iJRASET
International Journal For Research in
Applied Science and Engineering Technology

Volume: 10 Issue: XII Month of publication: December 2022

DOI: https://doi.org/10.22214/ijraset.2022.47861

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue XII Dec 2022- Available at www.ijraset.com

Video Chatting Platform with Video Enhancing Feature Using Super Resolution

Priya Nemani[1], Palvi singh[2], P Pushpanjali[3], Dr. Sudha Mishra[4]
[1, 2, 3]*Department of Information Technology, Guru Ghasidas University, Bilaspur,Chhattishgarh,India*
[4]*Assistant Professor, Department of Information Technology Guru Ghasidas University, Bilaspur, Chhattishgarh,India*

Abstract: This paper describes the process of creating a Multi-User Video Chatting Platform with Agora Video SDK and a Django Backend. The Video Super Resolution technique is being used to improve the quality of video calls. VSR is the process of reconstructing high-resolution videos from low-resolution ones. We employ the Dynamic Up sampling Filters Without Explicit Motion Compensation technique for VSR. We reconstruct an HR image directly from the input image using dynamic Up sampling filters, and the fine details are added using the computed residual.

Keywords: Agora, Super Resolution, Convolutional neural Network, Deep learning.

I. INTRODUCTION

Deep neural network-based VSR methods have recently made significant advances. However, published research on these methods is scarce. In comparison to previous methods, our network can generate much sharper HR videos with temporal consistency with the help of a new data augmentation technique. We also conduct extensive experiments on our network to demonstrate how it deals with motions implicitly. There are also architectural design and implementation details described. To the best of our knowledge, this project is expected to advance recent research in this field while also potentially deepening our understanding of VSR techniques. VSR has recently become increasingly important in providing HR content for ultra-high -definition displays. [3] While many deep learning-based VSR methods have been proposed, the majority of them rely heavily on motion estimation and compensation accuracy. In this project, we propose a novel end-to-end deep neural network in this paper that is fundamentally different from previous methods. Rather than explicitly computing and compensating for motion between input frames, the motion information is used to generate dynamic Up sampling filters implicitly [4] [5]. The HR frame is directly constructed by local filtering to the input center frame using the generated up sampling filters. We can generate much sharper and more temporally consistent HR videos because we do not rely on explicit motion computation and do not directly combine values from multiple frames. We achieve state-of-the-art performance compared to previous deep learning-based VSR algorithms by using a large number of training videos and a new data augmentation process. shows one example of how our method produces much sharper frames with less flickering than another method

II. LITERATURE REVIEW

1) The algorithm in Single Image Super-Resolution generates one high-resolution image from a single low-resolution image; it is thus a Single-Input-Single-Output approach. This task was traditionally accomplished by interpolating the values for missing pixels by taking an average of the neighbouring pixels using methods such as nearest neighbour, bilinear, and bicubic interpolation. Recently, several DL-based algorithms, such as SRGAN and ESRGAN [1], have been used to perform SISR and have been shown to outperform traditional methods.
2) Multiple low-resolution images [1] are used in this method to create one or more high-resolution target images. Because we fuse the information between the reference frame and the neighbouring frames, each neighbouring frame must be aligned in relation to the reference frame so that the resulting high-resolution frame is temporally aligned. This is typically caused by the varying motion of cameras or objects.
3) Existing approaches [2] employ inefficient alignment methodologies, such as exhaustive computation or inefficient implicit convolution-based alignment, with little regard for runtime. Most depend only on information from surrounding frames, ignoring the possibility of computation reuse between subsequent frames.
4) Conventional SR methods [2] often depend on redundancy and explicit motion estimates across video frames to efficiently rebuild a higher resolution output from several LR measurements. Although such conventional approaches can in theory result in a correct reconstruction of missing detail, their reliance on the quality of estimated motion between frames limits their ability to up - sample unconstrained real-world video with rapid motion, blur, occlusions, or presenting other common video processing challenges.

III. PROPOSED METHODOLOGY

A. Agora Implementation

To begin a session, do the following: When you join a channel, you can retrieve a token, which is a computer-generated string that authenticates the user. Retrieve the token from the Agora Console and create your own token generator, which you can then integrate into your production IAM system. Then call methods to create and join a channel, passing the same channel name to join the same channel. And now, all users in the channel send and receive video and audio streams.

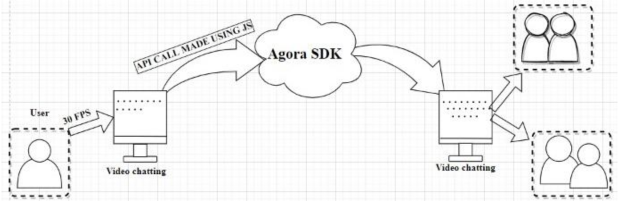


Fig.1. Once the project is set up, use the core APIs of the Agora Web SDK to implement the basic video call function.

This section shows how to use the Video SDK to implement Video Calling.

B. Dataset

One of the most important aspects of deep learning is the quantity and quality of training data. The training dataset must include videos with a wide range of complex real-world motions.

To achieve adequate generalisation We gathered 200 videos from the Internet that covered a wide range of topics, including education, creatures, activity, and geography. There are also a variety of textures and motions in the videos. We choose areas with enough motion to sample 160, 000 ground truth training data points at a spatial resolution of 144 × 144 [4].

C. Training

To obtain LR inputs, the ground truth training data are smoothed with a Gaussian filter before being subsampled with respect to the upscaling factor r. The spatial resolution of the input patch is fixed at 32 × 32. The mini-batch size is set to 16, and the method is used to initialize all variables. We use the Huber loss as the cost function for stable convergence and set the threshold to 0.01. We use the Adam optimizer, starting with a learning rate of 0.001 and increasing by 0.1 every 10 epochs [4]. Because of the direct up sample filtering, our approach's initial accuracy is quite high, and we have a good starting point for training.

The convergence speed is extremely fast when combined with residual learning, allowing us to complete the training in only 22 epochs [4]. We zero padded inputs in the temporal axis during the testing phase to keep the number of frames from decreasing.

D. Super Resolution Implementation

Unlike previous efforts, we are now tackling the challenge of developing a precise Super Resolution technique for high-quality video conferencing. As a result, we propose an approach based on recent advances. [4]

The goal of the VSR is to estimate HR frames from given LR frames. The LR frames are obtained by down sampling the corresponding GT frames, where t denotes the time step. With the VSR network G and network parameters θ:

$$\hat{Y}_t = G_\theta(X_{t-N:t+N})$$

An input tensor shape for G is T × H × W × C, where T = 2N + 1, H and W are the height and the width of the input LR frames, and C is the number of color channels. Corresponding output tensor shape is 1× rH ×rW ×C, where r is the upscaling factor. A set of input LR frames is used to generate a final HR frame by the network [4]. The input center frame is first locally filtered using dynamic up sampling filters, and the residual is then added to the upsampled result for the final output.

E. Dynamic Upsampling Filter

Traditional bilinear or bicubic Upsampling filter kernels are essentially fixed, with the only variation being a kernel shift based on the location of a newly created pixel in an upsampled image. [4] For those traditional up sampling processes, a set of 16 fixed kernels is used for the ×4 upsampling. They are quick, but they rarely restore sharp or textured regions. In contrast, we propose using dynamic upsampling filters based on the dynamic filter network (DFN). The upsampling filters in LR frames are generated locally and dynamically based on the spatio-temporal neighbourhood of each pixel. [9]

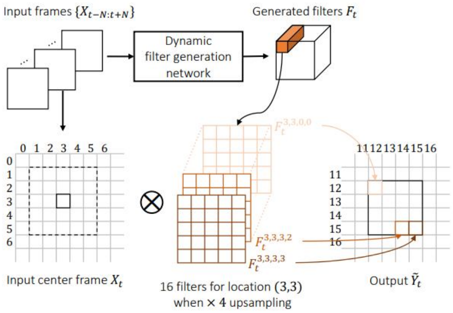


Fig.2. Without explicit motion compensation, dynamic up sampling is used. The figure shows an example of using the upscaling factor r = 4 to upscale a pixel at location (3, 3) in the centre input frame

Finally, each output HR pixel value is generated by applying the corresponding filter to an LR pixel in the input frame. The network can be trained from start to finish because it supports back-propagation. Our method differs significantly [4] from previous deep learning-based SR methods in that a deep neural network learns to reconstruct HR frames through a series of feature space convolutions. Instead, we train a deep neural network to learn the best up sampling filters, which are then used to reconstruct HR frames directly from LR frames. Because the filters are generated by looking at the spatio- temporal neighbourhoods of pixels, the dynamic filters are conceptually based on pixel motions, allowing us to avoid explicit motion compensation.

Because the result is still a weighted sum of input pixels after applying the dynamic upsampling filters alone, it lacks sharpness. Through linear filtering, some details may be lost. To address this, we estimate a residual image to enhance high frequency details. To generate the final output, a residual was added to the bicubically upsampled baseline. Our method differs in that the residual image is constructed from multiple input frames rather than a single input frame, and we use the dynamically upsampled frame as a better baseline that is then combined with the computed residual [4]. We can achieve spatial sharpness and temporal consistency in the resulting HR frames by combining these complementary components.

F. Network Design

Inorder to learn spatio-temporal features from video data, we need to replace 2D convolutional layers with 3D convolutional layers [6] [11] because it is more suitable in human action recognition and generic spatio-temporal feature extraction on video data. Each dense block [7] is made up of batch normalization, ReLU, 1×1×1 convolution, BN, ReLU, and 3×3×3 convolution in that order. Each input LR frame is first concatenated along the temporal axis by a shared 2D convolutional layer. [4] The resulting spatio-temporal feature maps are processed at separate branches that consist of several 2D convolutional layers to generate the two outputs after passing through our 3D dense block. The filtered output is combined with the generated residual to produce the final output.

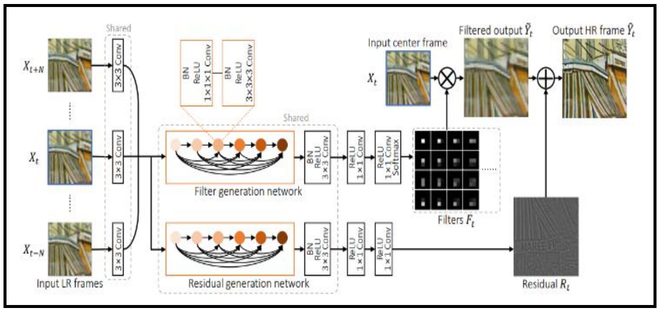


Fig.3. The Network architecture. The weights of filter and residual generation networks are shared for efficiency

We need corresponding training data to make the proposed network fully understand various and complex real-world motions. To generate such training data, we use a data augmentation in the temporal axis in addition to general data augmentation techniques such as random rotation and flipping. [4] The variable Temporal Augmentation, which determines the sampling interval of the temporal augmentation, is introduced here. For example, with Temporal Augmentation = 2, we will sample every other frame to simulate faster motion.

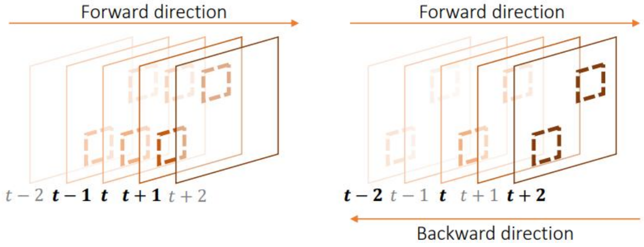
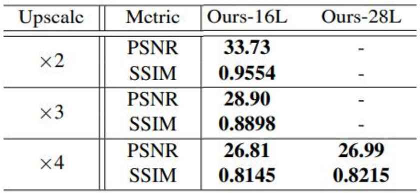


Fig.4. Data sampling from a video with a temporal radius of N = 1. With temporal augmentation, training data with faster or reverse motion can be sampled.

IV. RESULT

In addition to our basic 16-layer network, networks with 28 layers are tested. As with most super-resolution methods, scenes with thin lines that are close together are difficult to render, as demonstrated. [4] Increasing the number of layers in our algorithm yields better results for this type of difficult scene.

Fig.5 shows a quantitative comparison with other cutting-edge VSR methods. The results show that the network with more depth performs better, with the PSNR value of 28 layer increasing by 0.18dB from 16 layers with 0.2M more parameters [4]. Even with 16 Layer, we outperformed all other methods in terms of PSNR and SSIM for all upscale factors by a wide margin.



Upscale	Metric	Ours-16L	Ours-28L
×2	PSNR	**33.73**	-
	SSIM	**0.9554**	-
×3	PSNR	**28.90**	-
	SSIM	**0.8898**	-
×4	PSNR	**26.81**	**26.99**
	SSIM	**0.8145**	**0.8215**

Fig.5. Mean PSNR and SSIM are measured excluding spatial border pixels

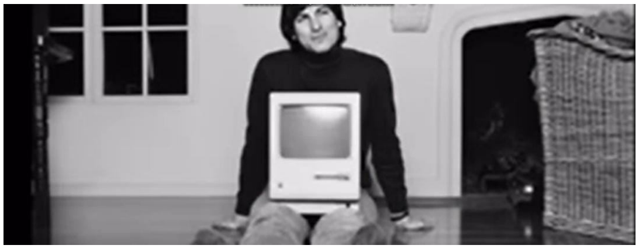


Fig.6. Qualitative comparison on videos: LR video Frame

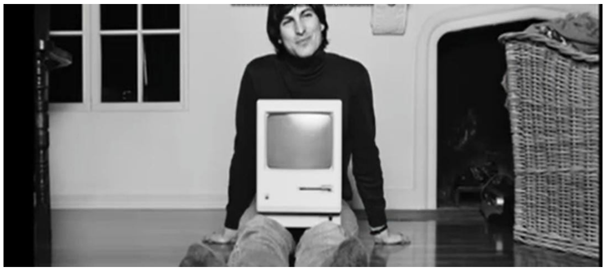


Fig.7. Qualitative comparison on videos: HR video Frame

V. CONCLUSION

Using our method, fine details and textures are better reconstructed. We outperform previous work and can see an improvement in performance with greater depth. More qualitative comparisons with other cutting-edge VSR methods are available. In comparison to other works, our results show sharper outputs with more smooth temporal transitions. The video we got as output has much less flickering.

In this paper, we present a new deep learning-based VSR framework that learns to output dynamic up sampling filters and the residual at the same time. With our new framework, we achieve cutting-edge performance, recover sharp HR frames, and maintain temporal consistency. Experiments have shown that our deep network can handle motion implicitly without explicit motion estimation and compensation. In our inference process, local filtering consumes roughly half of the runtime.

In the future, we hope to concentrate on speeding up our method to achieve real-time performance. We'd also like to expand our work to increase temporal resolution in addition to spatial resolution, such as creating a 60fps UHD video from a 30fps SD video. We can show a 20-100% boost in resolvability metric while running in real time (30fps) on a mid-segment consumer laptop.

AI elements such as real-time captioning, speech recognition, language translation, and facial alignment might be researched further for future development and can be added as features in video chatting platform.

VI. ACKNOWLEDGMENT

This paper and its research would not have been possible without the extraordinary assistance of our supervisor, Dr. Sudha Mishra. Her expertise and meticulous attention to detail have served as an inspiration and kept us on track to complete the final draught of this paper. The generosity and expertise of all have improved this study in numerable ways.

REFERENCES

[1] Neeboy Nogueira, Shawnon Guedes, Vaishnavi Mardolker, Amar Parab, Shailendra Aswale and Pratiksha Shetgaonkar, “Video Super Resolution Techniques: A Survey,” *(references)*

[2] Hongying Liu, Zhubo Ruan, Peng Zhao, Chao Dong, Fanhua Shang, Yuanyuan Liu, Linlin Yang and Radu TimofteI, “Video Super-Resolution Based on Deep Learning: A Comprehensive,” in https://arxiv.org/.

[3] Real-Time Video Super-Resolution on Smartphones with Deep Learning, Mobile AI 2021 Challenge: Report,” CVPR 2021 Workshop paper.

[4] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, Seon Joo Kim, Deep Video Super-Resolution Network Using Dynamic Upsampling Filters Without Explicit Motion Compensation, CVPR, 2018.

[5] 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. In 3231 CVPR, 2017

[6] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In ECCV, pages 184–199, 2014

[7] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In CVPR, 2017.

[8] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. TPAMI, 35(1):221– 231, 2013

[9] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang. Robust video super-resolution with learned temporal dynamics. In ICCV, 2017.

[10] Z. Ma, R. Liao, X. Tao, L. Xu, J. Jia, and E. Wu. Handling motion blur in multi-frame super-resolution. In CVPR, pages 5224–5232, 2015

[11] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In ICCV, pages 4489–4497, 2015

[12] T. Tong, G. Li, X. Liu, and Q. Gao. Image super-resolution using dense skip connections. In ICCV, 2017.

[13] M.S.M.Sajjadi, Raviteja Vemulapalli, Matthew Brown, “Frame-Recurrent Video Super-Resolution”.