.; Liao, S.; Shao, L. Unsupervised Adaptation Learning for Hyperspectral Imagery Su-per-Resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020, pp. 3070–3079.
41. Ma, C.; Jiang, Z.; Rao, Y.; Lu, J.; Zhou, J. Deep Face Super-Resolution With Iterative Collaboration Between Attentive Recovery and Landmark Estimation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2020; pp. 5568–5577.
42. Caballero, J.; Ledig, C.; Aitken, A.; Acosta, A.; Totz, J.; Wang, Z.; Shi, W. Real-Time Video Super-Resolution With Spatio-Temporal Networks and Motion Compensation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
43. Sajjadi, M.S.M.; Vemulapalli, R.; Brown, M. Frame-Recurrent Video Super-Resolution. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; IEEE, 2018; pp. 6626–6634.
44. Yan, B.; Lin, C.; Tan, W. Frame and Feature-Context Video Super-Resolution. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence; Association for the Advancement of Artificial Intelligence (AAAI), 2019; Vol. 33, pp. 5597–5604.
45. Wang, L.; Guo, Y.; Lin, Z.; Deng, X.; An, W. Learning for Video Super-Resolution Through HR Optical Flow Estimation. In Proceedings of the Constructive Side-Channel Analysis and Secure Design; Springer International Publishing, 2019; pp. 514–529.

46. Chu, M.; Xie, Y.; Mayer, J.; Leal-Taixé, L.; Thuerey, N. Learning temporal coherence via self-supervision for GAN-based video generation. *ACM Trans. Graph.* **2020**, *39*, 75–1, doi:10.1145/3386569.3392457.
47. Wang, W.; Ren, C.; He, X.; Chen, H.; Qing, L. Video Super-Resolution via Residual Learning. *IEEE Access* **2018**, *6*, 23767–23777, doi:10.1109/access.2018.2829908.
48. Hung, K.-W.; Qiu, C.; Jiang, J. Video Super Resolution via Deep Global-Aware Network. *IEEE Access* **2019**, *7*, 74711–74720, doi:10.1109/access.2019.2920774.
49. Jo, Y.; Oh, S.W.; Kang, J.; Kim, S.J. Deep Video Super-Resolution Network Using Dynamic Upsampling Filters Without Explicit Motion Compensation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; Institute of Electrical and Electronics Engineers (IEEE), 2018; pp. 3224–3232.
50. Wang, X.; Chan, K.C.; Yu, K.; Dong, C.; Loy, C.C. EDVR: Video Restoration With Enhanced Deformable Convolutional Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); Institute of Electrical and Electronics Engineers (IEEE), 16-17 June 2019; pp. 1954–1963.
51. Tian, Y.; Zhang, Y.; Fu, Y.; Xu, C. TDAN: Temporally-Deformable Alignment Network for Video Super-Resolution. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); Institute of Electrical and Electronics Engineers (IEEE),Long Beach, CA, USA, USA , 2020; pp. 3357–3366.
52. Kappeler, A.; Yoo, S.; Dai, Q.; Katsaggelos, A.K. Video Super-Resolution With Convolutional Neural Networks. *IEEE Trans. Comput. Imaging* **2016**, *2*, 109–122, doi:10.1109/tci.2016.2532323.