An Empirical Investigation of Efficient Spatio-Temporal Modeling in Video Restoration

Yuchen Fan, Jiahui Yu, Ding Liu, Thomas S. Huang
University of Illinois Urbana-Champaign
{yuchenf4, jyu79, dingliu2, t-huang1}@illinois.edu

Abstract

*We present a comprehensive empirical investigation of efficient spatio-temporal modeling in video restoration tasks. To achieve a better speed-accuracy trade-off, our investigation covers the intersection of three dimensions in deep video restoration networks: spatial-wise, channel-wise and temporal-wise. We enumerate various network architectures ranging from 2D convolutional models to their 3D extensions, and discuss their gain and loss in terms of training time, model size, boundary effects, prediction accuracy and the visual quality of restored videos. Under a strictly controlled computational budget, we also specifically explore the design inside each residual building block in a video restoration network, which consists a mixture of 2D and 3D convolutional layers. Our findings are summarized as follows: (1) In 3D convolutional models, setting more computation/channels for spatial convolution leads to better performance than on temporal convolution. (2) The best variant of 3D convolutional models is better than 2D convolutional models, but the performance gap is close. (3) In a very limited range, the performance can be improved by the increase of temporal window size (5 frames for 2D model) or padding size (6 frames for 3D model). Based on these findings, we propose the wide-activated 3D convolutional network for video restoration (**WDVR**), which achieves state-of-the-art restoration accuracy under constrained computational budgets with low runtime latency. Our solution based on WDVR also won 2nd places in three out of four tracks of NTIRE 2019 Challenge for Video Super-Resolution and Deblurring. Code and models are released at* https://github.com/ychfan/wdvr_ntire2019.

1. Introduction

While videos are one of the most popular media to deliver information and data, the distortion problem in the videos caused by camera calibration, motion blurring, compression over transmission and low-resolution sensors remains unsolved in many practical scenarios. The problem of video restoration is challenging over-time in the signal processing area. It is an ill-posed inverse problem that targets on recovering the original video from its degraded counterparts. Among different types of degradation in videos, video super-resolution and deblurring are two of the most important and representative problems, illustrated in Figure 1.

With the recent development of deep convolutional neural networks (CNNs) in image recognition [1, 2], 2D and 3D CNNs have also been successfully applied to the task of image and video restoration. These methods usually model image and video restoration as a direct mapping function trained with large-scale data via a deep neural network [3, 4, 5, 6, 7]. Recent works [4, 5] have shown that deeper networks lead to better performance in terms of peak signal-to-noise ratio and structural similarity. However, very heavy computation is also required for the deep network architecture during training and inference, which limits its usage in the industrial scenarios.

Previous arts have studied the speed-accuracy trade-offs in image super-resolution problem with deep neural networks [7]. In this work, we present a comprehensive empirical investigation of the speed-accuracy trade-offs for spatio-temporal modeling in video restoration tasks, including video supre-resolution and deblurring. Various network structures are enumerated, trained and tested, ranging from 2D convolutional models to their 3D extensions. Our investigation covers the intersection of three dimensions in deep video restoration networks, spatial-wise, channel-wise and temporal-wise. We discuss their gain and loss in terms of training time, model size, boundary effects, prediction accuracy and the visual quality of restored videos. We also specifically explore the design inside each residual building block, which consists a mixture of 2D and 3D convolutional layers, in a video restoration network under a strictly controlled computational budget.

In summary, our findings are as follows. First, in 3D convolutional models, setting more computation/channels

(a) HR (b) Deblurring (c) Deblurring with compression artifacts (d) Super-resolution (e) Super-resolution with blur

Figure 1: Examples of video restoration with WDVR.

for spatial convolution leads to superior performance over temporal convolution. Second, the best variant of 3D convolutional models is better than 2D convolutional models, but the performance gap is close. Third, in a very limited range, the performance can be improved by the increase of temporal window size (5 frames for 2D model) or padding size (6 frames for 3D model). Based on these findings, we introduce the wide-activated 3D convolutional network for video restoration (**WDVR**), which achieves state-of-the-art restoration accuracy under constrained computational budgets with low runtime latency. We provide extensive ablation studies to validate our findings. Our solution based on WDVR won 2nd places in three out of four tracks of NTIRE 2019 Challenge for Video Super-Resolution and Deblurring.

2. Related Work

In this section, we first review several related methods for the task of image restoration. We then go over related approaches on video restoration. We pay special attention to the deep learning (DL) based methods for image/video super-resolution and deblurring, which are closely relevant to our proposed method. Finally we review the topics and solutions on spatio-temporal modeling in videos.

2.1. Image Restoration

Recently, deep learning (DL) based approaches have been developed for image restoration, and have achieved impressive results over the conventional non-DL practices. Considering single image super-resolution as an example, the methods built on DL [3, 4, 5, 6, 7] have dramatically improved the performance over conventional non-DL methods [8, 9], in terms of peak signal-to-noise ratio, structural similarity [10] and perceptual visual quality. These methods usually treat the deep neural network as an end-to-end mapping function from a low-resolution image to its high-resolution counterpart. Based on this practice, extensive works have been focusing on how to design more efficient and effective network (sub-)architectures including residual blocks [5], network depth [11, 12, 13, 5, 14, 15], recurrent architectures [16, 17, 18], skip connections [15, 18, 14], upsampling layers [19, 20, 21], normalization layers [7, 6], non-local attention mechanism [22, 23] and activation functions [24].

Among these recent advances, we selectively review several techniques that are closely related in our work. To improve the efficiency of deep neural networks for image super-resolution, WDSR [7] demonstrated that models with wider features before ReLU activation have significantly better performance under the same parameters and computational budgets. Thus a super-resolution residual network was designed to have a slim identity mapping pathway with wider ($2\times$ to $4\times$) channels before activation in each residual block. Moreover, Yu *et al*. [7] found that training with weight normalization leads to better accuracy compared with batch normalization or no normalization. In addition, since the input and output of a super-resolution network have different spatial resolution, an upsampling layer is usually required. Shi *et al*. [25] proposed a sub-pixel convolutional neural network as a replacement of computationally expensive deconvolution operation. Odena *et al*. [21] also demonstrated that deconvolution could lead to checkerboard artifacts thus should be avoided.

As for image deblurring, recent solutions of this problem are mainly on estimating the deblur function by training a deep neural network learned from noise-clean data

pair. In the scale-recurrent network for deep image deblurring [26], Tao *et al*. investigated the coarse-to-fine scheme to gradually restore the sharp image on different resolutions in a pyramid. Zhang *et al*. [27] proposed a spatially variant neural network which is composed of three deep convolutional neural networks (CNNs) and a recurrent neural network (RNN). In [27], RNN is used as a deconvolution operator performed on feature maps extracted from the input image by one of the CNNs, and another CNN is used to learn the weights for the RNN at every location.

2.2. Video Restoration

The difference between video restoration and image restoration mainly lies in the exploitation of temporal information. Since natural videos usually have consecutive coherent frames with the same objects and scenes, efficiently and effectively leveraging the neighboring frames to predict the center frame could lead to better results. Here we focus on reviewing recent DL based approaches of video restoration, while using video super-resolution and video deblurring as examples.

In recent years, the major focus of video super-resolution (VSR) was on temporal modeling within the framework of deep neural networks. For example, Caballero *et al*. [28] explored the use of early fusion, slow fusion and 3D convolutions for the joint processing of multiple consecutive video frames. A joint motion compensation and video super-resolution network based on a fast multi-resolution spatial transformer module was introduced in [28]. Tao *et al*. [29] proposed a sub-pixel motion compensation (SPMC) layer in a CNN framework for video super-resolution. Liu *et al*. [30] developed a temporal adaptive network for VSR via using filters on various temporal scales to extract features. Moreover, Sajjadi *et al*. [31] proposed an end-to-end trainable frame-recurrent video super-resolution framework that uses the previously inferred HR estimate to super-resolve the subsequent frame.

In the task of video deblurring, the non-uniform blur is usually caused by unwanted camera shake and/or object motion in dynamic scenes. Wieschollek *et al*. [32] introduced a recurrent network architecture to deblur frames and considered temporal information into account. The trained network was able to efficiently handle arbitrary spatial and temporal input sizes. Kim *et al*. [33] proposed an online (sequential) video deblurring method based on a spatio-temporal recurrent network. Real-time performance can be achieved by the proposed network layer that enforces temporal consistency between consecutive frames with dynamic temporal blending. Segmentation information was incorporated with video deblurring when available in [34].

2.3. Spatio-Temporal Modeling in Videos

The ability to understand videos is one of the most important milestones in artificial intelligence. Such tasks include video classification, action recognition, optical flow estimation, video restoration and many others. With the rapid development and progress made with 2D convolutional networks on image-related tasks [2, 1, 35, 36, 37, 38] such as image classification, detection and control agents on playing Atari games, its direct extensions, 3D convolutional networks, are comprehensively explored for video-related tasks.

3D Convolution was firstly proposed in [39] where the authors used a homogeneous architecture with small $3 \times 3 \times 3$ convolution kernels in all layers for video classification. Tran *et al*. [40] further proposed a improved spatiotemporal convolutional block "R(2+1)D" based on the observation that 2D CNNs applied to individual frames of the video have remained solid performers in action recognition. Qiu *et al*. [41] proposed a Pseudo-3D Residual Net (P3D ResNet) to exploit all the variants of blocks but composes each in different placement of ResNet. The design is based on the assumption that enhancing structural diversity with going deep could improve the power of neural networks. Xie *et al*. [42] demonstrated that it is possible to replace many of the 3D convolutions by low-cost 2D convolutions, especially the ones on low-level semantics. It indicated that temporal representation learning on high-level semantic features is more useful.

3. Deep Network Models for Video Restoration

In this section, we describe in details the neural network architectures for video restoration from the models based on 2D convolutions to the models with 3D convolutions. We elaborated on the relation and difference between these two types of models in terms of spatio-temporal modeling information from videos.

3.1. 2D Convolutional Models

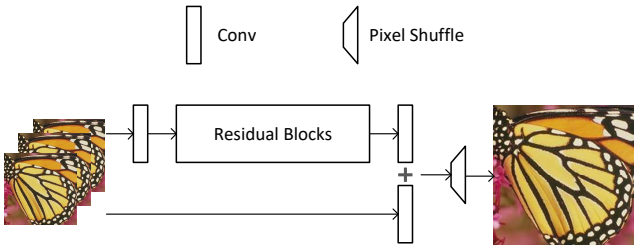

Figure 2: 2D convolutional models for video restoration.

In this section, we first explore the naive 2D convolutional design for video restoration. The 2D convolutional models for video restoration is adapted from models for

the single-image restoration task. We treat the temporal information as the input features, *i.e.*, for each frame output at frame index t, we take several input frames centered at frame index t and concatenate them as the network input. As shown in Figure 2, the models take multiple consecutive frames from a degraded video as input and restore the center frame. During training and inference, the 2D models work in a sliding window manner and propagate the restored images sequentially. The 2D models are deep structures stacked with multiple residual blocks and two additional convolutional layers to adapt the input and output dimensions. In the widely adopted approach of residual learning, along with the deep branch, a skip connection is used with single convolution directly from input to output. For super-resolution tasks, an additional pixel shuffle layer is appended to the very end of the models.

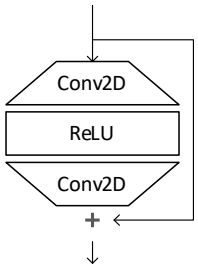

Figure 3: 2D wide-activated residual block.

The 2D wide-activated residual blocks, as shown in Figure 3, are the commonly used basic building block for the 2D convolutional models. In such a residual building block, there are two 2D convolution layers alternated with one ReLU activation layer. Meanwhile, an identity skip connection from block inputs is added up to outputs. The wide-activated design allows more channels in the activation layer, compared to inputs and outputs, which is more effective in performance and more efficient in speed. As suggested in [7], expanding features before ReLU allows more information to pass through while still maintaining high non-linearity of deep neural networks. Thus low-level features from shallow layers may be easier to propagate to the final layer for better dense pixel value predictions.

However, the receptive field in temporal domain is fixed in such 2D models, because the 2D models have pre-defined input filter sizes, which takes a fixed number of stacked frames, *i.e.*, window size of temporal receptive field. Thus, the number of input concatenation frames is one of the most important hyper-parameters for temporal modelling in 2D models. We will show in the experiment section that choosing proper window size has a major impact on performance.

In addition, it is noteworthy that such 2D models are not very efficient for exploring long-range temporal information. The first convolution layer fuses all the frames with single linear transform, and later layers have little effect on modeling additional temporal information. In this way, given a 2D model of fixed temporal scale, the more stacked input frames, the less representation there is for each frames in hidden layers.

3.2. 3D Convolutional Models

Here we explore the 3D convolutional models for efficient modeling of temporal information. The 3D convolutional operations are more intrinsic ways for joint spatial and temporal modelling. We extend a 2D convolutional model to a 3D one by following the same structure and replacing all the 2D convolutional layers to 3D ones. Different from the 2D convolutional model, a 3D model take a clip of video with arbitrary number of frames and predict their restored counterparts simultaneously. The deep structure with multiple 3D convolutional layers explore spatial and temporal information gradually and is expected to be both efficient and effective for context modelling of videos.

Compared with 2D models, 3D models are capable to capture temporal signals not only in the input layer but also in succeeding 3D convolutional layers. Figure 4.(a) and (b) illustrate the difference between 2D and 3D residual blocks. In 2D models, all the consecutive frames are already fused in the very first layer, then temporal dimension is squeezed and convolutional operations only perform on spatial dimensions. Compared to 3D models, 2D convolutional layers are fully connected in temporal axis but the temporal axis is limited by the number of input stacked frames. From computational perspective, the 2D convolutional layers are identical to 3D spatial convolutional layers. Because multiple consecutive frames are fused in the entry layers of 2D models, the temporal information can be kept only along the channel axis in a very limited manner.

In 3D convolution layers, all the parameters are related to temporal modelling. However, video restoration as a low-level vision task relies on more local information for restoring fine details. Without explicit motion compensation and frame warping, it is difficult to effectively capture information from videos containing fast motion or drastic scene changes by simple local 3D convolution operations. Hence, the ratio of parameters for temporal modelling is worth careful investigation.

Multiple 3D wide-activated residual blocks in 3D models are designed with different ratio of parameters for temporal modeling in this section. We explain them in details as follows. The most straightforward design of 3D is the inflated version of 2D wide-Activated blocks, named as *IAI* and shown in Figure 4.(b). To reduce the ratio of temporal parameters by half, the inflated 3D convolution after activation is replaced with spatial convolution, named as *IAS* and shown in Figure 4.(c). To further reduce the ratio, the other inflated 3D convolution is decomposed to spatio-temporal

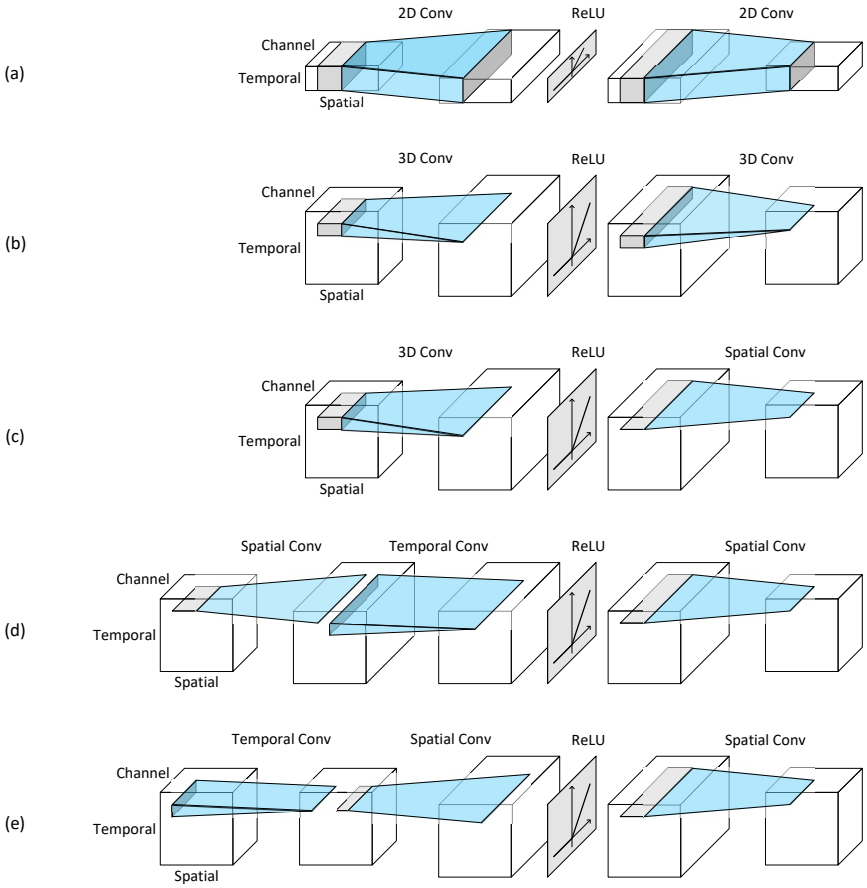

Figure 4: 3D wide-activated residual blocks with different designs. (a) 2D. (b) IAI. (c) IAS. (d) STAS. (e) TSAS.

convolution, named as *STAS* and shown in Figure 4.(d). The spatio-temporal convolution explicitly isolates the parameters for spatial and temporal modelling. In *STAS*, the temporal convolution is connected with activations, so it has more channels than input and output features. Switching the order of spatio-temporal convolution, named as *TSAS* and shown in Figure 4.(e), can reduce temporal parameters even more by moving temporal convolution to connect narrow block inputs. In the experiment section, the proposed designs of 3D residual blocks are carefully compared to find the most efficient spatio-temporal modelling approach.

4. Experiment

4.1. Main Results

In this section, we investigate the performance of 2D and 3D convolutional models with detailed experimental results.

4.1.1 Datasets and Metrics

We use the REDS video restoration dataset [**?**] in NTIRE 2019 Challenges. The dataset contains 300 videos and is split to 240, 30 and 30 for training, validation and testing set respectively. Each video comes with 100 consecutive frame with 1280x720 resolution. There are 4 different types of degradation, including 4x bi-cubic down-sampling for super-resolution, 4x blurry down-sampling for super-resolution, blurring and compression.

The models discussed in this section is based on 4x bi-cubic down-sampling for super-resolution task. They are trained with 240 videos in training set and evaluated with 30 videos in validation set. They are trained with multiple epochs based on small patches of frames, several patches per frames is sampled in each epoch. The models are trained with L1 loss and evaluated with peak signal-to-noise ratio (PSNR) or structural similarity (SSIM) in RGB channels of every frames in validation set.

4.1.2 Training Settings

Training image frame patches have 64 pixels in width and height, and 100 patches are sampled from every videos in one single epoch. ADAM optimizer [43] is used with $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 108$. Weight normalization [44] is

Window size	PSNR
1	28.45
3	29.04
5	29.14
7	29.08

Table 1: 2D convolutional models with different window sizes.

Design	Activation width	Temporal ratio	PSNR
IAI	42	1.00	28.70
IAS	64	0.50	28.79
STAS	88	0.31	28.92
TSAS	122	0.06	28.94

Table 2: 3D convolutional models with different residual block designs.

applied for all the convolution kernels. The learning rate is initialized the maximum convergent value, that is 1e-3, then multiplied by 0.1 for final steps. The models discussed in Section 4.1 are limited to 16 residual blocks and 32 nodes in residual connections for fairness in comparisons.

4.1.3 2D Convolutional Models

The implementation of 2D convolutional models are very similarly to image super-resolution framework but multiple consecutive frames are stacked as a single input features. The most important hyper-parameter for 2D models is the window size, that is the number of stacked frames.

The 2D models with different window sizes are compared in Table 1. The compared models are early-stopped at 10th epoch, due to their training speed. Comparing models with window size 1, that is single image super-resolution, other models with stacked context frames are significantly better in PSNR. The models with windows size 5 achieve the optimal results, and is better than models with window size 7.

The results show that 2D convolution has limited capability in modelling long-term temporal dependency. When multiple frames are stacked as single input, all the information has to be kept in hidden represents with limited dimensions. Given fixed number of parameters, the width of hidden represents also with fixed size have to compress information from individual frames, when the window size increases.

4.1.4 3D Convolutional Models

The training of 3D convolutional models is based on video clips instead images. Each training sample is a pair of degraded consecutive frame patches in four dimensions (time, height, width and channels) and corresponding target frame patches. Although the temporal receptive field of 3D models is determined by the number of 3D convolutional layers theoretically, the practical temporal receptive field is limited by the length of input video clips. To avoid boundary effect in temporal, more consecutive frames are padded for input video clips. The clip size is 4 and padding size is 8, that

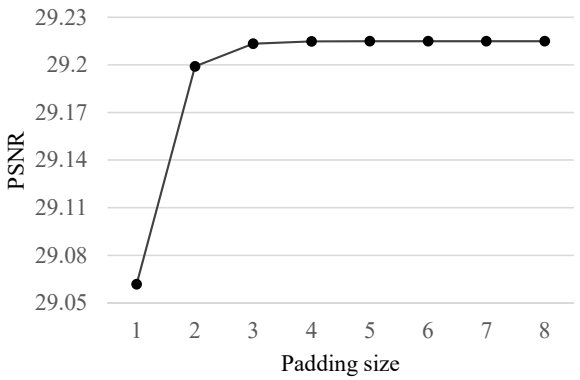

Figure 5: Padding size with 3D models performance.

is 20 consecutive frames for inputs and 4 central frames for targets, in experiments below.

Model Designs The proposed model designs have different ways for spatio-temporal modelling, hence different ratio of temporal parameters. Temporal modelling needs additional parameters comparing to 2D models, especially 3D convolution layers have one more parameter dimension. To ensure the fairness of comparison, the number of total parameters must be fixed. Given the fixed number of layers (16) and residual nodes (32), the number of activations inside residual blocks is adjusted to control the model size. The comparison of different residual block designs are shown in Table 2. The results show that the 3D models with less parameters for temporal modelling and more activations achieve better performance, which is coincident with our hypothesis: (1) spatial information is more important than temporal information in video restoration; (2) wide-activation is efficient model structure.

Temporal Modelling Capacity To measure the temporal modelling capacity of 3D models, the clip size is fixed to 1, and the 3D models are evaluated with different temporal padding size. As shown in Figure 6, greater padding size improves the performance until padding reaches 5 frames, which means the 3D model are capable to capture significant information from temporal neighbours up to 5 frames, equivalent to window size 11 in 2D models. The results show that the proposed 3D model has 2.5x temporal modelling capacity compared to 2D models.

PSNR/SSIM	Bicubic	Bayesian [45]	Liu *et al*. [30]	DUF[46]	WDVR
City	25.13 / 0.601	28.11 / 0.8916	26.50 / 0.8282	28.02 / 0.8950	26.92 / 0.8631
Walk	26.06 / 0.798	27.69 / 0.9122	28.35 / 0.9154	30.11 / 0.9475	30.16 / 0.9408
Calendar	20.54 / 0.571	23.78 / 0.8611	22.13 / 0.7802	23.84 / 0.8588	23.40 / 0.8333
Foliage	23.50 / 0.566	25.98 / 0.8477	25.09 / 0.7931	26.32 / 0.8419	25.99 / 0.8238
Avg.	23.81 / 0.634	26.39 / 0.8782	25.52 / 0.8292	27.07 / 0.8858	26.62 / 0.8653

Table 3: Quantitative comparison with state-of-the-art video super-resolution methods on Vid4 [47].

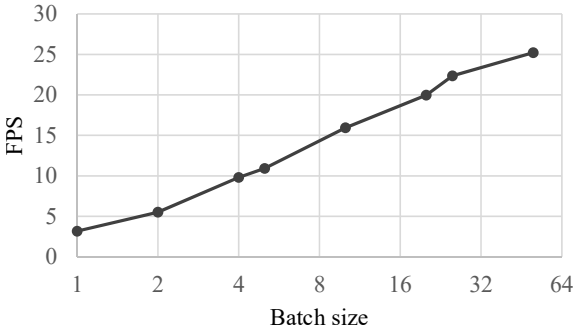

Figure 6: Batch size with 3D models speed.

Models	PSNR
2D	28.2085
3D	29.2148

Table 4: Comparison of 2D convolutional model and the best variant of 3D convolutional Models.

Batching Benefited by intrinsic of convolution in temporal, the 3D models can process video clips with multiple frames simultaneously, and reduce the redundant computation of overlapped consecutive frames. Results in Figure 6 show that batching can significant increase the speed of 3D models. Especially, when batch size is increased to 20, the performance can achieve 20 frames per second (FPS), which makes the 3D models can processing videos in real-time with only 1 second latency.

4.1.5 Comparison of 2D and 3D Models

Both 2D and 3D models are effective and efficient for video restoration. We further compared the models in Table 4. Results show that 3D models outperform 2D models, but the gap between them are not significant.

4.2. NTIRE 2019 Challenges

Based on our investigation, we found that the gap between 2D and 3D models are not very significant, while 3D models consume much more memory for training. Thus we participate the NTIRE 2019 Challenges with our investigated 2D convolutional models. The submitted models have 64 layers, 64 residual units and 256 activation units in each residual blocks.

Tracks	PSNR	SSIM
Deblurring Clean	34.17	0.9345
Deblurring Compression	28.73	0.7993
Super-Resolution Clean	29.86	0.8409
Super-Resolution Blur	28.68	0.8103

Table 5: Validation phase results in NTIRE 2019 Challenges.

Tracks	PSNR	SSIM	Ranking
Deblurring Clean	35.71	0.9522	2
Deblurring Compression	29.78	0.8285	2
Super-Resolution Clean	30.81	0.8748	6
Super-Resolution Blur	29.46	0.8430	2

Table 6: Test phase results in NTIRE 2019 Challenges.

For validation phase, the models are trained on training set for 20 epochs. The results on validation set are shown in Table 5.

For test phase, the models are further fune-tuned on training and validation set for 10 more epochs. The results on test set are shown in Table 6. 8x self-ensemble (including flips and rotations) is also used for our final submission. Our proposed methods outperform most of the teams [48, 49]. Our approach is also extremely efficient, which achieve 0.98 second per frame for video super-resolution task and can be reduced 8 times without self-ensemble.

4.3. Comparison of state-of-the-art

The proposed WDVR video restoration method are further evaluated on benchmark datasets for video super-resolution.

On Vid4 benchmark[47], our approach is comparable with state-of-the-art methods, as shown in Table 3. The PSNR and SSIM are calculated on luminance channel, 4 boundary pixels are ignored for 4x super-resolution. The Table 3 are re-calculated with published results of each

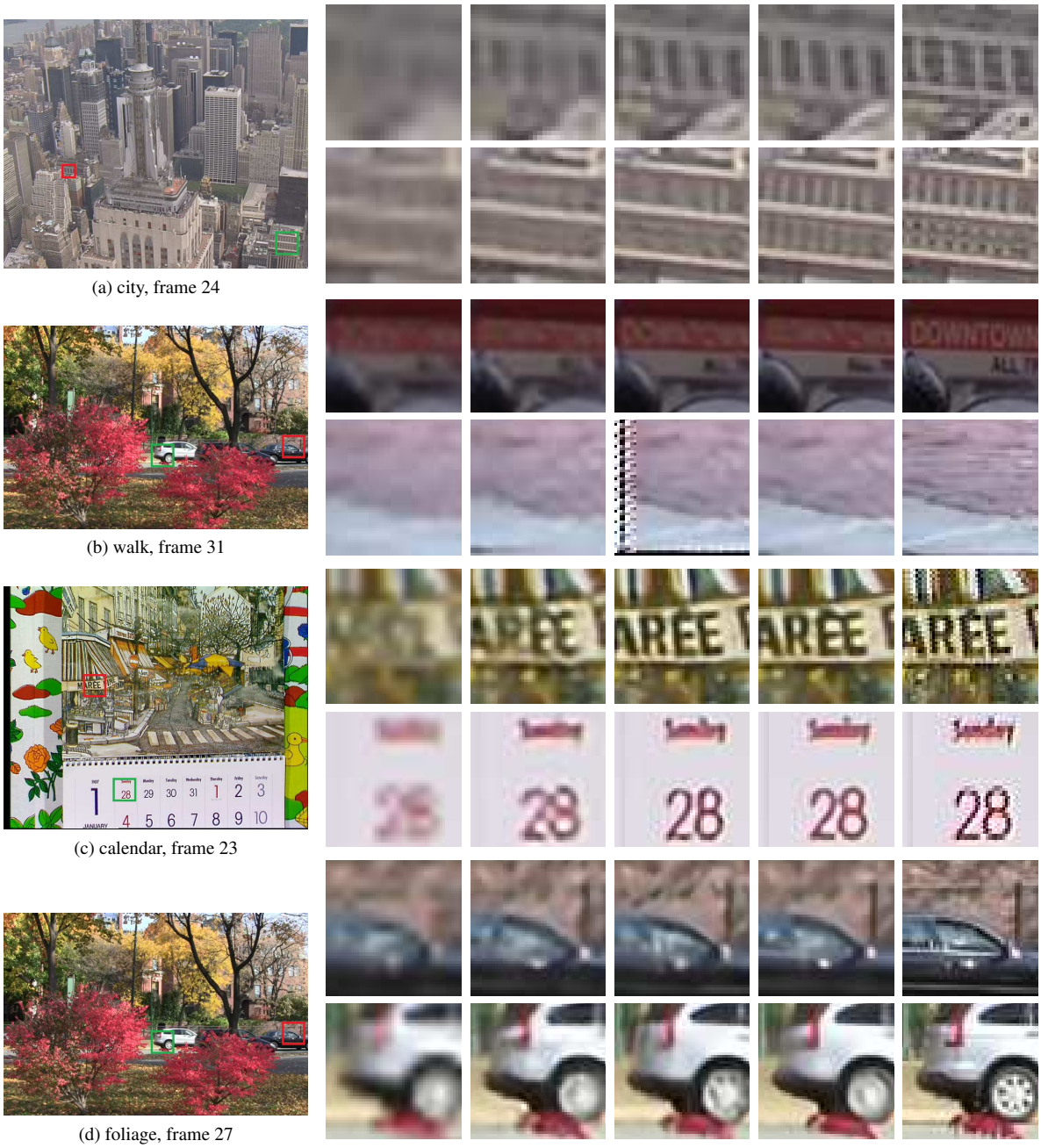

Figure 7: Visual comparisons with state-of-the-art video super-resolution methods on Vid4 [47].

method. For method, Liu *et al*. [30], the first two and last two frames in every video are ignored.

The qualitative results are shown in Figure 7. Although the DUF method perform better on 3 out of 4 videos, our approach is more robust with boundary effects as shown in video named walk.

5. Conclusion

In this work, we have investigated empirically of efficient spatio-temporal modeling in video restoration tasks. Based on our findings, we have introduced the **WDVR**, wide-activated 3D convolutional network for video restoration, which achieves a better accuracy under similar computational budgets and runtime latency.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. 1, 3

[2] J. Yu, L. Yang, N. Xu, J. Yang, and T. Huang, "Slimmable neural networks," *arXiv preprint arXiv:1812.08928*, 2018. 1, 3

[3] D. Liu, Z. Wang, B. Wen, J. Yang, W. Han, and T. S. Huang, "Robust single image super-resolution via deep networks with sparse prior," *IEEE TIP*, vol. 25, no. 7, pp. 3194–3207, 2016. 1, 2

[4] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee *et al.*, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2017, pp. 1110–1121. 1, 2

[5] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, vol. 1, no. 2, 2017, p. 3. 1, 2

[6] Y. Fan, J. Yu, and T. S. Huang, "Wide-activated deep residual networks based restoration for bpg-compressed images," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 1, 2

[7] J. Yu, Y. Fan, J. Yang, N. Xu, Z. Wang, X. Wang, and T. Huang, "Wide activation for efficient and accurate image super-resolution," *arXiv preprint arXiv:1808.08718*, 2018. 1, 2, 4

[8] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010. 2

[9] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730. 2

[10] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004. 2

[11] Y. Fan, H. Shi, J. Yu, D. Liu, W. Han, H. Yu, Z. Wang, X. Wang, and T. S. Huang, "Balanced two-stage residual networks for image super-resolution," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2017, pp. 1157–1164. 2

[12] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1646–1654. 2

[13] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network." 2

[14] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 4809–4817. 2

[15] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual Dense Network for Image Super-Resolution," *ArXiv e-prints*, Feb. 2018. 2

[16] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1637–1645. 2

[17] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network." 2

[18] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4539–4547. 2

[19] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*. Springer, 2014, pp. 184–199. 2

[20] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European Conference on Computer Vision*. Springer, 2016, pp. 391–407. 2

[21] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, 2016. [Online]. Available: http://distill.pub/2016/deconv-checkerboard 2

[22] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5505–5514. 2

[23] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, "Non-local recurrent network for image restoration," in *Advances in Neural Information Processing Systems*, 2018, pp. 1673–1682. 2

[24] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," *arXiv preprint arXiv:1806.03589*, 2018. 2

[25] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883. 2

[26] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8174–8182. 3

[27] J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M.-H. Yang, "Dynamic scene deblurring using spatially variant recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2521–2529. 3

[28] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4778–4787. 3

[29] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4472–4480. 3

[30] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2526–2534. 3, 7, 8

[31] M. S. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6626–6634. 3

[32] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang, "Deep video deblurring for hand-held cameras," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1279–1288. 3

[33] T. Hyun Kim, K. Mu Lee, B. Scholkopf, and M. Hirsch, "Online video deblurring via dynamic temporal blending network," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4038–4047. 3

[34] W. Ren, J. Pan, X. Cao, and M.-H. Yang, "Video deblurring via semantic segmentation and pixel-wise non-linear kernel," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1077–1085. 3

[35] J. Yu and T. Huang, "Universally slimmable networks and improved training techniques," *arXiv preprint arXiv:1903.05134*, 2019. 3

[36] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013. 3

[37] J. Yu and T. Huang, "Network slimming by slimmable networks: Towards one-shot architecture search for channel numbers," *arXiv preprint arXiv:1903.11728*, 2019. 3

[38] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "Unitbox: An advanced object detection network," in *Proceedings of the 24th ACM international conference on Multimedia*. ACM, 2016, pp. 516–520. 3

[39] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497. 3

[40] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459. 3

[41] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5533–5541. 3

[42] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 305–321. 3

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 5

[44] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 901–909. 5

[45] C. Liu and D. Sun, "On bayesian adaptive video super resolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 346–360, 2014. 7

[46] Y. Jo, S. Wug Oh, J. Kang, and S. 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*, 2018, pp. 3224–3232. 7, 8

[47] C. Liu and D. Sun, "A bayesian approach to adaptive video super resolution," in *CVPR 2011*. IEEE, 2011, pp. 209–216. 7, 8

[48] S. Nah, R. Timofte, S. Baik, S. Hong, G. Moon, S. Son, K. M. Lee, X. Wang, K. C. Chan, K. Yu, C. Dong, C. C. Loy, Y. Fan, J. Yu, D. Liu, T. S. Huang, H. Sim, M. Kim, D. Park, J. Kim, S. Y. Chun, M. Haris, G. Shakhnarovich, N. Ukita, S. W. Zamir, A. Arora, S. Khan, F. S. Khan, L. Shao, R. K. Gupta, V. Chudasama, H. Patel, K. Upla, H. Fan, G. Li, Y. Zhang, X. Li, W. Zhang, Q. He, K. Purohit, A. N. Rajagopalan, J. Kim, M. Tofighi, T. Guo, and V. Monga, "Ntire 2019 challenge on video deblurring: Methods and results," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 7

[49] S. Nah, R. Timofte, S. Gu, S. Baik, S. Hong, G. Moon, S. Son, K. M. Lee, X. Wang, K. C. Chan, K. Yu, C. Dong, C. C. Loy, Y. Fan, J. Yu, D. Liu, T. S. Huang, X. Liu, C. Li, D. He, Y. Ding, S. Wen, F. Porikli, R. Kalarot, M. Haris, G. Shakhnarovich, N. Ukita, P. Yi, Z. Wang, K. Jiang, J. Jiang, J. Ma, H. Dong, X. Zhang, Z. Hu, K. Kim, D. U. Kang, S. Y. Chun, K. Purohit, A. N. Rajagopalan, Y. Tian, Y. Zhang, Y. Fu, C. Xu, A. M. Tekalp, M. A. Yilmaz, C. Korkmaz, M. Sharma, M. Makwana, A. Badhwar, A. P. Singh, A. Upadhyay, R. Mukhopadhyay, A. Shukla, D. Khanna, A. Mandal, S. Chaudhury, S. Miao, Y. Zhu, and X. Huo, "Ntire 2019 challenge on video super-resolution: Methods and results," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 7