VideoFlow: Exploiting Temporal Cues for Multi-frame Optical Flow Estimation

Xiaoyu Shi[1,2] Zhaoyang Huang[1,2*] Weikang Bian[1] Dasong Li[1] Manyuan Zhang[1]
Ka Chun Cheung[2] Simon See[2] Hongwei Qin[3] Jifeng Dai[4] Hongsheng Li[1,5,6*]

[1]Multimedia Laboratory, The Chinese University of Hong Kong
[2]NVIDIA AI Technology Center [3]SenseTime Research [4]Tsinghua University
[5]Centre for Perceptual and Interactive Intelligence (CPII) [6]Shanghai AI Laboratory
{xiaoyushi@link, drinkingcoder@link, hsli@ee}.cuhk.edu.hk

Abstract

We introduce VideoFlow, a novel optical flow estimation framework for videos. In contrast to previous methods that learn to estimate optical flow from two frames, VideoFlow concurrently estimates bi-directional optical flows for multiple frames that are available in videos by sufficiently exploiting temporal cues.

We first propose a TRi-frame Optical Flow (TROF) module that estimates bi-directional optical flows for the center frame in a three-frame manner. The information of the frame triplet is iteratively fused onto the center frame. To extend TROF for handling more frames, we further propose a MOtion Propagation (MOP) module that bridges multiple TROFs and propagates motion features between adjacent TROFs. With the iterative flow estimation refinement, the information fused in individual TROFs can be propagated into the whole sequence via MOP. By effectively exploiting video information, VideoFlow presents extraordinary performance, ranking 1st on all public benchmarks. On the Sintel benchmark, VideoFlow achieves 1.649 and 0.991 average end-point-error (AEPE) on the final and clean passes, a 15.1% and 7.6% error reduction from the best published results (1.943 and 1.073 from FlowFormer++). On the KITTI-2015 benchmark, VideoFlow achieves an F1-all error of 3.65%, a 19.2% error reduction from the best published result (4.52% from FlowFormer++). Code is released at https://github.com/XiaoyuShi97/VideoFlow.

1. Introduction

Optical flow estimation is a fundamental computer vision task of estimating pixel-wise displacement fields between consecutive video frames. It is widely adopted to tackle various downstream video problems, including video restoration [32, 66, 12, 34, 5, 50, 72, 36], video object detection [78, 61, 79], video synthesis [69, 18, 19, 63, 62] and action recognition [58, 47, 76], providing valuable information on motion and cross-frame correspondence.

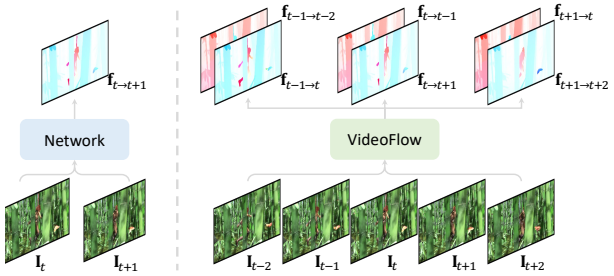


Figure 1. **Comparison between two-frame and multi-frame optical flow estimation.** (Left) Previous methods are limited to optical flow estimation of frame pairs. (Right) We introduce VideoFlow, a novel framework that concurrently estimates bi-directional optical flows for multiple consecutive frames.

With the evolvement of dedicated datasets [27] and model designs [21, 59], optical flow estimation has been greatly advanced over the years. However, we observe a divergence in its development from downstream demands. On the one hand, despite multiple frames being available in video streams, most efforts in this field are limited to flow estimation based on frame pairs, thereby ignoring valuable temporal cues from additional neighboring frames. Notably, a simple strategy for temporal information fusion, *e.g.*, the "warm-start" in RAFT [59], brings non-trivial performance gain. On the other hand, multi-frame bi-directional optical flows are imperative in many downstream video processing algorithms [5, 6, 46, 66, 39, 73, 23, 37, 35, 42, 41]. However, due to the lack of appropriate multi-frame optical flow models, existing algorithms have

*Corresponding author: Zhaoyang Huang and Hongsheng Li

to repeatedly estimate flows in a pair-wise manner. This highlights the need for optical flow models specifically designed for multi-frame scenarios.

In this paper, we introduce VideoFlow, as shown in Fig. 1, a novel framework that concurrently estimates optical flows for multiple consecutive frames. VideoFlow consists of two novel modules: 1) a TRi-frame Optical Flow module (TROF) that jointly estimates bi-directional optical flows for three consecutive frames in videos, and 2) a MOtion Propagation (MOP) module that splices TROFs for multi-frame optical flow estimation.

Specifically, we treat three-frame optical flow estimation as the basic unit for the multi-frame framework. We argue that the center frame of the triplet plays the key role of bridging temporal information, which motivates two critical designs of our proposed TROF model. Firstly, we propose to jointly estimate optical flows from the center frame to its two adjacent previous and next frames, which ensures the two flows originate from the same pixel and belong to the same continuous trajectory in the temporal dimension. Some previous three-frame methods [59, 49] rely on warping flow estimation from the preceding frame pair to facilitate the estimation of the current frame pair. The key difference arises in the presence of occlusion and out-of-boundary pixels, where the warped preceding predictions and current predictions might belong to entirely different objects. Such misalignment wastes valuable temporal cues and even introduces erroneous motion information. Secondly, TROF comprehensively integrates the bi-directional motion information in a recurrent updating process.

After constructing the strong three-frame model, we further propose a MOtion Propagation (MOP) module that extends our framework to handle more frames. This module passes bi-directional motion features along the predicted "flow trajectories". Specifically, in the recurrent updating decoder, MOP warps the bi-directional motion features of each TROF unit to its adjacent units according to current predicted bi-directional optical flows. The temporal receptive field grows as the recurrent process iterates so that our VideoFlow gradually utilizes wider temporal contexts to optimize all optical flow predictions jointly. This brings a significant advantage that the ambiguity and inadequate information in two-frame optical flow estimation will be primarily reduced when the information from multiple frames is sufficiently employed. For example, estimating optical flows for regions that are occluded or out of view in target images is too challenging in the two-frame setting but can be effectively improved when we take advantage of additional information from more contextual frames.

In summary, our contributions are four-fold: 1) We propose a novel framework, VideoFlow, that learns to estimate optical flows of videos instead of image pairs. 2) We propose TROF, a simple and effective basic model for three-frame optical flow estimation. 3) We propose a dynamic MOtion Propagation (MOP) module that bridges TROFs for handling multi-frame optical flow estimation. 4) VideoFlow outperforms previous methods by large margins on all benchmarks.

2. Related Work

Optical flow. Optical flow estimation traditionally is modeled as an optimization problem that maximizes the visual similarity between image pairs with regularization terms [17, 2, 3, 53]. FlowNet [10] is the first method that end-to-end learns to regress optical flows with a convolutional network. FlowNet2.0 [27] takes this step further, adopting a stacked architecture with the warping operation, which performs on par with state-of-the-art (SOTA) optimization-based methods. The success of FlowNets motivates researchers to design better network architectures for optical flow learning. A series of works, represented by SpyNet [48], PWC-Net [55, 56], LiteFlowNet [24, 25] and VCN [68], emergencies, employing coarse-to-fine and iterative estimation methodology. Despite the fast progress, these models inherently suffer from missing small fast-motion objects in the coarse stage and can only refine flows in limited steps. To remedy this issue, Teed and Deng [59] propose RAFT [59], which estimates optical flow in a coarse-and-fine (i.e. multi-scale search window in each iteration) and recurrent manner. The accuracy is significantly improved along with the recurrent iterations increasing. The iterative refinement paradigm is adopted in the following works [30, 64, 31, 71, 16]. Recently, transformer-based architectures [21, 51, 22, 38, 65] for optical flow, such as FlowFormer [21], shows great superiority against previous CNNs. FlowFormer++ [51, 22] further unleashes the capacity of transformers by pertaining with the Masked Cost-Volume Autoencoding (MCVA). Optical flow is also extended to more challenging settings, such as low-light [77], foggy [67], and lighting variations [20].

Multi-frame optical flow. In the traditional optimization-based optical flow era, researchers used Kalman filter [11, 7] to estimate optical flows with the temporal dynamics of motion for multi-frame optical flow estimation. "warm-start" [59], which simply warps the flows of the previous image pairs as the initialization for the next image pairs, improves RAFT series [52, 57] by non-trivial margins. PWC-Fusion [49], which fuses information from previous frames with a GRU-RCN at the bottleneck of U-Net, only achieves 0.65% performance gain over PWC-Net due to the rough feature encoding. ContinualFlow [45], Unsupervised Flow [29] and Starflow [14] only warp flow/features in one direction. ProFlow [43], similar to [49], focuses on the three-frame setting, individually predicting bi-directional flows and combining them with fusion model. [26] utilizes LSTM to propagate motion information, which shows in-

ferior performance than gradually increasing receptive field during recurrence in our experiment. SelfFlow [40] takes three-frame estimation as strong pretraining for two-frame estimation. In contrast, we integrate motion features from both forward and backward directions during flow refinement. Point-tracking [9, 15, 1, 60] is also closely related to multi-frame optical flow. In contrast to optical flow measuring dense correspondences between adjacent frames, point-tracking cares about the trajectories of the specified points on the following frames.

3. Method

Optical flow estimation targets at regressing a per-pixel displacement field $\mathbf{f}_{t\to t+1} : \mathbb{I}^{H\times W\times 2} \to \mathbb{R}^{H\times W\times 2}$, which maps each source pixel $\mathbf{x} \in \mathbb{I}^2$ of image $\mathbf{I}_t$ at time t to its corresponding coordinate $\mathbf{x}'_{t+1} = \mathbf{x} + \mathbf{f}_{t\to t+1}(\mathbf{x})$ on the target image $\mathbf{I}_{t+1}$. Existing methods typically focus on the two-frame setting, thereby ignoring the rich temporal cues from more neighboring frames. In view of this limitation, we propose VideoFlow, a novel framework to exploit the valuable information from wider temporal context to achieve more accurate optical flow estimation. VideoFlow mainly consists of two modules: 1) a TRi-frame Optical Flow (TROF) module designed to estimate bi-directional optical flows of three-frame clips, and 2) a MOtion Propagation (MOP) module that extends TROF for multi-frame optical flow estimation. In this section, we will elaborate the TROF and MOP modules.

3.1. TROF for Tri-frame Optical Flow Estimation

TROF learns to estimate bi-directional optical flows in a three-frame setting. As shown in Fig. 2, given a frame $\mathbf{I}_t$, its previous frame $\mathbf{I}_{t-1}$ and next frame $\mathbf{I}_{t+1}$, TROF iteratively estimates a sequence of bidirectional flows $\mathbf{f}^k \in \mathbb{R}^{H\times W\times 2\times 2}$, where $k = 1, 2, ..., N$ indicates the refinement iteration step. $\mathbf{f}^k$ includes a 2D flow to the previous frame $\mathbf{f}^k_{t\to t-1}$ and a 2D flow to the next frame $\mathbf{f}^k_{t\to t+1}$. For brevity, we omit the subscript t when the variables stands for the center frame $\mathbf{I}_t$.

Dual Correlation Volumes from Tri-frame Features. Correlation volumes, which measure pixel-wise visual similarities between image pairs, provide pivotal information for flow estimation in previous methods [59, 21, 51]. TROF also infers flows from the correlation volumes but it concurrently estimates bi-directional flows from the center frame to its previous frame $\mathbf{f}^k_{t\to t-1}$ and next frame $\mathbf{f}^k_{t\to t+1}$ by formulating a dual correlation volume. Specifically, we encode three input images with a feature encoder that outputs a feature map of shape $H \times W \times D$ for each frame. H and W are the feature height and width at $1/8$ resolution of original images, and we set feature dimension $D = 256$. TROF builds dual correlation volumes $\mathbf{Corr}_{t,t-1}, \mathbf{Corr}_{t,t+1} \in \mathbb{R}^{H\times W\times H\times W}$ by computing their pixel-wise dot-product similarities. Given the dual correlation volume, TROF concurrently predicts the bi-directional flows.

Bi-directional Motion Feature Fusion. The core of our TROF lies in fully fusing bi-directional motion information to the center frame. Similar to RAFT [59], TROF iteratively retrieves multi-scale correlation values $\mathbf{Corr}(\mathbf{f})$ centered at current flows to refine the bi-directional optical flow. Specifically, at the k-th iteration, we retrieve correlation values $\mathbf{c}^k_{t\to t-1} = \mathbf{Corr}_{t,t-1}(\mathbf{f}^k_{t\to t-1})$ and $\mathbf{c}^k_{t\to t+1} = \mathbf{Corr}_{t,t+1}(\mathbf{f}^k_{t\to t+1})$ from dual correlation volumes according to currently predicted bi-directional flows $\mathbf{f}^k_{t\to t-1}$ and $\mathbf{f}^k_{t\to t+1}$ respectively. We fuse and encode correlation feature $\mathbf{F}^k_{corr} \in \mathbb{R}^{H\times W\times D_c}$, flow feature $\mathbf{F}^k_{flow} \in \mathbb{R}^{H\times W\times D_f}$ at the center frame as

$$\begin{aligned}\mathbf{F}^k_{corr} &= \mathrm{CorrEncoder}(\mathbf{c}^k_{t\to t-1}, \mathbf{c}^k_{t\to t+1}),\\ \mathbf{F}^k_{flow} &= \mathrm{FlowEncoder}(\mathbf{f}^k_{t\to t-1}, \mathbf{f}^k_{t\to t+1}).\end{aligned} \tag{1}$$

The correlation values $\mathbf{c}^k_{t\to t-1}, \mathbf{c}^k_{t\to t+1}$ from the backward and forward flow fields are passed to a correlation encoder to obtain fused correlation features $\mathbf{F}^k_{corr}$. Similarly, we also fuse current predicted bi-directional flows $\mathbf{f}^k_{t\to t-1}$ and $\mathbf{f}^k_{t\to t+1}$ with the flow encoder to obtain the flow features $\mathbf{F}^k_{flow}$. The correlation feature $\mathbf{F}^k_{corr}$ and flow feature $\mathbf{F}^k_{flow}$ respectively encode the dual correlation values and bi-directional displacement information. Then we further encode $\mathbf{F}^k_{corr}$ and $\mathbf{F}^k_{flow}$ to obtain the bi-directional motion feature $\mathbf{F}^k_m \in \mathbb{R}^{H\times W\times D_m}$:

$$\mathbf{F}^k_m = \mathrm{MotionEncoder}_{\mathrm{TROF}}(\mathbf{F}^k_{corr}, \mathbf{F}^k_{flow}). \tag{2}$$

The information from the dual correlation volumes is well aligned at the center frame because the bi-directional flow fields originate from the same source pixels to the two neighboring frames. $\mathbf{F}^k_m$ contains rich motion and correlation information and is used to predict residual bi-directional flows as introduced below.

Bi-directional Recurrent Flow Updating. Following RAFT [59], we maintain a hidden state $\mathbf{h}^k$ to cache features for recurrent flow refinement. We also use the visual features $\mathbf{g}$ of the center frame $\mathbf{I}_t$ as the context feature and initialize the hidden state with the context feature as $\mathbf{h}^0 = \mathbf{g}$. By passing the motion feature $\mathbf{F}^k_m$, along with the context feature $\mathbf{g}$ and hidden state feature $\mathbf{h}^k$ from the previous iteration into a recurrent updating block to update the hidden state, we decode bidirectional residual flows $\mathbf{\Delta f}^k \in \mathbb{R}^{H\times W\times 2\times 2}$ from the hidden state:

$$\begin{aligned}&\mathbf{h}^{k+1} = \mathrm{Updater}(\mathbf{F}^k_m, \mathbf{g}, \mathbf{h}^k),\\ &\mathbf{\Delta f}^k_{t\to t-1,t\to t+1} = \mathrm{FlowHead}(\mathbf{h}^{k+1}),\\ &\mathbf{f}^{k+1}_{t\to t-1,t\to t+1} = \mathbf{f}^k_{t\to t-1,t\to t+1} + \Delta\mathbf{f}^k_{t\to t-1,t\to t+1}.\end{aligned} \tag{3}$$

The predicted flow residuals $\mathbf{\Delta f}^k$ is added to the currently predicted flow to iteratively refine the flow prediction.

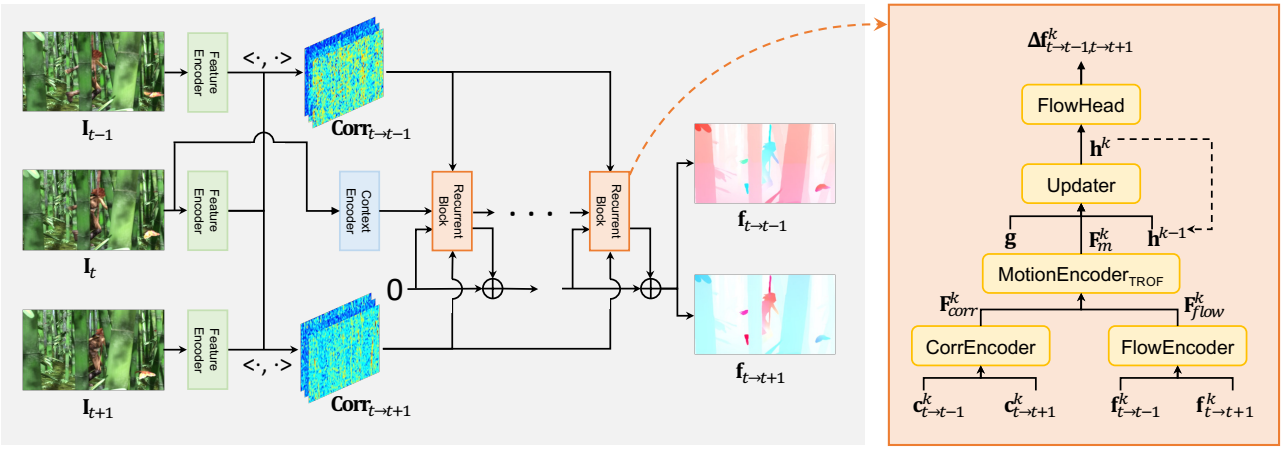


Figure 2. **Overview of VideoFlow in the three-frame setting.** Given a triplet of frames as input, VideoFlow jointly estimates bi-directional optical flows from the center frame to the adjacent previous and next frames. After building dual cost volumes, it recurrently fuses bi-directional flow features and correlation features to update flow predictions. The orange block on the right illustrates the recurrent flow refinement block.

For all the encoders, we use the SKBlock following SKFlow [57], which consists of large-kernel depth-wise convolution layers (details in the supplemental material).

3.2. Bridging TROFs with Motion Propagation

TROF is a simple and effective module for three-frame bi-directional flows. We further extend TROFs for tackling optical flow estimation of more than three frames with a MOtion Propagation (MOP) module. Give multiple frames in a video as input, they are split into overlapped frame triplets and a TROF is in charge of handling a triplet. In Fig. 3, we illustrate processing five input frames and use three TROFs on the triplets of $\{\mathbf{I}_{t-2}, \mathbf{I}_{t-1}, \mathbf{I}_t\}$, $\{\mathbf{I}_{t-1}, \mathbf{I}_t, \mathbf{I}_{t+1}\}$,$\{\mathbf{I}_t, \mathbf{I}_{t+1}, \mathbf{I}_{t+2}\}$. However, such a design does not take advantage of the extra temporal context information as the temporal fusion is limited within each single TROF. To efficiently propagate information across neighboring TROFs, MOP improves the motion encoder in TROF (Eq. (2)) by recurrently fusing motion features warped from adjacent TROFs. Therefore, the motion features in each TROF are gradually propagated to the entire sequence via recurrent iterations.

Specifically, we additionally maintain a motion state feature $\mathbf{M}_t^k \in \mathbb{R}^{H\times W\times D_m}$ for the TROF unit on the triplet $\{\mathbf{I}_{t-1}, \mathbf{I}_t, \mathbf{I}_{t+1}\}$ in the motion encoder of MOP. $\mathbf{M}_t^0$ is randomly initialized and learned via training. It serves to communicate motion features with adjacent TROFs centered at $t-1$ and $t+1$ and is updated to integrate temporal cues. In each iteration k, we retrieve temporal motion information $\mathbf{m}_{fwd}^k$, $\mathbf{m}_{bwd}^k$ from adjacent TROFs by warping their motion state features $\mathbf{M}_{t-1}^k$ and $\mathbf{M}_{t+1}^k$ according to currently predicted flows:

$$\begin{aligned}\mathbf{m}_{fwd}^k &= \mathrm{Warp}(\mathbf{M}_{\mathrm{t+1}}^{\mathrm{k}}; \mathbf{f}_{\mathrm{t}\to\mathrm{t+1}}^{\mathrm{k}}),\\ \mathbf{m}_{bwd}^k &= \mathrm{Warp}(\mathbf{M}_{\mathrm{t-1}}^{\mathrm{k}}; \mathbf{f}_{\mathrm{t}\to\mathrm{t-1}}^{\mathrm{k}}),\\ \mathbf{F}_{mop}^k &= \mathrm{Concat}(\mathbf{M}_t^k, \mathbf{m}_{fwd}^k, \mathbf{m}_{bwd}^k).\end{aligned} \tag{4}$$

After concatenating them with the motion state feature $\mathbf{M}_t^k$ of the current frame, we obtain the motion propagation feature $\mathbf{F}_{mop}^k$, which augments $\mathbf{F}_{corr}^k$ and $\mathbf{F}_{flow}^k$ by summarizing motion features of consecutive three TROFs in each iteration. The motion feature state $\mathbf{M}_t^k$ is also updated in each iteration.

$$\mathbf{F}_m^k, \mathbf{M}_t^{k+1} = \mathrm{MotionEncoder}_{\mathrm{MOP}}(\mathbf{F}_{corr}^k, \mathbf{F}_{flow}^k, \mathbf{F}_{mop}^k), \tag{5}$$

where $\mathbf{F}_m^k$ is passed to the updater to predict residual flows $\mathbf{\Delta f}^k$ as in Eq (3), and $\mathbf{M}_t^{k+1}$ will be used in the next iteration $k+1$ of Eq (4). As MOP absorbs the hidden motion features from adjacent TROFs at each iteration, the temporal receptive field of the hidden motion features $\mathbf{M}_t^k$ gradually increases as iterations proceed. We thus can process and integrate more TROFs jointly to expand the effective timespan with MOP. In practice, our network takes 5 frames as input and is trained to output 3 center frames' bi-directional flows during training. During inference, the network is used to predict T frames' flow at a time for input video clips of length $T+2$.

3.3. Training Loss

VideoFlow requires consecutive multiple frames with corresponding ground truth bi-directional optical flows $\mathbf{f}_{gt,t\to t-1}, \mathbf{f}_{gt,t\to t+1}$ for training. We train VideoFlow to predict T bidirectional flow with $\ell 1$ loss for all predicted

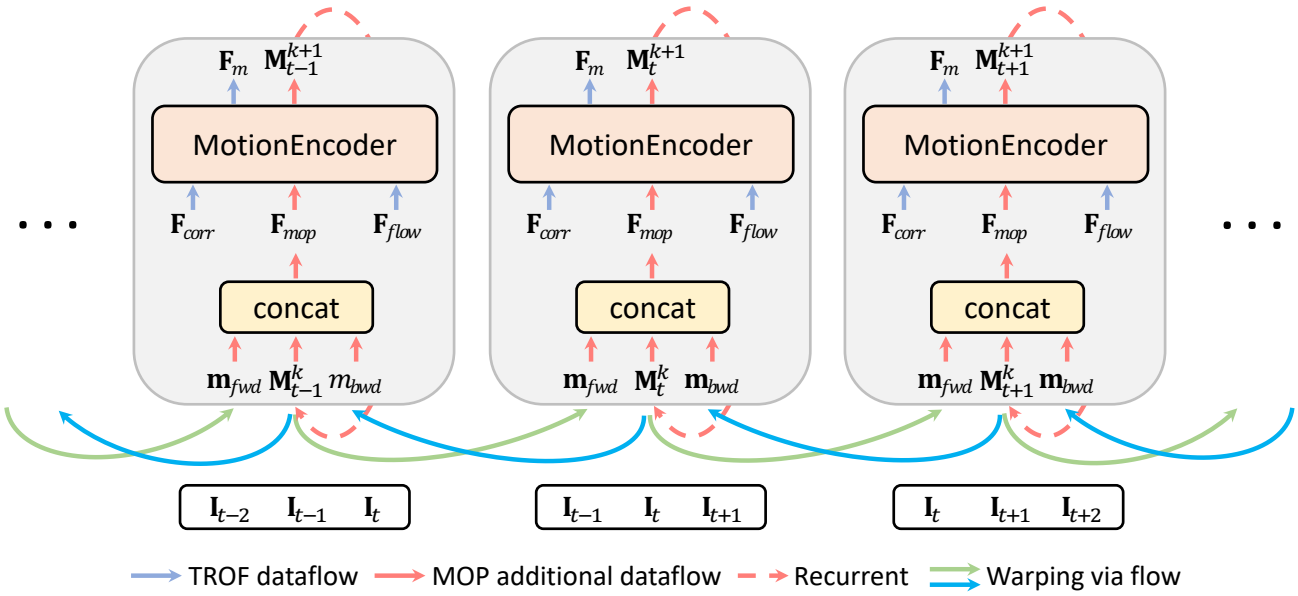


Figure 3. **MOP motion encoder.** We bridge TROF units with the motion propagation module for tackling optical flow estimation of more than three frames. Specifically, we additionally maintains a motion state feature $\mathbf{M}_t^k$ in the motion encoder of each TROF unit. In each iteration, the motion state features of adjacent TROF units are warped to the center frame as auxiliary information. $\mathbf{M}_t^k$ is updated by the motion encoder and exploits wider temporal contexts as the recurrent process iterates.

flows defined as:

$$\mathcal{L} = \sum_{t=1}^{T}\sum_{k=0}^{N}\gamma^{N-k}||\mathbf{f}_{gt,t\to t-1} - \mathbf{f}^k_{t\to t-1}||_1 + \gamma^{N-k}||\mathbf{f}_{gt,t\to t+1} - \mathbf{f}^k_{t\to t+1}||_1, \tag{6}$$

where N (set as 12 in our experiments) is the number of recurrent steps and γ is set as 0.85 to add higher weights on later predictions following RAFT [59].

4. Experiments

We evaluate our VideoFlow on the Sintel [4] and the KITTI-2015 [13] benchmarks. VideoFlow outperforms all previous methods on both benchmarks and reduces the average end-point-error (AEPE) to a subpixel level on the clean pass of Sintel for the first time.

Experimental setup. We adopt the average end-point-error (AEPE) and Fl-All(%) as the evaluation metrics. The AEPE denotes mean flow error over all valid pixels. The Fl-All computes the percentage of pixels with flow error larger than 3 pixels and over 5% of ground truth. The Sintel dataset consists of two passes rendered from the same model. The clean pass is easier with smooth shading and specular reflections, while the final pass enables more challenging rendering settings including atmospheric effects, motion blur and camera depth-of-field blur.

Implementation Details. Following the FlowFormer series [21, 51], we use the first two stages of ImageNet-pretrained Twins-SVT [8] as the image encoder and context encoder and fine-tune the parameters. We follow SKFlow [57] to replace ConvGRU with SKBlocks as an iterative flow refinement module. Since FlyingChairs [10] only contains two-frame training pairs, we skip it and directly pretrain VideoFlow on the FlyingThings [44] dataset. For our three-frame model, we pretrain it on the FlyingThings dataset for 300k iterations (denoted as 'T'). Then we finetune it on the data combined from FlyingThings, Sintel, KITTI-2015 and HD1K [33] (denoted as 'T+S+K+H') for 120k iterations. This model is submitted to the Sintel online test benckmark for evaluation. We further finetune the model on the KITTI-2015 dataset for 50k iterations. For our five-frame model, we follow the same schedule with fewer training iterations: 125k iterations on 'T', 40k iterations on 'T+S+K+H' and 25k iterations on 'K'. We choose AdamW optimizer and one-cycle learning rate scheduler. Batch size is set as 8 for all stages. The highest learning rate is set as 2.5×10^{-4} for FlyingThings and 1.25×10^{-4} on other training datasets. Please refer to supplementary for more details.

4.1. Quantitative Experiment

As shown in Table 1, we evaluate VideoFlow on the Sintel [4] and KITTI-2015 [13] benchmarks. Specifically, we compare the generalization performance of models on the training set of Sintel and KITTI-2015 (denoted as 'C+T' for other two-frame models and 'T' for our Vide-

Training Data	Method	Sintel (train)		KITTI-15 (train)		Sintel (test)		KITTI-15 (test)
		Clean	Final	Fl-epe	Fl-all	Clean	Final	Fl-all
A	Perceiver IO [28]	1.81	2.42	4.98	-	-	-	-
	PWC-Net [55]	2.17	2.91	5.76	-	-	-	-
	RAFT [59]	1.95	2.57	4.23	-	-	-	-
C+T	HD3 [70]	3.84	8.77	13.17	24.0	-	-	-
	LiteFlowNet [24]	2.48	4.04	10.39	28.5	-	-	-
	PWC-Net [55]	2.55	3.93	10.35	33.7	-	-	-
	LiteFlowNet2 [25]	2.24	3.78	8.97	25.9	-	-	-
	S-Flow [71]	1.30	2.59	4.60	15.9			
	RAFT [59]	1.43	2.71	5.04	17.4	-	-	-
	FM-RAFT [31]	1.29	2.95	6.80	19.3	-	-	-
	GMA [30]	1.30	2.74	4.69	17.1	-	-	-
	GMFlow [65]	1.08	2.48	-	-	-	-	-
	GMFlowNet [75]	1.14	2.71	4.24	15.4	-	-	-
	CRAFT [52]	1.27	2.79	4.88	17.5	-	-	-
	SKFlow [57]	1.22	2.46	4.47	15.5	-	-	-
	FlowFormer [21]	0.94	2.33	4.09	14.72	-	-	-
	FlowFormer++ † [51]	**0.90**	2.30	3.93	**14.13**	-	-	-
T	Ours (3 frames)	1.03	**2.19**	3.96	15.33	-	-	-
	Ours (5 frames)	1.18	2.56	**3.89**	14.20	-	-	-
C+T+S+K+H	LiteFlowNet2 [25]	(1.30)	(1.62)	(1.47)	(4.8)	3.48	4.69	7.74
	PWC-Net+ [56]	(1.71)	(2.34)	(1.50)	(5.3)	3.45	4.60	7.72
	VCN [68]	(1.66)	(2.24)	(1.16)	(4.1)	2.81	4.40	6.30
	MaskFlowNet [74]	-	-	-	-	2.52	4.17	6.10
	S-Flow [71]	(0.69)	(1.10)	(0.69)	(1.60)	1.50	2.67	4.64
	RAFT [59]	(0.76)	(1.22)	(0.63)	(1.5)	1.94	3.18	5.10
	FM-RAFT [31]	(0.79)	(1.70)	(0.75)	(2.1)	1.72	3.60	6.17
	GMA [30]	-	-	-	-	1.40	2.88	5.15
	GMFlow [65]	-	-	-	-	1.74	2.90	9.32
	GMFlowNet [75]	(0.59)	(0.91)	(0.64)	(1.51)	1.39	2.65	4.79
	CRAFT [52]	(0.60)	(1.06)	(0.57)	(1.20)	1.45	2.42	4.79
	FlowFormer [21]	(0.48)	(0.74)	(0.53)	(1.11)	1.16	2.09	4.68
	FlowFormer++ † [51]	(0.40)	(0.60)	(0.57)	(1.16)	1.07	1.94	4.52
	PWC-Fusion* [56]	-	-	-	-	3.43	4.57	7.17
	RAFT* [59]	(0.77)	(1.27)	-	-	1.61	2.86	5.10
	GMA* [30]	(0.62)	(1.06)	(0.57)	(1.2)	1.39	2.47	5.15
	SKFlow* [57]	(0.52)	(0.78)	(0.51)	(0.94)	1.28	2.23	4.84
T+S+K+H	Ours (3 frames)	(0.37)	(0.54)	(0.52)	(0.85)	1.00	1.71	4.44
	Ours (5 frames)	(0.46)	(0.66)	(0.56)	(1.05)	**0.99**	**1.65**	**3.65**

Table 1. **Experiments on Sintel [4] and KITTI [13] datasets.** 'A' denotes the autoflow dataset. 'C + T' denotes training only on the FlyingChairs and FlyingThings datasets. '+ S + K + H' denotes finetuning on the combination of Sintel, KITTI, and HD1K training sets. * denotes methods that use three frames for prediction. PWC-Fusion [49] fuses two independently predicted flows. Other methods use the warm-start strategy [59], which warps the estimation of the preceding frame pair to initialize the current estimation. † denotes that FlowFormer++ [51] has an additional pre-training stage. We use **bold** and _ to highlight the methods that rank 1st and 2nd.

oFlow). We then compare the dataset-specific fitting ability of optical flow models after dataset-specific finetuning (denoted as 'C+T+S+K+H' for other two-frame models and 'T+S+K+H' for our VideoFlow). 'A' refers to pre-training models on another synthetic dataset Autoflow [54], while its training code is not publicly available.

Generalization Performance. In Table 1, the 'T'/'C+T' settings reflect the cross-dataset generalization ability of models. Our VideoFlow achieves comparable performance with FlowFormer series [21, 51] and outperforms other models. It is worth noting that FlowFormer and FlowFormer++ have 35% more parameters than VideoFlow (18.2M vs 13.5M). FlowFormer++ is additionally pre-trained with masked autoencoding strategy. Specifically, 3-frame VideoFlow ranks first on the challenging final pass of Sintel training set. The 5-frame VideoFlow achieves best performance on KITTI-2015 Fl-epe metric and is only second to FlowFormer++ on the Fl-all metric.

Dataset-specific Performance. After training our VideoFlow in the 'T+S+K+H' setting, we submit it to the online Sintel benchmark. As shown in Table 1, our 3-frame model already outperforms all published methods, achieving 1.00 and 1.71 AEPE on the clean and final passes, respectively. 5-frame VideoFlow further increases the accu-

racy. Specifically, it achieves 0.99 and 1.65 AEPE on the clean and final passes, a 7.6% and 15.1% error reduction from FlowFormer++, with much fewer parameters. Then we further finetune VideoFlow on the KITTI-2015 training set and submit it to the online benchmark. The 3-frame VideoFlow achieves an Fl-all error of 4.44%, surpassing all previous published methods. Our five-frame VideoFlow further obtains 3.65%, a 19.2% error reduction from the previous best-published method FlowFormer++.

Multi-frame Methods Comparison. Besides VideoFlow, there are four other methods that can be regarded as estimating optical flow from three-frame information, i.e. PWC-Fusion [56], RAFT* [59], GMA* [30], and SKFlow* [57]. PWC-Fusion adopts a temporal GRU at the bottleneck of PWC-Net [55] to fuse motion information from previous frames. In the three-frame structure, the other three methods adopt the warm-start technique, which warps the flows from the former pair to the later pair as the initialization. Such designs do not take information from future frames and only fuse information once at a coarse level, thereby bringing little benefits. In contrast, VideoFlow deeply integrates information from both directions during iterative flow refinement. 3-frame VideoFlow outperforms PWC-Fusion by 70.3% and SKFlow* by 20.3% on clean pass. Moreover, our 5-frame version further reduces the error (10.3% on Sintel final pass and 21.6% on KITTI), which beyond the capability of the warm-start technique because it can not draw benefits from longer sequences.

Performance Analysis on Sintel Test. To investigate the superior performance of VideoFlow, we provide additional metrics in Table 2, where 'unmatched' refers to EPE over occluded or out-of-boundary pixels and s_{0-10}, s_{10-40}, s_{40+} denote EPE over pixels with ground truth flow motion magnitude falling to $0-10$, $10-40$ and more than 40 pixels, respectively. We select SKFlow*, FlowFormer, and FlowFormer++, which are the most competitive methods, for comparison. Compared with "matched" pixels, "unmatched" pixels are hard cases because they are invisible in the target image. Similarly, pixels whose flow motion magnitudes are larger are more challenging especially on the final pass because faster movement leads to more severe motion blur. On the clean pass, VideoFlow does not show performance gain over 'Matched' pixels compared with FlowFormer and FlowFormer++ because these cases are rather easy. However, VideoFlow presents dominating superiority over the other metrics that measure flows of challenging pixels: 'unmatched' pixels, large-motion pixels, and even "matched" pixels on the final pass. The 5-frame VideoFlow reduces 18.5% AEPE of 'Matched' pixels on the final pass from FlowFormer++. The clear performance improvements obtained by our VideoFlow on unmatched pixels indicate that VideoFlow effectively reduces the ambiguity of out-of-view pixels with the wider integrating temporal cues. Besides, our VideoFlow brings significant gains over pixels with large movements, especially on the more challenging final pass, which denotes that VideoFlow alleviates distractions from motion blurs by context information.

4.2. Qualitative Experiment

We visualize flow predictions of FlowFormer++ [51] and our VideoFlow on Sintel and KITTI test sets in Fig. 4 to show the superior performance of VideoFlow over FlowFormer++. By utilizing temporal cues, the blue rectangles highlight that our VideoFlow preserves more details and handles ambiguity better: in the first row, VideoFlow shows the gaps between barriers but FlowFormer++ only produces blurry flows; in the second row, FlowFormer++ produces accident artifacts at the right of the car while VideoFlow erases them because wider temporal cues significantly improve the flow robustness. Moreover, FlowFormer++ fails to distinguish the shadows from the ground for the bicycle and the car while our VideoFlow predicts better flows.

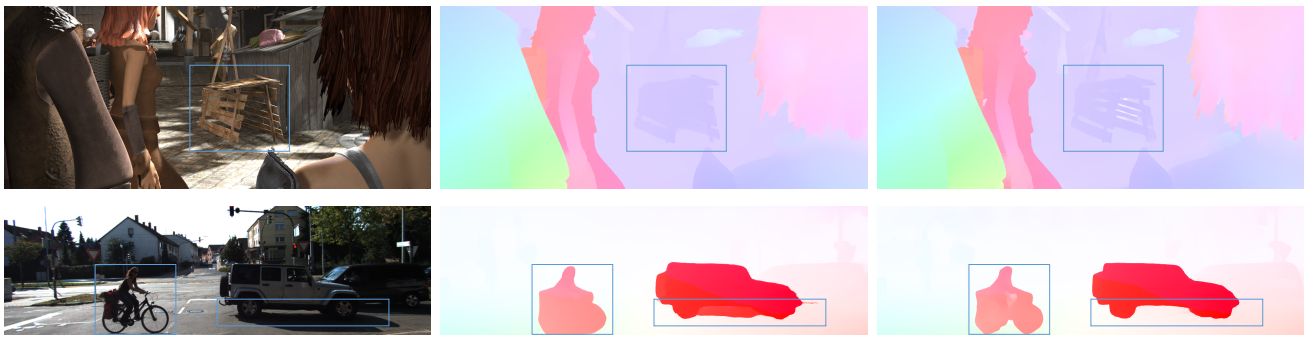
4.3. Ablation Study

We conduct a series of ablation studies to show the effectiveness of our designs.

Three-frame model design. We verify the two critical designs of our three-frame model TROF as in Table 3. We re-implement [49] based on our network as baseline (the first row of Table 3), which warps the correlation features and flow predictions of the first frame pair to align with current frame pair. We first convert it to predicting bi-directional optical flows originating from the center frame (the second row of Table 3). Then we remove the independent fusion layer and fuse bi-directional motion features through the recurrent process (the third row of Table 3). Results show that the bi-directional estimation brings clear performance gains over the uni-directional baseline on most metrics. Moreover, the recurrent fusion further boosts the performance.

Motion propagation module design. We propose motion propagation module to bridge individual TROFs. One naive strategy is to only pass correlation features and bi-directional flow features to adjacent units (the first row in 4), which has limited temporal receptive field. We propose to additionally maintain a motion state feature $\mathbf{M}_t^k$ (the third row in Table 4). In this way, the temporal receptive field grows with the recurrent updating process. We also tried adding a temporal GRU module to pass motion state feature through all TROF units in each iteration (the second row of Table 4). But this strategy brings performance drop on the FlyingThings and KITTI-2015 datasets.

Bi-directional flows comparison. Our VideoFlow jointly estimates bi-directional flows. In Table 5, we compare the accuracy of bi-directional flows for both three-frame and five-frame VideoFlow models. Specifically, for the backward flow test, we pass the input image sequence in re-



Input FlowFormer++ VideoFlow (Ours)

Figure 4. **Qualitative comparison on Sintel and KITTI test sets.** VideoFlow preserves clearer details (row 1st). By better utilizing temporal cues, VideoFlow successfully distinguishes the ground from shadows and avoids accident artifacts (row 2nd).

Method	Sintel Test (clean)						Sintel Test (final)					
	All	Matched	Unmatched	s_{0-10}	s_{10-40}	s_{40+}	All	Matched	Unmatched	s_{0-10}	s_{10-40}	s_{40+}
SKFlow* [57]	1.28	0.57	7.25	0.28	0.95	7.17	2.26	1.14	11.42	0.58	1.68	12.02
FlowFormer [21]	1.16	0.42	7.16	0.26	0.82	6.44	2.09	0.96	11.30	0.46	1.47	11.66
FlowFormer++ [51]	1.07	0.39	6.64	0.25	0.80	5.81	1.94	0.88	10.63	0.44	1.40	10.71
VideoFlow (3 frames)	1.02	**0.38**	6.19	**0.22**	**0.69**	5.75	1.84	0.86	9.81	0.42	1.29	10.19
VideoFlow (5 frames)	**0.99**	0.40	**5.83**	0.23	**0.69**	**5.48**	**1.65**	**0.79**	**8.66**	**0.40**	**1.24**	**8.80**

Table 2. **Sintel test results analysis.** 'Unmatched' refers to occluded or out-of-boundary pixels and s_{0-10}, s_{10-40}, s_{40+} denote pixels with ground truth flow motion magnitude falling in $0-10$, $10-40$, and more than 40 pixels, respectively. VideoFlow obtains clear improvements in challenging cases, including 'Unmatched' pixels and pixels with large motions.

Bi-directional	Recurrent Fusion	Things (val)		Sintel (train)		KITTI-15 (train)	
		Clean	Final	Clean	Final	Fl-epe	Fl-all
✗	✗	2.70	2.53	1.55	2.62	4.82	17.48
✓	✗	2.61	2.52	1.49	2.58	4.60	18.05
✓	✓	**2.54**	**2.49**	**1.48**	**2.49**	**4.51**	**16.52**

Table 3. **Three-frame model design.** Bi-directional estimation can better utilize temporal information as motion features are well aligned in the center frame. Recurrent fusion further benefits motion features integration.

$\mathbf{M}_t^k$?	Propagation Range	Things (val)		Sintel (train)		KITTI-15 (train)	
		Clean	Final	Clean	Final	Fl-epe	Fl-all
✗	Adjacent Units	1.61	1.43	1.15	2.53	4.02	14.68
✓	All Units	1.56	1.40	**1.07**	**2.47**	4.04	14.55
✓	Adjacent Units	**1.48**	**1.36**	1.16	2.56	**3.89**	**14.2**

Table 4. **Motion propagation design.** Our motion propagation module maintains a motion state feature $\mathbf{M}_t^k$ which absorbs adjacent units' motion state features and integrates wider temporal cues over iterations (the third row).

verse order and compare the estimated backward flows with ground truth. As shown in Table 5, the bi-directional predictions achieve similar accuracy because of the symmetry of our model. Such high-quality bi-directional flows naturally fits downstream video processing algorithms.

Frame Number	Flow Direction	Things (val)		Sintel (train)		KITTI-15 (train)	
		Clean	Final	Clean	Final	Fl-epe	Fl-all
3	Forward	1.62	1.42	1.03	2.19	3.96	15.33
3	Backward	1.63	1.42	0.98	2.23	4.05	14.74
5	Forward	1.48	1.36	1.16	2.56	3.89	14.2
5	Backward	1.49	1.37	1.20	2.66	3.79	14.44

Table 5. **Forward and backward flows comparison.** Our VideoFlow predicts multi-frame bi-directional flows, naturally fitting downstream video processing algorithms.

5. Conclusion

We propose VideoFlow, which takes TRi-frame Optical Flow (TROF) module as building block in a three-frame manner. We further extend it to handle more frames by bridging TROF units with motion propagation module. Our method outperforms previous methods with large margins on all benchmarks.

6. Acknowledgements

This project is funded in part by National Key R&D Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)'s InnoHK, by General Research Fund of Hong Kong RGC Project 14204021. Hongsheng Li is a PI of CPII under the InnoHK.

References

[1] Weikang Bian, Zhaoyang Huang, Xiaoyu Shi, Yitong Dong, Yijin Li, and Hongsheng Li. Context-tap: Tracking any point demands spatial context features. *arXiv preprint arXiv:2306.02000*, 2023. 3

[2] Michael J Black and Padmanabhan Anandan. A framework for the robust estimation of optical flow. In *1993 (4th) International Conference on Computer Vision*, pages 231–236. IEEE, 1993. 2

[3] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International journal of computer vision*, 61(3):211–231, 2005. 2

[4] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012. 5, 6

[5] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4947–4956, 2021. 1

[6] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5972–5981, 2022. 1

[7] Toshio M Chin, William Clement Karl, and Alan S Willsky. Probabilistic and sequential computation of optical flow using temporal coherence. *IEEE Transactions on Image Processing*, 3(6):773–788, 1994. 2

[8] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting spatial attention design in vision transformers. *arXiv preprint arXiv:2104.13840*, 2021. 5

[9] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens Continente, Kucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. In *NeurIPS Datasets Track*, 2022. 3

[10] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 2, 5

[11] Michael Elad and Arie Feuer. Recursive optical flow estimation—adaptive filtering approach. *Journal of Visual Communication and image representation*, 9(2):119–138, 1998. 2

[12] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *European Conference on Computer Vision*, pages 713–729. Springer, 2020. 1

[13] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 5, 6

[14] Pierre Godet, Alexandre Boulch, Aurélien Plyer, and Guy Le Besnerais. Starflow: A spatiotemporal recurrent cell for lightweight multi-frame optical flow estimation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2462–2469. IEEE, 2021. 2

[15] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *ECCV*, 2022. 3

[16] Markus Hofinger, Samuel Rota Bulò, Lorenzo Porzi, Arno Knapitsch, Thomas Pock, and Peter Kontschieder. Improving optical flow on a pyramid level. In *European Conference on Computer Vision*, pages 770–786. Springer, 2020. 2

[17] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. 2

[18] Xiaotao Hu, Zhewei Huang, Ailin Huang, Jun Xu, and Shuchang Zhou. A dynamic multi-scale voxel flow network for video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6121–6131, 2023. 1

[19] Zhihao Hu and Dong Xu. Videocontrolnet: A motion-guided video-to-video translation framework by using diffusion model with controlnet. *arXiv preprint arXiv:2307.14073*, 2023. 1

[20] Zhaoyang Huang, Xiaokun Pan, Weihong Pan, Weikang Bian, Yan Xu, Ka Chun Cheung, Guofeng Zhang, and Hongsheng Li. Neuralmarker: A framework for learning general marker correspondence. *ACM Transactions on Graphics (TOG)*, 41(6):1–10, 2022. 2

[21] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. *arXiv preprint arXiv:2203.16194*, 2022. 1, 2, 3, 5, 6, 8

[22] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Yijin Li, Hongwei Qin, Jifeng Dai, Xiaogang Wang, and Hongsheng Li. Flowformer: A transformer architecture and its masked cost volume autoencoding for optical flow. *arXiv preprint arXiv:2306.05442*, 2023. 2

[23] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Rife: Real-time intermediate flow estimation for video frame interpolation. *arXiv preprint arXiv:2011.06294*, 2020. 1

[24] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018. 2, 6

[25] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. A lightweight optical flow cnn—revisiting data fidelity and regularization. *IEEE transactions on pattern analysis and machine intelligence*, 43(8):2555–2569, 2020. 2, 6

[26] Junhwa Hur and Stefan Roth. Self-supervised multi-frame monocular scene flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2684–2694, 2021. 2

[27] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 1, 2

[28] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021. 6

[29] Joel Janai, Fatma G"uney, Anurag Ranjan, Michael J. Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *European Conference on Computer Vision (ECCV)*, volume Lecture Notes in Computer Science, vol 11220, pages 713–731. Springer, Cham, Sept. 2018. 2

[30] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. *arXiv preprint arXiv:2104.02409*, 2021. 2, 6, 7

[31] Shihao Jiang, Yao Lu, Hongdong Li, and Richard Hartley. Learning optical flow from a few matches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16592–16600, 2021. 2, 6

[32] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5792–5801, 2019. 1

[33] Daniel Kondermann, Rahul Nair, Katrin Honauer, Karsten Krispin, Jonas Andrulis, Alexander Brock, Burkhard Gussefeld, Mohsen Rahimimoghaddam, Sabine Hofmann, Claus Brenner, et al. The hci benchmark suite: Stereo and flow ground truth with uncertainties for urban autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 19–28, 2016. 5

[34] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017. 1

[35] Dasong Li, Xiaoyu Shi, Yi Zhang, Ka Chun Cheung, Simon See, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. A simple baseline for video restoration with grouped spatial-temporal shift. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9822–9832, June 2023. 1

[36] Dasong Li, Yi Zhang, Ka Chun Cheung, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. Learning degradation representations for image deblurring. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 736–753, Cham, 2022. Springer Nature Switzerland. 1

[37] Dasong Li, Yi Zhang, Ka Lung Law, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. Efficient burst raw denoising with variance stabilization and multi-frequency denoising network, 2022. 1

[38] Yijin Li, Zhaoyang Huang, Shuo Chen, Xiaoyu Shi, Hongsheng Li, Hujun Bao, Zhaopeng Cui, and Guofeng Zhang. Blinkflow: A dataset to push the limits of event-based optical flow estimation. *arXiv preprint arXiv:2303.07716*, 2023. 2

[39] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 1

[40] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4571–4580, 2019. 3

[41] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Decoupled spatial-temporal transformer for video inpainting. *arXiv preprint arXiv:2104.06637*, 2021. 1

[42] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14040–14049, 2021. 1

[43] Daniel Maurer and Andrés Bruhn. Proflow: Learning to predict optical flow. *arXiv preprint arXiv:1806.00800*, 2018. 2

[44] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 5

[45] Michal Neoral, Jan Šochman, and Jiří Matas. Continual occlusion and optical flow estimation. In *Asian Conference on Computer Vision*, pages 159–174. Springer, 2018. 2

[46] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5437–5446, 2020. 1

[47] AJ Piergiovanni and Michael S Ryoo. Representation flow for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9945–9953, 2019. 1

[48] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 2

[49] Zhile Ren, Orazio Gallo, Deqing Sun, Ming-Hsuan Yang, Erik B Sudderth, and Jan Kautz. A fusion approach for multi-frame optical flow estimation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2077–2086. IEEE, 2019. 2, 6, 7

[50] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, 2018. 1

[51] Xiaoyu Shi, Zhaoyang Huang, Dasong Li, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, Jifeng Dai, and

Hongsheng Li. Flowformer++: Masked cost volume autoencoding for pretraining optical flow estimation. *arXiv preprint arXiv:2303.01237*, 2023. 2, 3, 5, 6, 7, 8

[52] Xiuchao Sui, Shaohua Li, Xue Geng, Yan Wu, Xinxing Xu, Yong Liu, Rick Goh, and Hongyuan Zhu. Craft: Cross-attentional flow transformer for robust optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17602–17611, 2022. 2, 6

[53] Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014. 2

[54] Deqing Sun, Daniel Vlasic, Charles Herrmann, Varun Jampani, Michael Krainin, Huiwen Chang, Ramin Zabih, William T Freeman, and Ce Liu. Autoflow: Learning a better training set for optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10093–10102, 2021. 6

[55] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 2, 6, 7

[56] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1408–1423, 2019. 2, 6, 7

[57] Shangkun Sun, Yuanqi Chen, Yu Zhu, Guodong Guo, and Ge Li. Skflow: Learning optical flow with super kernels. *arXiv preprint arXiv:2205.14623*, 2022. 2, 4, 5, 6, 7, 8

[58] Shuyang Sun, Zhanghui Kuang, Lu Sheng, Wanli Ouyang, and Wei Zhang. Optical flow guided feature: A fast and robust motion representation for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1390–1399, 2018. 1

[59] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 1, 2, 3, 5, 6, 7

[60] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. *arXiv preprint arXiv:2306.05422*, 2023. 3

[61] Shiyao Wang, Yucong Zhou, Junjie Yan, and Zhidong Deng. Fully motion-aware network for video object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1

[62] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. 1

[63] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*, 2023. 1

[64] Haofei Xu, Jiaolong Yang, Jianfei Cai, Juyong Zhang, and Xin Tong. High-resolution optical flow from 1d attention and correlation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10498–10507, 2021. 2

[65] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022. 2, 6

[66] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2019. 1

[67] Wending Yan, Aashish Sharma, and Robby T Tan. Optical flow in dense foggy scenes using semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13259–13268, 2020. 2

[68] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. *Advances in neural information processing systems*, 32:794–805, 2019. 2, 6

[69] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. *arXiv preprint arXiv:2306.07954*, 2023. 1

[70] Zhichao Yin, Trevor Darrell, and Fisher Yu. Hierarchical discrete distribution decomposition for match density estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6044–6053, 2019. 6

[71] Feihu Zhang, Oliver J Woodford, Victor Adrian Prisacariu, and Philip HS Torr. Separable flow: Learning motion cost volumes for optical flow estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10807–10817, 2021. 2, 6

[72] Yi Zhang, Dasong Li, Xiaoyu Shi, Dailan He, Kangning Song, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. Kbnet: Kernel basis network for image restoration, 2023. 1

[73] Yi Zhang, Xiaoyu Shi, Dasong Li, Xiaogang Wang, Jian Wang, and Hongsheng Li. A unified conditional framework for diffusion-based image restoration. *arXiv preprint arXiv:2305.20049*, 2023. 1

[74] Shengyu Zhao, Yilun Sheng, Yue Dong, Eric I Chang, Yan Xu, et al. Maskflownet: Asymmetric feature matching with learnable occlusion mask. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6278–6287, 2020. 6

[75] Shiyu Zhao, Long Zhao, Zhixing Zhang, Enyu Zhou, and Dimitris Metaxas. Global matching with overlapping attention for optical flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17592–17601, 2022. 6

[76] Yuxuan Zhao, Ka Lok Man, Jeremy Smith, Kamran Siddique, and Sheng-Uei Guan. Improved two-stream model for human action recognition. *EURASIP Journal on Image and Video Processing*, 2020(1):1–9, 2020. 1

[77] Yinqiang Zheng, Mingfang Zhang, and Feng Lu. Optical flow in the dark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6749–6757, 2020. 2

[78] Xizhou Zhu, Jifeng Dai, Lu Yuan, and Yichen Wei. Towards high performance video object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7210–7218, 2018. 1

[79] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 408–417, 2017. 1