ISSN: 0976-9102(ONLINE)
DOI: 10.21917/ijivp.2012.0066

ANNOTATION SUPPORTED OCCLUDED OBJECT TRACKING

Devinder Kumar[1] and Amarjot Singh[2]
Department of Electrical Engineering, National Institute of Technology Warangal, India
E-mail: [1]devinderkumar@ieee.org and [2]amarjotsingh@ieee.org

Abstract

Tracking occluded objects at different depths has become as extremely important component of study for any video sequence having wide applications in object tracking, scene recognition, coding, editing the videos and mosaicking. The paper studies the ability of annotation to track the occluded object based on pyramids with variation in depth further establishing a threshold at which the ability of the system to track the occluded object fails. Image annotation is applied on 3 similar video sequences varying in depth. In the experiment, one bike occludes the other at a depth of 60cm, 80cm and 100cm respectively. Another experiment is performed on tracking humans with similar depth to authenticate the results. The paper also computes the frame by frame error incurred by the system, supported by detailed simulations. This system can be effectively used to analyze the error in motion tracking and further correcting the error leading to flawless tracking. This can be of great interest to computer scientists while designing surveillance systems etc.

Keywords:

Image Annotation, Object Tracking, Depth, Occlusion, Contour

1. INTRODUCTION

Motion tracking plays an important role in the analysis of any video sequence. Over the years motion tracking is being applied widely in multiple fields like biomechanics [2], avionics [3], sport analysis [4], medical [5] etc. Despite the ability of the present systems, occlusion, depth variation, and blurriness are some of the issues which can hinder the effective object tracking. The real world video sequences consist of complex cases of occlusion that are difficult to handle thus occlusion. One of the tedious areas of interest with multiple applications like scene recognition, surveillance, object tracking etc is tracking occluded objects at different depths.

This paper focuses on using the annotation tool provided in [1] to label and track different objects in three similar video sequences varying in depth. The system provides a robust algorithm to track object in above mentioned complex occluded video sequences. The effort in labeling and tracking the object is greatly decreased by allowing the user to make as well as label the contour of object in any one frame followed by the automatic tracking of the contour in other frames. The human interaction plays a pivotal role in labeling the objects as the user can correct the label in different frames thus removing the error produced by the computer vision system, hence increasing the efficiency of the system.

The paper tries to analyze a very important and crucial aspect related to tracking occluded objects. We establish a threshold after which ability of the system to track the occluded object fails. Three similar videos having two moving objects, one occluding the other with variation in depth, were analyzed. In the videos two bikes, one occluding the other at depth 60cm, 80cm and 100cm were analyzed. Another experiment is performed on tracking humans with similar depth to authenticate the results. The results were explained on the grounds of pyramids. The paper also computes the frame by frame error incurred by the system, supported by detailed simulations. This can be of great use to computer scientists especially who design system for surveillance, defense, ballistics etc.

The following paper has been divided into five sections. The next elaborates the algorithm implemented by the paper for tracking the contour of the object. Section 3 explains the simulation results while the final section 4, discusses the summary of the paper.

2. HUMAN BASED ANNOTATION

The system used in the paper makes use of human assisted layer segmentation, and automatic estimation of optical flow for object contour tracking. In order to increase the robustness of the system, the objective functions of flow estimation and flow interpolation are modeled on lagrange's L1 form [1]. Many techniques such as iterative reweighted least square (IRLS) [5, 6] and pyramid based coarse-to-fine search [4, 6] were used at large for the optimization of these non linear object functions.

2.1 HUMAN ASSISTED LAYER SEGMENTATION

This module works on the basis of human interaction with the labeling. The first step is the Initialization of contour in one frame. Due to background cluttering or other changes like shadow etc in the frame, errors can occur in the contour formed by the user. The error in contour can be corrected anytime by the user in any frame which is further automatically passed to the other frames. The forward and backward tracking of the target is simulated automatically by the system. Particle filter is used to track the object in the system as real time performance is considered more important than accuracy [7]. In addition, Occlusion handling technique has also been included in the contour tracker itself [1].

Suppose a function is defined using landmarks points as $M = \{a_p : a_p \in R^2\}_{p=1}^{n}$ at frame F_1. The motion vector v_p represents each landmark at frame F_2. Depending upon whether the tracking is back or forth, the frame F_2 can be after or before F_1. Instinctively, we want the movement of contour to be persistent and should match with the image features. In order for the movement to be persistent, we use optimization. The objective function is defined as,

$$B(\{v_p\}) = \sum_{p=1}^{T}\sum_{c\in T_p} m_p(c)\,|\,F_2(a_p + v_p + c) - F_1(a_p + c)\,| + \omega\sum_{p=1}^{T} S_p\,|v_p - v_{p+1}| \quad (1)$$

$v_{T+1} = v_T$, where v is the motion vector. In the equation, the length between the contour points a_p and a_{p+1} is calculated by using the weight S_p; we define, $S_p = \frac{\bar{l}}{l_p + \bar{l}}$ where, $l_p = \|a_p - a_{p+1}\|$ and l is the average of l_p. It's evident from these equations that closer the points in the contour formation, more the probability that the points move together. Variable T_p is a square neighborhood at a_p, while m_p is the region of support for a_p, a binary mask which indicates the presence of each neighboring pixel c inside the pixel, modulated by a two dimensional Gaussian function. In Eq.(1) the objective function mentioned is nonlinear, hence Taylor expansion is used to linearize the data term followed by the optimization of objective function performed through iterative reweighted least square (IRLS) [5, 6] and pyramid based coarse-to-fine search [4, 6]. In order to account for the changes in the lighting condition, the images in F_1 and F_2 contain the first and second order derivative of luminance instead of just RGB channels. The rigidity of the object is controlled by the coefficient ω. The user can set the value of ω before tracking.

For handling occlusion, the user is allowed to specify relative depth and the depth is automatically interpolated (as time function) for the rest of the frames. The contour tracker is driven by a 2nd-order dynamical model for prediction. The prediction is used as an initialization for optimizing Eq.(1). The tracking algorithm iterates between the following two steps to handle occlusion:

(1) Check whether each landmark Z_p is occluded by other layers with smaller depth values. If occlusion is detected for Z_p then set $r_p(c) = 0$, $\forall c \in N_p$, in Eq.(1). This means there is no region to support tracking Z_p.

(2) Optimize Eq.(1) using the coarse-to-fine scheme.

The contour tracker worked fine for most of the cases, but it fails in case of drift from the position especially when the object rotates. To overcome this drawback, the system allows the correction of a landmark to be made at any frame and the change is transferred to the other frames. In the temporal propagation [1], to reconstruct the point modified by the user, the linear regression coefficients for the other points are estimated. The algorithm proposed works astonishingly well. In comparison to the complicated contour tracking/modification algorithm proposed in [8], are too expensive to be implemented for real-time long distance environments.

2.2 LAYER BY LAYER OPTICAL FLOW ESTIMATION

The mask showing the visibility of each layer is the main difference between layer by layer optical flow estimation and traditional flow estimation for the whole frame. The pixels lying inside the mask are only used for matching. For occlusion handling problem, apart from the normal procedure, outlier detection is also performed to segregate occlusion in the evaluation of optical flow to compensate the irregularity caused in the evaluation due to arbitrary shape of the mask.

For baseline line model for optical flow estimation the system uses optical flow algorithm [5,6], while to improve the accuracy symmetric flow, computation is included. Let E_1 and E_1 be the visible mask of a layer at frame F_1 and F_2, (g_1, h_1) be the flow field from F_1 to F_2, and (g_2, h_2) the flow field from F_2 to F_1. Following terms constitute the objective function for approximating the layer by layer optical flow. In the first step, the matching of images with the visible data term is formulated as mentioned in below,

$$B_{data}^{(1)} = \int u * E_1(x,y)\,|F_1(x + g_1, y + h_1) - F_2(x,y)| \quad (2)$$

where, u is the Gaussian filter. The data term $B_{data}^{(2)}$ for (g_2, h_2) is similarly defined. To account for outliers in matching, L1 norm is used. In the second step, smoothness is imposed by,

$$B_{smooth}^{(1)} = \int \left(|\nabla g_1|^2 + |\nabla h_1|^2\right)^{\gamma} \quad (3)$$

where, γ varies between 0.5 and 1. Finally, symmetric matching can be achieved by,

$$B_{sym}^{(1)} = \int |g_1(x+y) + g_2(x + g_1, y + h_1)| + |h_1(x+y) + h_2(x + g_1, y + h_1)| \quad (4)$$

The sum of the above three equation gives the objective function described below,

$$B(g_1, h_1, g_2, h_2) = \sum_{j=1}^{2} B_{data}^{(j)} + \sigma B_{smooth}^{(j)} + \delta B_{sym}^{(j)} \quad (5)$$

IRLS proposed in [5,6] is used as equivalent to outer and inner fixed-points, together with the coarse-to-fine search [4,6] and image wrapping for the optimization of this objective function. After computing the flow at each level of pyramid, the visible layer mask E_1 is approximated on the basis of estimated flow:

- If $B_2(x + g_1, y + h_1) = 0$, then set $B_1(x, y) = 0$
- If in the Eq.(4), the symmetry term is beyond the threshold at (x, y), then set $E_1(x, y) = 0$

Same rule can be used to update E_2. As course to fine technique is used for the algorithm, we get two bidirectional flow fields and cropped visible layer masks that exhibit occlusion. The user is allowed to change the values of σ, δ and γ and in Eq.(5).

2.3 HUMAN ASSISTED MOTION LABELING

On failure of optical flow estimation fails, the user by the help of feature points can specify the sparse correspondence between two frames. The system then automatically produces a parametric motion or interpolates a dense flow field based on the specified sparse correspondence. For the specification of sparse correspondence the user can either use the help of computer for increasing efficiency or manually, taking full control of motion annotation.

Minimum SSD matching and Lucas-Kanade transform [7] is used by the system for finding the best match in the next frame for the feature point specified by user in previous frame. The system depends on the number of feature points specified to determine the mode of parametric motion i.e. translation, affine

transform or homography followed by the estimation of the motion parameters accordingly. The modes mentioned above can also be selected by the user directly and the user even have an option to choose to generate a smooth flow field interpolated using the preconditioned conjugate gradient algorithm.

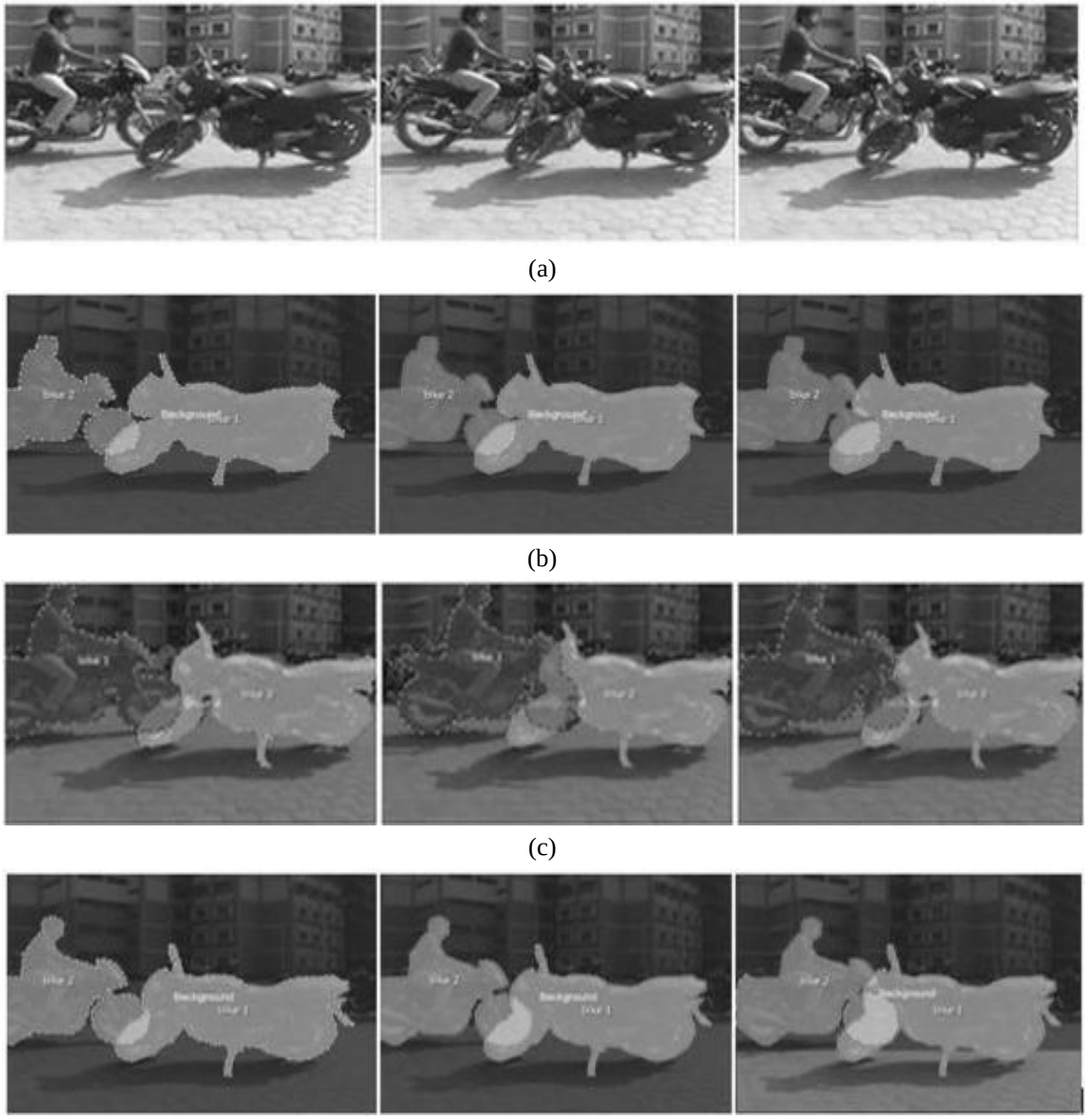
However, defining corner like features for sequences in which only line structure is present can be a difficult task for these kinds of sequences. In order to solve this problem, uncertainty matching and probalisitic parametric motion were included in the algorithm so that the user can have a freedom to choose any pixel for correspondence. In the case of uncertainty matching, a probability map $w_p(x)$ is produced to match the feature point p at location $c_p \in R^2$. A mean χ_p and covariance matrix $\sum_p$ are used to approximate the probability map $H_p(x)$. For the determination of the probabilistic motion estimation, the system loops around two points. In the first step, the current estimate of mean and covariance are used for motion approximation. Mathematically, let $s(c_p;\phi):R^2 \rightarrow R^2$ be a parametric motion applied to the estimation of parametric motion computed by,

$$\phi^* = \arg\min_{\phi} \sum_p \left(s\left(c_p;\phi\right)-\tau_p\right)^T \sum_p \left(s\left(c_p;\phi\right)-\tau_p\right) \quad (6)$$

In second step, estimation of the mean and covariance is done where a new probability map is used which is reweighted by the current motion,

$$\left\{\tau_p, \Sigma_p\right\} \leftarrow i_p(x) N\left(s\left(c_p;\phi^*\right), \phi^2 F\right) \quad (7)$$

Convergence of this algorithm occurs within a few iterations. A dense flow field (i.e. ϕ) can also be obtained for the motion $s(c_p;\phi)$. Also, the feature point specified by the user can be used in the next frame. For providing the human assistance the users interact with the tool through the interface provided in the system developed by the authors of [1].



(a) (b) (c) (d)

Fig.1. a) Contour generated on the objects (b) Annotated sequence for 60cm depth (c) Annotated sequence for 80cm depth (d) Annotated sequence for 100cm depth

(a) (b) (c)

Fig.2. Contour generated on the Human subject for (a) Annotated sequence for 60cm depth (b) Annotated sequence for 80cm depth (c) Annotated sequence for 100cm depth

Table.1. Error incurred during contour based tracking of each frame with respect to ground truth frame in percentage and pixels for at depth (a) 60 (b) 80 (c) 100 (in cm)

(a) 60			
Frame	**Number of pixels**	**Error in pixels with ground frame**	**Error (%) with respect to ground truth frame**
2	70339	00006	0.009
3	70295	00050	0.071
4	69337	01084	1.433
5	69063	01282	1.822
6	68244	02101	2.987
7	67851	02494	3.546
8	66375	03970	5.643
9	64420	05925	8.423
10	60496	09849	14.001
11	58082	12263	17.432
(b) 80			
Frame	**Number of pixels**	**Error in pixels with ground frame**	**Error (%) with respect to ground truth frame**
2	70229	00009	0.0012
3	70197	00041	0.058
4	69374	00864	1.234
5	69052	01186	1.689
6	68517	01721	2.45
7	68161	02077	2.957
8	66893	03345	4.763
9	64667	05571	7.932
10	61918	08320	11.843
11	59308	10930	15.561
(c) 100			
Frame	**Number of pixels**	**Error in pixels with ground frame**	**Error (%) with respect to ground truth frame**
2	10054	001	0.009
3	10064	009	0.089
4	10055	000	0.000
5	10136	081	0.805
6	10161	106	1.050
7	10340	285	2.832
8	10357	302	2.990
9	10402	247	3.450
10	10458	403	4.010
11	10918	863	8.58

Table.2. Error incurred during contour based tracking of each frame with respect to ground truth frame in percentage and pixels for at depth (a) 60 (b) 80 (c) 100 (in cm)

(a) 60			
Frame	**Number of pixels**	**Error in pixels with ground frame**	**Error (%) with respect to ground truth frame**
2	74142	00986	0.30
3	73571	01551	1.04
4	72144	02986	2.97
5	71002	04128	4.48
6	69853	05272	6.61
7	68868	06257	8.37
8	66676	08453	11.27
9	65093	10020	13.32
10	63667	11462	12.28
11	62221	12895	15.13
(b) 80			
Frame	**Number of pixels**	**Error in pixels with ground frame**	**Error (%) with respect to ground truth frame**
2	74151	00964	0.61
3	73570	01535	1.17
4	72183	02922	1.47

5	71031	04124	3.29
6	69891	05234	4.20
7	68817	06268	7.73
8	66683	08416	10.25
9	65036	10079	12.35
10	63681	11424	13.26
11	62216	12889	16.17

(c) 100

Frame	Number of pixels	Error in pixels with ground frame	Error (%) with respect to ground truth frame
2	74131	00984	1.31
3	73570	01555	2.07
4	72193	02982	3.97
5	71031	04124	5.49
6	69861	05274	7.62
7	68827	06258	8.33
8	66653	08456	11.25
9	65076	10029	13.35
10	63631	11464	15.26
11	62236	12899	17.17

3. RESULTS

The aim of the experiment is to analyze the tracking ability of annotation with depth variations. In the videos two bikes, one occluding the other at depth 60cm, 80cm and 100cm were analyzed. Second experiment is performed on tracking humans with similar depth to authenticate the results. The results were explained on the grounds of pyramids. The section further also computes the percentage error in all the sequence at different depths. Finally we establish a threshold at which the system is effectively able to track the occluded object. The simulations were carried out on windows 7 running on an Intel i3 2.26 GHz processor machine. The automatic tracking for all frames in different video sequences takes less than 2 seconds on an average to compute the results.

The focus is to study frames which incur major occlusion, occurring from frame 2 to frame 11 in the first, second and third video sequence respectively as shown in Fig.1(a), 1(b) and 1(c). The contour of the object is tracked from the reference frame to succeeding frame using the layer by layer optical flow estimation as shown in Fig.1.

Error is defined as the total number of extra pixels classified or unclassified in the contour of the succeeding frame over the total number of pixels in the reference contour in the first frame. The ground truth pixels for the reference frame are 70301, 70345 & 70238 for the first, second and third video sequence respectively across which all the error for each frame is evaluated. Detail of the total number of pixels in the succeeding frames along with the error in tracking with respect to the ground truth frame for all the three cases (60cm, 80cm, 100cm) are also shown in Table.1(a), 1(b), 1(c). The error for tracking the object in 100cm sequence varies from a minimum value of 0.012(%) to a maximum of 15.561(%), the error in the 80cm tracking condition varies in a range of 0.009(%) to 17.342(%) while the error in tracking under the 60cm distance outdoor sequence varies from 0.014(%) to 20.343(%).

In the second experiment, we study frames which incur major occlusion, occurring from frame 2 to frame 11 in the first, second and third video sequence respectively as shown in Fig.2(a), 2(b) and 2(c). The contour of the object is tracked from the reference frame to succeeding frame using the layer by layer optical flow estimation as shown in Fig.2.

The ground truth pixels for the reference frame are 74142, 74151 & 74131 for the first, second and third video sequence respectively across which all the error for each frame is evaluated. Detail of the total number of pixels in the succeeding frames along with the error in tracking with respect to the ground truth frame for all the three cases (60cm, 80cm, 100cm) are also shown in Table.2(a), 2(b), 2(c). The error for tracking the object in 100cm sequence varies from a minimum value of 0.30(%) to a maximum of 15.13(%), the error in the 80cm tracking condition varies in a range of 0.61(%) to 16.17(%) while the error in tracking under the 60cm distance outdoor sequence varies from 1.31(%) to 17.17(%). From the error analysis as shown in Table.1 and Table.2, we can state the thresholds at which the ability of the system to track the occluded objects fails is at 3.54 % error at 60cm depth at frame 7, at 4.76 % error at 80cm depth at frame 8 and at 3.45 % error for 100cm depth at frame 9.

4. CONCLUSION

The system is applied to track object in three different video sequences involving different depth between the objects in the sequences. The error plots justify that tracking an objects with larger depth is much more accurate than at less distance between them. It can be justified by the error thresholds also as the ability of the system to track the occluded object fails at frame 6 for 60cm depth while at frame 8 and 9 for 80cm and 100cm depth respectively, hence proving that the occluded object can be tracked more efficiently at one frame with larger depth between the occluded and non occluded objects. The system has vast application in areas where flawless tracking is of great importance. The summary can be effectively used by computer scientists in designing system using image annotation for tracking. Overall, the system can be efficiently used to track the objects in normal as well as occluded conditions in all the different cases.

REFERENCES

[1] Ce Liu ,William T. Freeman, Edward H. Adelson and Yair Weiss, "Human-Assisted Motion Annotation", *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2008.

[2] J.T. Long, N. Jannetto, S. Bakker, S. Smith and G.F. Harris, "Biomechanics of cranial dynamics during daily living artivities", *26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vol. 1, pp. 2417-2419, 2004.

[3] George M. Siouris, Guanrong Chen and Jianrong Weng "Tracking of Incoming ballistic missile using an extended Interval kalman filter", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, No. 1, pp. 232-240, 1997.

[4] Alberto Tomita, Tomio Echigo, Masato Kurokawa, Hisahi Miyamori and Shun-ichi Iisaku, "A visual tracking system for sports video annotation in unconstrained environments", *International Conference on Image Processing*, Vol. 3, pp. 242-245, 2000.

[5] Ting Chen, Member, Xiaoxu Wang, Sohae Chung, Dimitris Metaxas and Leon Axel, "Automated 3D Motion Tracking Using Gabor Filter Bank, Robust Point Matching and Deformable Models", *IEEE Transactions on Medical Imaging*, Vol. 29, No. 1, pp. 1-11, 2010.

[6] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision", *Proceedings of the 7th International joint Conference on Artificial Intelligence*, Vol. 2, pp. 674–679, 1981.

[7] M. J. Black and P. Anandan, "The robust estimation of multiple motions: parametric and piecewise-smooth flow fields", *Computer Vision and Image Understanding*, Vol. 63, No. 1, pp. 75-104, 1996.

[8] 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*, pp. 25–36, 2004.

[9] A. Bruhn, J. Weickert and C. Schnorr, "Lucas/Kanade meets Horn/schunk: combining local and global optical flow methods", *International Journal of Computer Vision*, Vol. 61, No. 3, pp. 211-231, 2005.

[10] M. Isard and A. Blake, "CONDENSATION – Conditional Density Propagation for Visual Tracking", *International Journal of Computer Vision*, Vol. 29, No. 1, pp. 5–28, 1998.

[11] L. Alvarez, R. Deriche, T. Papadopoulo and J. S´anchez, "Symmetrical dense optical flow estimation with occlusions detection", *International Journal of Computer Vision*, Vol. 75, No. 3, pp. 371-385, 2007.