Persistent Object Tracking in Road Panoramic Videos

Yuan Zhou, Zhong Zhou*, Ke Chen, and Wei Wu

State Key Laboratory of Virtual Reality Technology and Systems, Beihang University
School of Computer Science and Engineering, Beihang University
Beijing 100191, P.R. China
zz@vrlab.buaa.edu.cn

Abstract. Panorama has the full directional view of the scene and can provide an object vision persistently from its emerging to vanishing except occlusion. Though, traditional tracking algorithms are apt to fail since the object may change its appearance or even disappear occasionally during its display lifetime. A persistent object tracking algorithm for static objects in panoramic videos is proposed in this paper. It creates several auxiliary trackers to guide the tracking. Once the object is obscured or deformed, the auxiliary trackers are engaged in estimating the position of the main tracker. Even though the appearance of the object changes a lot, its position still could be estimated with help of the auxiliary trackers. Experiment results illustrated that this algorithm provides a real-time tracking effectively on signs in road panoramic videos. This algorithm is easy to perform and especially valuable for road sign labeling and management.

Keywords: persistent, object tracking, panoramic video, auxiliary tracker.

1 Introduction

Object tracking has a variety of vision applications such as surveillance, video classification or labeling and traffic monitoring. Object tracking across multiple camera views has been regarded with high priority in many fields since single camera has limited field of view. Panoramic video has a stitched full directional view, and has a persistent vision of a target from its emerging to disappearing except occlusion. A panoramic camera is designed for video surveillance vehicle to monitor pedestrians [1]. Cylindrical panoramic images are easy to be mapped to a plane and then pursue regular object tracking algorithms for videos. A detection and tracking algorithm for human [2] utilizes a combination of frame differencing, face detection and adaptive color blob tracking to detect and track people in cylindrical panoramic videos. Spherical panorama videos have more distortion in images, and they can be transformed to cubic panorama. The cubic panorama can pursue regular tracking in each side image, but need an expansion for continuous tracking when the object goes through faces [3]. Fig. 1 shows one cubic panoramic frame.

* Corresponding author.

W. Lin et al. (Eds.): PCM 2012, LNCS 7674, pp. 359–368, 2012.

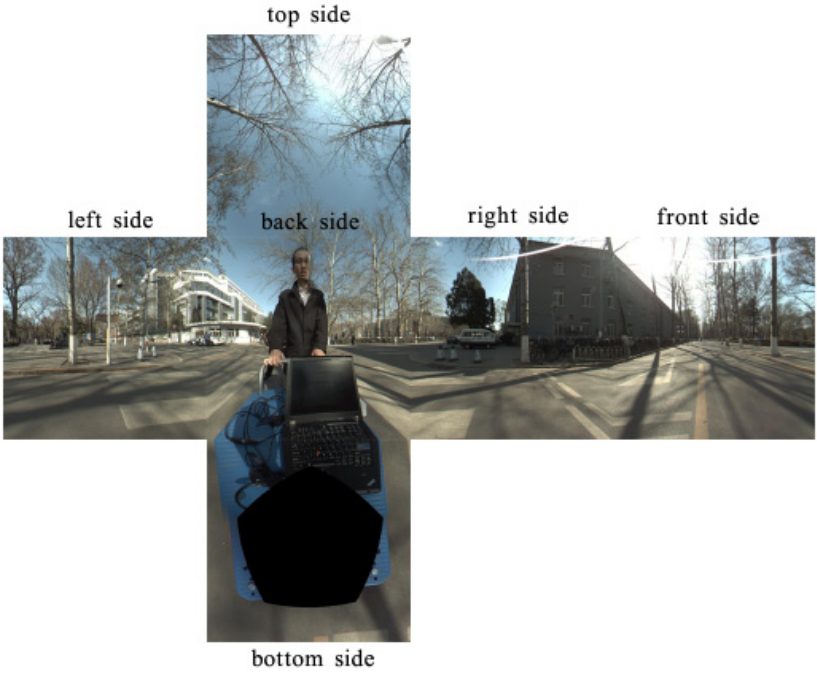


Fig. 1. Cubic panoramic frame

A noticeable phenomenon for multiple camera surveillance is that the appearance of the object may change a lot with the change of the camera on it although it still exists in the stitched panorama frame. Recently some algorithms have been proposed to deal with the problem of appearance change during tracking. Ross et al. proposed incremental learning for visual tracking which is robust to changes in pose, scale, and illumination [4]. Han et al. present a probabilistic sensor fusion technique which shows robustness to severe occlusion, clutter, and sensor failures [5]. W. Du et al. integrated multiple cues, edge, and color in a probabilistic framework [6], and B. Stenger et al. fuse multiple observation models with parallel and cascaded evaluation [7]. Kwon et al. proposed a visual tracking de-composition scheme for the efficient design of observation and motion models as well as trackers [8]. Although these tracking algorithms perform well for the change of the object appearance, they are not applicable to the occasional disappearance of the target object. There is also a huge literature on the subject which uses modern particular filtering approaches. Okuma et al. develop a boosted particle filter that combines detection and tracking [9]. Yin et al. treat tracking as a numerical optimization problem and switch from local to global mode seeking after an occlusion in attempt to detect the position of the object [10]. These methods are used to predict the location of the object in the next frame, which is quite useful in case of short, transient occlusion. For long period occlusions the prediction of these filtering methods will degrade because the tracked object is more probable to move in a way that is not modeled by the filter [11].

Traditional tracking algorithms are apt to fail since the object zone may not only change the appearance but also even be minimized during its display lifetime. It's challenging to make a persistent tracking from an object's emerging to vanishing, regardless of appearance change or even occasional disappearance. The persistent tracking is very important for road sign labeling or management. To the best of our knowledge, little work has been done on persistent tracking in panoramic videos.

We propose an easy-to-perform tracking algorithm in this paper. It doesn't need any 3D calculation or depth analysis, and works on the videos directly. The main contribution of this paper is to create several auxiliary trackers to guide the object tracking in cubic panoramic videos. The auxiliary trackers are preserved to guide the tracking when the main tracker loses the target. Even when the target object disappears for some time, its position still can be estimated with the help of the auxiliary trackers.

2 Motivation and Main Idea

Cubic panoramic video consists of six side images which can be expanded for continuous tracking. Besides the deformation and discontinuity at the boundaries [3], the appearance of an object may change a lot with the change of the camera monitoring it. In fact, road panoramic videos cannot always provide adequate visual size for object observation because of the relative movement.

To illustrate the appearance change of a target object, we count the pixels of the target and calculate the average YUV values in the sign zone. Fig. 2 shows the appearance changing of a road sign in the frame sequence. Furthermore, Fig. 3 shows the number of pixels of the traffic sign in each frame. The pixels diminish in the first 50 frames, reach the minimum at the 50th frame and then increase gradually. When the number of pixels is very few, traditional tracking cannot work. And they cannot be found back since the videos now cover its back instead of front side i.e. the other reason for the failure of directly tracking is the sudden change in the color of the target zone. Fig. 4 shows the average YUV values of the sign in each frame. The change of the color is discontinuous and sharp breaks occur from the 40th frame to the 60th. These two factors make it impossible to track the target directly only with their textures.

Fig. 2. The appearance of a road sign in different frames

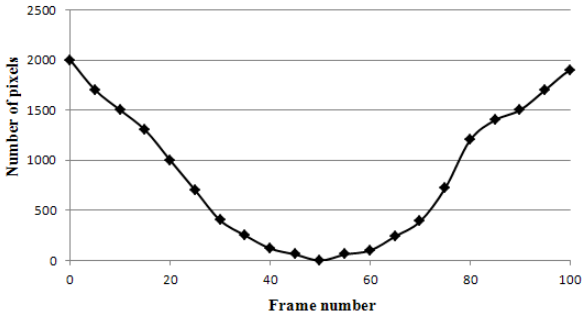


Fig. 3. The number of pixels of the traffic sign in each frame

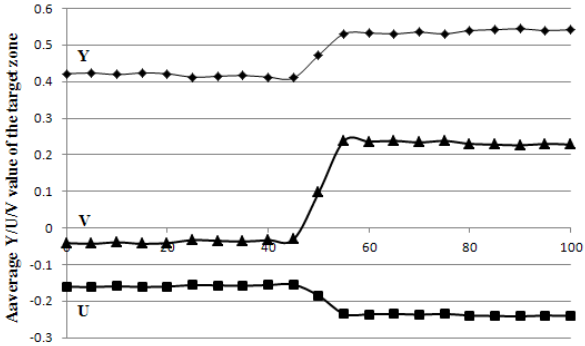


Fig. 4. Average Y/U/V value of the pixels in the target zone

In fact, static objects like traffic signs or trees in close proximity to the camera usually will have similar motion due to resolution limitation. Therefore, the position of the object can be estimated by other ones. We propose to create a main tracker and several auxiliary trackers which can guide the tracking. On the one hand, the main tracker uses a tradition tracking algorithm to track the target object. On the other hand, the auxiliary trackers are composed of dozens of blocks whose motion vectors are the same to the target. The motion vector of each block can be computed by block-matching motion estimation (BMME) because BMME is similar to a rough optical flow and its computation is much lower [12]. When the main tracker loses the target, the auxiliary trackers will help estimate the position of the target. With these auxiliary trackers, the proposed algorithm can track a target object persistently regardless of the appearance change or even occasional disappearance.

The method is very easy to perform. Since nearly all the videos will be encoded before storage or transportation, the motion vectors could be obtained directly. Therefore road signs could be recognized and tracked after block motion estimation in the encoding process, or a real-time object tracking could be applied accompanying the video decoding.

3 Auxiliary Trackers

Auxiliary trackers are created when the observation of the target object is adequate. They will be utilized to estimate the position of the target object when the target is deformed or occluded. We divide every road panoramic video frame into blocks of size 16*16 and compute every motion vector for each block between two adjacent frames and compute the average motion vector $\overline{MV}$ of the target object. Fortunately, the motion vectors can be obtained directly from the video encoding/decoding process.

For simplicity, an auxiliary tracker S is defined as a rectangle area. Suppose that S has n blocks in total in which m ones' motion vectors are $\overline{MV}$. For robustness, S must satisfy two criterion functions J_1 and J_2 as follows:

$$\begin{aligned} J_1 &= n \geq n_0 \\ J_2 &= m/n \geq p_0 \end{aligned} \tag{1}$$

where n_0 and p_0 are two empirical values which indicate the size and reliability restriction respectively.

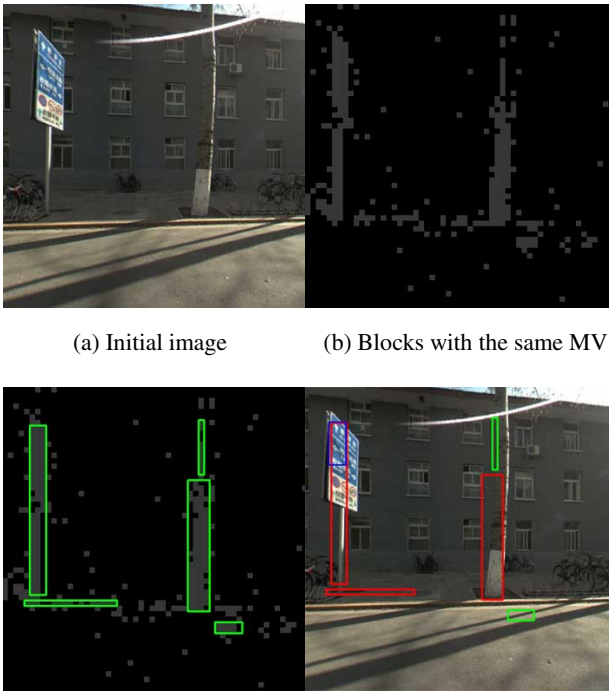
According the criterion functions J_1 and J_2, we cluster those blocks whose motion vectors are the same as that of the target. Let $B=\{B_1,B_2,\ldots,B_N\}$ be the blocks whose motion vectors are $\overline{MV}$ and $R=\phi$ be the initial set of clusters. The details of the clustering algorithm are described as follows:

1. Initialize the first classification $R_1 = \{B_1\}$, add R_1 to R and remove B_1 from B.
2. Suppose there have been K classifications R_1, R_2, …, R_K in the frame image. Choose the next element B_i ($1 \leq i \leq N$) of B. For each cluster R_k in R, extend R_k exactly enough to cover B_i, count the m and n in the extended area and compute the criterion J_2.
3. If R_k satisfies J_2, then assign B_i to R_k and remove B_i from B. If B_i satisfies more than one classification, B_i is supposed to be put into the classification which has the biggest result of J_2 after including B_i.
4. If B_i is not assigned to any clusters R, a new cluster $R_{K+1} = \{B_i\}$ is added to R and remove B_i from B.
5. Repeat the process from (2) to (4), until B is empty.
6. For each classification R_k in R, compute J_1 and if it doesn't satisfy J_1, remove it from R.

Those classifications remained in R are the candidate areas for auxiliary trackers. We build a MAX-HEAP with elements of the candidate areas and sort them by frequency of occurrence in the frame sequence. Candidate areas with high frequency of occurrence are chosen as the auxiliary trackers with priority.

When the observation of the tracked target is adequate for the main tracker, we should identify which two selected candidate areas in adjacent frames are actually of

the same one. When the main tracker becomes invalid, we stop matching the same candidate area and start to track the candidate areas existed in the MAX-HEAP. Candidate areas appearing in every frame are selected as the auxiliary trackers. The process of auxiliary trackers selection is illustrated in Fig. 5 where (a) is the initial image. The blocks with the same MV to the target area's are showed in Fig.5(b). Then the blocks are clustered into several classifications in green rectangles as candidate areas in Fig.5(c), and three are selected as auxiliary trackers in red rectangles as Fig.5(d) in the end.



(a) Initial image (b) Blocks with the same MV

(c) Cluster of blocks (d) Auxiliary trackers selection

Fig. 5. Process of selecting auxiliary trackers (green: candidate areas; red: selected auxiliary trackers)

4 Persistent Tracking

The Background Eliminated Mean-Shift algorithm is chosen as the tracking method for the main tracker, and the observation of the main tracker I_k in frame k is defined by the sum of back projection value of each pixel in the tracked area. We set the original observation of tracked object as I_0, the center of the tracked object as P_0. Starting from the second frame, we compute the observation changing rate d_k in frame k by I_k/I_0. When d_k is bigger than 0.5, the main tracker is supposed to be in the adequate-observation state. In this state we directly use the main tracker's tracking

result as the final position P_k of the tracked object, i.e. $P_k = M_k$, where M_k is the center of the main tracker. At the same time, we compute the relevant distance $L_{k,i}$ between every auxiliary tracker and main tracker by:

$$L_{k,i} = M_k - S_{k,i} \tag{2}$$

where $S_{k,i}$ denotes the center of the i-th auxiliary tracker in the k-th frame. If the value of d_k is between 0.5 and 0.1, the main tracker is supposed to be in the semi-adequate-observation state. In this state, we stop computing the relevant distance, and calculate the final tracking position P_k as a weighted value with the following formula:

$$P_k = M_k d_k + \frac{1}{n}\sum_{i=1}^{n}\left(S_{k,i} + L_{k-1,i}\right)\left(1 - d_k\right) \tag{3}$$

In formula 3, results from both the main tracker and the auxiliary trackers are taken into consideration. When d_k<0.1, the main tracker is supposed to be in the lack-observation state. In this state, we abandon the main tracker's tracking result and directly use the auxiliary trackers to determine the final position P_k:

$$P_k = \frac{1}{n}\sum_{i=1}^{n}\left(S_{k,i} + L_{k-1,i}\right) \tag{4}$$

For each frame during this state, the color histograms of the tracking area of the present frame are compared with the previous one by the following formula:

$$Match\left(h(k-1), h(k)\right) = \sum_{i}^{n}\left|b_{k,i} - b_{k-1,i}\right| \tag{5}$$

where $h(k)$ is the histogram of the k-th frame, $b_{k,i}$ is the i-th bin of the $h(k)$. When the value of function *Match*() is close to 0, we estimate that the observation of the main tracker recovers. Then the main tracker is set to the adequate-observation state, and as before, we directly use the main tracker's tracking result as the final position P_k of the tracked object, i.e. $P_k = M_k$.

The tracking need to stop in conditions below including 1) the main tracker goes into the central zone of the back image of cubic panorama, eg. 20% central window; 2) the target object covers a small zone, eg. 5*5 pixels; 3) all the auxiliary trackers become invalid and the main tracker still doesn't recovers. Thresholds can be set according to the requirements.

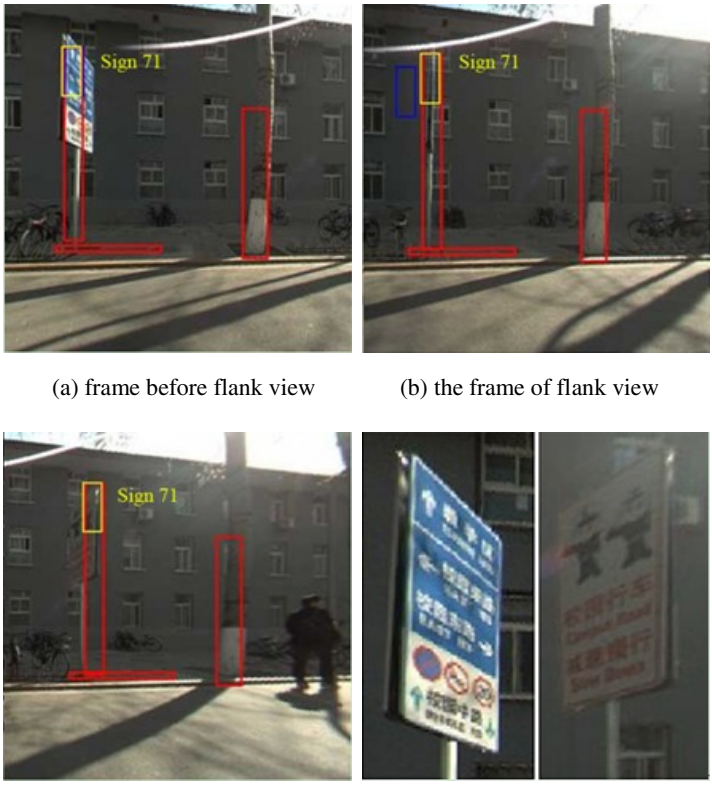
5 Experiment Evaluation

The panoramic videos used in our experiments are captured by Pointgrey Ladybug3 panoramic device in university campus, high ways etc. with the frame rate 15fps. The

resolution of the panoramic video is 3000*1500, and each side image in cubic format is 512*512. During the whole tracking process, side images are expanded so that the traversing between side images is avoided. This will bring some distortion to the boundaries of the side images but don't affect the proposed methods.

The starting of our work comes from the application requirements of road sign labeling and management by the high way management bureau of ShanXi province, China. Two examples of road sign tracking are demonstrated in Fig. 6 and Fig. 7. Each one shows three critical frames of the tracking process as (a) - (c) together with the comparison of the sign's front and back views in (d).

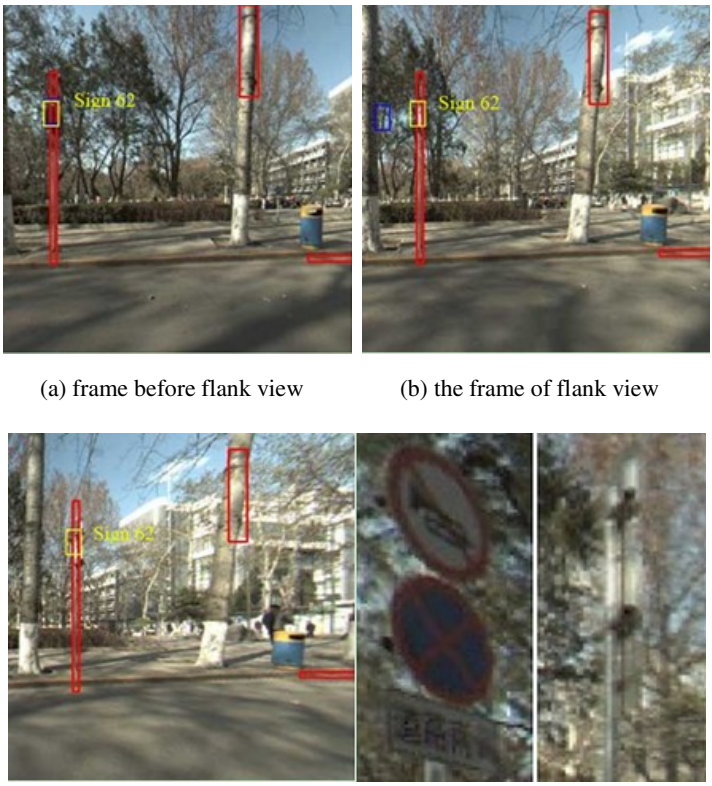
We can see that the sign's front view and back view have different texture in Fig. 6. There are more complex background and illumination in Fig. 7, which bring more difficulties for regular tracking. The blue and red rectangles in the figures denote the main and auxiliary trackers, and the yellow one is the final tracking zone. The two examples show that our method can perform well in these conditions.



(a) frame before flank view (b) the frame of flank view

(c) frame after flank view (d) Sign's Front vs. Back views

Fig. 6. Example 1 of road sign tracking



(a) frame before flank view (b) the frame of flank view

(c) frame after flank view (d) Sign's Front vs. Back views

Fig. 7. Example 2 of road sign tracking (yellow: final tracking zone)

The performance experiment is conducted in a laptop with CPU Intel core2 duo 2.0GHz and 1G memory. The motion vectors can be provided as a given condition by the video. The algorithm mainly includes two parts, the block clustering and tracking of the main tracker. Table 1 gives the time cost in videos of different scenes. It's sufficient for real-time tracking in panoramic videos. The algorithm could be further optimized with parallel programming.

Table 1. Time cost in videos of different scenes (ms)

Sequences	Clustering for Auxiliary Trackers	Mean-shift Tracking for Main tracker	Total
Campus	23	12	35
Street	30	13	43
High Way	12	12	24

6 Conclusion

An algorithm of persistent object tracking in road panoramic videos has been proposed in this paper. This algorithm could track static objects from its emerging to vanishing regardless of their appearance change or occasional disappearance, which preserves several auxiliary trackers to guide the tracking. Experimental results show that the method is effective in static object tracking such as road signs in panoramic videos. This algorithm could use the motion vectors in video coding and is very easy to perform. It's especially valuable for road sign labeling and management.

However, current tracking zone is only one of the clusters of the same MVs. Due to the noises or errors in motion estimation, the final zone isn't exactly consistent to the target boundaries. Further improvement could be made if the similar MVs are clustered.

Acknowledgement. This work is supported by the National 863 Program of China under Grant No.2012AA011801, the Natural Science Foundation of China under Grant No.61170188, and the National 973 Program of China under Grant No. 2009CB320805.

References

1. Yuan, P.-H., Yang, K.-F., Tsai, W.-H.: Real-Time Security Monitoring Around a Video Surveillance Vehicle With a Pair of Two-Camera Omni-Imaging Devices. Vehicular Technology, 3603–3614 (October 2011)
2. Koch, A., Dipanda, A., Bourgeois-Republique, C.: 3D Panoramic Reconstruction with an Uncalibrated System of Stereovision Using Evolutionary Algorithms. In: Signal Image Technology & Internet Based Systems, SITIS (2008)
3. Zhou, Z., Niu, B., et al.: Static Object Tracking in Road Panoramic Videos. In: International Symposium on Multimedia, ISM (2010)
4. Ross, D.A., Lim, J., Lin, R., Yang, M.: Incremental learning for robust visual tracking. International Journal of Computer Vision, IJCV 77, 125–141 (2008)
5. Han, B., Joo, S., Davis, L.S.: Probabilistic fusion tracking using mixture kernel-based Bayesian filtering. In: International Conference on Computer Vision, ICCV (2007)
6. Du, W., Piater, J.: A Probabilistic Approach to Integrating Multiple Cues in Visual Tracking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 225–238. Springer, Heidelberg (2008)
7. Stenger, B., Woodley, T., Cipolla, R.: Learning to track with multiple observers. In: Computer Vision and Pattern Recognition, CVPR (2009)
8. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: Computer Vision and Pattern Recognition, CVPR (2010)
9. Okuma, K., Taleghani, A., de Freitas, N., Little, J.J., Lowe, D.G.: A Boosted Particle Filter: Multitarget Detection and Tracking. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004, Part I. LNCS, vol. 3021, pp. 28–39. Springer, Heidelberg (2004)
10. Yin, Z., Collins, R.: Object tracking and detection after occlusion via numerical hybrid local and global mode seeking. In: Computer Vision and Pattern Recognition, CVPR (2008)
11. Abramson, H., Avidan, S.: Tracking through scattered occlusion. In: Computer Vision and Pattern Recognition Workshops, CVPRW (2011)
12. Sun, D., Roth, S., Black, M.J.: Secrets of Optical Flow Estimation and Their Principles. In: Computer Vision and Pattern Recognition, CVPR (2010)