A Prediction-Correction Approach for Real-Time Optical Flow Computation Using Stereo

Maxime Derome(✉), Aurelien Plyer, Martial Sanfourche, and Guy Le Besnerais

ONERA - The French Aerospace Lab, Palaiseau, France
{maxime.derome,aurelien.plyer,martial.sanfourche, guy.le_besnerais}@onera.fr

Abstract. Estimating the optical flow robustly in real-time is still a challenging issue as revealed by current KITTI benchmarks. We propose an original two-step method for fast and performant optical flow estimation from stereo vision. The first step is the prediction of the flow due to the ego-motion, efficiently conducted by stereo-matching and visual odometry. The correction step estimates the motion of mobile objects. Algorithmic choices are justified by empirical studies on real datasets. Our method achieves framerate processing on images of realistic size, and provides results comparable or better than methods having computation times one or two orders of magnitude higher.

1 Introduction

Estimating the dense temporal matching between two consecutive images —also called *Optical Flow* (OF)— is a well know problem in computer vision. From a single camera, it provides a clue on the motion and the 3D structure of the scene. From a stereorig, although the 3D structure can be directly estimated by stereo-matching, computing the OF is still necessary to provide dynamic scene perception and estimate moving objects displacement. In this context, estimating OF together with depth maps allows to compute the 3D motion field—called *Scene Flow* (SF), which essentially summarizes all the information about ego-motion, 3D structure and moving objects.

Recent works on stereo matching have lead to both accurate and fast algorithms [7,9,10,19]. As for the OF estimation, numerous progress have been made concerning robustness to outliers and illumination change [22] and large displacements: through multi-scale or pyramidal approaches [1] or using feature matching to drive the OF estimation [2,23]. However, compared to the significant efforts made to improve the accuracy of OF, we feel that little attention has been paid to the real-time capability of OF algorithms [16,25]. If we look at the KITTI ranking for OF [8], we notice that the few algorithms that achieve video rate computation give significantly lower performances than top-ranking methods. We also observe that these top-ranking methods make use of stereo information [14,20] or epipolar geometry [24].

B. Rosenhahn and B. Andres (Eds.): GCPR 2016, LNCS 9796, pp. 365–376, 2016.
DOI: 10.1007/978-3-319-45886-1_30

In fact, most of these algorithms estimate the SF and use an advanced modelization of the scene [20] or of the objects [14] to improve the accuracy of the results. Such approaches achieve leading performances but are computationally expensive and require several minutes to perform. Algorithms proposed in [24] perform one or two orders of magnitude faster but they are still too slow for real-time application on KITTI's image dimensions and, moreover, they rely on the assumption of a static scene.

1.1 Contributions

In contrast with previous approaches that use stereo, our main purpose is not to improve the OF's accuracy but to reach the shortest processing time while keeping a good estimation performance. Our approach is in the line to the work of Wedel et al. [21] which decouples the stereo and the OF problems and first estimates the disparity map to bootstrap SF estimation. Here we intend to benefit as much as possible from the geometric information we can get from a stereorig, and go even further in the decoupling by using a visual odometer to estimate the camera motion. This can be related to Muller et al. [15] which uses ego-motion information obtained from an inertial sensor to support the OF estimation and avoid the *staircase effects* induced by the total variation regularization. But, again, our goal here is to reduce computation time, not improve OF estimation.

Note that, although we consider a stereovision context, we focus on OF rather than SF estimation. Let us first note that having estimated the OF and knowing the depth maps at the two time instants, SF computation essentially reduces to a change of coordinate frame. As shown by the literature review, the main computational bottleneck then lies in the OF estimation, which motivates the present work. Besides, there are reactive tasks in vision-based robotics which can exploit information deduced from the OF without calling for SF estimation. A sparse representation of the scene, where moving objects are reduced to a bounding box with a mean 3D velocity, can be sufficient for designing an obstacle avoidance system, and requires much less computation than the full SF derivation.

Our contribution is two-fold. First, we present a simple prediction-correction approach that exploits geometric information obtained by grayscale stereo images to improve the OF estimation. Second, by carefully selecting the best trade-off between computational efficiency and performance for the algorithms involved in each step (stereo-matching, visual odometry, optical flow), we demonstrate framerate processing (10 Hz) while achieving good results on KITTI datasets.

2 Prediction-Correction of Computation in Stereo

The main idea behind our approach is to split the OF estimation into two steps: (i) estimate the static part of the OF, which amounts to *predict* the optical flow between two time instants from 3D structure and camera motion; (ii) *correct* the OF to account for moving objects.

The static part of the OF can be deduced from the disparity map and the ego-motion respectively computed by stereo and visual odometry algorithms. Thanks to the epipolar constraint, the estimation of disparity with calibrated stereo images boils down to a 1D search whereas OF requires a 2D search. Besides, unlike OF algorithms, stereo algorithms handle well boundaries. Recovering the ego-motion by visual odometry is a 6-parameters estimation problem that can be solved efficiently even in presence of moving objects, using a robust estimation procedure such as RANSAC [6].

The correction step mainly aims at estimating the part of the OF due to moving objects. It also corrects errors in the depth map propagated by the prediction step. In many situations, the associated residual OF is of low magnitude and good results can be obtained from fast local OF estimation algorithms. Local problems can be encountered when dealing with fast moving objects, though.

Before presenting the prediction and the correction stages of the proposed Prediction-Correction OF (PCOF) approach, we briefly define the notations used in this paper.

2.1 Convention and Notations

We consider the pinhole camera model for the left camera frame which is taken as reference.

- I_t, left camera (grayscale) image at time instant t
- f, camera focal in pixel
- b, stereorig baseline
- (x, y), pixel location
- (x_0, y_0), coordinates of the optical center projection into the image plane
- (u, v), optical flow between I_t and I_{t+1}
- d_t, disparity map at time instant t
- X_t, 3D coordinates in (left) camera frame at time instant t
- $X_t(x, y, d) = -\frac{b}{d(x,y)}\,(x - x_0, y - y_0, f)$, the triangulation function at t
- Π, the projection operator which maps a 3D point to the image plan
- R and T, the camera rotation and translation between t and $t+1$ such that $X_{t+1} = RX_t + T$ for a static point.

2.2 Prediction Stage

The predicted OF (u_{pred}, v_{pred}) is computed by assuming that the scene static between t and $t+1$:

$$\begin{pmatrix} u_{pred} \\ v_{pred} \end{pmatrix}(x, y) = \xi_{t+1}^{pred}(x, y, d_t) - \begin{pmatrix} x \\ y \end{pmatrix}, \tag{1}$$

where $\xi_{t+1}^{pred}(x, y, d_t)$ is the predicted position in I_{t+1}, of a point located in (x, y) in I_t:

$$\xi_{t+1}^{pred}(x, y, d_t) = \Pi\big(RX_t(x, y, d_t) + T\big). \tag{2}$$

From the predicted optical flow, one can predict the image that should be observed if there were no moving objects in the scene [5]. This can be done from t to $t+1$ (forward scheme) or from $t+1$ to t (backward scheme). As we intend to estimate the OF in the geometry of image t, we adopt, as in [4], a backward scheme and use $\xi_{t+1}^{pred}(x, y, d_t)$ to warp I_{t+1} into frame t. Another advantage of the backward scheme is that the predicted image I_t^{pred} can be synthesized by a simple interpolation starting from a regular grid

$$I_t^{pred}(x, y) = I_{t+1}(\xi_{t+1}^{pred}(x, y, d_t)). \tag{3}$$

while the forward scheme leads to an irregular interpolation.

2.3 Correction Stage

For the sake of comparison we present two different correction strategies that respectively use the predicted OF (u_{pred}, v_{pred}) and the predicted image I_t^{pred}.

Prediction Warping Optical Flow (PWOF). A natural way to compensate the camera motion is to initialize an OF algorithm with (u_{pred}, v_{pred}) and run it on one pyramid level only. This strategy is similar to the one of Revaud et al. [17], who initialize a TV-L1 OF algorithm with a flow obtained by the interpolation of sparse set of matches. Here the initial warping corresponds to the static part of the optical flow, hence in a non-pyramidal approach, the motion of fast moving objects might be difficult to estimate.

Prediction Correction Optical Flow (PCOF). Another option is to compute the OF $(\delta u, \delta v)$ between I_t and I_t^{pred}. In this context the brightness consistency equation writes:

$$I_t(x, y) = I_t^{pred}(x + \delta u(x, y), y + \delta v(x, y)). \tag{4}$$

By definition (cf. Eq. 3), I_t^{pred} writes:

$$I_t^{pred}(x, y) = I_{t+1}(x + u_{pred}(x, y), y + v_{pred}(x, y)) \tag{5}$$

Thus, by combining Eqs. 4 and 5, we obtain the final OF estimate (u,v) after correction:

$$u(x, y) = \delta u(x, y) + u_{pred}(x + \delta u(x, y), y + \delta v(x, y)) \tag{6}$$

$$v(x, y) = \delta v(x, y) + v_{pred}(x + \delta u(x, y), y + \delta v(x, y)) \tag{7}$$

3 Algorithmic Choices: Odometry, Stereo and OF

In this section we present the various algorithms which have been considered and justify our choices by an empirical performance/cost study on KITTI datasets.

3.1 Ego-Motion Estimation

To estimate the camera motion from stereo images, we choose the visual odometer eVO proposed in [18]. Given 2D-3D matchings $\{[x_{t+1}^n, y_{t+1}^n], X_t^n\}$, it minimizes the following reprojection error:

$$\mathcal{E}(R,T) = \frac{1}{N}\sum_{n=1}^{N}\left|\left|\begin{bmatrix} x_{t+1}^n \\ y_{t+1}^n \end{bmatrix} - \Pi(RX_t^n + T)\right|\right|^2. \tag{8}$$

In [18], feature points extraction and 3D triangulation is done only at keyframes, so as to reduce computation load and reduce drift. Here we only consider two consecutive stereo pairs, so the feature points processing is performed at each frame. Despite this setting, the camera motion can be estimated at 20 Hz on a single CPU. In practice, the optimization of (8) is done within the RANSAC framework [6] to ensure the system robustness in presence of moving objects as long as the major part of the image is covered by static elements.

3.2 Dense Stereo Matching

Dense stereo matching, or dense disparity estimation in our rectified stereo context, has been extensively studied and several fast algorithms are available. They include Semi-Global Matching (SGM) [10] whose implementation on FPGA [7] allows stereo matching at 25 Hz (for 128 disparities and images of 740 × 480 pixels), Efficient Large-scale Stereo (ELAS) [9] running at 20 Hz on a CPU (for KITTI size images, when enabling sub-sampling option) and Adaptive Coarse-to-fine-stereo (ACTF) [19] that reaches 32 Hz on a 240 cores GPU (for 256 disparities and images of 640 × 480 pixels). We choose the last one which offers a remarkable trade-off between speed and accuracy, and is particularly efficient for preserving the boundaries of foreground objects. Yet, as SGM is considered as a reference method in stereovision, we also consider it in Subsect. 4.1 for comparisons purpose — actually, we use the Semi-Global Bloc Matching (SGBM) which is the OpenCV version of SGM that differs by its cost function. Because of the uniquess criterion enforced by the left-right checking, some parts of the disparity map might remain undefined. As done in [19], we fill these parts with the nearest valid pixel disparity found by an horizontal and vertical search.

Regardless of the stereo algorithm used, the disparity map is then regularized by performing the first iteration of the consensus framework introduced in [3]. It essentially fits plane models on the disparity over small overlapping windows—we use 16 × 16 and 32 × 32 overlapping windows. This smoothing preserves the boundaries quite well, while efficiently filtering the map and rejecting outliers. Besides, this algorithm is highly parallelizable and the gain brought seems to worth the small computational overhead.

3.3 Optical Flow

In the following, we consider High Accuracy Optical Flow (HAOF) [1], TV-L1 optical flow [25] and Large Displacement Optical Flow (LDOF) [2] algorithms, that are often cited as references among variational methods derived from the seminal work of Horn and Schunck [11]. We also consider FOLKI [12], an OF algorithm based on Lucas-Kanade paradigm [13], that achieves real-time performances on a GPU [16]. In our experiments, we use LDOF and OpenCV version of HAOF and TV-L1 with default parameters, and FOLKI with: $J = 5$ resolution levels, $K = 4$ iterations, two window radii {8;4} and rank order 4 (more explanations about these parameters can be found in [16]).

4 Results on KITTI Datasets and Processing Time

For the evaluation, we have considered KITTI 2012 [8] and KITTI 2015 [14] datasets, which contain stereo images acquired from a moving car in urban environment. The 2012 dataset presents static scenes, whereas the 2015 dataset focuses on moving objects.

4.1 Comparison of the Correction Approaches and OF Algorithms

Experiments on KITTI 2012 dataset are summarized in Table 1. We compare the performances of both PWOF and PCOF strategies described in Subsect. 2.3, while testing different OF algorithms: FOLKI, HOAF and TV-L1. In all cases, as explained in Subsect. 3.2, we choose ACTF stereo followed by a filtering step, for the dense stereo matching used in the prediction stage.

Table 1 first demonstrates that the PCOF strategy, based on OF estimation on a predicted image, performs equally or better than PWOF whatever the OF method. An illustration is given in Fig. 1 in a case where the prediction is particularly inaccurate because of a visual odometry error. Leading performances are achieved with PCOF and FOLKI optical flow. This might appear surprising at first, since local methods like FOLKI are reputed to be less accurate than variational methods like HAOF and TV-L1. However FOLKI is very robust and converges faster than HOAF and TV-L1. Moreover the higher performance of the latter methods to adaptively smooth the OF are not very relevant here because the sought residual flow is often rather inhomogeneous. Finally the last line of Table 1 shows that using SGBM instead of ACTF stereo slightly increases the results, but at the expense of a computational overhead (cf. Subsect. 4.3).

4.2 Advantages of the Prediction-Correction Approach

Static Environments. In the case of a static environment, Table 1 demonstrates that the prediction step, which exploits ego-motion and the dense stereo matching computation, already performs far better than FOLKI OF. This is partly due to the fact that object boundaries are over-smoothed by window-based OF methods such as FOLKI. Moreover, despite the static environment, the correction stage brings a significant improvement.

Table 1. Optical flow performances on KITTI - 2012 training set in term of percentage of outliers (*Out-*) and average error in pixels (*Avg-*) on the whole image (*-All*) and on the non-occluded area (*-Noc*)

Optical flow	Out-Noc	Out-All	Avg-Noc	Avg-All
FOLKI	21.95 %	31.01 %	5.5 px	11.2 px
Predicted OF	6.70 %	9.85 %	1.2 px	1.7 px
PWOF - HAOF	5.96 %	9.18 %	1.1 px	1.6 px
PWOF - TV-L1	6.66 %	15.59 %	1.2 px	2.3 px
PWOF - FOLKI	8.03 %	11.79 %	1.6 px	2.2 px
PCOF - HAOF	5.92 %	9.18 %	1.1 px	1.6 px
PCOF - TV-L1	7.39 %	10.42 %	1.3 px	1.8 px
PCOF - FOLKI	5.43 %	8.77 %	1.1 px	1.6 px
PCOF-SGBM - FOLKI	5.27 %	8.08 %	1.1 px	1.8 px

Dynamic Environments. In presence of moving objects, the correction stage becomes necessary to estimate the OF which cannot be deduced only from the camera motion and the scene geometry, see Table 2. Figure 2 presents two examples from KITTI 2015 dataset with moving cars. The prediction fails to estimate the OF on moving objects which is provided by the correction step. PCOF significantly improves FOLKI results, but still encounters difficulties with very large motions (cf. Fig. 2, second column) for which LDOF [2] is more suitable (cf. Table 2) but leads to an important computational overhead (cf. Table 5).

Table 2. Optical flow performances improvement brought by the PCOF approach, on KITTI - 2015 training set. Percentages of OF outliers, on the whole image (-all), on the background (e.g. static part, -bg) and the foreground (i.e. on moving objects, -fg)

Optical flow	Fl-bg	Fl-fg	Fl-all
FOLKI	40.10 %	55.53 %	44.57 %
Predicted OF	17.53 %	86.69 %	29.46 %
PCOF	14.46 %	55.55 %	22.62 %
PCOF - LDOF	13.97 %	29.62 %	18.16 %

4.3 Processing Time

Algorithms on CPU. Tests have been done on a workstation with 6 CPUs (Interl Core i7 at 3.2GHz). With 400 feature points, the visual odometer [18] runs in 50 ms using a single CPU. Most of the processing time is spent in the feature points extraction and could be significantly speeded up by parallelization. SGBM performs in 730 ms (OpenCV implementation) and appears unfit for framerate processing.

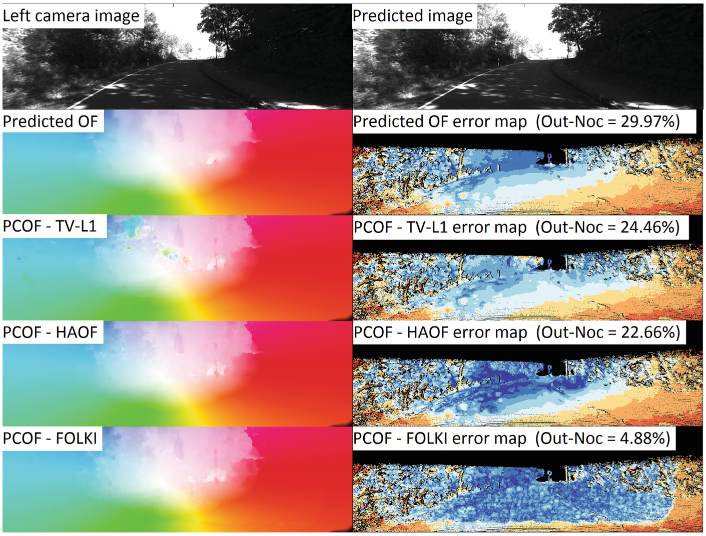

Fig. 1. First row: left camera image (left) and predicted image (right). From second to fifth row: OF (left) and error map (right) displayed using KITTI color convention, for the predicted OF, and PCOF using TV-L1, HAOF, and FOLKI. Blue corresponds to low error on the OF estimation, and orange to high error. *Out-Noc* is also displayed. (Color figure online)

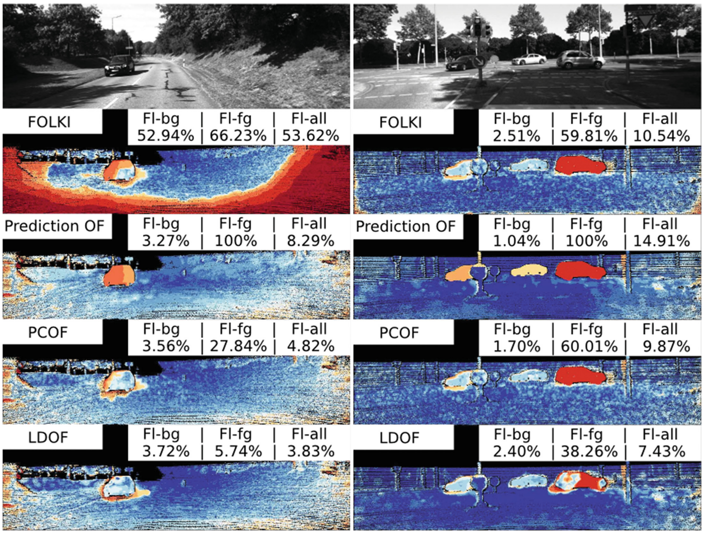

Fig. 2. Error on the estimated OF fields (blue corresponds to low error, and red to high error), and percentage of outliers (i.e. points for which the OF end-point error is $> 3\,px$ and $> 5\,\%$) for the optical flow (*Fl-all*, *Fl-bg* and *Fl-fg*) on the whole image but also on the background (e.g. static part) and the foreground (i.e. on moving objects). (Color figure online)

Table 3. Average processing time (in ms) on KITTI dataset, with two different GPUs, for each step of our pipeline and for different optical flow algorithms with and without (in brackets) pyramidal approach

	ACTF + filtering	Prediction	FOLKI	HOAF	TV-L1
GTX TITAN (2688 cores)	11 + 7	1	27 (15)	127 (31)	128 (226)
Quadro 2000 (192 cores)	50 + 31	1	150 (113)	843 (283)	703 (1401)

Algorithms on GPU. Except for the ego-motion estimation, every step of PCOF can be done on GPU. We provide in Table 3 runtimes obtained on a high performance GPU (GeForce GTX TITAN) and a standard one (Quadro 2000). Among the OF algorithms tested with PCOF, only FOLKI has proven to be fast enough for framerate processing (10 Hz). HAOF can also be considered using PWOF (i.e. no image pyramid) on the TITAN.

Note that the stereo part (ACTF and consensus filtering) runs on the GPU, in parallel with the visual odometry performed on the CPU in 50 ms. Thus PCOF processing time is around 80 ms with the GeForce GTX TITAN, which fits framerate processing on KITTI sequences (10 Hz).

Table 4. Extract of KITTI 2012 OF ranking, with methods using stereo image (st), epipolar geometry (ms) or more than two temporally adjacent images (mv)

	Method	Setting	Out-Noc	Out-All	Avg-Noc	Avg-All	Runtime
1	PRSM	st mv	2.46 %	4.23 %	0.7 px	1.0 px	300 s
...	...	...	...	...	...	...	...
4	SPS-Fl	ms	3.38 %	10.06 %	0.9 px	2.9 px	11 s
...	...	...	...	...	...	...	...
9	MotionSLIC	ms	3.91 %	10.56 %	0.9 px	2.7 px	11 s
10	CNN-HPM		4.89 %	13.01 %	1.2 px	3.0 px	23 s
...	...	...	...	...	...	...	...
13	PCOF-SGBM	st	5.40 %	8.73 %	1.2 px	2.1 px	0.8
14	PPM-Fast		5.41 %	15.19 %	1.2 px	3.4 px	2.8 s
15	PCOF	st	5.59 %	9.69 %	1.2 px	1.9 px	0.08 s
...	...	...	...	...	...	...	...
49	eFolki		19.31 %	28.79 %	5.2 px	10.9 px	0.026 s
...	...	...	...	...	...	...	...
58	FlowNetS + ft		37.05 %	44.49 %	5.0 px	9.1 px	0.08 s

4.4 KITTI 2012 and 2015 Optical Flow Rankings

Looking at KITTI 2012 and 2015 Optical Flow benchmarks (Tables 4 and 5), we can conclude that the stereo information pays off. On one hand, the proposed approach performs favourably compared to pure OF algorithms. On the

Table 5. Extract of KITTI 2015 OF ranking, with methods using stereo image (st), epipolar geometry (ms) or more than two temporally adjacent images (mv)

	Method	Data	Fl-bg	Fl-fg	Fl-all	Time
1	PRSM	st mv	5.33 %	17.02 %	7.28 %	300 s
...	...	...	...	...	...	...
8	PCOF-LDOF	st	14.34 %	41.30 %	18.83 %	50 s
9	CNN-HPM		18.33 %	24.96 %	19.44 %	23 s
10	MR-Flow		19.42 %	27.65 %	20.79 %	12 s
...	...	...	...	...	...	...
14	PCOF + ACTF	st	14.89 %	62.42 %	22.80 %	0.08 s
15	CPM-Flow		22.32 %	27.79 %	23.23 %	4.2 s
16	MotionSLIC	ms	14.86 %	66.21 %	23.40 %	30 s

other hand, although our approach gives a higher percentage of outliers than SF algorithms that also use stereo information (noted *st*), it runs several order of magnitude faster. Finally, let us emphasize that PCOF is among the only three algorithms able to achieve framerate processing (10 Hz) on the KITTI datasets.

As for the KITTI 2015 ranking, the percentage of OF outliers (Fl-all) is significantly higher, mainly due to the large motions induced by fast moving objects. Using LDOF instead of FOLKI significantly improves the performances in this case, but at the expense of an important computational overhead (cf. PCOF-LDOF and PCOF + ACTF in Table 5).

5 Conclusion

We have proposed to fully exploit stereo information for fast and performant OF estimation. In the prediction step, the static part of the OF is derived from depth and visual odometry. The correction step improves the estimation and estimates the OF associated with moving objects, with corrrect performance except for the fastest ones. Relying on relevant algorithmic choices, we have been able to reach framerate (10 Hz) processing while maintaining high quality OF estimation, as shown by a performance evaluation on KITTI datasets. On-going work concerns the integration of covariance propagation in the prediction/correction process and the application of the proposed pipeline to an obstacle avoidance system for lightweight ground and aerial robots.

Acknowledgment. This work was sponsored by the Direction Générale de l'Armement (DGA) of the French Ministry of Defense.

References

1. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J.G. (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
2. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variational motion estimation. PAMI **33**(3), 500–513 (2001)
3. Chakrabarti, A., Xiong, Y., Gortler, S.J., Zickler, T.: Low-level vision by consensus in a spatial hierarchy of regions. In: CVPR (2015)
4. Derome, M., Plyer, A., Sanfourche, M., Le Besnerais, G.: Real-time mobile object detection using stereo. In: ICARCV (2014)
5. Derome, M., Plyer, A., Sanfourche, M., Le Besnerais, G.: Moving object detection in real-time using stereo from a mobile platform. Unmanned Syst. **3**(4), 253–266 (2015)
6. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. ACM **24**(6), 381–395 (1981)
7. Gehrig, S.K., Eberli, F., Meyer, T.: A real-time low-power stereo vision engine using semi-global matching. In: Fritz, M., Schiele, B., Piater, J.H. (eds.) ICVS 2009. LNCS, vol. 5815, pp. 134–143. Springer, Heidelberg (2009)
8. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. Int. J. Robot. Res. **32**(11), 1231–1237 (2013)
9. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 25–38. Springer, Heidelberg (2011)
10. Hirschmller, H.: Stereo processing by semiglobal matching and mutual information. Pattern Anal. Mach. Intell. **30**(2), 328–341 (2008)
11. Horn, B.K., Schunck, B.G.: Determining optical flow. Artif. Intell. **17**, 185–203 (1981)
12. Le Besnerais, G., Champagnat, F.: Dense optical flow by iterative local window registration. In: ICIP (2005)
13. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI, pp. 674–679 (1981)
14. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: CVPR (2015)
15. Müller, T., Rannacher, J., Rabe, C., Franke, U.: Feature- and depth-supported modified total variation optical flow for 3D motion field estimation in real scenes. In: CVPR (2011)
16. Plyer, A., Le Besnerais, G., Champagnat, F.: Massively parallel Lucas Kanade optical flow for real-time video processing applications. J. Real-Time Image Proc. **11**(4), 713–730 (2016)
17. Revaud, J., Weinzaepfel, P., Harchaoui, Z., Schmid, C.: EpicFlow: edge-preserving interpolation of correspondences for optical flow. In: CVPR (2015)
18. Sanfourche, M., Vittori, V., Le Besnerais, G.: eVO: a realtime embedded stereo odometry for MAV applications. In: IROS (2013)
19. Sizintsev, M., Kuthirummal, S., Samarasekera, S., Kumar, R., Sawhney, H.S., Chaudhry, A.: GPU accelerated realtime stereo for augmented reality. In: 3DPVT (2010)
20. Vogel, C., Schindler, K., Roth, S.: 3D scene flow estimation with a piecewise rigid scene model. Int. J. Comput. Vis. **115**, 1 (2015)

21. Wedel, A., Brox, T., Vaudrey, T., Rabe, C., Franke, U., Cremers, D.: Stereoscopic scene flow computation for 3D motion understanding. Int. J. Comput. Vis. **95**(1), 29–51 (2011)
22. Wedel, A., Pock, T., Zach, C., Bischof, H., Cremers, D.: An improved algorithm for TV-L^1 optical flow. In: Cremers, D., Rosenhahn, B., Yuille, A.L., Schmidt, F.R. (eds.) Statistical and Geometrical Approaches to Visual Motion Analysis. LNCS, vol. 5604, pp. 23–45. Springer, Heidelberg (2009)
23. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Deepflow: large displacement optical flow with deep matching. In: ICCV (2013)
24. Yamaguchi, K., McAllester, D., Urtasun, R.: Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 756–771. Springer, Heidelberg (2014)
25. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-L^1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 214–223. Springer, Heidelberg (2007)