Learning Spatio-Temporal Downsampling for Effective Video Upscaling

Xiaoyu Xiang[1], Yapeng Tian[2], Vijay Rengaranjan[1], Lucas D. Young[1], Bo Zhu[1], and Rakesh Ranjan[1]

[1] Meta Reality Labs, [2] University of Texas at Dallas
{xiangxiaoyu,apvijay,bozhufrl,rakeshr}@fb.com
{tianyapeng92,lucasyoung482}@gmail.com

Abstract. Downsampling is one of the most basic image processing operations. Improper spatio-temporal downsampling applied on videos can cause aliasing issues such as moiré patterns in space and the wagon-wheel effect in time. Consequently, the inverse task of upscaling a low-resolution, low frame-rate video in space and time becomes a challenging ill-posed problem due to information loss and aliasing artifacts. In this paper, we aim to solve the space-time aliasing problem by learning a spatio-temporal downsampler. Towards this goal, we propose a neural network framework that jointly learns spatio-temporal downsampling and upsampling. It enables the downsampler to retain the key patterns of the original video and maximizes the reconstruction performance of the upsampler. To make the downsamping results compatible with popular image and video storage formats, the downsampling results are encoded to uint8 with a differentiable quantization layer. To fully utilize the space-time correspondences, we propose two novel modules for explicit temporal propagation and space-time feature rearrangement. Experimental results show that our proposed method significantly boosts the space-time reconstruction quality by preserving spatial textures and motion patterns in both downsampling and upscaling. Moreover, our framework enables a variety of applications, including arbitrary video resampling, blurry frame reconstruction, and efficient video storage.

Keywords: downsampling, anti-aliasing, video upscaling

1 Introduction

Resizing is one of the most commonly used operations in digital image processing. Due to the limit of available memory and transfer bandwidth in compact devices, *e.g.* mobile phones and glasses, the high resolutions, high frame rate videos captured by such devices trade off either spatial or temporal resolution [1, 82]. While nearest-neighbor downsampling is the standard operation to perform such reduction in space and time, it is not the best option: it folds over high-frequency information in the downsampled frequency domain, leading to aliasing as indicated by the Nyquist theorem [60]. One way to avoid aliasing is to deliberately smudge high-frequency information by allowing more space and time to capture

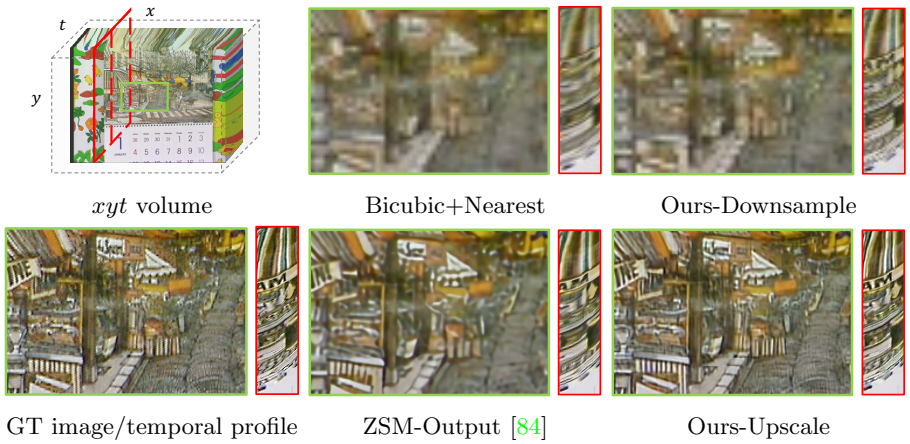

Fig. 1: *Effect of learned downsampling and upscaling for space-time reconstruction.* Compared to previous methods, the outputs of our learned downsampler maintain better spatial-temporal patterns, thus leading to more visually-appealing reconstruction results with better space-time consistency.

a single sample, *i.e.* by optical blur [26, 52] and motion blur [11, 31, 64], respectively. These spatial and temporal anti-aliasing filters band-limit frequencies, making it possible to reconstruct fine details during post-capture. Pre-designed anti-aliasing filters can be employed with the downsampler during capture time; for instance, using an optical low-pass filter [67] for spatial blur, and computational cameras such as the flutter shutter [51] camera for temporal blur.

To design the optimal filter for a specific task, it is natural to incorporate the downstream performance in the loop, where the weights of the downsampling filter are updated by the objective function of the task [69]. Kim *et al.* [28] inverts the super-resolution network as a downscaling encoder. Zhang *et al.* [91] proposes to add blur layers before each downsampling operation, and Zou *et al.* [94] adaptively predicts filter weights for each spatial location.

For the video restoration task, a major benefit of pre-designing the downsampler is to allow the co-design of an upsampler that recovers missing high-frequency details. While traditional methods mainly focus on upsampling – super-resolution (SR) in the case of space, and video frame interpolation (VFI) in the case of time – they assume the downsampler to be a trivial module. The low-resolution (LR) images for SR tasks are usually acquired by bicubic downsampling. For temporal reconstruction tasks like VFI, the low-fps (frames per second) frames are acquired by nearest-neighbor sampling, which keeps one frame per interval. Obviously, these operations are not optimal - the stride in time will lead to temporal aliasing artifacts. This independent tackling of the upsampling stage makes solving the inverse problem harder, and it is typical to employ heavy priors on texture and motion, which results in hallucination of lost details. On

the other hand, jointly handling upsampling together with downsampling would enable better performance in retaining and recovering spatio-temporal details. In this paper, we explore the design of a joint framework through simultaneous learning of a downsampler and an upsampler that effectively captures and reconstructs high-frequency details in both space and time.

Based on the above observations, we propose a unified framework that jointly learns spatio-temporal downsampling and upsampling, which works like an auto-encoder for low-fps, low-resolution frames. To handle the ill-posed space-time video super-resolution problem, we first make the downsampler to find the optimal representation in the low-resolution, low-fps domain that maximizes the restoration performance. Moreover, considering the downsampled representation should be stored and transmitted in the common image and video data formats, we quantize them to be uint8 with a differentiable quantization layer that enables end-to-end training. Finally, the downsampled frames are upscaled by our upsampler. To improve the reconstruction capability, we devise space-time pixel-shuffle and deformable temporal propagation modules to better exploit the space-time correspondences.

The main contributions of our paper are summarized as follows: (1) We provide a new perspective for space-time video downsampling by learning it jointly with upsampling, which preserves better space-time patterns and boosts restoration performance. (2) We observe that naive 3D convolution cannot achieve high reconstruction performance, and hence, we propose the deformable temporal propagation and space-time pixel-shuffle modules to realize a highly effective model design. (3) Our proposed framework exhibits great potential to inspire the community. We discuss the following applications: video resampling with arbitrary ratio, blurry frame reconstruction, and efficient video storage.

2 Related Works

2.1 Video Downsampling

Spatial. Spatial downsampling is a long-standing research problem in image processing. Classical approaches, such as the box, nearest, bicubic, and Lasnczos [13, 27, 53], usually design image filters to generate low resolution (LR) images by removing high frequencies and mitigating aliasing. Since most visual details exist in the high frequencies, they are also removed during downsampling. To address the problem, a series of structure and detail-preserving image downscaling methods [30, 48, 79] are proposed. Although these approaches can produce visually appealing LR images, they cannot guarantee that upscaling methods can restore the original high resolution (HR) images due to aliasing and non-uniform structure deformation in the downscaled LR images. To boost the restoration quality, [66, 80] proposed to learn a downsampler network. Pioneering research [73, 74] demonstrates that it is possible to design filters that allow for reconstructing the high-resolution input image with minimum error. The key is to add a small amount of optical blurring before sampling. Inspired

by this, we propose to automatically learn the best blurring filters during downsampling for more effective visual upscaling.

Temporal. A simple way to downsample along the temporal dimension is to increase the exposure time of a frame and capture the scene motion via blur. The loss of texture due to averaging is traded off for the ability to embed motion information in a single frame. Blur-to-video methods [2, 22, 23, 50, 62, 90] leverage motion blur and recover the image sequence, constraining the optimization with spatial sharpness and temporal smoothness. To regularize the loss of texture during capture, Yuan *et al*. [88] use a long and short exposure pair for deblurring, while Rengarajan *et al*. [54] exploit the idea to reconstruct high-speed videos. Coded exposure methods replace the box-filter averaging over time with a broadband filter averaging by switching the shutter on and off multiple times with varying on-off durations within a single exposure period. This results in better reconstruction owing to the preservation of high-frequency details over space. Raskar *et al*. [51] use such a coded exposure camera for deblurring, while Holloway *et al*. [18] recover a high-speed video from a coded low frame rate video. Our work contributes an extension of this previous work by learning the optimal temporal filters during downsampling for restoring sharp high-fps videos.

2.2 Video Upscaling

Video Super-Resolution. The goal of video super-resolution (VSR) is to restore HR video frames from their LR counterparts. Due to the existence of visual motion, the core problem to solve in VSR is how to temporally align neighboring LR frames with the reference LR frame. Optical flow methods seek to compute local pixel shifts and capture motions. Thus, a range of VSR approaches [7, 17, 58, 70, 76, 87] use optical flow to estimate motion and then perform motion compensation with warping. However, optical flow is generally limited in handling large motions, and flow warping can introduce artifacts into aligned frames. To avoid computing optical flow, implicit temporal alignment approaches, such dynamic upsampling filters [24], recurrent propagation [19, 20, 32, 34, 35], and deformable alignment [8, 9, 71, 77] are utilized to handle complex motions.

Video Frame Interpolation. Video frame interpolation (VFI) aims to synthesize intermediate video frames in between the original frames and upscale the temporal resolution of videos. Meyer *et al*. [40] utilizes phase information to assist frame interpolation. [25,39] proposed an encoder-decoder framework to directly predict intermediate video frames. Niklaus *et al*. [45,46] utilizes a spatially-adaptive convolution kernel for each pixel for synthesizing missing frames. Similar to VSR, optical flow is also adopted in VFI approaches [3,4,21,38,44,65] to explicitly handle motions.

Space-Time Video Super-resolution The pioneering work to extend SR to both space and time domains was proposed by Shechtman *et al*. [61]. Compared with VSR and VFI tasks, STVSR is even more ill-posed since pixels are missing along both spatial and temporal axes. To constrain the problem, early approaches [42, 59, 61, 68] usually exploited space-time local smoothness priors.

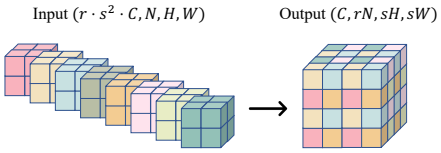

Fig. 2: *Space-Time Pixel-Shuffle.* It rearranges elements with shape $(r \cdot s^2 \cdot C, N, H, W)$ into the shape (C, rN, sH, sW), which enables efficient sub-pixel convolutions in space and time (xyt) dimensions.

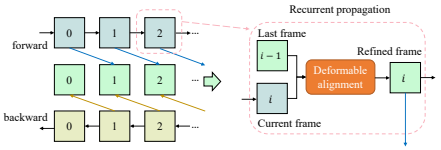

Fig. 3: *Deformable temporal modeling* with recurrent temporal propagation: at each step, we model the correspondence between the last output and the current input with deformable alignment to refine the current frame.

Very recently, deep learning-based STVSR frameworks [14, 16, 29, 83–86] have been developed. These methods directly interpolate the missing frame features and then upsample all frames in space. Like the video frame-interpolation methods, the input frames are anchors of timestamps, which limits the upscale ratio in the temporal dimension. Unlike these methods, we aim to freely resize the space-time volume with arbitrary scale ratios in this work.

3 Space-Time Anti-Aliasing (STAA)

We first explain the intuition behind STAA: treating video as spatio-temporal xyt volume and leveraging the characteristic spatio-temporal patterns for video reconstruction. Towards this end, we propose efficiently utilizing the spatio-temporal patterns in upscaling with space-time pixel-shuffle. However, naively regarding time as an additional dimension beyond space has a limitation: the pixel (x, y) at the i-th frame is usually not related to the same (x, y) at the $i + k$-th frame. To tackle this problem, we model the temporal correspondences with deformable sampling.

3.1 Intuition

To tackle the aliasing problem, previous methods insert low-pass filters either in space or time. However, we noticed that the space and time dimensions of an xyt volume are not independent: as shown in Fig. 1, the temporal profiles xt and yt display similar patterns as the spatial patches. In an ideal case where the 2D objects move with a constant velocity, the temporal profile will appear as a downsampled version of the object [49, 59], as illustrated in Fig. 5. This space-time patch recurrence makes it possible to aid the reconstruction of the under-sampled dimension with abundant information from other dimensions. Based upon this observation, we adopt a 3D low-pass filter on both space and time to better utilize the correspondences across dimensions. Accordingly, our upsampler network also adopts the 3D convolutional layers as the basic building block due to its capability of jointly handling the spatio-temporal features.

3.2 Module Design

Space-Time Pixel-Shuffle. Pixel-shuffle [63] is a widely-used operation in single image super-resolution (SISR) for efficient subpixel convolution. It has two advantages: the learned upscaling filter can achieve the optimal performance; computational complexity is reduced by rearranging the elements in LR feature maps. Inspired by its success in SISR, we extend it to space-time. Fig. 2 illustrates the shuffling operation: for an input tensor with shape $(r \cdot s^2 \cdot C, N, H, W)$, the elements are shuffled periodically into the shape (C, rN, sH, sW).

Naive Deconvolution Is Insufficient. If we simply regard the space-time upscaling as the reverse process of the downsampling, then conceptually, a deconvolution should be enough to handle this process. To investigate this idea, we build a small network using 3D convolutions and the aforementioned space-time pixel-shuffle layers with a style of ESPCN [63]. Although this network does converge, it only improves the PSNR by ~ 0.5 dB compared with trilinear upscaling the xyt cube – such improvement is too trivial to be considered effective, particularly when compared to the success of ESPCN in SISR.

Enhance Temporal Modeling Capacity. As noted above, this suboptimal result was expected due to the lost correspondences between the i-th and $i+k$-th frames. Thus, 3D convolution alone cannot guarantee a good reconstruction performance due to its relatively small field of view. Understanding "what went where" [81] is the fundamental problem for video tasks. Such correspondence is even more critical in our framework: our STAA downsampler encodes the motion by dispersing the space feature along the temporal dimension. Correspondingly, during the reconstruction stage, the supporting information can come from neighboring frames. Motivated by this, we devise a deformable module to build temporal correspondences and enhance the model's capability to handle dynamic scenes: for a frame at time i, it should look at adjoining $i-k, \ldots i+k$ frames and refine the current feature by aggregating the relevance. For efficient implementation, we split the information propagation into forward and backward directions, where the temporal correspondence is built and passed recurrently per direction, as shown in Fig. 3. The refined features from both forward and backward passes are aggregated to yield the output. Hence, the difficulty of perceiving long-range information within the 3D convolutional receptive field is alleviated.

4 Joint Downscaling and Upscaling Framework

Our framework architecture is shown in Fig. 4: given a sequence of video frames $V = \{I_i\}_1^{rN}$ where each I_i is an RGB image of dimensions $sW \times sH$ (r and s as scale factors), our goal is to design (a) a downsampler which would produce $V_\downarrow = \{D(I_i)\}_1^N$ where each $D(I_i)$ has the dimensions $W \times H$, and, (b) an upsampler which would produce $\widetilde{V} = \{U(D(I_i))\}_1^{rN}$ where, in the perfect case, $V = \widetilde{V}$.

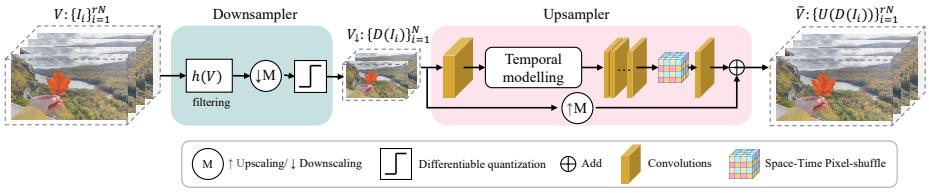

Fig. 4: Our training framework functions as an auto-encoder in which we train the downsampler D (encoder), and the upsampler U (decoder) jointly in an end-to-end manner.

4.1 Downsampler

Our downsampler consists of a 3D low pass filter $h(\cdot)$ followed by a downsampling operation by striding. Given a sequence of input frames $V = \{I_i\}_1^{rN}$ that needs to be downsampled, we first convolve it with the filter h as follows: $h(V)[t, x, y] = \sum_{i,j,k \in \Omega} h[i, j, k] \cdot V[t - i, x - j, y - k]$.

To ensure that the learned filters are low-pass, we add a softmax layer to regularize the weight values within the range $[0, 1]$ and the sum to be 1. We then use striding to produce our desired downsampled frames in both space and time. An ideal anti-aliasing filter should restrict the bandwidth to satisfy the Nyquist theorem without distorting the in-band frequencies.

Analysis of learned filters. We present a study based on the frequency domain analysis of spatio-temporal images to compare different types of low-pass filters. We analyze the canonical case of a single object moving with uniform velocity. The basic setup is shown in the first column of Fig. 5, where the top row shows the static object and the bottom row shows the xyt volume corresponding to the motion. Figs. 5(a) to (f) show the temporal profiles xt corresponding to the 1D scanline (marked in red) for various scenarios/filters in the top row and their corresponding Fourier domain plots in the bottom row. Please check the supplementary material for more details about how the Fourier plots are calculated and what the spectra components represent.

In Fig. 5, (a) is a space-time diagram for a static scene (zero velocity), so there is no change along the time (vertical) dimension. In (b), we can see that the motion causes a time-varying effect, which results in shear along the spatial x direction. This shows the coupling of spatial and temporal dimensions. Applying just the nearest-neighbor downsampling in time leads to severe aliasing, as shown by duplication of streaks in (c) bottom row. Thus, the plain downsampling method leads to temporal pattern distortions. Characteristic spatio-temporal patterns relate to events in the video [10, 47, 89]. Thus, good downsampling methods should also retain the "textures" in time dimension.

Figs. 5(d), (e), and (f) show the images and frequency plots for the case of applying Gaussian, box, and our STAA low pass filters, respectively, first, followed by nearest-neighbor downsampling. The convolution with these filters causes blurring across space and time dimensions, as shown in the images, and

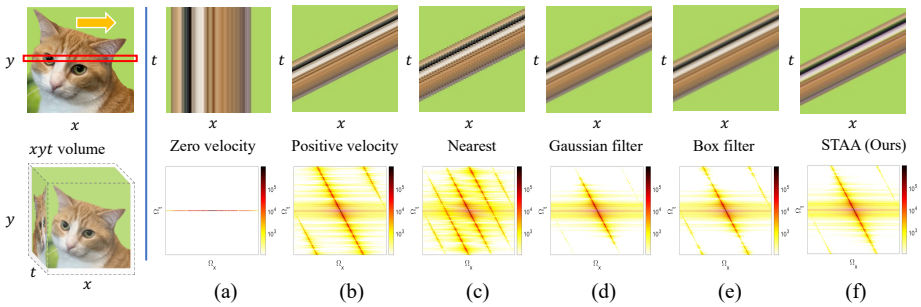

Fig. 5: Space-time and Fourier domain plots for a moving object.

since convolution corresponds to frequency domain multiplication, we can see the benefits of these filters visually in the frequency domain plots. The Gaussian filter reduces the subbands and high spatiotemporal frequencies in (d). The Box filter along the temporal dimension, which is used to describe the motion blur [6, 15, 56], destroys spatial details and attenuates certain temporal frequencies, which causes post-aliasing [41] during reconstruction. Our proposed STAA filter attenuates high frequencies and the subbands like the Gaussian filter does, but at the same time, it preserves more energy in the main spectra component, as shown in (f). This characteristic ensures the prefiltered image maintains a good spatio-temporal texture, thus benefiting the reconstruction process.

Connection to coded exposure. Traditional motion blur caused by long exposure can be regarded as filtering with a temporal box filter. While long exposure acts as a natural form of filtering during the capture time itself, the box filter is not the best filter for alias-free reconstruction [41]. Hence, coded exposure methods "flutter" the shutter in a designed sequence to record the motion without completely smearing the object across the spatial dimension to recover sharp high-frame-rate images and videos [18, 51]. Our learned downsampler can be regarded as a learned form of the coded exposure: considering the temporal kernel size as an exposure window, we aggregate the pixels at each time step to preserve an optimal space-time pattern for better reconstruction.

Differentiable quantization layer. The direct output of our downsampler is a floating-point tensor, while in practical applications, images are usually encoded as 8-bit RGB (uint8) format. Quantization is needed to make our downsampled frames compatible with popular image storage and transmission pipelines. However, this operation is not differentiable. The gap between float and discrete integer causes training unstable and a drop in performance. To bridge the gap, we adopt a differentiable quantization layer that enables end-to-end training [5]. More details can be found in *supplementary material*.

4.2 Upsampler

Given a sequence of downsampled frames, $V_{\downarrow} = \{D(I_i)\}_{i=1}^{N}$, the upsampler U aims to increase the resolution in both space and time. The estimated upscaled video $\widetilde{V} = \{U(D(I_i)\}_{i=1}^{rN}$ should be as close to the original input as possible.

To achieve this purpose, we choose 3D convolution as our basic building block for the upsampler. The input sequence is converted to the feature domain $\mathcal{F}$ by a 3D convolution. We adopt a deformable temporal modeling (DTM) subnetwork to aggregate the long-range dependencies recurrently. It takes the last aggregated frame feature $DTM(f_{i-1})$ at time step $i-1$ and the current feature f_i as inputs, outputting the current aggregated feature:

$$DTM(f_i) = T(f_i, DTM(f_{i-1})), \tag{1}$$

where f_i is the frame feature at time step i, and T denotes a general function that finds and aligns the corresponding information to the current feature. We adopt the deformable sampling function [12,93] as T to capture such correspondences. To fully exploit the temporal information, we implement a bidirectional DTM that aggregates the refined features from both forward and backward passes.

The refined sequence is then passed to the reconstruction module that is composed of 3D convolutions. To fully explore the hierarchical features from these convolutional layers, we organize them into residual dense blocks [92]. It densely connects the 3D convolution layers into local groups and fuses the features of different layers. Following the previous super-resolution networks [36], no BatchNorm layer is used in our reconstruction module. Finally, a space-time pixel-shuffle layer is adopted to rearrange the features with a periodic shuffling across the xyt volume [57].

We denote the output just after the space-time pixel-shuffle as $F(V_{\downarrow})$, where $F(\cdot)$ is all the previous operations for upscaling the input $V_{\downarrow}$. To help the main network focus on generating high-frequency information, we bilinearly upscale the input sequence and add it to the reconstructed features as the final output:

$$U(V_{\downarrow}) = V_{\downarrow} \uparrow_M + F(V_{\downarrow}). \tag{2}$$

This long-range skip-connection allows the low-frequency information of the input to bypass the major network and makes the major part of the network predict the residue. It "lower-bounds" the reconstruction performance and increases the convergence speed of the network.

5 Experiments

We use the Peak Signal-to-Noise-Ratio (PSNR) and Structural Similarity Index (SSIM) [78] metrics to evaluate video restoration. We also compare the number of parameters (million, M) to evaluate model efficiency. Please find our datasets and implementation details in the *supplementary materials*.

Table 1: Comparisons with SOTA cascaded video frame interpolation (VFI) and super-resolution (VSR), and space-time super-resolution (STVSR) methods.

Upscale rate time/space	Downsampler time/space	Reconstruction Method VFI	VSR	Params/M	Vimeo-90k PSNR	Vimeo-90k SSIM	Vid4 PSNR	Vid4 SSIM
2×/1×	Nearest/-	XVFI [65]	-	5.7	34.76	0.9532	29.21	0.9496
		FLAVR [25]	-	42.1	36.73	0.9632	29.83	0.9585
	STAA	Ours		15.9	**45.01**	**0.9912**	**39.78**	**0.9926**
1×/4×	-/Bicubic	-	BasicVSR++ [9]	7.3	35.91	0.9383	26.24	0.8214
		-	VRT [33]	35.6	36.35	0.9420	26.39	0.8248
		-	RVRT [34]	10.8	36.30	0.9417	26.44	0.8285
	STAA	Ours		15.9	**37.35**	**0.9629**	**30.10**	**0.9517**
2×/4×	Nearest/Bicubic	XVFI [65]	BasicVSR++ [9]	5.7+7.3	32.41	0.9123	24.90	0.7726
		FLAVR [25]	BasicVSR++ [9]	42.1+7.3	32.74	0.9119	24.79	0.7678
		ZSM [83]		11.1	33.48	0.9178	24.82	0.7763
		TMNet [86]		12.3	33.66	0.9200	24.90	0.7803
		STDAN [75]		8.3	33.59	0.9192	24.91	0.7832
	STAA	Ours		16.0	**34.53**	**0.9426**	**27.31**	**0.9173**

5.1 Comparison with State-of-the-Art Methods

We compare the performance of reconstructing a video in space and time with SOTA VFI, VSR and STVSR methods. For two input frames, previous VFI methods generate one interpolated frame along with the two inputs, while our STAA generates four upsampled ones. For an apples-to-apples comparison, we only calculate the PSNR/SSIM of the synthesized frames. Quantitative results on Vimeo-90k [87] and Vid4 [37] are shown in Tab. 1.

Our method outperforms the previous methods by a large margin on all datasets and settings. For temporal upscaling, adopting the STAA downsampling and upscaling exceeds the second-best method by 8.28 dB on Vimeo-90k and 9.95 dB on Vid4, which validates the importance of anti-aliasing in the temporal dimension. For $s \times 4$ spatial upscaling, the STAA pipeline exceeds the SOTA VSR method under bicubic degradation by 1 dB on Vimeo-90k. For the challenging case of 4× space/2× time, our method still demonstrates remarkable improvement by 2.4 dB on the Vid4 and more than 1 dB on the Vimeo-90k datasets. Such significant improvement brought by the co-design of downsampling filter and upscaling network provides a new possibility for improving current video restoration methods.

5.2 Ablation Studies

Downsampler. To verify the effectiveness of our learned downsampling filter, we compare the reconstruction performance by switching downsamplers. Since the reconstruction capability of the same upsampler architecture is unchanged, a better reconstruction result means that the downsampler produces a better space-time representation. We compare our method with other learned downsampling networks: CAR [66] and PASA [94] (see Tab. 2). Since the existing methods only perform spatial down+upsampling learning, we set $t = 1$ and

Table 2: Comparison of spatial downsamplers (1×t, 4×s).

Downsampler	Params/M	GFLOPs/MP	PSNR	SSIM
CAR [66]	9.896	2305.77	35.96	0.9400
PASA [94]	0.003	6.144	35.37	0.9524
Ours	**0.002**	**0.081**	**37.35**	**0.9629**

Table 3: Quantitative comparison of downsampling filters ($2 \times t$, $2 \times s$). The best two results are highlighted in red and blue, respectively.

Time	Space	PSNR	SSIM
Nearest	Bicubic	28.88	0.9073
Gaussian		37.44	0.9679
STAA$_{no}$		39.44	0.9775
STAA$_{soft}$		40.40	0.9812
STAA$_{quant}$		40.42	0.9811
STAA$_{ada}$		38.13	0.9720

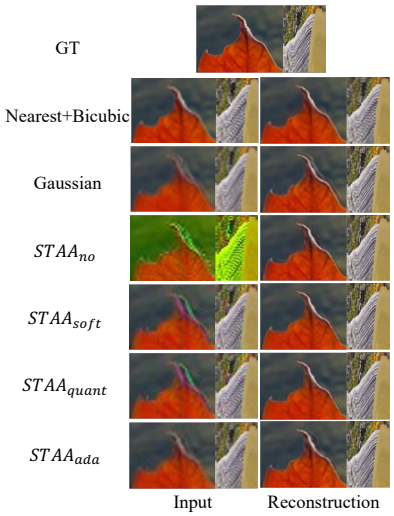

Fig. 6: Visual comparison at $2 \times t/2 \times s$ setting in the left table.

$s = 4$. Our STAA filter has significantly fewer number of parameters and computational cost compared with the other two methods while demonstrating better performance in terms of reconstruction PSNR and SSIM.

We compare with other 3d downsampling filters in Tab. 3. The nearest-bicubic downsampling, which is adopted by previous video reconstruction tasks, provides the worst representation among all. For the $2 \times t/2 \times s$ setting, the reconstruction network cannot converge to global optimal. Although it is still the dominant setting, it cannot handle the temporal-aliasing issue and might hinder the development of video reconstruction methods. Pre-filtering with a 3D Gaussian blur kernel can alleviate the aliasing problem, which exceeds the nearest-neighbor downsampling in time and the bicubic downsampling in space. Still, the Gaussian filter cannot produce the optimal spatio-temporal textures. Compared with these classical methods, our STAA filters improve the reconstruction performance by a large margin, as shown in the last four rows. We believe that our proposed STAA downsampler has the potential to serve as a new benchmark for video reconstruction method design and inspire the community from multiple perspectives.

We visualize the downsampled frames and their corresponding reconstruction results in Fig. 6. For nearest-bicubic downsampled results, the temporal profile has severe aliasing. In comparison, the anti-aliasing filters make the downsampled frames "blurry" to embed the motion information.

Constraints of the encoded frames. The classical downsampling filters, *e.g.*, nearest and Gaussian, can generate downsampled frames that resemble the input's appearance. However, in our auto-encoder framework, there is actually no

Table 4: Ablation of upsampler design.

Naive 3D Conv	3D RDB	DTM	Params/M	GFlops/MP	PSNR	SSIM
✓			0.3	3.2	28.67	0.8536
	✓		11.0	114.4	31.35	0.9016
		✓	5.3	51.7	31.55	0.9032
	✓	✓	16.0	164.7	32.00	0.9109

Table 5: Upsampling methods.

Method	3D deconv	Up+3Dconv	ST-pixelshuffle
PSNR	34.53	31.69	**34.56**
SSIM	0.9409	0.9025	**0.9413**

guarantee that the encoded frames look like the original input. A straightforward way is to use the classical downsampled results as supervision, but it might impede the downsampler from learning the optimal spatio-temporal representation. So we turn to regularize the downsampling filter with following experiments: (1) *no*: no constraints; (2) *soft*: use the softmax to regularize the weights; (3) *quant*: add the differentiable quantization layer; (4)*ada*: dynamically generate filters for each spatial location according to the input content (also with softmax).

From the last four rows of Tab. 3, all STAA filters outperform the classical ones for reconstruction. The filter without any constraint is not necessary to be low-pass. Besides, it may cause color shifts in the encoded frames. Constraining the filter weights with softmax can alleviate color shifts and improve the reconstruction results due to anti-aliasing. Still, the moving regions are encoded as the color difference. Adding the quantization layer does not cause performance degradation, which validates the effectiveness of our differentiable implementation. Making the filter weights conditioned on the input content creates visually pleasing LR frames. However, the reconstruction performance degrades, probably because the changing weights of the downsampling filter confuse the upsampler.

Comparing different downsampling settings, we observe that our STAA is more robust to temporal downsampling than previous methods. Specifically, the reconstruction quality is correlated to the logarithm of the percentage of pixels in the downsampling representation. More discussions are in our *Appendix*.

Effectiveness of proposed modules. In Tab. 4, we compare the video reconstruction results and the computational cost with different modules of the upsampler. We check the FLOPs per million pixels (MP) using the open-source tool fvcore [55]. From the first row, we can observe that naive 3D convolution performs bad. Changing it to a more complex 3D residual-dense block (RDB) improves the performance by 2.68 dB, with a rapid increase of the computational cost. Although this network still cannot explicitly find the temporal correspondence, the deeper structure enlarges the perceiving area, thus enabling capturing dependencies with large displacement. In the third row, adopting deformable temporal modeling (DTM) shows a great performance improvement with relatively low computational cost, which validates the importance of aggregating the displaced information across space and time. Such spatio-temporal aggregated features can be effectively utilized by the 3D CNN, resulting in improved PSNR and SSIM results (see the last row).

In Tab. 5, we compare our space-time pixel-shuffle (ST-pixelshuffle) with two other upscaling methods: 3D deconvolution, and trilinear upscaling + 3D convolution. Our proposed space-time pixel-shuffle achieves the best performance in terms of PSNR and SSIM.

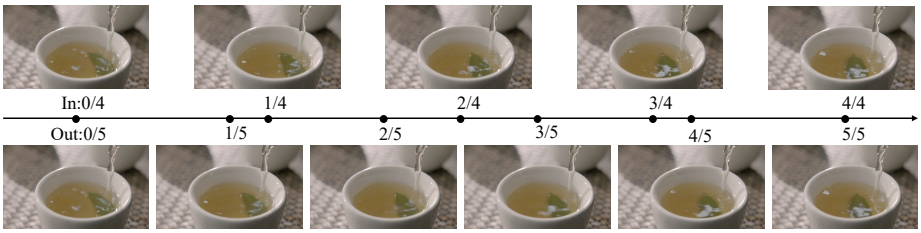

Fig. 7: Our methods enable smooth frame rate conversion with arbitrary rates, *e.g.* 20 fps (top row) to 24 fps (bottom row). We plot a timeline in the middle and mark the timestamp of each frame. The generated frames have natural motion transition and vivid textures, *e.g.* water flow, reflection, and refraction.

5.3 Applications

The STAA downsampler can be used to reduce the resolution and frame rate of a video for efficient video transmission. The learned upsampler can also be applied to process natural videos with a simple modification. Besides, our upsampler network can also reconstruct crisp clean frames from a blurry sequence.

Video Resampling. The space-time pixel-shuffle module makes it possible to change the frame rate with arbitrary ratios while keeping the motion patterns. Previous VFI methods can only synthesize new frames in-between two inputs. Anchored by the input timestamps, their scale ratio can only be integers. ffmpeg [72] change frame rate by dropping or duplication at a certain interval, which changes the motion pattern of the original timestamps and cannot generate smooth results. Another option is to use frame blending to map the intermediate motion between keyframes while creating fuzzy and ghosting artifacts. Some softwares adopt optical flow warping, which can synthesize better results than the above two methods. Still, it cannot handle large motions or morph.

Our upsampler can maintain the space-time patterns when upscaling the temporal dimension at any given ratio: we show an example of converting 20 fps to 24 fps ($1.2\times t$) in Fig. 7, which does synthesize the correct motion at the non-existent time steps, leading to smoother visual results. Our temporal modeling module can map long-range dependencies among the input frames, and together with the space-time convolutional layers, can reconstruct sharp and crisp frames.

Blurry Frame Reconstruction. As discussed in Sec 4.1, motion blur is a temporal low-pass filter. It is a real-world case of our STAA filter: the temporal kernel size is the exposure time window, and the weights at each time step are equal. Hence, there is a good reason to believe that our designed upsampler can be applied on blurry frame reconstruction, which turns the low-resolution blurry sequence into a high frame-rate and high-resolution clean sequence. We trained our upsampler with a $4\times s, 2\times t$ upscale setting using the REDS-blur [43] data. We show the restoration images in Fig. 8. Even when the motion is rather

Fig. 8: Our upsampler can also be used for reconstructing sharp and crisp details from videos with motion-blur. The left column shows the overlaid two LR blurry inputs, and the right four columns are our reconstruction results with $2\times$ in time and $4\times$ in space, which recover shapes and textures from motion blur.

large and the object texture is badly smeared, our upsampler does a good job in reconstructing the shape and structures at the correct timestep.

Efficient Video Storage. Since the downsampler output is still in the same color space and data type (*e.g.* 8-bit RGB) as the input, it can be processed by any existing encoding and compression algorithms for storage and transmission without extra elaborations. Especially, the downsampled frames still preserve the temporal connections implying their compatibility with video codecs.

6 Conclusions

In this paper, we propose to learn a space-time downsampler and upsampler jointly to optimize the intermediate downsampled representations and ultimately boost video reconstruction performance. The downsampler includes a learned 3D low-pass filter for spatio-temporal anti-aliasing and a differentiable quantization layer ensuring the downsampled frames are encoded in uint8. For the upsampler, we propose the space-time pixel-shuffle to enable upscaling the xyt volume at any given ratio. We further exploit the temporal correspondences between consecutive frames by explicit temporal modeling. Due to the advantages of these designs, our framework outperforms state-of-the-art works in VSR and VFI by a large margin. Moreover, we demonstrate that our proposed upsampler can be used for highly accurate arbitrary frame-rate conversion, generating high-fidelity motion and visual details at the new timestamps for the first time. Our network can also be applied to blurry frame reconstruction and efficient video storage. We believe that our approach provides a new perspective on space-time video super-resolution tasks and has a broad potential to inspire novel methods for future works such as quantization-aware image/video reconstruction, restoration-oriented video compression, and hardware applications such as coded exposure and optical anti-aliasing filter.

References

1. Allebach, J., Wong, P.W.: Edge-directed interpolation. In: IEEE International Conference on Image Processing. vol. 3, pp. 707–710. IEEE (1996)
2. Argaw, D.M., Kim, J., Rameau, F., Kweon, I.S.: Motion-blurred video interpolation and extrapolation. In: AAAI Conference on Artificial Intelligence (2021)
3. Bao, W., Lai, W.S., Ma, C., Zhang, X., Gao, Z., Yang, M.H.: Depth-aware video frame interpolation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3703–3712 (2019)
4. Bao, W., Lai, W.S., Zhang, X., Gao, Z., Yang, M.H.: Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
5. Bengio, Y., Léonard, N., Courville, A.: Estimating or propagating gradients through stochastic neurons for conditional computation. arXiv preprint arXiv:1308.3432 (2013)
6. Brooks, T., Barron, J.T.: Learning to synthesize motion blur. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 6840–6848 (2019)
7. Caballero, J., Ledig, C., Aitken, A., Acosta, A., Totz, J., Wang, Z., Shi, W.: Real-time video super-resolution with spatio-temporal networks and motion compensation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 4778–4787 (2017)
8. Chan, K.C., Wang, X., Yu, K., Dong, C., Loy, C.C.: Understanding deformable alignment in video super-resolution. arXiv preprint arXiv:2009.07265 **4**(3), 4 (2020)
9. Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. arXiv preprint arXiv:2104.13371 (2021)
10. Cooper, M., Liu, T., Rieffel, E.: Video segmentation via temporal pattern classification. IEEE Transactions on Multimedia **9**(3), 610–618 (2007)
11. Dachille, F., Kaufman, A.: High-degree temporal antialiasing. In: Proceedings Computer Animation. pp. 49–54. IEEE (2000)
12. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: IEEE International Conference on Computer Vision. pp. 764–773 (2017)
13. Duchon, C.E.: Lanczos filtering in one and two dimensions. Journal of Applied Meteorology and Climatology **18**(8), 1016–1022 (1979)
14. Dutta, S., Shah, N.A., Mittal, A.: Efficient space-time video super resolution using low-resolution flow and mask upsampling. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 314–323 (2021)
15. Egan, K., Tseng, Y.T., Holzschuch, N., Durand, F., Ramamoorthi, R.: Frequency analysis and sheared reconstruction for rendering motion blur. ACM Transactions on Graphics **28**(3), 93–1 (2009)
16. Haris, M., Shakhnarovich, G., Ukita, N.: Space-time-aware multi-resolution video enhancement. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2859–2868 (2020)
17. Haris, M., Shakhnarovich, G., Ukita, N.: Recurrent back-projection network for video super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3897–3906 (2019)

18. Holloway, J., Sankaranarayanan, A.C., Veeraraghavan, A., Tambe, S.: Flutter shutter video camera for compressive sensing of videos. In: International Conference on Computational Photography. pp. 1–9. IEEE (2012)
19. Huang, Y., Wang, W., Wang, L.: Video super-resolution via bidirectional recurrent convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(4), 1015–1028 (2017)
20. Isobe, T., Jia, X., Gu, S., Li, S., Wang, S., Tian, Q.: Video super-resolution with recurrent structure-detail network. In: European Conference on Computer Vision. pp. 645–660. Springer (2020)
21. Jiang, H., Sun, D., Jampani, V., Yang, M.H., Learned-Miller, E., Kautz, J.: Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 9000–9008 (2018)
22. Jin, M., Hu, Z., Favaro, P.: Learning to extract flawless slow motion from blurry videos. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 8112–8121 (2019)
23. Jin, M., Meishvili, G., Favaro, P.: Learning to extract a video sequence from a single motion-blurred image. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 6334–6342 (2018)
24. Jo, Y., Wug Oh, S., Kang, J., Joo Kim, S.: Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3224–3232 (2018)
25. Kalluri, T., Pathak, D., Chandraker, M., Tran, D.: Flavr: Flow-agnostic video representations for fast frame interpolation. arXiv preprint arXiv:2012.08512 (2020)
26. Keelan, B.: Handbook of image quality: characterization and prediction. CRC Press (2002)
27. Keys, R.: Cubic convolution interpolation for digital image processing. IEEE Transactions on Acoustics, Speech, and Signal Processing **29**(6), 1153–1160 (1981)
28. Kim, H., Choi, M., Lim, B., Lee, K.M.: Task-aware image downscaling. In: European Conference on Computer Vision. pp. 399–414 (2018)
29. Kim, S.Y., Oh, J., Kim, M.: Fisr: Deep joint frame interpolation and super-resolution with a multi-scale temporal loss. In: AAAI Conference on Artificial Intelligence. pp. 11278–11286 (2020)
30. Kopf, J., Shamir, A., Peers, P.: Content-adaptive image downscaling. ACM Transactions on Graphics **32**(6), 1–8 (2013)
31. Korein, J., Badler, N.: Temporal anti-aliasing in computer generated animation. In: Annual Conference on Computer Graphics and Interactive Techniques. pp. 377–388 (1983)
32. Li, Y., Jin, P., Yang, F., Liu, C., Yang, M.H., Milanfar, P.: Comisr: Compression-informed video super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2543–2552 (2021)
33. Liang, J., Cao, J., Fan, Y., Zhang, K., Ranjan, R., Li, Y., Timofte, R., Van Gool, L.: Vrt: A video restoration transformer. arXiv preprint arXiv:2201.12288 (2022)
34. Liang, J., Fan, Y., Xiang, X., Ranjan, R., Ilg, E., Green, S., Cao, J., Zhang, K., Timofte, R., Van Gool, L.: Recurrent video restoration transformer with guided deformable attention. arXiv preprint arXiv:2206.02146 (2022)
35. Lim, B., Lee, K.M.: Deep recurrent resnet for video super-resolution. In: Asia-Pacific Signal and Information Processing Association Annual Summit and Conference. pp. 1452–1455. IEEE (2017)

36. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 136–144 (2017)
37. Liu, C., Sun, D.: A bayesian approach to adaptive video super resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 209–216. IEEE (2011)
38. Liu, Z., Yeh, R.A., Tang, X., Liu, Y., Agarwala, A.: Video frame synthesis using deep voxel flow. In: IEEE International Conference on Computer Vision. pp. 4463–4471 (2017)
39. Long, G., Kneip, L., Alvarez, J.M., Li, H., Zhang, X., Yu, Q.: Learning image matching by simply watching video. In: European Conference on Computer Vision. pp. 434–450. Springer (2016)
40. Meyer, S., Wang, O., Zimmer, H., Grosse, M., Sorkine-Hornung, A.: Phase-based frame interpolation for video. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1410–1418 (2015)
41. Mitchell, D.P., Netravali, A.N.: Reconstruction filters in computer-graphics. ACM Siggraph Computer Graphics **22**(4), 221–228 (1988)
42. Mudenagudi, U., Banerjee, S., Kalra, P.K.: Space-time super-resolution using graph-cut optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5), 995–1008 (2010)
43. Nah, S., Baik, S., Hong, S., Moon, G., Son, S., Timofte, R., Lee, K.M.: Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (June 2019)
44. Niklaus, S., Liu, F.: Context-aware synthesis for video frame interpolation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1701–1710 (2018)
45. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive convolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 670–679 (2017)
46. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. In: IEEE International Conference on Computer Vision. pp. 261–270 (2017)
47. Niyogi, S.A., Adelson, E.H.: Analyzing gait with spatiotemporal surfaces. In: IEEE Workshop on Motion of Non-rigid and Articulated Objects. pp. 64–69. IEEE (1994)
48. Oeztireli, A.C., Gross, M.: Perceptually based downscaling of images. ACM Transactions on Graphics **34**(4), 1–10 (2015)
49. Pollak Zuckerman, L., Naor, E., Pisha, G., Bagon, S., Irani, M.: Across scales & across dimensions: Temporal super-resolution using deep internal learning. In: European Conference on Computer Vision. Springer (2020)
50. Purohit, K., Shah, A., Rajagopalan, A.: Bringing alive blurred moments. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 6830–6839 (2019)
51. Raskar, R., Agrawal, A., Tumblin, J.: Coded exposure photography: Motion deblurring using fluttered shutter. ACM Transactions on Graphics **25**(3), 795–804 (2006)
52. Ray, S.: Scientific photography and applied imaging. Routledge (1999)
53. Reinhard, E., Heidrich, W., Debevec, P., Pattanaik, S., Ward, G., Myszkowski, K.: High dynamic range imaging: acquisition, display, and image-based lighting. Morgan Kaufmann (2010)
54. Rengarajan, V., Zhao, S., Zhen, R., Glotzbach, J., Sheikh, H., Sankaranarayanan, A.C.: Photosequencing of motion blur using short and long exposures. In: IEEE

Conference on Computer Vision and Pattern Recognition Workshops. pp. 510–511 (2020)
55. Research, M.: fvcore. https://github.com/facebookresearch/fvcore (2019)
56. Rim, J., Kim, G., Kim, J., Lee, J., Lee, S., Cho, S.: Realistic blur synthesis for learning image deblurring. arXiv preprint arXiv:2202.08771 (2022)
57. Rogozhnikov, A.: Einops: Clear and reliable tensor manipulations with einstein-like notation. In: International Conference on Learning Representations (2021)
58. Sajjadi, M.S., Vemulapalli, R., Brown, M.: Frame-recurrent video super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 6626–6634 (2018)
59. Shahar, O., Faktor, A., Irani, M.: Space-time super-resolution from a single video. IEEE (2011)
60. Shannon, C.: Communication in the presence of noise. Proceedings of the IRE **37**(1), 10–21 (Jan 1949), https://doi.org/10.1109/jrproc.1949.232969
61. Shechtman, E., Caspi, Y., Irani, M.: Increasing space-time resolution in video. In: European Conference on Computer Vision. pp. 753–768. Springer (2002)
62. Shen, W., Bao, W., Zhai, G., Chen, L., Min, X., Gao, Z.: Blurry video frame interpolation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 5114–5123 (2020)
63. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1874–1883 (2016)
64. Shinya, M.: Spatial anti-aliasing for animation sequences with spatio-temporal filtering. In: Annual Conference on Computer Graphics and Interactive Techniques. pp. 289–296 (1993)
65. Sim, H., Oh, J., Kim, M.: Xvfi: extreme video frame interpolation. In: IEEE International Conference on Computer Vision (2021)
66. Sun, W., Chen, Z.: Learned image downscaling for upscaling using content adaptive resampler. IEEE Transactions on Image Processing **29**, 4027–4040 (2020)
67. Suzuki, T.: Optical low-pass filter (Jan 1987)
68. Takeda, H., Van Beek, P., Milanfar, P.: Spatiotemporal video upscaling using motion-assisted steering kernel (mask) regression. In: High-Quality Visual Experience, pp. 245–274. Springer (2010)
69. Talebi, H., Milanfar, P.: Learning to resize images for computer vision tasks. In: IEEE International Conference on Computer Vision. pp. 497–506 (October 2021)
70. Tao, X., Gao, H., Liao, R., Wang, J., Jia, J.: Detail-revealing deep video super-resolution. In: IEEE International Conference on Computer Vision. pp. 4472–4480 (2017)
71. Tian, Y., Zhang, Y., Fu, Y., Xu, C.: Tdan: Temporally-deformable alignment network for video super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3360–3369 (2020)
72. Tomar, S.: Converting video formats with ffmpeg. Linux Journal **2006**(146), 10 (2006)
73. Trentacoste, M., Mantiuk, R., Heidrich, W.: Blur-aware image downsampling. In: Computer Graphics Forum. vol. 30, pp. 573–582. Wiley Online Library (2011)
74. Triggs, B.: Empirical filter estimation for subpixel interpolation and matching. In: IEEE International Conference on Computer Vision. vol. 2, pp. 550–557. IEEE (2001)

75. Wang, H., Xiang, X., Tian, Y., Yang, W., Liao, Q.: Stdan: Deformable attention network for space-time video super-resolution. arXiv preprint arXiv:2203.06841 (2022)
76. Wang, L., Guo, Y., Lin, Z., Deng, X., An, W.: Learning for video super-resolution through hr optical flow estimation. In: Asian Conference on Computer Vision. pp. 514–529. Springer (2018)
77. Wang, X., Chan, K.C., Yu, K., Dong, C., Change Loy, C.: EDVR: Video restoration with enhanced deformable convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019)
78. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
79. Weber, N., Waechter, M., Amend, S.C., Guthe, S., Goesele, M.: Rapid, detail-preserving image downscaling. ACM Transactions on Graphics **35**(6), 1–6 (2016)
80. Wei, Y., Chen, L., Song, L.: Video compression based on jointly learned downsampling and super-resolution networks. In: 2021 International Conference on Visual Communications and Image Processing (VCIP). pp. 1–5. IEEE (2021)
81. Wills, J., Agarwal, S., Belongie, S.: What went where [motion segmentation]. In: IEEE Conference on Computer Vision and Pattern Recognition (2003)
82. Xiang, X., Lin, Q., Allebach, J.P.: Boosting high-level vision with joint compression artifacts reduction and super-resolution. In: International Conference on Pattern Recognition. IEEE (2020)
83. Xiang, X., Tian, Y., Zhang, Y., Fu, Y., Allebach, J.P., Xu, C.: Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3370–3379 (2020)
84. Xiang, X., Tian, Y., Zhang, Y., Fu, Y., Allebach, J.P., Xu, C.: Zooming slowmo: An efficient one-stage framework for space-time video super-resolution. arXiv preprint arXiv:2104.07473 (2021)
85. Xiao, Z., Xiong, Z., Fu, X., Liu, D., Zha, Z.J.: Space-time video super-resolution using temporal profiles. In: ACM International Conference on Multimedia. pp. 664–672 (2020)
86. Xu, G., Xu, J., Li, Z., Wang, L., Sun, X., Cheng, M.M.: Temporal modulation network for controllable space-time video super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 6388–6397 (2021)
87. Xue, T., Chen, B., Wu, J., Wei, D., Freeman, W.T.: Video enhancement with task-oriented flow. International Journal of Computer Vision **127**(8), 1106–1125 (2019)
88. Yuan, L., Sun, J., Quan, L., Shum, H.Y.: Image deblurring with blurred/noisy image pairs. ACM Transactions on Graphics **26**(3) (2007)
89. Zelnik-Manor, L., Irani, M.: Event-based analysis of video. In: IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, pp. II–II. IEEE (2001)
90. Zhang, K., Luo, W., Stenger, B., Ren, W., Ma, L., Li, H.: Every moment matters: Detail-aware networks to bring a blurry image alive. In: ACM International Conference on Multimedia. pp. 384–392 (2020)
91. Zhang, R.: Making convolutional networks shift-invariant again. In: International Conference on Machine Learning. pp. 7324–7334. PMLR (2019)
92. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2472–2481 (2018)

93. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 9308–9316 (2019)
94. Zou, X., Xiao, F., Yu, Z., Lee, Y.: Delving deeper into anti-aliasing in convnets. In: British Machine Vision Conference (2020)