# A Fusion Approach for Multi-Frame Optical Flow Estimation

Zhile Ren     Orazio Gallo   Deqing Sun     Ming-Hsuan Yang     Erik B. Sudderth     Jan Kautz

Georgia Tech       NVIDIA       UC Merced       UC Irvine       NVIDIA

## Abstract

*To date, top-performing optical flow estimation methods only take pairs of consecutive frames into account. While elegant and appealing, the idea of using more than two frames has not yet produced state-of-the-art results. We present a simple, yet effective fusion approach for multi-frame optical flow that benefits from longer-term temporal cues. Our method first warps the optical flow from previous frames to the current, thereby yielding multiple plausible estimates. It then fuses the complementary information carried by these estimates into a new optical flow field. At the time of writing, our method ranks first among published results in the MPI Sintel and KITTI 2015 benchmarks. Our models will be available on* https://github.com/NVlabs/PWC-Net.

## 1. Introduction

Optical flow estimation is an important low-level vision task with numerous applications. Despite recent advances, it is still challenging to account for complicated motion patterns, as shown in Figure 1. At video rates, even such complicated motion patterns are smooth for longer than just two consecutive frames. This suggests that information from frames that are adjacent in time could be used to improve optical flow estimates.

Indeed, numerous methods have been developed that either impose temporal smoothness of the flow [6, 9], or explicitly reason about the trajectories of pixels across multiple frames [40]. Despite the fact that multiple frames carry additional information about scene motion, none of the top three optical flow algorithms on the major benchmark datasets uses more than two frames [11, 33].

This may be due to the fact that motion and its statistics do change over time. If the optical flow field changes dramatically, the visual information contained in longer frame sequences may be less useful, and, in fact, potentially detrimental [44]. The ability to decide when visual information from past frames is useful is paramount to the success of multi-frame optical flow algorithms. Some early methods account for the temporal dynamics of motion using a



Input         Ground Truth

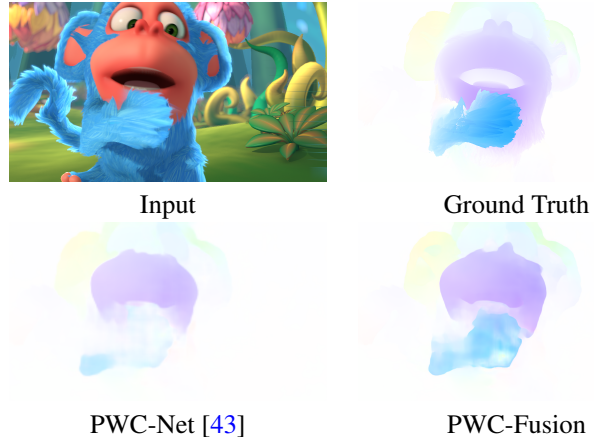PWC-Net [43]       PWC-Fusion

Figure 1: Given a scene with challenging motions, even state-of-the-art two-frame optical flow methods fail to predict the correct motion. Our flow fusion approach offers an effective method to incorporate temporal information and improve the predicted flow.

Kalman filter [17, 14]. More recent approaches attenuate the temporal smoothness strength when a sudden change is detected [39, 44].

We observe that, for some types of motion and in certain regions, past frames may carry more valuable information than recent ones, even if the optical flow changes abruptly—as is the case of occlusion regions and out-of-boundary pixels. Kennedy and Taylor [26] also leverage this observation, and select which *one* of multiple flow estimates from adjacent frames is the best for a given pixel. Rather than simply attenuating the contribution of past frames or making an error-prone binary selection, we propose a method to *fuse* the available information. We warp multiple optical flow estimations from the past to the current frame. Such estimates represent plausible candidates of optical flow for the current frame, which we can fuse with a second neural network module. This approach offers several advantages. First, it allows to fully capitalize on motion information from past frames, in particular when this offers a better estimate than the current frame. Furthermore, to optimally fuse the different optical flow estimates, our method can use information from a large neighborhood. Since our fusion network is agnostic to the algorithm that generates the pair-wise optical

---

flow estimations, any standard method can be used as an input, making our framework flexible and straightforward to upgrade when improved two-frame algorithms become available. Finally, if the underlying optical flow algorithm is differentiable, as is the case for the current state-of-the-art methods, our approach can be trained end-to-end. We show that the proposed algorithm outperforms published state-of-the-art, two-frame optical flow methods by significant margins on the KITTI [21] and Sintel [11] benchmark datasets. To further validate our results, we present alternative baseline approaches incorporating recurrent neural networks with the state-of-the-art deep-learning optical flow estimation methods, and show that significant performance gain can be achieved by using the fusion approach.

## 2. Related Work

Since the seminal work of Horn and Schunck [23], a rich literature of variational approaches has flourished (*e.g.*, [41, 10, 38]). Given the large number of publications in this area, we refer the reader to [41, 2] for a more complete survey focusing on two-frame optical flow methods. Since the optical flow field between two frames captures the displacement of pixels over time, exploiting longer time sequences may help in ambiguous situations such as occluded regions [8, 18, 42]. Despite the long history of multi-frame optical flow estimation [5, 34], few methods have concretely demonstrated the benefits of multi-frame reasoning. In the following, we briefly review these multi-frame optical flow methods.

**Temporal smoothing.** The most straightforward way to leverage information across more than a pair of frames is to encourage temporal smoothness of the optical flow. Nagel imposes a spatio-temporal oriented smoothness constraint that is enforced along image edges, but relaxed across them [35]. Weickert and Schnörr also propose to combine temporal and spatial smoothness into a variational penalty term, but relax this constraint at flow discontinuities [46]. Zimmer *et al.* propose a principled way to attenuate the temporal discontinuity: they weigh the spatio-temporal smoothness based on how well it predicts subsequent frames [51]. Instead of using variational temporal smoothness terms, optical flow estimation can also be cast as a Kalman filtering problem [17, 14].

**Explicit trajectory reasoning.** Enforcing temporal smoothness of the optical flow is helpful, but it neglects higher level information such as the trajectory of objects, which may cause large and small displacements to coexist. A different, arguably more principled solution is to warp the optical flow estimation from previous frames to the current. Assuming the acceleration of a patch to be constant, Black and Anandan warp the velocity predicted from the previous frame to the current one, obtaining $\hat{\mathbf{w}}(t)$ [7]. They formulate the temporal constraint as $f(\hat{\mathbf{w}}(t) - \mathbf{w}(t))$,

where $\mathbf{w}(t)$ is the velocity estimated at the current frame. One shortcoming of this approach is that information is only propagated forward. Werlberger *et al.* tackle this problem by using the forward flow from $t-1$ and the backward flow from $t+1$ in the brightness constancy term for frame $t$ [47]. On the other hand, Chaudhury and Mehrotra pose optical flow estimation as the problem of finding the shortest path with the smallest curvature between pixels in frames of a sequence [13]. Similarly, Volz *et al.* enforce temporal consistency from past and future frames in two ways: accumulating errors across all frames leads to spatial consistency, and trajectories are also encouraged to be smooth [44].

**Multi-frame approaches.** Irani observes that the set of plausible optical flow fields across multiple frames lie on a lower dimensional linear subspace and does not require explicitly enforcing temporal or spatial smoothness [25]. However, the observation applies to rigid scenes and cannot be directly applied to dynamic scenes. Noting that the optical flow may not be temporally consistent, Sun *et al.* propose to impose consistency of the scene structure, *i.e.*, layered segmentation [45], over time, and combine motion estimation and segmentation in a principled way [42]. Wulff *et al.* decompose a scene into rigid and non-rigid parts, represent the flow in the rigid regions via depth, and regularize the depth over three frames [48]. Both these formulations, unfortunately, lead to a large optimization problem that is computationally expensive to solve.

**Flow fusion.** Fusion is a widely used approach to integrate different estimates of the flow fields into a more accurate one. Lempitsky *et al.* generate flow candidates by applying Lucas and Kanade [28] and Horn and Schunck [23] with different parameters settings to the same two input frames, and fuse these candidates via energy minimization [27]. To deal with small and fast moving objects, Xu *et al.* obtain flow candidates by feature matching and fuse them with the estimates by a variational approach in a coarse-to-fine pyramid [50]. Mac Aodha *et al.* generate the flow candidates by applying a variety of different algorithms and fuse them using a random forest classifier [29]. These methods are all designed for two input frames and do not consider multi-frame temporal cues. Kennedy and Taylor acquire several flow estimates from adjacent frames and fuse them using a random forest classifier [26]. However, their system requires an additional post-processing step and is not end-to-end trainable. One limitation of the discrete fusion approach is that a pixel only picks up flow vectors from the flow candidates at the same spatial location, although neighboring regions may provide better candidates. Recently, Maurer and Bruhn proposed a fusion-based flow estimation method that is closely related to our approach, but the algorithm does not outperform two-frame methods on all datasets [31].

**Deep learning-based two-frame methods.** The recent success of deep learning inspired several approaches based on neural networks, which can either replace parts of the traditional pipeline [49, 1], or can be used in an end-to-end fashion [16, 24, 37, 43]. The top-performing deep-learning optical flow algorithms are still based on two frames and do not consider additional temporal information. In this paper, we apply our fusion approach to both FlowNetS and PWC-Net and show that temporal information leads to significant improvements to both methods.

## 3. Proposed Model

For presentation clarity, we focus on three-frame optical flow estimation. Given three input frames $\mathbf{I}_{t-1}$, $\mathbf{I}_t$, and $\mathbf{I}_{t+1}$, our aim is to estimate the optical flow from frame $t$ to frame $t+1$, $\mathbf{w}_{t\rightarrow t+1}^f$. The superscript '$f$' indicates that it fuses information from all of the frames, as opposed to $\mathbf{w}_{t\rightarrow t+1}$, which only uses $\mathbf{I}_t$ and $\mathbf{I}_{t+1}$.

We hypothesize that the ability to leverage the information from $\mathbf{w}_{t-1\rightarrow t}$ can help in several ways. First, it can act as a regularizer, since the optical flow between consecutive pairs of frames changes smoothly in most regions. Perhaps more importantly, however, $\mathbf{w}_{t-1\rightarrow t}$ can offer information that is complementary to that of $\mathbf{w}_{t\rightarrow t+1}$, especially for pixels that are occluded or out of boundary in the current frame or the next. Finally, given those two observations, the optimal estimate for $\mathbf{w}_{t\rightarrow t+1}^f$ may also benefit from information from local neighborhoods in both images.

In this section, we begin by verifying that temporal information indeed helps and by determining the image regions where it is most informative. We then introduce the proposed multi-frame fusion architecture, and discuss two deep learning baselines for multi-frame optical flow estimation.

### 3.1. Temporal Information: An "Oracle" Study

We evaluate the benefits of temporal information using the virtual KITTI [19] and Monkaa [32] datasets. We split each video sequence into three-frame mini-sequences and use two-frame methods to estimate three motion fields, $\mathbf{w}_{t\rightarrow t+1}$, $\mathbf{w}_{t-1\rightarrow t}$, and $\mathbf{w}_{t\rightarrow t-1}$. We first warp these optical flow fields into a common reference frame. Specifically, we backward warp $\mathbf{w}_{t-1\rightarrow t}$ with $\mathbf{w}_{t\rightarrow t-1}$:

$$\widehat{\mathbf{w}}_{t\rightarrow t+1} = \mathcal{W}(\mathbf{w}_{t-1\rightarrow t}; \mathbf{w}_{t\rightarrow t-1}), \qquad (1)$$

where $\mathcal{W}(\mathbf{x}; \mathbf{w})$ denotes the result of warping the input $\mathbf{x}$ using the flow field $\mathbf{w}$.

We then compare warped and current flows with the ground truth at every pixel, and select the one closer to the ground truth to obtain an "oracle" flow field. We test the "oracle" flow fields for two flow methods, FlowNetS and PWC-Net. As shown in Table 1, the "oracle" flow fields are more accurate than the current flow estimated by two-frame

methods, particularly in occluded and out-of-boundary regions. The results confirm that the warped previous optical flow provides complementary information and also suggest a straightforward mechanism for multi-frame flow estimation.

### 3.2. Temporal FlowFusion

In a nutshell, our architecture consists of two neural networks. The first computes the optical flow between adjacent frames. These optical flow estimates are then warped onto the same frame, producing two optical flow candidates. A second network then fuses these two. Both networks are pre-trained and then fine-tuned in an end-to-end fashion, see Figure 2.

More specifically, our system first uses a two-frame method, such as PWC-Net [43], to estimate three flow fields $\mathbf{w}_{t\rightarrow t+1}$, $\mathbf{w}_{t-1\rightarrow t}$, and $\mathbf{w}_{t\rightarrow t-1}$. Any other two-frame method could be used in place of PWC-Net. However, we pick PWC-Net as it currently is the top-performing optical flow method on both the KITTI 2015 and Sintel final benchmarks.

Now we have two candidates for the same frame: $\widehat{\mathbf{w}}_{t\rightarrow t+1}$ and $\mathbf{w}_{t\rightarrow t+1}$. Thanks to the warping operation, these two optical flow should be similar (identical in regions where the velocity is constant, the three optical flow estimations are correct, and there are no occlusions). To fuse these, we take inspiration from the work of Ilg *et al.* who perform optical flow fusion in the spatial domain for two-frame flow estimation [24]. Specifically, they propose to estimate two optical flows, one capturing large displacements and one capturing small displacements, and to fuse them using a U-Net architecture. The input to their fusion network are the flow candidates, their brightness constancy error, flow magnitude, and the first frame. The magnitude of the flow is an important feature for their network to distinguish large and small motion.

We extend this approach to the temporal domain. Our fusion network takes two flow estimates $\widehat{\mathbf{w}}_{t\rightarrow t+1}$ and $\mathbf{w}_{t\rightarrow t+1}$, the corresponding brightness constancy errors

$$E_{\widehat{\mathbf{w}}} = |\mathbf{I}_t - \mathcal{W}(\mathbf{I}_{t+1}; \widehat{\mathbf{w}}_{t\rightarrow t+1})|, \text{ and} \qquad (2)$$

$$E_{\mathbf{w}} = |\mathbf{I}_t - \mathcal{W}(\mathbf{I}_{t+1}; \mathbf{w}_{t\rightarrow t+1})|, \qquad (3)$$

as well as the current frame $\mathbf{I}_t$. Since we are not trying to fuse optical flow fields that were estimated at different scales, the magnitude of the flow, which Ilg *et al.* employ, may not be useful and potentially detrimental. Other inputs could be used, such as a measure of smoothness between the previous flow, $\mathbf{w}_{t-1\rightarrow t}$, and the current, $\mathbf{w}_{t\rightarrow t+1}$. However, we found that the simpler and lighter-weight input we use works best in our experiments. We provide more analysis of the behavior of our fusion network in Section 4.

Note that, while we take inspiration from FlowNet2 [24] for the fusion step, our approach and our goal differ from
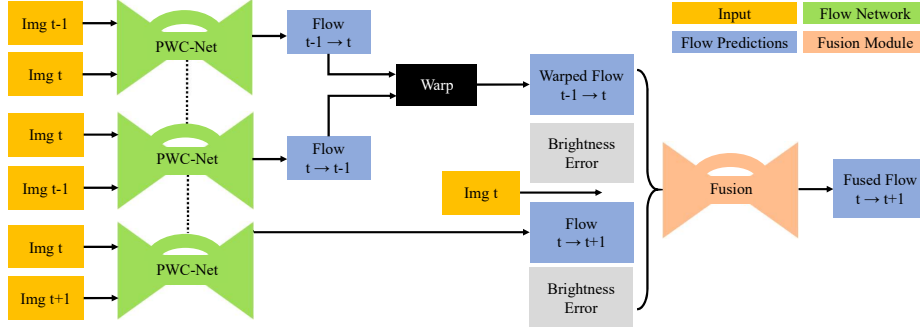
Figure 2: Architecture of the proposed fusion approach for three-frame optical flow estimation. The dashed line indicates that the PWC-Nets share the same weights. PWC-Net can be replaced with other two-frame flow methods like FlowNetS.
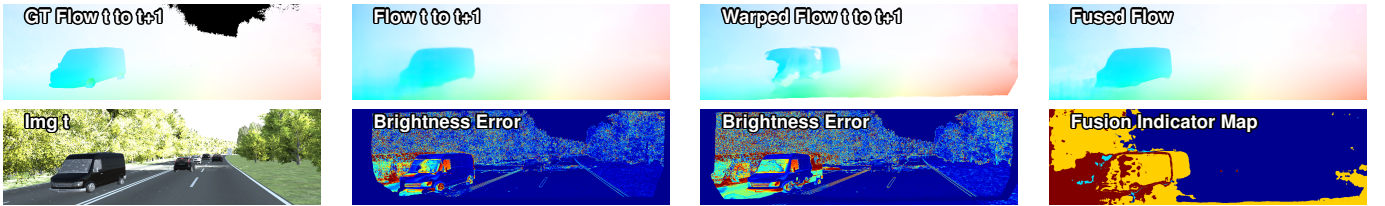


Figure 3: Visualizing the input and behavior of the fusion network. In the fusion indicator map, we compare the fused flow with two flow candidates. Blue: both candidates are similar ($< 5$px), yellow: the fused flow is similar to $\mathbf{w}_{t \to t+1}$, and cyan: the fused flow is similar to $\widehat{\mathbf{w}}_{t \to t+1}$. Red: the fused flow is different from all flow candidates (mostly occluded regions).

theirs conceptually: we explore and leverage the benefits of temporal information.

## 3.3. Deep-Learning Baseline Methods

To the best of our knowledge, our flow fusion method is the first realtime deep learning-based method to fuse temporal information in the context of optical flow. To evaluate the performance of our approach, we build on standard deep-learning optical flow algorithms, and propose two baseline methods. We extensively validate our method in Section 4 and find that our fusion network, despite its simplicity, works consistently better than those baselines. Here, we first briefly review the methods upon which we build, and then introduce the proposed baselines.

**FlowNetS++** The most widely used deep network structure for optical flow prediction is FlowNetS [16], which is a standard U-Net structure. The most natural way to extend this model to multi-frame prediction tasks is to copy the encoded features from the previous pair of images to the current frame (Figure 4, left). The inner-most high-dimensional feature of the previous frame may carry instructive information to guide motion prediction at the next frame. We call this FlowNetS++.

**FlowNetS + GRU** In deep learning, temporal information is often captured with Recurrent Neural Networks (RNNs). In recent years, the popular LSTM [22] and GRU [15] network architectures have been applied to a variety of vi-
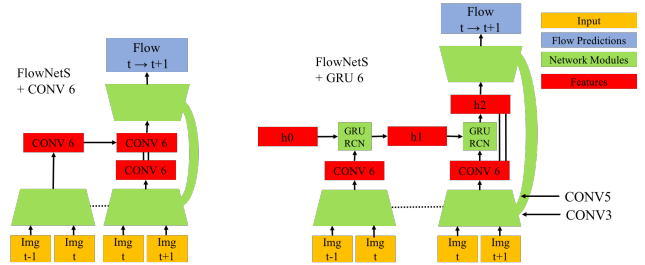


Figure 4: Baseline network structures for modeling temporal information. FlowNetS++ (left) simply concatenates the encoded features across different frames, and FlowNetS + GRU6 (right) propagates the encoded features through GRU-RCN units. The dotted lines indicate that two sub-networks share the same weights, while the double vertical lines denote the feature concatenation operation.

sion tasks, such as action recognition [30] and segmentation [12]. GRUs have a comparatively smaller model size and the convolutional extension GRU-RCN [3] is widely used to extract abstract representations from videos. We propose to use GRU-RCN to propagate encoded features in previous frames through time in a GRU-RCN unit and introduce a network structure, which we dub FlowNetS + GRU (Figure 4, right). We preserve the overall U-Net structure and apply GRU-RCN units at different levels of the encoder with different spatial resolutions. Encoded features at the

sixth level are the smallest in resolution.

**PWC-Net + GRU** We can apply a similar strategy to PWC-Net [43], a recently introduced network that achieves excellent performance for two-frame optical flow prediction tasks. The network first feeds two images into separate siamese networks, which consist of a series of convolutional structures. Then it decodes features and learns abstract representations at different levels. Similarly to FlowNetS + GRU, we can also feed encoded features at different levels to GRU-RCN units and we call it PWC-Net + GRU. A depiction of this second network structure is omitted for clarity.

## 4. Experimental Results

We use two different architectures as building blocks: FlowNetS [16] for its widespread adoption, and PWC-Net [43] for its efficiency and performance on standard benchmarks. We pretrain each of these on the "FlyingChairs" dataset [16] first, and then use the learned weights to initialize the different network structures for multiple frame optical flow estimations. For consistency among different multi-frame algorithms, we use three frames as inputs, and report ablation studies on the virtual KITTI dataset [20] and the Monkaa dataset [32]. Finally, we report the performance of our algorithms on the two public benchmarks KITTI [21] and Sintel [11].

### 4.1. Implementation Details

Traditional optical datasets [2, 4] contain a small number of images and thus are not suitable for training modern deep network structures. Dosovitskiy *et al.* find that accurate models can be trained using computer graphics rendered images and introduce the "FlyingChairs" dataset as well as a corresponding learning rate schedule [16]. A follow-up work then investigates a better learning rate schedule for training using multiple datasets [24]. We use the same strategy $S_{\text{long}}$ proposed by Ilg *et al.* [24] to pretrain our two-frame optical flow algorithms using the "FlyingChairs" and "FlyingThings3D" datasets.

Since we aim at predicting optical flow from multi-frame inputs, we train on datasets that offer videos, rather than just pairs of images. The Monkaa dataset consists of 24 video sequences with 8591 frames and the virtual KITTI dataset consists of 5 different scenes with 21,260 frames. For our ablation study, we split both Monkaa and virtual KITTI datasets into non-overlapping training and testing portions. Specifically, Monkaa has 55104 frames for training and 10944 frames for validation and virtual KITTI has three sequences (14600 frames) for training and two sequences (5600 frames) for validation.

After pretraining our Fusion model using Monkaa and virtual KITTI, we fine-tune them using the training set of the Sintel and KITTI benchmarks, and compare our method with state-of-the-art algorithms on the two public benchmarks. When fine-tuning on the Sintel dataset, we use pretrained weights from the Monkaa dataset because both datasets consist of animated cartoon characters generated from the open source Blender software. For fine-tuning on the KITTI dataset, we use pretrained weights from the virtual KITTI dataset. We adopt the robust training loss from Sun *et al.* [43]:

$$L(\Theta) = \sum_{l=l_0}^{L} \alpha_l \sum_{\mathbf{x}} (|\mathbf{w}_\Theta^l(\mathbf{x}) - \mathbf{w}_{GT}^l(\mathbf{x})| + \epsilon)^q + \gamma\|\Theta\|_2^2.$$

We follow the notation defined by Sun *et al.* [43] and we refer readers to their paper for more details. Briefly, $\alpha_l$ is the weight at pyramid level $l$; for smaller spatial resolutions, we set a higher weight for $\alpha_l$. The robustness parameter $q$ controls the penalty for outliers in flow predictions, $\epsilon > 0$ is a small constant, and $\gamma$ determines the amount of regularization. Parameters are set as follows:

$$\alpha = (0.005, 0.01, 0.02, 0.08, 0.32)$$
$$\epsilon = 0.01, q = 0.4, \gamma = 0.0004.$$

For multi-frame networks using GRU-RCN units, we use 256-dimensional hidden units. For fusion networks, the network structure is similar to FlowNet2 [24] except for the first convolution layer, because our input to the fusion network has different channels. For the single optical flow prediction output by our fusion network, we set $\alpha = 0.005$ in the loss function and use learning rate 0.0001 for fine-tuning. We implement our models using the PyTorch framework [36], including the pretrained models for FlowNetS and PWC-Net. Our implementation matches the official Caffe implementation of PWC-Net.

### 4.2. Ablation Study

#### 4.2.1 Two-frame and Multi-frame Methods

We perform an ablation study of the two-frame and multi-frame methods using the virtual KITTI and Monkaa datasets, as summarized in Table 1.

FlowNetS++ shows a small but consistent improvement over FlowNetS, in every different region, including pixels that move within (Inside) and out of (Outside) the image boundaries or get occluded. The results show the benefit of using temporal information for optical flow, even in the most naïve and straightforward way. FlowNetS+GRU has a relatively larger improvement compared to FlowNetS++, suggesting the advantage of using better models to capture the dynamics over time. However, there is no obvious winner among architectures augmented by using GRUs at different pyramidal levels.

The Fusion approach consistently outperforms all other methods, including those using the GRU units. This find-

Table 1: Ablation study on the *validation* set of the virtual KITTI dataset (top) and Monkaa dataset (bottom). Training a temporal network structure helps improve two-frame optical flow predictions, but the fusion approach has significantly lower end-point-error (EPE). Inside means pixels move within the image boundary and Outside means pixels that move out of the image boundaries.

| | FlowNetS | FlowNetS++ | GRU 3 | GRU 4 | GRU 5 | GRU 6 | Fusion | Oracle | PWC-Net | GRU 3 | GRU 4 | GRU 5 | GRU 6 | Fusion | Oracle |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EPE All | 6.12 | 5.90 | 5.26 | 5.40 | 5.15 | 5.32 | **5.00** | 4.35 | 2.34 | 2.17 | 2.13 | 2.12 | 2.16 | **2.07** | 1.80 |
| EPE Inside | 4.03 | 3.87 | 3.61 | 3.64 | 3.58 | 3.59 | **3.14** | 2.62 | 1.60 | 1.44 | 1.41 | 1.40 | 1.42 | **1.37** | 1.20 |
| EPE Outside | 28.97 | 27.57 | 23.26 | 24.60 | **22.28** | 24.25 | 25.15 | 21.83 | 10.43 | 10.01 | 9.94 | 10.02 | 9.86 | **9.71** | 7.90 |
| EPE Occlusion | 7.44 | 7.11 | **5.93** | 6.27 | 5.82 | 6.18 | 6.14 | 4.63 | 2.41 | 2.29 | **2.24** | **2.24** | 2.26 | 2.27 | 1.82 |
| | FlowNetS | FlowNetS++ | GRU 3 | GRU 4 | GRU 5 | GRU 6 | Fusion | Oracle | PWC-Net | GRU 3 | GRU 4 | GRU 5 | GRU 6 | Fusion | Oracle |
| EPE All | 2.07 | 2.06 | 2.56 | 2.45 | 2.34 | 2.27 | **1.97** | 1.89 | 1.19 | 1.26 | 1.23 | 1.27 | 1.27 | **1.18** | 1.03 |
| EPE Inside | 1.91 | 1.89 | 2.37 | 2.27 | 2.16 | 2.09 | **1.8** | 1.75 | 1.19 | 1.26 | 1.23 | 1.27 | 1.27 | **1.18** | 1.03 |
| EPE Outside | 11.47 | **11.12** | 13.13 | 12.76 | 12.65 | 12.43 | 11.43 | 9.79 | 8.00 | 8.16 | 8.11 | 8.55 | 8.42 | **7.94** | 6.90 |
| EPE Occlusion | 8.16 | 8.02 | 9.17 | 9.07 | 8.84 | 8.71 | **7.89** | 7.00 | 5.83 | 5.97 | 5.90 | 5.80 | 5.79 | **5.67** | 4.88 |

Table 2: Experimental results on the MPI Sintel dataset [11] (left) and KITTI [21] (right). Our PWC-Fusion method ranks first among all published methods.

| | EPE | Match | Unmatch | d0-10 | d10-60 | d60-140 | s0-10 | s10-40 | s40+ | | Fl-all-Occ | Fl-fg-Occ | Fl-bg-Occ | Fl-all-Ncc | Fl-fg-Ncc | Fl-bg-Ncc |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PWC-Fusion | **4.566** | **2.216** | 23.732 | **4.664** | **2.017** | **1.222** | 0.893 | 2.902 | 26.810 | PWC-Fusion | **7.17** | **7.25** | **7.15** | **4.47** | **4.25** | **4.52** |
| PWC-Net | 4.596 | 2.254 | **23.696** | 4.781 | 2.045 | 1.234 | 0.945 | 2.978 | **26.620** | PWC-Net | 7.90 | 8.03 | 7.87 | 5.07 | 5.04 | 5.08 |
| ProFlow | 5.015 | 2.659 | 24.192 | 4.985 | 2.185 | 1.771 | 0.964 | 2.989 | 29.987 | LiteFlowNet | 9.38 | 7.99 | 9.66 | 5.49 | 5.09 | 5.58 |
| DCFlow | 5.119 | 2.283 | 28.228 | 4.665 | 2.108 | 1.440 | 1.052 | 3.434 | 29.351 | MirrorFlow | 10.29 | 17.07 | 8.93 | 7.46 | 12.95 | 6.24 |
| FlowFieldsCNN | 5.363 | 2.303 | 30.313 | 4.718 | 2.020 | 1.399 | 1.032 | 3.065 | 32.422 | SDF | 11.01 | 23.01 | 8.61 | 8.04 | 18.38 | 5.75 |
| MR-Flow | 5.376 | 2.818 | 26.235 | 5.109 | 2.395 | 1.755 | 0.908 | 3.443 | 32.221 | UnFlow | 11.11 | 15.93 | 10.15 | 7.46 | 12.36 | 6.38 |
| LiteFlowNet | 5.381 | 2.419 | 29.535 | 4.090 | 2.097 | 1.729 | **0.754** | **2.747** | 34.722 | MRFlow | 12.19 | 22.51 | 10.13 | 8.86 | 17.91 | 6.86 |
| S2F-IF | 5.417 | 2.549 | 28.795 | 4.745 | 2.198 | 1.712 | 1.157 | 3.468 | 31.262 | ProFlow | 15.04 | 20.91 | 13.86 | 10.15 | 17.9 | 8.44 |

ing suggests that the warping operation can effectively capture the motion trajectory over time, which is missing in the GRU units. We provide visualizations of the predicted optical flow in the ablation study in Figure 5.

#### 4.2.2 The Design of the Fusion Network

We also performed an additional ablation study to evaluate the relative importance of each input. Our fusion network takes as inputs the optical flow candidates, the brightness errors, and the input image. On the validation set of virtual KITTI, its average EPE is **2.07**. We observe a moderate performance degradation if we remove the input image (EPE=**2.15**) or the brightness error (EPE=**2.16**). Adding the flow magnitude to the inputs, on the other hand, leads to almost identical performance (EPE=**2.09**). FlowNet2 fuses optical flow predictions of two different networks trained for large and small displacements given the same two frames, and thus flow magnitude is a valuable feature for fusion. In contrast, we fuse temporal rather than spatial information. The warped and current flows are estimated by the same network, making the magnitude less valuable for fusion. On average, the multi-frame fusion step only takes **0.02s** in our implementation.

#### 4.2.3 Using More than Three Frames as Input Helps

We also ran oracle experiments using input images with up to 5 frames, and EPE scores are 1.80 (3 frames), 1.64 (4 frames) and 1.56 (5 frames). The results suggest that using more than three frames does help further, but the relative improvements become smaller. In this paper, our main contribution is a deep-learning framework that allows to leverage temporal information. Its extension to more than three frames does not require any conceptual modifications, and is a promising area for future research.

### 4.3. The Behavior of the Fusion Network

We analyzed the average EPE (AEPE) of the flow fields where the fused flow is different than either the current or the warped flow (red regions in Figure 3). On the validation set of virtual KITTI, the AEPE of the current flow is 41.88, much higher than 32.71 of the warped flow, suggesting that the latter carries useful information. Moreover, the oracle flow has an AEPE of 21.89, confirming that the flow in those regions is incorrectly estimated by both. The fused flow has an AEPE of 28.86, lower than both candidate flows but higher than the oracle flow—it does not use the ground truth after all. However, we notice that the fused flow has lower errors than the oracle flow for **37.65%** of pixels in the red regions, meaning that the estimate is better than any of the inputs. Again, this is one main advantage of our approach with respect to methods that simply select one of the input flow vectors [26].

### 4.4. Comparison with the State-of-the-art

We take the pretrained models of PWC-Net [43] along with fusion network trained on virtual KITTI and Monkaa, and fine-tune the fusion network using the training data from the public benchmarks KITTI [21] and Sintel [11]. On the Sintel benchmark (Table 2, left), PWC-Fusion consistently outperforms PWC-Net [43], particularly in
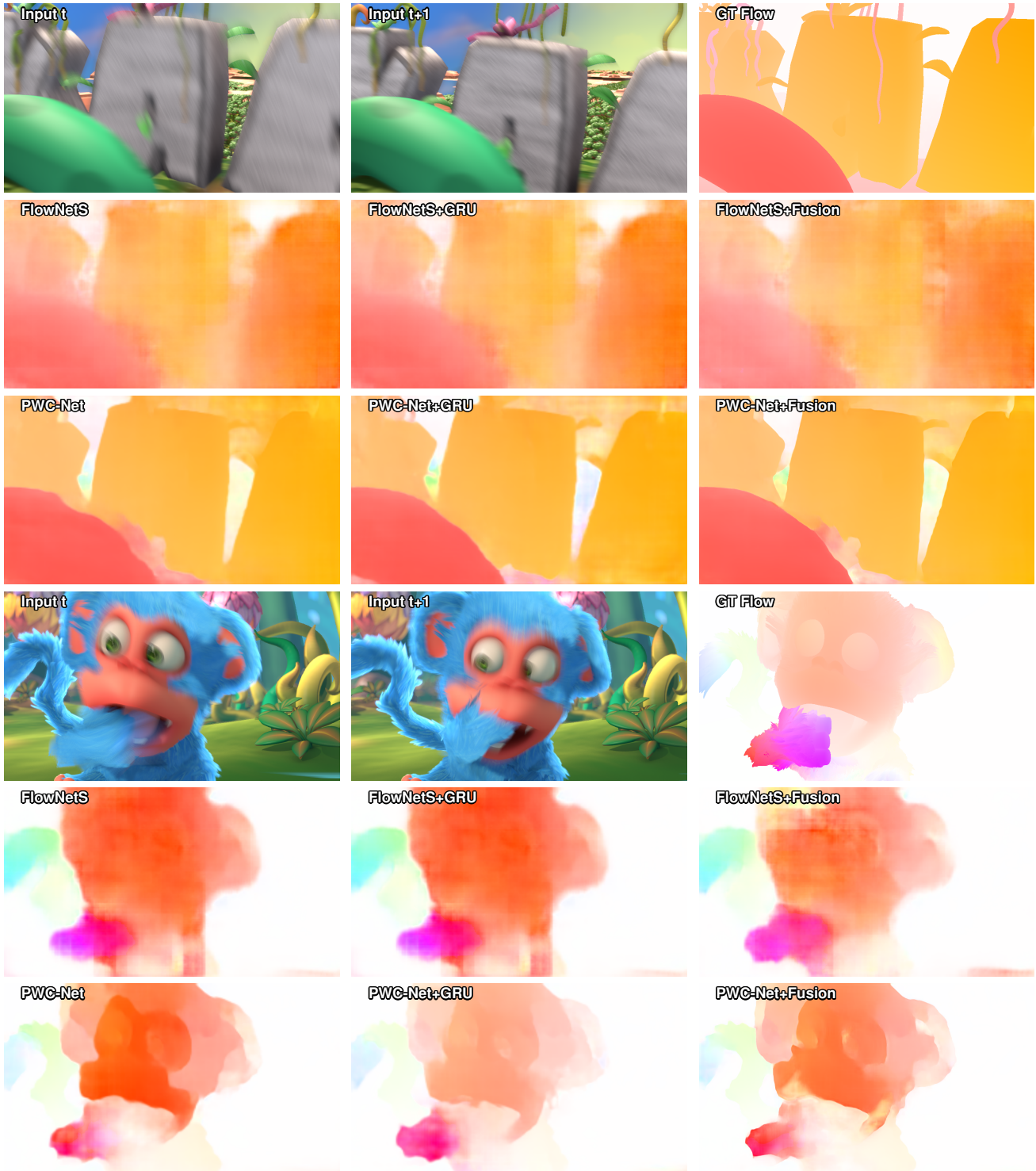
Figure 5: Visualizations of optical flow outputs in the ablation study.

unmatched (occlusion) regions. Its performance is also better than a recent multi-frame optical flow estimation method [31]. On the KITTI benchmark (Table 2, right), PWC-Fusion outperforms all two-frame optical flow methods including the state-of-the-art PWC-Net. This is also the first time a multi-frame optical flow algorithm consis-
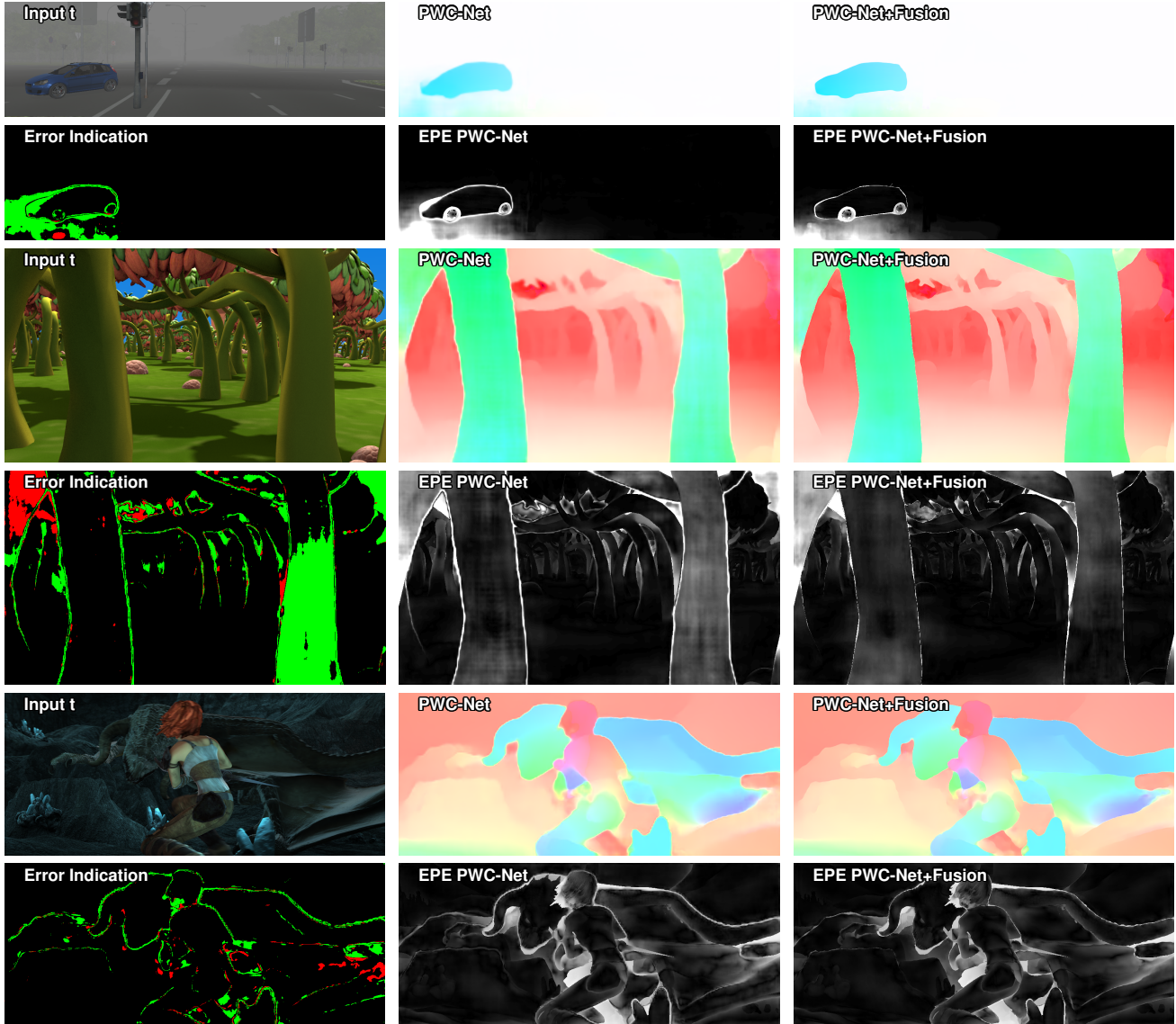
Figure 6: Visual results of our fusion method. Our method produces consistently better results across different datasets. Notably, our prediction make more accurate predictions at motion boundaries (green pixels in the indication map mean that PWC-Net+Fusion is more accurate than PWC-Net, and red pixels mean that PWC-Net is better).

tently outperforms two-frame approaches across different datasets. We provide some visual results in Figure 6, and provide indicator maps to demonstrate the behavior of fusion network.

## 5. Conclusions

We have presented a simple and effective fusion approach for multi-frame optical flow estimation. Our main observation is that multiple frames provide new information beyond what is available only looking at two adjacent frames, in particular for occluded and out-of-boundary pixels. Thus we proposed to fuse the warped previous optical flow with the current optical flow estimate. Extensive experiments demonstrate the benefit of our fusion approach:

it outperforms both two-frame baselines and sensible multi-frame baselines based on GRUs. Moreover, it is top-ranked among all published flow methods on the MPI Sintel and KITTI 2015 benchmark.

While the performance of two-frame methods has been steadily pushed over the years, we begin to see diminishing returns. Our work shows that it could be fruitful to exploit temporal information for optical flow estimation and we hope to see more work in this direction.

# References

[1] C. Bailer, K. Varanasi, and D. Stricker. Cnn-based patch matching for optical flow with thresholded hinge loss. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[2] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision (IJCV)*, 2011.

[3] N. Ballas, L. Yao, C. Pal, and A. Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015.

[4] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision (IJCV)*, 1994.

[5] M. J. Black and P. Anandan. A model for the detection of motion over time. In *Proc. of the IEEE International Conference on Computer Vision (ICCV)*, pages 33–37, 1990.

[6] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1991.

[7] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1991.

[8] M. J. Black and D. J. Fleet. Probabilistic detection and tracking of motion boundaries. *International Journal of Computer Vision (IJCV)*, 38(3):231–245, July 2000.

[9] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2004.

[10] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2011.

[11] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012.

[12] W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki. Scene labeling with lstm recurrent neural networks. In *CVPR*, pages 3547–3555, 2015.

[13] K. Chaudhury and R. Mehrotra. A trajectory-based computational model for optical flow estimation. *IEEE Transactions on Robotics and Automation (TRA)*, 1995.

[14] T. M. Chin, W. C. Karl, and A. S. Willsky. Probabilistic and sequential computation of optical flow using temporal coherence. *IEEE Transactions on Image Processing (TIP)*, 1994.

[15] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[16] A. Dosovitskiy, P. Fischery, E. Ilg, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox, et al. Flownet: Learning optical flow with convolutional networks. In *Proc. of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[17] M. Elad and A. Feuer. Recursive optical flow estimation-adaptive filtering approach. *Journal of Visual Communication and Image Representation*, 1998.

[18] D. Feldman and D. Weinshall. Motion segmentation and depth ordering using an occlusion detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(7):1171–1185, July 2008.

[19] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *CVPR*, 2016.

[20] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[21] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[22] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[23] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 1981.

[24] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[25] M. Irani. Multi-frame optical flow estimation using subspace constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.

[26] R. Kennedy and C. J. Taylor. Optical flow with geometric occlusion estimation and fusion of multiple frames. In *EMM-CVPR*, pages 364–377. Springer, 2015.

[27] V. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion moves for Markov random field optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(8):1392–1405, Aug. 2010.

[28] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, pages 674–679, 1981.

[29] O. Mac Aodha, G. J. Brostow, and M. Pollefeys. Segmenting video into classes of algorithm-suitability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1778–1785, 2010.

[30] J. Martinez, M. J. Black, and J. Romero. On human motion prediction using recurrent neural networks. In *CVPR*, Piscataway, NJ, USA, July 2017. IEEE.

[31] D. Maurer and A. Bruhn. Proflow: Learning to predict optical flow. *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[32] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. 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 (CVPR)*, 2016.

[33] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[34] D. W. Murray and B. F. Buxton. Scene segmentation from visual motion using global optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 9(2):220–228, Mar. 1987.

[35] H.-H. Nagel. Extending the oriented smoothness constraint into the temporal domain and the estimation of derivatives of optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 1990.

[36] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. De-Vito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[37] A. Ranjan and M. J. Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[38] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[39] A. Salgado and J. Sánchez. Temporal constraints in large optical flow estimation. In *Computer Aided Systems Theory – EUROCAST*, 2007.

[40] P. Sand and S. Teller. Particle video: Long-range motion estimation using point trajectories. *International Journal of Computer Vision (IJCV)*, 80(1):72–91, Oct. 2008.

[41] D. Sun, S. Roth, and M. J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision (IJCV)*, 2014.

[42] D. Sun, E. B. Sudderth, and M. J. Black. Layered segmentation and optical flow estimation over time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1768–1775, 2012.

[43] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[44] S. Volz, A. Bruhn, L. Valgaerts, and H. Zimmer. Modeling temporal coherence for optical flow. In *Proc. of the IEEE International Conference on Computer Vision (ICCV)*, 2011.

[45] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5):625–638, Sept. 1994.

[46] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 2001.

[47] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-$L^1$ optical flow. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2009.

[48] J. Wulff, L. Sevilla-Lara, and M. J. Black. Optical flow in mostly rigid scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[49] J. Xu, R. Ranftl, and V. Koltun. Accurate optical flow via direct cost volume processing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[50] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2012.

[51] H. Zimmer, A. Bruhn, and J. Weickert. Optic flow in harmony. *International Journal of Computer Vision (IJCV)*, 2011.