Robust Local Optical Flow: Dense Motion Vector Field Interpolation

Jonas Geistert
Communication Systems Group
Technische Universität Berlin
Berlin, Germany
geistert@nue.tu-berlin.de

Tobias Senst
Communication Systems Group
Technische Universität Berlin
Berlin, Germany
senst@nue.tu-berlin.de

Thomas Sikora
Communication Systems Group
Technische Universität Berlin
Berlin, Germany
sikora@nue.tu-berlin.de

***Abstract*—Optical flow methods integrating sparse point correspondences have made significant contribution in the field of optical flow estimation. Especially for the goal of estimating motion accurately and efficiently, sparse-to-dense interpolation schemes for feature point matches have shown outstanding performances. Concurrently, local optical flow methods have been significantly improved with respect to long-range motion estimation in environments with varying illumination. This motivates us to propose a sparse-to-dense approach based on the Robust Local Optical Flow method. Compared to state-of-the-art methods the proposed approach is significantly faster while retaining competitive accuracy on Middlebury, KITTI 2015 and MPI-Sintel data-set.**

I. Introduction

Optical flow estimation is an important component of state-of-the-art computer vision tools for processing video data. Especially in the field of global optical flow methods there is an ongoing progress that leads to new innovative solutions with high accuracies [1]. With the introduction of the Total Variation approach by Brox *et al.* [2] the optical flow based methods have been proven to be a reliable tool for estimating dense motion vector fields.

Nowadays approaches successfully cope with the weakness of classical methods with respect to preservation of motion discontinuities e.g. by layers [3], changing illumination conditions e.g. by high-order constancy assumptions [4] or long-range motion usually by multi-scale approaches. In general global optical flow methods generate highly accurate motion fields due to a global spatial coherence. This poses the problem of motion estimation as the optimization of a global energy function as it takes the whole image data into account. The main disadvantage of the global approach is that solving this functional is computationally expensive and thus rather slow.

However, dealing with long-range motion to a large extent is still an open problem. A major drawback of the multi-scale schemes is the error-propagation from the coarsest to the finest scale. Recent approaches for handling large displacements and avoiding the error propagation are optical flow methods that are based on integration of feature point matching. Brox *et al.* extend the variational framework based on Histogram of Oriented Gradient (HOG) descriptors and Weinzaepfel *et al.* [5] integrate the local information of the point correspondence by extending the global energy term with a matching term. However, while their computational complexity is still high, they reveal the potential of local point matching techniques.

Recent approaches take advantage of sparse-to-dense interpolation schemes for feature matches. Revaud [6] proposed an edge-preserving interpolation filter to retrieve the dense motion fields from sparse DeepMatches [5]. With 15 seconds per frame on the KITTI data-set sequences the run-time is still too high for many computer vision applications. The potential of this technique to efficiently compute dense motion fields has motivated the work of Wulff and Black, who proposed in [7] to reconstruct dense motion fields from sparse fields with a highly efficient feature point matching method [8]. With 3.2 seconds per frame on the KITTI data-set sequences this approach significantly improves the performance w.r.t. accuracy and run-time.

In our previous work, we focused on the improvement of the accuracy and run-time performance of local optical flow methods. Knowing that in general the accuracy of global optical flow methods is higher than that of local methods when comparing dense motion fields, we focused on the fields of application that apply sparse motion information e.g. tracking, video-based surveillance or video coding. The ability of performing well at low run-time accounts for the important role of local methods in these domains as they are scalable with respect to the number of motion vectors to be estimated. In recent work, we proposed the Robust Local Optical Flow (RLOF) [9] as a robust derivative of the well-known pyramidal Lucas Kanade method [10]. RLOF has been further improved with respect to the preservation of motion discontinuities [11], the improvement of the computational efficiency [12] and the enhancement of the robustness for environments containing changing illuminations and long-range motions [13]. In [12] it has been shown that for the evaluation of sparse motion fields the RLOF method is competitive to state-of-the-art global ones.

The recent developments on local optical flow methods and sparse-to-dense interpolation schemes motivate us to propose a novel sparse-to-dense interpolation scheme based on sparse sets of motion vectors estimated by the RLOF feature tracking method.

II. Robust Local Optical Flow

This section describes the RLOF method. The Robust Local Optical Flow (RLOF) method is a gradient-based local optical flow method. The goal of this method is to estimate a set $S = \{\mathbf{d}_0, \ldots, \mathbf{d}_{n-1}\}$ of n motion vectors $\mathbf{d} \in \mathbb{R}^2$ at defined locations $P = \{\mathbf{x}_0, \ldots, \mathbf{x}_{n-1}\}$ with $\mathbf{x} \in \mathbb{R}^2$ with a motion estimation function $f : P \rightarrow S$ based on two consecutive images. In general the motion vector $\mathbf{d}$ is fraction of the parameter vector $\mathbf{p}$ that solves the following minimization problem:

$$\min_{\mathbf{p}} \sum_{\mathbf{x}\in\Omega} w(\mathbf{x}) \cdot \rho\left(\tilde{g}(\mathbf{x},\mathbf{p}), \sigma\right), \tag{1}$$

where ρ denotes an arbitrary norm, Ω a local support region around $\mathbf{x}$, σ its scale parameters, $w(\mathbf{x})$ a prior weighting function and $\tilde{g}$ the first-order Taylor approximation of an appearance model function g. The intensity constancy assumption (ICA) proposed by Horn and Schunk [14] is the most common appearance model. A prominent example is the pyramidal Lucas Kanade [10] as a part of the well-known Kanade-Lucas-Tomasi (KLT) tracker [15]. However the assumption of constancy of the consecutive intensity values $I(\mathbf{x}, t)$ and $I(\mathbf{x}+\mathbf{d}, t+1)$ almost never holds on real-world video footage. Therefore we have proposed in [13] a RLOF version based on a Gennert and Negahdaripour linear illumination model [16], which showed significant improvements in the presence of varying illuminations. For the RLOF the motion vector $\mathbf{d}$ is a solution of the following minimization problem:

$$\min_{[\mathbf{d}\ m\ c]} \sum_{\mathbf{x}\in\Omega} w(\mathbf{x}) \cdot \rho\left(T(\mathbf{x}) \cdot \begin{bmatrix} \mathbf{d} \\ m \\ c \end{bmatrix} + I_t(\mathbf{x}), \sigma\right) \tag{2}$$

with $T(\mathbf{x}) = [\nabla I(\mathbf{x})\ \ -I(\mathbf{x})\ \ -1]^T$, $\nabla I(\mathbf{x})$ the spatial and $I_t(\mathbf{x})$ the temporal image gradients, ρ the shrinked Hampel norm as introduced in [9] and $[\mathbf{d}\ m\ c]^T$ a parameter vector with m a multiplicative and c an additive illumination parameter. If $\rho(x) = x^2$, $w(x) = 1$ and $m = c = 0$ then Eq. 2 describes the Lucas Kanade formulation [17].

To handle long-range motions the multi-scale coarse-to-fine scheme that starts from the top level of a pair of image pyramids built on repeatedly low-pass filtered and down-sampled images and propagates interim motion results from the coarse to the fine level until the finest level is reached.

To cope with small linearization errors, an iterative Newton-Rahpson fashion-like scheme presented by Bouguet [10] is used. Starting from an initial value $[\mathbf{d}\ m\ c]^T_{i+1}$ this scheme updates the parameter iteratively:

$$[\mathbf{d}\ m\ c]^T_{i+1} = [\mathbf{d}\ m\ c]^T_i + [\Delta\mathbf{d}\ \Delta m\ \Delta c]^T_i, \tag{3}$$

where starting from the coarsest level for each level the iteration is initialized with the results of the previous level. In [13] it has been shown that a multi-scale approach has its limitation on estimating very large-range motions. Initializing the iteration process at the coarsest level with a motion vector prediction obtained from a previously estimated global motion model of the scene additionally improves the run-time and accuracy when estimating long-range motions.

For RLOF, the incremental parameter can be estimated directly as a solution of Eq. 2 by:

$$\begin{aligned} [\Delta\mathbf{d}_i\ \Delta m_i\ \Delta c_i]^T &= \mathbf{G}_{IM}^{-1} \cdot \mathbf{b}_{IM,i} \\ \mathbf{b}_{IM,i} &= \sum_{\mathbf{x}\in\Omega} w(\mathbf{x}) \cdot T(\mathbf{x}) \cdot \psi\left(I_{t,i}(\mathbf{x})\right) \\ \mathbf{G}_{IM} &= \sum_{\mathbf{x}\in\Omega} w(\mathbf{x}) \cdot T(\mathbf{x}) \cdot \psi\left(T(\mathbf{x})^T\right), \end{aligned} \tag{4}$$

where $\psi = \dot{\rho}$ is the influence function, i.e. the derivative, of the shrinked Hampel norm. The weighting function $w(\mathbf{x})$ implements an adaptive support region that has been introduced in [11] in order to prevent violations of the local motion constancy assumption and to improve the motion estimation precision at motion boundaries.

III. Sparse Motion Estimation

In this section we will discuss our dense optical flow approach that is based on the interpolation of sparse motion vector sets. The approach comprises of a sparse motion estimation framework based on the previously discussed RLOF and a subsequent image-guided motion vector interpolation. Beside a high accuracy, the key requirement to each component of the system is an efficient operation to obtain a low overall run-time. The sparse motion estimation framework consists of three subsequent modules: feature detection, motion estimation and outlier filtering.

The objective of the feature detection is to sample the initial points P from where to estimate the motion information. This can be based on the geometry of the image, e.g. on a grid or on differential signatures of salient image patterns such as edge-like features. The distribution of the point set P should allow to reconstruct the overall motion field based on the corresponding motion vector set S. In this paper we will evaluate three feature detection methods, namely GRID, GFT and FAST.

The GRID detector is simply defining the points in P at the nodes of a regularly sampled grid. The potential advantages of this method are that the computational effort is negligible and that the samples have a uniform distribution. As a disadvantage, motion vectors are computed at non-cooperative e.g. homogeneous region and can cause errors. The good feature to track (GFT) detector [18] evaluates the minimal eigenvalue of the gradient matrix $\mathbf{G}$, see Eq. 4, if $w(\mathbf{x}) = 1$, $T(\mathbf{x}) = \nabla I(\mathbf{x})$ and $\psi(x) = x$ and was developed with respect to the Lucas Kanade method. The assumption, the minimal eigenvalue of $\mathbf{G}$ to be greater than zero is a necessary condition to solve the Lucas Kanade as well as the RLOF method. The GFT detector and pyramidal Lucas-Kanade (PLK) method is known as KLT tracker. The Features from Accelerated Segment Test (FAST) [19] is a highly-efficient corner detector and has a superior run-time performance and thus used in various real-time applications. The latter two detectors generate sample points most likely at edge-like and less likely in homogeneous

Fig. 1. Example of interpolation results for different feature detection methods for the Rubberwhale sequence of the Middlebury data-set. Left column shows starting points that correspond to valid (green) and filtered (red) motion vectors. Right column shows the geodesic interpolated dense motion field. From top to bottom: grid detector, GFT detector and FAST detector.

regions. The main disadvantage is that this can lead to missing motion information in regions with weak textures but which are strong enough to provide reliable motion information.

For each location $\mathbf{x} \in P$ a motion vector is estimated by applying RLOF. However, it is unavoidable that the estimates contain errors. Confidence measures are an indicator of corrupted motion estimates and are used to automatically post-filter unreliable results. The forward-backward confidence is a well-established measure that is based on the end-point distance between the motion vector $\mathbf{d}$ estimated for the forward path from $I(x,t) \rightarrow I(x,t+1)$ and the motion vector $\mathbf{d}_B$ estimated for the backward path $I(x,t+1) \rightarrow I(x,t)$ with:

$$EP_{FB}(\mathbf{x}) = ||\mathbf{d}(\mathbf{x}) + \mathbf{d}_B(\mathbf{x}+\mathbf{d}(\mathbf{x}))||. \tag{5}$$

Although it requires a dual estimation of the motion vectors, it is rarely coupled to systematic errors that undergo the RLOF estimation, as the boundary conditions of the backward estimation differ from the forward estimation. In the final motion vector set $MV = \{\mathbf{d}(\mathbf{x}) \mid \forall \mathbf{x} \in P, EP_{FB}(\mathbf{x}) < t_{FB}\}$ is a filtered set S in respect to the corresponding forward backward error EP_{FB}.

Figure 1 shows an example of the reconstructed dense vector fields, based on GRID, FAST and GFT feature detections and corresponding motion vectors. This example demonstrates the significance of a well distributed motion vector set. Comparing the motion field at the red wheel shows that FAST and GFT sample too sparsely at these locations.

IV. Motion Vector Field Interpolation

In this section we will discuss four motion field interpolation methods that have shown to be computationally efficient and accurate. The goal of these methods is to estimate a dense motion vector field from the sparse motion vectors set MV.

The geodesic nearest neighbor interpolation (GEO) is a highly-efficient image-guided method. The GEO method is based on the geodesic distance [20]. The geodesic distance between the two image locations $\mathbf{x}_0$ and $\mathbf{x}_1$ is defined as:

$$D(\mathbf{x}_0, \mathbf{x}_1, \nabla I) = \min \ d(\mathbf{x}_0, \mathbf{x}_1), \tag{6}$$

with

$$d(\mathbf{x}_0, \mathbf{x}_1) = \inf_{\Gamma \in \mathcal{P}_{\mathbf{x}_0,\mathbf{x}_1}} \int_0^{l(\Gamma)} \sqrt{1 + \gamma^2 \cdot (\nabla I(r) \cdot \Gamma'(r))^2} dr \tag{7}$$

where $\mathcal{P}_{\mathbf{x}_0,\mathbf{x}_1}$ is the set of all possible paths Γ between $\mathbf{x}_0$ and $\mathbf{x}_1$, $\Gamma'(r)$ denotes the normalized tangent of the path and γ is a parameter that defines the influence between the gradient and spatial distances. In practice, geodesic distance is computed as the shortest path in an image graph created from the corresponding image. The image graph consists of nodes $V(\mathbf{x})$ that correspond to all pixel positions $\mathbf{x}$ and where each node is connected to its four adjacent neighbors. The weight $w_e(\mathbf{x}, \mathbf{x}_N)$ of the edge between the node at $\mathbf{x}$ and the neighboring node $\mathbf{x}_N$ is defined as:

$$w_e(\mathbf{x}, \mathbf{x}_N) = \sqrt{1 + \gamma^2 \cdot ||I_{RGB}(\mathbf{x}) - I_{RGB}(\mathbf{x}_N)||^2} \tag{8}$$

The nearest motion vector for a given position $\mathbf{x}$ is the motion vector that corresponds to a node $V(\mathbf{x}_i)$ with the shortest path to the node $V(\mathbf{x})$. The search has been implemented by a parallelized raster scan algorithm.

The edge-preserving interpolation (EPIC) has been proposed in [6] for dense optical flow. The EPIC interpolates motion vectors by fitting an affine transformation to k nearest support points which are estimated using the geodesic distance. Revaud *et al.* applied geodesic distance on a structured edge map. To produce highly accurate results they incorporate the DeepMatcher [5] to get sparse point correspondences. The EPIC interpolation methods used in this paper will be based on image gradients instead of these structured edge maps.

Furthermore we evaluate the PCA-flow and PCA-layers proposed by Wulff and Black [7]. The idea is to approximate optical flow fields as a weighted sum over a small amount of basis flow fields. These basis flow fields are principle components which have been trained offline from a large database of natural dense flow fields. Both methods reconstruct a dense motion vector field based on sparsely sampled point correspondences. Wulff and Black used point correspondences estimated by a highly efficient feature point matcher from Geiger *et al.* [8]. To reconstruct a dense motion field from a sparse set of motion vectors a regression is applied to find a valid dense representation that explains the given point correspondences. In order to improve the accuracy in the presence of motion boundaries Wulff and Black used an approach that combines different layers of motion fields.

Fig. 2. Example of different motion vector interpolation methods for Sintel data-set. From left to right and top to bottom: frame 13 ambush 5, ground-truth, geodesic interpolation, edge-preserving interpolation [6], PCA [7] based reconstruction and PCA-layers [7] based reconstruction.

TABLE I
COMPARISON OF GEODESIC INTERPOLATED DENSE MOTION VECTOR FIELDS BASED MOTION VECTORS ESTIMATED WITH RLOF AND GFT, FAST, GRID (SIZE 6) FEATURE DETECTORS, AND ESTIMATED BY FEATURE POINT MATCHING AS PROPOSED BY GEIGER *et al.* [8].

Method	Middlebury		Kitti15		MPI-Sintel	
	AEE	t in sec	R3.0	t in sec	AEE	t in sec
RLOF-GFT	0.95	0.12	58.41	0.22	8.17	0.17
RLOF-FAST	0.54	0.19	43.83	0.41	8.00	0.21
RLOF-GRID	**0.35**	0.26	**38.60**	0.53	**5.19**	0.49
FP Matches[8]	1.30	0.06	43.01	0.10	6.19	0.09

TABLE II
COMPARISON OF SPARSE-TO-DENSE INTERPOLATION METHODS BASED ON RLOF MOTION VECTORS ON A GRID SIZE OF 6 USING GEODESIC INTERPOLATION (GEO), EDGE-PRESERVING INTERPOLATION EPIC [6], PCA [7] BASED AND PCA-LAYERS [7] BASED RECONSTRUCTION, AND REFERENCE PCA APPROACHES AS IN [7].

Method	Middlebury		Kitti15		MPI-Sintel	
	AEE	t in sec	R3.0	t in sec	AEE	t in sec
RLOF-GEO	0.35	0.26	38.60	0.53	5.19	0.49
RLOF-EPIC	**0.33**	0.31	**34.89**	0.61	5.00	0.56
RLOF-PCA	0.55	0.43	39.41	0.73	5.52	0.69
RLOF-PCA-layer	0.50	2.68	36.81	3.41	4.98	3.70
PCA-flow [7]	0.70	**0.11**	38.60	**0.13**	5.41	**0.13**
PCA-layers[7]	0.67	1.76	36.30	2.09	**4.74**	2.07

Figure 2 shows the result of the interpolation process with the different methods on the same set of sparse motion vectors, the corresponding color image and the ground truth for this sequence. It is apparent that the flow field of the PCA method is missing sharp boundaries while that produced with the GEO method contains the sharpest boundaries.

V. EXPERIMENTS

In this section we want to evaluate the performance of the proposed sparse-to-dense interpolation scheme based on RLOF local optical flow method. All experiments have been performed on the training sequences of the Middlebury [21], KITTI 2015 [22] and Sintel [23] optical flow data-sets. If not specified we use default configurations for all applied methods. In the first experiment we compare the motion vectors obtained by the RLOF based framework proposed in section III with point correspondences obtained by the high performant feature matching method proposed by Geiger *et al.* [8] and used by the PCA-flow and PCA-layer algorithm [7]. The RLOF configuration is as follows: 4 pyramidal levels, maximal support region size 21, minimal support region size 9, 30 maximal number of iterations, color threshold $\tau = 35$, $t_{FB} = 0.2$ for Middlebury and $t_{FB} = 0.41$ for KITTI 2015 and MPI-Sintel data-set. For the GEO $\gamma = 1000$.

Comparing the results in table I shows the RLOF with grid based feature detector outperforming the feature matching and the RLOF based on GFT and FAST detections for each benchmark. The experiments show the significance of the distribution of the sparse motion information when interpolating with geodesic method. The main advantage of local optical flow methods in this process is the ability to determine the locations of the motion vectors to be estimated.

In the second experiment we compare the interpolation methods based on geodesic (GEO), edge-preserving (EPIC)[1], PCA-flow [7] and PCA-layers [7][2]. As reference we chose the PCA and PCA-layer method based on feature matching as it was published in [7]. To our best knowledge this method is the best compromise between accuracy and run-time. Figure 3 shows the accuracy and run-time results for the four interpolation methods. In this experiment we have used a subset of images for the MPI-Sintel data-set. Interesting results are shown for the Middlebury and Sintel results where the error only slightly increases with increasing grid size for all interpolation methods. In contrast, the run-time reduces significantly for increasing grid sizes. The outliers at grid size 8 and 9 on the Sintel data-set for EPIC occur due to a gross-error in the fitting of the local affine transformation. The plot for KITTI 2015 shows a strong dependency of the accuracy related to grid size. This strong dependency can be explained by the strong zoom which is typical for the KITTI data-set.

Table II summarizes the final error and run-time of the proposed framework with the four interpolation methods and compares them to the matching based PCA reference methods as in [7]. In total RLOF-EPIC method shows best performance when comparing run-time and accuracy. In comparison to state-of-the-art PCA based approaches significant improve-

[1]implementation used from https://github.com/opencv/opencv_contrib
[2]available at https://github.com/jswulff/pcaflow

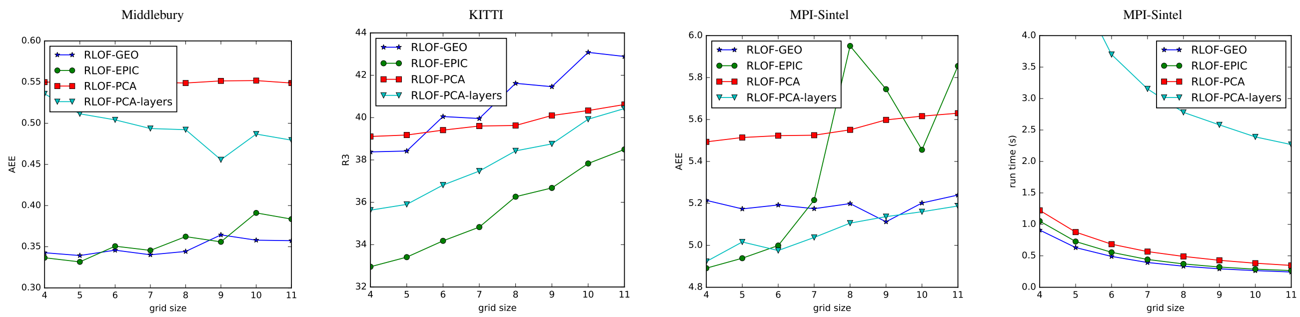

Fig. 3. Evaluation results on training sequences of Middlebury, KITTI and MPI-Sintel benchmarks. This evaluation shows the dependency of the grid size for the sparse motion estimation. For the Middlebury and Sintel Sequences the plot shows the average end-point error (AEE) for the KITTI data-set the R3 measure and for the Sintel the performed run-time.

ments could be achieved for the accuracy on Middlebury and KITTI 2015 data-sets. Compared to the accurate but more complex PCA-layers approach the RLOF-EPIC approach performs about 5 times faster on Middlebury and about 3 times faster on KITTI and MPI-Sintel data. Compared to the less accurate but very fast PCA-flow approach the RLOF-EPIC achieved a 2 times lower AEE on Middlebury and an about 9% improved accuracy on KITTI and MPI-Sintel.

VI. CONCLUSION

The motivation of this work was to develop a run-time efficient optical flow method for dense motion field estimation. Therefore, we proposed an approach based on interpolating sparse motion vectors. The interpolation is performed on motion vectors located at nodes of a regular sampled grid and estimated by Robust Local Optical Flow method. We studied different interpolation methods and found the edge-preserving interpolation to be most suitable. Interestingly, the accuracy of our method decreases only slightly but the run-time can be reduced significantly when increasing the grid size. With 0.56 seconds on MPI-Sintel data (1024×436) the proposed method is one of the fastest approaches compared to the state-of-the-art that has competitive accuracy.

ACKNOWLEDGMENT

The research leading to these results has received funding from the BMWi-ZIM and BMBF-VIP under grant agreement number 16KN022845 (iFore) and number 03V0328 (Qua-Video).

REFERENCES

[1] D. Fortun, P. Bouthemy, and C. Kervrann, “Optical flow modeling and computation: A survey,” *Computer Vision and Image Understanding*, vol. 134, pp. 1–21, 2015.

[2] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, “High accuracy optical flow estimation based on a theory for warping,” *European Conference on Computer Vision*, vol. 3024, pp. 25–36, 2004.

[3] D. Sun, E. B. Sudderth, and M. J. Black, “Layered segmentation and optical flow estimation over time,” *Conference on Computer Vision and Pattern Recognition*, pp. 1768–1775, 2012.

[4] H. Zimmer, A. Bruhn, and J. Weickert, “Optic flow in harmony,” *International Journal of Computer Vision*, vol. 93, no. 3, pp. 368–388, 2011.

[5] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, “DeepFlow: Large Displacement Optical Flow with Deep Matching,” in *International Conference on Computer Vision*, 2013, pp. 1385–1392.

[6] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, “EpicFlow : Edge-Preserving Interpolation of Correspondences for Optical,” in *Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1164–1172.

[7] J. Wulff and M. J. Black, “Efficient Sparse-to-Dense Optical Flow Estimation using a Learned Basis and Layers,” in *Conference on Computer Vision and Pattern Recognition*, 2015, pp. 120–130.

[8] A. Geiger, J. Ziegler, and C. Stiller, “Stereoscan: Dense 3d reconstruction in real-time,” in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 963–968.

[9] T. Senst, V. Eiselein, and T. Sikora, “Robust Local Optical Flow for Feature Tracking,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 9, pp. 1377–1387, 2012.

[10] J.-Y. Bouguet, “Pyramidal Implementation of the Lucas Kanade Feature Tracker,” Intel Corporation Microprocessor Research Lab, Technical Report, 2000.

[11] T. Senst, T. Borgmann, and T. Sikora, “Cross based Robust Local Optical Flow,” in *IEEE International Conference on Image Processing*, 2014, pp. 1967–1971.

[12] T. Senst, J. Geistert, I. Keller, and T. Sikora, “Robust local optical flow estimation using bilinear equations for sparse motion estimation,” in *IEEE International Conference on Image Processing*, 2013, pp. 2499–2503.

[13] T. Senst, J. Geistert, and T. Sikora, “Robust Local Optical Flow: Long-Range Motions and Varying Illuminations,” in *International Conference on Image Processing*, 2016.

[14] B. K. P.Horn and B. G. Schunck, “Determining optical flow,” *Artifical Intelligence*, vol. 17, pp. 185–203, 1981.

[15] C. Tomasi and T. Kanade, “Detection and tracking of point features,” CMU, Technical Report CMU-CS-91-132, 1991.

[16] M. A. Gennert and S. Negahdaripour, “Relaxing the brightness constancy assumption in computing optical flow,” MIT AI Lab. Memo No.975, Tech. Rep., 1987.

[17] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *International Joint Conference on Artificial Intelligence*, 1981, pp. 674–679.

[18] J. Shi and C. Tomasi., “Good features to track,” in *Computer Vision and Pattern Recognition*, 1994, pp. 593 –600.

[19] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *European Conference on Computer Vision*, 2006, pp. 430–443.

[20] A. Criminisi, T. Sharp, C. Rother, and P. Perez, “Geodesic image and video editing,” *Invited talk at SIGGRAPH, Aug 2011, Vancouver. Published in ACM Transactions on Graphics, 2010.*, August 2011.

[21] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, “A database and evaluation methodology for optical flow,” in *International Conference on Computer Vision*, 2007, pp. 1–8.

[22] M. Menze and A. Geiger, “Object scene flow for autonomous vehicles,” in *Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3061–3070.

[23] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, “A naturalistic open source movie for optical flow evaluation,” in *European Conf. on Computer Vision*, 2012, pp. 611–625.