Video Rescaling Networks with Joint Optimization Strategies for Downscaling and Upscaling

Yan-Cheng Huang[1,*] Yi-Hsin Chen[1,*] Cheng-You Lu[1]
Hui-Po Wang[2] Wen-Hsiao Peng[1] Ching-Chun Huang[1]
[1] National Yang Ming Chiao Tung University, Taiwan
[2] CISPA Helmholtz Center for Information Security
{s0756722.iie07g, yhchen.iie07g, johnny305.cs04}@nctu.edu.tw
hui.wang@cispa.saarland, {wpeng, chingchun}@cs.nctu.edu.tw

Abstract

This paper addresses the video rescaling task, which arises from the needs of adapting the video spatial resolution to suit individual viewing devices. We aim to jointly optimize video downscaling and upscaling as a combined task. Most recent studies focus on image-based solutions, which do not consider temporal information. We present two joint optimization approaches based on invertible neural networks with coupling layers. Our Long Short-Term Memory Video Rescaling Network (LSTM-VRN) leverages temporal information in the low-resolution video to form an explicit prediction of the missing high-frequency information for upscaling. Our Multi-input Multi-output Video Rescaling Network (MIMO-VRN) proposes a new strategy for downscaling and upscaling a group of video frames simultaneously. Not only do they outperform the image-based invertible model in terms of quantitative and qualitative results, but also show much improved upscaling quality than the video rescaling methods without joint optimization. To our best knowledge, this work is the first attempt at the joint optimization of video downscaling and upscaling.

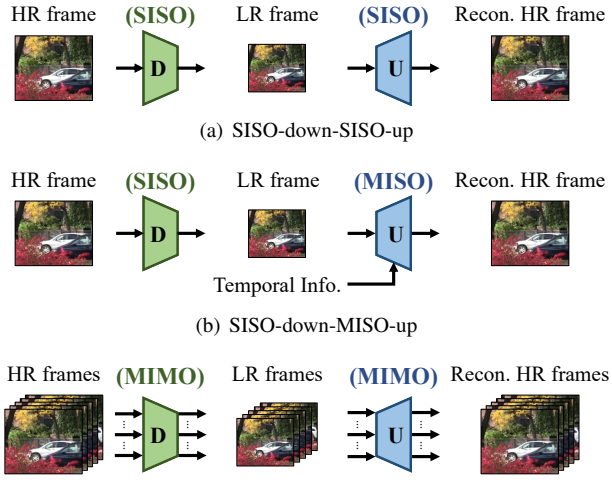


Figure 1. Comparison of video rescaling frameworks according to the downscaling and upscaling strategies: (a) single-input single-output (SISO) for both operations, (b) SISO for downscaling and multi-input single-output (MISO) for upscaling, and (c) multi-input multi-output (MIMO) for both operations (the proposed method).

1. Introduction

With the increasing popularity of video capturing devices, a tremendous amount of high-resolution (HR) videos are shot every day. These HR videos are often downscaled to save storage space and streaming bandwidth, or to fit screens with lower resolutions. It is also common that the downscaled videos need to be upscaled for display on HR monitors [11, 16, 21, 2, 28].

In this paper, we address the joint optimization of video downscaling and upscaling as a combined task, which is referred to as *video rescaling*. This task involves downscaling an HR video into a low-resolution (LR) one, followed by upscaling the resulting LR video back to HR. Our aim is to optimize the HR reconstruction quality while regularizing the LR video to offer comparable visual quality to the bicubic-downscaled video for human perception. It is to be noted that the rescaling task differs from the super-resolution task; at inference time, the former has access to the HR video while the latter has no such information.

One straightforward solution to video rescaling is to downscale an HR video by predefined kernels and upscale the LR video with super-resolution methods [14, 17, 31, 25, 3, 6, 1, 22, 19, 10, 24, 30, 8, 15, 9]. With this solution,

*Both authors contributed equally to this work.

the downscaling is operated independently of the upscaling although the upscaling can be optimized for the chosen downscaling kernels. The commonly used downscaling (e.g. bicubic) kernels suffer from losing the high-frequency information [20] inherent in the HR video, thus creating a many-to-one mapping between the HR and LR videos. Reconstructing the HR video by upscaling its LR representation becomes an ill-posed problem. The independently-operated downscaling misses the opportunity of optimizing the downscaled video to mitigate the ill-posedness.

The idea of jointly optimizing downscaling and upscaling was first proposed for image rescaling [11, 16, 21, 2]. It adds a new dimension of thinking to the studies of learning specifically to upscale for a given downscaling method [14, 17, 31, 25, 3, 6]. Recognizing the reciprocality of the downscaling and upscaling operations, IRN [28] recently introduced a coupling layer-based invertible model, which shows much improved HR reconstruction quality than the non-invertible models.

These jointly optimized image-based solutions (Fig. 1(a)) are not ideal for video rescaling. For example, a large number of prior works [1, 22, 19, 10, 24, 30, 8, 15, 9] for video upscaling have adopted the Multi-Input Single-Output (MISO) strategy to reconstruct one HR frame from multiple LR frames and/or previously reconstructed HR frames (Fig. 1(b)). They demonstrate the potential for recovering the missing high-frequency component of a video frame from temporal information. However, image-based solutions do not consider temporal information. In addition, two issues remain widely open as (1) how video downscaling and upscaling could be jointly optimized and (2) how temporal information could be utilized in the joint optimization framework to benefit both operations.

In this paper, we present two joint optimization approaches to video rescaling: Long Short-Term Memory Video Rescaling Network (LSTM-VRN) and Multi-Input Multi-Output Video Rescaling Network (MIMO-VRN). The LSTM-VRN downscales an HR video frame-by-frame using a similar coupling architecture to [28], but fuses multiple downscaled LR video frames via LSTM to estimate the missing high-frequency component of an LR video frame for upscaling (Fig. 1(b)). LSTM-VRN shares similar downscaling and upscaling strategies to the traditional video rescaling framework. In contrast, our MIMO-VRN introduces a completely new paradigm by adopting the MIMO strategy for both video downscaling and upscaling (Fig. 1(c)). We develop a group-of-frames-based (GoF) coupling architecture that downscales multiple HR video frames simultaneously, with their high-frequency components being estimated also simultaneously in the upscaling process. Our contributions include the following:

- To the best of our knowledge, this work is the first attempt at jointly optimizing video downscaling and upscaling with invertible coupling architectures.
- Our LSTM-VRN and MIMO-VRN outperform the image-based invertible model [28], showing significantly improved HR reconstruction quality and offering LR videos comparable to the bicubic-downscaled video in terms of visual quality.
- Our MIMO-VRN is the first scheme to introduce the MIMO strategy for video upscaling and downscaling, achieving the state-of-the-art performance.

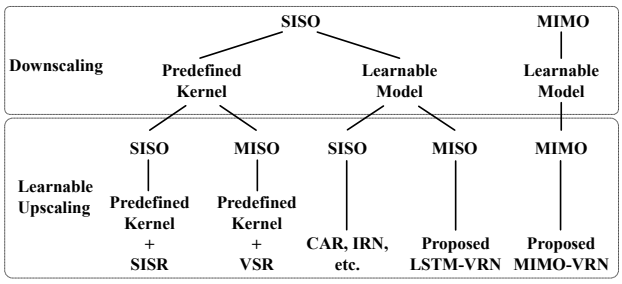


Figure 2. Taxonomy of the prior works on image/video rescaling. The SISO, MISO and MIMO indicate the strategies (i.e. the input/output format) for downscaling and upscaling. SISR and VSR stand for single image super-resolution and video super-resolution, respectively. CAR [21] and IRN [28] are joint optimization schemes for image rescaling.

2. Related Work

This section surveys video rescaling methods, with a particular focus on their downscaling and upscaling strategies. We regard the image-based rescaling methods as possible solutions for video rescaling. Fig. 2 is a taxonomy of these prior works.

2.1. Upscaling with Predefined Downscaling

The traditional image super-resolution [14, 17, 31, 25, 3, 6] or video super-resolution [1, 22, 19, 10, 24, 30, 8, 15, 9] methods are candidate solutions to video upscaling. The former is naturally a single-input single-output (SISO) upscaling strategy, which generates one HR image from one LR image. The latter usually involves more than one LR video frame in the upscaling process, i.e. the MISO upscaling strategy, in order to leverage temporal information for better HR reconstruction quality. Most of the approaches in this category adopt a SISO downscaling strategy with a pre-defined kernel (e.g. bicubic) chosen independently of the upscaling process. Therefore, they are unable to adapt the downscaled images/videos to the upscaling.

2.2. Upscaling with Jointly Learned Downscaling

To mitigate the ill-posedness of the image upscaling task, some works learn upscaling and downscaling jointly by

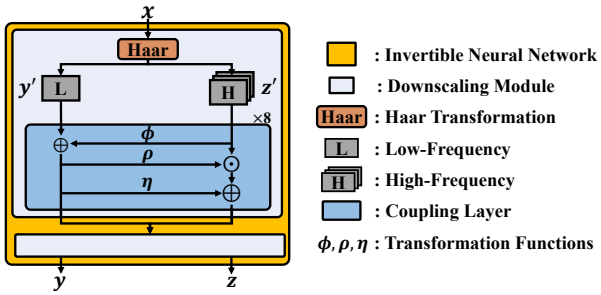


Figure 3. The detailed downscaling operation of IRN [28]. The model performs $\times 4$ downscaling with two downscaling modules, each of which comprises a 2-D Haar transform and eight coupling layers. Each downscaling module halves the horizontal and vertical resolutions of the input image.

encoder-decoder architectures [11, 16, 21, 2]. They turn the fixed downscaling method into a learnable model in order to adapt the LR image to the upscaling process that is learned jointly. The training objective usually requires the LR image to be also suitable for human perception. Recently, IRN [28] introduces an invertible model [4, 5, 13] to this joint optimization task. It is able to perform image downscaling and upscaling by the same set of neural networks configured in the reciprocal manner. It provides a means to model explicitly the missing high-frequency information due to downscaling by a Gaussian noise.

2.3. Invertible Rescaling Network

IRN [28] is an invertible model designed specifically for image rescaling. The forward model of IRN comprises a 2-D Haar transform and eight coupling layers [4, 5, 13], as shown in Fig. 3. By applying the 2-D Haar transform, an input image $x \in \mathbb{R}^{C\times H\times W}$ is first decomposed into one low-frequency band $y' \in \mathbb{R}^{C\times \frac{H}{2}\times \frac{W}{2}}$ and three other high-frequency bands $z' \in \mathbb{R}^{3C\times \frac{H}{2}\times \frac{W}{2}}$. These two components y', z' are subsequently processed via the coupling layers in a way that the output y becomes a visually-pleasing LR image and the z encodes the complementary high-frequency information inherent in the input HR image x. In theory, the inverse coupling layers can recover x losslessly from y and z because the model is invertible. In practice, z is unavailable for upscaling at inference time. The training of IRN requires z to follow a Gaussian distribution so that at inference time, a Gaussian sample $\hat{z}$ can be drawn as a substitute for the missing high-frequency component.

Although IRN achieves superior results on the image rescaling task, it is not optimal for video rescaling. Essentially, IRN is an image-based method. This work presents the first attempt at jointly optimizing video downscaling and upscaling with an invertible coupling architecture (Fig. 3).

3. Proposed Method

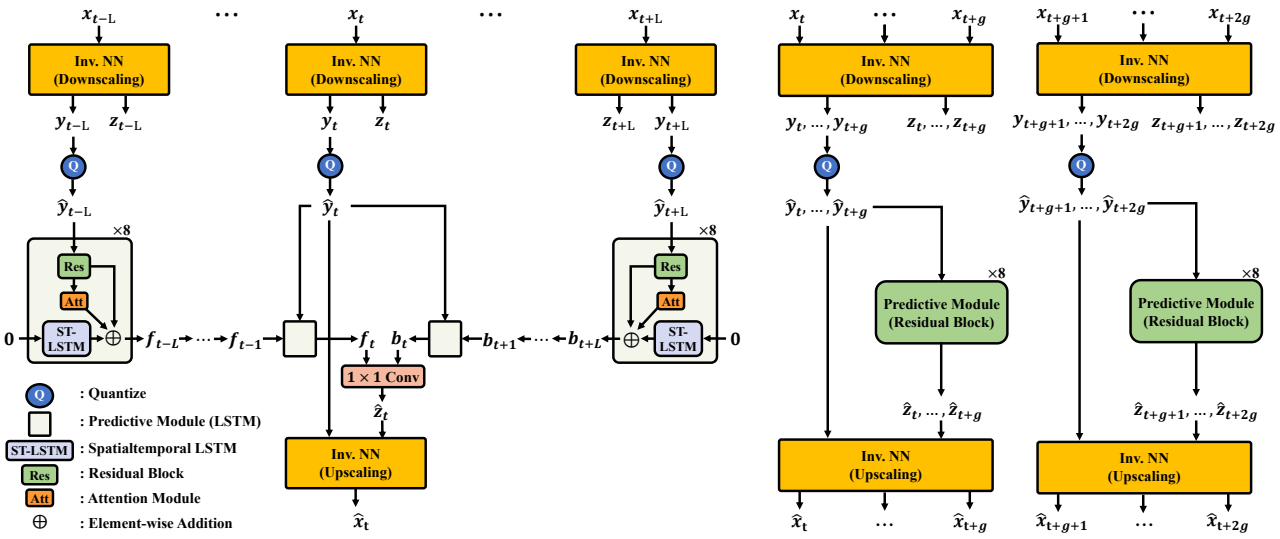
Given an HR video composed of N video frames $\{x_t\}_{t=1}^N$, where $x_t \in \mathbb{R}^{C\times H\times W}$, the video rescaling task involves (1) downscaling every video frame x_t to its LR counterpart $y_t \in \mathbb{R}^{C\times \frac{H}{4}\times \frac{W}{4}}$, where the quantized version $\hat{y}_t$ of which forms collectively an LR video $\{\hat{y}_t\}_{t=1}^N$, and (2) upscaling the LR video to arrive at the reconstructed HR video $\{\hat{x}_t\}_{t=1}^N$. Unlike most video super-resolution tasks, which focus primarily on learning upscaling for a given downscaling method, this work optimizes jointly the downscaling and upscaling as a combined task. It has been shown in many traditional video super-resolution works [1, 22, 19, 7, 23, 24, 10, 30, 9, 15, 8] that the extra temporal information in videos allows the lost high-frequency component of a video frame due to downscaling to be recovered to some extent. This work makes the first attempt to explore how such temporal information could assist downscaling in producing an LR video that can be upscaled to offer better super-resolution quality in an end-to-end fashion. In a sense, our focus is on both downscaling and upscaling. The objective is to minimize the distortion between $\{\hat{x}_t\}_{t=1}^N$ and $\{x_t\}_{t=1}^N$ in such a combined task while the LR video $\{\hat{y}_t\}_{t=1}^N$ is regularized to offer comparable visual quality to the bicubic-downscaled video for human perception. It is to be noted that the LR video is not meant to be exactly the same as the bicubic-downscaled video since doing so may not lead to the optimal downscaling and upscaling in our task.

The reciprocality of the downscaling and upscaling operations motivates us to choose an invertible network for our task. With the superior performance of coupling layer architectures in recovering high-frequency details of LR images [28], we develop our downscaling and upscaling networks, especially for video, using a similar invertible architecture (Sec. 2.3) as the basic building block.

We propose two approaches, LSTM-VRN and MIMO-VRN, to configure or extend these building blocks for joint learning of video downscaling and upscaling. Their overall architectures are depicted in Fig. 4, with detailed operations given in the following sections.

3.1. LSTM-based Video Rescaling Network

Like most video super-resolution techniques, the LSTM-VRN (Fig. 4(a)) adopts the SISO strategy to downscale HR video frames $\{x_t\}_{t=1}^N$ individually to their LR ones $\{\hat{y}_t\}_{t=1}^N$ by the forward model of the invertible network. The operation is followed by the MISO-based upscaling, which departs from the idea of drawing an input-agnostic Gaussian noise [28] for complementary high-frequency information. Specifically, we fuse the current LR frame $\hat{y}_t$ and its neighbouring frames $\{\hat{y}_{t-i}, \hat{y}_{t+i}\}_{i=1}^L$ by a LSTM-based predictive module to form an estimate $\hat{z}_t$ of the missing high-frequency component z_t at inference time. The resulting



(a) LSTM-VRN (b) MIMO-VRN

Figure 4. Overview of the proposed LSTM-VRN and MIMO-VRN for video rescaling. Both schemes involve an invertible network with coupling layers for video downscaling and upscaling. In part (a), LSTM-VRN downscales every video frame x_t independently and forms a prediction $\hat{z}_t$ of the high-frequency component z_t from the LR video frames $\{\hat{y}_i\}_{i=t-L}^{t+L}$ by a bi-directional LSTM that operates in a sliding window manner. In part (b), MIMO-VRN downscales a group of HR video frames $\{x_i\}_{i=t}^{t+g}$ into the LR video frames $\{\hat{y}_i\}_{i=t}^{t+g}$ simultaneously. The upscaling is also done on a group-by-group basis, with the high-frequency components $\{z_i\}_{i=t}^{t+g}$ estimated from the $\{\hat{y}_i\}_{i=t}^{t+g}$ by a predictive module.

$\hat{z}_t$ is fed to the inverse model together with the $\hat{y}_t$ for reconstructing the HR video frame $\hat{x}_t$. The fact that z_t needs to be estimated from multiple LR frames $\{\hat{y}_i\}_{i=t-L}^{t+L}$ determines what information should remain in the LR video to facilitate the prediction. This connects the upscaling process tightly to the downscaling process, stressing the importance of their joint optimization. In addition, we rely on the inter-branch pathways of the coupling layer in the forward model to correlate z_t and y_t in such a way that z_t could be better predicted from $\hat{y}_t$ and its neighbors $\{\hat{y}_{t-i}, \hat{y}_{t+i}\}_{i=1}^{L}$.

The predictive module plays a key role in fusing information from $\hat{y}_t$ and $\{\hat{y}_{t-i}, \hat{y}_{t+i}\}_{i=1}^{L}$. We incorporate Spatiotemporal-LSTM (ST-LSTM) [26] for propagating temporal information in both forward and backward directions, in view of its recent success in video extrapolation tasks. Eq. (1) details the forward mode of the predictive module for time instance t:

$$\begin{aligned} h_t^f &= ST\text{-}LSTM(f_{t-1}, h_{t-1}^f) \\ h_t^y &= ResidualBlock(\hat{y}_t) \\ a_t &= \sigma(W \otimes h_t^y) \\ f_t &= (1 - a_t) \odot h_t^f + a_t \odot h_t^y \end{aligned} \tag{1}$$

where σ is a sigmoid function, $\otimes$ is the standard convolution, and $\odot$ is Hadamard product. Note that an attention signal a_t guided by the the current LR frame $\hat{y}_t$ combines the temporally-propagated hidden information h_t^f and the features h_t^y of $\hat{y}_t$ to yield the output f_t. As Fig. 4(a) shows, the forward propagated f_t is further combined with the backward propagated b_t to predict $\hat{z}_t$ through a 1x1 convolution.

For upscaling every LR video frame $\hat{y}_t$, the proposed predictive module works in a sliding-window manner with a window size of $2L + 1$. That is, the forward (respectively, backward) ST-LSTM always starts with a reset state 0 when accepting the input $\hat{y}_{t-L}$ (respectively, $\hat{y}_{t+L}$). This design choice is out of generalization and buffering considerations. We avoid running a long ST-LSTM at inference time because the training videos are rather short. Moreover, the backward ST-LSTM introduces delay and buffering requirements.

Finally, we note in passing that LSTM-VRN exploits temporal information across LR video frames only for upscaling while its downscaling is still a SISO-based scheme, which does not take advantage of temporal information in HR video frames for downscaling.

3.2. MIMO-based Video Rescaling Network

Our MIMO-VRN (Fig. 4(b)) is a new attempt that adopts a MIMO strategy for both upscaling and downscaling, making explicit use of temporal information in these operations. Here, we propose a new basic processing unit, called Group of Frames (GoF). To begin with, the HR input video $\{x_t\}_{t=1}^{N}$ is decomposed into non-overlapping groups of frames, with each group including g frames, namely $\{x_t\}_{t=1}^{g}, \{x_t\}_{t=g+1}^{2g}, \ldots$. The downscaling pro-

ceeds on a group-by-group basis; each GoF is downscaled independently of each other. Within a GoF, every HR video frame $x_t \in \mathbb{R}^{C\times H\times W}$ is first transformed individually using 2-D Haar Wavelet, to arrive at its low-frequency $y'_t \in \mathbb{R}^{C\times \frac{H}{2}\times \frac{W}{2}}$ and high-frequency $z'_t \in \mathbb{R}^{3C\times \frac{H}{2}\times \frac{W}{2}}$ components. We then group the low-frequency components $\{y'_i\}_{i=t}^{t+g}$ in a GoF as one group-type input $\mathcal{Y}'_t \in \mathbb{R}^{gC\times \frac{H}{2}\times \frac{W}{2}}$ to the coupling layers (i.e. replacing y' in Fig. 3 with $\mathcal{Y}'_t$) and the remaining high-frequency components $\{z'_i\}_{i=t}^{t+g}$ as the other group-type input $\mathcal{Z}'_t \in \mathbb{R}^{3gC\times \frac{H}{2}\times \frac{W}{2}}$ (i.e. replacing z' in Fig. 3 with $\mathcal{Z}'_t$). Because each group-type input contains information from multiple video frames, the coupling layers are able to utilize temporal information inherent in one group-type input to update the other. With two downscaling modules, the results are a group of quantized LR frames $\hat{\mathcal{Y}}_t = \{\hat{y}_i\}_{i=t}^{t+g}$ and the group high-frequency component $\mathcal{Z}_t = \{z_i\}_{i=t}^{t+g}$. It is worth noting that due to the nature of group-based coupling, there is no one-to-one correspondence between the signals in $\hat{\mathcal{Y}}_t \in \mathbb{R}^{gC\times \frac{H}{4}\times \frac{W}{4}}$ and $\mathcal{Z}_t \in \mathbb{R}^{3gC\times \frac{H}{4}\times \frac{W}{4}}$.

The upscaling proceeds also on a group-by-group basis, with the group size g and the group formation fully aligned with those used for downscaling. As depicted in Fig. 4(b), we employ a residual block-based predictive module to form a prediction of the missing high-frequency components $\{z_i\}_{i=t}^{t+g}$ from the corresponding group of LR frames $\{\hat{y}_i\}_{i=t}^{t+g}$. Similar to the notion of the group-type inputs for downscaling, the LR frames $\{\hat{y}_i\}_{i=t}^{t+g}$ and the estimated high-frequency components $\{\hat{z}_i\}_{i=t}^{t+g}$ comprise respectively the two group-type inputs $\hat{\mathcal{Y}}_t$ and $\hat{\mathcal{Z}}_t$ to the invertible network operated in inverse mode. With this MIMO-based upscaling, a group of HR frames $\{\hat{x}_i\}_{i=t}^{t+g}$ are reconstructed simultaneously.

3.3. Training Objectives

LSTM-VRN. The training of LSTM-VRN involves two loss functions to reflect our objectives. First, to ensure that the LR video $\{\hat{y}_t\}_{t=1}^{N}$ is visually pleasing, we follow common practice to require that $\{\hat{y}_t\}_{t=1}^{N}$ have similar visual quality to the bicubic-downscaled video $\{x_t^{bic}\}_{t=1}^{N}$; to this end, we define the LR loss as

$$\mathcal{L}_{LR} = \frac{1}{N}\sum_{t=1}^{N} \|x_t^{bic} - \hat{y}_t\|^2. \tag{2}$$

Second, to maximize the HR reconstruction quality, we minimize the Charbonnier loss [14] between the original HR video $\{x_t\}_{t=1}^{N}$ and its reconstructed version $\{\hat{x}_t\}_{t=1}^{N}$ subject to downscaling and upscaling:

$$\mathcal{L}_{HR} = \frac{1}{N}\sum_{t=1}^{N} \sqrt{\|x_t - \hat{x}_t\|^2 + \epsilon^2}, \tag{3}$$

where ϵ is set to 1×10^{-3}. The total loss is $\mathcal{L}_{total} = \mathcal{L}_{HR} + \lambda\mathcal{L}_{LR}$, where λ is a hyper-parameter used to trade-off between the quality of the LR and HR videos.

MIMO-VRN. The training of MIMO-VRN shares the same $\mathcal{L}_{LR}$ and $\mathcal{L}_{HR}$ losses as LSTM-VRN because they have common optimization objectives. We however notice that MIMO-VRN tends to have uneven HR reconstruction quality over video frames in a GoF (Sec. 4.4). To mitigate the quality fluctuation in a GoF, we additionally introduce the following center loss for MIMO-VRN:

$$\mathcal{L}_{center} = \frac{1}{M\times g}\sum_{m=1}^{M}\sum_{t=(m-1)g+1}^{mg} \left| \|x_t - \hat{x}_t\|^2 - c_m \right|, \tag{4}$$

where g is the group size, $c_m = \sum_{t=(m-1)g+1}^{mg} \|x_t - \hat{x}_t\|^2/g$ denotes the average HR reconstruction error in a GoF, and M is the number of GoF's in a sequence. Eq. (4) encourages the HR reconstruction error of every video frame in a GoF to approximate the average level c_m.

4. Experimental Results

4.1. Setup

Datasets. For a fair comparison, we follow the common test protocol to train our models on Vimeo-90K dataset [29]. It has 91,701 video sequences, each is 7 frames long. Among them, 64,612 sequences are for training and 7,824 are for test. Each sequence has a fixed spatial resolution of 448 × 256. The performance evaluation is done on two standard test datasets, Vimeo-90K-T and Vid4 [18]. Vid4 includes 4 video clips, each having around 40 frames.

Implementation and Training Details. Our proposed models adopt the settings from IRN [28], which consists of two downscaling modules (Fig. 3). Each module is composed of one 2-D Haar transform and eight coupling layers. Both LSTM-VRN and MIMO-VRN have eight predictive modules (Fig. 4) replicated and stacked for a better prediction of the missing high-frequency component. The sliding window size for LSTM-VRN is set to 7, which includes the current LR video frame together with 6 neighbouring LR frames (3 from the past and 3 from the future). The GoF size g for MIMO-VRN is set to 5. For data augmentation, we randomly crop training videos to 144 × 144 as HR inputs and use their bicubic-downscaled versions (of size 36 × 36) as LR ground-truths. We also apply random horizontal and vertical flipping. LSTM-VRN and MIMO-VRN share the same LR and HR training objectives (Eq. (2) and Eq. (3)), with the λ for $\mathcal{L}_{LR}$ set to 64. The training of MIMO-VRN additionally includes the center loss (Eq. (4)), the hyper-parameter of which is chosen to be 16. We use Adam optimizer [12], with $\beta_1 = 0.9$, $\beta_2 = 0.5$ and a batch size of 16. The weight decay is set to 1×10^{-12}. We use an

Table 1. PSNR-Y / SSIM-Y comparison on Vid4 for $\times 4$ upscaling. '†' represents the model adopting the joint optimization for downscaling and upscaling. Red, green, and blue indicate the best, the second best, and the third best performance, respectively.

Downscale	Upscale	Method	Calendar	City	Foliage	Walk	Average
SISO	SISO	DRN-L [6]	22.47 / 0.7289	26.25 / 0.7011	24.88 / 0.6681	28.84 / 0.8752	25.61 / 0.7433
		CAR† [21]	24.48 / 0.8143	30.19 / 0.8444	26.98 / 0.7841	31.59 / 0.9250	28.28 / 0.8421
		IRN† [28]	26.62 / 0.8850	33.48 / 0.9337	29.71 / 0.8871	35.36 / 0.9696	31.29 / 0.9188
	MISO	DUF [10]	24.04 / 0.8110	28.27 / 0.8313	26.41 / 0.7709	30.60 / 0.9141	27.33 / 0.8318
		EDVR-L [24]	24.05 / 0.8147	28.00 / 0.8122	26.34 / 0.7635	31.02 / 0.9152	27.35 / 0.8264
		PFNL [30]	24.37 / 0.8246	28.09 / 0.8385	26.51 / 0.7768	30.65 / 0.9135	27.40 / 0.8384
		TGA [9]	24.47 / 0.8286	28.37 / 0.8419	26.59 / 0.7793	30.96 / 0.9181	27.59 / 0.8419
		RSDN [8]	24.60 / 0.8355	29.20 / 0.8527	26.84 / 0.7931	31.04 / 0.9210	27.92 / 0.8505
		LSTM-VRN†	27.31 / 0.9039	34.36 / 0.9482	31.13 / 0.9213	36.18 / 0.9742	32.24 / 0.9369
MIMO	MIMO	MIMO-VRN†	29.23 / 0.9389	35.49 / 0.9573	33.25 / 0.9535	37.17 / 0.9812	33.79 / 0.9577
		MIMO-VRN-C†	28.83 / 0.9322	35.13 / 0.9544	32.72 / 0.9476	36.93 / 0.9808	33.40 / 0.9537

Table 2. PSNR-Y / SSIM-Y comparison on Vimeo-90K-T for $\times 4$ upscaling. '†' represents the model adopting the joint optimization for downscaling and upscaling. Red, green, and blue indicate the best, the second best, and the third best performance, respectively.

Downscale	Upscale	Method	Average
SISO	SISO	DRN-L [6]	35.63 / 0.9262
		CAR† [21]	37.69 / 0.9493
		IRN† [28]	40.83 / 0.9734
	MISO	DUF [10]	36.37 / 0.9387
		EDVR-L [24]	37.63 / 0.9487
		TGA [9]	37.59 / 0.9516
		RSDN [8]	37.23 / 0.9471
		LSTM-VRN†	41.42 / 0.9764
MIMO	MIMO	MIMO-VRN†	43.26 / 0.9846
		MIMO-VRN-C†	42.53 / 0.9820

initial learning rate of 1×10^{-4}, which is decreased by half for every $30k$ iterations. Our code is available online [1].

Baselines. We include three categories of baselines for comparison: (1) SISO-down-SISO-up with predefined downscaling kernels (e.g. DRN-L [6]), (2) SISO-down-SISO-up with jointly optimized downscaling and upscaling (e.g. CAR [21] and IRN [28]), and (3) SISO-down-MISO-up with predefined downscaling kernels (e.g. DUF [10], EDVR-L [24], PFNL [30], TGA [9], and RSDN [8]). The first two categories perform video downscaling and upscaling on a frame-by-frame basis. The third category includes the state-of-the-art video super-resolution methods, where the predefined downscaling is done frame-by-frame and the learned upscaling is MISO-based. The predefined downscaling uses the bicubic interpolation method. It is to be noted that the methods adopting the learned downscaling perform upscaling based on their respective LR videos, which would not be the same as the bicubic-downscaled videos. The results for the methods in categories (1) and (2) are produced using the pre-trained models released by the authors. Those in category (3) are taken from the papers since these baselines share exactly the same setting as ours. We report results for a downscaling/upscaling factor of 4 only, following the common setting for video rescaling.

Metrics. For quantitative comparison, we adopt the standard test protocol in the super-resolution tasks to evaluate Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [27] on the Y channel, denoted respectively by PSNR-Y and SSIM-Y.

4.2. Comparison of Quantitative Results

Tables 1 and 2 report the PSNR-Y and SSIM-Y results of the reconstructed HR videos on Vid4 and Vimeo-90K-T. Table 3 summarizes the results for the downscaled videos. The following observations are immediate:

(1) *Optimizing jointly video downscaling and upscaling improves the HR reconstruction quality.* This is confirmed by the fact that LSTM-VRN achieves considerably higher PSNR-Y (32.24dB on Vid4 and 41.42dB on Vimeo-90K-T) than the baselines with video super-resolution methods for upscaling (27.33-27.92dB on Vid4 and 36.37-37.59dB on Vimeo-90K-T) [24, 10, 30, 9, 8], which adopt the same SISO-down-MSIO-up strategy yet with a predefined downscaling kernel. We note that the image-based joint optimization schemes, e.g. IRN [28] and CAR [21], achieve better HR reconstruction quality than the traditional video-based baselines, even without using temporal information for upscaling. The superior performance of joint optimization schemes is attributed to the fact that they can better embed HR information in LR frames for upscaling.

(2) *Incorporating temporal information in the LR video improves further on the HR reconstruction quality.* The result is evidenced by the 0.95dB and 0.59dB PSNR-Y gains of LSTM-VRN over IRN [28] on Vid4 and Vimeo-90K-T. Both share a similar invertible network for downscaling, but our LSTM-VRN additionally leverages information from multiple LR video frames to predict the high-

[1] https://ding3820.github.io/MIMO-VRN/

Figure 5. Sample LR video frames from Vid4. Our models show comparable visual quality to the bicubic method.

Table 3. PSNR-Y and SSIM-Y results measured between the $\times 4$ downscaled LR videos and the bicubic-downscaled videos.

Method	Vid4	Vimeo-90K-T
IRN [28]	40.77 / 0.9908	46.24 / 0.9956
LSTM-VRN	42.36 / 0.9940	47.14 / 0.9968
MIMO-VRN	45.05 / 0.9965	49.11 / 0.9975
MIMO-VRN-C	45.51 / 0.9969	49.34 / 0.9976

frequency component of a video frame during upscaling.

(3) *MIMO-VRN achieves the best PSNR-Y/SSIM-Y results.* It outperforms LSTM-VRN by 1.55dB and 1.84dB in PSNR-Y on Vid4 and Vimeo-90K-T, respectively, while LSTM-VRN already shows a significant improvement over the other baselines. The inclusion of the center loss (see MIMO-VRN-C) causes a modest decrease in PSNR-Y/SSIM-Y but helps to alleviate the quality fluctuation in both the resulting LR and HR videos (Sec. 4.4). These results highlight the benefits of incorporating temporal information into both downscaling and upscaling in an end-to-end optimized manner.

(4) *Both LSTM-VRN and MIMO-VRN produce visually-pleasing LR videos.* Table 3 shows that the LR videos produced by our models have a PSNR-Y of more than 40dB when compared against the bicubic-downscaled videos. This together with the SSIM-Y results suggests that they are visually comparable to the bicubic-downscaled videos, as is also confirmed by the subjective quality comparison in Fig. 5 and the supplementary document.

4.3. Comparison of Qualitative Results

Figs. 6 presents a qualitative comparison on Vid4. As shown, our models produce higher-quality HR video frames with much sharper edges and finer details. The other methods show blurry image quality and fail to recover image details. From Fig. 5, our downscaling models produce visually comparable results to the bicubic downscaling method, which indicates the visually-pleasing property of our LR videos. The reader is referred to our project page [1] for more results.

4.4. Ablation Experiments

Temporal Propagation Methods in LSTM-VRN. Table 4 presents results for three temporal propagation schemes in LSTM-VRN. The first runs LSTM in forward direction without reset. The second and the third implement the proposed method with uni- or bi-directional propagation, respectively. We see that the sliding window-based reset is advantageous to the HR reconstruction quality. This may be attributed to the fact that the training videos in Vimeo-90K are rather short. When trained on Vimeo-90K, the first variant may not generalize well to unseen long videos in Vid4. As expected, with the access to both the past and future LR frames, the bi-directional propagation performs better than the uni-directional one (i.e. Fig. 4(a) without the backward path).

Table 4. Ablation study of the propagation methods for LSTM-VRN. Results are reported on Vid4.

Sliding Window	Bi-directional	PSNR-Y
		31.16
√		31.53
√	√	32.24

Table 5. PSNR-Y of different GoF sizes on Vid4. IRN_Ret is the re-trained IRN with Vimeo-90K, as compared to IRN, the pre-trained model from [28].

Method	HR	LR
IRN [28]	31.29	41.13
IRN_Ret	30.72	45.06
GoF1	30.69	44.38
GoF3	33.61	43.85
GoF5	33.79	45.05
GoF7	33.45	45.13

GoF Size. Table 5 studies the effect of the GoF size on MIMO-VRN's performance. The setting GoF1 reduces to the SISO-up-SISO-down method, which is similar to IRN [28] except that it introduces a prediction of the high-frequency component from the LR video frame. For a fair comparison, we re-train IRN [28] on Vimeo-90K and denote the re-trained model by IRN_Ret. Note that the pre-trained IRN [28] performs better than IRN_Ret since it is trained on a different (image-based) dataset. We see that GoF1 and IRN_Ret show comparable performance, especially on the HR videos. This suggests that without additional temporal information, the prediction of the high-frequency component from the LR video is ineffective. However, increasing the GoF size, which involves more temporal information in downscaling and upscaling, improves the quality of the HR video significantly. GoF5 is seen to be the best setting.

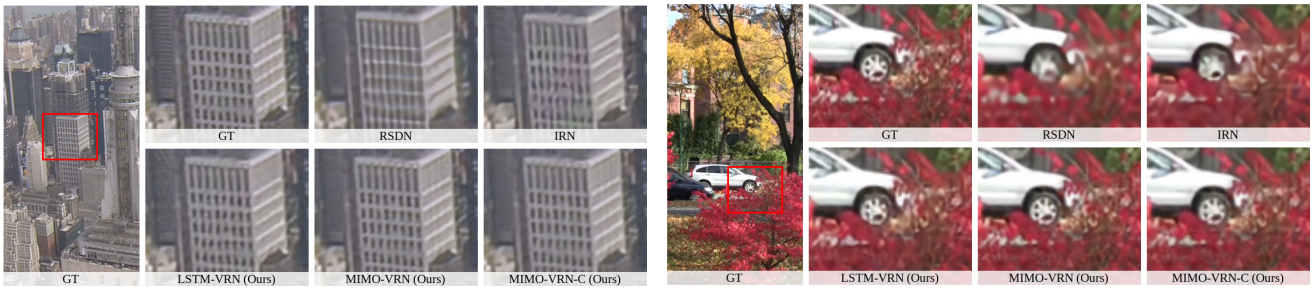
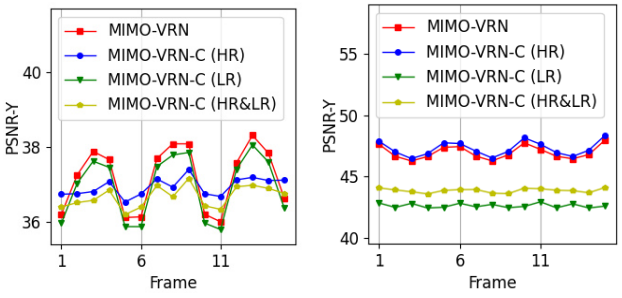


Figure 6. Qualitative comparison on Vid4 for 4× upscaling. Zoom in for better visualization.



(a) Reconstructed HR video (b) Downscaled LR video

Figure 7. The impact of the center loss on the quality of the HR and LR videos. The per-frame PSNR-Y is visualized as a function of frame indices. The GoF size is 5. **MIMO-VRN**: no center loss. **HR**: the center loss imposed on the HR video only. **LR**: the center loss imposed on the LR video only. **HR&LR**: the center loss imposed on both the HR and LR videos.

Table 6. PSNR-Y of MIMO-VRN with and without the center loss on Vid4. The mean absolute deviation (MAD) indicates the average absolute deviation of the per-frame PSNR-Y from the GoF mean.

Center loss		PSNR-Y		MAD	
HR	LR	HR	LR	HR	LR
		33.79	45.05	0.88	0.55
√		33.40	45.54	0.28	0.63
	√	33.55	42.42	0.86	0.38
√	√	33.13	43.32	0.31	0.29

Center Loss. Fig. 7 visualizes the PSNR-Y of the HR and LR videos produced by MIMO-VRN as functions of time. Without the center loss (see MIMO-VRN), the PSNR-Y of both the HR and LR videos fluctuates periodically by as much as 2dB. Observe that the crest points of the HR video occur roughly at the GoF centers while the trough points are at the GoF boundaries. Table 6 performs an ablation study of how this center loss would affect the HR and/or LR videos when it is imposed on these videos. We observe that introducing the center loss largely mitigates the quality fluctuation in the corresponding HR and/or LR video (see the MAD results in Table 6 and Fig. 7). It however degrades the HR and/or LR quality in terms of PSNR-Y, as compared to the case without the loss. We make the choice of imposing the center loss on the HR video only for two reasons. First, this leads to a minimal impact on the HR reconstruction quality. The second is that the quality fluctuation in the LR video is less problematic in terms of subjective quality because the PSNR-Y measured against the bicubic-downscaled video is way above 40dB. On closer visual inspection, these LR videos hardly show any artifacts in the temporal dimension.

4.5. Complexity-performance Trade-offs

LSTM-VRN and MIMO-VRN present different complexity-performance trade-offs. (1) LSTM-VRN is relatively lightweight, having 9M network parameters as compared to 19M with MIMO-VRN. (2) LSTM-VRN does not require additional buffering/delay and storage for downscaling as is necessary for MIMO-VRN. (3) MIMO-VRN has better LR/HR quality while LSTM-VRN has more consistent LR/HR quality temporally. They use depends on the complexity constraints and performance requirements of the application.

5. Conclusion

This work presents two joint optimization approaches to video rescaling. Both incorporate an invertible network with coupling layer architectures to model explicitly the high-frequency component inherent in the HR video. While our LSTM-VRN shows that the temporal information in the LR video can be utilized to good advantage for better upscaling, our MIMO-VRN demonstrates that the GoF-based rescaling is able to make full use of temporal information to benefit both upscaling and downscaling. Our models demonstrate superior quantitative and qualitative performance to the image-based invertible model. They outperform, by a significant margin, the video rescaling framework without joint optimization.

Acknowledgements. This work is supported by Qualcomm technologies, Inc. (NAT-439543), Ministry of Science and Technology, Taiwan (109-2634-F-009-020) and National Center for High-performance Computing, Taiwan.

References

[1] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[2] Yuzhao Chen, Xi Xiao, Tao Dai, and Shu-Tao Xia. Hrnet: Hamiltonian rescaling network for image downscaling. In *IEEE International Conference on Image Processing (ICIP)*, 2020.

[3] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[4] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

[5] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2017.

[6] Yong Guo, Jian Chen, Jingdong Wang, Qi Chen, Jiezhang Cao, Zeshuai Deng, Yanwu Xu, and Mingkui Tan. Closed-loop matters: Dual regression networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[8] Takashi Isobe, Xu Jia, Shuhang Gu, Songjiang Li, Shengjin Wang, and Qi Tian. Video super-resolution with recurrent structure-detail network. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020.

[9] Takashi Isobe, Songjiang Li, Xu Jia, Shanxin Yuan, Gregory Slabaugh, Chunjing Xu, Ya-Li Li, Shengjin Wang, and Qi Tian. Video super-resolution with temporal group attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[10] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[11] Heewon Kim, Myungsub Choi, Bee Lim, and Kyoung Mu Lee. Task-aware image downscaling. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

[13] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[14] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[15] Wenbo Li, Xin Tao, Taian Guo, Lu Qi, Jiangbo Lu, and Jiaya Jia. Mucan: Multi-correspondence aggregation network for video super-resolution. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020.

[16] Yue Li, Dong Liu, Houqiang Li, Li Li, Zhu Li, and Feng Wu. Learning a convolutional neural network for image compact-resolution. *IEEE Transactions on Image Processing (TIP)*, 2018.

[17] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[18] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2013.

[19] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[20] Claude Elwood Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 1949.

[21] Wanjie Sun and Zhenzhong Chen. Learned image downscaling for upscaling using content adaptive resampler. *IEEE Transactions on Image Processing (TIP)*, 2020.

[22] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[23] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally-deformable alignment network for video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[24] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[25] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *The European Conference on Computer Vision Workshops*, 2018.

[26] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and S Yu Philip. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[27] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 2004.

[28] Mingqing Xiao, Shuxin Zheng, Chang Liu, Yaolong Wang, Di He, Guolin Ke, Jiang Bian, Zhouchen Lin, and Tie-Yan

Liu. Invertible image rescaling. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020.

[29] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 2019.

[30] Peng Yi, Zhongyuan Wang, Kui Jiang, Junjun Jiang, and Jiayi Ma. Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2019.

[31] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018.