Flow-Guided Sparse Transformer for Video Deblurring

Jing Lin [*1] **Yuanhao Cai** [*1] **Xiaowan Hu** [1] **Haoqian Wang** [†1] **Youliang Yan** [2]
Xueyi Zou [†2] **Henghui Ding** [3] **Yulun Zhang** [3] **Radu Timofte** [3] **Luc Van Gool** [3]

Abstract

Exploiting similar and sharper scene patches in spatio-temporal neighborhoods is critical for video deblurring. However, CNN-based methods show limitations in capturing long-range dependencies and modeling non-local self-similarity. In this paper, we propose a novel framework, Flow-Guided Sparse Transformer (FGST), for video deblurring. In FGST, we customize a self-attention module, Flow-Guided Sparse Window-based Multi-head Self-Attention (FGSW-MSA). For each $query$ element on the blurry reference frame, FGSW-MSA enjoys the guidance of the estimated optical flow to globally sample spatially sparse yet highly related key elements corresponding to the same scene patch in neighboring frames. Besides, we present a Recurrent Embedding (RE) mechanism to transfer information from past frames and strengthen long-range temporal dependencies. Comprehensive experiments demonstrate that our proposed FGST outperforms state-of-the-art (SOTA) methods on both DVD and GOPRO datasets and yields visually pleasant results in real video deblurring. https://github.com/linjing7/VR-Baseline

1. Introduction

Video deblurring is a fundamental yet challenging task in low-level computer vision and graphics communities. It aims to restore the latent frames from a blurry video sequence. Serving as a preprocessing technique, video deblurring has wide applications such as video stabilization (Matsushita et al., 2006), tracking (Jin et al., 2005), autonomous driving (Yin et al., 2021), *etc.* Hand-held devices are more and more popular in capturing videos of dynamic scenes, where prevalent depth variations, abrupt camera shakes, and high-speed object movements lead to undesirable blur in videos. To alleviate the effect of motion blur, researchers have put a lot of efforts into video deblurring.

Conventional methods are mainly based on hand-crafted priors and assumptions, which limits the model capacity. Besides, the assumptions on motion blur and latent frames usually lead to complex energy functions that are difficult to solve. Also, the inaccurately estimated motion blur kernel with hand-crafted priors may easily result in severe artifacts.

In the past decade, video deblurring has witnessed significant progresses with the development of deep learning. Convolutional neural network (CNN) applies a powerful model to learn the mapping from blurry videos to sharp videos under the supervision of a large-scale dataset of blurry-sharp video pairs. CNN-based methods yield impressive performance but show limitations in modeling long-range spatial dependencies and capturing non-local self-similarity.

Recently, the emergence of Transformer provides an alternative to alleviate the constraints of CNN-based methods. **Firstly**, Transformer excels at modeling long-range spatial dependencies. The contextual information and spatial correlations are critical to restoring the motion blur. **Secondly**, similar and sharper scene patches from neighboring frames provide crucial cues for video deblurring. Fortunately, the self-attention module in Transformer is dedicated to calculating the correlations among pixels and capturing the self-similarity along the temporal sequence. Thus, Transformer inherently resonates with the goal of learning similar information from spatio-temporal neighborhoods. **Nevertheless**, directly using existing Transformers for video deblurring has two issues. **On one hand**, when the standard global Transformer (Dosovitskiy et al., 2021) is utilized, the computational cost is quadratic to the spatio-temporal dimensions. This burden is nontrivial and sometimes unaffordable. Meanwhile, the global Transformer attends to redundant key elements, which may easily cause non-convergence issue (Zhu et al., 2020) and over-smoothing results (Li et al., 2019). **On the other hand**, when the local window-based Transformer (Liu et al., 2021) is used, the self-attention is calculated within position-specific windows, causing limited receptive fields. The model may neglect some key

[*]Equal contribution [1]Shenzhen International Graduate School, Tsinghua University [2]Huawei Noah's Ark Lab [3]ETH Zürich. Correspondence to: Haoqian Wang <wanghaoqian@tsinghua.edu.cn>, Xueyi Zou <zouxueyi@huawei.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

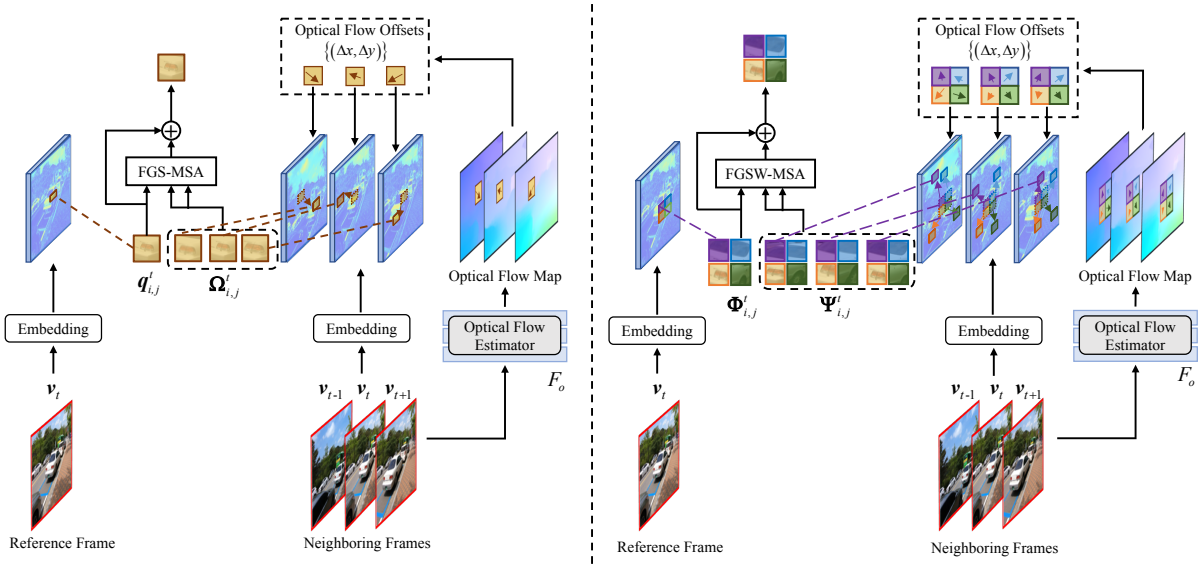

Figure 1. The illustration of our flow-guided self-attention mechanisms. (a) FGS-MSA globally samples spatially sparse yet highly related key elements of similar and sharper patches in neighboring frames. (b) Instead of sampling a single key element on each neighboring frame, FGSW-MSA robustly samples all the key elements corresponding to the $query$ elements of the window on the reference frame.

elements of similar and sharper scene patches in the spatio-temporal neighborhood when fast motions are present. We summarize the main reason for the above problems, *i.e.*, previous Transformers lack the guidance of motion information, when calculating self-attention. We note that the motion information can be estimated by optical flow.

Exploiting an optical flow estimator to capture motion information and align neighboring frames is a common strategy in video restoration (Makansi et al., 2017; Su et al., 2017; Xue et al., 2019; Pan et al., 2020). Previous flow-based methods mainly adopt the pre-warping strategy. Specifically, they employ an optical flow estimator to produce motion offsets, warp neighboring frames, and align regions corresponding to the same object but misaligned in neighboring image or feature domains. This scheme suffers from the following issues: **(i)** The interpolating operations in the warping module modify the original image information. As a result, some critical image priors such as self-similarity and sharp textures may be sacrificed. Undesirable artifacts may be introduced to the restored video and the deblurring performance may degrade. **(ii)** The frame alignment and subsequent representation aggregation are separated. This paradigm is inflexible and does not make full use of optical flow. Besides, the deblurring results are easily affected by the performance of the optical flow estimator. The robustness of this scheme can be further improved.

This work aims to cope with the above problems. We propose a novel method, Flow-Guided Sparse Transformer (FGST), for video deblurring. **Firstly**, we adopt Transformer instead of CNN as the deblurring model because of its advantages of capturing long-range spatial dependencies and non-local self-similarity. **Secondly**, to alleviate the limitations of previous Transformers and the pre-warping strategy, we customize Flow-Guided Sparse Multi-head Self-Attention (FGS-MSA) as shown in Fig. 1 (a). For each $query$ element on the reference frame, FGS-MSA guided by an optical flow estimator globally samples spatially sparse key elements corresponding to the same scene patch but misaligned in the neighboring frames. These sampled key elements provide self-similar and highly related image prior information, which is critical to restoring motion blur. Different from original global and local Transformers, our FGST neither blindly samples redundant key elements nor suffers from limited receptive fields. Meanwhile, our alignment scheme is different from the pre-warping operation mainly used by previous flow-based methods. Instead of warping the neighboring frames, our FGST samples key elements in consecutive frames to calculate the self-attention. Thus, the original image prior information can be preserved. **Thirdly**, we promote FGS-MSA to Flow-Guided Sparse Window-based Multi-head Self-Attention (FGSW-MSA) as shown in Fig. 1 (b). The feature maps are split into non-overlapping windows. Instead of sampling a single key element on each neighboring frame for a single $query$ element, FGSW-MSA samples key elements assigned by the optical flow corresponding to all the $query$ elements of the window on the reference frame. Thus, FGSW-MSA is more robust to accommodate pixel-level flow offset prediction deviations. **Finally**, our FGSW-MSA is calculated within a short tem-

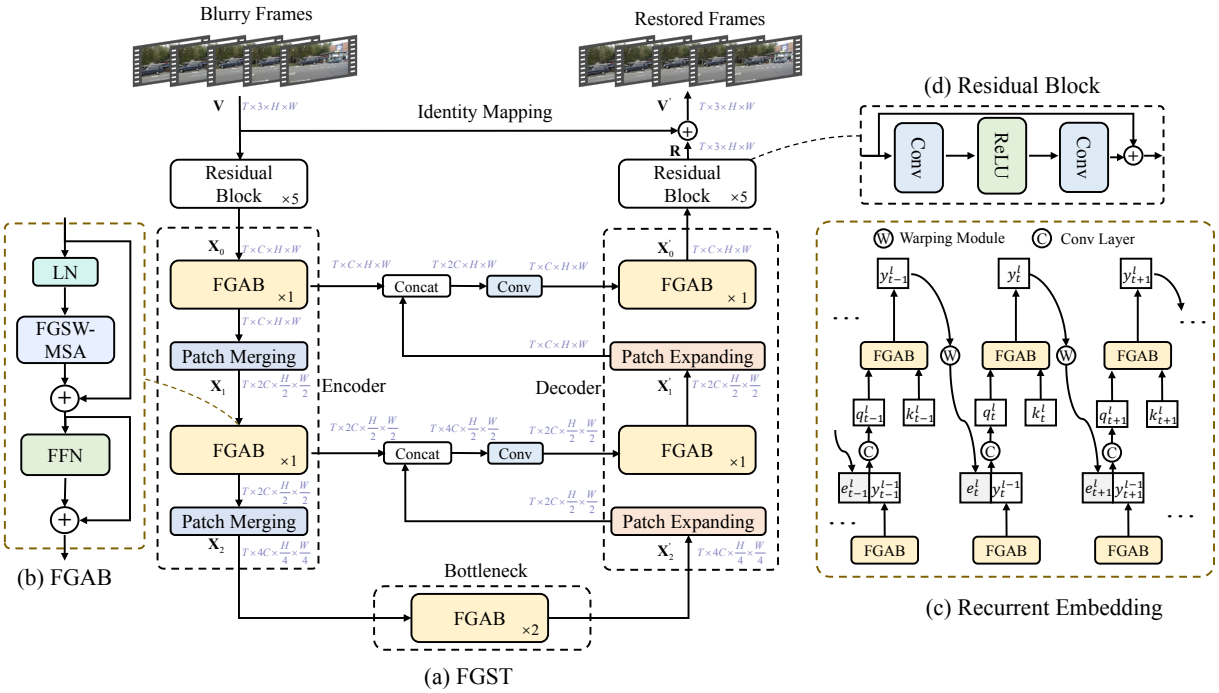

Figure 2. The architecture of FGST. (a) FGST consists of an encoder, a bottleneck, and a decoder. FGST is built up by FGABs. (b) FGAB is composed of a layer normalization, an FGSW-MSA, and a feed-forward network. (c) RE aggregates the output of the last frame and the input of the current frame. Some intermediate steps between FGABs are omitted. (d) The components of residual block.

poral sequence reducing the computational cost. Hence, the receptive field of FGSW-MSA is spatially global but temporally local. Motivated by RNN-based methods (Nah et al., 2019; Zhong et al., 2020), we propose Recurrent Embedding (RE) to transfer information of past frames and capture long-range temporal dependencies.

Our contributions can be summarized as follows:

- We propose a new method, FGST, for video deblurring. To the best of our knowledge, it's the first attempt to explore the potential of Transformer in this task.
- We customize a novel self-attention mechanism, FGS-MSA, and its improved version, FGSW-MSA.
- We design an embedding scheme, RE, to transfer frame information and capture temporal dependencies.
- Our FGST outperforms SOTA methods on DVD and GOPRO datasets by a large margin and yields more visually pleasing results in real-world video deblurring.

2. Related Work

2.1. Video Deblurring

In recent years, the deblurring research focus is shifting from single image deblurring (Zoran et al., 2011; Chakrabarti, 2016; Purohit et al., 2020) to the more challenging video deblurring (Cho et al., 2012; Matsushita et al., 2006). Traditional methods (Li et al., 2010; Zhang et al., 2013) are based on hand-crafted image priors and assumptions, which lead to limited generality and representing capacity. With the development of deep learning, recent methods are mainly CNN-based or RNN-based. (Zhang et al., 2018) employ 3D convolutions to model spatio-temporal relations of frames. (Hyun Kim et al., 2017) and (Nah et al., 2019) use RNN-based models to restore the latent frames. However, CNN-based methods show limitations in capturing long-range dependencies while RNN-based methods are not sensitive to patch-level spatial correlation and motion information.

2.2. Vision Transformer

Transformer is firstly proposed by (Vaswani et al., 2017) for machine translation. Recently, Transformer has been introduced to high-level (Dosovitskiy et al., 2021; Liu et al., 2021; Zhu et al., 2020; Zheng et al., 2021; Ali et al., 2021; Carion et al., 2020; Li et al., 2021b; Ramachandran et al., 2019; Wu et al., 2020; Cai et al., 2020) and low-level vision (Chen et al., 2021; Cai et al., 2022; Cao et al., 2021b; Cai et al., 2021; Hu et al., 2021). (Arnab et al., 2021) factorize the spatial and temporal dimensions of the input video and propose a Transformer model for video classification. (Chen et al., 2021) present a large model IPT pre-trained on large-scale datasets with a multi-task learning scheme. (Cao et al., 2021b) propose VSR-Transformer that performs feature fusion with the global multi-head self-attention mechanism for video super-resolution task. (Wang et al., 2022) use Swin Transformer (Liu et al., 2021) blocks to build up a U-shaped structure for single image restoration. In

(Vaswani et al., 2021; Cao et al., 2021a; Liu et al., 2021), window-based local self-attention is adopted to replace the global self-attention module of the standard Transformer. However, directly using previous global or local Transformers for video deblurring leads to unaffordable computational cost or limited receptive fields.

2.3. Flow-based Video Restoration

Optical flow estimators are widely used in video restoration tasks (Gast & Roth, 2019; Xue et al., 2019; Gong et al., 2017; Sun et al., 2015; Makansi et al., 2017; Su et al., 2017; Pan et al., 2020) to align highly related but mis-aligned frames. Previous flow-based video deblurring methods (Xue et al., 2019; Makansi et al., 2017; Su et al., 2017; Pan et al., 2020; Gast & Roth, 2019) mainly adopt the pre-warping strategy, which firstly estimates the optical flow and then warps the neighboring frames. For example, (Su et al., 2017) aligns the input images to the reference frame by pre-warping them based on optical flow methods. Nonetheless, this flow-based pre-warping scheme separates the frame alignment and subsequent information aggregation. The original frame information is sacrificed and the guidance effect of optical flow is not fully explored.

3. Method

3.1. Overall Architecture

Figure 2 (a) shows the architecture of FGST that adopts the widely used U-shaped structure, consisting of an encoder, a bottleneck, and a decoder. Figure 2 (b) depicts the basic unit of FGST, $i.e.$, Flow-Guided Attention Block (FGAB).

The input is a blurry video $\mathbf{V} \in \mathbb{R}^{T\times 3\times H\times W}$, where T denotes the sequence length, H and W denote the width and height of the frame. **Firstly**, FGST exploits 5 residual blocks to map $\mathbf{V}$ into tokens $\mathbf{X}_0 \in \mathbb{R}^{T\times C\times H\times W}$, where C denotes the channel number. The details of residual block are shown in Fig. 2 (d). **Secondly**, $\mathbf{X}_0$ passes through two FGABs and patch merging layers to generate hierarchical features. The patch merging layer is a strided 4×4 convolution that downsamples the feature maps and doubles the channels. Thus, the tokens of the i_{th} layer in the encoder are denoted as $\mathbf{X}_i \in \mathbb{R}^{T\times 2^i C\times \frac{H}{2^i}\times \frac{W}{2^i}}$. **Thirdly**, $\mathbf{X}_2$ passes through the bottleneck, which consists of two FGABs.

Subsequently, following the spirit of U-Net (Ronneberger et al., 2015), we customize a symmetrical decoder, which is composed of two FGABs and patch expanding layers. The patch expanding layer is a strided 2×2 deconvolution that upsamples the feature maps. To alleviate the information loss caused by downsampling, skip connections are used for feature fusion between the encoder and decoder.

After undergoing the decoder, the feature maps pass through 5 residual blocks to generate a residual frame sequence $\mathbf{R} \in \mathbb{R}^{T\times 3\times H\times W}$. **Finally**, the deblurred video $\mathbf{V}' \in \mathbb{R}^{T\times 3\times H\times W}$ can be derived by $\mathbf{V}' = \mathbf{V} + \mathbf{R}$.

3.2. Flow-Guided Attention Block

As analyzed in Sec. 1, the standard global Transformer brings quadratic computational complexity with respect to the token number and easily leads to non-convergence issue and over-smoothing results. The previous window-based local Transformers suffer from the limited receptive fields.

To address these problems, we propose to use optical flow as the guidance to sample key elements from spatio-temporal neighborhoods when calculating the self-attention. Based on this motivation, we customize the basic unit, FGAB as shown in Fig. 2 (b). FGAB consists of a layer normalization (LN), a Flow-Guided Sparse Window-based Multi-head Self-Attention (FGSW-MSA), a feed-forward network (FFN), and two identity mappings. The FFN is composed of 5 consecutive residual blocks. In this part, we first introduce Flow-Guided Sparse Multi-head Self-Attention (FGS-MSA) and then its improved version, FGSW-MSA.

FGS-MSA. The details of FGS-MSA are shown in Fig. 1 (a). Given the $\boldsymbol{t}_{th}$ input blurry video frame $\boldsymbol{v}_t \in \mathbb{R}^{3\times H\times W}$ as the reference frame, $\boldsymbol{q}^t_{i,j}$ and $\boldsymbol{k}^t_{i,j} \in \mathbb{R}^C$ respectively denote the *query* and *key* elements at the position (i,j) on $\boldsymbol{v}_t$. FGS-MSA aims to model long-range spatial dependencies and capture non-local self-similarity. To this end, FGS-MSA produces keys from the key elements of similar and sharper scene patches in the spatio-temporal neighborhood of $\boldsymbol{v}_t$. The key sampling is directed by the motion information that is predicted by an optical flow estimator. This set of key elements is corresponding to $\boldsymbol{q}^t_{i,j}$ and we denote it as

$$\boldsymbol{\Omega}^t_{i,j} = \{\boldsymbol{k}^f_{i+\Delta x_f, j+\Delta y_f} \mid |f-t| \le r\}, \tag{1}$$

where r represents the temporal radius of the neighboring frames. $(\Delta x_f, \Delta y_f)$ denotes the value at position (i,j) of the estimated motion offset map, which is predicted from the reference frame $\boldsymbol{v}_t$ to the neighboring frame $\boldsymbol{v}_f$:

$$(\Delta x_f, \Delta y_f) = [F_o(\boldsymbol{v}_t, \boldsymbol{v}_f)\ (i,j)], \tag{2}$$

where F_o denotes the mapping function of the optical flow estimator and $[\cdot]$ refers to the rounding operation. Subsequently, FGS-MSA can be formulated as

$$\text{FGS-MSA}(\boldsymbol{q}^t_{i,j}, \boldsymbol{\Omega}^t_{i,j}) = \sum_{n=1}^{N} \mathbf{W}_n \sum_{\boldsymbol{k}\in\boldsymbol{\Omega}^t_{i,j}} \mathbf{A}_{n\boldsymbol{q}^t_{i,j}\boldsymbol{k}}\, \mathbf{W'}_n\, \boldsymbol{k}, \tag{3}$$

where N is the number of the attention heads. $\mathbf{W}_n \in \mathbb{R}^{C\times d}$ and $\mathbf{W'}_n \in \mathbb{R}^{d\times C}$ are learnable parameters, where $d = \frac{C}{N}$ denotes the representation dimension per head. $\mathbf{A}_{n\boldsymbol{q}^t_{i,j}\boldsymbol{k}}$ is the self-attention of the n_{th} head, which is formulated as

$$\mathbf{A}_{n\boldsymbol{q}^t_{i,j}\boldsymbol{k}} = \underset{\boldsymbol{k}\in\boldsymbol{\Omega}^t_{i,j}}{\text{softmax}}\left(\frac{(\boldsymbol{q}^t_{i,j})^T \mathbf{U}_n^T \mathbf{V}_n \boldsymbol{k}}{\sqrt{d}}\right), \tag{4}$$

where $\mathbf{U}_n$ and $\mathbf{V}_n \in \mathbb{R}^{d\times C}$ are learnable parameters. Given an input $\mathbf{V} \in \mathbb{R}^{T\times 3\times H\times W}$, the computational cost of the global MSA (Dosovitskiy et al., 2021) and FGS-MSA are

$$O(\text{global MSA}) = 4(THW)C^2 + 2(THW)^2C,$$
$$O(\text{FGS-MSA}) = 2(THW)C\big(2(r+1)C + 2r + 1\big). \quad (5)$$

The standard global MSA leads to quadratic ($(THW)^2$) computational complexity while our proposed FGS-MSA contributes to much cheaper linear computational cost with respect to the token number (THW). Detailed analysis are provided in the supplementary material (SM).

FGSW-MSA. For each neighboring frame, FGS-MSA only samples a single key element. When the optical flow estimation is inaccurate, the deblurring performance may be easily affected. To further improve the robustness and reliability of our method, we promote FGS-MSA to FGSW-MSA. As shown in Fig. 1 (b), the feature maps are split into non-overlapping windows. The spatial size of each window is $M \times M$. $\mathbf{\Phi}^t_{i,j}$ denotes the set of $query$ elements in the window centering at position (i, j) of the t_{th} frame:

$$\mathbf{\Phi}^t_{i,j} = \{\boldsymbol{q}^t_{m,n} \big|\, |m-i| \le M/2, |n-j| \le M/2\}. \quad (6)$$

For each $\boldsymbol{q}^t_{m,n} \in \mathbf{\Phi}^t_{i,j}$, FGSW-MSA samples not only its corresponding key elements in $\mathbf{\Omega}^t_{m,n}$ (Eq. (1)) assigned by the flow offsets but also the key elements corresponding to other $query$ elements in $\mathbf{\Phi}^t_{i,j}$. We denote the set of these key elements as $\mathbf{\Psi}^t_{i,j}$, which can be formulated as

$$\mathbf{\Psi}^t_{i,j} = \bigcup_{|m-i|\le M/2,\ |n-j|\le M/2} \mathbf{\Omega}^t_{m,n}. \quad (7)$$

Instead of attending to a single key element on each neighboring frame for a single $query$, FGSW-MSA pays attention to the key elements from similar and sharper scene patches corresponding to all $query$ elements in $\mathbf{\Phi}^t_{i,j}$. The attending region is enlarged from pixel to window. Thus, FGSW-MSA is more robust to accommodate pixel-level flow prediction deviations. FGSW-MSA can be formulated as

$$\text{FGSW-MSA}(\mathbf{\Phi}^t_{i,j}, \mathbf{\Psi}^t_{i,j}) = \{\text{FGS-MSA}(\boldsymbol{q}, \mathbf{\Psi}^t_{i,j}) | \boldsymbol{q} \in \mathbf{\Phi}^t_{i,j}\}. \quad (8)$$

Given the input $\mathbf{V}$, the computational complexity is

$$O(\text{FGSW-MSA}) = 2(THW)C\big(C + (2r+1)(C+M^2)\big). \quad (9)$$

The computational cost of FGSW-MSA is linear with respect to the number of tokens (THW). Eq. (5) and (9) reveal the high efficiency and resource economy of our FGST. Please refer to the SM for more detailed analysis.

Discussion. **(i)** Our FGSW-MSA enjoys much larger receptive fields than W-MSA (Liu et al., 2021). Specifically, according to Eq. (1), (2), (6), and (7), the receptive field of FGSW-MSA can cover the whole input feature map when the estimated flow offset is large enough. In practice, the motion offset predicted by the optical flow estimator between two adjacent frames can reach 40 and 38 pixels on GOPRO and DVD datasets. The input spatial size is 256×256. M is set to 3. Thus, the receptive field of FGSW-MSA can reach 83×83 (83 = 40×2+3) and 79×79 while that of W-MSA is still 3×3. **(ii)** Unlike previous flow-based methods that adopt the pre-warping operation sacrificing the original image information as shown in Fig. 3, our FGST combines motion cues with self-attention calculation. Thus, the original image information can be preserved and the guidance effect of the optical flow can be further explored. In addition, our flow-guided scheme enjoys higher flexibility and robustness because adjacent FGABs sample contents independently. Please refer to the SM for detailed discussions.

Figure 3. The pre-warping strategy mainly adopted by previous video deblurring methods sacrificies the input image information.

3.3. Recurrent Embedding

Our FGSW-MSA is calculated within a short temporal sequence for the computational complexity consideration (approximately linear to the temporal radius r in Eq. (9)). Therefore, the receptive field of FGSW-MSA is temporally local and overlooking the distant frames limits the video deblurring performance. To further capture more robust long-range temporal dependencies, we propose Recurrent Embedding (RE) mechanism. RE is motivated by Recurrent Neural Network (RNN). More specifically, as shown in Fig. 2 (c), we exploit RE in each Transformer layer to transfer information from past frames and establish long-range temporal correlations. With RE, the FGAB is calculated in a recurrent manner for T time steps. $\boldsymbol{y}^l_t$, $\boldsymbol{e}^l_t, \boldsymbol{q}^l_t$, $\boldsymbol{k}^l_t$ respectively denote the output, RE, $query$ elements, and key elements of the l_{th} FGAB in the t_{th} time step. We have

$$\boldsymbol{e}^l_t = f_w(\boldsymbol{y}^l_{t-1}),\ \ \boldsymbol{q}^l_t = f_c([\boldsymbol{e}^l_t, \boldsymbol{y}^{l-1}_t]),$$
$$\boldsymbol{k}^l_t = \bigcup_{|j-t|\le r} \boldsymbol{y}^{l-1}_j,\ \ \boldsymbol{y}^l_t = \text{FGAB}(\boldsymbol{q}^l_t, \boldsymbol{k}^l_t), \quad (10)$$

where $f_w(\cdot)$ represents the spatial warping that align the feature map at t and $t-1$ time step, $[\cdot,\cdot]$ is the concatenating operation, $f_c(\cdot)$ denotes 3×3 convolution to aggregate the recurrent embedding $\boldsymbol{e}^l_t$ and the output from last FGAB layer $\boldsymbol{y}^{l-1}_t$, and $\boldsymbol{y}^l_t = \text{FGAB}(\boldsymbol{q}^l_t, \boldsymbol{k}^l_t)$ is formulated in details as

$$\boldsymbol{o}^l_t = \text{FGSW-MSA}(\text{LN}(\boldsymbol{q}^l_t), \text{LN}(\boldsymbol{k}^l_t)) + \boldsymbol{q}^l_t,$$
$$\boldsymbol{y}^l_t = \text{FFN}(\boldsymbol{o}^l_t) + \boldsymbol{o}^l_t, \quad (11)$$

where LN denotes the layer normalization and FFN refers to the Feed Forward Network. Our RE sequentially propagates the information from the first frame to the last frame, thus capturing reliable long-range temporal dependencies.

Figure 4. Visual comparisons between FGST and SOTA methods on DVD dataset (Su et al., 2017). Please zoom in for a better view.

Method	Kim and Lee (Kim et al., 2015)	Gong *et al.* (Gong et al., 2017)	Su *et al.* (Su et al., 2017)	Kim *et al.* (Hyun Kim et al., 2017)	STFAN (Zhou et al., 2019)	Xiang *et al.* (Xiang et al., 2020)	TSP (Pan et al., 2020)	Suin *et al.* (Suin et al., 2021)	ARVo (Li et al., 2021a)	**FGST (Ours)**
PSNR ↑	26.94	28.27	30.01	29.95	31.15	31.68	32.13	32.53	32.80	**33.36**
SSIM ↑	0.816	0.846	0.888	0.869	0.905	0.916	0.927	0.947	0.935	**0.950**

Table 1. Video deblurring results compared with other methods on the DVD benchmark (Su et al., 2017). FGST achieves SOTA results.

4. Experiment

4.1. Datasets

DVD. The DVD (Su et al., 2017) dataset consists of 71 videos with 6,708 blurry-sharp image pairs. It is divided into `train/test` subsets with 61 videos (5,708 image pairs) and 10 videos (1,000 image pairs). DVD is captured with mobile phones and DSLR at a frame rate of 240 fps.

GOPRO. The GOPRO (Nah et al., 2017) benchmark is composed of over 3,300 blurry-sharp image pairs of dynamic scenes. It is obtained by a high-speed camera. The training and testing subsets are split in proportional to 2:1.

Real Blurry Videos. To validate the generality of FGST, we evaluate models on real blurry datasets collected by (Cho et al., 2012). Because the ground truth (GT) is inaccessible, we only compare visual results of FGST and others.

4.2. Implementation Details

We implement FGST in PyTorch. We adopt a pre-trained SPyNet (Ranjan et al., 2017) as the optical flow estimator. All the modules are trained with the Adam (Kingma & Ba, 2015) optimizer ($\beta_1 = 0.9$ and $\beta_2 = 0.999$) for 600 epochs. The initial learning rate is set to 2×10^{-4} and 2.5×10^{-5} respectively for the deblurring model and optical flow estimator. The learning rate is halved every 200 epochs during the training procedure. Patches at the size of 256×256 cropped from training frames are fed into the models. The batch size is 8. The temporal radius r of the neighboring frames is set to 1. The sequence length T is set to 9 in training and the whole video length in testing. The horizontal and vertical flips are performed for data augmentation. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) (Wang et al., 2004) are adopted as the evaluation metrics. The models are trained with 8 V100 GPUs. $\mathcal{L}_1$ loss between the restored and GT videos is used for supervision.

4.3. Quantitative Results

The comparisons between FGST and other SOTA methods are listed in Tabs. 1, 2, and 3c. As can be observed: **(i)** Our FGST outperforms SOTA methods by a large margin on the two benchmarks. Specifically, as shown in Tab. 1, our FGST surpasses the recent best algorithm ARVo (Li et al., 2021a) by 0.56 dB on DVD. As reported in Tab. 2, our method exceeds Suin *et al.* (Suin et al., 2021) and TSP (Pan et al., 2020) by 0.80 dB and 1.23 dB respectively on GOPRO. These results demonstrate the effectiveness of our method. **(ii)** Tab. 3c exhibits efficiency comparisons of different algorithms on GOPRO. The FLOPS is tested at the input size of $1\times3\times240\times240$. The running time per frame is tested at the spatial size of $1{,}280\times720$ on the same RTX 2080 GPU. Our FGST is more cost-effective and achieves a better trade-off between PSNR, Params, FLOPS, and inference speed. For instance, when compared to TSP (Pan et al., 2020), FGST only requires 59.9% (9.70 / 16.19) Params and 36.8% (131.6 / 357.9) FLOPS while achieving even 1.23 dB improvement and 2.34× (579.7 / 247.8) speed. This evidence suggests the promising efficiency advantage of our proposed FGST.

4.4. Qualitative Results

We provide visual comparisons on DVD, GOPRO, and real blurry videos as shown in Figs. 4, 5, and 7. Previous methods are less favorable to restore abrupt motion blur. They either yield over-smoothing images sacrificing fine textural details and structural contents or introduce redundant blotchy texture and chromatic artifacts when fast motions exists. In contrast, our FGST excels at modeling long-range dependencies and exploits motion information to guide the self-attention module to capture non-local self-similarity in spatio-temporal neighborhoods. As a result, FGST is capable of restoring structural contents and textural details while preserving spatial smoothness of the homogeneous regions. Supplementary file provides more visual results.

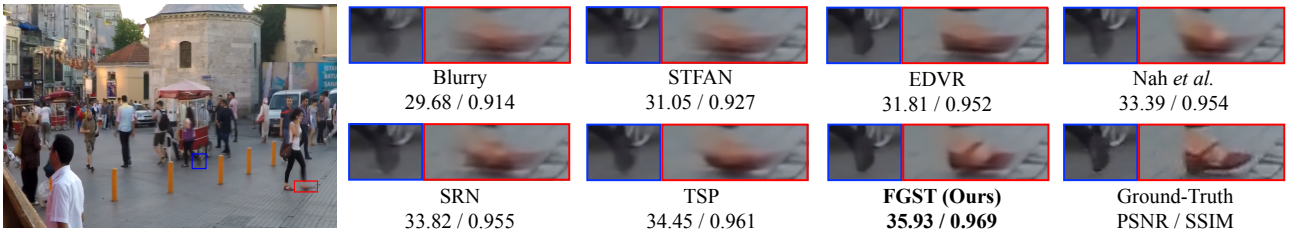

Figure 5. Visual comparisons between our FGST and SOTA methods on GOPRO dataset (Nah et al., 2017). Zoom in for a better view.

Method	Gong *et al.* (Gong et al., 2017)	Kim *et al.* (Hyun Kim et al., 2017)	EDVR (Wang et al., 2019)	Su *et al.* (Su et al., 2017)	STFAN (Zhou et al., 2019)	Nah *et al.* (Nah et al., 2019)	Tao *et al.* (Tao et al., 2018)	TSP (Pan et al., 2020)	Suin *et al.* (Suin et al., 2021)	**FGST (Ours)**
PSNR ↑	26.06	26.82	26.83	27.31	28.59	29.97	30.29	31.67	32.10	**32.90**
SSIM ↑	0.863	0.825	0.843	0.826	0.861	0.895	0.901	0.928	0.960	**0.961**

Table 2. Video deblurring results compared with other methods on the GOPRO dataset (Nah et al., 2017). FGST achieves SOTA results.

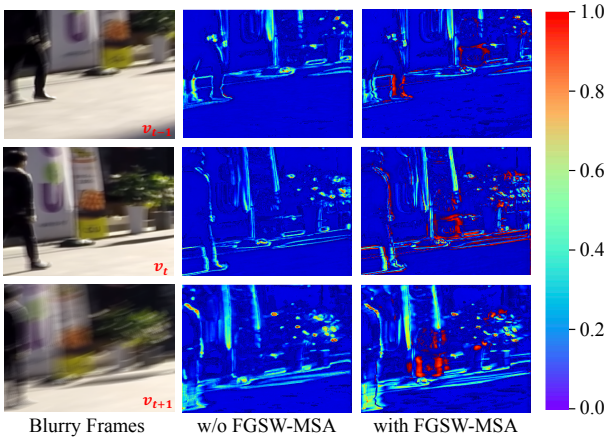

Figure 6. We visualize the last feature maps of the deblurring models with and without FGSW-MSA. The model using our FGSW-MSA pays more attention to similar but misaligned scene patches.

4.5. Ablation Study

In this part, we conduct ablation studies on GOPRO dataset. The baseline model is derived by directly removing all the proposed RE and FGSW-MSA modules from our FGST.

Break-down Ablation. We firstly conduct a break-down ablation to investigate the effect of each component toward better performance. The results are reported in Tab. 3a. The baseline model yields 31.18 dB. After applying RE and FGSW-MSA respectively, the deblurring model achieves 1.16 dB and 1.66 dB improvements. While using both RE and FGSW-MSA modules, the model gains by 1.72 dB. The results suggest the effectiveness of RE and FGSW-MSA.

Self-Attention Mechanism. We compare our self-attention mechanisms with other competitors in Tab. 3b. The baseline model yields 31.18 dB while costing 5.15M Params and 43.93G FLOPS. **(i)** When using global MSA (Dosovitskiy et al., 2021), the feature maps are downsampled into $\frac{1}{4}$ size and the channel is increased by 4 times to avoid out of memory and information loss. The deblurring model degrades by 1.98 dB while costing 12.5× Params and 3.2× FLOPS. This is mainly because global MSA attends to too redundant key elements, requiring a large amount of computation and memory resources while leading to ambiguous gradients for input features (Zhu et al., 2020) and thus non-convergence problem. Meanwhile, features from global aggregation tend to over-smooth the predictions of small patterns (Li et al., 2019). **(ii)** When using local W-MSA (Liu et al., 2021), the model gains by only 0.53 dB while adding 3.11M Params and 64.16G FLOPS. The improvement is limited while the additional burden is nontrivial. That is because W-MSA calculates self-attention within position-specific windows. The receptive field is limited. **(iii)** Our FGS-MSA exploits the optical flow as the guidance to sample spatially sparse $keys$ of similar and sharper regions in the spatio-temporal neighborhood for each $query$ on the reference frame. Compared to global MSA, the key elements of FGST are less but highly related to the selected $query$. Thus, when using FGS-MSA, the model gains by 1.30 dB while adding 4.54M Params and 81.15G FLOPS. These results show that FGS-MSA costs cheaper resources but achieves better performance than global MSA. When exploiting FGSW-MSA, the model yields an improvement of 1.72 dB while adding 4.55M Params and 87.69G FLOPS. This evidence suggests: **(a)** FGSW-MSA is more effective than W-MSA in fast motion blur restoration. **(b)** FGSW-MSA is more reliable than FGS-MSA and achieves better deblurring performance.

In addition, we conduct visual analysis on three adjacent frames by visualizing the last feature map of models with and without (w/o) FGSW-MSA in Fig. 6. Deeper color indicates larger weights. It can be observed that the model without FGSW-MSA responds weakly to similar regions in the neighboring frames. In contrast, the model equipped with FGSW-MSA generates much stronger responses to highly related but misaligned scene patches. Moreover, FGST pays more attention to the regions with fast motion blur. These results demonstrates the effectiveness of FGSW-MSA in capturing non-local self-similarity in dynamic scenes.

Flow-Guided Deformable Convolution. We compare our FGSW-MSA with deformable convolution (DeConv) (Wang et al., 2019) and recent flow-guided deformable convolution (FGDeConv) (Chan et al., 2021) in Tab. 3d. Our proposed

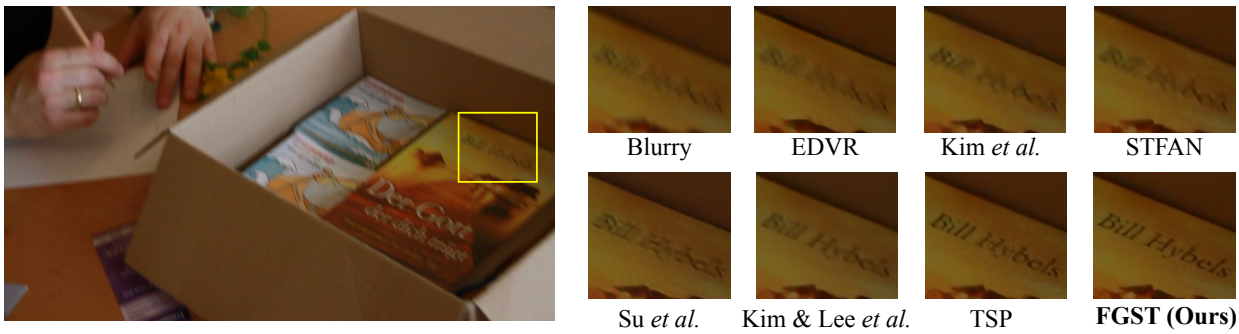

Figure 7. Visual results of FGST and SOTA methods on the real blurry videos of (Cho et al., 2012). Please zoom in for a better view.

Baseline	RE	FGSW-MSA	PSNR ↑	SSIM ↑
✓			31.18 (+0.00%)	0.924 (+0.00%)
✓	✓		32.34 (+3.72%)	0.943 (+2.06%)
✓		✓	32.84 (+5.32%)	0.957 (+3.57%)
✓	✓	✓	**32.90 (+5.52%)**	**0.961 (+4.00%)**

(a) Break-down ablation study toward better performance.

Method	Baseline	Global MSA	Local W-MSA	FGS-MSA	FGSW-MSA
PSNR	31.18	29.20	31.71	**32.48**	**32.84**
SSIM	0.924	0.880	0.938	**0.944**	**0.957**
Params	5.15	64.40	8.26	9.69	9.70
FLOPS	43.93	138.68	108.09	125.08	125.67

(b) Ablation study of using different self-attention mechanisms.

Method	EDVR	Su *et al.*	STFAN	TSP	FGST (Ours)
PSNR	26.83	27.31	28.59	31.67	32.90
Params (M)	23.60	15.30	5.37	16.19	9.70
FLOPS (G)	159.2	38.7	35.4	357.9	131.6
Time (ms/f)	268.5	133.2	145.9	579.7	247.8

(c) Efficiency comparisons with SOTA CNN-based methods.

Method	Baseline	+ DeConv	+ FGDeConv	+ FGSW-MSA
PSNR	31.18	32.35	32.59	**32.84**
SSIM	0.924	0.941	0.954	**0.957**
Params (M)	5.15	8.34	9.78	9.70
FLOPS (G)	43.93	108.38	125.96	125.67

(d) FGSW-MSA *v.s.* FGDeConv and DeConv on GOPRO dataset.

Method	Local W-MSA	pre-warping	FGSW-MSA
PSNR	31.71	32.54	**32.84**
SSIM	0.938	0.953	**0.957**

(e) Pre-warping *v.s.* our FGSW-MSA.

Win Size	1×1	2×2	3×3	4×4	5×5
PSNR	32.48	32.62	**32.90**	32.71	32.66
SSIM	0.944	0.955	**0.961**	0.955	0.957

(f) Ablation study of window sizes.

Method	Baseline	FlowNet	SPyNet	PWC-Net
PSNR ↑	31.18	32.85	32.90	**33.03**
SSIM ↑	0.924	0.960	0.961	**0.964**

(g) Ablation study of optical flow estimators.

Table 3. Ablation studies. The models are trained and tested on GOPRO. PSNR, SSIM, Params, FLOPS, and inference time are reported.

FGSW-MSA achieves the most significant improvement. This mainly stems from that FGSW-MSA excels at capturing non-local similarity and long-range dependencies, which are the limitations of CNN-based methods.

Pre-warping Strategy. We compare our FGSW-MSA with the pre-warping strategy mainly adopted by previous methods in Tab. 3e. We start from the baseline model equipped with W-MSA. It can be observed that using FGSW-MSA is 0.30 dB and 0.004 in terms of PSNR and SSIM higher than using pre-warping operation. This performance gap is mainly because the model using our FGSW-MSA can learn from non-corrupted representations of input video and further explore the guidance effect of the optical flow.

Window Size. We change the window size of FGSW-MSA to study its effect. The results are listed in Tab. 3f. We start by setting the window size at 1×1 and then gradually increase it. The performance achieves its maximum when the window size is 3×3. Thus, the optimal setting is 3×3.

Optical Flow Estimator. We respectively adopt three representative optical flow estimators (FlowNet (Dosovitskiy et al., 2015), SPyNet (Ranjan et al., 2017), and PWC-Net (Sun et al., 2018)) to investigate their effects as shown in Tab. 3g. **(i)** No matter what optical flow estimator is used, our FGST reliably outperforms the baseline model, suggesting the robustness and generality of our method. **(ii)** The performance of FGST can be further improved by using a better optical flow estimator. To be specific, when equipped with PWC-Net, FGST is 0.18 dB and 0.13 dB higher than those using FlowNet and SPyNet respectively. These results demonstrate that our FGST can directly and conveniently enjoy the benefits of SOTA optical flow estimators.

5. Conclusion

In this paper, we propose a novel Transformer-based method, FGST, for video deblurring. In FGST, we customize a self-attention mechanism, FGS-MSA, and then promote it to FGSW-MSA. Guided by an optical flow estimator, FGSW-MSA samples spatially sparse but highly related key elements corresponding to similar and sharper scene patches in the spatio-temporal neighborhoods. Besides, we present an embedding scheme, RE, to transfer information of past frames and capture long-range temporal dependencies. Comprehensive experiments demonstrate that our FGST significantly surpasses SOTA methods and generates more visually pleasant results in real video deblurring.

Acknowledgements: This work is partially supported by the NSFC fund (61831014), the Shenzhen Science and Technology Project under Grant (CJGJZD20200617102601004, JSGG20210802153150005).

References

Ali, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al. Xcit: Cross-covariance image transformers. In *NeurIPS*, 2021.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., and Schmid, C. Vivit: A video vision transformer. In *CVPR*, 2021.

Cai, Y., Wang, Z., Luo, Z., Yin, B., Du, A., Wang, H., Zhou, X., Zhou, E., Zhang, X., and Sun, J. Learning delicate local representations for multi-person pose estimation. In *ECCV*, 2020.

Cai, Y., Hu, X., Wang, H., Zhang, Y., Pfister, H., and Wei, D. Learning to generate realistic noisy images via pixel-level noise-aware adversarial training. In *NeurIPS*, 2021.

Cai, Y., Lin, J., Hu, X., Wang, H., Yuan, X., Zhang, Y., Timofte, R., and Van Gool, L. Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In *CVPR*, 2022.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., and Wang, M. Swin-unet: Unet-like pure transformer for medical image segmentation. *arXiv preprint arXiv:2105.05537*, 2021a.

Cao, J., Li, Y., Zhang, K., and Van Gool, L. Video super-resolution transformer. *arXiv preprint arXiv:2106.06847*, 2021b.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *ECCV*, 2020.

Chakrabarti, A. A neural approach to blind motion deblurring. In *ECCV*, 2016.

Chan, K. C., Zhou, S., Xu, X., and Loy, C. C. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. 2021.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. Pre-trained image processing transformer. In *CVPR*, 2021.

Cho, S., Wang, J., Lee, S., b, and c. Video deblurring for hand-held cameras using patch-based synthesis. In *ACM TOG*, 2012.

Dosovitskiy, A., Fischer, P., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., v.d. Smagt, P., Cremers, D., and Brox, T. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Gast, J. and Roth, S. Deep video deblurring: The devil is in the details. In *ICCVW*, 2019.

Gong, D., Yang, J., Liu, L., Zhang, Y., Reid, I., Shen, C., Van Den Hengel, A., and Shi, Q. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *CVPR*, 2017.

Hu, X., Ma, R., Liu, Z., Cai, Y., Zhao, X., Zhang, Y., and Wang, H. Pseudo 3d auto-correlation network for real image denoising. In *CVPR*, 2021.

Hyun Kim, T., Mu Lee, K., Scholkopf, B., and Hirsch, M. Online video deblurring via dynamic temporal blending network. In *ICCV*, 2017.

Jin, H., Favaro, P., Cipolla, R., b, and c. Visual tracking in the presence of motion blur. In *CVPR*, 2005.

Kim, H., Mu Lee, K., a, and b. Generalized video deblurring for dynamic scenes. In *CVPR*, 2015.

Kingma, D. P. and Ba, J. L. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Li, D., Xu, C., Zhang, K., Yu, X., Zhong, Y., Ren, W., Suominen, H., and Li, H. Arvo: Learning all-range volumetric correspondence for video deblurring. In *CVPR*, 2021a.

Li, K., Wang, S., Zhang, X., Xu, Y., Xu, W., and Tu, Z. Pose recognition with cascade transformers. In *CVPR*, 2021b.

Li, X., Zhang, L., You, A., Yang, M., Yang, K., and Tong, Y. Global aggregation then local distribution in fully convolutional networks. In *BMVC*, 2019.

Li, Y., Kang, S. B., Joshi, N., Seitz, S. M., and Huttenlocher, D. P. Generating sharp panoramas from motion-blurred videos. In *CVPR*, 2010.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

Makansi, O., Ilg, E., Brox, T., b, and c. End-to-end learning of video super-resolution with motion compensation. In *GCPR*, 2017.

Matsushita, Y., Ofek, E., Ge, W., Tang, X., and Shum, H.-Y. Full-frame video stabilization with motion inpainting. *TPAMI*, 2006.

Nah, S., Hyun Kim, T., Mu Lee, K., and b. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017.

Nah, S., Son, S., Lee, K. M., and b. Recurrent neural networks with intra-frame iterations for video deblurring. In *CVPR*, 2019.

Pan, J., Bai, H., Tang, J., and b. Cascaded deep video deblurring using temporal sharpness prior. In *CVPR*, 2020.

Purohit, K., Rajagopalan, A., b, and c. Region-adaptive dense network for efficient motion deblurring. In *AAAI*, 2020.

Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., and Shlens, J. Stand-alone self-attention in vision models. In *NeurIPS*, 2019.

Ranjan, A., Black, M. J., b, and c. Optical flow estimation using a spatial pyramid network. In *CVPR*, 2017.

Ronneberger, O., Fischer, P., Brox, T., a, and b. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., and Wang, O. Deep video deblurring for hand-held cameras. In *CVPR*, 2017.

Suin, M., Rajagopalan, A. N., b, and c. Gated spatio-temporal attention-guided video deblurring. In *CVPR*, 2021.

Sun, D., Yang, X., Liu, M.-Y., and Kautz, J. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR*, 2018.

Sun, J., Cao, W., Xu, Z., and Ponce, J. Learning a convolutional neural network for non-uniform motion blur removal. In *CVPR*, 2015.

Tao, X., Gao, H., Shen, X., Wang, J., and Jia, J. Scale-recurrent network for deep image deblurring. In *CVPR*, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B., and Shlens, J. Scaling local self-attention for parameter efficient visual backbones. In *CVPR*, 2021.

Wang, X., Chan, K. C., Yu, K., Dong, C., and Change Loy, C. Edvr: Video restoration with enhanced deformable convolutional networks. In *CVPRW*, 2019.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncell, E. P. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004.

Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., and Li, H. Uformer: A general u-shaped transformer for image restoration. In *CVPR*, 2022.

Wu, B., Xu, C., Dai, X., Wan, A., Zhang, P., Yan, Z., Tomizuka, M., Gonzalez, J., Keutzer, K., and Vajda, P. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020.

Xiang, X., Wei, H., Wai, H., and Pan, J. Deep video deblurring using sharpness features from exemplars. *TIP*, 2020.

Xue, T., Chen, B., Wu, J., Wei, D., and Freeman, W. T. Video enhancement with task-oriented flow. *IJCV*, 2019.

Yin, T., Zhou, X., Krähenbühl, P., b, and c. Center-based 3d object detection and tracking. In *CVPR*, 2021.

Zhang, H., Wipf, D., and Zhang, Y. Multi-image blind deblurring using a coupled adaptive sparse prior. In *CVPR*, 2013.

Zhang, K., Luo, W., Zhong, Y., Lin Ma, W. L., and Li, H. Adversarial spatio-temporal learning for video deblurring. *TIP*, 2018.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P. H., and Zhang, L. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, 2021.

Zhong, Z., Gao, Y., Zheng, Y., and Zheng, B. Efficient spatio-temporal recurrent neural network for video deblurring. In *ECCV*, 2020.

Zhou, S., Zhang, J., Pan, J., Xie, H., Zuo, W., and Ren, J. Spatio-temporal filter adaptive network for video deblurring. In *ICCV*, 2019.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., and Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

Zoran, D., Weiss, Y., b, and c. From learning models of natural image patches to whole image restoration. In *ICCV*, 2011.