Self-Enhanced Convolutional Network for Facial Video Hallucination

Chaowei Fang, Guanbin Li, *Member, IEEE*, Xiaoguang Han, *Member, IEEE*, and Yizhou Yu, *Fellow, IEEE*

***Abstract*— As a domain-specific super-resolution problem, facial image hallucination has enjoyed a series of breakthroughs thanks to the advances of deep convolutional neural networks. However, the direct migration of existing methods to video is still difficult to achieve good performance due to its lack of alignment and consistency modelling in temporal domain. Taking advantage of high inter-frame dependency in videos, we propose a self-enhanced convolutional network for facial video hallucination. It is implemented by making full usage of preceding super-resolved frames and a temporal window of adjacent low-resolution frames. Specifically, the algorithm first obtains the initial high-resolution inference of each frame by taking into consideration a sequence of consecutive low-resolution inputs through temporal consistency modelling. It further recurrently exploits the reconstructed results and intermediate features of a sequence of preceding frames to improve the initial super-resolution of the current frame by modelling the coherence of structural facial features across frames. Quantitative and qualitative evaluations demonstrate the superiority of the proposed algorithm against state-of-the-art methods. Moreover, our algorithm also achieves excellent performance in the task of general video super-resolution in a single-shot setting.**

***Index Terms*— Facial video hallucination, recurrent frame fusion, sequential feature encoding, deep learning.**

I. Introduction

FACE hallucination, also known as face super-resolution (SR), is a fundamental problem in computer vision because of its vast application scenarios, such as video surveillance, facial attribute analysis and visual content enhancement. Recently reconstructing static high-resolution (HR) face images from low-resolution (LR) ones has been widely studied [1], [2]. However the development of SR techniques in facial videos is far less explored due to its high complexity and requirement in effective spatio-temporal modelling. In this paper we focus on hallucinating high-resolution (HR) videos of talking faces from low-resolution (LR) ones. Faces in most of such videos do not have large or sudden motions, but with relatively small rotations.

Video super-resolution is a notorious ill-posed problem. The challenge of this problem resides in restoring individual frames with high-definition appearances while requiring natural inter-frame consistency and visual friendliness. Additionally a video SR method should exploit effective and relevant information from the rest of the video for signal reconstruction. Traditional methods [5], [6] mainly focus on reconstructing HR images via estimating blur kernel, inter-frame flow fields and extra noises. With the development and wide application of deep learning techniques, CNN-based methods turn out to be the mainstream in video super-resolution, which are significantly superior to traditional methods. One of the most intuitive solutions is to perform the restoration of the current frame by registering other adjacent frames and using CNN based feature fusion [3], [7], [8]. Unfortunately, this kind of method has the following shortcomings: 1) it is usually arduous to accurately register two frames within a long time interval which is highly likely to have a negative impact on subsequent fusion; 2) the fusion of all frames leads to a sharp increase in the amount of computation, which greatly reduces the overall efficiency; 3) using relatively small number of frames ignores much spatial and temporal information which could be otherwise very helpful. When directly applied to facial video hallucination, existing state-of-the-art video SR methods [3], [4] can successfully generate temporally coherent results with acceptable appearances in smooth regions such as cheek and nose. However they are not competent for the super-resolution reconstruction of image components with relatively complicate structures or textures, for example the regions of eyes and teeth as shown in Fig. 1.

To address the above issue, we present a so-called *Self-Enhanced Convolutional Network*, which is a novel end-to-end learning framework and can fully exploit both long-term spatial and temporal information for enhancing the hallucination inference of later frames. The self-enhancement of our method is inspired by the following two perspectives. 1) The spatial information of the preamble frames are crucial for restoring subsequent frames as there is a large amount of inter-frame redundancy especially in facial videos. Thus multiple super-resolved results of previous frames are propagated to enhance the prediction of subsequent frames. 2) Considering temporal information is paramount to reason the appearance of later

Manuscript received April 10, 2019; revised October 21, 2019; accepted November 13, 2019. Date of publication December 3, 2019; date of current version January 28, 2020. This work was supported in part by the Hong Kong Research Grants Council through Research Impact Fund under Grant R-5001-18, in part by the National Natural Science Foundation of China under Grant 61976250 and Grant U1811463, and in part by the Fundamental Research Funds for the Central Universities under Grant 18lgpy63, and in part by the Pearl River Talent Recruitment Program Innovative and Entrepreneurial Teams in 2017 under Grant 2017ZT07X152, and in part by the Shenzhen Fundamental Research Fund under Grant KQTD2015033114415450 and Grant ZDSYS201707251409055, and in part by Department of Science and Technology of Guangdong Province Fund under Grant 2018B030338001. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Lisimachos P. Kondi. *(Corresponding author: Yizhou Yu.)*

C. Fang was with the Department of Computer Science, The University of Hong Kong, Hong Kong. He is now with the School of Artificial Intelligence, Xidian University, Xi'an 710071, China.

G. Li is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China.

X. Han is with the Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Hong Kong.

Y. Yu is with the Department of Computer Science, The University of Hong Kong, Hong Kong (e-mail: yizhouy@acm.org).

Digital Object Identifier 10.1109/TIP.2019.2955640

Fig. 1. Our method aims to generate high-resolution facial video frames from low-resolution inputs. Taking a specific frame as an example, existing video super-resolution methods SPMC [3] and FRVSR [4] have difficulty in recovering components with complicate structures such as the right eye (green box) and teeth (blue box). Our method is capable of achieving more promising results. The input image is visualized using pixel duplication. 'GT' represents ground truth image. (Best viewed in close-up).

frames, ConvLSTM [9] is applied to enhance the feature representation of every frame by sequentially encoding the features from the past frames. The self-enhancement model is implemented through an encoder-decoder architecture. It absorbs in an initial prediction of current frame and registered HR frames of past frames, resulting in a refinement map of the initial prediction. The feature representation of each frame resides in-between the encoder and decoder. To involve in information of future frames and further boost the SR performance, neighboring LR images are used to generate the initial HR estimation for each frame via a local frame fusion network. Except for facial videos, our proposed method also has a strong advantage for the super-resolution of general videos as it is particularly good at learning and capturing structural and temporal consistency, especially for the reconstruction of scenes with intricate and trifling structures (e.g. buildings).

In summary, this paper has the following contributions.

- A self-enhanced convolutional network is proposed for facial video hallucination. The uniqueness of our model is that it makes use of both spatial and temporal information across all preceding frames.
- Three ConvLSTM-based recurrence strategies are devised to excavate temporal information for enhancing the feature representation of every frame.
- Our proposed method has achieved state-of-the-art performance in: two facial video datasets, VoxCeleb [10] and RAVDESS [11]; two single-shot generic video datasets, VID4 [5] and Harmonic collected from the Internet.

II. Related Work

Image/video super resolution has been studied for a long time. We refer to [12] for a detailed survey. In this section, we mainly discuss the related works based on deep learning.

A. Image/Video Super-Resolution

The basic idea of recent deep learning based methods [13], [14] is to design a CNN architecture to map low resolution images to their HR versions. In [15], an improved version of [14] is proposed with the help of convolutional non-local operation [16]. A novel super-resolution method is developed in [17] using residual-in-residual dense blocks [18]. During the training stage, Relativistic GAN [19] is employed for achieving realistic predictions. Video super resolution, as an extension of image super resolution, attracts more attentions for its practicality but being more challenging. To extend single frame SR model to its multi-frame version, [7] attempts to utilize multiple motion compensated frames/features when super-resolving each frame. Reference [8] and [3] utilize consecutive neighboring frames to produce the super-resolution output of the current frame with the help of flow based motion correction. A joint upsampling and warping operation [20] is proposed for fusing neighboring frames in video super-resolution. Reference [21] super-resolves every LR image using multiple frames via learning a dynamic upsampling filter for each pixel in the target HR image and a residual image. Reference [22] devises a multi-scale temporal adaptive neural network and a spatial alignment network for utilizing inter-frame dependency in video super-resolution. Considering high inter-frame repeatability in videos, a frame recurrence strategy is proposed in [4] to propagate the spatial information of previously estimated HR frames to all subsequent frames and enhance their HR predictions. As only one previous frame is fused into the inference procedure of current frame, temporal connection across frames is weak which might miss lots of inter-frame spatial dependencies, especially information provided by future frames. Our method differs it from two perspectives. First, multiple neighboring LR frames are utilized to generate an initial prediction through a local frame fusion network. Second, ConvLSTM-based recurrence module is devised to enhance the feature representation of every frame. Reference [23] devises a bidirectional recurrent convolutional network to learn long-term temporal and contextual information for video super-resolution. But each frame relies on intermediate features from both past and subsequent frames. All images in the input clip are required to be processed simultaneously and the memory cost grows linearly with respect to its length. Our devised bidirectional recurrent module avoids this shortcoming as each input frame can be super-resolved independently after obtaining features of previous frames.

B. Face Hallucination

As a special case of image/video super-resolution, facial image hallucination has drawn much more attentions due to its wider application scenarios. Most deep neural network based methods attempt to integrate facial prior knowledge into the CNN architectures. Reference [24], [25] implicitly exploit global facial features learned using fully connected layers. Reference [1] utilizes a reinforcement learning policy to generate HR face image patch by patch iteratively. Reference [2], [26], [27] explicitly make use of facial priors (landmarks/parsing maps) to help inferring the restoration of HR face images or training neural networks. Wavelet coefficients of HR images are inferred from the embedded features of the low resolution faces and are then used to reconstruct the

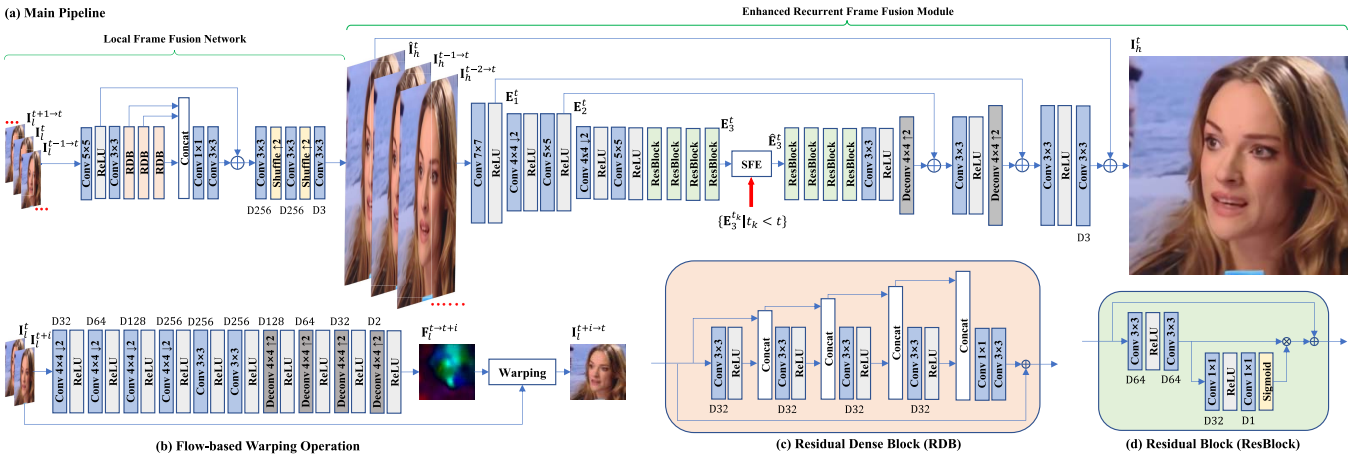


Fig. 2. Our network architecture for super-resolving 64 × 64 images to 256 × 256 ones. 'D[D]' represents a convolution or deconvolution layer with the number of output channels set to D. The $x \times y$ beside the box of convolution/deconvolution layer indicates kernel size. '↑ 2' means the deconvolution layer upsample the feature map to 4 times while '↓ 2' means the convolution layer downsample the feature map to one quarter. The pixel shuffle layer rearranges the input $s^2D \times H \times W$ tensor to a $D \times sH \times sW$ tensor. $s = 2$ for each shuffle layer. The main pipeline (a) of our method consists of two stages. The first one utilizes a local frame fusion network built on residual dense blocks (c) to produce an initial super-resolution inference. The second stage takes advantage of the previously estimated results and encodes feature maps of past frames sequentially (SFE) to enhance the feature representation of current frame. The adopted residual block is shown in (d). The optical flow module is shown in (b).

expected HR image [28]. Based on PixelCNN [29], a novel face image super-resolution model [30] is set up to recurrently reconstruct every pixel. This method is hard to restore images with large spatial sizes because of its computational cost. On the other hand, generative adversarial models [31] are widely used in face super-resolution. UR-DGN [32] is claimed to be the first face SR method using generative adversarial network. Reference [33] discusses the efficacy of Wasserstein GAN [34], [35] in training face SR networks. Reference [36] learns a CNN model to super-resolve blurry face and text images with a complicated objective function consisting of pixel-wise MSE, feature matching and adversarial loss. As far as we know, no literature published by conference/journal on face video SR using deep learning is found. Without additional constraints, single face image SR methods can hardly work well in face video SR because of the deformity to guarantee smoothness across frames.

III. Method

Denote a sequence of facial video frames as $\{\mathbf{X}^t\}$, where $t \in [1, N]$ and $t \in \mathcal{N}_+$. $\mathbf{X}^t$ is a single frame with resolution $w \times h$. Facial video hallucination aims to generate the high-resolution counterpart $\mathbb{Y} = \{\mathbf{Y}^t\}$ composed of frames with resolution $rw \times rh$ where r is the upscaling factor. In the following, we first give an overview of our proposed network architecture and then describe the details of each module.

A. Self-Enhanced Convolutional Architecture

The overall architecture of our proposed self-enhanced convolutional network is illustrated in Fig. 2. The base of our method consists of two cascaded subnetworks. The first subnetwork is named as local frame fusion network, which takes multiple aligned neighbouring LR frames as input and aims at generating an initial super-resolved result for each independent frame. The second subnetwork is named as enhanced recurrent frame fusion module which refines the result of the local frame fusion network with the help of aligned super-resolved images and features from previous frames.

B. Local Frame Fusion Network

For sake of involving in information of future frames and providing a high starting point for subsequent subnetwork, we set up a local frame fusion network based on the residual dense network (RDN [37]). $2T_1 + 1$ consecutive LR frames $\{\mathbf{X}^k | k \in [t - T_1, t + T_1]\}$ are used as the input when super-resolving frame t. First of all, to make up inter-frame differences caused by facial/camera motions, an optical flow module is exploited to warp every LR image $\mathbf{X}^k$ to frame t as shown in Fig. 2 (b). Practically the optical flow field $\mathbf{F}^{t\to k}$ from $\mathbf{X}^t$ to $\mathbf{X}^k$ is used to bi-linearly sample an aligned counterpart of $\mathbf{X}^k$. We define the warped result of $\mathbf{X}^k$ as $\mathbf{X}^{k\to t}$. The mean square error loss with total variation regularization is imposed on the optical flow module,

$$L_f^{t,k} = \frac{1}{cwh}\|\mathbf{X}^{k\to t} - \mathbf{X}^t\|_2^2 + \frac{\alpha}{2wh}(\|\nabla_x \mathbf{F}^{t\to k}\|_2^2 + \|\nabla_y \mathbf{F}^{t\to k}\|_2^2), \tag{1}$$

where α is a constant and c is the channel of input image. ∇_x and ∇_y are horizontal and vertical derivation operation respectively. The overall loss function for training the optical flow network is as follows,

$$L_f^t = \frac{\gamma}{2T_1} \sum_{k=t-T_1, k\neq 0}^{t+T_1} L_f^{t,k}. \tag{2}$$

RDN is a state-of-the-art SR method for static image SR method. We extend it into a multi-frame version by replacing the single input image with the concatenation of aligned LR images $\{\mathbf{X}^{k\to t}\}$. Let the output be $\hat{\mathbf{Y}}^t$. We use the following loss function for training the local frame fusion network,

$$L_l^t = \frac{1}{cr^2wh}\|\hat{\mathbf{Y}}^t - \mathbf{G}^t\|_2^2, \tag{3}$$

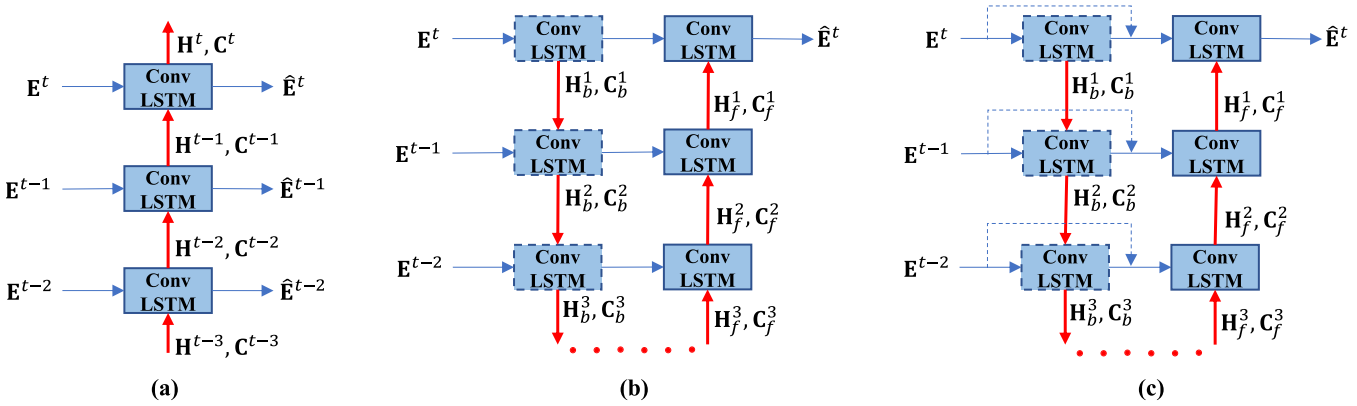


Fig. 3. Three sequential feature encoding strategies. The left (a) propagates the cell output and hidden variable to next frame recurrently. The second (b) collects T_3 features from past frames and employs bidirectional ConvLSTM to encode them at every frame. In the third (c) strategy, the input feature maps are also fed into the forward pass.

where $\mathbf{G}^t$ represents the ground-truth image of frame t. Details about the network architecture and residual dense blocks are presented in Fig. 2 (a) and (c).

C. Enhanced Recurrent Frame Fusion Module

1) Encoder-Decoder Framework: Super-resolving HR images from LR images requires recovering both accurate global appearances and visual friendly details such as textures and sharp edges. Inspired by [38], we adopt an encoder-decoder framework with skip connections to extract multi-scale convolutional features. Detailed network architecture is presented in right part of Fig. 2 (a). The encoder module extracts three scales of features using convolution layers, ReLUs [39] and residual blocks [40] from the input. Features in the first two scales are responsible for restoring details. They are forwarded to the decoder via skip connections. Compared to the first two scales, 4 extra residual blocks are applied in the third scale for sake of enlarging receptive field and producing deeper features. Denote feature map in the i-th scale as $\mathbf{E}_i^t$. The decoder possesses almost symmetric architecture with the encoder, generating the final restored result $\mathbf{Y}^t$ through refining the initial super-resolved result $\hat{\mathbf{Y}}^t$. It should be noted in particular that: 1) The input of our encoder-decoder model is formed by concatenating $\hat{\mathbf{Y}}^t$ and the aligned super-resolved images of previous frames; 2) Feature maps of past frames in the third scale $\{\mathbf{E}_3^k | k < t\}$ are accumulated to enhance $\mathbf{E}_3^t$ to $\hat{\mathbf{E}}_3^t$ via sequential encoding strategies which will be introduced in Section III-C.3. The following mean square error loss function is exploited to train the above model,

$$L_e^t = \frac{1}{cr^2wh}\|\mathbf{Y}^t - \mathbf{G}^t\|_2^2. \tag{4}$$

2) Recurrent Frame Fusion: Because of the high dependency across frames in video, propagating super-resolved result of previous frame is helpful to infer the result of current frame in video super-resolution [4]. Here we take advantage of multiple previous frames $\{\mathbf{Y}^k | k = t-1, \cdots, t-T_2\}$ when predicting HR image of frame t. Each previous frame $\mathbf{Y}^k$ is firstly aligned to frame t using the bi-linearly interpolated optical flow field $\mathbf{F}^{t\to k}$. Suppose the aligned result be $\mathbf{Y}^{k\to t}$. Afterwards $\{\mathbf{Y}^{k\to t} | k = t-1, \cdots, t-T_2\}$ are concatenated with $\hat{\mathbf{Y}}^t$ and then fed into the encoder. When $T_2 \le T_1$, warping past HR frames does not entail much more computational cost as all optical flow fields have been calculated in Section III-B. The differences with [4] are that multiple previous frames are utilized and no space-to-depth transformation is required to convert the HR image into a tensor with same spatial size as the LR image.

Fig. 4. Given a sequence of low-resolution images (1st row), preliminary HR images are generated using the local frame fusion network (2nd row). Then it is refined by our enhanced recurrent frame fusion module (3rd row).

3) Sequential Feature Encoding: Feature-level temporal information benefits facial video hallucination from the following perspectives: facial motions could be used to infer future frames which is paramount to restore lost appearance information in future frames; motions in most regions of talking faces are usually not severe, making spatial dependencies in high-level feature maps could be easily obtained for ensuring

Algorithm 1 One-Way ConvLSTM

Input: Feature of current frame: $\mathbf{E}^t$;
ConvLSTM variables of previous frame: $\mathbf{H}^{t-1}$, $\mathbf{C}^{t-1}$;
ConvLSTM parameters: θ_l.
Output: Enhanced feature: $\hat{\mathbf{E}}^t$.
1: $\mathbf{H}^t, \mathbf{C}^t = \text{ConvLSTM}(\mathbf{H}^{t-1}, \mathbf{C}^{t-1}, \mathbf{E}^t, \theta_l)$.
2: $\hat{\mathbf{E}}^t \leftarrow \mathbf{H}^t$.

content coherence in restored videos. Considering the above two points, we adopt ConvLSTM [9] to extract temporal information for enhancing the feature representation of current frame. Given a sequence of input features $\{\mathbf{E}^k\}$. We summarize the formulation of ConvLSTM as follows,

$$\mathbf{H}^k, \mathbf{C}^k = \text{ConvLSTM}(\mathbf{H}^{k-1}, \mathbf{C}^{k-1}, \mathbf{E}^k, \theta_l), \tag{5}$$

where θ_l contains all the weights and biases of convolution kernels in the ConvLSTM cell. $\mathbf{H}^k$ and $\mathbf{C}^k$ is hidden state and cell output respectively. $\mathbf{H}^0 = \mathbf{C}^0 = \mathbf{0}$. Detailed computation steps of ConvLSTM are given below.

1. $\mathbf{A}_i^k = \varsigma(\mathbf{E}^k * \mathbf{W}_{ei} + \mathbf{H}^{k-1} * \mathbf{W}_{hi} + \mathbf{b}_i)$;
2. $\mathbf{A}_f^k = \varsigma(\mathbf{E}^k * \mathbf{W}_{ef} + \mathbf{H}^{k-1} * \mathbf{W}_{hf} + \mathbf{b}_f)$;
3. $\mathbf{A}_g^k = \tanh(\mathbf{E}^k * \mathbf{W}_{eg} + \mathbf{H}^{k-1} * \mathbf{W}_{hg} + \mathbf{b}_g)$;
4. $\mathbf{C}^k = \mathbf{A}_f^k \circ \mathbf{C}^{k-1} + \mathbf{A}_i^k \circ \mathbf{A}_g^k$;
5. $\mathbf{A}_o^k = \varsigma(\mathbf{E}^k * \mathbf{W}_{io} + \mathbf{H}^{k-1} * \mathbf{W}_{ho} + \mathbf{b}_o)$;
6. $\mathbf{H}^k = \mathbf{A}_o^k \circ \tanh(\mathbf{C}^k)$,

where $\varsigma(\cdot)$ is the Sigmoid function. $\mathbf{W}$-s and $\mathbf{b}$-s are the weights and biases of convolution kernels with size of 3×3. '$\circ$' represents the Hadamard product. $\mathbf{A}_i^k$, $\mathbf{A}_f^k$ and $\mathbf{A}_o^k$ represent input, forget and output gate for the k-th data sample, respectively.

We provide three strategies to encode collected features based on ConvLSTM units: one-way ConvLSTM, cascaded bidirectional ConvLSTM and fused bidirectional ConvLSTM.

One-way ConvLSTM: To capture long-term temporal information, we propagate the cell output $\mathbf{C}$ and hidden state $\mathbf{H}$ to next frame recurrently as shown in Fig. 3 (a). Consequently at any frame t all past features $\{\mathbf{E}_3^k | k < t\}$ are exploited to enhance $\mathbf{E}_3^t$,

$$\mathbf{H}^t, \mathbf{C}^t = \text{ConvLSTM}(\mathbf{H}^{t-1}, \mathbf{C}^{t-1}, \mathbf{E}_3^t, \theta_l). \tag{6}$$

The result of the enhanced feature $\hat{\mathbf{E}}_3^t$ is $\mathbf{H}^t$ exactly. The computation procedure is concluded in Algorithm 1.

Cascaded Bidirectional ConvLSTM: According to [41], bidirectional RNN framework outperforms regular recurrent model with one-way pass. Thus we can devise a bidirectional ConvLSTM module as shown in Fig. 3 (b). Here only T_3 past features at most should be considered, preventing the time and memory cost from increasing continuously as t grows. Then one backward and forward passes are adopted to processing the ordered feature sequence $\mathbb{E}^t = \{\mathbf{E}_3^k | k = t, \cdots, t - T_t + 1;\ T_t = \min(t, T_3 + 1)\}$,

$$\mathbf{H}_b^k, \mathbf{C}_b^k = \text{ConvLSTM}(\mathbf{H}_b^{k+1}, \mathbf{C}_b^{k+1}, \mathbf{E}_3^k, \theta_l^b); \tag{7}$$

$$\mathbf{H}_f^k, \mathbf{C}_f^k = \text{ConvLSTM}(\mathbf{H}_f^{k-1}, \mathbf{C}_f^{k-1}, \mathbf{H}_b^k, \theta_l^f). \tag{8}$$

$\mathbf{H}_b^k$, $\mathbf{C}_b^k$ and θ_l^b are the hidden state, cell output and parameter of the backward pass respectively while $\mathbf{H}_f^k$, $\mathbf{C}_f^k$ and θ_l^f represents the hidden state, cell output and parameter of the forward pass respectively. The final result $\hat{\mathbf{E}}_3^t$ is $\mathbf{H}_f^1$. $\mathbf{H}_b^{t+1} = \mathbf{C}_b^{t+1} = \mathbf{0}$. $\mathbf{H}_f^{t-T_t} = \mathbf{C}_f^{t-T_t} = \mathbf{0}$. The computation procedure is summarized in Algorithm 2.

Algorithm 2 Cascaded Bidirectional ConvLSTM

Input: Features: $\{\mathbf{E}^k | k = t, \cdots, t - T_t + 1\}$;
ConvLSTM parameters: θ_l^b, θ_l^f.
Output: Enhanced feature: $\hat{\mathbf{E}}^t$.
1: $\mathbf{H}_b^{t+1} = \mathbf{C}_b^{t+1} = \mathbf{H}_f^{t-T_t} = \mathbf{C}_f^{t-T_t} = \mathbf{0}$.
2: **for** $k = t$ to $t - T_t + 1$ **do**
3: $\quad \mathbf{H}_b^k, \mathbf{C}_b^k = \text{ConvLSTM}(\mathbf{H}_b^{k+1}, \mathbf{C}_b^{k+1}, \mathbf{E}^k, \theta_l^b)$.
4: **end for**
5: **for** $k = t - T_t + 1$ to t **do**
6: $\quad \mathbf{H}_f^k, \mathbf{C}_f^k = \text{ConvLSTM}(\mathbf{H}_f^{k-1}, \mathbf{C}_f^{k-1}, \mathbf{H}_b^k, \theta_l^f)$.
7: **end for**
8: $\hat{\mathbf{E}}^t \leftarrow \mathbf{H}_f^t$.

Algorithm 3 Fused Bidirectional ConvLSTM

Input: Features: $\{\mathbf{E}^k | k = t, \cdots, t - T_t + 1\}$;
ConvLSTM parameters: θ_l^b, θ_l^f.
Output: Enhanced feature: $\hat{\mathbf{E}}^t$.
1: $\mathbf{H}_b^{t+1} = \mathbf{C}_b^{t+1} = \mathbf{H}_f^{t-T_t} = \mathbf{C}_f^{t-T_t} = \mathbf{0}$.
2: **for** $k = t$ to $t - T_t + 1$ **do**
3: $\quad \mathbf{H}_b^k, \mathbf{C}_b^k = \text{ConvLSTM}(\mathbf{H}_b^{k+1}, \mathbf{C}_b^{k+1}, \mathbf{E}^k, \theta_l^b)$.
4: **end for**
5: **for** $k = t - T_t + 1$ to t **do**
6: $\quad \mathbf{H}_f^k, \mathbf{C}_f^k = \text{ConvLSTM}(\mathbf{H}_f^{k-1}, \mathbf{C}_f^{k-1}, [\mathbf{H}_b^k, \mathbf{E}^k], \theta_l^f)$.
7: **end for**
8: $\hat{\mathbf{E}}^t \leftarrow \mathbf{H}_f^t$.

Fused Bidirectional ConvLSTM: To prevent loss of forward motion information, we devise another sequential feature encoding strategy as shown in Fig. 3 (c). Feature maps are not only fed into the ConvLSTM cell of the backward pass, but also constitute proportion of the input of the forward ConvLSTM cell as shown in Algorithm 3. The backward pass is the same as (7) while the forward pass (8) is replaced with the following procedure,

$$\mathbf{H}_f^k, \mathbf{C}_f^k = \text{ConvLSTM}(\mathbf{H}_f^{k+1}, \mathbf{C}_f^{k+1}, [\mathbf{H}_b^k, \mathbf{E}_3^k], \theta_l^f). \tag{9}$$

4) Self-Learned Attention: Spatial attention is significant in facial image hallucination as faces consist of specific components. Inspired by [42], we integrate a spatial attention mechanism into the residual block as shown in Fig. 2 (d). Two convolution layers and one ReLU layer are used to produce a spatial attention map, which is subsequently applied to suppress activations of pixels with low attention values. This attention mechanism enables every residual block to emphasize particular regions. Examples of our self-learned attention maps are presented in Fig. 5.

D. Network Training

Summing up (2) (3) and (4), we can obtain the overall training loss,

$$L = \frac{1}{N} \sum_{t=1}^{N} (L_e^t + L_l^t + \gamma L_f^t), \tag{10}$$

TABLE I
DATASETS USED IN FACIAL AND GENERIC VIDEO SUPER-RESOLUTION

	training				validation				testing			
	persons	sequences	frames	size	persons	sequences	frames	size	persons	sequences	frames	size
VoxCeleb	922	140,334	1,102,792	280×280	4	14	1,209	256×256	80	334	28,636	256×256
RAVDESS	0	0	0	-	0	0	0	-	24	96	10,013	256×256
Harmonic	-	7,456	1,163,056	300×300	-	4	798	512×512	-	51	9,335	512×640
VID4	-	-	-	-	-	-	-	-	-	4	171	not fixed

Algorithm 4 One Training Step of Our Self-Enhanced Convolutional Network

Input: LR and ground-truth image sequences: $\mathcal{X} = \{\mathbf{X}^t\}$ and $\mathcal{Y} = \{\mathbf{Y}^t\}$ where $t = 1, \cdots, N$; initialized network parameters: θ.
Output: Optimized network parameters: θ.
1: $\mathrm{L} \leftarrow 0$, $\mathcal{E} \leftarrow \varnothing$, $\mathcal{Y} \leftarrow \varnothing$.
2: **for** $t = 1$ to N **do**
3: Fetch LR frames from $\mathcal{X}$: $\mathbb{X}^t = \{\mathbf{X}^k | k = t - T_1, \cdots, t + T_1\}$. If $k \leq 0$, $\mathbf{X}^k = \mathbf{X}^1$; if $k > N$, $\mathbf{X}^k = \mathbf{X}^N$.
4: Compute motion fields from frame $\mathbf{X}^t$ to all frames in $\mathbb{X}^t$: $\{\mathbf{F}^{t \to k} | k = t - T_1, \cdots, t + T_1;\ \mathbf{F}^{t \to t} = \mathbf{0}\}$.
5: Warp $\mathbb{X}^t$ to $\tilde{\mathbb{X}}^t = \{\mathbf{X}^{k \to t}\}$ using above motion fields.
6: Compute the initial SR result $\hat{\mathbf{Y}}^t$ using the local frame fusion network in Section II-B with input $\tilde{\mathbb{X}}^t$.
7: Fetch super-resolved results from $\mathcal{Y}$: $\mathbb{Y}^t = \{\mathbf{Y}^k | k = t - 1, \cdots, t - T_2;\}$. If $k \leq 0$, $\mathbf{Y}^k = \mathbf{0}$.
8: Warp frames in $\mathbb{Y}^t$ to frame t using bi-linearly upsampled motion fields, forming $\tilde{\mathbb{Y}}^t = \{\mathbf{Y}^{k \to t}\}$.
9: Input $\tilde{\mathbb{Y}}^t$ and $\hat{\mathbf{X}}^t$ into the encoder of recurrent frame fusion module in Section II-C and extract feature maps $\mathbf{E}_1^t$, $\mathbf{E}_2^t$ and $\mathbf{E}_3^t$.
10: Fetch feature maps from $\mathcal{E}$: $\mathbb{E}^t = \{\mathbf{E}_3^k | k = t - 1, \cdots, t - T_t + 1;\ T_t = \min(t, T_3 + 1)\}$.
11: Sequentially encode $\{\mathbf{E}_3^t\} \bigcup \mathbb{E}^t$ into $\hat{\mathbf{E}}_3^t$ using ConvLSTM cells.
12: Feed $\mathbf{E}_1^t$, $\mathbf{E}_2^t$ and $\hat{\mathbf{E}}_3^t$ into the decoder, resulting to $\mathbf{Y}^t$.
13: Compute L_f^t, L_l^t, and L_e^t according to (2), (3) and (4) respectively.
14: $L \leftarrow L + L_e^t + L_l^t + \gamma L_f^t$.
15: Append $\mathbf{E}_3^t$ and $\mathbf{Y}^t$ into $\mathcal{E}$ and $\mathcal{Y}$ respectively.
16: **end for**
17: $L \leftarrow L/N$; update θ using Adam.

where γ is a constant. The loss function (10) is optimized using Adam [43] with learning rate of 10^{-4}. One step of optimization is illustrated in Algorithm 4. During the inference stage, only those super-resolved images and features required in next frame are preserved at the end of each frame.

IV. EXPERIMENTS

A. Dataset

Two facial video datasets, VoxCeleb [10] and RAVDESS [11], are used to validate the performance of our method. In addition, we also use generic single-shot videos from Harmonic and VID4 [5] to test deep video super-resolution methods. Table I presents the split of training, validation and testing sets.

(1) The **VoxCeleb** dataset contains over 100,000 utterances by 1,251 celebrities, providing sequences of tracked faces in the form of bounding boxes. We select out 140,334 sequences of face images with high quality. For each sequence, we compute a box enclosing the faces from all frames and use it to crop face images from the original video. All face images are resized to 280×280. Only the central 256×256 region is used in validating and testing.

Fig. 5. Visualizations of self-learned spatial attention maps from the 1st, 2nd, 4th and 6th residual blocks.

(2) The **RAVDESS** dataset encloses 2,452 sequences captured from 24 persons speaking/singing with various expressions and motions. We choose 4 sequences for each person forming the other testing set of facial video hallucination.

(3) The **Harmonic** dataset includes 18 videos captured from natural scenes containing buildings, birds, animals, etc. Sequences without scene switching are manually selected to serve as our single-shot video super-resolution dataset. Images are resized to 540×960. For training set we uniformly sample 8 sequences of 300×300 images from every video clip along horizontal axis with stride of 220 and vertical axis with stride of 240. Validation set is generated via cropping the centering 512×512 regions. Testing images with size of 512×640 are also cropped out in the center.

(4) The **VID4** dataset containing 4 sequences ('calendar', 'city', 'foliage' and 'walk') of images has been widely used to validate video SR methods.

Input LR images are synthesized through blurring HR images with a Gaussian kernel (standard deviation of 1.5), and then downscaling them via sampling 1 pixel out of every 4 pixels in each dimension. The following strategies are adopted for data augmentation during the training stage:

1) Random shuffle is utilized to reorganize the order of image sequences in each epoch;
2) N consecutive frames are selected from each sequence in the batch via starting at a random position and sampling one frame out of every l frames where l is random integer

Fig. 6. Comparison of super-resolution algorithms in two facial image sequences. The predicted SR images from our method ('Ours') are closer to the ground truth than other algorithms.

within $[1, 2]$ for facial videos and $[1, 4]$ for general videos;

3) 256×256 patches are randomly cropped from HR images of these frames, serving as ground-truth;
4) The chronologically order of the selected frames is reversed randomly;
5) Images are randomly flipped horizontally.

Parameters & Training Settings N, α and γ is set as 8, 0.01 and 0.1 respectively. According to the discussion in [8], we fix the value of T_1 as 2 in this paper. Without specification, T_2 and T_3 is set to 2 and 6 respectively; 8 residual blocks are adopted. In the recurrent frame fusion part, the error back propagation to the optical flow module is cut off to alleviate the instability in training. The local frame fusion network is pretrained for 2×10^5 iterations. Then the overall model is trained for the other 1.5×10^5 steps. For the generic video SR task, models are additionally finetuned for 5×10^4 iterations using sequences of 512×512 ground-truth images randomly cropped from the original 540×960 HR images. Batch size in each training iteration is set as 4. We periodically (every 200 iterations) test the model in the validation set. The version with the best performance is regarded as the final model. 4 TITAN Xp 12GB GPUs are utilized for training. Network parameters are initialized by default in PyTorch.

Abbreviations For conciseness, we use the following abbreviations to mark variants/settings of our method: 'SECNet', self-enhanced convolutional network as shown in Fig. 2; 'LFFNet', local frame fusion network as described in Section III-B; 'ERFFNet', network formed from the enhanced frame fusion

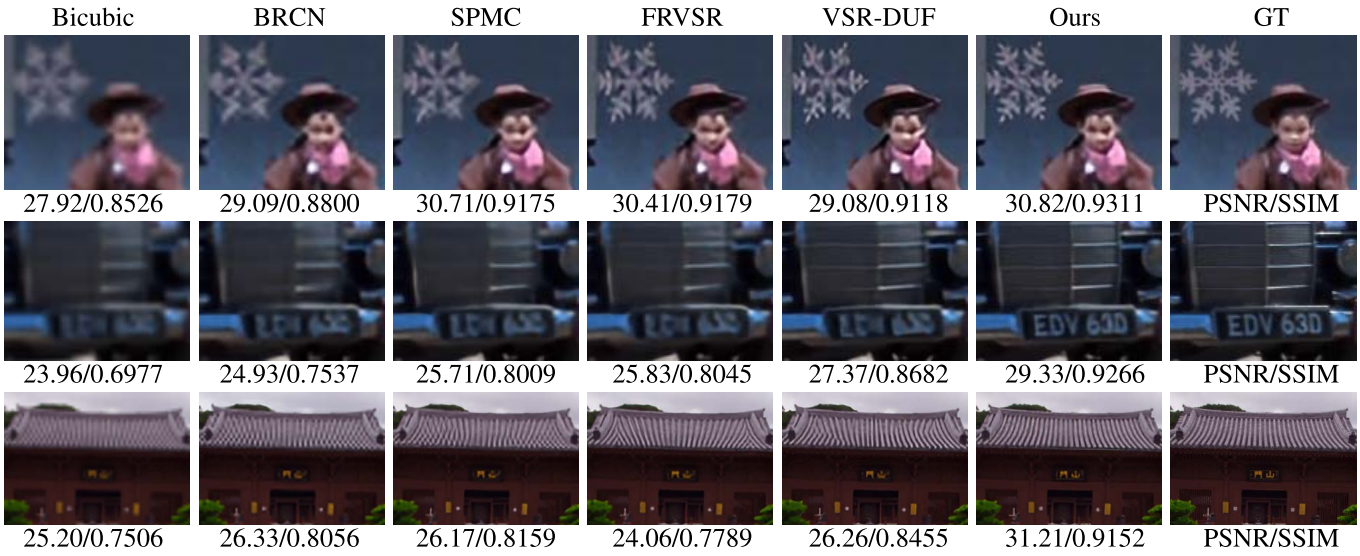


Fig. 7. Comparison of super-resolution algorithms in generic video super-resolution.

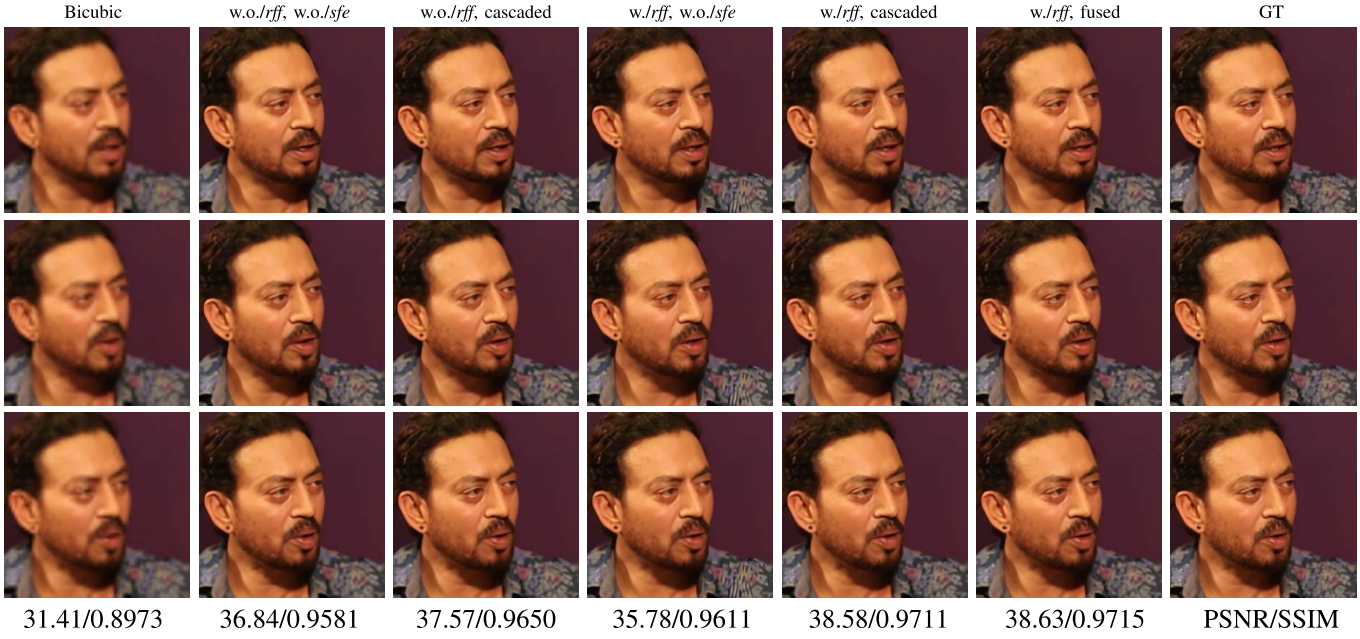


Fig. 8. Comparison of different variants of our method. 'w.o.' represents 'without' and 'w.' represents 'with'. 'cascaded' means cascaded bidirectional ConvLSTM is adopted for *sfe* strategy while 'fused' means fused bidirectional ConvLSTM is applied.

module (Section III-C) through using bicubicly upsampled LR images and super-resolved images of previous frames as inputs; 'CLSTM', ConvLSTM; 'BCLSTM', bi-directional ConvLSTM; '*rff*', recurrent frame fusion; '*sfe*', sequential feature encoding.

B. Evaluation Metrics

PSNR and SSIM are employed to evaluate the performance of video SR methods. PSNR is computed using the mean squared error of image sequence $\{\mathbf{Y}^t\}$ in comparison to $\{\mathbf{G}^t\}$,

$$\mathrm{PNSR} = \min\left(\log_{10}\frac{R}{\sqrt{\sum_{t=1}^{N}\|\mathbf{Y}^t-\mathbf{G}^t\|_2^2/(Ncr^2hw)}}, 100\right), \tag{11}$$

where R is the pixel range. R is set to 1 as all images are normalized to [0,1].

SSIM is widely used for evaluating perceived quality of digital images. In this paper it is calculated over individual RGB images of videos. Given two patches x, y from super-resolved and GT images respectively, the SSIM measure is calculated as follows,

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{12}$$

where $c_1 = (0.01R)^2$, $c_2 = (0.03R)^2$. μ_x, σ_x and σ_{xy} is the pixel average of x, standard deviation of x and covariance

TABLE II

PERFORMANCE OF FACIAL VIDEO HALLUCINATION ON THE TESTING SETS OF VOXCELEB AND RAVDESS. '#PARAMETERS' INDICATES NUMBER OF TRAINABLE PARAMETERS IN EACH SR MODEL. 'FPS' INDICATES NUMBER OF FRAMES PROCESSED BY EACH METHOD PER SECOND

	VoxCeleb			RAVDESS			#Parameters	FPS
	PSNR(F_t/F_p)	SSIM_{vh}(F_t/F_p)	SSIM_{vt}	PSNR(F_t/F_p)	SSIM_{vh}(F_t/F_p)	SSIM_{vt}		
Bicubic	29.56(201/0.0)	0.8776(249/0.0)	0.8957	26.88(317/0.0)	0.9036(200/0.0)	0.9100	-	-
GLN [25]	32.71(83.0/0.0)	0.9243(119/0.0)	0.9321	32.60(42.3/0.0)	0.9460(40.3/0.0)	0.9426	40,818,842	745.2
WaveletSRNet [28]	32.75(84.0/0.0)	0.9257(117/0.0)	0.9333	32.32(54.4/0.0)	0.9408(62.0/0.0)	0.9400	51,353,968	146.8
[2]	32.85(79.0/0.0)	0.9263(113/0.0)	0.9337	32.81(34.5/0.0)	0.9471(35.9/0.0)	0.9440	1,332,355	194.9
LapSRNet [44]	32.91(76.9/0.0)	0.9281(107/0.0)	0.9351	32.75(36.6/0.0)	0.9473(35.3/0.0)	0.9440	873,824	418.0
FSRNet [26]	32.94(78.5/0.0)	0.9303(101/0.0)	0.9369	32.70(38.9/0.0)	0.9488(29.1/0.0)	0.9456	9,074,899	75.8
SRResCNN [14]	33.00(75.0/0.0)	0.9275(111/0.0)	0.9345	32.61(42.6/0.0)	0.9489(28.3/0.0)	0.9459	1,549,335	136.4
SRResCNN-GAN [14]	30.82(152/0.0)	0.8868(220/0.0)	0.8983	30.73(118/0.0)	0.9348(78.8/0.0)	0.9327	-	-
SRResACNN [15]	33.35(61.6/0.0)	0.9319(94.6/0.0)	0.9383	33.23(18.7/0.0)	0.9508(20.7/0.0)	0.9475	1,557,688	115.2
SRResACNN-GAN [15]	32.23(97.9/0.0)	0.9167(139/0.0)	0.9243	32.18(56.3/0.0)	0.9437(46.0/0.0)	0.9400	-	-
ESRCNN [17]	33.74(49.1/0.0)	0.9375(73.0/0.0)	0.9433	33.52(8.23/0.0)	0.9527(13.0/0.0)	0.9493	16,697,987	17.7
ESRCNN-GAN [17]	32.43(89.9/0.0)	0.9191(130/0.0)	0.9262	32.27(50.9/0.0)	0.9444(43.5/0.0)	0.9407	-	-
BRCN [23]	31.63(121/0.0)	0.9091(166/0.0)	0.9200	30.50(134/0.0)	0.9329(92.8/0.0)	0.9309	90,828	295.6
VESPCN [8]	32.62(85.7/0.0)	0.9266(110/0.0)	0.9331	31.32(96.8/0.0)	0.9393(67.2/0.0)	0.9354	109,528	109.0
SPMC [3]	33.08(70.4/0.0)	0.9309(96.0/0.0)	0.9372	32.67(40.1/0.0)	0.9474(34.5/0.0)	0.9439	1,731,363	49.1
FRVSR [4]	34.33(29.8/0.0)	0.9458(40.5/0.0)	0.9493	33.26(17.7/0.0)	0.9515(18.1/0.0)	0.9475	5,281,509	106.7
VSR-DUF* [21]	33.63(52.0/0.0)	0.9455(41.4/0.0)	0.9483	32.14(61.5/0.0)	0.9493(27.6/0.0)	0.9443	5,821,952	9.3
LFFNet	33.76(48.2/0.0)	0.9398(64.3/0.0)	0.9435	32.74(37.9/0.0)	0.9484(31.4/0.0)	0.9436	3,382,405	105.5
ERFFNet	34.93(10.5/0.0)	0.9528(10.2/0.0)	0.9552	33.49(9.50/0.0)	0.9544(6.26/0.0)	0.9503	4,551,237	28.8
SECNet	**35.26(0.00/1.0)**	**0.9550(0.00/1.0)**	**0.9572**	**33.75(0.00/1.0)**	**0.9558(0.00/1.0)**	**0.9517**	5,334,792	28.5

between x and y respectively,

$$\mu_x = \sum_{i=-d}^{d}\sum_{j=-d}^{d} w_{ij}x_{ij}/w, \tag{13}$$

$$\sigma_x^2 = \sum_{i=-d}^{d}\sum_{j=-d}^{d} w_{ij}(x_{ij}-\mu_x)^2/w, \tag{14}$$

$$\sigma_{xy} = \sum_{i=-d}^{d}\sum_{j=-d}^{d} w_{ij}(x_{ij}-\mu_x)(y_{ij}-\mu_y)/w, \tag{15}$$

where x_{ij} is the pixel value at (i, j) in x. $w_{ij} = e^{\frac{-(i^2+j^2)}{2\rho^2}}$ and $w = \sum_{i=-d}^{i=d}\sum_{j=-d}^{j=d} w_{ij}$. d represents the radius of the patch. We use $\rho = 1.5$ and $d = 5$ here. The SSIM between two images can be obtained through averaging the values of (12) at all positions. For any image sequence, the average SSIM across all frames is denoted as SSIM_{vh}. To measure the quality of recovered temporal structures, we also slice a video along the horizontal axis and compute average SSIM of all images spanned by the vertical and temporal axes, denoted as SSIM_{vt}.

C. Quantitative and Qualitative Analysis

1) Comparisons Against State-of-the-Art Methods: Comparisons between our final model SECNet and other state-of-the-art methods are presented in Table II and III. In the facial video hallucination task (Table II), we compare our proposed method with several state-of-the-art SR methods including GLN [25], LapSRNet [44], [2], SRResCNN [14], SRResACNN [15], ESRCNN [17], BRCN [23], VESPCN [8], SPMC [3], FRVSR [4] and VSR-DUF [21]. All methods are trained using the same datasets and settings as described in Section IV-A, except for these marked with '$*$' which adopt results released by the authors or generated by provided models. To avoid defects nearby the image borders, input LR images of VSR-DUF are padded with 2 pixels. We conduct T-test between every contrast method and our proposed method (the last row of the table), to indicate improvement significance. The t-statistic F_t and p-value F_p are presented in the parentheses after PSNR and SSIM_{vh}. Our SECNet surpasses all previous methods. Practically it outperforms the second best method FRVSR by 2.7% higher PSNR and 1.0% larger SSIM_{vh} on VoxCeleb.

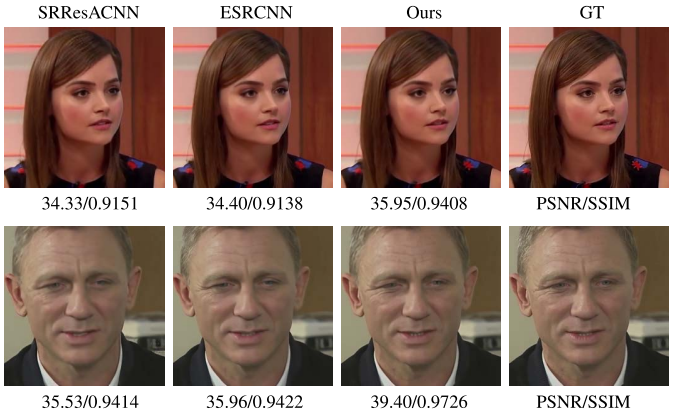


Fig. 9. Comparison with GAN-based methods. Both SRResACNN [15] and ESRCNN [17] are trained under the guidance of GAN as introduced in their original papers. (Best viewed in close-up).

Comparison of our method against other video super-resolution methods in Harmonic and VID4 datasets is reported in Table III. The most peripheral 8 pixels are excluded when computing PSNR and SSIM-s. The self-learned attention is not used in this task. Again our method achieves the best performance. The PNSR and SSIM_{vh} of our model SECNet are

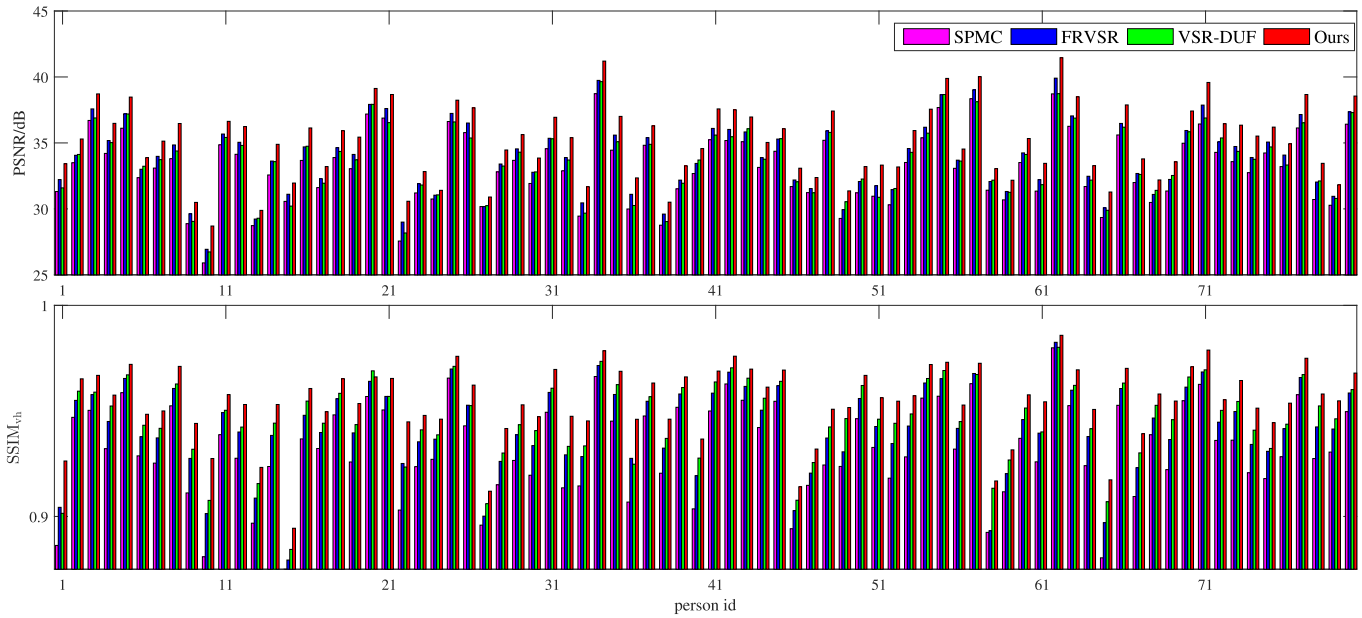


Fig. 10. Comparison of averaged PSNR and SSIM in facial videos from individual persons. All 80 persons in the testing set of VoxCeleb are considered. Our method performs consistently better than SPMC, FRVSR and VSR-DUF.

TABLE III

PERFORMANCE OF GENERIC VIDEO SUPER-RESOLUTION ON THE TESTING SETS OF HARMONIC AND VID4

	Harmonic			VID4		
	PSNR (F_t/F_p)	SSIM$_{\rm vh}$ (F_t/F_p)	SSIM$_{\rm vt}$	PSNR (F_t/F_p)	SSIM$_{\rm vh}$ (F_t/F_p)	SSIM$_{\rm vt}$
Bicubic	29.02(71.5/0.00)	0.8075(70.4/0.00)	0.8245	22.18(12.4/0.00)	0.6125(21.9/0.00)	0.6860
BRCN [23]	30.45(46.9/0.00)	0.8415(49.6/0.00)	0.8511	22.86(9.37/0.00)	0.6730(17.5/0.00)	0.7337
VSRNet* [7]	-	-	-	23.26(6.96/0.00)	0.6809(16.5/0.00)	0.7446
VESPCN* [8]	-	-	-	23.66(5.20/0.00)	0.7038(14.5/0.00)	0.7612
VESPCN [8]	31.04(36.9/0.00)	0.8553(39.3/0.00)	0.8627	23.44(6.77/0.00)	0.7143(14.3/0.00)	0.7668
SPMC* [3]	-	-	-	24.51(1.78/0.08)	0.7583(8.94/0.00)	0.8025
SPMC [3]	32.13(19.9/0.00)	0.8787(20.9/0.00)	0.8821	24.18(3.57/0.00)	0.7670(7.93/0.00)	0.8069
VSR-LTD* [22]	-	-	-	24.01(3.83/0.00)	0.7323(11.6/0.00)	0.7865
FRVSR [4]	32.24(18.0/0.00)	0.8818(18.1/0.00)	0.8858	24.21(3.49/0.00)	0.7742(6.74/0.00)	0.8141
VSR-DUF* [21]	32.52(13.7/0.00)	0.8901(11.0/0.00)	0.8916	24.23(4.00/0.00)	0.8017(3.04/0.00)	0.8328
LFFNet	31.82(24.7/0.00)	0.8724(26.1/0.00)	0.8766	23.96(4.64/0.00)	0.7552(9.79/0.00)	0.7987
SECNet	**33.42(0.00/1.00)**	**0.9026(0.00/1.00)**	**0.9045**	**25.02(0.00/1.00)**	**0.8179(0.00/1.00)**	**0.8487**

respectively 0.90 and 0.0125 larger than those of the second best method VSR-DUF in Harmonic dataset.

To discuss the efficacy brought by LFFNet, we also transform the enhanced recurrent frame fusion module into an independent SR model called ERFFNet. Apparently it is inferior to SECNet as reported in Table II.

A qualitative comparison of facial video hallucination between our method and other SR methods are shown in Fig. 6. The super-resolved results from our method tend to be more appealing and clearer than those from other methods especially in the eye regions. The super-resolving quality in generic single-shot video datasets is shown in Fig. 7. Our method recovers the buildings (top image), digits (middle image) and tiles (bottom image) more accurately. Comparison with GAN-based methods [15] and [17] is presented in Fig. 9.

2) Performance Across Persons: Performance comparison in facial videos from independent persons is presented in Fig. 10. It indicates our method consistently performs better than SPMC, FRVSR and VSR-DUF across characters.

3) Performance Across Frames: We also report averaged PSNR for $t \in [1, 70]$ in Fig. 11, where the significance and efficacy of our self-enhanced convolutional network can be clearly observed. The PSNR-s of FRVSR and our proposed models rise rapidly during the initial frames because both reuses the estimated results of preceding frames recurrently. However the rising period of SECNet is longer than FRVSR. Overall, our method achieves the highest performance among all considered state-of-the-art SR methods.

4) Discussions of Temporal Fusion Strategies: The facial video super-resolution performances of our final models using one-way, cascaded and fused BCLSTM-s are presented in Table IV. Self-attention is not used in all of our models in this subsection. To study the effectiveness of recurrent frame fusion (abbr. *rff*) and sequential feature encoding (abbr. *sfe*), we trained models without using recurrent frame fusion which means that T_2 is set to 0, or not using past features to enhance the feature representation of current frame. Compared to the model not using *rff* or *sfe*, adopting any of *rff* and *sfe* brings significant improvement. For example the adoption of *sfe* (equipped with cascaded BCLSTM) and *rff* gives rise to results with 0.39dB and 0.62dB higher PSNR-s respectively than the version in which neither is utilized. Turning off

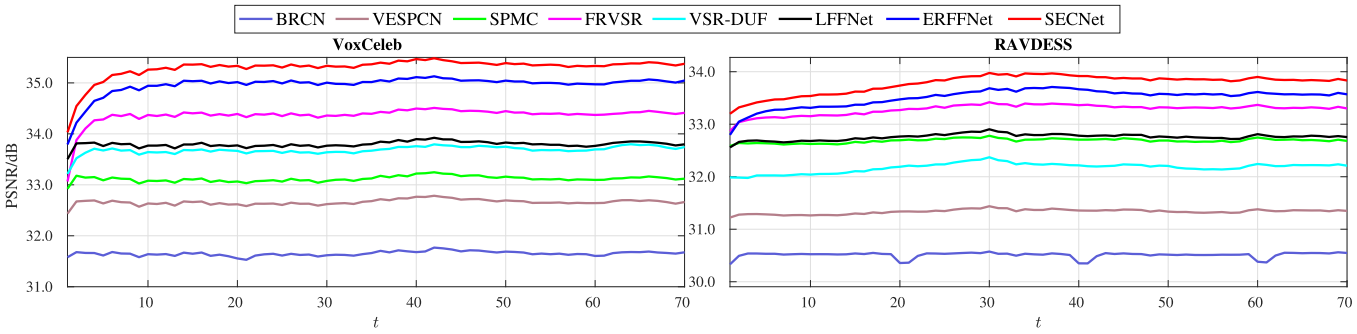


Fig. 11. Comparison of averaged PSNR at different frames. The PSNR of our method rises quickly as it runs forward from the beginning. Afterwards it steadily maintains at a higher PSNR rate than other methods.

TABLE IV
COMPARISONS OF DIFFERENT FUSION STRATEGIES ON VOXCELEB

rff	*sfe*	PSNR(F_t)	SSIM_{vh}(F_t)
×	×	34.16(32.5)	0.9437(47.6)
✓	×	34.78(12.2)	0.9514(15.2)
×	one-way CLSTM	34.42(24.0)	0.9470(34.3)
×	cascaded BCLSTM	34.55(20.0)	0.9485(27.6)
✓	non-local attention $T_3 = 2$	34.95(6.94)	0.9527(9.47)
✓	flow-guided attention $T_3 = 2$	34.97(6.57)	0.9527(9.49)
✓	fused BCLSTM $T_3 = 2$	35.14(1.22)	0.9542(2.54)
✓	recurrent unit in [23]	34.91(8.39)	0.9523(11.2)
✓	one-way CLSTM	34.98(6.26)	0.9531(7.33)
✓	cascaded BCLSTM $T_3 = 6$	35.08(3.06)	0.9536(5.36)
✓	fused BCLSTM $T_3 = 6$	**35.17(0.00)**	**0.9547(0.00)**

rff or *sfe* causes dramatically drop to all metrics. For example abandoning *rff* leads to decrease of 0.53dB for PSNR, in the framework using cascaded BCLSTM as *sfe* strategy. In conclusion, any of recurrent frame fusion and sequential feature encoding can benefit facial video hallucination independently. Adopting both of them leads to better results as they are able to complement each other. Besides, the bidirectional *sfe* strategies outperform one-way strategies. Qualitative comparison of our method using different temporal fusion strategies is presented in Fig. 8.

Two alternative temporal fusion strategies are tried to replace the BCLSTM based recurrence module. The convolutional non-local operation [16] can be applied to exploit temporal dependencies. The feature aggregation method in [45] can also be applied to fuse temporal features based on attention maps which are calculated between the feature of the reference frame and features of previous frames. Optical flow fields are used to align features of previous frames to the reference frame. The comparison is enclosed in Table IV. Considering the computation load of the non-local operation, T_3 is set to 2. The fused BCLSTM performs better than the above two temporal fusion methods. Additionally, we can replace the CLSTM with the recurrent unit in [23], forming a variant of our method which produces results with 0.26dB lower PSNR.

5) Discussions of Attention Strategies: We discuss the performance of using pairwise attention calculated with the non-local operation [16], channel-wise and spatial attentions computed by squeeze-and-excitation (SE) [42]. The non-local operation is integrated into the 4-th and 8-th residual blocks. The quantitative comparisons on VoxCeleb dataset are presented in Table V. Using squeeze-and-excitation based attention achieves better results than using non-local pairwise attention. The spatial attention slightly benefits the super-resolved results while incorporation of additional channel-wise attention fails to bring further improvement.

TABLE V
COMPARISONS OF DIFFERENT ATTENTION STRATEGIES ON VOXCELEB

attention strategy	PSNR(F_t/F_p)	SSIM_{vh}(F_t/F_p)
without	35.17(1.69/0.09)	0.9547(1.68/0.09)
non-local operation	35.15(2.56/0.01)	0.9543(3.72/0.00)
spatial SE	**35.26(-1.16/0.24)**	0.9550(0.54/0.59)
channel-wise & spatial SE	35.22(0.00/1.00)	**0.9551(0.00/1.00)**

TABLE VI
PERFORMANCES OF USING DIFFERENT T_2 AND T_3 IN THE VALIDATION SET OF VOXCELEB

T_2	T_3	PSNR	SSIM_{vh}	SSIM_{vt}
0	6	35.43	0.9596	0.9595
1	6	35.78	0.9645	0.9641
3	6	35.93	0.9643	0.9638
2	0	35.82	0.9638	0.9635
2	2	36.01	0.9647	0.9643
2	4	36.03	0.9647	0.9644
2	6	**36.04**	**0.9649**	**0.9644**

6) Choices for T_2 and T_3: The results of choosing different T_2 and T_3 are reported in Table VI. All experimental results are obtained from testing in the validation set of VoxCeleb. The cascaded BCLSTM is adopted to implement *sfe*. Using 2 previous frames produces better results than using 1 previous frame, but more frames do not help improving super-resolution performance which might be caused by the increased difficulty in learning the dependency between current and previous frames. Increasing T_3 from 0 to 2 brings gain of 0.19dB in PSNR. Adopting different values 2, 4 and 6 for T_3 leads to almost equivalent performance.

7) Choices for Number of Residual Blocks: We present the performances of using various numbers of residual blocks in Table VII. Using 8 residual blocks achieves the best performance, while 16 blocks can not bring better results.

TABLE VII
COMPARISONS OF VARIANTS USING DIFFERENT NUMBERS OF RESIDUAL BLOCKS ON VOXCELEB

#residual blocks	PSNR	SSIM_{vh}	SSIM_{vt}
2	35.06	0.9538	0.9562
4	35.18	0.9544	0.9568
8	**35.26**	**0.9550**	**0.9572**
16	35.24	0.9548	0.9571

V. CONCLUSION

To solve the facial video hallucination problem, we have proposed a self-enhanced convolutional network, which utilizes recurrent frame fusion and sequential feature encoding based on ConvLSTM to take advantage of both spatial and temporal information from past video frames. Furthermore a local frame fusion network is utilized to involve in information from future frames. Our method achieves state-of-the-art performance in both facial video hallucination and more generic single-shot video SR tasks. In the future, it deserves in-depth research to exploit deliberately devised attentions in facial video hallucination, based on facial priors, motion units, expressions, etc.

REFERENCES

[1] Y. Shi, G. Li, Q. Cao, K. Wang, and L. Lin, "Face hallucination by attentive sequence optimization with reinforcement learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[2] A. Bulat and G. Tzimiropoulos, "Super-FAN: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with GANs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 109–117.

[3] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4472–4480.

[4] M. S. M. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6626–6634.

[5] C. Liu and D. Sun, "On Bayesian adaptive video super resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 346–360, Feb. 2014.

[6] Z. Ma, R. Liao, X. Tao, L. Xu, J. Jia, and E. Wu, "Handling motion blur in multi-frame super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5224–5232.

[7] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Trans. Comput. Imag.*, vol. 2, no. 2, pp. 109–122, Jun. 2016.

[8] J. Caballero *et al.*, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4778–4787.

[9] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[10] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," 2017, *arXiv:1706.08612*. [Online]. Available: https://arxiv.org/abs/1706.08612

[11] S. R. Livingstone, K. Peck, and F. A. Russo, "Ravdess: The ryerson audio-visual database of emotional speech and song," in *Proc. Annu. Meeting Can. Soc. Brain, Behaviour Cognit. Sci.*, 2012, pp. 205–211.

[12] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3791–3799.

[13] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.

[14] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4681–4690.

[15] H. N. Pathak, X. Li, S. Minaee, and B. Cowan, "Efficient super resolution for large-scale images using attentional GAN," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 1777–1786.

[16] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[17] X. Wang *et al.*, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, Sep. 2018, pp. 63–79.

[18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[19] A. Jolicoeur-Martineau, "The relativistic discriminator: A key element missing from standard GAN," 2018, *arXiv:1807.00734*. [Online]. Available: https://arxiv.org/abs/1807.00734

[20] O. Makansi, E. Ilg, and T. Brox, "End-to-end learning of video super-resolution with motion compensation," in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2017, pp. 203–214.

[21] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3224–3232.

[22] D. Liu *et al.*, "Robust video super-resolution with learned temporal dynamics," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2507–2515.

[23] Y. Huang, W. Wang, and L. Wang, "Bidirectional recurrent convolutional networks for multi-frame super-resolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 235–243.

[24] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin, "Learning face hallucination in the wild," in *Proc. 29th AAAI Conf. Artif. Intell.*, Mar. 2015, pp. 3871–3877.

[25] O. Tuzel, Y. Taguchi, and J. R. Hershey, "Global-local face upsampling network," 2016, *arXiv:1603.07235*. [Online]. Available: https://arxiv.org/abs/1603.07235

[26] Y. Chen, Y. Tai, X. Liu, C. Shen, and J. Yang, "FSRNet: End-to-end learning face super-resolution with facial priors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2492–2501.

[27] Y. Song, J. Zhang, S. He, L. Bao, and Q. Yang, "Learning to hallucinate face images via component generation and enhancement," 2017, *arXiv:1708.00223*. [Online]. Available: https://arxiv.org/abs/1708.00223

[28] H. Huang, R. He, Z. Sun, and T. Tan, "Wavelet-SRNet: A wavelet-based CNN for multi-scale face super resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2017, pp. 1689–1697.

[29] A. Van Den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," 2016, *arXiv:1601.06759*. [Online]. Available: https://arxiv.org/abs/1601.06759

[30] R. Dahl, M. Norouzi, and J. Shlens, "Pixel recursive super resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5439–5448.

[31] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[32] X. Yu and F. Porikli, "Ultra-resolving face images by discriminative generative networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 318–333.

[33] Z. Chen and Y. Tong, "Face super-resolution through Wasserstein GANs," 2017, *arXiv:1705.02438*. [Online]. Available: https://arxiv.org/abs/1705.02438

[34] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*. [Online]. Available: https://arxiv.org/abs/1701.07875

[35] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5769–5779.

[36] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang, "Learning to super-resolve blurry face and text images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2017, pp. 251–260.

[37] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2472–2481.

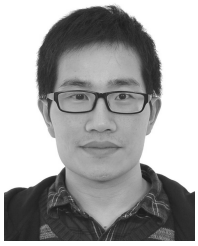
[38] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang, "Deep video deblurring for hand-held cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 237–246.

[39] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[41] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[44] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 624–632.
[45] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei, "Flow-guided feature aggregation for video object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 408–417.

Chaowei Fang received the B.E. degree from Xi'an Jiaotong University in 2013 and the Ph.D. degree from The University of Hong Kong in 2019. He is currently with the School of Artificial Intelligence, Xidian University. His research interests include image processing, medical image analysis, computer vision, and machine learning.



Guanbin Li (M'15) received the Ph.D. degree from The University of Hong Kong in 2016. He is currently an Associate Professor with the School of Data and Computer Science, Sun Yat-sen University. He has authored or coauthored more than 20 articles in top-tier academic journals and conferences. His current research interests include computer vision, image processing, and deep learning. He has been serving as a reviewer for numerous academic journals and conferences, such as TPAMI, TIP, TMM, TC, CVPR, AAAI, and IJCAI. He serves as the Area Chair for the conference of VISAPP.

Xiaoguang Han received the B.Sc. degree in mathematics from NUAA in 2009, the M.Sc. degree in applied mathematics from Zhejiang University in 2011, and the Ph.D. degree from HKU in 2017. He is currently a Research Assistant Professor with the Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong. His research mainly focuses on computer vision, computer graphics, and 3D deep learning.

Yizhou Yu (M'10–SM'12–F'19) received the Ph.D. degree from the University of California at Berkeley in 2000. He is currently a Professor with The University of Hong Kong and a Faculty Member with the University of Illinois at Urbana–Champaign for twelve years. His current research interests include computer vision, deep learning, biomedical data analysis, computational visual media, and geometric computing. He was a recipient of the 2002 US National Science Foundation CAREER Award, the 2007 NNSF China Overseas Distinguished Young Investigator Award, and the ACCV 2018 Best Application Paper Award. He has served on the Editorial Board of *IET Computer Vision*, *The Visual Computer*, and the IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS. He has also served on the program committee of many leading international conferences, including SIGGRAPH, SIGGRAPH Asia, and International Conference on Computer Vision.