Mixing Frames: A Video-oriented Approach to Video Super-resolution

Junyi Zhu
MIT CSAIL
32 Vassar Street
junyizhu@mit.edu

Megan C. Chao
MIT CSAIL
32 Vassar Street
megchao@mit.edu

Abstract

Recent advances in image super resolution convolutional neural networks, along with rapidly growing computational power, make the recovery of finer texture details in videos possible. So far, most of the research work has largely focused on minimizing the mean squared reconstruction error and improving high peak signal-to-noise ratios [1, 2]. However, video frame is differing from single image as it usually contains more information in order to form a sequential visual effect. Some state-of-art methods nowadays start to noticing the limitations of single-frame generating models, and instead, predicting the center frame from its surrounding frames [4]. These implementations take more characteristics of video frames into consideration compared to previous approaches, and generally output more smooth HR videos.

In this work, we take one step further than that and propose a purely video-feature oriented approach: instead of building an all-in-one neural network for all types of videos, we instead train several different networks with various performance features, and based on the video clip's characteristics (i.e. color saturation, movement frequency in terms of frame blurriness), pick the most suitable model(s) for each part, even mixing them up if necessary.

We implemented super-resolution ResNet (SRResNet) and super-resolution generative adversarial network (SRGAN) neural networks to enhance the resolution of downsampled videos. We synthesized these models, along with bicubic upsampling, to create three different video super-resolution neural networks, for a total of five neural networks. We used these networks to construct 4x upscaled versions of videos. We found that the SRResNet-based networks performed better on videos with slower movement and saturated colors, where image sharpness is more important, while the SRGAN-based networks performed better on videos with faster movement and desaturated colors, where we prioritize less noise in the video over sharpness.

1. Introduction

Image resolution refers the amount of information and detail contained in an image. High-resolution images are important because they not only make it easier for humans to interpret the contents of the image, but also help machines do the same. For instance, high-resolution images are useful in computer vision applications such as automatic image recognition, where machines can classify specific objects in photos. However, sometimes there are only low-resolution available to use, and as a result, the technique image super-resolution (SR) seeks to improve the enhance the resolution of an existing image by constructing a high-resolution image from several low-resolution frames.

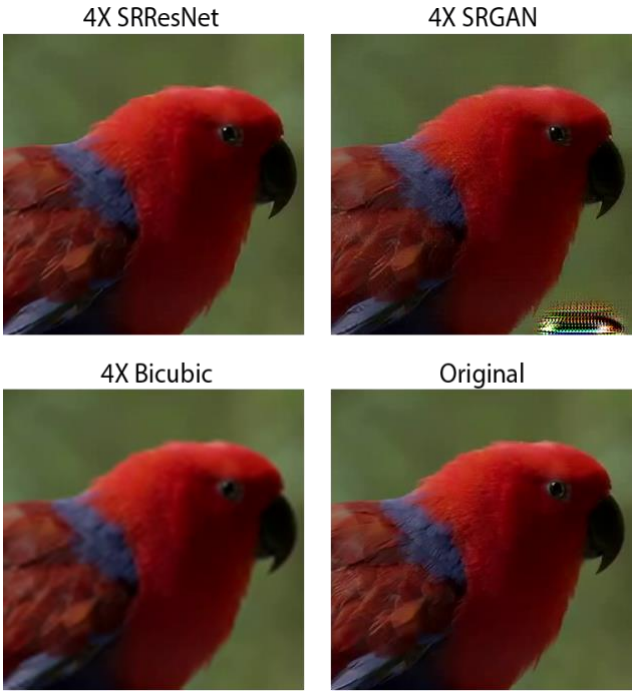

Figure 1: Sample super-resolved frames of various models implemented. [4X upscale]

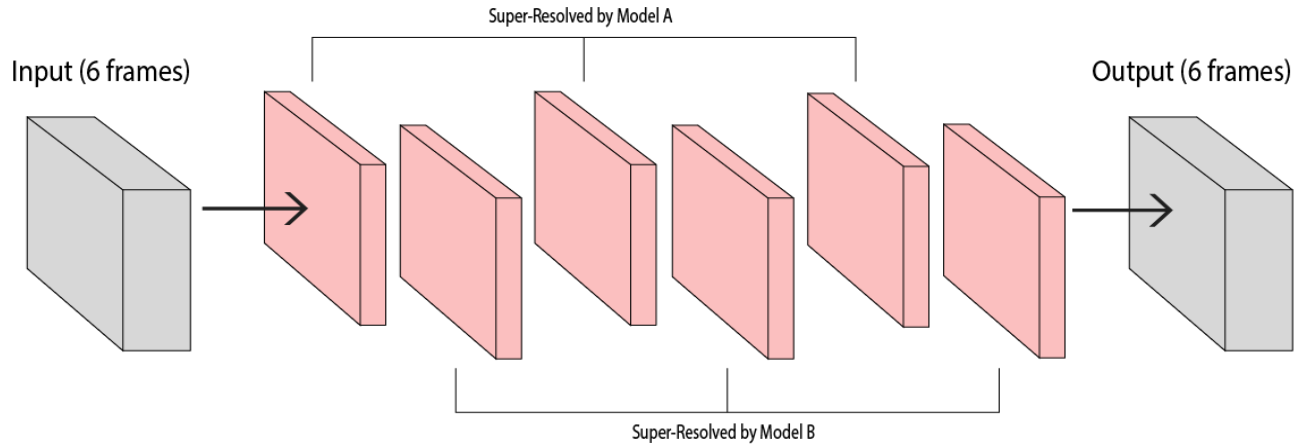

Figure 2: Frame Generating Network Architecture

Image super-resolution has been extensively explored in computer vision literature, but there is not as much research on super-resolution for videos. Video super-resolution may not be as simple as applying image super-resolution on individual frames of a video, as it only optimizes based on information contained within each frame. Because consecutive video frames will be dependent on each other, viewers may perceive noise between the frames of the video as it plays. The goal of this project is to create an original system that improves upon this technique.

In this paper, we implement two neural networks, SRResNet and SRGAN. We also combine these two networks, and combine each of them with bicubic upsampling to create three new networks: SRResNet-SRGAN, SRResNet-Bicubic, and SRGAN-Bicubic. We tested how these networks performed on improving the resolution of videos, varying the speed of the movement in the video (slow vs. fast) and the saturation of the colors in the video (saturated vs. desaturated) and found that SRResNet-based networks performed best on videos with slower movement and saturated colors, while SRGAN-based networks performed best on videos with faster movement and desaturated colors. Some sample generated frames with various neural networks are shown in Figure 1.

2. Related Work

Much of the previous contributions to the field of image super-resolution involves convolutional neural networks. Some of the literature describes new neural network architecture that achieves a better result, such as Shi et al. [1], who describe a sub-pixel convolutional neural network and then apply it to enhance images and video frames. Similarly, Ledig et al. [2] propose a GAN for image super-resolution that outperforms existing image super-resolution neural networks. Lastly, Johnson et al. [3] improves image super-resolution results by changing the loss function in their neural network from a sub-pixel loss function to a perceptual loss function.

3. Approach

First, we implemented SRResNet and SRGAN networks based on the paper by Ledig et al. [2] and applied them on the frames of input videos.

SRResNet

We implemented the SRResNet for image super-resolution with 16 layers based on the SRResNet in [2]. The activation function used in this ResNet is Leaky ReLU. SRResNet aims to optimize by minimizing pixel-wise error measurements between the input and output images using the mean square error (MSE). As a result, SRResNet is predicted to have good PSNR results, since PSNR is inversely proportional to MSE. To apply this neural network to videos, we split videos into frames and ran SRResNet on each frame of the video.

SRGAN

We implemented the SRGAN for image super-resolution with 16 layers based on the model in [2].

The idea behind training a GAN is that instead of trying to minimize the MSE like in the ResNet implementation, we train a discriminator to distinguish real images from super-resolved images and a generator that creates super-resolved images that are similar enough to fool the discriminator.

The generator network has B identical residual blocks. We use two sub-pixel convolutional layers with 3x3 kernels and 64 feature maps followed by batch-normalization layers. The activation function is ReLU.

The discriminator network has 8 convolutional layers with increasing numbers of 3x3 kernels (64, 64, 128, 128, 256, 256, 512, 512) Each time the number of features doubles, we use strided convolutions to reduce the image resolution. Afterwards, we have two dense layers and a sigmoid activation function returning the probability that the input image is real.

To apply this neural network to videos, we split videos into frames and ran SRGAN on each frame of the video.

Mixed Models

Our primary contribution is the synthesis of existing models into mixed models. We used an implementation of bicubic interpolation to upscale each frame in the video for the purpose of combining it with the SRResNet and SRGAN networks. To create the mixed models, we took every other frame from the output of one model and every other frame from the output of the bicubic model and interspersed them. Using this method, we created the SRResNet-Bicubic and SRGAN-Bicubic models. The logic behind this approach is that to enhance the overall resolution of videos, the best approach may not be to maximize the resolution of every frame, which may lead to perceived artifacts in the video without temporal smoothing, but instead combine different methods of super-resolution with a simple smooth upscaling. We also mixed the SRResNet and SRGAN networks to create the SRResNet-SRGAN model to see if it would produce a better result than each of the unmixed networks.

Training Details

All networks were trained on Amazon Web Services (AWS) p2.xlarge and p2.8xlarge GPU instances on the BSDS500 dataset containing 300 training images (distinct from our testing images). The SRResNet model was trained for 28 epochs while the SRGAN was trained for 140 epochs. We try to mimic the implementation from Ledig et al. [1] in order to achieve best possible results within limited time. More specifically, for the SRResNet, we use 16 residual blocks, 0.0 ReLU leakiness, with 1e-4 learning rate. As for SRGAN, we use [64, 64, 128, 128, 256, 256, 512, 512] filters per layer, [1, 2, 1, 2, 1, 2, 1, 2] strides, [1024, 1] dense layers, LReLU activation function for the discriminator; generator adversarial losses "gan" 0.001, generator loss "VGG19" 0.28, learning rate 1e-4 and decay factor 0.1 for the generator. In order to improve the performance based on the paper, we also use the Adam optimizer for both the networks.

Quantitative Evaluation

To evaluate our networks quantitatively, we will measure the peak signal to noise ratio (PSNR) and structural similarity index (SSIM) on our SRResNet and SRGAN networks when they are tested on images in the Set5 and Set14 datasets. The PSNR is calculated using the MSE between the input and output images:

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N}$$

$$PSNR = 10 \log_{10}\left(\frac{R^2}{MSE}\right)$$

In these equations, the images have dimensions MxN and R is 1, corresponding to the maximum fluctuation in the image data type.

SSIM is calculated on various windows in the input and output images using the following formula:

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

In this equation, x is the input image and y is the output image. Each μ corresponds to the average of a window, σ_x and σ_y corresponds to the variance of a window, and σ_{xy} is the covariance between the two windows. C_1 and C_2 are the constants 0.01 and 0.03, which help stabilize the SSIM value if the denominator is close to 0.

Higher PSNR means that there will be less noise in the images, but the images may also be blurry. Higher SSIM means that not only is there less noise in the image, but the edge sharpness in the image is also preserved.

4. Experimental Results

In this section, we present the 1) quantitative results of the different neural network models we trained, in comparison of the original papers' stats, 2) the ranking results of our user study.

4.1 Quantitative Results

As shown in the Table 1, our results of SRGAN are quite comparable to the original paper [2] we referred to in mean squared reconstruction error and improving high peak signal-to-noise ratios, for both Set5 and Set14 validation dataset. The SRResNet turns out be a bit off, which is probably due to down-sized training set and limited training time and resources. In order to achieve a more convincing result for user study, we found a pre-trained SRResNet model on GitHub, and implement the same training factor for another 140 epochs on BSDS500 training set, which then reaches the PSNR/SSIM of 31.9391/0.8959 for Set5 and 28.55/0.7881 for Set14.

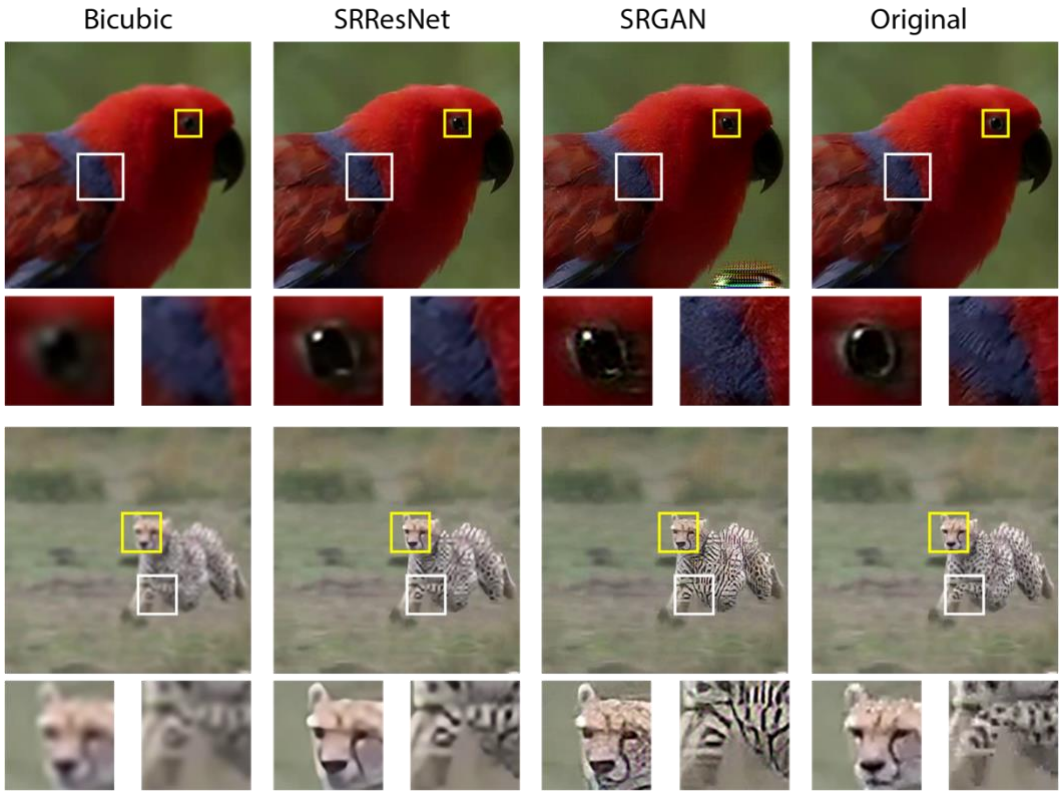

Figure 3: Bicubic, SRResNet and SRGAN reconstruction sample frames and corresponding reference HR frames. [4X upscaling]

Table 1: Performance of different loss functions for Bicubic, SRResNet and SRGAN. Both the reference and our self-trained models included. [4X upscale]

	Bicubic	**SRResNet** (paper)	**SRGAN** (paper)	**SRResNet** (ours)	**SRGAN** (ours)
Set5					
PSNR	28.43	32.05	29.40	27.7122	28.9800
SSIM	0.8211	0.9019	0.8472	0.7695	0.8738
Set14					
PSNR	25.99	28.49	26.02	24.8684	25.9061
SSIM	0.7486	0.8184	0.7397	0.6091	0.6884

4.2 User Study

We performed super-resolution by trying to upscale videos by a factor of 4. Sample videos (ground truth) were downsampled by scaling them down by a factor of 4, and the downscaled video was used as input for each of the networks. We performed a user study by running all five of our models (SRResNet, SRGAN, SRResNet-SRGAN, SRResNet-Bicubic, SRGAN) on two 10-second videos.

Because our SRResNet was only trained for 28 epochs, we used a pretrained SRResNet for the purposes of this user study. We had 10 users rank the quality of the (unlabeled) input videos along with the ground truth from best to worst. Our first video of a bird included slower movements and more saturated colors, while our second video of a cheetah included faster movements and more desaturated colors.

The results of our user study are summarized in Table 2. For the bird video, the ground truth was consistently recognized as the best video. The network that performed the best was SRResNet, followed by SRGAN and SRResNet-Bicubic, which performed equally well. The mixed models SRResNet-SRGAN and SRGAN-Bicubic consistently performed the worst. Users reported flickering in these videos as well as blurriness.

For the cheetah video, the ground truth video mostly ranked the best, but sometimes the SRGAN and SRGAN-Bicubic videos outranked even the ground truth. The SRResNet-SRGAN and SRGAN networks performed the best. SRResNet-Bicubic, which performed well on the bird video, appeared to perform poorly on the cheetah video.

Table 2: User rankings of two feature-weighted output videos, with various frame generating network implementations. [4X upscaling]

Frame Network	Video Features: Bird, Slow Movement, Color Saturated										Mean	Std distribution	Mode
	User #1	User #2	User #3	User #4	User #5	User #6	User #7	User #8	User #9	User #10			
Ground Truth	1	1	1	1	1	1	1	1	1	1	1	0	1
SRResNet	2	2	3	3	4	4	2	2	2	2	2.6	0.843	2
SRResNet&Bicubic	3	4	6	4	3	2	3	4	3	4	3.6	1.075	3
SRGAN	4	3	2	2	2	3	4	3	6	3	3.2	1.229	3
SRGAN&SRResNet	5	6	4	6	5	5	5	5	4	5	5	0.667	5
SRGAN&Bicubic	6	5	5	5	6	6	6	6	5	6	5.6	0.516	6

Frame Network	Video Features: Cheetah, Fast Movement, Color Desaturated										Mean	Std distribution	Mode
	User #1	User #2	User #3	User #4	User #5	User #6	User #7	User #8	User #9	User #10			
Ground Truth	1	1	3	1	4	1	1	1	1	2	1.6	1.075	1
SRResNet	2	3	5	4	5	4	5	2	4	4	3.8	1.135	4
SRResNet&Bicubic	5	5	6	6	6	6	6	4	3	6	5.3	1.059	6
SRGAN	3	2	1	3	2	2	4	3	6	1	2.7	1.494	3
SRGAN&SRResNet	4	4	2	2	1	3	2	5	5	3	3.1	1.370	2
SRGAN&Bicubic	6	6	4	5	3	5	3	6	2	5	4.5	1.434	6

5. Conclusion

We implemented two neural networks for image super-resolution, SRResNet and SRGAN, based on the description in the paper by Ledig et al. [2] We combined these networks with a simple bicubic interpolation on the frames to obtain three original neural networks. Based on the user study we conducted, the ResNet-based networks such as SRResNet and SRResNet-Bicubic performed the best at 4x upscaling videos with slower motion and saturated colors, while GAN-based networks such as SRResNet-SRGAN and SRGAN performed better at upscaling videos with faster motion and desaturated colors. We conclude that the implementation of SRResNet we used was better at maintaining sharpness because it was pretrained, and so it performed the best on videos where sharpness mattered more, such as when the movement is slower and the colors are brighter. Using SRGAN-based networks resulted in a smoother video, which is important when the movement in the video is fast. Therefore, they performed better than SRResNet when sharpness did not matter as much. Mixing the SRResNet and SRGAN models improved the sharpness of videos without creating noticeable artifacts when movement was fast, so our mixed model actually outperformed both the unmixed SRResNet and SRGAN models on the cheetah video.

One limitation we had was that we were unable to train both our SRResNet and SRGAN networks for a satisfactory number of epochs. In the future, we would like to be able to train both our SRResNet and SRGAN models on a large number of epochs. In addition, we would like to include temporal smoothing processing on output videos to reduce the flickering and extra artifacts that some of our mixed models introduced. Lastly, we would like to implement a smarter way of combining each of our models (SRResNet, SRGAN, bicubic upscaling) by optimizing on the movement speed and saturation in the video. For instance, a video that starts out in slow motion and speeds up could start with frames produced SRResNet and use more and more frames from SRGAN as the video continues.

6. Individual Contribution

In this work, I mainly implemented the neural network training for SRGAN model and the models validation. I also build the mixing-frame frameworks for the various model and video generating system. Both of the team member then contributed on the presentation and paper together.

References

[1] W. Shi, J. Caballero, F. Husz´ar, 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," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[2] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

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

[4] M. Sajjadi, R. Vemulaplli, and M. Brown. "Frame-Recurrent Video Super-Resolution," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018