SFU

CMPT 733: Final Project Report

Video Deblurring with Transformer

Hao Wu, Luxi Wang, Long Jin, Yifu Zhang

April 5, 2023

https://github.com/Cole9712/DeepDeblur-Transformer

1 Introduction

Video deblurring is a fundamental yet challenging problem in computer vision that aims to restore clear and sharp content in videos degraded due to camera shake, slow shutter speed, or out-of-focus shots. As a low-level computer vision problem, video deblurring can improve video quality and clarity, thereby improving the effect of downstream high-level vision problems such as object recognition and video semantic segmentation.

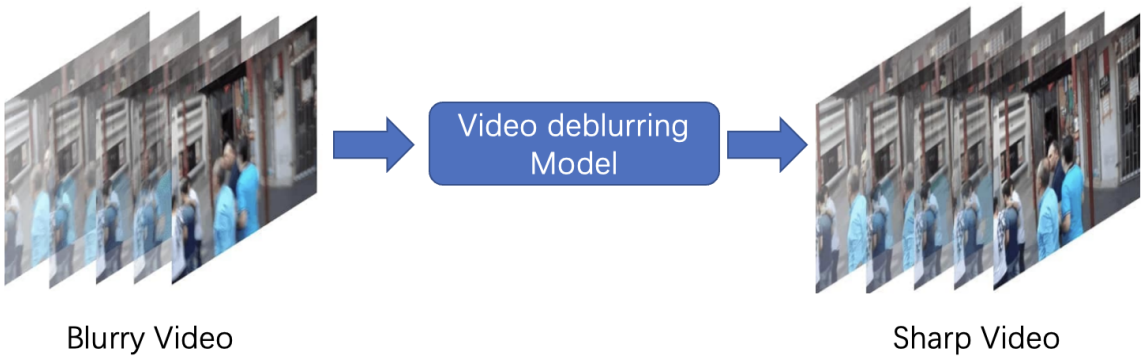

Figure 1: The input and output of video deblurring model

As shown in Fig. 1, the input to the video deblurring problem is a blurred video, and the output is a clear and sharp video that closely resembles the original content. The problem of video deblurring is difficult because it requires the estimation of both

the blur and the original clear content from a single degraded video. And the difficulty in solving this problem arises from the complex nature of blur, which can vary spatially and temporally, making it challenging to model and remove accurately. Additionally, noise and other factors, such as moving objects, complicate the deblurring process further.

2 Related Work

In the early days, traditional methods rely on certain assumptions regarding motion blur and latent frames in order to recover the latent frames from a blurred sequence [1, 2, 3, 4, 5]. Optical flow is commonly used to model the motion blur in these methods [1, 2, 5]. To achieve favorable outcomes, these techniques jointly estimate the optical flow and latent frames under the constraints imposed by some manually-crafted priors. While these algorithms are inspired by physical principles and deliver encouraging results, the assumptions on motion blur and latent frames often result in intricate energy functions that are challenging to solve.

With the development of deep learning, A lot of CNN-based models came up. Kim et al. [6] created an optical flow estimation procedure to align the frames and recover latent ones by combining data from neighboring frames. Zhou et al. enhanced frame alignment by extending the kernel prediction network [7]. Wang et al. [8] implemented pyramid, cascading, and deformable convolution techniques to enhance alignment accuracy. These models can handle more complex motion blur, but hard to capture long-distance dependencies. Some researchers try to use RNN to address this issue [9, 10], these models easy to capture global information, but they are often inefficient and hard to fit real-time demands.

3 Proposed Methodology

Our goal is to enhance the performance of video deblurring methods by incorporating transformers into existing CNN-based architectures.We begin by presenting a new Transformer block called the Locally-Enhanced Window (LeWin) Transformer block that utilizes non-overlapping window-based self-attention in place of global self-attention.

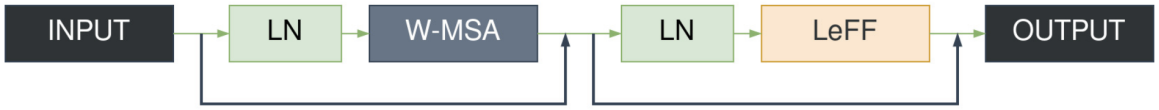

Figure 2: Lewin Transformer Block

LeWin is a visual transformer structure first proposed by Liu et al [11].As shown in Fig. 2, the key components of LeWin Transformer block are Window-based multi-head self-attention block (W-MSA) and Locally-enhanced Feed-forward Network (LeFF). W-MSA restricts the attention mechanism to a local region or window of the input sequence. This approach uses a fixed window size and a stride to control the overlap between adjacent windows. It reduces the computational complexity and memory usage of the Self-Attention mechanism, while still capturing the relevant information from the input sequence. This technique is particularly useful for processing large input sequences, such as images or videos, where it is important to capture the spatial or temporal structure of the data.

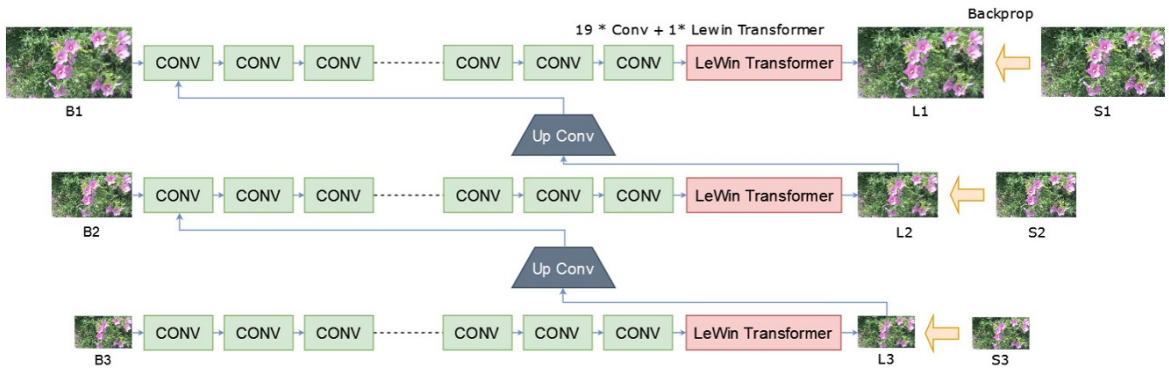

Figure 3: MSRTNet Lite

As shown in Fig. 3 and Fig. 4, we propose two different structures. The first is 19 convolutional layers followed by 1 LeWin Transformer block called MSRTNet Lite. We want first to make sure the performance of the LeWin Transformer is indeed better than ResBlock, and also test the computation time and consumption. Then we propose the second architecture, consisting of 3 convolutional layers and one LeWin Transformer called MSRTNet. We take it as one combination, each multi-scale layer will have 3 combinations. And there are 9 convolutional layers and 3 Lewin Transformer blocks in total.

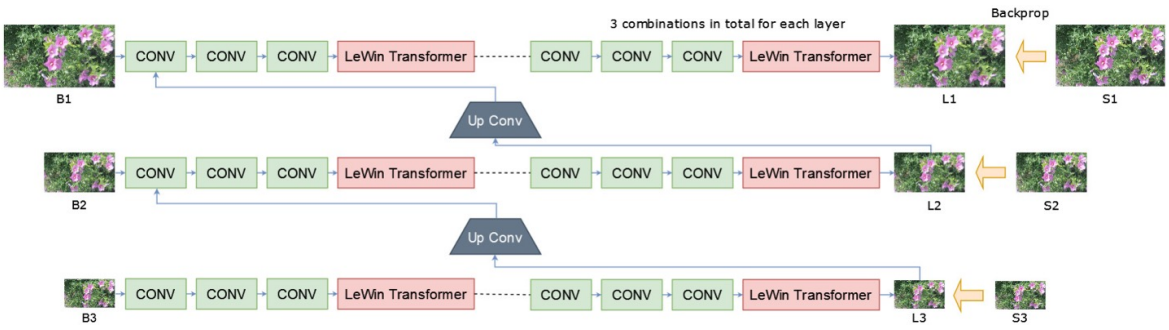

Figure 4: MSRTNet

4 Results and Discussion

For fair comparisons, we use the two commonly-used public datasets: GOPRO_Large [12] and DVD [13], for the model training and testing. The GOPRO_Large dataset consists of 3,214 blurred images with a size of 1,280×720 and the DVD dataset consists of 14,732 blurry-sharp image pairs across 11 scenes. We use the similar data augmentation method to [6] to generate training data. The size of each image patch is 128 × 128 pixels. We use an NVIDIA A100 GPU with 80G memory to train our model. When training on GOPRO_Large dataset, we set epoch to 600, batch size to 200, and learning rate to 2e-4. When training on DVD dataset, we set epoch to 400, batch size to 100, and learning rate to 1e-4.

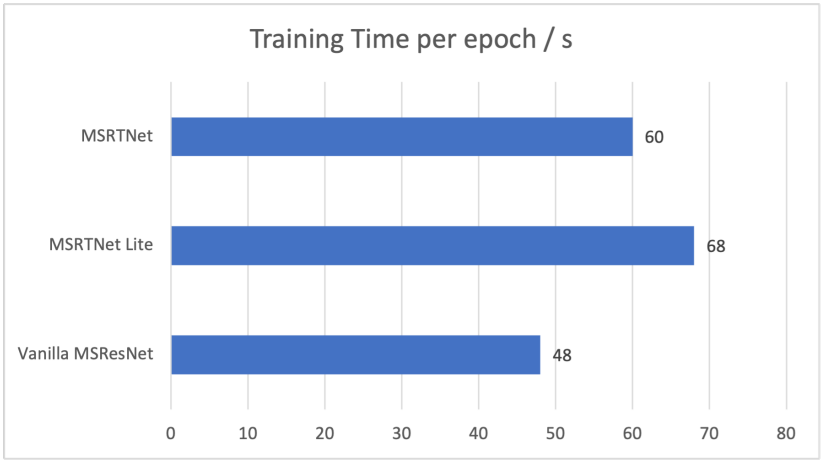

Figure 5: Model training speed comparison

Fig. 5 shows the model training speed comparison. Comparing the time consump-

tion of each epoch between Vanilla MSResNet and MSRTNet Lite, we can see that the introduction of LeWin Transformer Block does increase the training time. Comparing the time consumption of each epoch between MSRTNet Lite and MSRTNet, it can be seen that by reducing the number of ResBlocks and increasing the number of LeWin Transformer Blocks, the training time can be reduced while ensuring the model effect.

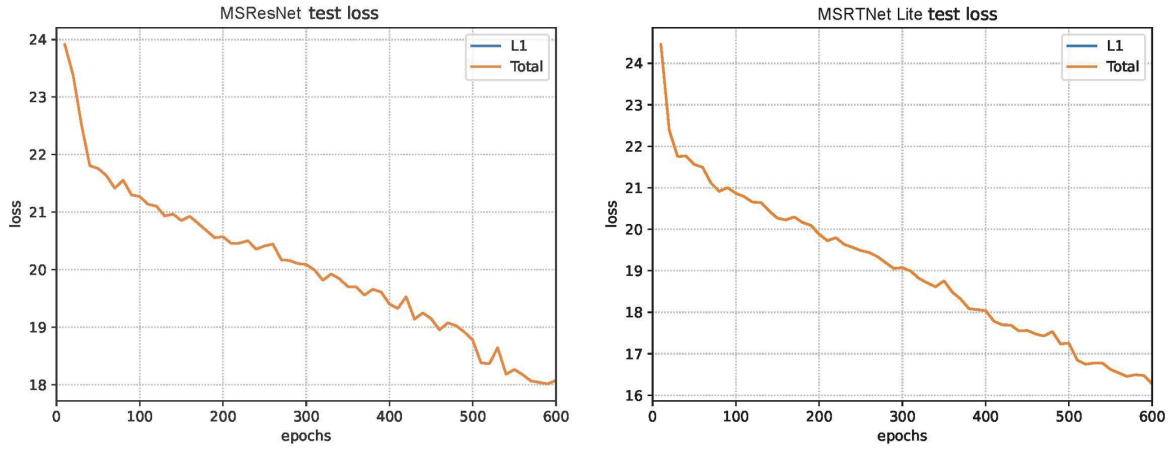

Figure 6: Comparison of training loss on GOPRO_Large dataset

Fig. 6 shows the comparison of training loss on GOPRO_Large dataset. We can observe that the test loss decreases steadily over time, with a gradual decrease, suggesting that the model is generalizing well to new, unseen data. The MSRTNet Lite model shows a similar trend, but at a faster pace, which shows it works better than the default model.

Fig. 7 shows the comparison of training loss on DVD dataset. For DVD dataset, we trained the vanilla MSResNet model and our MSRTNet model to compare. Similar to the GOPRO dataset, it clearly shows our model has better performance. However, due to time limitation, we are not able to get other models to compare since most paper's pretrained model are based on 256 patch size, we have to train it locally to get a model with 128 patch size to compare.

Fig. 8 shows some test results on the GOPRO_Large dataset. As can be seen from the figure, both MSRTNet Lite and MSRTNet have better deblurring effects on the logo than MSResNet. At the same time, since MSRTNet Lite uses more ResBlocks, the network depth increases, making the effect due to MSRTNet, but its inference takes longer. In quantitative comparison, the PSNR values of MSResNet for the two images

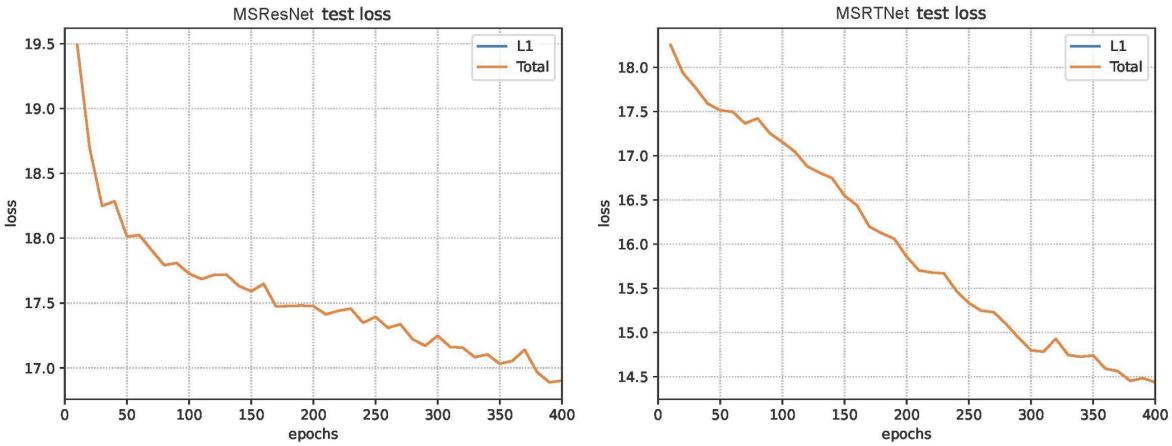

Figure 7: Comparison of training loss on DVD dataset

are 33.91 dB and 35.64 dB, and the SSIM values are 0.91 and 0.90. The PSNR values of MSRTNet Lite for the two images are 34.22 dB and 37.38 dB, and the SSIM values are 0.921 and 0.955. The PSNR values of the graph are 34 dB and 36.64 dB, and the SSIM values are 0.915 and 0.945. This also quantitatively shows that the results of MSRTNet Lite and MSRTNet are better than MSResNet.

Fig. 9 shows some test results on the DVD dataset. As can be seen from the figure, MSRTNet is better than MSResNet for restoring vehicle details and removing blur of text. Regarding quantitative comparison, the PSNR values of MSResNet for the two images are 31.02 dB and 32.66 dB, and the SSIM values are 0.617 and 0.778. The PSNR values of MSRTNet for the two images are 31.4 dB and 33.69 dB, and the SSIM values are 0.698 and 0.855. This also quantitatively shows that the results of MSRTNet are better than MSResNet.

5 Conclusion

Our project introduces the MSRTNet model, which is based on existing ConvNet structures and enhanced with the LeWin Transformer block for video deblurring tasks. The LeWin Transformer allows our model to effectively capture long-range dependencies and handle local context. Our extensive experiments demonstrate that our model outperforms vanilla ConvNet-based models.

Figure 8: Test results on the GOPRO_Large dataset. (a) Ground Truth sharp images. (b) Results of MSResNet. (c) Results of MSRTNet Lite. (d) Results of MSRTNet.

References

[1] Tae Hyun Kim and Kyoung Mu Lee. Generalized video deblurring for dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5426–5434, 2015.

[2] Leah Bar, Benjamin Berkels, Martin Rumpf, and Guillermo Sapiro. A variational framework for simultaneous motion estimation and restoration of motion-blurred video. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

Figure 9: Test results on the DVD dataset. (a) Ground Truth sharp images. (b) Results of MSResNet. (c) Results of MSRTNet.

[3] Sunghyun Cho, Jue Wang, and Seungyong Lee. Video deblurring for hand-held cameras using patch-based synthesis. *ACM Transactions on Graphics (TOG)*, 31(4):1–9, 2012.

[4] Tae Hyun Kim and Kyoung Mu Lee. Segmentation-free dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2766–2773, 2014.

[5] Jonas Wulff and Michael Julian Black. Modeling blurred video with layers. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 236–252. Springer, 2014.

[6] Tae Hyun Kim, Mehdi SM Sajjadi, Michael Hirsch, and Bernhard Scholkopf. Spatio-temporal transformer network for video restoration. In *Proceedings of the*

European Conference on Computer Vision (ECCV), pages 106–122, 2018.

[7] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2502–2510, 2018.

[8] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[9] Zhihang Zhong, Ye Gao, Yinqiang Zheng, and Bo Zheng. Efficient spatio-temporal recurrent neural network for video deblurring. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 191–207. Springer, 2020.

[10] Seungjun Nah, Sanghyun Son, and Kyoung Mu Lee. Recurrent neural networks with intra-frame iterations for video deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8102–8111, 2019.

[11] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[12] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring, 2016.

[13] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. pages 237–246, 07 2017.