Discrete Optimization for Optical Flow

Moritz Menze[1], Christian Heipke[1], and Andreas Geiger[2]

[1] Leibniz Universität Hannover, Germany
[2] Max Planck Institute for Intelligent Systems, Tübingen, Germany

Abstract. We propose to look at large-displacement optical flow from a discrete point of view. Motivated by the observation that sub-pixel accuracy is easily obtained given pixel-accurate optical flow, we conjecture that computing the integral part is the hardest piece of the problem. Consequently, we formulate optical flow estimation as a discrete inference problem in a conditional random field, followed by sub-pixel refinement. Naïve discretization of the 2D flow space, however, is intractable due to the resulting size of the label set. In this paper, we therefore investigate three different strategies, each able to reduce computation and memory demands by several orders of magnitude. Their combination allows us to estimate large-displacement optical flow both accurately and efficiently and demonstrates the potential of discrete optimization for optical flow. We obtain state-of-the-art performance on MPI Sintel and KITTI.

1 Introduction

Estimating dense optical flow is a fundamental problem in computer vision. Despite significant progress over the last decades, realistic scenes with displacements of several hundred pixels, strong changes in illumination and textureless or specular regions remain challenging to date [9, 17]. Traditionally, dense optical flow estimation has been formulated as a *continuous* optimization problem [20, 25], and many of today's most successful methods leverage elaborate variants of the original formulation, allowing for more robust penalties or improving optimization. As continuous methods typically require linearizing the highly non-convex data term, they only permit the estimation of very small displacements up to a few pixels. Thus, in order to handle large displacements in real-world videos, a simple heuristic is often employed: Optical flow is estimated in a coarse-to-fine manner, thereby guaranteeing an upper bound to the maximal displacement at each level of the image pyramid. Unfortunately, this strategy is highly susceptible to local minima as small structures and textural details vanish at coarse image resolutions, leading to oversmoothing artifacts in the estimated flow field.

In contrast to optical flow, the most successful approaches to stereo matching typically rely on *discrete* inference in graphical models. While such models are loopy by nature and thus lead to NP-hard optimization problems, good approximate solutions can often be efficiently computed using graph cuts, belief propagation or mean field approximations. Importantly, no image pyramids are required as the full data cost volume is considered at the same time during inference. Unfortunately, the application of discrete methods to the problem of

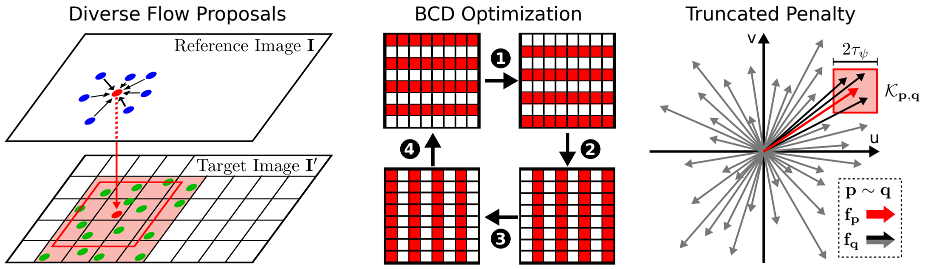

Diverse Flow Proposals
Reference Image $\mathbf{I}$
Target Image $\mathbf{I}'$
BCD Optimization
❶
❷
❸
❹
Truncated Penalty
v
u
$2\tau_\psi$
$\mathcal{K}_{\mathbf{p},\mathbf{q}}$
$\mathbf{p} \sim \mathbf{q}$
$\mathbf{f}_\mathbf{p}$
$\mathbf{f}_\mathbf{q}$

Fig. 1: **Strategies for Efficient Discrete Optical Flow.** Left: We create a large set of diverse flow proposals per pixel (red node) by combining nearest neighbors in feature space from a set of grid cells (green nodes) with winner-takes-all solutions from neighboring pixels (blue nodes). The red square indicates the search region. Middle: We apply block coordinate descent, iteratively optimizing all image rows and columns (red) conditioned on neighboring blocks (white) via dynamic programming. Right: Taking advantage of robust penalties, we reduce pairwise computation costs by pre-computing the set of non-truncated ($< \tau_\psi$) neighboring flow proposals (black) for each flow vector (red).

optical flow is not straightforward and hence there exists only relatively little work in this direction. The main reason for this is the huge size of the label space which needs to be considered for the 2D large-displacement optical flow problem as opposed to the 1D stereo problem.

In this paper, we propose three different strategies to make discrete inference in a pairwise conditional random field (CRF) applicable to the estimation of optical flow, see Fig. 1 for an illustration. First, we restrict the label set by considering only the L most likely matches per pixel which we obtain via approximate nearest neighbor search in feature space subject to non-maxima suppression constraints. To validate this restriction, we experimentally show that the oracle solution of the restricted set outperforms all existing optical flow techniques by a significant margin. Second, our inference scheme takes advantage of efficient convergent block coordinate descent (BCD) and iteratively updates all image rows and columns conditioned on the remaining variables via dynamic programming. Third, we exploit the special form of the pairwise potentials used by our formulation to further decrease computational complexity, thereby making very large unordered label sets with hundreds of labels tractable. Upon convergence, we remove outliers (e.g., in occluded regions) using strategies borrowed from the stereo literature. Finally, we regress a real-valued dense flow field from our semi-dense integer flow estimate using variational techniques [32, 42]. We experimentally validate the proposed method on two challenging benchmarks: MPI Sintel [9] and KITTI [17]. Our experiments show that the proposed method attains state-of-the-art performance. Importantly, our results indicate that discrete optimization can be a powerful tool for solving optical flow problems - even when considering pairwise flow priors only. Our code and supplementary material are available from our project page: http://www.cvlibs.net/projects/discrete_flow.

2 Related Work

Global estimation of optical flow has traditionally been formulated as a continuous variational optimization problem with linearized data terms [4,20] and many of the most successful works still follow the same paradigm to date [8,12,30,34,35,37,40,43]. To estimate displacements larger than 1 pixel, as required by modern benchmarks [1,9,17] and for many applications, continuous methods typically rely on image pyramids [7]. Unfortunately, this heuristic is prone to local minima as texture details and fine structures vanish at small scales. As a consequence, methods leveraging sparse feature correspondences to guide the optimization process have been popularized [5,6,32,38,42], incorporating feature matches into initialization or as an additional data term. Our approach shares similarity with these methods in the sense that we also refine our integer-valued flow result to sub-pixel accuracy as proposed in [32,42]. However, in contrast to the above-mentioned works, our integral matches are obtained via discrete optimization with optical flow priors. This allows our algorithm to establish denser correspondences than possible with independently matched sparse features and, in combination with sub-pixel refinement, leads to better results.

Alternatively, optical flow can be viewed as a discrete optimization problem. In the absence of regularization constraints, this corresponds to a complete search of the discretized 2D flow space which has to be carried out for each pixel [2,47]. While local support windows [33] alleviate the effect of border bleeding, they generally cannot compete with methods that leverage global regularization. Incorporating smoothness constraints, however, is much more difficult for discrete optical flow than for the related stereo matching problem due to the extremely large label space of the discretized 2D flow field. To avoid this difficulty, a number of approaches formulate optical flow as a segmentation problem: Based on a small set of dense flow field proposals, the most likely flow field at each pixel subject to regularization constraints is approximated via quadratic pseudo-boolean optimization (QPBO) or belief propagation (BP) [11,23,44,45,48]. A reduced label set for all pixels is proposed by [28] restricting the method to scenes with little and non-complex motion. In contrast, here we pursue a more direct approach to discrete optical flow which does not rely on proposed flow fields or a global label set but allows for an arbitrary set of flow proposals *per pixel*. Compared to the approaches of Steinbrücker et al. [36] and Liu et al. [24], we neither require a complete search nor image pyramids [24]. Our approach is also related to particle-based methods such as particle BP [46] or PatchMatch BP [3,21]. In contrast to those methods, our algorithm is not restricted to a relatively small set of particles or local resampling which can lead to local minima. Instead, we maintain diverse distributions over much larger state spaces throughout the whole inference process.

Very recently, object recognition and feature learning have been exploited as a powerful source of information for optical flow. Notable examples are the data-driven flow transfer method of Wei et al. [41] and FlowNets based on deep neural networks by Fischer et al. [15]. In this paper our focus lies on a more generic model without the need for large annotated training sets. However, these ideas could be easily incorporated into the proposed model, e.g., via additional unary terms, promising further gains in performance in the future.

3 Model

Let us assume two input images $\mathbf{I}$ and $\mathbf{I}'$ of size $W \times H$. Our goal is to assign as many pixels as possible from the reference image $\mathbf{I}$ to pixels in the target image $\mathbf{I}'$. In other words, we aim for a semi-dense integer-valued flow field, establishing correspondences which are visible in both views. Flow vectors are selected from a diverse set of proposals (Section 3.1) based on a CRF model (Section 3.2) which is efficiently optimized (Section 3.3). Interpolation, extrapolation and sub-pixel refinement are then addressed in a subsequent post-processing step (Section 3.4).

3.1 Diverse Flow Proposals

Unfortunately, naïve pixel-wise discretization of all possible flow vectors would lead to an intractably large set of labels at each pixel. Consider for instance a flow range of ±250 pixels in u- and v-direction. While this range is sufficiently large for the datasets tackled in this paper (see supplementary material), it leads to more than $60,000$ labels per pixel for which even the data term alone would be challenging to calculate and to store. In this section, we therefore propose a mechanism which extracts a subset of L flow proposals ($L = 500$ in our experiments) while maintaining a high recall of ground truth optical flow.

Towards this goal, we partition the target image into cells of equal size as illustrated in Fig. 1 (left). For each cell, we apply the randomized k-d tree algorithm [29] on the feature descriptors[1] of the pixels falling into that cell, yielding an efficient approximate search structure. Next, for each pixel in the reference image (red node in Fig. 1), we find all relevant cells in the target image according to the desired optical flow range (red shaded cells in Fig. 1) and concatenate all M flow vectors corresponding to the K nearest neighbor matches from the k-d tree of each cell (green nodes in Fig. 1). In contrast to a single search structure constructed for the entire target image, this strategy has two advantages: First, arbitrary optical flow search ranges can be implemented with little computational overhead by searching only in relevant cells. Second, the retrieved flow vectors respect an approximate non-maximal-suppression constraint, i.e., we obtain exactly K flow vectors per cell.

As neighboring pixels often exhibit similar optical flow, we additionally sample N random pixels (blue nodes in Fig. 1) from a local Gaussian distribution centered at the reference pixel and add the respective winner-takes-all solution (i.e., the flow vector corresponding to the best match at the sampled pixel) to the proposal set of the current pixel. In case it is already present in the proposal set, we proceed with the next best flow vector, ensuring that the final set of $L = M + N$ flow vectors is unique. In our experiments we use $M = 300$ nearest neighbors and $N = 200$ proposals from neighboring pixels.

[1] As feature descriptor, we leverage DAISY [39] due to its computational efficiency and robustness against changes in illumination.

3.2 Random Field Model

We associate a discrete label $l_{\mathbf{p}} \in \{1, \ldots, L\}$ with each pixel $\mathbf{p} = (x, y)$ in the reference image, corresponding to the (unique) integer-valued proposal flow vectors $\mathbf{f}_{\mathbf{p}}(l_{\mathbf{p}}) \in \mathbb{Z}^2$ described in Section 3.1. We consider optical flow estimation as MAP inference in a pairwise CRF. More specifically, we aim at minimizing

$$E(\mathbf{l}) = \lambda \sum_{\mathbf{p} \in \mathcal{P}} \underbrace{\varphi_{\mathbf{p}}(l_{\mathbf{p}})}_{\text{data}} + \sum_{\mathbf{p} \sim \mathbf{q}} \underbrace{\psi_{\mathbf{p},\mathbf{q}}(l_{\mathbf{p}}, l_{\mathbf{q}})}_{\text{smoothness}} \tag{1}$$

with respect to the set of image labels $\mathbf{l} = \{l_{\mathbf{p}} | \mathbf{p} \in \mathcal{P}\}$ where $\mathcal{P}$ denotes the set of pixels in image $\mathbf{I}$. Here, we have dropped the dependency on the input images $\mathbf{I}$ and $\mathbf{I}'$ for clarity and $\sim$ denotes all neighbors on a 4-connected image grid. The relative weight between the data term (measuring data fidelity) and the smoothness term (encouraging smooth flow fields) is defined by λ.

The data term $\varphi_{\mathbf{p}}(l_{\mathbf{p}})$ encodes the cost at pixel $\mathbf{p}$ given label $l_{\mathbf{p}}$. We model it as the truncated ℓ_1-penalty evaluating the difference between feature descriptors

$$\varphi_{\mathbf{p}}(l_{\mathbf{p}}) = \min\left(\|\mathbf{d}_{\mathbf{p}} - \mathbf{d}'_{\mathbf{p}}(l_{\mathbf{p}})\|_1, \tau_{\varphi} \right) \tag{2}$$

where $\mathbf{d}_{\mathbf{p}} \in \mathbb{R}^D$ denotes the feature descriptor at pixel $\mathbf{p}$ in the reference image, $\mathbf{d}'_{\mathbf{p}}(l_{\mathbf{p}}) \in \mathbb{R}^D$ denotes the feature descriptor associated with pixel $\mathbf{p}$ and label $l_{\mathbf{p}}$ in the target image, and τ_{φ} is the truncation threshold of the data term.

The second term in Eq. 1 encourages smooth flow fields and is modeled as the weighted truncated ℓ_1-distance of neighboring optical flow vectors

$$\psi_{\mathbf{p},\mathbf{q}}(l_{\mathbf{p}}, l_{\mathbf{q}}) = w_{\mathbf{p},\mathbf{q}} \ \min\left(\|\mathbf{f}_{\mathbf{p}}(l_{\mathbf{p}}) - \mathbf{f}_{\mathbf{q}}(l_{\mathbf{q}})\|_1, \tau_{\psi} \right) \tag{3}$$

where $\mathbf{f}_{\mathbf{p}}(l_{\mathbf{p}}) \in \mathbb{R}^2$ denotes the flow vector at pixel $\mathbf{p}$ associated with label $l_{\mathbf{p}}$ and τ_{ψ} is the truncation threshold of the smoothness term. We remark that a truncated penalty is not only robust against outliers and thus preferable from a statistical point of view [34], but it is also critical for tractable inference in our model as described in Section 3.3. Moreover, note that Eq. 3 is a generalization of the pairwise potential proposed for stereo matching by Hirschmüller [19] to arbitrary truncation thresholds τ_{ψ}. As we expect flow boundaries to coincide with image edges, we additionally weight the smoothness term by a weight factor $w_{\mathbf{p},\mathbf{q}} = \exp\left(-\alpha\, \kappa^2_{\mathbf{p},\mathbf{q}}\right)$. Here, $\kappa_{\mathbf{p},\mathbf{q}} \in [0, 1]$ measures the strength of the edge between pixel $\mathbf{p}$ and pixel $\mathbf{q}$. We calculate $\kappa_{\mathbf{p},\mathbf{q}}$ using structured edge detection [13].

3.3 Inference

Despite the simplicity of the model described in Section 3.2 and the label set of size $L = 500$, inference using max-product loopy BP is still prohibitively slow. To enable efficient inference, we therefore exploit two additional strategies. First, instead of loopy belief propagation, we perform block coordinate descent (BCD).

More specifically, we follow [10] and iteratively update alternating image rows and columns conditioned on the MAP solution of the remaining variables as illustrated in Fig. 1 (middle). Second, we exploit the truncated form of the pairwise potentials in Eq. 3. This is illustrated in Fig. 1 (right). In combination, both steps reduce computational complexity by about four orders of magnitude.

Without loss of generality, consider the optimization of image row y. The naïve dynamic programming algorithm recursively fills the cumulative cost matrix $\mathbf{C}$ for each x from 1 to W using the following update equation:

$$\begin{aligned}\mathbf{C}(x,l) = \lambda\,\varphi_{(x,y)}(l) &+ \psi_{(x,y),(x,y-1)}(l,l^*_{x,y-1}) + \psi_{(x,y),(x,y+1)}(l,l^*_{x,y+1}) \\ &+ \min_{0\le k<L}\left(\psi_{(x,y),(x-1,y)}(l,k) + \mathbf{C}(x-1,k)\right)\end{aligned} \tag{4}$$

Here, $l^*_{\mathbf{p}}$ denotes the assignment of the fixed variables, i.e., the variables outside row y. While the global problem is NP-hard and can be solved only approximately, each sub-problem corresponds to a chain MRF for which we obtain an optimal solution via dynamic programming (backtracking). This leads to the desirable property that this algorithm (unlike loopy BP) is guaranteed to converge.

In case of ordered label sets (e.g., in stereo matching [14] or depth reconstruction [10]) the efficient distance transform [14] can be employed to lower the complexity of this algorithm from $O(WL^2)$ to $O(WL)$ by calculating the expression in the last row of Eq. 4 in linear time. Unfortunately, in our case the flow vectors involved in $\psi_{(x,y),(x-1,y)}(l,k)$ are sparse and unordered, therefore prohibiting the application of this trick. However, we are still able to utilize the fact that our pairwise potentials ψ are truncated at τ_ψ: First, note that for practical truncation thresholds (i.e., $\tau_\psi < 15$), the majority of the L^2 pairwise terms evaluates to τ_ψ. We exploit this observation by partitioning the labels of each neighboring pixel pair $(\mathbf{p},\mathbf{q})$ into sets

$$\mathcal{K}_{\mathbf{p},\mathbf{q},l} = \{k \in \{1,\ldots,L\} \,|\, \|\mathbf{f}_{\mathbf{p}}(l) - \mathbf{f}_{\mathbf{q}}(k)\|_1 < \tau_\psi\} \tag{5}$$

which contain all labels k at pixel $\mathbf{q}$ for which the flow $\mathbf{f}_{\mathbf{q}}(k)$ is within τ_ψ from the flow $\mathbf{f}_{\mathbf{p}}(l)$ associated with label l at pixel $\mathbf{p}$. Fig. 1 (right) illustrates $\mathcal{K}_{\mathbf{p},\mathbf{q}}$ for a single flow vector at pixel $\mathbf{p}$ (shown in red) and a set of flow vectors at pixel $\mathbf{q}$ (black+gray). Given this definition, the last term in Eq. 4 can be written as

$$\min\left(\min_{k\in\mathcal{K}_{(x,y),(x-1,y),l}}\left(\psi_{(x,y),(x-1,y)}(l,k) + \mathbf{C}(x-1,k)\right), c\right) \tag{6}$$

where the constant c is given by

$$c = \min_{0\le k<L}\left(w_{(x,y),(x-1,y)}\tau_\psi + \mathbf{C}(x-1,k)\right) \tag{7}$$

As c does not depend on l and can be calculated in $O(L)$, Eq. 6 can be evaluated (for all l) in $O(\sum_l |\mathcal{K}_{\mathbf{p},\mathbf{q},l}|)$ instead of $O(L^2)$, where $|\mathcal{K}_{\mathbf{p},\mathbf{q},l}| \ll L$ in practice due to the diversity of the proposal set as evidenced by our experimental evaluation in Section 4. It is important to note that the sets $\mathcal{K}_{\mathbf{p},\mathbf{q},l}$ can be pre-computed and reused during all BCD iterations. In our implementation, we further accelerate this pre-computation step using hash maps for efficiently retrieving flow vectors.

	EPE (px)	Out (%)
DM+DeepFlow [42]	2.85	10.22
DM+EpicFlow [32]	**2.25**	8.63
Ours+DeepFlow	2.64	9.14
Ours+EpicFlow	**2.25**	**8.06**
Ours+Oracle	0.85	3.97

(a) **MPI Sintel**

	EPE (px)	Out (%)
DM+DeepFlow [42]	1.40	7.13
DM+EpicFlow [32]	1.41	7.43
Ours+DeepFlow	**1.17**	6.13
Ours+EpicFlow	**1.17**	**5.64**
Ours+Oracle	0.58	1.01

(b) **KITTI**

Table 1: **Pilot Study on MPI Sintel and KITTI Training Sets.** See text.

3.4 Postprocessing

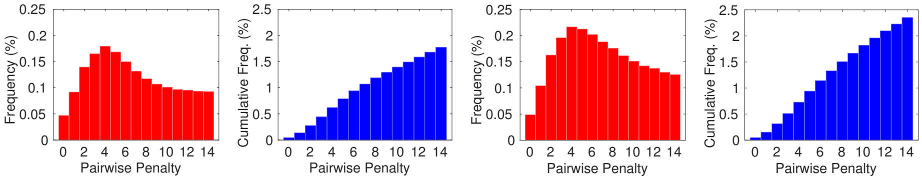
The inference algorithm described above solves the correspondence problem, i.e., it assigns each pixel in the reference image to a pixel in the target image. As we do not model occlusions explicitly, also occluded pixels and pixels leaving the image domain are assigned a label. We therefore remove outliers from our result, borrowing ideas from the stereo matching literature [19]. More specifically, we perform forward-backward consistency checking, i.e., we calculate the optical flow forwards and backwards in time and retain only flow vectors which are consistent. In addition, we remove small isolated segments which often correspond to wrong optical flow estimates using connected component labeling with a minimum segment size of 100 px and a flow consistency threshold of 10 px. Finally, in order to obtain sub-pixel flow values and to inter-/extrapolate into unmatched regions, we refine our results using the state-of-the-art flow refinement techniques of DeepFlow and EpicFlow.

4 Experimental Evaluation

We evaluate our approach on the challenging MPI Sintel [9] and KITTI [17] datasets. Our evaluation starts with a pilot study where we experimentally validate the claims from the introduction, followed by a quantitative and qualitative evaluation of our model. The parameters in our model are set via block coordinate descent on the respective training set (see supplementary material). To further increase efficiency we use a stride of 4 pixels. For subpixel interpolation of our results, we leverage the *interpolation stages* of DeepFlow ("Ours+DeepFlow") and EpicFlow ("Ours+EpicFlow"). On average, our MATLAB/C++ implementation requires about 3 minutes for one 0.5 megapixel color image pair from the KITTI training set (10% descriptor extraction, 40% proposal generation, 35% BCD, 15% postprocessing and overhead).

4.1 Pilot Study

Table 1 shows results on the non-occluded parts of the MPI Sintel and KITTI training sets in terms of endpoint error (EPE) and outliers (threshold: 3 px). We show (from top-to-bottom) the results of DeepFlow [42], EpicFlow [32], our results in combination with the refinement stages of DeepFlow [42] and EpicFlow [32], as well as an "Oracle" which refers to the flow map obtained by selecting



(a) **MPI Sintel** (b) **KITTI**

Fig. 2: **Frequency of Neighboring Flow Proposals wrt. Distance.** This figure shows the frequency of neighboring flow proposal vectors with respect to their endpoint distance on MPI Sintel (a) and KITTI (b) in red. The blue plots depict the cumulative frequency corresponding to $|\mathcal{K}_{\mathbf{p},\mathbf{q},l}|/L$ over the distance threshold τ_ψ, respectively.

	EPE (px)		
	All	Noc	Occ
Ours+EpicFlow	**3.567**	1.108	**23.626**
FlowFields	3.748	**1.056**	25.700
DM+EpicFlow [32]	4.115	1.360	26.595
PH-Flow [48]	4.388	1.714	26.202
AggregFlow [16]	4.754	1.694	29.685
TF+OFM [22]	4.917	1.874	29.735
Deep+R	5.041	1.481	34.047
SPM-BP	5.202	1.815	32.839
SparseFlowFused [38]	5.257	1.627	34.834
DM+DeepFlow [42]	5.377	1.771	34.751

(a) **Clean Set**

	EPE (px)		
	All	Noc	Occ
FlowFields	**5.810**	**2.621**	31.799
Ours+EpicFlow	6.077	2.937	**31.685**
DM+EpicFlow [32]	6.285	3.060	32.564
TF+OFM [22]	6.727	3.388	33.929
Deep+R	6.769	2.996	37.494
SparseFlowFused [38]	7.189	3.286	38.977
DM+DeepFlow [42]	7.212	3.336	38.781
FlowNetS+ft+v [15]	7.218	3.752	35.445
NNF-Local	7.249	2.973	42.088
SPM-BP	7.325	3.493	38.561

(b) **Final Set**

Table 2: **Evaluation on MPI Sintel Test Set.** This table lists the top 10 out of 44 ranked methods on the MPI Sintel flow benchmark in terms of endpoint error (EPE) in all, matched (Noc) and unmatched (Occ) regions. The results on the clean images are shown in (a). The results on the final set are shown in (b).

the flow with the smallest EPE at each pixel from our proposal set. Note that our approach uses only the refinement stages of [32,42] and not the DeepMatches (DM) which form the foundation for [32, 42]. As evidenced by this experiment, our method obtains state-of-the-art performance, outperforming [32, 42] in most of the error metrics. Furthermore, our proposal set contains flow vectors close to the ground truth for most of the pixels while being diverse enough to avoid local minima and reduce the computational complexity of the pairwise terms. Fig. 2 shows histograms of the frequency of neighboring flow proposals with respect to the distance of their endpoints for $L = 500$. While the pairwise term in Eq. 3 can take $L^2 = 250,000$ states in total, only 2% of them fall below the truncation value of 15 px and need to be evaluated in Eq. 6. Thus, pre-calculating the partitions $\mathcal{K}_{\mathbf{p},\mathbf{q},l}$ saves almost two orders of magnitude in computation time.

4.2 Quantitative Results

We also compare our method to the state-of-the-art on the challenging held-out test data of MPI Sintel [9] and KITTI [17]. As requested by the evaluation protocol, we only submitted our strongest entry ("Ours+EpicFlow") for evaluation

	Outliers (%)		EPE (px)	
	Noc	All	Noc	All
PH-Flow [48]	**5.76**	**10.57**	**1.3**	2.9
FlowFields	5.77	14.01	1.4	3.5
NLTGV-SC [30]	5.93	11.96	1.6	3.8
DDS-DF [41]	6.03	13.08	1.6	4.2
TGV2ADCSIFT [5]	6.20	15.15	1.5	4.5
Ours+EpicFlow	6.23	16.63	**1.3**	3.6
AnyFlow	6.37	15.80	1.5	4.3
BTF-ILLUM [12]	6.52	11.03	1.5	**2.8**
DeepFlow2	6.61	17.35	1.4	5.3
CRT-TGV	6.71	12.09	2.0	3.9
Data-Flow [40]	7.11	14.57	1.9	5.5
DM+DeepFlow [42]	7.22	17.79	1.5	5.8
RME	7.25	17.74	3.1	8.5
DM+EpicFlow [32]	7.88	17.08	1.5	3.8
TVL1-HOG [31]	7.91	18.90	2.0	6.1

(a) **Outlier Threshold: 3 px**

	Outliers (%)		EPE (px)	
	Noc	All	Noc	All
Ours+EpicFlow	**3.89**	12.46	**1.3**	3.6
PH-Flow [48]	3.93	**7.72**	**1.3**	2.9
FlowFields	3.95	10.21	1.4	3.5
DDS-DF [41]	4.41	10.41	1.6	4.2
NLTGV-SC [30]	4.50	9.42	1.6	3.8
AnyFlow	4.51	12.55	1.5	4.3
TGV2ADCSIFT [5]	4.60	12.17	1.5	4.5
BTF-ILLUM [12]	4.64	8.11	1.5	**2.8**
DeepFlow2	4.73	14.19	1.4	5.3
CRT-TGV	5.01	8.97	2.0	3.9
TVL1-HOG [31]	5.26	15.45	2.0	6.1
DM+DeepFlow [42]	5.31	14.69	1.5	5.8
Data-Flow [40]	5.34	11.72	1.9	5.5
DM+EpicFlow [32]	5.36	12.86	1.5	3.8
RME	5.49	14.70	3.1	8.5

(b) **Outlier Threshold: 5 px**

Table 3: **Evaluation on KITTI Test Set.** This table lists the top 15 out of 57 ranked methods on the KITTI optical flow benchmark in terms of outliers and endpoint error (EPE) in non-occluded and all regions. For comparability, only pure optical flow methods are shown, excluding motion stereo methods and techniques which use stereo information or more than two frames as input.

on the test set. Table 2 and Table 3 compare our results to the state-of-the-art in two-frame optical flow. On MPI Sintel we obtain rank 1 on the "clean" set and rank 2 on the "final" set out of 44 ranked methods in total. On KITTI our method ranks 6th (out of 57 submitted results) for an outlier threshold of 3 pixels and 1st when using a threshold of 5 pixels. Note that we have excluded motion stereo methods and methods using additional information from Table 3 for a fair comparison. For full results, we refer to the online leaderboards[2].

4.3 Qualitative Results

Fig. 3 depicts qualitative results of our method on a subset of MPI Sintel (top four images) and KITTI (bottom four images). As we require ground truth for visualizing the error maps we utilize the training sets for this purpose. We show the input image (left), our results in combination with EpicFlow postprocessing (middle) and the color-coded error map with respect to the ground truth flow field (right). At the top of each error map we specify the percentage of outliers and the endpoint error (EPE). The color coding visualizes outliers (> 3 px EPE) in red and inliers (< 3 px EPE) in blue on a logarithmic color scale. Pixels without ground truth value are shown in black. Note how our method is able to capture fine details in the optical flow. The last result of each dataset shows a failure case. For MPI Sintel, errors can be mostly attributed to homogeneous regions, occlusions and large changes in motion blur. On KITTI, we identified large perspective distortions, illumination artifacts and saturated or reflective regions as the primary sources of error.

[2] http://sintel.is.tue.mpg.de
http://www.cvlibs.net/datasets/kitti

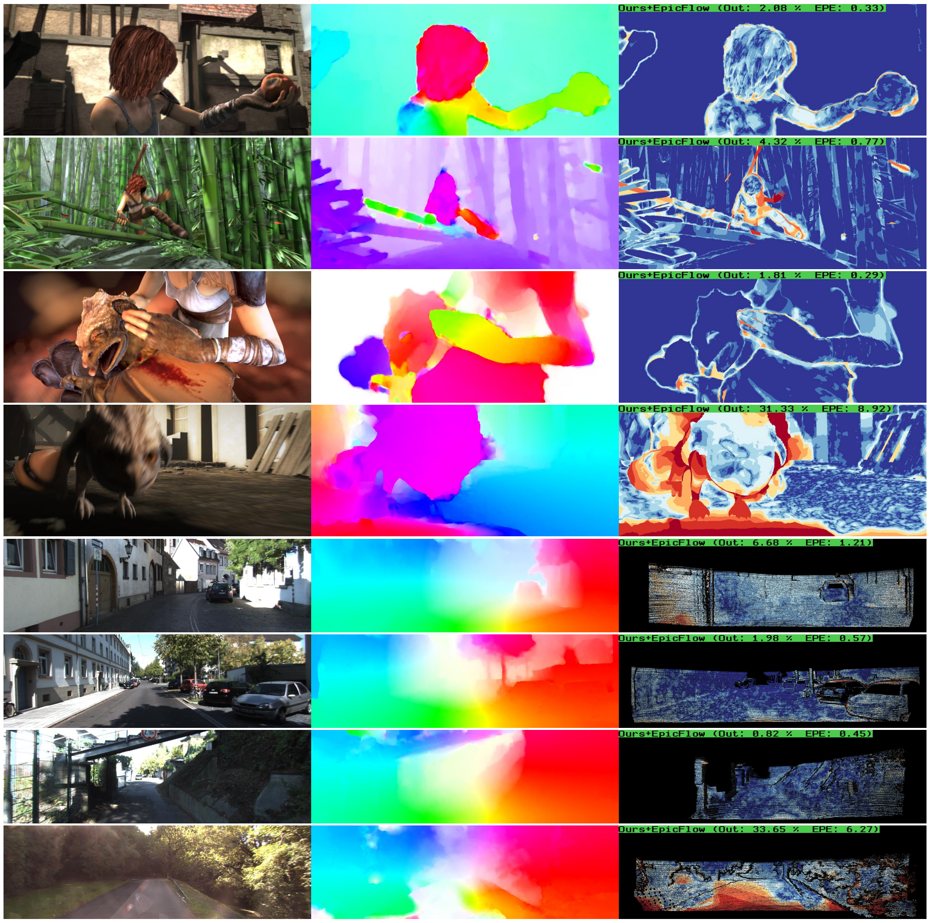

Ours+EpicFlow (Out: 2.08 % EPE: 0.33)
Ours+EpicFlow (Out: 4.32 % EPE: 0.77)
Ours+EpicFlow (Out: 1.81 % EPE: 0.29)
Ours+EpicFlow (Out: 31.33 % EPE: 8.92)
Ours+EpicFlow (Out: 6.68 % EPE: 1.21)
Ours+EpicFlow (Out: 1.98 % EPE: 0.57)
Ours+EpicFlow (Out: 0.82 % EPE: 0.45)
Ours+EpicFlow (Out: 33.65 % EPE: 6.27)

Fig. 3: **Qualitative Results on MPI Sintel and KITTI Training Sets.** From left-to-right: Input image, "Ours+EpicFlow", inliers (blue)/outliers (red).

5 Conclusions

We presented a discrete solution to the estimation of optical flow by exploiting three strategies which limit computational and memory requirements. Our experiments show that discrete optimization can lead to state-of-the-art optical flow results using a relatively simple prior model. In the future, we plan to integrate richer optical flow priors into our approach. In particular, we aim at jointly reasoning about several image scales and at incorporating semantic information [18, 41] into optical flow estimation. Furthermore, we plan to extend our approach to the estimation of 3D scene flow [26, 27].

References

1. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., Szeliski, R.: A database and evaluation methodology for optical flow. IJCV 92, 1–31 (2011)
2. Bao, L., Yang, Q., Jin, H.: Fast edge-preserving PatchMatch for large displacement optical flow. In: CVPR (2014)
3. Besse, F., Rother, C., Fitzgibbon, A., Kautz, J.: PMBP: PatchMatch Belief Propagation for correspondence field estimation. IJCV 110(1), 2–13 (2014)
4. Black, M.J., Anandan, P.: A framework for the robust estimation of optical flow. In: ICCV (1993)
5. Braux-Zin, J., Dupont, R., Bartoli, A.: A general dense image matching framework combining direct and feature-based costs. In: ICCV (2013)
6. Brox, T., Malik, J.: Large displacement optical flow: Descriptor matching in variational motion estimation. PAMI 33, 500–513 (March 2011)
7. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: ECCV (2004)
8. Bruhn, A., Weickert, J., Schnörr, C.: Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. IJCV 61(3), 211–231 (2005)
9. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: ECCV (2012)
10. Chen, Q., Koltun, V.: Fast MRF optimization with application to depth reconstruction. In: CVPR (2014)
11. Chen, Z., Jin, H., Lin, Z., Cohen, S., Wu, Y.: Large displacement optical flow from nearest neighbor fields. In: CVPR (2013)
12. Demetz, O., Stoll, M., Volz, S., Weickert, J., Bruhn, A.: Learning brightness transfer functions for the joint recovery of illumination changes and optical flow. In: ECCV (2014)
13. Dollár, P., Zitnick, C.L.: Structured forests for fast edge detection. In: ICCV. pp. 1841–1848 (2013)
14. Felzenszwalb, P., Huttenlocher, D.: Efficient belief propagation for early vision. IJCV 70(1), 41–54 (October 2006)
15. Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazirbas, C., Smagt, V.G.P., Cremers, D., Brox, T.: FlowNet: Learning optical flow with convolutional networks. arXiv.org 1504.06852 (2015)
16. Fortun, D., Bouthemy, P., Kervrann, C.: Aggregation of local parametric candidates and exemplar-based occlusion handling for optical flow. arXiv.org 1407.5759 (2014)
17. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR (2012)
18. Güney, F., Geiger, A.: Displets: Resolving stereo ambiguities using object knowledge. In: CVPR (2015)
19. Hirschmüller, H.: Stereo processing by semiglobal matching and mutual information. PAMI 30(2), 328–341 (2008)
20. Horn, B.K.P., Schunck, B.G.: Determining optical flow. AI 17(1-3), 185–203 (1980)
21. Hornacek, M., Besse, F., Kautz, J., Fitzgibbon, A.W., Rother, C.: Highly overparameterized optical flow using PatchMatch Belief Propagation. In: ECCV (2014)
22. Kennedy, R., Taylor, C.J.: Optical flow with geometric occlusion estimation and fusion of multiple frames. In: EMMCVPR (2014)
23. Lempitsky, V.S., Roth, S., Rother, C.: Fusionflow: Discrete-continuous optimization for optical flow estimation. In: CVPR (2008)

24. Liu, C., Yuen, J., Torralba, A.: SIFT flow: Dense correspondence across scenes and its applications. PAMI 33(5), 978–994 (2011)
25. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI (1981)
26. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: CVPR (2015)
27. Menze, M., Heipke, C., Geiger, A.: Joint 3d estimation of vehicles and scene flow. In: ISA (2015)
28. Mozerov, M.: Constrained optical flow estimation as a matching problem. TIP 22(5), 2044–2055 (May 2013)
29. Muja, M., Lowe, D.G.: Scalable nearest neighbor algorithms for high dimensional data. PAMI 36(11), 2227–2240 (2014)
30. Ranftl, R., Bredies, K., Pock, T.: Non-local total generalized variation for optical flow estimation. In: ECCV (2014)
31. Rashwan, H.A., Mohamed, M.A., García, M.A., Mertsching, B., Puig, D.: Illumination robust optical flow model based on histogram of oriented gradients. In: GCPR (2013)
32. Revaud, J., Weinzaepfel, P., Harchaoui, Z., Schmid, C.: EpicFlow: Edge-preserving interpolation of correspondences for optical flow. In: CVPR (2015)
33. Rhemann, C., Hosni, A., Bleyer, M., Rother, C., Gelautz, M.: Fast cost-volume filtering for visual correspondence and beyond. In: CVPR (2011)
34. Roth, S., Black, M.J.: On the spatial statistics of optical flow. IJCV 74(1), 33–50 (2007)
35. Sevilla-Lara, L., Sun, D., Learned-Miller, E.G., Black, M.J.: Optical flow estimation with channel constancy. In: ECCV (2014)
36. Steinbrücker, F., Pock, T., Cremers, D.: Large displacement optical flow computation without warping. In: ICCV. pp. 1609–1614 (2009)
37. Sun, D., Roth, S., Black, M.J.: A quantitative analysis of current practices in optical flow estimation and the principles behind them. IJCV 106(2), 115–137 (2013)
38. Timofte, R., Gool, L.V.: Sparse flow: Sparse matching for small to large displacement optical flow. In: WACV (2015)
39. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide baseline stereo. PAMI 32(5), 815–830 (May 2010)
40. Vogel, C., Roth, S., Schindler, K.: An evaluation of data costs for optical flow. In: GCPR (2013)
41. Wei, D., Liu, C., Freeman, W.: A data-driven regularization model for stereo and flow. In: 3DV (2014)
42. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: DeepFlow: Large displacement optical flow with deep matching. In: ICCV (2013)
43. Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D., Bischof, H.: Anisotropic Huber-L1 optical flow. In: BMVC (2009)
44. Wulff, J., Black, M.J.: Efficient sparse-to-dense optical flow estimation using a learned basis and layers. In: CVPR (2015)
45. Xu, L., Jia, J., Matsushita, Y.: Motion detail preserving optical flow estimation. PAMI 34(9), 1744–1757 (2012)
46. Yamaguchi, K., McAllester, D., Urtasun, R.: Robust monocular epipolar flow estimation. In: CVPR (2013)
47. Yang, H., Lin, W., Lu, J.: DAISY filter flow: A generalized discrete approach to dense correspondences. In: CVPR (2014)
48. Yang, J., Li, H.: Dense, accurate optical flow estimation with piecewise parametric model. In: CVPR (2015)