Separable Flow: Learning Motion Cost Volumes for Optical Flow Estimation

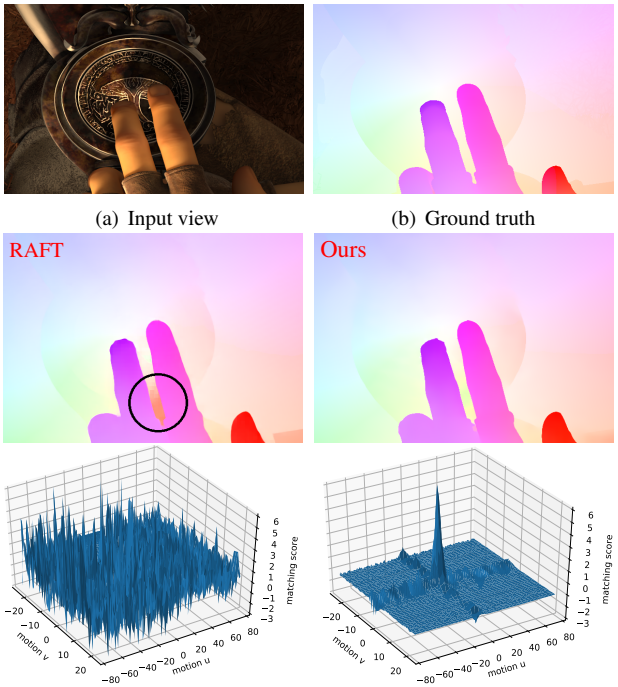
Feihu Zhang* Oliver J. Woodford Victor Prisacariu* Philip H.S. Torr*
*University of Oxford

Abstract

Full-motion cost volumes play a central role in current state-of-the-art optical flow methods. However, constructed using simple feature correlations, they lack the ability to encapsulate prior, or even non-local knowledge. This creates artifacts in poorly constrained ambiguous regions, such as occluded and textureless areas. We propose a separable cost volume module, a drop-in replacement to correlation cost volumes, that uses non-local aggregation layers to exploit global context cues and prior knowledge, in order to disambiguate motions in these regions. Our method leads both the now standard Sintel and KITTI optical flow benchmarks in terms of accuracy, and is also shown to generalize better from synthetic to real data.

1. Introduction

Optical flow is the task of estimating per-pixel 2D motion between two images or video frames. This low-level vision task is a fundamental building block of many higher level tasks, such as object tracking, scene reconstruction and video compression. A common approach to this task, used in both hand designed [5, 19] and more modern deep-learning methods [53, 54], is to first compute a cost volume for motions of all pixels, then use this to infer or refine a motion per pixel. While state-of-the-art methods [54, 62] tend to use this approach, it suffers from two key challenges. First, the cost volume size is exponential in the dimensionality of the search space. Therefore memory and computation requirements for optical flow, with its 2D search space, grow quadratically with the range of motion. In contrast, such costs for the 1D stereo matching task grow only linearly with the range of disparity. Secondly, resolving ambiguities caused by occlusion, lack of texture, or other such issues requires a more global, rather than local, understanding of the scene, as well as prior knowledge. Cost volumes generally do not encapsulate such information, leaving the job of resolving such ambiguities to the second stage of each method. As Fig. 1 & 4 illustrate, this makes it harder to compute accurate motion in such regions.



(a) Input view (b) Ground truth

(c) RAFT [54] output & cost volume (d) Our output & cost volume

Figure 1: Performance illustrations. (a) Input view from Sintel. (b) Ground truth optical flow. (c) The optical flow result and 2D motion cost volume (for a single pixel in the circled region) of the state of the art, RAFT [54]. (d) Result and cost volume (for the same pixel) learned by our Separable Flow. RAFT does not predict motion accurately in the ambiguous regions, such as occlusions (highlighted by the circle). Indeed, there are many false peaks in the cost volume for this region. In contrast, Separable Flow predicts accurate flow results in these challenging regions, by integrating separable, non-local matching cost aggregations. The resulting learned cost volume has one large peak, that correctly matches the ground truth. See sec. 4.2 for more details.

This work proposes a new separable cost volume computation module, which plugs into existing cost-volume-based optical flow frameworks, with two key innovations that address these challenges. The first is to separate the 2D motion of optical flow into two independent 1D problems, horizontal and vertical motion, compressing the 4D cost volume

Code: https://github.com/feihuzhang/SeparableFlow

into two smaller 3D volumes using a self-adaptive separation layer. This factored representation significantly reduces the memory and computing resources required to infer (and thus also learn) the cost volumes, making them linear in the range of motion, without loss in accuracy. Moreover, it enables the second innovation: the use of non-local aggregation layers to learn a refined cost volume. Such layers have previously been used for 1D stereo problems [67, 68], where they improve both accuracy in ambiguous regions, and cross-domain generalization. We apply them here to optical flow for the first time, learning cost volumes with non-local, prior knowledge via a one-step motion regression that is able to predict a low-resolution (*i.e*. 1/8), but high-quality motion. This prediction also serves as a better input to the interpolation and refinement module.

We train and evaluate our Separable Flow module on the standard Sintel [7] and KITTI [16] optical flow datasets. We achieve the current best accuracy among all published optical flow methods on both these benchmarks. Moreover, in the cross-domain case of training on synthetic and testing on real data (*i.e*. KITTI), our results improve the previous state of the art by a greater margin, even outperforming some DNN models (*e.g*. FlowNet2 [28] and PWC-Net [53]) fine-tuned on the target KITTI scenes. We provide an ablation study to show how much of this improvement is attributable to each of our contributions. We reiterate that any optical flow framework that computes a cost volume can benefit from these improvements.

2. Related Work

We now review prior work related to our method, with a focus on traditional and neural-network-based optical flow, and cost aggregation methods in stereo.

2.1. Traditional Approaches

There are three main types of traditional optical flow method. The first is usually based on local filtering [20], interpolation [21, 48, 63], nearest neighbor search [2, 22, 39, 40, 49] or dense inverse search [34]. The second usually optimizes a global energy function that consists of a local matching cost data term and an MRF-based smoothness regularization term, using gradient-based solvers [5, 6, 19, 45, 47, 57, 66].

Methods of the third type use discrete solvers [10, 43, 60] to find more globally optimal solutions to the global energy function. However, large motion ranges mean each pixel can be paired with any of thousands of discrete correspondences, leading to a huge search space. To address this issue, Menez *et al*. [43] prune the search space using feature descriptors, and optimize using message passing, whereas Chen *et al*. [10] use a distance transform to solve the global optimization problem over the full search space.

2.2. Deep Neural Networks for Optical Flow

A multitude of deep neural networks (DNNs) have been proposed to infer optical flow between a pair of frames, addressing many different aspects of the task. These include occlusion handling [70], robust loss functions [3, 15], feature representations [50, 69], refinement/interpolation [26, 54, 73], uncertainty estimation [27], lightweight architecture [24], data resampling [4], and motion estimation in dark scenes [71]. Several works jointly learn segmentation and optical flow [1, 11, 51, 58, 58], segmenting the image into objects or backgrounds and computing motion depending on the region type. Coarse-to-fine processing has emerged as a popular ingredient in many recent works [4, 18, 24–26, 46, 53, 62, 65, 70]. Self-supervised optical flow networks [29–31, 36, 37, 56, 64, 72] and semi-supervised frameworks [35, 61] have also been explored.

Among these methods, explicit cost volumes appear frequently, [18, 20, 23, 38, 53–55, 62], storing the data matching costs for each pixel's potential correspondences, and thus playing an important role in generating accurate flow fields. For example, PWC-Net [53] develops a DNN model using image pyramids, warping, and cost volumes. Xiao *et al*. [59] learn cost volumes using the Cayley representation, but without effective cost aggregations. Hui *et al*. [23] address the ambiguous matching challenge by improving the cost volume through an adaptive modulation prior, exploiting local flow consistency. Hofinger *et al*. [18] improve the cost volume construction process via a sampling-based strategy that revises the gradient flow across pyramid levels. Wang *et al*. [55] reshape a 4D cost volume into 3D via a displacement-aware projection (DAP) layer, learning the high-dimensional cost volume with low-dimensional convolutions. However, it can only process a fixed and small displacement range (*e.g*. -3,..,3). Yang *et al*. [62] propose a 5D volumetric encoder-decoder architecture with separable volumetric filtering. Designed for a local search window (*e.g*. -9,..,9), it cannot capture non-local knowledge in the cost volume.

In contrast to these methods, ours can learn and refine a full-range cost volume over the whole motion space, using non-local aggregations, as a result of our Separable Flow model. This is similar to Xu *et al*. [60], who construct a 4D cost volume using DNN features and apply improved semi-global matching [17] for cost aggregations. This strategy is impractical for end-to-end training of DNNs, since the cost aggregation step is not differentiable, and incurs huge memory and computational costs. The current state-of-the-art optical flow model, RAFT [54] also builds multi-scale 4D correlation volumes for all pairs of pixels. However, limited by its huge memory and computational costs, RAFT does not apply any cost aggregation to the 4D volume.

2.3. Cost Volumes in Stereo Matching

Full-range cost volumes built over the whole displacement space have been widely used in state-of-the-art stereo matching DNNs [9, 12, 14, 32, 67, 68]. Matching cost aggregation in cost volumes has also become a critical component in stereo matching [32, 67], since local, feature-based matching is often ambiguous due to occlusions, repetitive or homogeneous texture, reflections, noise *etc*. Based on the full-range cost volume, several cost aggregation approaches have been developed, such as geometry and context networks [32], and pyramid matching networks [9] that use 3D convolutions with a pyramidal encoder-decoder for cost volume learning, and guided aggregation networks [67] that use non-local, semi-global matching layers for non-local cost aggregations. Our Separable Flow motion representation makes it possible to use these effective local and non-local matching cost aggregation layers to learn a better cost volume for optical flow estimation.

3. Method

This section first describes the prototypical optical flow framework to which our Separable Flow module can be applied, then details the module itself, and finally presents the method used to train it.

3.1. Prototypical cost-volume-based optical flow

Cost volume based optical flow methods [53, 54] usually consist of the following stages: 1) image feature extraction, 2) cost volume computation and 3) motion refinement. Our work addresses stage two by introducing the separable cost volume and the cost aggregation modules. We briefly describe the common blocks in existing approaches, but refer the reader to prior works [53, 54] for the full details.

Image feature extraction. A convolutional network (*e.g.* ResNet [54]) is trained to extract per-pixel, local features from an image, and produces a feature tensor, $F \in \mathbb{R}^{H\times W\times D}$, where $F(i,j)$ is the D-dimensional feature of the pixel at location i,j.

Cost volume computation. Given the feature tensors F_1 and F_2 of the two optical flow images, a cost volume, $\mathbf{C} \in \mathbb{R}^{H\times W\times|U|\times|V|}$ is computed, where $U = \{u_{min},..,0,..,u_{max}\}$ and $V = \{v_{min},..,0,..,v_{max}\}$ are the sets of discrete horizontal and vertical motions considered for each pixel. Each entry in the 4D volume is typically [53, 54] computed for pixel i,j and pixel motion u,v via a dot product of feature vectors, thus:

$$C(i,j,u,v) = F_1(i,j)\cdot F_2(i+u,j+v) \tag{1}$$

Using this approach, higher "costs" represent greater similarity. Our work proposes a new way to represent and compute this cost volume, as described in section 3.2.

Motion refinement. Motion is estimated through iterative updates, usually in a coarse-to-fine framework [53–55]. The update layers take as input the current motion estimate, the cost volume, and context features, and output an additive motion update. Motion is usually initialized to zero. This work uses regression (sec. 3.2.3) to better initialize motion.

3.2. Separable Flow

We propose to replace the purely correlation-based cost volume of previous optical flow methods with an efficient, separable cost volume. Our Separable Flow module consists of the following three stages, functionally described below: self-adaptive cost separation, non-local cost aggregation, and motion regression. Fig. 2 provides a high level schematic of the design, while parameter and layer settings of the whole architecture can be found in the supplementary material.

3.2.1 Self-adaptive Cost Separation

In order to improve memory and computational efficiency, and enable non-local aggregation in a learned cost volume, we separate and compress the 4D cost volume, $\mathbf{C}$, into two 3D, K-dimensional feature tensors, $\mathbf{C}_u \in \mathbb{R}^{H\times W\times|U|\times K}$ and $\mathbf{C}_v \in \mathbb{R}^{H\times W\times|V|\times K}$, where $K \ll |U|,|V|$, representing horizontal and vertical motion respectively.

The first two channels (indexed by superscripts) of $\mathbf{C}_u$ are computed as:

$$C_u^1(i,j,u) = \frac{1}{|V|}\sum_{v\in V} C(i,j,u,v), \tag{2}$$

$$C_u^2(i,j,u) = \max_{v\in V} C(i,j,u,v). \tag{3}$$

Since mean and maximum select predetermined values of the cost volume, we propose to learn an adaptive selection for the remaining $K-2$ channels, with an attention module. Using the first two channels of the compressed $\mathbf{C}_v$ for efficiency, this self-adaptive compression is realized by

$$\mathbf{A_u} = \phi_u(\mathbf{C}_v^{1:2}), \qquad \in \mathbb{R}^{H\times W\times|V|\times K-2} \tag{4}$$

$$C_u^{k+2}(i,j,u) = \sigma\left(A_u^k(i,j)\right)\cdot C(i,j,u,:), \tag{5}$$

where ϕ_u is a single, 3D convolutional layer, and $\sigma(\cdot)$ represents the softmax operation. Note that $\mathbf{C}_u$ can be computed without storing the intermediate 4D cost volume $\mathbf{C}$. A similar approach is used to compute $\mathbf{C}_v$. Here we use $K = 4$.

This adaptive compression has several advantages over mean, maximum or convolutional compression. Convolutions, for example, require a fixed range of $|U|$ and $|V|$, while our method can handle variable search spaces. More importantly, convolutions are translationally invariant, but motion varies spatially. Our attention module outputs translationally varying weights, allowing it to adapt to different motions, learning better cost volume representations.

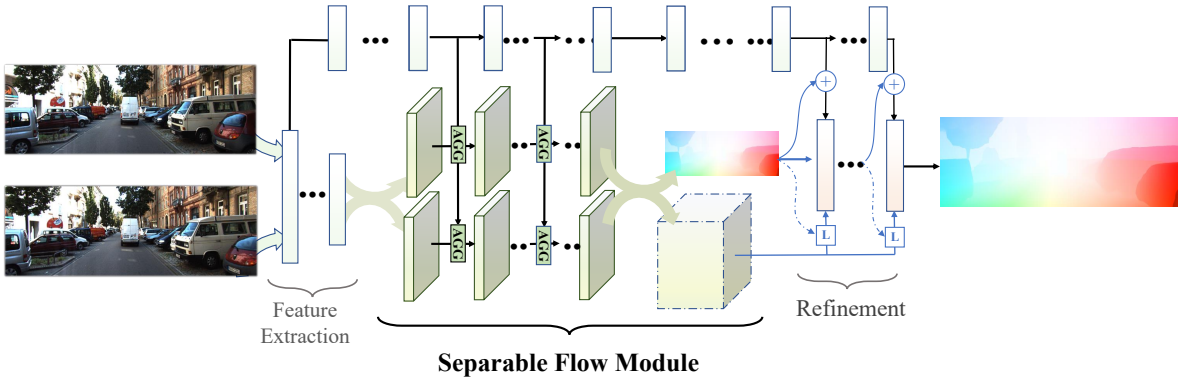


Figure 2: Architecture overview. Our model consists of three main parts: 1) feature extraction network, 2) our Separable Flow module of cost volume separation and aggregation networks, and 3) refinement modules. The top layers are a context network [54] that learns weights and context information for cost aggregations, refinement and upsampling (some models do without this). Our Separable Flow module separates the 4D motion cost volume generated from the features into two independent 3D displacement cost volumes. These volumes go through several non-local aggregation layers, as shown. The refined volumes, plus an initial flow estimate regressed from them, are input into the refinement network for further coarse-to-fine improvement and interpolation.

3.2.2 Learning cost aggregation

Semi-global matching aggregates non-local information in traditional stereo [17], and more recently optical flow [60], methods. Similarly effective aggregation layers have since been applied to neural networks for stereo matching [11, 32, 67], to great effect, but have not yet been shown to be practical for optical flow networks. However, our separable framework enables us to apply these aggregation layers directly to separated 2D motion.

Our cost aggregation module uses an encoder-decoder architecture that consists of four non-local, semi-global aggregation (SGA) layers, proposed in GANet [67], and eight 3D convolutional layers, to refine $\mathbf{C}_u$ from a $H \times W \times |U| \times K$ feature tensor to a $H \times W \times |U|$ cost volume, $\mathbf{C}_u^A$. A similar network is trained to compute $\mathbf{C}_v^A$.

3.2.3 Motion regression

Disparity regression has been used in stereo matching [32], where it is shown to be more robust than classification-based methods, and can generate sub-pixel accuracy. Furthermore, regression has been used to learn stereo cost volumes that are rich in geometry and contextual information [32, 67]. It is computed as the sum of each disparity, weighted by its probability, computed via a softmax over the cost volume.

We use a similar approach here to learn optical flow regression, $\mathbf{f}_0 = \{\hat{\mathbf{u}}, \hat{\mathbf{v}}\}$, as follows for each pixel i, j, prior to motion refinement:

$$\hat{u}(i,j) = U \cdot \sigma\left(C_u^A(i,j,:)\right), \tag{6}$$

$$\hat{v}(i,j) = V \cdot \sigma\left(C_v^A(i,j,:)\right). \tag{7}$$

Then, the initial flow prediction $\mathbf{f}_0$ and the learned cost volumes, C_u^A, C_v^A, are sent to the refinement module to compute a final motion prediction. Where motion refinement previously used correlation cost $C(i,j,u,v)$, it is instead fed concatenated, aggregated costs $[C_u^A(i,j,u), C_v^A(i,j,v)]$.

This motion regression learns a lower-resolution (*e.g.* 1/8, as used in RAFT [54]), but high-quality motion prediction that serves as a better input to the refinement module, considering that previous methods initialize with zero motion [53, 54]. As our ablation study shows (section 4.3), initializing motion with this regressed estimate is key to improving the prediction quality. It is worth noting that a standard (*i.e.* non-separated) 2D motion regression is naturally separable:

$$\mathbf{C}' = \sigma(C(i,j,:,:)), \tag{8}$$

$$\hat{u}(i,j) = \sum_u \sum_v U(u)\, C'(u,v), \tag{9}$$

$$= U \cdot \sum_v C'(:,v), \tag{10}$$

such that the $\sigma\left(C_u^A(i,j,:)\right)$ plays a similar role to $\sum_v C'(:,v)$. Given its efficacy in the stereo domain [32, 67], and the separable nature of 2D motion regression, this gives us some intuition into why motion regression can be used to effectively learn two separable 3D cost volumes that are also rich in prior contextual and geometry information.

3.3. Loss Function

Following RAFT [54], we use the L_1 loss between the predicted and ground truth flow for a sequence of N refinement predictions of optical flow, $\{\mathbf{f}_1, ..., \mathbf{f}_N\}$. However, in addition we also have the motion regressed flow, $\mathbf{f}_0$. Given

Training Data	Method	Sintel (train)		KITTI-15 (train)		Sintel (test)		KITTI-15 (test)
		Clean	Final	Epe-all	Fl-all	Clean	Final	Fl-all
-	FlowFields [2]	-	-	-	-	3.75	5.81	15.31
-	FlowFields++ [49]	-	-	-	-	2.94	5.49	14.82
S	DCFlow [60]	-	-	-	-	3.54	5.12	14.86
S	MRFlow [58]	-	-	-	-	2.53	5.38	12.19
C + T	HD3 [65]	3.84	8.77	13.17	24.0	-	-	-
	PWC-Net [53]	2.55	3.93	10.35	33.7	-	-	-
	LiteFlowNet2 [25]	2.24	3.78	8.97	25.9	-	-	-
	VCN [62]	2.21	3.68	8.36	25.1	-	-	-
	MaskFlowNet [70]	2.25	3.61	-	23.1	-	-	-
	FlowNet2 [28]	2.02	3.54	10.08	30.0	3.96	6.02	-
	DICL-Flow [55]	1.94	3.77	8.70	23.6		-	-
	RAFT [54]	1.43	2.71	5.04	17.4	-	-	-
	Ours	**1.30**	**2.59**	**4.60**	**15.9**	-	-	-
C + T + V	RAFT [54]	1.45	2.75	3.20	9.13	-	-	-
	Ours	**1.32**	**2.61**	**2.60**	**7.74**	-	-	**7.92**
C+T+S/K	FlowNet2 [28]	(1.45)	(2.01)	(2.30)	(6.8)	4.16	5.74	11.48
	PWC-Net [53]	-	-	-	-	4.39	5.04	9.60
	LiteFlowNet [24]	(1.35)	(1.78)	(1.62)	(5.58)	4.54	5.38	9.38
	HD3 [65]	(1.87)	(1.17)	(1.31)	(4.1)	4.79	4.67	6.55
	ScopeFlow [4]	-	-	-	-	3.59	4.10	6.82
	DICL-Flow [55]	(1.11)	(1.60)	(1.02)	(3.60)	2.12	3.44	6.31
	VCN+LCV [59]	(1.62)	(2.22)	(1.13)	(3.80)	2.83	4.20	6.25
	RAFT+LCV [59]	(0.94)	(1.31)	(1.06)	(3.77)	2.75	3.55	6.26
	RAFT [54]	(0.77)	(1.20)	(0.64)	(1.5)	2.08	3.41	5.27
	Ours	(0.71)	(1.14)	(0.68)	(1.57)	**1.99**	**3.27**	**4.89**
C+T+S+K+H	PWC-Net+ [52]	(1.71)	(2.34)	(1.50)	(5.3)	3.45	4.60	7.72
	VCN [62]	(1.66)	(2.24)	(1.16)	(4.1)	2.81	4.40	6.30
	MaskFlowNet [70]	-	-	-	-	2.52	4.17	6.10
	RAFT (2-view)	(0.76)	(1.22)	(0.63)	(1.5)	1.94	3.18	5.10
	RAFT (warm-start)	(0.77)	(1.27)	-	-	1.61	2.86	-
	Ours	(0.69)	(1.10)	(0.69)	(1.60)	**1.50**	**2.67**	**4.64**

Table 1: Results on Sintel and KITTI datasets. C+T: We test the generalization performance on KITTI (train) after training on FlyingChairs (C) and FlyingThings (T). C+T+V: We also provide extra synthetic driving scenes from Virtual KITTI (V) [8] to further boost the generalization on real driving scenes. Our method outperform existing methods for synthetic to real generalization. We also evaluate our model on public benchmarks after finetuning. C+T+S/K includes methods which finetune only on Sintel data when evaluating on Sintel, or only KITTI data when evaluating on KITTI. C+T+S+K+H includes methods that combine KITTI, HD1K, and Sintel data when finetuning. Separable Flow outperforms previous state-of-the-art approaches, ranking 1st among *all* published optical flow approaches on both Sintel (clean and final passes) and KITTI 2015 optical flow benchmarks.

ground truth flow $\mathbf{f}_{gt}$, our loss is thus defined as

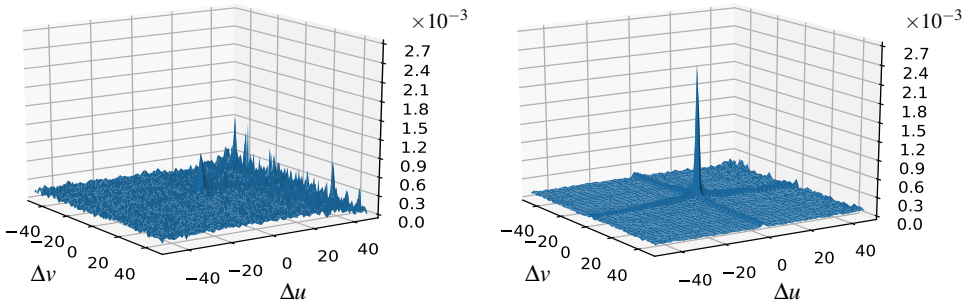
$$\mathscr{L} = \sum_{i=0}^{N} \lambda^{N-i} \, ||\mathbf{f}_{gt} - \mathbf{f}_i||_1 \tag{11}$$

where $\lambda = 0.8$ in our experiments, weighting later refinement steps higher to ensure convergence.

4. Experiments

This section details the experiments and results that demonstrate our Separable Flow module is the new state of the art in accuracy for optical flow. It also demonstrates its improved cross-domain generalization, as well as the specific categories of error that our model fixes, with a discussion on why. An ablation study rounds off the evaluation.

Implementation Details: Our model is implemented in



(a) RAFT (b) Ours

Figure 3: Comparisons of cost volumes. $(\Delta u, \Delta v)$ are the shifts from ground truth flow (origin of coordinates). The average normalized cost volumes over all pixels in occlusion regions across 100 image pairs are visualized. The center value (at the ground truth displacement) is 4.2× higher for our method vs. RAFT.

PyTorch [44] and we follow the training setting of RAFT [54]. Unless otherwise stated (*e.g.* sec. 4.3), we use the feature extraction and refinement modules from RAFT [54].

Following RAFT [54], we train our network on FlyingChairs [13] for 100k iterations (with batch size 12), then FlyingThings [41] for 100k iterations (batch size 6), and finally finetune on a combination of data from FlyingThings [41], Sintel [7], KITTI-2015 [42], and HD1K [33], for another 100k iterations (batch size 6). All other learning settings (including data augmentation) are the same as those in RAFT [54].

4.1. Quantitative Evaluation

We evaluate our Separable Flow model on the now standard, online, Sintel [7] and KITTI [16] benchmarks. We evaluate two models on each benchmark. The first is finetuned on the training set of the specific benchmark (*i.e.* Sintel or KITTI). The second is finetuned on the combined training set described above. Results are presented in the bottom two sections of Table 1 respectively. When compared with other methods trained on the same data, our method is leading in both the epe (end-point-error) and the Fl-all (threshold error rates) evaluations. On both benchmarks, best results for our method are achieved using the mixed training set. On Sintel, the average end point errors (EPE) of 1.50 (clean) and 2.67 (final) are both reductions of 7% over the previous best result, from RAFT [54]. On KITTI, the 4.64% error rate is a 9% reduction over the previous best result, also achieved by RAFT.

4.1.1 Cross-domain Generalization

Since collecting ground truth for real data is costly, generalization abilities are particularly important in real application scenarios. We test the cross-domain generalization performance of our model on Sintel (train) and KITTI (train) after training on synthetic FlyingChairs (C) and FlyingThings (T), with results shown in Table 1, second section. Our model again outperforms all existing published methods. Moreover, on real KITTI evaluations, our model achieves an error rate of 15.9%, which is far better than most existing models, and a 9% reduction over the previous best (once again, RAFT [54]).

In addition, we use extra synthetic driving scenes [8] to boost the generalization from synthetic scenes to a real driving dataset. By training only with these synthetic data (FlyingChairs, FlyingThings and Virtual KITTI2 [8]), our model achieves an error rate of 7.60% (Table 1, third section) on the real KITTI training set, and 7.92% on the KITTI test set. Several DNNs (*e.g.* PWC-Net [53], FlowNet2 [28] and LitleFlowNet [24]) perform worse than this, even when finetuned on the target KITTI training set.

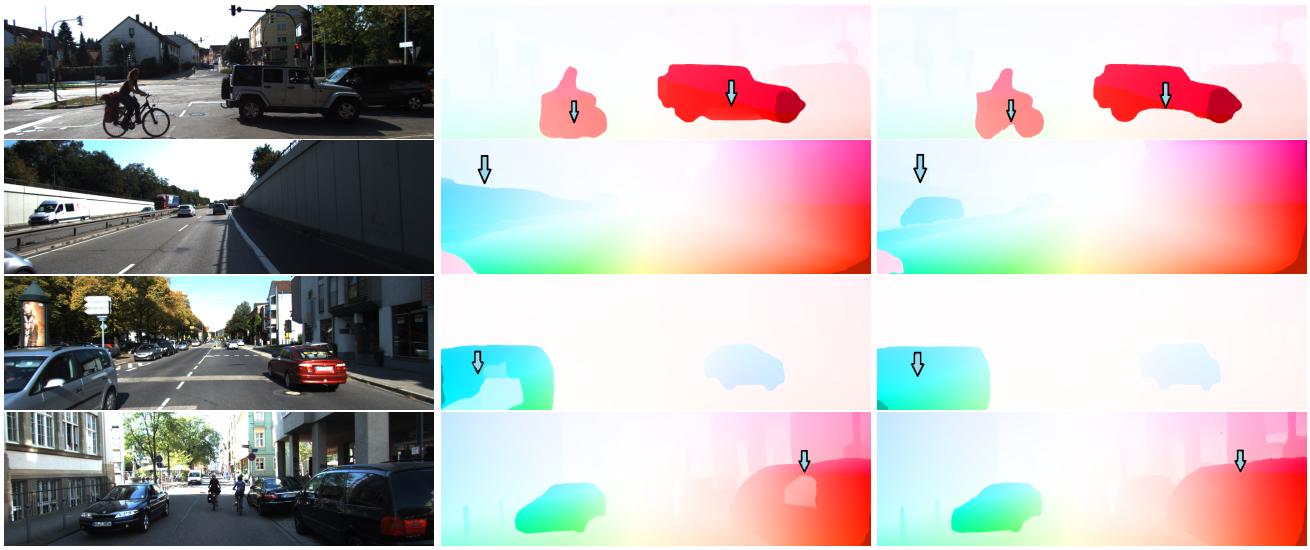
We thus find that Separable Flow provides even greater performance gains when applied to cross-domain scenarios. We attribute these generalization abilities to our separable non-local aggregations, which capture more robust, non-local geometry and contextual information, instead of local, domain-sensitive features. *Visualized results and comparisons are shown in the supplementary materials.*

4.2. Qualitative Analysis

Separable Flow produces a clear quantitative improvement in accuracy. In this section we seek to explain qualitatively where these improvements arise, and why.

Fig. 3 visualizes the averaged & normalized Separable Flow cost volume (b) for a challenging occlusion regions and those of RAFT [54]. It can be seen that our cost volume offers a single, large peak at the ground truth motion, in contrast to the RAFT which has many noisy, false peaks in its cost volume. A similar effect can be seen in the reflection region (*available in the supplementary materials*). This demonstrates that our learned cost volume is able to overcome regional ambiguities, by exploiting global geometry and contextual information.

Fig. 4 compares optical flow outputs from our model with those of RAFT [54]. In challenging regions such as



(a) Input views (b) RAFT [54] (c) Ours

Figure 4: Qualitative comparisons. (b) Results of the state-of-the-art RAFT [54]. (c) Results of our Separable Flow. Significant improvements are highlighted by arrows. The cost aggregations can effectively aggregate motion information to large, textureless regions (*e.g.* white wall behind the car), and reflective regions (*e.g.* car windows), give precise estimates. By learning contextual object information, it also preserves object boundaries very well (top row).

large textureless areas (*e.g.* the white wall behind the car), and reflection areas (*e.g.* the car windows), the matching information is usually ambiguous, and thus leads to wrong matches in RAFT [54]. The non-local aggregations in our Separable Flow allow it to recognise and capture long-range contextual information, generating more accurate motion estimates in these regions. This rich contextual information also preserves object boundaries very well (top row).

4.3. Ablation Study

We perform a set of ablation experiments to validate the need for, and show the relative importance of, each of the components of the Separable Flow module that we propose. All ablation models are trained on FlyingChairs (C) + FlyingThings (T) and evaluated on the Sintel and KITTI training set.

Componentwise ablations: Results of componentwise ablations are shown in Table 2. In each section of the table, we test a specific component of our approach in isolation, with the settings used in our final model underlined.

Separation Channels: the attention layers of our self-adaptive cost separation provide a significant boost over just mean or max aggregation. *Aggregation layers* all improve performance, with SGA layers [67] providing the most benefit, highlighting the need for non-local aggregation. *Shared Agg. Weights*: cost aggregation networks for computing $\mathbf{C}_u^A$ and $\mathbf{C}_v^A$ can either share weights, or learn separate weights. The latter generates a reasonable advantage, due to the rotational variance of natural scenes. *Aggregation Blocks*: The hourglass blocks used in the [9, 67] are too resource heavy here. Instead, we tested using UNet and ResNet blocks, with the former providing better performance. *Motion Regression* substantially increases performance when used to initialize the motion refinement block, without increasing network bandwidth. This suggests it helps the network to learn *better*, rather than more.

These experiments validate the importance of each of the contributions of this work.

Different frameworks: Table 3 shows the performance gains using Separable Flow in different frameworks, and vice versa. We apply Separable Flow to two popular optical flow frameworks [53, 54], which differ in their refinement modules. Both frameworks are significantly improved, PWC-Net [53] even more so than RAFT [54], with reduction in errors ranging between 11-31% (compared to the latter's already reported 7%). Given our separated motion cost volumes, we are also able to use many different stereo matching backbones to process these volumes independently, and predict the motion directly. We test both PSMNet [9] and GANet [67]. Even without coarse-to-fine optical flow refinement modules, these models can still estimate motions more accurately than some popular optical flow models (*e.g.* PWC-Net [53]), demonstrating the flexibility of a separable motion cost volume representation.

4.4. Timing, Parameter and Accuracy

In Table 4, we compare the parameter counts, inference time, and training iterations for our method versus several

Experiment	Variations	Sintel (train) Clean	Sintel (train) Final	KITTI-15 (train) Epe-all	KITTI-15 (train) Fl-all	Parameters
Baseline [54]	–	1.43	2.71	5.04	17.4	5.3M
Separation Channels	Mean	1.39	2.65	4.80	16.7	5.9M
	Max	1.38	2.65	4.74	16.5	5.9M
	Attention	1.32	2.62	4.72	16.2	6.0M
	All	1.30	2.59	4.60	15.9	6.0M
Aggregation Layers	2× 3D conv	1.39	2.68	4.91	16.8	5.9M
	8× 3D conv	1.33	2.63	4.75	16.4	6.2M
	2× SGA	1.34	2.64	4.71	16.2	6.0M
	4× SGA	1.30	2.59	4.60	15.9	6.0M
Shared Agg. Weights	No	1.30	2.59	4.60	15.9	6.0M
	Yes	1.34	2.65	4.72	16.3	5.7M
Aggregation Block	ResNet	1.33	2.63	4.74	16.1	6.0M
	UNet	1.30	2.59	4.60	15.9	6.0M
Motion Regression*	No	1.37	2.65	4.89	16.8	6.0M
	Yes	1.30	2.59	4.60	15.9	6.0M

Table 2: Ablation experiments. Settings used in our final model are underlined. See Sec. 4.3 for details.

Cost Agg. module	Refinement module	Sintel (train) final	Sintel (train) clean	KITTI (train) Fl-all (%)
–	PWC-Net [53]	2.55	3.93	33.7
Ours	PWC-Net [53]	1.89	3.51	23.1
–	RAFT [54]	1.43	2.71	17.4
Ours	RAFT [54]	1.30	2.59	15.9
Ours+PSMNet [9]	–	3.21	4.32	32.8
Ours+GANet [67]	–	2.49	3.81	28.1

Table 3: Performance using different refinement and aggregation modules. Models are trained on FlyingChairs and FlyingThings datasets and evaluated on Sintel and KITTI training sets.

Method	Param	Speed	Iterations	KITTI Fl-all (%)
FlowNet2 [28]	162.5M	0.1s	7100K	11.48
VCN [62]	6.2M	0.18s	300k	6.30
VCN+LCV [59]	6.3M	0.26s	–	6.25
RAFT [54]	5.3M	0.2s	350k	5.10
Ours	6.0M	0.25s	350k	4.64

Table 4: Comparisons of parameter counts, inference time, and training iterations *vs*. accuracy, of our model against recent cost-volume-based optical flow networks [54, 59, 62]. Speed measurements are from the KITTI2015 benchmark.

recent cost-volume-based optical flow networks [54,59,62]. Separable Flow has a similar number of parameters and running speed as another cost-volume-based method [59], but achieves 24% lower error rates. Compared with state-of-the-art RAFT [54], our Separable Flow introduces about 0.7M new parameters, and is slightly slower. The main benefit of our method is therefore its improved accuracy.

5. Conclusion

We have introduced the Separable Flow module, a cost-volume computation module for optical flow inference that is able to exploit non-local cost aggregation through the use of a separable cost volume representation, and motion regression. Our experimental results, which beat the previous state of the art in accuracy with a consistent 7% reduction in error, demonstrate that this module both resolves ambiguities in occluded, textureless and other such regions, through the use of non-local, contextual information and prior knowledge, and also improves cross-domain generalization when appyling a synthetically trained network to real data. Our ablation study validates the importance of each of the blocks that make up our Separable Flow module. We note that this module can benefit a broad class of optical flow methods, based on cost volumes.

Our model fails in just a few cases, where objects (*e.g.* cars) move in occluded regions.This is a common limitation of optical flow approaches: when a moving object is visible in only one image, networks predict the object to be stationary since this is the most plausible motion (for cars in KITTI, at least). To address this issue, multi-view or video inputs can be employed.

Acknowledgements This work was supported by Snap Inc., Turing AI Fellowship: EP/W002981/1, EPSRC/MURI grant EP/N019474/1. We would also like to thank the Royal Academy of Engineering and FiveAI.

References

[1] Min Bai, Wenjie Luo, Kaustav Kundu, and Raquel Urtasun. Exploiting semantic information and deep matching for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 154–170. Springer, 2016. 2

[2] Christian Bailer, Bertram Taetz, and Didier Stricker. Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 4015–4023, 2015. 2, 5

[3] Christian Bailer, Kiran Varanasi, and Didier Stricker. Cnn-based patch matching for optical flow with thresholded hinge embedding loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3250–3259, 2017. 2

[4] Aviram Bar-Haim and Lior Wolf. Scopeflow: Dynamic scene scoping for optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7998–8007, 2020. 2, 5

[5] Michael J Black and Padmanabhan Anandan. A framework for the robust estimation of optical flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 231–236. IEEE, 1993. 1, 2

[6] Thomas Brox, Christoph Bregler, and Jitendra Malik. Large displacement optical flow. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 41–48. IEEE, 2009. 2

[7] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 611–625. Springer, 2012. 2, 6

[8] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. 5, 6

[9] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 3, 7, 8

[10] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4706–4714, 2016. 2

[11] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of the IEEE international conference on computer vision*, pages 686–695, 2017. 2, 4

[12] Xinjing Cheng, Peng Wang, and Ruigang Yang. Learning depth with convolutional spatial propagation network. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2361–2379, 2019. 3

[13] 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. 6

[14] Xianzhi Du, Mostafa El-Khamy, and Jungwon Lee. Amnet: Deep atrous multiscale stereo disparity estimation networks. *arXiv preprint arXiv:1904.09099*, 2019. 3

[15] David Gadot and Lior Wolf. Patchbatch: A batch augmented loss for optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4236–4245, 2016. 2

[16] 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. 2, 6

[17] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007. 2, 4

[18] Markus Hofinger, Samuel Rota Bulò, Lorenzo Porzi, Arno Knapitsch, and Peter Kontschieder. Improving optical flow on a pyramidal level. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[19] Berthold KP Horn and Brian G Schunck. Determining optical flow. In *Techniques and Applications of Image Understanding*, volume 281, pages 319–331. International Society for Optics and Photonics, 1981. 1, 2

[20] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, 2012. 2

[21] Yinlin Hu, Yunsong Li, and Rui Song. Robust interpolation of correspondences for large displacement optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 481–489, 2017. 2

[22] Yinlin Hu, Rui Song, and Yunsong Li. Efficient coarse-to-fine patchmatch for large displacement optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5704–5712, 2016. 2

[23] Tak-Wai Hui and Chen Change Loy. Liteflownet3: Resolving correspondence ambiguity for more accurate optical flow estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 169–184. Springer, 2020. 2

[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, 5, 6

[25] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. A lightweight optical flow cnn–revisiting data fidelity and regularization. *arXiv preprint arXiv:1903.07414*, 2019. 2, 5

[26] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5754–5763, 2019. 2

[27] Eddy Ilg, Ozgun Cicek, Silvio Galesso, Aaron Klein, Osama Makansi, Frank Hutter, and Thomas Brox. Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 652–667, 2018. 2

[28] 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 andF pattern recognition*, pages 2462–2470, 2017. 2, 5, 6, 8

[29] Woobin Im, Tae-Kyun Kim, and Sung-Eui Yoon. Unsupervised learning of optical flow with deep feature similarity. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–188. Springer, 2020. 2

[30] Joel Janai, Fatma Guney, Anurag Ranjan, Michael Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 690–706, 2018. 2

[31] Rico Jonschkowski, Austin Stone, Jonathan T Barron, Ariel Gordon, Kurt Konolige, and Anelia Angelova. What matters in unsupervised optical flow. *arXiv preprint arXiv:2006.04902*, 1(2):3, 2020. 2

[32] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017. 3, 4

[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. 6

[34] Till Kroeger, Radu Timofte, Dengxin Dai, and Luc Van Gool. Fast optical flow using dense inverse search. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 471–488. Springer, 2016. 2

[35] Wei-Sheng Lai, Jia-Bin Huang, and Ming-Hsuan Yang. Semi-supervised learning for optical flow with generative adversarial networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 353–363, 2017. 2

[36] Liang Liu, Jiangning Zhang, Ruifei He, Yong Liu, Yabiao Wang, Ying Tai, Donghao Luo, Chengjie Wang, Jilin Li, and Feiyue Huang. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6489–6498, 2020. 2

[37] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4571–4580, 2019. 2

[38] Yao Lu, Jack Valmadre, Heng Wang, Juho Kannala, Mehrtash Harandi, and Philip Torr. Devon: Deformable volume network for learning optical flow. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2705–2713, 2020. 2

[39] Josef Maier, Martin Humenberger, Markus Murschitz, Oliver Zendel, and Markus Vincze. Guided matching based on statistical optical flow for fast and robust correspondence analysis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 101–117. Springer, 2016. 2

[40] Daniel Maurer, Nico Marniok, Bastian Goldluecke, and Andrés Bruhn. Structure-from-motion-aware patchmatch for adaptive optical flow estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 565–581, 2018. 2

[41] 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. 6

[42] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3061–3070, 2015. 6

[43] Moritz Menze, Christian Heipke, and Andreas Geiger. Discrete optimization for optical flow. In *German Conference on Pattern Recognition*, pages 16–28. Springer, 2015. 2

[44] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 6

[45] René Ranftl, Kristian Bredies, and Thomas Pock. Non-local total generalized variation for optical flow estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 439–454. Springer, 2014. 2

[46] 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

[47] Sathya N Ravi, Yunyang Xiong, Lopamudra Mukherjee, and Vikas Singh. Filter flow made practical: Massively parallel and lock-free. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3549–3558, 2017. 2

[48] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1164–1172, 2015. 2

[49] René Schuster, Christian Bailer, Oliver Wasenmüller, and Didier Stricker. Flowfields++: Accurate optical flow correspondences meet robust interpolation. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 1463–1467. IEEE, 2018. 2, 5

[50] Tal Schuster, Lior Wolf, and David Gadot. Optical flow requires multiple strategies (but only one network). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4950–4959, 2017. 2

[51] Laura Sevilla-Lara, Deqing Sun, Varun Jampani, and Michael J Black. Optical flow with semantic segmentation and localized layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3889–3898, 2016. 2

[52] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *arXiv preprint arXiv:1809.05571*, 2018. 5

[53] 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. 1, 2, 3, 4, 5, 6, 7, 8

[54] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 402–419. Springer, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[55] Jianyuan Wang, Yiran Zhong, Yuchao Dai, Kaihao Zhang, Pan Ji, and Hongdong Li. Displacement-invariant matching cost learning for accurate optical flow estimation. *arXiv preprint arXiv:2010.14851*, 2020. 2, 3, 5

[56] Yang Wang, Yi Yang, Zhenheng Yang, Liang Zhao, Peng Wang, and Wei Xu. Occlusion aware unsupervised learning of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4884–4893, 2018. 2

[57] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013. 2

[58] Jonas Wulff, Laura Sevilla-Lara, and Michael J Black. Optical flow in mostly rigid scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4671–4680, 2017. 2, 5

[59] Taihong Xiao, Jinwei Yuan, Deqing Sun, Qifei Wang, Xin-Yu Zhang, Kehan Xu, and Ming-Hsuan Yang. Learnable cost volume using the Cayley representation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–499. Springer, 2020. 2, 5, 8

[60] Jia Xu, René Ranftl, and Vladlen Koltun. Accurate optical flow via direct cost volume processing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1289–1297, 2017. 2, 4, 5

[61] 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

[62] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. In *Advances in Neural Information Processing Systems*, pages 793–803, 2019. 1, 2, 5, 8

[63] Yanchao Yang and Stefano Soatto. S2f: Slow-to-fast interpolator flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2087–2096, 2017. 2

[64] Yanchao Yang and Stefano Soatto. Conditional prior networks for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 271–287, 2018. 2

[65] Zhichao Yin, Trevor Darrell, and Fisher Yu. Hierarchical discrete distribution decomposition for match density estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6044–6053, 2019. 2, 5

[66] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint pattern recognition symposium*, pages 214–223. Springer, 2007. 2

[67] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 185–194, 2019. 2, 3, 4, 7, 8

[68] Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant stereo matching networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[69] Feihu Zhang and Benjamin W Wah. Fundamental principles on learning new features for effective dense matching. *IEEE Transactions on Image Processing*, 27(2):822–836, 2017. 2

[70] 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. 2, 5

[71] 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

[72] Yiran Zhong, Pan Ji, Jianyuan Wang, Yuchao Dai, and Hongdong Li. Unsupervised deep epipolar flow for stationary or dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12095–12104, 2019. 2

[73] Shay Zweig and Lior Wolf. Interponet, a brain inspired neural network for optical flow dense interpolation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4563–4572, 2017. 2