Learning to Estimate Hidden Motions with Global Motion Aggregation

Shihao Jiang[1,2] Dylan Campbell[3] Yao Lu[1,2] Hongdong Li[1,2] Richard Hartley[1,2]

[1]Australian National University [2]ACRV [3]University of Oxford

Figure 1. **Global motion aggregation helps resolve ambiguities caused by occlusions.** Occlusions—a term we extend to include any parts of a scene that disappear in the next frame—cause large ambiguities in the optical flow estimation problem that cannot be resolved by local approaches. Based on the assumption that points on an object have homogeneous motions, which often holds approximately, we propose to globally aggregate motion features of pixels that are likely to belong to the same object. In this example, most pixels on the blade move out-of-frame from frame 2 to frame 3. When only these two frames are provided, global aggregation allows motion information to be passed from non-occluded pixels to occluded pixels, which helps resolve ambiguities caused by occlusions.

Abstract

Occlusions pose a significant challenge to optical flow algorithms that rely on local evidences. We consider an occluded point to be one that is imaged in the reference frame but not in the next, a slight overloading of the standard definition since it also includes points that move out-of-frame. Estimating the motion of these points is extremely difficult, particularly in the two-frame setting. Previous work relies on CNNs to learn occlusions, without much success, or requires multiple frames to reason about occlusions using temporal smoothness. In this paper, we argue that the occlusion problem can be better solved in the two-frame case by modelling image self-similarities. We introduce a global motion aggregation module, a transformer-based approach to find long-range dependencies between pixels in the first image, and perform global aggregation on the corresponding motion features. We demonstrate that the optical flow estimates in the occluded regions can be significantly improved without damaging the performance in non-occluded regions. This approach obtains new state-of-the-art results on the challenging Sintel dataset, improving the average end-point error by 13.6% on Sintel Final and 13.7% on Sintel Clean. At the time of submission, our method ranks first on these benchmarks among all published and unpublished approaches. Code is available at `https://github.com/zacjiang/GMA`.

1. Introduction

How can we estimate the 2D motion of a point we only see once? This is the problem faced by optical flow algorithms for points that become occluded between frames. Estimating the optical flow, that is, the apparent motion of pixels in an image as the camera and scene move, is a classic problem in computer vision studied since the seminal work of Horn and Schunck [14]. There are many factors that make optical flow prediction a hard problem, including large motions, motion and defocus blur, and featureless regions. Among these challenges, occlusion is one of the most difficult and under-explored. In this paper, we propose an approach that specifically targets the occlusion problem in the case of two-frame optical flow prediction.

We first define what we mean by occlusion in the context of optical flow estimation. In this paper, an occluded point is defined as a 3D point that is imaged in the reference frame but is not visible in the matching frame. This definition incorporates several different scenarios, such as the query point moving out-of-frame or behind another object (or itself), or another object moving in front of the query point, in the active sense. One particular case of occlusion is shown in Figure 1, where part of the blade moves out-of-frame.

The challenge posed by occlusions can be understood by looking at the underlying assumptions of optical flow algorithms. Traditional optical flow algorithms apply the bright-

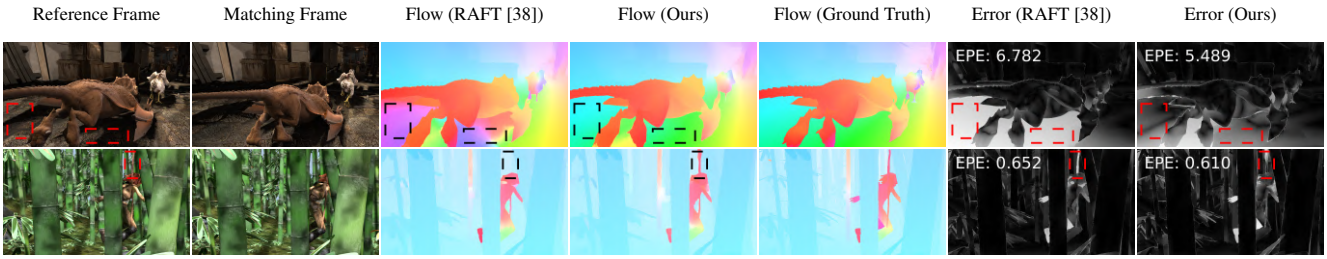

Figure 2. **Recovering hidden motions.** In row 1, the bottom left corner of the ground moves out-of-frame, but reasoning that it belongs to the background allows the motion to be recovered from other parts of the image. In row 2, the girl's staff is mostly occluded in the second frame, but strong cues from the visible parts can resolve its motion. Our approach can estimate many hidden motions despite the presence of occlusions. The flow maps and the error maps have been fetched from the Sintel server [8]. Best viewed in colour on a screen.

ness constancy constraint [14], where pixels related by the flow field are assumed to have the same intensities. It is clear that occlusions are a direct violation of such a constraint. In the deep learning era, correlation (cost) volumes [15] are used to give a matching cost for each potential displacement of a pixel. However, correlations of appearance features are unable to give meaningful guidance for learning the motion of occluded regions. Most existing approaches use smoothness terms in an MRF to interpolate occluded motions [9] or use CNNs to directly learn the neighbouring relationships, hoping to learn to estimate occluded motions based on the neighbouring pixels [38, 36]. However, state-of-the-art methods still fail to estimate occluded motions correctly when occlusions are more significant and local evidence is insufficient to resolve the ambiguity.

In contrast, humans are able to synthesise information from across the image and apply plausible motion models to accurately estimate occluded motions. This capability is valuable to emulate, because we fundamentally care about recovering the real 3D motion of objects in a scene, for which estimating occluded motion is necessary. Downstream applications, including tracking and activity detection [23], can also benefit from short-term predictions of the motion of occluded points, particularly if they reappear later or exhibit some characteristic of interest (*e.g.*, high-velocity out-of-frame motions).

Let us consider how to estimate these hidden motions for the two-frame case. When direct (local) matching information is absent, the motion information has to be propagated from other pixels. Using convolutions to propagate this information has the drawback of limited range since convolution is a local operation. We propose to aggregate the motion features with a non-local approach. Our design is based on the assumption that the motions of a single object (in the foreground or background) are often homogeneous. One source of information that is overlooked by existing works is self-similarities in the reference frame. For each pixel, understanding which other pixels are related to it, or which object it belongs to, is an important cue for accurate optical flow predictions. That is, the motion information of non-occluded self-similar points can be propagated to the occluded points. Inspired by the recent success of transformers [39], we introduce a global motion aggregation (GMA) module, where we first compute an attention matrix based on the self-similarities of the reference frame, then use that attention matrix to aggregate motion features. We use these globally aggregated motion features to augment the successful RAFT [38] framework and demonstrate new state-of-the-art results in optical flow estimation, such as those examples in Figure 2.

The key contributions of our paper are as follows. We show that long-range connections, implemented using the attention mechanism of transformer networks, are highly beneficial for optical flow estimation, particularly for resolving the motion of occluded pixels where local information is insufficient. We show that self-similarities in the reference frame provide an important cue for selecting the long-range connections to prioritise. We demonstrate that our global motion feature aggregation strategy leads to a significant improvement in optical flow accuracy in occluded regions, without damaging the performance in non-occluded regions, and analyse this extensively. We improve the average end-point error (EPE) by 13.6% (2.86 $\rightarrow$ 2.47) on Sintel Final and 13.7% (1.61 $\rightarrow$ 1.39) on Sintel Clean, compared to the strong baseline of RAFT [38]. Our approach ranks first on both datasets at the time of submission.

2. Related Work

Occlusions in optical flow. Occlusion poses a key challenge in optical flow estimation due to its violation of the brightness constancy constraint [14]. Most traditional optical flow algorithms treat occlusions as outliers and so develop and optimise robust objective functions. In continuous optimisation for optical flow, Brox *et al*. [6] used the L^1 norm due to its robustness to outliers caused by occlusions or large brightness variations. Zach *et al*. [46] added total variation regularisation and proposed an efficient numerical scheme to optimise the energy functional. This formulation

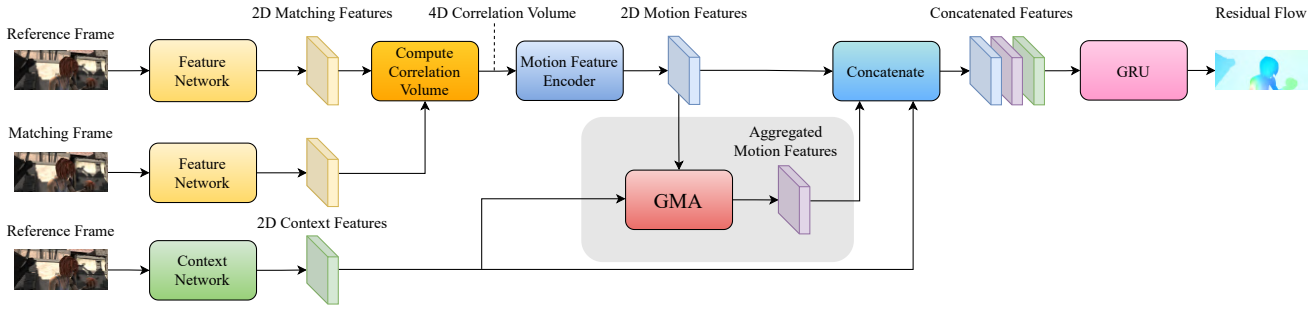

Figure 3. **Proposed architecture.** Our network is based on the successful RAFT [38] architecture. The proposed global motion aggregation (GMA) module is contained inside the shaded box, a self-contained addition to RAFT with low computational overhead that significantly improves performance. It takes the visual context features and the motion features as input and outputs aggregated motion features that share information across the image. These aggregated global motion features are then concatenated with the local motion features and the visual context features to be decoded by the GRU into residual flow. This gives the network the flexibility to choose between or combine the local and global motion features, depending on the needs of the specific pixel location. For example, a location with poor local image evidence, caused by occlusion for instance, could preference the global motion features.

was later improved by Wedel *et al*. [42]. Later work introduced additional robust optimisation terms, including the Charbonnier potential [7] and the Lorentzian potential [4].

More recently, discrete optimisation approaches, especially Markov Random Fields (MRFs) [5], have been used to estimate optical flow. These algorithms [27, 9, 43] first estimate the forward and backward flows separately using a robust, truncated data term. They then conduct a forward–backward consistency check to determine the occluded regions. Lastly, as a post-processing step, they use interpolation methods [31] to fill in the optical flow of the occluded regions.

Other work incorporates occlusion estimation as a joint objective together with optical flow estimation. Alvarez *et al*. [1] use forward–backward consistency as an optimisation objective, thus estimating time-symmetric optical flow. In addition to forward–backward consistency, MirrorFlow [18] incorporates occlusion–disocclusion symmetry in the energy function and achieves performance improvements. Since occlusions are caused by 3D motions, other works [35, 33] explicitly model local depth relationships into layers and reason about occlusions.

Contrary to the above approaches, we do not overload the loss function with explicit occlusion reasoning. Instead, we adopt a learning approach, similar to other supervised deep optical flow learning approaches [36, 45, 17, 19, 44, 2, 47, 38, 22]. Rather than estimating an occlusion map explicitly, our goal is to improve the optical flow accuracy at occluded regions. We take an implicit approach to globally aggregate motion features, which provides extra information to correctly predict flow at occluded regions. Our approach can be thought of as a non-local interpolation approach [34], in contrast to local interpolation approaches [31]. In the deep learning literature, the occlusion problem has been addressed in an unsupervised learning setting [41, 21, 25], however, existing supervised learning approaches all rely on convolutions to interpolate in occluded regions, which are prone to failure for more significant occlusions.

Self-attention and transformers. Our design principle is inspired by the recent successes of the transformer literature [39]. The transformer architecture was first successful in natural language processing (NLP), due to its ability to model long-range dependencies and its scalability for GPU parallel processing. Among various modules in the transformer architecture, self-attention is the key design feature that make transformers work. Recently, researchers have introduced the transformer and related attention ideas to the vision community, mostly in high-level tasks such as image classification [30, 10] and semantic segmentation [12, 40, 16]. To the best of our knowledge, we are the first to use the idea of attention to solve the optical flow problem. Different from many existing works in the transformer literature, we do not use self-attention in our work. Self-attention refers to the query, key and value vectors coming from the same features. In our case, query and key vectors come from the context features modelling the appearance of the image while value vectors come from the motion features, which is an encoding of the correlation volume.

3. Method

3.1. Background

We base our network design on the successful RAFT architecture [38]. Our overall network diagram is shown in Figure 3. For completeness, we briefly describe the main contributions of RAFT from which our model benefits. The first contribution is the introduction of an all-pairs correlation volume, which explicitly models matching correlations

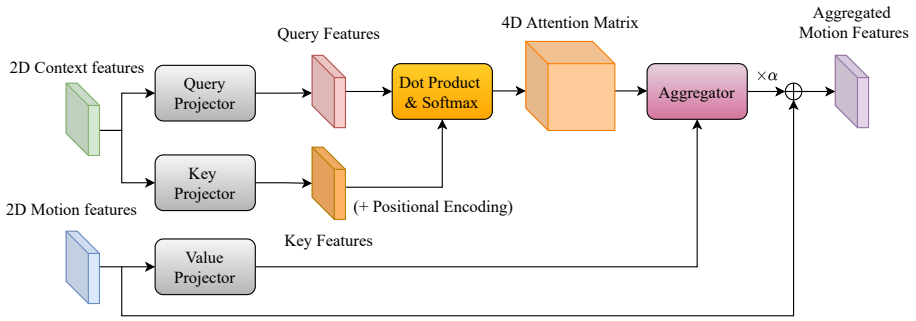

Figure 4. **Details of the GMA module.** To model the self-similarity of the first frame, we project the context feature map to a query feature map and a key feature map. We then take the dot product of the two feature maps and a softmax to obtain an attention matrix, which encodes self-similarity in appearance feature space. Similar to transformer networks [39], we also take the dot product between the query feature map and a set of positional embedding vectors which augments the attention matrix with positional information. Separately, the motion feature map encoded from the correlation volume is projected using the learned value projector. Its weighted sum, using the obtained attention matrix, produces the aggregated global motion features.

for all possible displacements. The benefit of using all-pairs correlations is its ability to handle large motions. The second major contribution is the use of a gated recurrent unit (GRU) decoder for iterative residual refinement [19]. The constructed 4D correlation volumes are encoded to 2D motion features, which are iteratively decoded to predict the residual flow. The final flow prediction is a sum of the sequence of residual flows. The benefit of using a GRU to perform iterative refinement lies in the reduction of the search space. In RAFT, convolutions are used in the GRU decoder, which learn to model spatial smoothness. Due to the local nature of convolutions, they can learn to handle small occlusions but tend to fail when these become more significant and local evidence is insufficient to resolve the motion.

3.2. Overview

In his first paper from 1976, Geoffrey Hinton wrote that "local ambiguities have to be resolved by finding the best global interpretation" [13]. This idea still holds true in the modern deep learning era. To resolve ambiguities caused by occlusions, our core idea is to allow the network to reason at a higher level, that is, to globally aggregate the motion features of similar pixels, having implicitly reasoned about which pixels are similar in appearance feature space. We hypothesise that the network will be able to find points with similar motions by looking for points with similar appearance in the reference frame. This is motivated by the observation that the motions of points on a single object are often homogeneous. For example, the motion vectors of a person running to the right have a bias towards the right, which holds even if we do not see where a large part of the person ends up in the matching frame due to occlusion. We can use this statistical bias to propagate motion information from non-occluded pixels, with high (implicit) confidence, to occluded pixels, with low confidence. Here, confidence can be interpreted as whether there exists a distinct matching, *i.e.*, a high correlation value at the correct displacement.

With these ideas, we take inspiration from transformer networks [39], which are known for their ability to model long-range dependencies. Different from the self-attention mechanism in transformers, where query, key and value come from the same feature vectors, we use a generalized variant of attention. Our query and key features are projections of the context feature map, which are used to model the appearance self-similarities in frame 1. The value features are projections of the motion features, which themselves are an encoding of the 4D correlation volume. The attention matrix computed from the query and key features is used to aggregate the value features which are hidden representations of motions. We name this a Global Motion Aggregation (GMA) module. The aggregated motion features are concatenated with the local motion features as well as the context features, which is to be decoded by the GRU. A detailed diagram of GMA is shown in Figure 4.

3.3. Mathematical Formulation

Let $\mathbf{x} \in \mathbb{R}^{N \times D_c}$ denote the context (appearance) features and $\mathbf{y} \in \mathbb{R}^{N \times D_m}$ denote the motion features, where $N = HW$ and H and W are the height and width of the feature map, D refers to the channel dimension of the feature map. The i^{th} feature vector is denoted $\mathbf{x}_i \in \mathbb{R}^{D_c}$. Our GMA module computes the feature vector update as an attention-weighted sum of the projected motion features. The aggregated motion features are given by

$$\hat{\mathbf{y}}_i = \mathbf{y}_i + \alpha \sum_{j=1}^{N} f(\theta(\mathbf{x}_i), \phi(\mathbf{x}_j))\sigma(\mathbf{y}_j), \qquad (1)$$

where α is a learned scalar parameter initialised to zero, θ, ϕ and σ are the projection functions for the query, key, and

(a) Reference Frame (b) Matching Frame (c) Occ (d) Occ-in (e) Occ-out

Figure 5. **Examples of the Sintel Albedo dataset and occlusion maps.** The Albedo dataset is rendered without the illumination effects. The occlusion map in this example contains mostly foreground objects occluding the background scene as well as the background on the left moving out of the field-of-view. Figure 5(c) is the occlusion map (Occ) for this example. Figure 5(d) and Figure 5(e) are the in-frame (Occ-in) and out-of-frame (Occ-out) occlusion maps respectively.

value vectors, and f is a similarity attention function given by

$$f(\mathbf{a}_i, \mathbf{b}_j) = \frac{\exp\left(\mathbf{a}_i^\mathsf{T}\mathbf{b}_j/\sqrt{D}\right)}{\sum_{j=1}^{N}\exp\left(\mathbf{a}_i^\mathsf{T}\mathbf{b}_j/\sqrt{D}\right)}. \tag{2}$$

The projection functions for the query, key and value vectors are given by

$$\theta(\mathbf{x}_i) = \mathbf{W}_{\text{qry}}\mathbf{x}_i, \tag{3}$$

$$\phi(\mathbf{x}_i) = \mathbf{W}_{\text{key}}\mathbf{x}_i, \tag{4}$$

$$\sigma(\mathbf{y}_i) = \mathbf{W}_{\text{val}}\mathbf{y}_i, \tag{5}$$

where $\mathbf{W}_{\text{qry}}, \mathbf{W}_{\text{key}} \in \mathbb{R}^{D_{\text{in}} \times D_{\text{c}}}$ and $\mathbf{W}_{\text{val}} \in \mathbb{R}^{D_{\text{m}} \times D_{\text{m}}}$. The learnable parameters in our GMA module include $\mathbf{W}_{\text{qry}}, \mathbf{W}_{\text{key}}, \mathbf{W}_{\text{val}}$ and α.

The final output is $[\mathbf{y} \mid \hat{\mathbf{y}} \mid \mathbf{x}]$, a concatenation of the three feature maps. The GRU decodes this to obtain the residual flow. Concatenation allows the network to intelligently select from or combine the motion vectors, modulated by the global context feature, without prescribing exactly how it is to do this. It is plausible that the network learns to encode some notion of uncertainty, and decodes the aggregated motion vector only when the model cannot be certain of the flow from the local evidence.

We also explore the use of a 2D relative positional embedding [3], allowing the attention map to depend on both the feature self-similarity and the relative position from the query point. For this, we compute the aggregated motion vector as

$$\hat{\mathbf{y}}_i = \mathbf{y}_i + \alpha \sum_{j=1}^{N} f(\theta(\mathbf{x}_i), \phi(\mathbf{x}_j) + \mathbf{p}_{j-i})\sigma(\mathbf{y}_j), \tag{6}$$

where $\mathbf{p}_{j-i}$ denotes the relative positional embedding vector indexed by the pixel offset $j - i$. Separate embedding vectors are learned for the vertical and horizontal offsets and are summed to obtain $\mathbf{p}_{j-i}$. If it is useful to suppress pixels that are very close or very far from the query point when aggregating the motion vectors, then this positional embedding has the capacity to learn this behaviour.

We also investigated computing the attention map from only the query vectors and positional embedding vectors, without any notion of self-similarity. That is,

$$\hat{\mathbf{y}}_i = \mathbf{y}_i + \alpha \sum_{j=1}^{N} f(\theta(\mathbf{x}_i), \mathbf{p}_{j-i})\sigma(\mathbf{y}_j). \tag{7}$$

This can be regarded as learning long-range aggregation without reasoning about the image content. It is plausible that positional biases in the dataset could be exploited by such a scheme. In Table 2, the results for (6) and (7) are denoted as Ours (+p) and Ours (p only).

4. Experiments

4.1. Experimental Setup

We follow the standard optical flow training procedure [20, 36, 38] of first pre-training our model on FlyingChairs [11] for 120k iterations with a batch size of 8 and then on FlyingThings [26] for another 120k iterations with a batch size of 6. We then fine-tune on a combination of FlyingThings, Sintel [8], KITTI 2015 [28] and HD1K [24] for 120k iterations for Sintel evaluation and 50k on KITTI 2015 [28] for KITTI evaluation. A batch size of 6 is set for fine-tuning. We train our model on two 2080Ti GPUs with the `PyTorch` library [29] using the mixed precision strategy. We adopt the same hyperparameters as RAFT [38] for the base network. We adopt the one-cycle learning rate policy [32] with the highest learning rate set to 2.5×10^{-4} for FlyingChairs then 1.25×10^{-4} for the rest. For GMA, we choose channel dimensions $D_{\text{in}} = D_{\text{c}} = D_{\text{m}} = 128$.

The main evaluation metric we use is average end-point error (AEPE), which refers to the mean pixelwise flow error. KITTI also uses the Fl-All (%) metric which refers to the percentage of optical flow vectors whose end-point error is larger than 3 pixels or over 5% of ground truth.

The Sintel dataset has been created with different rendering passes that have different levels of complexity. For training and test evaluation on the Sintel server, we used the Clean and Final passes. The Clean pass is rendered with illumination including smooth shading and specular reflections. The Final pass is created with full rendering, which includes motion blur, camera depth-of-field blur, and atmospheric effects.

In the Sintel training set, they also provided the Albedo pass, which is rendered without illumination effects and has

Sintel Dataset	Type	RAFT (AEPE)	Ours (AEPE)	Rel. Impr. (%)
Clean (train)	Noc	0.32	**0.29**	9.3
	Occ	5.36	**4.25**	20.7
	Occ-in	4.45	**3.81**	14.4
	Occ-out	7.01	**5.03**	**28.2**
	All	0.74	**0.62**	16.2
Final (train)	Noc	0.66	**0.59**	10.6
	Occ	7.09	**6.22**	12.2
	Occ-in	6.21	**5.30**	**14.6**
	Occ-out	8.71	**7.90**	9.3
	All	1.19	**1.06**	10.9
Albedo (test)	Noc	0.34	**0.32**	5.9
	Occ	6.35	**5.58**	12.1
	Occ-in	5.83	**5.23**	10.3
	Occ-out	7.29	**6.20**	**15.0**
	All	0.84	**0.76**	9.5

Table 1. **Optical flow error for different Sintel datasets**, partitioned into occluded ('Occ') and non-occluded ('Noc') regions. In-frame and out-of-frame occlusions are further split and denoted as 'Occ-in' and 'Occ-out'. The best results and the largest relative improvement in each dataset are styled in bold.

roughly piecewise-constant colours. An example is shown in Figure 5. We do not use this set for training, but reserve it as an evaluation dataset. The motivation for doing so is that the Albedo set adheres to brightness constancy everywhere apart from occluded regions. By evaluating and analysing on the occluded regions and non-occluded regions separately, we can clearly see how well our method performs when addressing the occlusion problem.

4.2. Occlusion Analysis

To verify the effectiveness of our proposed GMA module at estimating the motion of occluded points, we make use of the occlusion maps provided in the Sintel training set, which partition the pixels into non-occluded (Noc) and occluded (Occ) pixels. We further divide the occluded pixels into in-frame ('Occ-in') and out-of-frame ('Occ-out') occlusions, depending on whether the ground-truth flow vector points inside or outside the image frame. An example is shown in Figure 5.

We evaluated on all three rendering passes of Sintel, where the results for Clean and Final are training set errors and those for Albedo are test set errors. We evaluated the AEPE for different regions, results of which are shown in Table 1. We observe that the relative improvement of our method compared to RAFT is predominantly attributable to better predictions of the flow for occluded points. This is reinforced by the results on the Albedo dataset where the brightness constancy assumption holds exactly for non-occluded points, removing confounding factors. Finally, out-of-frame occlusions are more challenging than in-frame occlusions for both models, but we still observe a significant improvement for these pixels. We hypothesise that the improvement in non-occluded regions is due to GMA's ability to resolve ambiguities caused by other brightness variations, for example specular reflections, blurs, and other sources. This result strongly supports our claim that global aggregation can help resolve ambiguities caused by occlusion.

4.3. Comparison with Prior Works

Having shown that our approach can improve optical flow estimates for occluded regions, we compare against prior works on the overall performance. We evaluate our approach on the Sintel dataset [8] and the KITTI 2015 optical flow dataset [28]. At the time of submission, we have obtained the best results on both the Sintel Final and Clean benchmarks among all submitted results published and unpublished. Compared with our baseline approach RAFT [38], we have improved the AEPE from 2.86 to 2.47 (13.6% improvement) on Sintel Final and 1.61 to 1.39 (13.7% improvement) on Sintel Clean. This significant improvement over RAFT validates our claim that our approach can improve flow prediction for occluded regions without damaging the performance of non-occluded regions. The Sintel server also reports the metric 'EPE unmatched', which measures the endpoint error over regions that are visible only in one frame, predominantly caused by occlusion. Our approach also ranks first under this metric in both Clean and Final, with a margin of 0.9 EPE on Clean (2.2 w.r.t. RAFT) and 1.3 EPE on Final (1.7 w.r.t. RAFT). Overall, our model achieves a new state-of-the-art result in optical flow estimation, which demonstrates the usefulness of addressing the occlusion problem in optical flow.

On the KITTI 2015 test set, our results are on par with RAFT. 'Ours (p only)', which uses positional attention only, outperforms RAFT, while 'Ours', which uses content self-similarity attention, slightly underperforms. It is likely that the lack of improvement on this dataset is due to having insufficient training data (only 200 pairs of images) for the network to learn high-level appearance feature similarities.

4.4. Qualitative Results

Qualitative results are shown in Figure 2 for two examples in the Sintel Clean dataset. The optical flow error in regions of the image that move out-of-frame or behind another object is significantly reduced compared to RAFT. These scenes are highly challenging with lots of motion and occlusion. For example, it is not unreasonable that RAFT is unable to keep track of the wooden staff that becomes partially occluded in the second image, given that it is well-camouflaged in a forest, fast-moving, and very thin. However, our model is able to very accurately predict the staff's

Training Data	Method	Sintel (train)		KITTI-15 (train)		Sintel (test)		KITTI-15 (test)
		Clean	Final	AEPE	Fl-all (%)	Clean	Final	Fl-all (%)
C + T	VCN[44]	2.21	3.68	8.36	25.1	-	-	-
	MaskFlowNet[47]	2.25	3.61	-	23.1	-	-	-
	FlowNet2[20]	2.02	3.54	10.08	30.0	3.96	6.02	-
	RAFT[38]	1.43	**2.71**	5.04	17.4	-	-	-
	Ours (p only)	1.48	2.88	5.01	16.9	-	-	-
	Ours (+p)	1.33	2.87	4.83	**16.6**	-	-	-
	Ours	**1.30**	2.74	**4.69**	17.1	-	-	-
C + T + S/K (+ H)	FlowNet2 [20]	(1.45)	(2.01)	(2.30)	(6.8)	4.16	5.74	11.48
	PWC-Net+[37]	(1.71)	(2.34)	(1.50)	(5.3)	3.45	4.60	7.72
	VCN [44]	(1.66)	(2.24)	(1.16)	(4.1)	2.81	4.40	6.30
	MaskFlowNet[47]	-	-	-	-	2.52	4.17	6.10
	RAFT[38]	(0.76)	(1.22)	(0.63)	(1.5)	1.61⋆	2.86⋆	5.10
	Ours (p only)	(0.64)	(1.08)	(**0.56**)	(**1.2**)	1.48⋆	2.56⋆	**4.93**
	Ours (+p)	(0.65)	(1.11)	(0.58)	(1.3)	1.54⋆	2.63⋆	5.08
	Ours	(**0.62**)	(**1.06**)	(0.57)	(**1.2**)	**1.39**⋆	**2.47**⋆	5.15

Table 2. **Quantitative results on Sintel and KITTI 2015 datasets.** We report the average end-point error (AEPE) where not otherwise stated, as well as the Fl-all measure for the KITTI dataset, which is the percentage of optical flow outliers with an error larger than 3 pixels. "C + T" refers to results that are pre-trained on the Chairs and Things datasets. "S/K (+ H)" refers to methods that are fine-tuned on the Sintel and KITTI datasets, with some also fine-tuned on the HD1K dataset. Parentheses denote training set results and bold font denotes the best result. "Ours (p only)" denotes the position-only attention model defined in (7). "Ours (+p)" denotes the joint position and content-wise attention model defined in (6). "Ours" denotes our main content-only self-similarity attention model defined in (1). ⋆Results evaluated with the "warm-start" strategy detailed in the RAFT paper [38].

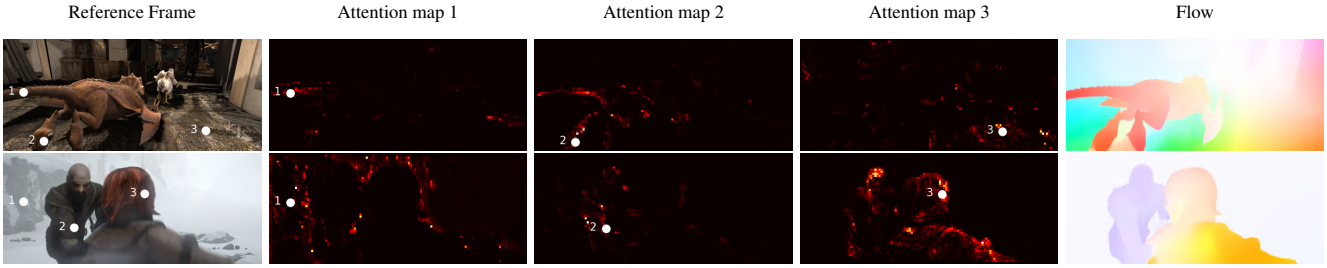

Figure 6. **Attention map visualisations.** For each row, we show the first frame and three query points. Then we show the three attention maps corresponding to these query points (brighter colours mean higher attention weights). We also give a visualisation of the predicted optical flow for comparison. Best viewed on screen.

motion, despite these challenges.

We also present visualisations of the learned attention maps for two examples in Figure 6. To train effectively, the network should learn to attend to pixels that share similar motion vectors. For foreground points, we expect this to be most easily achieved by attending to points on the same object, while for background points it may be sufficient to attend to any other background point. These examples justify this expectation and provide support for the argument that appearance (and higher-order) self-similarity is being learned by the network, and that this is helpful for estimating the flow of the occluded points.

4.5. Ablation Results

To verify our design, we conducted the following ablation experiments. We first compare the performance of the tested variants of the model, where positional attention replaces (p only) or adds to (+p) the self-similarity attention, as presented in Table 2. We find that self-similarity is sufficient to achieve the performance improvements, with the positional encoding only helping for the KITTI dataset. This coincides with our intuition that long-range connections are helpful and that distance-based suppression is unnecessary. In addition, we ablate over three design choices: (1) learning the scalar parameter α vs fixing it at 1, (2) concatenating with local motion features vs replacing local mo-

Component	Chairs (val)	Things Clean (test)	Things Final (test)	Sintel Clean (train)	Sintel Final (train)
1	0.82	**3.10**	**2.78**	1.35	2.82
$\underline{\alpha}$	**0.79**	3.14	2.80	**1.30**	**2.74**
replace	0.88	3.16	2.94	1.41	2.79
<u>concatenate</u>	**0.79**	**3.14**	**2.80**	**1.30**	**2.74**
w/o residual	0.88	**3.13**	2.83	1.40	2.75
<u>w/ residual</u>	**0.79**	3.14	**2.80**	**1.30**	**2.74**

Table 3. **Ablation experiment results.** Settings used in our final model are underlined.

Metric	RAFT [38]	Ours
Parameters	5.3M	5.9M
Timing	60ms	72ms
GPU Memory	16.0GB	17.7GB

Table 4. **Timing, parameters and memory.** The GMA module has a modest computational overhead.

tion features, and (3) using a residual connection (adding the output of the aggregator to the local motion features) vs not using residual connection (directly concatenating the output of the aggregator with the motion features and context features). The results are shown in Table 3.

The key experiment here is showing that concatenation is an important part of the network design. The hypothesis was that the network should learn how to select or combine the local and globally-aggregated features, based on some implicit measure of uncertainty. That is, it is not helpful to replace local features in most non-occluded regions, where they may be more reliable and precise than the aggregated features. While the residual connection may also be able to handle this, using both mechanisms leads to the best performance.

4.6. Timing, Parameter Counts and Memory

We demonstrate that the computational overhead of GMA is low relative to the performance improvement, as shown in Table 4. The parameter count for our model is 5.9M compared to RAFT which is 5.3M. We tested the inference time on a single RTX 3090 GPU, with RAFT taking 60ms on average and ours taking 72ms for a single pair of image in the Sintel dataset. The image size is 436×1024. The GRU iteration number is set to 12. We also tested the GPU memory consumption for training. When training on FlyingChairs on a single 3090 card, with a random crop of 368×496 and batch size of 8, RAFT takes 16.0GB memory while our network takes 17.2GB memory. We can see that overall the computational overhead is modest while the improvement in results is significant.

5. Discussion

We have demonstrated empirically that long-range connections, weighted by image self-similarities, are very effective at resolving the optical flow of occluded 3D points. The intuition is that if the network can determine which non-occluded points are moving in the same way, this information can be transmitted to 'in-paint' the motion of the occluded points. Determining which points have similar motion characteristics is a non-trivial task and relies on the exploitation of statistical biases. Similar flow vectors are frequently observed for points belonging to the same class, due to the homogeneous motion in 3D. This suggests that we should enable the network to aggregate over motions of the same scene objects, which motivates our choice to explicitly expose the self-similarity of image features to our GMA module. However, additive aggregation of this kind is only helpful when the flow field of the attended locations is approximately homogeneous. This does not hold exactly for general object and camera motions, where the flow fields may be far from homogeneous, even on the same rigid object. An example is an object that is directly in front of the camera and rotating about the optical axis, where the flow vectors are in opposite directions. To deal with such scenarios, one possible future work is to first transform the motion features based on the relative positions and perform aggregation afterwards.

6. Conclusion

Occlusions have long been considered a significant challenge and a major source of error in optical flow estimation. Inspired by the recent success of transformers, we introduce a global motion aggregation module to globally aggregate motion features based on appearance self-similarity of the first image. This has been validated by experiments that show significantly improved optical flow predictions for occluded regions, particularly the large reduction of EPE on Sintel Clean and Final. Our approach of aggregating information over long-range connections using self-similarity is a simple and effective way to introduce higher-order reasoning into the optical flow problem and is applicable to any supervised flow network. We expect that further development of aggregation mechanisms or alternatives would lead to additional performance improvements.

Acknowledgements

This research is funded in part by the ARC Centre of Excellence for Robotic Vision (CE140100016), ARC Discovery Project grant (DP200102274) and (DP190102261), and Continental AG (D.C.). S.J. would like to thank Jing Zhang and Yujiao Shi for helpful discussions. We thank the anonymous reviewers for their valuable comments.

References

[1] Luis Alvarez, Rachid Deriche, Théo Papadopoulo, and Javier Sánchez. Symmetrical dense optical flow estimation with occlusions detection. *IJCV*, 2007.

[2] Aviram Bar-Haim and Lior Wolf. Scopeflow: Dynamic scene scoping for optical flow. *CVPR*, 2020.

[3] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. *ICCV*, 2019.

[4] Michael J Black and Paul Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *CVIU*, 1996.

[5] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *TPAMI*, 2004.

[6] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. *ECCV*, 2004.

[7] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *IJCV*, 2005.

[8] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. *ECCV*, 2012.

[9] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. *CVPR*, 2016.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2020.

[11] 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. *ICCV*, 2015.

[12] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. *CVPR*, 2019.

[13] G Hinton. Using relaxation to find a puppet. *Proceedings of the 2nd Summer Conference on Artificial Intelligence and Simulation of Behaviour*, 1976.

[14] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Techniques and Applications of Image Understanding*, 1981.

[15] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *TPAMI*, 2012.

[16] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. *ICCV*, 2019.

[17] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. *CVPR*, 2018.

[18] Junhwa Hur and Stefan Roth. Mirrorflow: Exploiting symmetries in joint optical flow and occlusion estimation. *ICCV*, 2017.

[19] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. *CVPR*, 2019.

[20] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. *CVPR*, 2017.

[21] Joel Janai, Fatma Guney, Anurag Ranjan, Michael Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. *ECCV*, 2018.

[22] Shihao Jiang, Yao Lu, Hongdong Li, and Richard Hartley. Learning optical flow from a few matches. *CVPR*, 2021.

[23] Dieter Koller, Joseph Weber, and Jitendra Malik. Robust multiple car tracking with occlusion reasoning. *ECCV*, 1994.

[24] 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. *CVPR Workshop*, 2016.

[25] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. *CVPR*, 2019.

[26] 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. *CVPR*, 2016.

[27] Moritz Menze, Christian Heipke, and Andreas Geiger. Discrete optimization for optical flow. *GCPR*, 2015.

[28] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS Workshop on Image Sequence Analysis*, 2015.

[29] 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. *NIPS Workshop*, 2017.

[30] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models. *NeurIPS*, 2019.

[31] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. *CVPR*, 2015.

[32] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, 2019.

[33] Deqing Sun, Ce Liu, and Hanspeter Pfister. Local layering for joint motion estimation and occlusion detection. *CVPR*, 2014.

[34] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. 2010.

[35] Deqing Sun, Erik B Sudderth, and Michael J Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. *NIPS*, 2010.

[36] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. *CVPR*, 2018.

[37] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *TPAMI*, 2019.

[38] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. *ECCV*, 2020.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NIPS*, 2017.

[40] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *CVPR*, 2018.

[41] Yang Wang, Yi Yang, Zhenheng Yang, Liang Zhao, Peng Wang, and Wei Xu. Occlusion aware unsupervised learning of optical flow. *CVPR*, 2018.

[42] Andreas Wedel, Thomas Pock, Christopher Zach, Horst Bischof, and Daniel Cremers. An improved algorithm for tv-l 1 optical flow. *Statistical and Geometrical Approaches to Visual Motion Analysis*, 2009.

[43] Jia Xu, René Ranftl, and Vladlen Koltun. Accurate optical flow via direct cost volume processing. *CVPR*, 2017.

[44] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. *NeurIPS*, 2019.

[45] Zhichao Yin, Trevor Darrell, and Fisher Yu. Hierarchical discrete distribution decomposition for match density estimation. *CVPR*, 2019.

[46] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. *Joint Pattern Recognition Symposium*, 2007.

[47] Shengyu Zhao, Yilun Sheng, Yue Dong, Eric I Chang, Yan Xu, et al. Maskflownet: Asymmetric feature matching with learnable occlusion mask. *CVPR*, 2020.