CROSS BASED ROBUST LOCAL OPTICAL FLOW

Tobias Senst, Thilo Borgmann, Ivo Keller and Thomas Sikora

Technische Universität Berlin
Communication Systems Group
EN 1, Einsteinufer 17, 10587 Berlin, Germany

ABSTRACT

In many computer vision applications local optical flow methods are still a widely used. Such methods, like the Pyramidal Lucas Kanade and the Robust Local Optical Flow, have to address the trade–off between run time and accuracy. In this work we propose an extension to these methods that improves the accuracy especially at object boundaries. This extension makes use of the cross based variable support region generation proposed in [1] accounting for local intensity discontinuities. In the evaluation using Middlebury data set we prove the ability of the proposed extension to increase the accuracy by a slight increase of run time.

Index Terms— Optical Flow, KLT, RLOF, Feature Tracking, Cross-based region construction

1. INTRODUCTION

In a wide area of video-based computer vision applications motion information has become an important cue. The concept of optical flow formulates the base of the most common motion estimation approaches, not at least because the accuracy and efficiency of these techniques have been substantially improved in recent years.

The taxonomy of optical flow based methods denotes global and local approaches. These two classes differ in the way the spatial coherence has been implemented which is needed to solve the data conservation or the so called intensity constancy constraint. Global methods rely on the data of the whole image through coupled energy terms as, e.g, the smoothness constraint proposed by Horn and Schunck [2]. In this way, these methods are able to provide a very accurate and dense motion field. However their computational complexity is related to the input images size and not to the number of estimated motion vectors.

For many applications in the field of video-based surveillance [3], medical imaging [4] and video coding [5] only a sparse set of motion vectors is required. These applications not only benefit from the accuracy but also from computational efficiency of local motion estimation techniques. Local approaches incooperate the spatial coherence with the textural information of a surrounding image region. In this way they are scalable with respect to the number of motion vectors to be estimated and thus are very efficient in estimating sparse motion information.

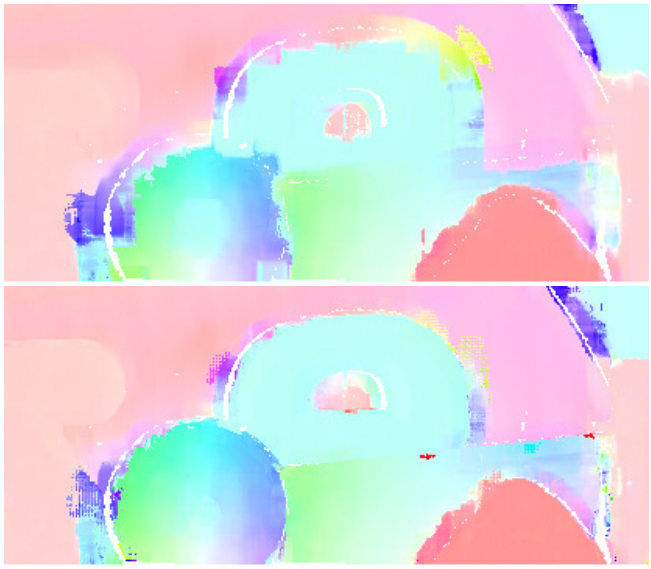

Fig. 1. Exemplary motion vector field estimated by the BERLOF (top) and the proposed adaptive support region modification (bottom) for the RubberWhale sequence.

In general global methods are more accurate than local by comparing dense motion estimates. But the impetus for the development of new local optical flow based methods involves the consideration of accuracy and run time aspects for a sparse set of motion vectors. For example in [6] it has been shown that for the evaluation of sparse motion fields local methods are competitive to state-of-the-art global ones.

The research on local optical flow is mainly based on the KLT (Kanade Lucas Tomasi) tracker [7]. Several methods address the run time aspect by improving the performance through parallelization, e.g. GPU implementations were presented by Sinha *et al.* [8], or reducing the computational complexity through additional approximations of the data term by using integral projections [9]. In [6] the bilinear interpolation filter for the iterative scheme has been examined to reduce

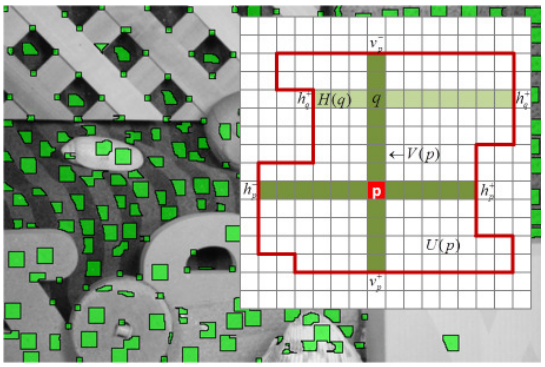

Fig. 2. Schema of the construction of the adaptive support region for the anchor pixel p. Green areas denote exemplary support regions.

computational complexity without losses in accuracy.

To enhance the accuracy Odobez *et al.* [10], Kim *et al.* [11] and Senst *et al.* [12] investigate into norms that are robust against outliers. In addition the shape of the region could be enhanced e.g. by Gaussian and Laplacian of Gaussian weighting functions as proposed in [13] or by changing the region size in relation the residual error and texture structure as proposed in [12].

In this paper we will further enhance the previous work resulting into the RLOF (Robust Local Optical Flow) [12] and its accelerate derivative the BERLOF [6]. We observe that a limitation of the previous work lies in the rectangular shape of the support region. As shown in [12] this leads to the generalized aperture problem, where a large support region increases the probability for containing multiple moving objects. To enhance the accuracy in that case we propose to use an adaptive support region in order to respect motion boundaries. Therefore we utilize the generation of adaptive support regions proposed by Zhang *et al.* [1]. This method has proven to be computational efficient and is therefore well suited in respect to the run time requirements. In the following we provide a general framework to apply the adaptive support regions for gradient-based local optical flow methods. An implementation will be given for the PLK (Pyramidal Lucas Kanade) method and the RLOF.

2. CROSS BASED ROBUST LOCAL OPTICAL FLOW

The RLOF as well as the PLK could be formulated by the energy term of the generalized gradient-based optical flow equation [12]

$$\min_{\mathbf{d}} \sum_{\Omega} w(\mathbf{x}) \cdot \rho \left(\nabla I(\mathbf{x})^T \cdot \mathbf{d} + I_t(\mathbf{x}), \boldsymbol{\sigma} \right) \qquad (1)$$

where $\mathbf{d}$ denotes the displacement for a small region Ω at a time t. The displacement is estimated depending on the spatial derivative $\nabla I(\mathbf{x})$ and temporal derivative $I_t(\mathbf{x})$ of a grayscale image $I(\mathbf{x}, t)$ with $I_t(\mathbf{x}) = I(\mathbf{x}, t) - I(\mathbf{x}, t+1)$ with $\mathbf{x} \in \Omega$, $w(\mathbf{x})$ is a weighting function and ρ a norm with its scale parameters $\boldsymbol{\sigma}$. The least square estimator is applied by the PLK to solve eq. 1 while the more robust shrinked Huber norm is applied by the RLOF to reduce the influence of outliers. In addition a pyramidal implementation and an iterative scheme in a Newton-Raphson fashion is applied, so that

$$\Delta \mathbf{d}^i = \mathbf{G}^{-\mathbf{1}} \cdot \mathbf{b}^{i-1} \qquad (2)$$

denotes the incremental motion vector and

$$\mathbf{d}^i \leftarrow \mathbf{d}^{i-1} + \Delta \mathbf{d}^i \qquad (3)$$

with the gradient matrix $\mathbf{G}^{-\mathbf{1}}$ and the mismatch vector $\mathbf{b}^{i-1}$ for the current iteration i. For further details refer to [14, 12].

For each iteration i the mismatch vector has to be updated according to the previous displacement. In order to account for displacement in the subpixel domain a bilinear interpolation is applied. To apply the interpolation four support points are necessary. The support points surrounding the endpoint of the motion vector $\mathbf{d}^i$ and are located at the respective integer value positions. In [6] it is shown mathematically that the incremental motion vector depends on these support vectors and the systems of bilinear equation. Consequently, the update of the temporal derivatives is not necessary if the support points are constant and the final motion vector could be estimated directly by a systems of bilinear equations under certain conditions as described in [6].

In general the weighting function for local methods is chosen to be $w(\mathbf{x}) = 1$. This leads to the violation of the Lucas Kanade constant motion constraint that assumes a single motion component in a support region since multiple moving objects can be covered by the rigid shape. To account for local motion discontinuities we apply the adaptive support region definition as proposed in [1]. Therefore we assume that the boundaries of different moving objects lead to motion discontinuities. Thus, the shape of the support region has to correspond to the motion discontinuities during flow estimation. Assuming that the local distribution of the intensity values correspond to the object boundaries, we apply the cross based method of [1], which utilizes the color similarity for generation of a variable support region.

The construction of the local support regions is done in two stages. First, two horizontal and two vertical arms are created for every pixel. Second, the support region is build by a combination of all horizontal arms defined by the pixels within the vertical arm of the center pixel respectively. The definition of the horizontal and vertical arms for a given pixel depends on a threshold value τ that defines the maximum absolute intensity difference between the center pixel at position p and the respective pixel at p_n located on the corresponding arm. The resulting arm length r^* is then defined to consist of all pixels directly connected to the center pixel for which the absolute intensity difference does not exceed τ. The defini-

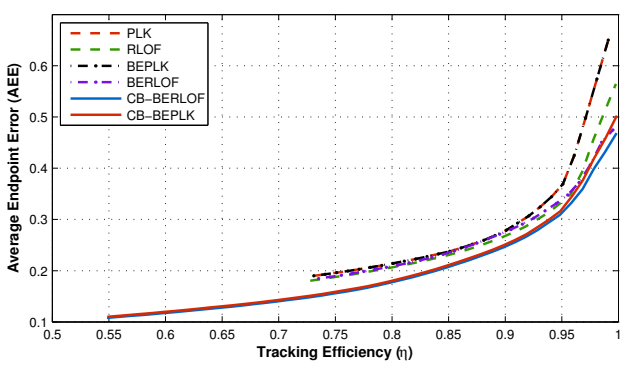

Fig. 3. Evaluation of the tracking performance averaged over all Middlebury sequences.

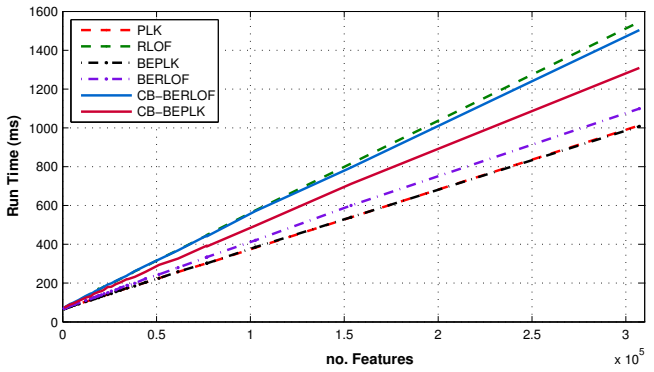

Fig. 4. Run time comparison of the local optical flow methods for the Grove3 sequence of the Middlebury dataset.

tion of r^* is formulated as follows:

$$r^* = \underset{r\in[1,L]}{max} \left(r \cdot \prod_{n\in[1,r]} \delta(p, p_n) \right) \tag{4}$$

Where L denotes the maximum arm length and $\delta(p, p_n)$ is computed considering all three RGB color channels by:

$$\delta(p, p_n) = \begin{cases} 1, & \underset{c\in[R,G,B]}{max} (|I_c(p) - I_c(p_n)|) < \tau \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The resulting four arms lengths $h_p^-, h_p^+, v_p^-, v_p^+$ can then be stored efficiently in the four channels of a dedicated image used as a lookup table during the second stage to define the resulting adaptive support region. Following [1], the support region is defined by the combination of all horizontal $H(p) = \{\mathbf{x} = (x, y)|x \in [x_p - h_p^-, x_p + h_p^+], y = y_p\}$ and vertical arms $V(p) = \{\mathbf{x} = (x, y)|x = x_p, y \in [y_p - v_p^-, y_p + v_p^+]\}$. The integration of the adaptive support region into the minimization functional given in eq. 1 is applied by the weighting function which is then given by

$$w(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in U(p) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

with the support region

$$U(p) = \bigcup_{q\in V(p)} H(q) \tag{7}$$

Figure 2 gives an example of estimated support regions and the schematic overview of the support region construction by the corresponding arms.

3. EVALUATION

In section we evaluate the proposed adaptive support region as a generic tool for local optical flow methods. Therefore we modify the existing BEPLK and BERLOF algorithms [6] which are enhanced versions of the PLK [1] and RLOF [2] methods [14, 12]. The performance of the new methods CB-BEPLK and CB-BERLOF have been evaluated in terms of run time and accuracy using the Middlebury optical flow data set [15]. All methods are implemented using SIMD extension of the CPU (SSE2) and multi-threading on a PC with 4x3.56 GHz Intel CPU. For each method we use the same basic configuration, i.e. 3 pyramid levels, $\Omega = 19 \times 19$ region size, the convergence criteria are set to a maximum of 20 iterations and $\epsilon = 0.1$. For the methods based on the RLOF the following additional parameters are set, $\sigma = (32, 160)$ and 7×7 for the small region size. For the modified methods using the adaptive support region the color threshold is $\tau = 25$ with an maximum arm length of $L = 9$. To avoid the aperture problem we enforce a minimum arm length of $L = 3$. In figure 1 we show an exemplary result of the improved motion estimation at object boundaries.

The accuracy of the proposed methods is compared in a sparse manner by using a feature tracking framework. Following [6] a forward–backward confidence measure is applied to reject false estimates. Table 1 shows the result for each sequence of the Middlebury data set comparing the Average Endpoint Error (AEE) and the tracking efficiency η. While the Average Endpoint Error is an accuracy measure, the tracking efficiency is a measure of successfully tracked features. Both measures have to be taken into account to rate the quality of the different methods, which is done by the tracking performance plot. Figure 3 shows the tracking performance [12] averaged over the whole data set, where low AAE and high η are preferable.

The tracking performance plot was computed by varying the confidence threshold and provides an overview of the AEE related to the number of tracked features. Similar result can be observed for the PLK/RLOF and the corresponding BEPLK/BERLOF methods. In contrast the proposed extended

[1] download at http://www.opencv.org/ (v2.4.6)

[2] download at http://www.nue.tu-berlin.de/menue/forschung/projekte/rlof/

	Dimetrodon		Grove2		Grove3		Hydrangea		RubberWhale		Urban2		Urban3		Venus	
	AEE	η	AEE	η	AEE	η	AEE	η	AEE	η	AEE	η	AEE	η	AEE	η
PLK	0.09	99.6	0.23	96.1	0.56	88.2	0.13	97.2	0.14	97.8	0.15	84.4	0.62	84.0	0.20	92.2
RLOF	0.09	99.7	0.21	96.4	0.54	88.2	0.13	97.3	0.14	97.7	0.14	85.7	0.60	84.2	0.20	92.8
BEPLK	0.09	99.6	0.23	96.1	0.56	88.2	0.13	97.2	0.14	97.8	0.15	84.4	0.62	84.0	0.20	92.2
BERLOF	0.09	99.6	0.21	95.1	0.54	86.3	0.13	97.0	0.14	97.2	0.14	85.1	0.61	84.0	0.20	91.6
CB-BERLOF	0.09	98.9	0.13	91.6	0.32	78.3	0.11	92.4	0.09	94.9	0.13	83.4	0.40	78.2	0.18	84.5
CB-BEPLK	0.09	98.9	0.13	91.2	0.33	77.9	0.12	92.3	0.09	94.8	0.13	83.3	0.41	78.2	0.18	84.4

Table 1. Results of the Middlebury training sequences for sparse motion estimation and a fixed confidence threshold of 0.5.

methods using variable support regions, CB-BEPLK/CB-BERLOF, clearly outperform the other methods in terms of accuracy.

Figure 4 provides a run time comparison between all evaluated methods. The run time was measured related to a varying number of features to track. Both proposed extended methods, like the other methods, still feature a linear computational complexity related to the number of features. The overall run time of the proposed methods can still cope with the existing ones. Therefore, the proposed methods should be preferred for their enhanced accuracy in all cases where the extended run time is negligible.

4. CONCLUSION

In this paper we propose to apply an adaptive support region based on color information to increase the estimation accuracy of the optical flow. The modification is implemented by replacing the constant weighting function in the generalized local optical flow equation with a new weighting function derived from the adaptive support region. This especially increases the motion estimations around object boundaries. The evaluation with the Middlebury data set shows an improved tracking performance for the cost of a slide increase in run time. But still the computational complexity remains linear related to the number of features to track. The proposed extension is designed to be generic and therefore applicable for most state-of-the-art methods that have been shown exemplary by for the Pyramidal Lucas Kanade and the Robust Local Optical Flow methods.

Acknowledgment

The research leading to these results has received funding from the European Community's FP7 under grant agreement number 261743 (VideoSense) and number 261776 (MOSAIC).

5. REFERENCES

[1] Ke Zhang, Jiangbo Lu, and Gauthier Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073–1079, 2009.

[2] Berthold K. P. Horn and Brian. G. Schunck, "Determining optical flow," *Artifical Intelligence*, vol. 17, pp. 185–203, 1981.

[3] H. Fradi and J.-L. Dugelay, "Crowd density map estimation based on feature tracks," in *International Workshop on Multimedia Signal Processing (MMSP 2013)*, 2013, pp. 40–045.

[4] Christoph B. H. Antink, Tarunraj Singh, Puneet Singla, and Matthew Podgorsak, "Evaluation of advanced lukaskanade optical flow on thoracic 4d-ct," *Journal of Clinical Monitoring and Computing*, vol. 27, no. 4, pp. 433–441, 2013.

[5] Michael Tok, Alexander Glantz, Andreas Krutz, and Thomas Sikora, "Monte-carlo-based parametric motion estimation using a hybrid model approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 607–620, 2013.

[6] Tobias Senst, Jonas Geistert, Ivo Keller, and Thomas Sikora, "Robust local optical flow estimation using bilinear equations for sparse motion estimation," in *IEEE International Conference on Image Processing (ICIP 2013)*, 2013.

[7] Carlo Tomasi and Takeo Kanade, "Detection and tracking of point features," Technical report CMU-CS-91-132, CMU, 1991.

[8] Sudipta N. Sinha, Jan-Michael Frahm, Marc Pollefeys, and Yakup Genc, "Gpu-based video feature tracking and matching," Technical report 06-012, UNC Chapel Hill, 2006.

[9] Tobias Senst, Volker Eiselein, Michael Pätzold, and Thomas Sikora, "Efficient real-time local optical flow estimation by means of integral projections," in *International Conference on Image Processing (ICIP 2011)*, 2011, pp. 2393–2396.

[10] J.M. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of*

Visual Communication and Image Representation, vol. 6, no. 4, pp. 348–365, 1995.

[11] Yeon-Ho Kim, , Aleix M. Martinez, and Avi C. Kak, "A local approach for robust optical flow estimation under varying illumination," in *British Machine Vision Conference (BMVC 2004)*, 2004.

[12] Tobias Senst, Volker Eiselein, and Thomas 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.

[13] Meghna Singh, Mrinal K. Mandal, and Anup Basu, "Gaussian and laplacian of gaussian weighting functions for robust feature based tracking," *Pattern Recognition Letters*, vol. 26, no. 13, pp. 1995–2005, 2005.

[14] Jean-Yves Bouguet, "Pyramidal implementation of the lucas kanade feature tracker," Technical report, Intel Corporation Microprocessor Research Lab, 2000.

[15] Simon Baker, Daniel Scharstein, J.P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski, "A database and evaluation methodology for optical flow," Technical report MSR-TR-2009-179, Microsoft Research, 2009.