High-Frequency Refinement for Sharper Video Super-Resolution

Vikram Singh Akshay Sharma* Sudharshann Devanathan* Anurag Mittal
Computer Vision Lab, Indian Institute of Technology-Madras
{vsingh,amittal}@cse.iitm.ac.in, akshaysh@andrew.cmu.edu, sd3770@nyu.edu

Abstract

*A video super-resolution technique is expected to generate a 'sharp' upsampled video. The sharpness in the generated video comes from the precise prediction of the high-frequency details (*e.g. *object edges). Thus high-frequency prediction becomes a vital sub-problem of the super-resolution task. To generate a sharp-upsampled video, this paper proposes an upsampling network architecture 'HFR-Net' that works on the principle of 'explicit refinement and fusion of high-frequency details'. To implement this principle and to train HFR-Net, a novel technique named 2-phase progressive-retrogressive training is being proposed. Additionally, a method called dual motion warping is also being introduced to preprocess the videos that have varying motion intensities (slow and fast). Results on multiple video datasets demonstrate the improved performance of our approach over the current state-of-the-art.*

1. Introduction

Video super-resolution refers to the task of increasing the resolution (upsampling) of a video while maintaining its quality that is typically measured in terms of Peak Signal-to-Noise Ratio (PSNR [9]), and Structural Similarity (SSIM [28]) w.r.t. the ground-truth. Recent advances in the display device technologies (Full HD/4K/8K) and an abundance of low-resolution videos have brought a surge in the application of video super-resolution. Consequently, it has become a prominent Computer Vision problem of immense academic and commercial interests.

Current state-of-the-art methods for video super-resolution (as discussed in Section 1.1) lack in the generation/prediction of an upsampled video that is 'sharp'. Bringing in the 'sharpness' in an upsampled video primarily requires the prediction of high-frequency/fine details that are found in the regions of the high spatial gradient, *e.g.* object edges. In this work, we propose a high-frequency refinement based video super-resolution network 'HFR-Net' that upsamples a video frame and then takes explicit measure to increase its sharpness. For increasing the sharpness, the network divides the upsampled frame into two frames that have high and low-frequency details separately. The network then refines the frame with the high-frequency details and finally, fuse the refined high-frequency frame with the low-frequency frame to generate the sharper version of the upsampled frame. To the best of our knowledge, none of the existing approach works on this principle, and we later show in Section 3 that the proposed principle is indeed effective and that our network (HFR-Net) that is designed on this principle generates better predictions than the current state-of-the-art.

The sub-network of HFR-Net that performs the fusion of refined high-frequency and low-frequency frames is trained using a novel method that we call *2-phase progressive-retrogressive* training. Unlike conventional training that typically utilises the ground-truth only to compute the training loss, the proposed 2-phase method also requires the ground-truth to be one of the training inputs. Here, we emphasise that the proposed network HFR-Net requires ground-truth as an input only during the training and not during inference. We further elaborate on the 2-phase training method in Section 2.5.

Motion-based direct frame warping has been used in the prior works of *Liu et al*. [15] and *Tao et al*. [20] to preprocess the video before sending it to the super-resolution network. However, such warping might not always be useful, especially when the video has varying motion intensities (fast and slow). To preprocess such a video, we propose a method that we call *dual motion warping*. Later in Section 3.2, we show that the proposed preprocessing method gives better performance as compared to direct frame warping used earlier. We further elaborate upon this method in Section 2.1.

1.1. Related work

In this section, we discuss a few state-of-the-art techniques for video super-resolution. *Caballero et al*. [2] proposed a sub-pixel Convolutional network to exploit tem-

*Akshay Sharma from Carnegie Mellon University and Sudharshann Devanathan from New York University were interns at Computer Vision Lab, IIT-Madras when this work was performed.

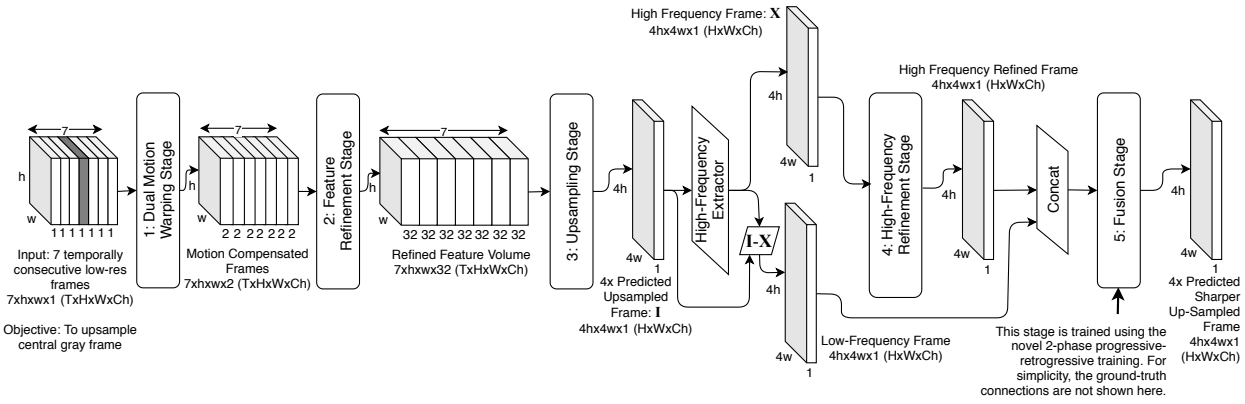


Figure 1: Illustration of the proposed five-stage network 'HFR-Net' (High-Frequency Refinement Network) for video super-resolution.

poral redundancies that improved the upsampling quality while maintaining a real-time speed. *Tao et al*. [20] performed frame alignment using direct motion warping that further improved the results. *Liu et al*. [15] identified the importance of complex motion estimation and estimated it using a spatial frame alignment network with temporal filters at different scales. HFR-Net has superior motion compensation in the form of dual motion warping and 3D Convolutions at different scales to capture Spatio-temporal information.

Prior works have leveraged recurrent networks to model temporal or sequential data, *Huang et al*. [8] used RCNN to model temporal dependency between video frames at patch level rather than frame level, while *Yang et al*. [26], *Wang et al*. [23] and *Yi et al*. [27] used RCNN to build residual memory blocks that modelled the inter and intra-frame relationships. Optical flow is a powerful tool to estimate and compensate motion between the frames. *Makansi et al*. [17] estimated optical flow within their network to perform a direct warp from low to high-resolution frame. HFR-Net also has an RCNN, as it has been shown to work effectively for upsampling in the cited works. HFR-Net performs the proposed *dual warping* instead of the direct warping as it was empirically found that dual warping gives better results than direct warping.

Compensating the motion for every neighbouring frame is a computationally intensive task; thus, *Sajjadi et al*. [18] proposed to upsample the current frame, using the previous upsampled frame and current low-resolution frame. On the other side, some techniques attempted to circumvent motion compensation altogether, for instance, *Jo et al*. [10] avoided it by generating dynamic upsampling filters and residual frames based on the local spatial and temporal neighbourhood of each pixel. *Kim et al*. [12] opted for 3D-ConvNet to do away with motion compensation. *Brifman et al*. [1] implemented a denoising approach, free from motion-estimation. Generative adversarial networks have also been used in video super-resolution to predicted realistic upsampled frames, for instance, *Lucas et al*. [16] used a GAN with distance loss in feature and pixel space as the regulariser. HFR-Net compensates the motion to generate better upsampling results. It not only takes three frames from the past but also takes three frames from the future for compensating the motion before upsampling. HFR-Net does have 3D Convolutions but is free from GAN; instead, it focuses on the refinement of high-frequency details to generate realistic-looking sharp upsampled frames.

Wang et al. [22] proposed a network with residual blocks and long skip connections to predict high-frequency details required for super-resolution. *Zhu et al*. [29] designed a multi-component network to generate spatial and temporal features separately and fuse them to predict the upsampled frame. *Yan et al*. [25] proposed a frame and feature context network to avoid flickering, jitter and jagged artifacts. *Haris et al*. [5] proposed a model that considered each frame as a separate source of information and iteratively combined them to generate the upsampled frame. *Li et al*. [13] focused on reducing computational complexity while maintaining state-of-the-art accuracy.

To the best of our knowledge, none of the existing video super-resolution approaches has used or proposed any idea that is being claimed as a contribution in this work and those contributions are:

- A network designed for upsampling and sharpening a video (HFR-Net, illustrated in Figure 1) that works on the principle of 'explicit refinement and fusion of high-frequency details'.
- A method named 2-phase (*progressive-retrogressive*) training to train the network to perform the fusion of high and low-frequency frames.
- A technique named *dual motion warping* to preprocess videos with varying motion intensities.

2. HFR-Net: High-frequency refinement network

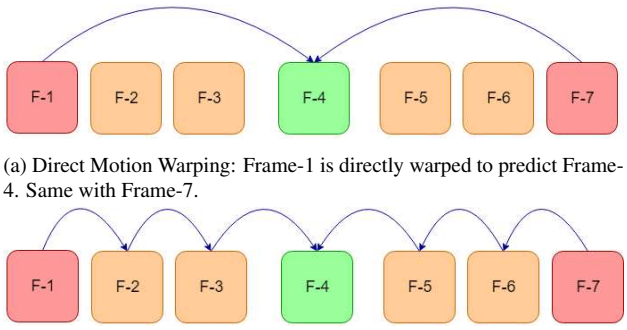
This work proposes a five-stage high-frequency refinement network ('HFR-Net') for video super-resolution, as shown in Figure 1. For upsampling a video, the network takes as input a set of seven temporally consecutive frames and upsamples the target/central (4^{th}) frame. Because the sensitivity to Luminance change is high in human beings, similar to prior works by *Liu et al*. [15] *Kappeler et al*. [11], and *Shi et al*. [19], HFR-Net is also set up to accept and to predict/upsample ($4\times$) only the Luminance (Y) channel (in YCbCr colour-space) of the frame. The other channels are upsampled using the bi-cubic interpolation.

As illustrated in Figure 1, the seven input frames undergo the proposed *dual motion warping* in the first stage. The second stage extracts and refines the essential features to generate a volume that is required to upsample the target frame. The third stage consumes the feature volume to generate a single intermediate upsampled version of the target frame. In the fourth stage, the upsampled frame is separated into two separate frames containing the high-frequency and low-frequency details, respectively. In the fourth stage itself, the extracted high-frequency frame is refined, and finally, in the last/fifth stage the refined high-frequency frame is fused with the low-frequency frame to generate the final network prediction, *i.e*. a $4\times$ sharper upsampled frame. The proposed ideas of *dual motion warping* and 2-phase *progressive-retrogressive* training are applied to the first and last stages of the network, hence they are elaborated along with the details of their respective stage. We now describe each stage in detail.

2.1. Stage-1: Dual motion warping

Video super-resolution techniques improve the quality of an upsampled frame by making use of multiple low-resolution frames that are adjacent to the target frame being upsampled. Prior methods by *Liu et al*. [15] and *Tao et al*. [20] have shown that a direct optical flow-based warp (explained in Figure 2a) of the neighbouring source frames to predict the target frame before upsampling, improves the network's upsampling performance. However, we argue that such a direct warping might not work well if the motion in the video is very fast, as fast-motion videos will have a significant visual disparity between frames, resulting in weak warping predictions.

For such fast motion videos, we propose to perform 'sequential warping' as shown and explained in Figure 2b to predict the target frame. Nonetheless, this warping, too, can be erroneous if the motion in the video is very slow. A single frame warp operation contains interpolation and optical flow computation errors that accumulate with each sequential warp, and thus sequential warping does not perform well in videos with slow motion. Succinctly speaking, a warping technique designed to work on fast motion videos might not work well on slow-motion videos and the same is true the other way around as well.



(a) Direct Motion Warping: Frame-1 is directly warped to predict Frame-4. Same with Frame-7.

(b) Sequential Motion Warping: Frame-1 (I) is warped to predict Frame-2 (I+1). The prediction is re-warped to predict Frame-3 (I+2) and so on. Similar with Frame-7.

Figure 2: Dual Motion Warping: F1-F7 are temporally consecutive frames given as input to HFR-Net to upsample the target Green frame. Red frames indicate the source frames to be warped to predict the Green frames. Orange frames are those adjacent frames that will be warped but whose warp is not illustrated in the figure for simplicity. The optical flow used for the warps is given by *Weinzaepfel et al*. [24].

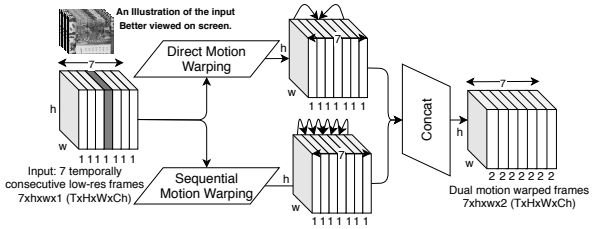


Figure 3: Stage-1: *Dual motion warping*. This stage takes in the seven temporally consecutive low-resolution frames and performs the dual motion warping, as explained in Section 2.1. Notation- T: Time/Temporal, H: Height, W: Width, Ch: Channel.

To allow for both the scenarios, we propose the *dual motion warping* where both sequential and direct warpings are performed, and the results are concatenated, as shown in Figure 3 before further processing.

2.2. Stage-2: Feature refinement

This stage generates the features required for upsampling the target frame by consuming the *dual motion warped* frames. The architecture of this stage is depicted in Figure 4. This stage has a $4\times$ bicubic upsampling operation to makes the rest of the network learn to improve upon the bicubic prediction. The stage also has a space-to-depth [21] operator that rearranges the pixels of its input, thereby increasing the number of channels by reducing the spatial

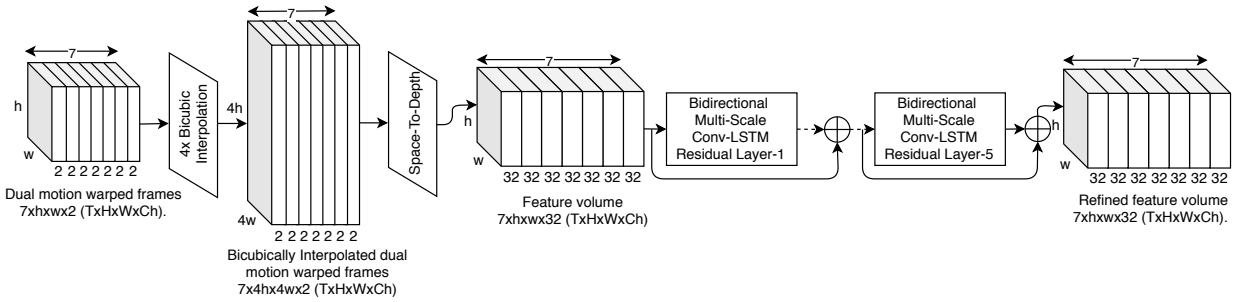


Figure 4: Stage-2: Feature refinement. This stage generates the features required for upsampling the target frame using the illustrated operators and network that are explained in Section 2.2.

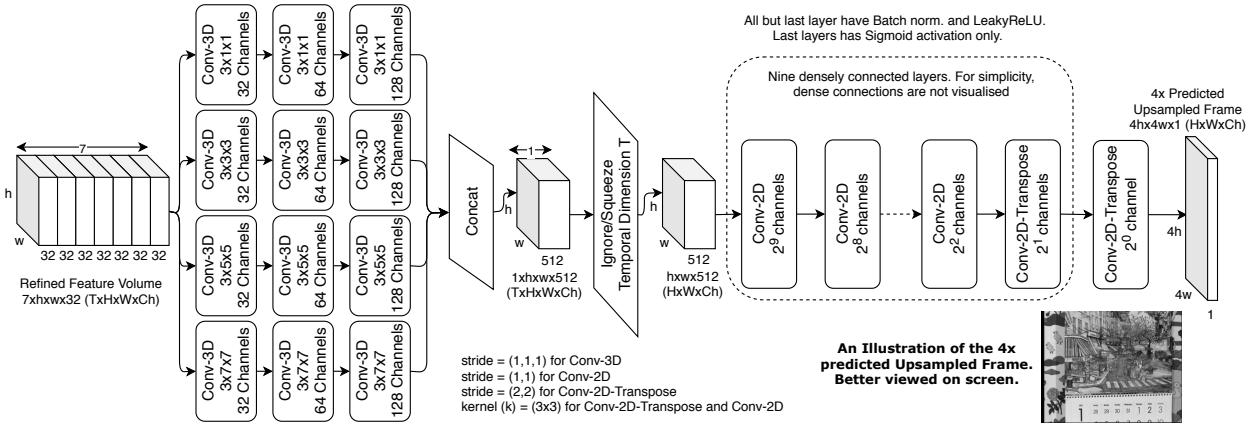


Figure 5: Stage-3: Upsampling. The task of this stage is to generate the intermediate upsampled version of the target frame, as explained in Section 2.3.

(height and width) size of the input. This helps in delaying the upsampling to the later stages and allows the current stage to focus only on the feature refinement. Lastly, this stage has bidirectional multi-scale convolutional LSTM (proposed by *Guo et al*. [4], initialised by *He et al*. [6]) to improve the quality of features with the available temporal-information.

To generate the stage ground-truth, we take the Y (Luminance) channel of the model ground-truth (size 4h×4w×1 scaled in range 0-1) and concatenate it with its replica on the channel dimension to get a frame of size 4h×4w×2. Seven replicas of this frame are concatenated on the temporal dimension to get a new frame of size 7×4h×4w×2. Finally, we apply space-to-depth [21] on it, to obtain the stage ground-truth of size 7×h×w×32. The stage loss is the mean-squared-error between the stage prediction and stage ground-truth.

2.3. Stage-3: Upsampling

This stage performs the actual upsampling of the intended frame, given the refined feature volume generated at Stage-2. The architecture of this stage is shown in Figure 5. This stage has operators like Conv-3D, Conv-2D with dense connections. Their use is apparent, but succinctly, Conv-3D consumes features of temporal (T) dimension to enrich features in channel (Ch) dimension. Conv-2D further improves the channel features, Conv-2D-Transpose upsamples and generates the frame as required, and dense connections (proposed by *Huang et al*. [7]) help in training the network efficiently.

The stage ground-truth is the Y (Luminance) channel of the model ground-truth scaled in the range 0-1. The stage loss is the sum of mean-squared-error(stage prediction, stage ground-truth) and (1 - SSIM(stage prediction, stage ground truth)). Minimisation of mean-square-error maximises the PSNR while minimisation of (1-SSIM) (*i.e*. the structural dissimilarity) maximises the SSIM.

2.4. Stage-4: High frequency refinement

This stage extracts and refines the high-frequency details of the intermediate upsampled frame generated by Stage-3, as shown in Figure 6. To extract/separate the high-frequency frame from the given frame (scaled in range 0-1), Sobel filters X and Y are applied on it to generate $\mathbb{D}_x$,

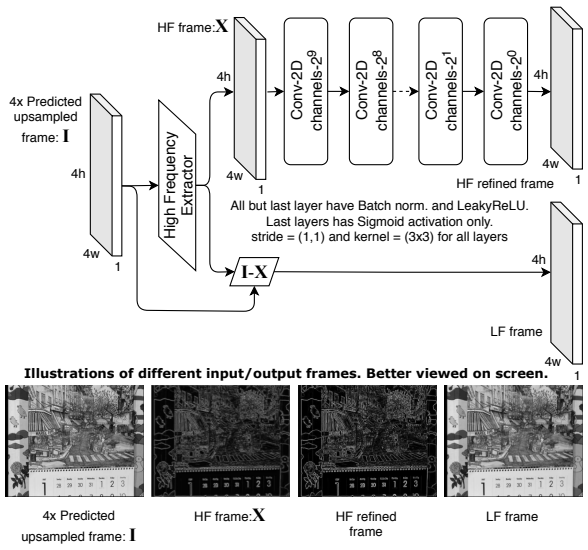


Figure 6: Stage-4: High-frequency refinement. This stage separates the intermediate upsampled frame into two frames with high and low-frequency details separately and explicitly refines the high-frequency frame, as explained in Section 2.4.

$\mathbb{D}_y$. that are the approximations of the derivatives for the horizontal and vertical changes.

$$X = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}, \quad Y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

$\mathbb{D}_x$, $\mathbb{D}_y$ are used to compute the high-frequency frame, using Eq.: 1.

$$\text{high-frequency frame} = \mathbb{S}\left(\sqrt{\mathbb{D}_x^2 + \mathbb{D}_y^2}\right) \qquad (1)$$
$$\mathbb{S} \text{ scales the values in range } [0, 1]$$

Low-frequency frame is obtained by subtracting the high-frequency frame from the given image/frame. The extracted high-frequency frame is processed through a Convolutional network that refines its features. The ground-truth for this stage is the high-frequency frame extracted from the Y (Luminance) channel of the model ground-truth, and the stage loss is the mean-squared-error between the stage prediction and the stage ground-truth. Both the refined high-frequency frame, along with the low-frequency frame goes into the next stage for fusion.

2.5. Stage-5: Frame fusion

The last stage of HFR-Net fuses the low and refined high-frequency frames coming from Stage-4 to predict the final sharpened upsampled frame, as shown in Figure 7. This stage is trained using the proposed 2-phase (*progressive-retrogressive*) training that is described next.

2.5.1 2-Phase *progressive-retrogressive* training

Conventionally, a neural network is trained progressively, implying that its training begins with a randomly initialised state and ends in a good/trained state. Such training (in the supervised paradigm) requires the ground-truth, only to compute the training loss. In contrast, the proposed 2-phase *progressive-retrogressive* training requires the ground-truth not only to compute the loss but also as one of the inputs to the network.

Specifically, this training has two phases: **1) The progressive phase**- In this phase, the network is trained with the ground-truth as one of its inputs. Due to the presence of ground-truth in the input, network attains a very good state and predicts high-quality results by the end of progressive phase. However, during inference, the network does not have access to the ground-truth; still, it has to perform well. To account for this performance, we retrain the network retrogressively.

2) The retrogressive phase- This phase starts when the performance metric of the network in the progressive phase saturates. While retraining, the ground-truth input is gradually replaced with a substitute that is generated by any previous stage of the network. This substitute must be an approximation to the ground-truth and must be generated using those inputs that are available during the inference also. By the end of the retrogressive phase, the ground-truth at the input of the network is completely replaced with the substitute. The purpose of the retrogressive training phase is to remove the network's dependence on the ground-truth input and also to make it learn to predict the same high-quality results as it was predicting after the progressive training phase but without the ground-truth input.

As can be thought of, the quality of the prediction deteriorates as the ground-truth starts to get replaced from the network input in the retrogressive phase. However, we show in Section 3.3 that the 2-phase training, sets the network in a much better state/local-minima as compared to conventional 1-phase training and that a 2-phase trained network generated better results than a conventionally trained network. Next, we explain how the proposed 2-phase training is performed in the fusion stage of HFR-Net.

2.5.2 2-Phase training for fusion in stage-5

Stage-5 of HFR-Net fuses the low and the refined high-frequency frames coming from Stage-4, to predict the final model prediction that is also the sharper version of the intermediate upsampled frame generated at Stage-3. A conventionally (1-phase) trained network might perform this fusion by concatenating the low and high-frequency frame and then processing it with a Convolutional network. HFR-Net does the same, but in 2-phases, as illustrated in Figure 7.

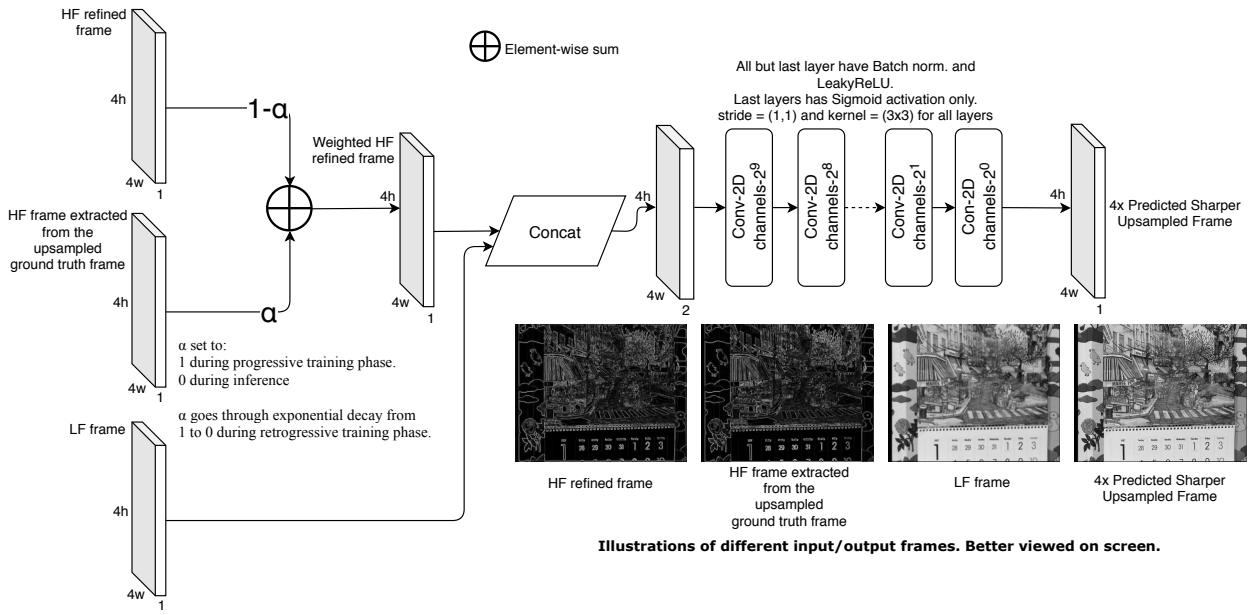


Figure 7: Stage-5: Frame fusion. This stage is trained with the proposed 2-phase *progressive-retrogressive* training. During training, it takes in 3 inputs: Refined high and low-frequency frame coming from Stage-4 and another high-frequency frame that is extracted from the given model ground-truth. In the progressive phase, α (a hyper-parameter) is set at 1 to fuse the ground-truth high-frequency frame with the low-frequency frame. In the retrogressive phase, α is exponentially decayed till 0 to gradually replace the ground-truth high-frequency frame with the refined high-frequency frame. Retrogressive phase ends when α becomes 0, disconnecting the ground-truth input from the network. Hence, ground-truth is not required for any further training or inference. During inference, the stage takes only 2 inputs from Stage-4, and α is permanently set at 0.

In the first progressive phase, instead of fusing the refined high and low-frequency frames generated at Stage-4, HFR-Net fuses the low-frequency frame with the high-frequency frame that is extracted from the given model ground-truth. As expected, the prediction generated during progressive training phase becomes much sharper and very close to the ground-truth as it uses the high-frequency details, extracted from the ground-truth itself. Once the performance metric saturates in the progressive phase, the training is restarted in the retrogressive phase.

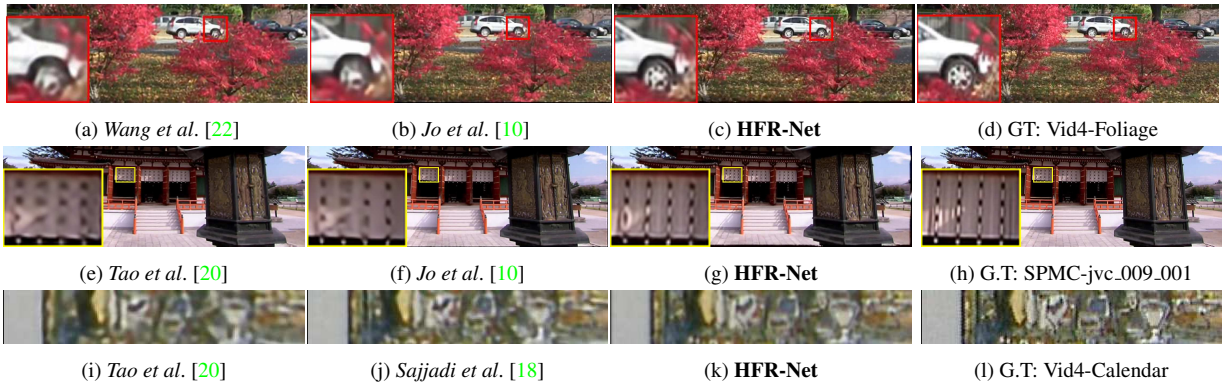
In the retrogressive phase, exponential decay is used to gradually replace the high-frequency frame that is extracted from the ground-truth with the refined high-frequency frame that is generated by Stage-4. By the end of the retrogressive phase, the ground-truth and the high-frequency frame extracted from that ground-truth are no longer needed by HFR-Net, and the entire network becomes self-sufficient to predict the upsampled frame during inference by taking-in only the seven consecutive low-resolution frames as input. The stage ground-truth is the Y (Luminance) channel of the model ground-truth scaled in range 0-1. The loss that is minimised to train this stage is the sum of mean-square-error and structural dissimilarity between stage prediction and stage ground-truth.

2.6. Training Details

The method proposed by *Glorot et al*. [3] has been used to initialise the network. Adam, with a learning rate of 10^{-4}, has been used as the optimiser. Stages were trained in sequence, freezing the previous stage. The training for each stage was stopped when no significant improvement was observed on the validation data for three epochs. The dataset provided by *Wang et al*. [23] has been used for training and validation. Low-resolution frames have been generated using the procedure given by *Jo et al*. [10]. Random flipping and rotation were used to augment the training data.

3. Experiments and Analysis

Datasets and the evaluation metric: Extensive experiments were performed on three publicly available video super-resolution datasets viz. Vid4 (by *Liu et al*. [14]), Derf4 (by *Wang et al*. [22]), and SPMCS (by *Tao et al*. [20]) to test and evaluate the performance of the proposed approach. PSNR (by *Irani et al*. [9]) and SSIM (by *Wang et al*. [28]) have been used to measure the performance. They are computed on the Luminance (Y) channel in the YCbCr colour-space of all the frames except the first and last two frames of each video clip. The average of all the computed values for a dataset is recorded as the performance value on



(a) *Wang et al*. [22] (b) *Jo et al*. [10] (c) **HFR-Net** (d) GT: Vid4-Foliage

(e) *Tao et al*. [20] (f) *Jo et al*. [10] (g) **HFR-Net** (h) G.T: SPMC-jvc_009_001

(i) *Tao et al*. [20] (j) *Sajjadi et al*. [18] (k) **HFR-Net** (l) G.T: Vid4-Calendar

Figure 8: Visualisation of upsampled frames for qualitative comparison with the current state-of-the-art methods on 4× scaling factor. Frame-crops have been obtained from the author's code/paper. The images are better viewed on-screen after zooming. GT: ground-truth.

that dataset.

3.1. Comparison with the state-of-the-art

Table 1 presents the results of HFR-Net and state-of-the-art methods for quantitative comparison. It can be seen that the proposed method outperforms the cited methods by a significant margin on all the datasets tested upon. Figure 8 has the output of HFR-Net along with the output of other state-of-the-art methods for visual/qualitative comparison. It can be observed that the upsampled frames generated by HFR-Net are sharper and visually more similar to the ground-truth than others. The idea of *dual motion warping* and 'explicit refinement of high-frequency details using 2-phase training', should be credited for this performance improvement. The 2-phase training possibly sets the network in a better local-minima than conventional training, leading to higher quality results during inference.

3.2. Effectiveness of dual motion warping

To check the effectiveness of the proposed *dual motion warping*, we replace it with three other possibilities (only direct, only sequential and no warping) and record the results in Table 2. It can be observed in the table that the *dual motion warping* performs better than any other warp-setting. Direct warping provides better results than sequential warping on all the three datasets as possibly their videos have more slow-motion segments than fast-motion segments. For such mixed motion videos, *dual motion warping* performs better.

3.3. Effectiveness of the 2-phase training

To analyse the effectiveness of the 2-phase *progressive-retrogressive* training, we compare its performance along with that of the 1-phase conventional training on the training dataset and present the results in Figure 9. It can be seen in the figure that the red line (represents the PSNR values obtained by 2-phase training) is above the green line (represents the PSNR values obtained by 1-phase training) before the diamond (*i.e*. progressive phase) due to the presence of ground-truth in the input. After the diamond (in retrogressive phase) when the ground-truth-input is slowly removed, the red line starts to fall. By the end of the training, *i.e*. after complete ground-truth removal from the input, the red line remains above the green line. This indicates that 2-phase training sets the model in a better state/local-minima than the state obtained with 1-phase training.

We further analyse the effectiveness of 2-phase train-

Table 1: PSNR and SSIM values for quantitative comparison with the state-of-the-art methods for 4× video super-resolution. All SSIM values have been multiplied by 100 to maintain symmetry.

	Vid4 dataset		SPMCS dataset		Derf4 dataset	
Model	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
[20]	26.16	78.67	30.51	86.62	30.35	83.25
[22]	26.17	78.23	30.72	86.54	30.33	82.74
[18]	26.69	81.03	31.16	86.95	30.87	85.56
[29]	26.77	81.87	31.27	87.32	30.90	86.38
[5]	26.96	81.39	31.45	87.78	31.12	85.95
[10]	27.13	82.67	31.56	89.04	31.25	87.23
[25]	27.37	83.34	31.73	89.65	31.53	87.94
HFR-Net	**29.30**	**88.49**	**33.75**	**91.90**	**33.22**	**90.31**

Table 2: Results obtained with different motion-warpings: 1) None: No warping applied, 2) Direct: Only direct warping applied, 3) Sequential: Only sequential warping applied, and 4) Dual: Both direct and sequential warpings applied. All SSIM values have been multiplied by 100.

Dataset	Warping→	None	Direct	Sequential	Dual
Vid4	PSNR	29.02	29.23	29.13	**29.30**
	SSIM	87.65	88.29	88.02	**88.49**
SPMCS	PSNR	33.14	33.34	33.30	**33.75**
	SSIM	91.04	91.64	91.47	**91.90**
DERF4	PSNR	32.85	33.07	32.98	**33.22**
	SSIM	89.24	89.95	89.73	**90.31**

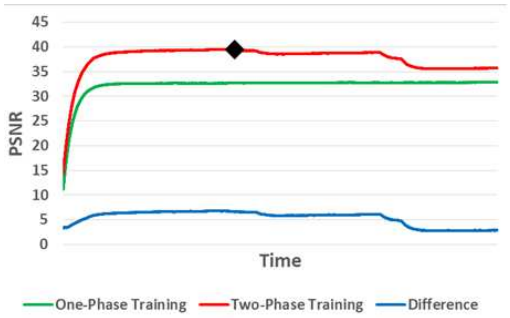


Figure 9: Comparison of 1-phase and 2-phase training performance on HFR-Net: The black diamond indicates the switch from progressive to retrogressive phase.

Table 3: Results obtained on test datasets after training the Fusion stage in conventional 1-phase, and proposed 2-phase. All SSIM values have been multiplied by 100 for symmetry.

Dataset	Vid4		SPMCS		Derf4	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
1-Phase	28.04	84.25	32.77	87.21	31.44	85.75
2-Phase	**29.30**	**88.49**	**33.75**	**91.90**	**33.22**	**90.31**

ing on test datasets by running the inference on the model trained using 1) Conventional 1-phase, and 2) Proposed 2-phase training. The results obtained are presented in Table 3. It can be seen that a significant improvement is obtained when the network is trained in 2-phases, as compared to 1-phase training. The result in training and test datasets indicate the effectiveness of the 2-phase training and that it sets the network to a better local-minima.

3.4. Effectiveness of different stages

For analysing the efficacy of different stages of HFR-Net, its stages are removed one at a time, and the network is retrained and tested. The results have been recorded in Table 4. Some important points about this experiment are: 1) HFR-Net will not work without upsampling stage as there will be no upsampled frame generated, 2) Absence of the High-frequency refinement stage, sets Fusion stage to perform the refinement of the upsampled frame and there is no fusion of refined high and low-frequency frame as those frames do not exist, 3) Removal of fusion stage implies removal of the High-frequency refinement stage also as it is of no use without the Fusion stage. In this case, the output of the upsampling stage becomes the final output, 4) Upon removing the Feature refinement stage, the rest of the stages try to compensate for its absence.

It can be observed in the table that each stage constructively contributes to the final results and performance of the network, but the 2-phase trained Fusion stage is the major contributor to the performance.

Table 4: Results obtained upon individual removal of different stages of the network. An 'x' denotes not applicable. All SSIM values have been multiplied by 100 for symmetry.

Stage Removed	Dataset→	Vid4	SPMCS	DERF4
Motion Warping	PSNR	29.02	33.14	32.85
	SSIM	87.65	91.04	89.24
Feature refinement	PSNR	29.16	33.51	33.09
	SSIM	88.07	91.47	89.94
Upsampling	PSNR	x	x	x
	SSIM	x	x	x
HF Refinement	PSNR	27.79	32.65	31.12
	SSIM	83.45	87.29	85.17
Fusion	PSNR	27.79	32.64	31.10
	SSIM	83.43	87.28	85.15
None	PSNR	**29.30**	**33.75**	**33.22**
	SSIM	**88.49**	**91.90**	**90.31**

4. Summary, limitations and future work

This work proposed 1) HFR-Net, an upsampling network that works on the principle of 'explicit refinement and fusion of high-frequency details', 2) 2-phase progressive-retrogressive training technique that temporarily requires ground-truth as an input to set the network in a better local-minima, as compared to conventional training, 3) *Dual motion warping* to preprocess videos with varying motion intensities. The quantitative and qualitative results along with ablation outcomes indicated that the proposed ideas, the working principle and the network architecture are indeed effective and when applied together, provide better results than the current state-of-the-art video super-resolution techniques.

The 2-phase training is better but also slower than 1-phase training and takes almost double the training time (keeping everything else fixed). However, the inference time of the trained network is the same irrespective of the chosen training methodology; thus, when performance is the priority, training time is not a significant limitation.

The proposed technique is being tested on similar problems such as Image super-resolution, Deraining, Dehazing, and Deblurring. The preliminary results are promising, and thus in the future, we plan to apply the proposed ideas on the mentioned problems and conduct conclusive experiments.

Acknowledgements

Authors would like to thank Dr Athira Nambiar (CV Lab, IIT-Madras) and the WACV-2020 committee (Reviewers, Area and Program chairs) for providing constructive feedback that helped in improving this work.

References

[1] A. Brifman, Y. Romano, and M. Elad. Unified single-image and video super-resolution via denoising algorithms. *IEEE*

Transactions on Image Processing, 28(12):6063–6076, Dec 2019. 2

[2] 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 *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2848–2857, July 2017. 1

[3] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In Y. W. Teh and M. Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. 6

[4] J. Guo and H. Chao. Building an end-to-end spatial-temporal convolutional network for video super-resolution. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 4053–4060. AAAI Press, 2017. 4

[5] M. Haris, G. Shakhnarovich, and N. Ukita. Recurrent back-projection network for video super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 7

[6] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, Dec 2015. 4

[7] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, July 2017. 4

[8] Y. Huang, W. Wang, and L. Wang. Video super-resolution via bidirectional recurrent convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):1015–1028, April 2018. 2

[9] M. Irani and S. Peleg. Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation*, 4(4):324 – 335, 1993. 1, 6

[10] 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 *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3224–3232, June 2018. 2, 6, 7

[11] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, June 2016. 3

[12] S. Y. Kim, J. Lim, T. Na, and M. Kim. Video super-resolution based on 3d-cnns with consideration of scene change. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2831–2835, Sep. 2019. 2

[13] S. Li, F. He, B. Du, L. Zhang, Y. Xu, and D. Tao. Fast spatio-temporal residual network for video super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[14] C. Liu and D. Sun. A bayesian approach to adaptive video super resolution. In *CVPR 2011*, pages 209–216, June 2011. 6

[15] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang. Robust video super-resolution with learned temporal dynamics. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2526–2534, Oct 2017. 1, 2, 3

[16] A. Lucas, S. Lpez-Tapia, R. Molina, and A. K. Katsaggelos. Generative adversarial networks and perceptual losses for video super-resolution. *IEEE Transactions on Image Processing*, 28(7):3312–3327, July 2019. 2

[17] O. Makansi, E. Ilg, and T. Brox. End-to-end learning of video super-resolution with motion compensation. In V. Roth and T. Vetter, editors, *Pattern Recognition*, pages 203–214, Cham, 2017. Springer International Publishing. 2

[18] M. S. M. Sajjadi, R. Vemulapalli, and M. Brown. Frame-recurrent video super-resolution. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, June 2018. 2, 7

[19] W. Shi, J. Caballero, F. Huszr, 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 *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1874–1883, June 2016. 3

[20] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia. Detail-revealing deep video super-resolution. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4482–4490, Oct 2017. 1, 2, 3, 6, 7

[21] TensorFlow. Space-to-depth. https://www.tensorflow.org/api_docs/python/tf/nn/space_to_depth. Accessed: 2019-09-28. 3, 4

[22] W. Wang, C. Ren, X. He, H. Chen, and L. Qing. Video super-resolution via residual learning. *IEEE Access*, 6:23767–23777, 2018. 2, 6, 7

[23] Z. Wang, P. Yi, K. Jiang, J. Jiang, Z. Han, T. Lu, and J. Ma. Multi-memory convolutional neural network for video super-resolution. *IEEE Transactions on Image Processing*, 28(5):2530–2544, May 2019. 2, 6

[24] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. Deepflow: Large displacement optical flow with deep matching. In *2013 IEEE International Conference on Computer Vision*, pages 1385–1392, Dec 2013. 3

[25] B. Yan, C. Lin, and W. Tan. Frame and feature-context video super-resolution. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI'19, pages 5597–5604. AAAI Press, 2019. 2, 7

[26] W. Yang, J. Feng, G. Xie, J. Liu, Z. Guo, and S. Yan. Video super-resolution based on spatial-temporal recurrent residual networks. *Computer Vision and Image Understanding*, 168:79 – 92, 2018. Special Issue on Vision and Computational Photography and Graphics. 2

[27] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma. Multi-temporal ultra dense memory network for video super-resolution. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2019. 2

[28] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. 1, 6

[29] X. Zhu, Z. Li, X.-Y. Zhang, C. Li, Y. Liu, and Z. Xue. Residual invertible spatio-temporal network for video super-resolution. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI'19, pages 5981–5988. AAAI Press, 2019. 2, 7