APPEARANCE BASED OBJECT TRACKING IN STEREO SEQUENCES

Olga Zoidi, Nikos Nikolaidis, Ioannis Pitas

Department of Informatics
Aristotle University of Thessaloniki
Box 451, Thessaloniki 54124, GREECE
{ozoidi,nikolaid,pitas}@aiia.csd.auth.gr

ABSTRACT

A novel algorithm is proposed, that performs tracking of rigid objects in 3D videos, without knowledge of the camera calibration parameters, by exploiting only visual information obtained from the left and right video channels, namely luminance and disparity information. The proposed algorithm exploits noisy disparity maps that have been extracted by a real-time disparity estimation algorithm. The algorithm employs two appearance-based representation methods for describing the object texture. The first one combines luminance with disparity information and the second one employs Local Steering Kernel (LSK) descriptors.

Index Terms— stereo object tracking, disparity maps, local steering kernels

1. INTRODUCTION

The task of visual object tracking refers to the identification of moving objects' trajectories in videos. Exploitation of the extracted trajectories occurs in a wide range of applications in computer vision [1][2]. Traditionally, visual object tracking is applied to monocular videos acquired in the single-camera setting. However, the replacement of single-camera systems from multi-camera ones created the need for developing visual object tracking algorithms exploiting information from multiple videos [3]. The most common multiview-camera setting is the one that consists of a stereo camera. These systems exploit the additional information obtained by exploiting the stereo geometry, namely the disparity information. In the stereo configuration, object tracking may be performed on either the left or the right [4], or both [5] video channels.

The majority of stereo object tracking algorithms operate on videos captured by fixed position cameras in constrained environments with known calibration parameters [3][6][7][8]. However, the vast majority of the available stereo videos, coming from 3D cinema and 3D television, or home-made videos captured from low-cost stereo cameras, are captured in unconstrained environments with no knowledge about the intrinsic or extrinsic parameters of the stereo system. Therefore, such tracking algorithms cannot be applied to these videos.

In this paper, we present a stereo object tracking algorithm, that operates concurrently on the left and right video channel of the stereo camera. The algorithm exploits low quality stereo information obtained from a real-time disparity estimation algorithm, for predicting the object position separately in the left and right channel and for ensuring stereo consistency of the object displacement in the two channels. The proposed algorithm can be applied to any video captured from a stereo camera, in any environmental setting, requiring no knowledge about the camera calibration parameters. Similar stereo tracking approaches that rely only on visual information are the ones in [4] and [5]. The tracking algorithm in [4] is designed for person tracking only (consists of face detection+skin color segmentation) in a single video channel plus disparity framework, while our proposed method performs generic object tracking concurrently on the left and right video channels of the stereo system. The tracking algorithm in [5] performs generic object tracking by exploiting disparity information in the calculation of the object 3D velocity while the proposed algorithm incorporates disparity information in the object appearance model.

The proposed algorithm performs tracking of rigid objects in 3D videos, without knowledge of the camera calibration parameters, by exploiting only visual information obtained from the left and right video channels, namely luminance and disparity information. Disparity is the displacement (in pixels) of the projection of a 3D point in the left and right video channel [9]. The proposed algorithm exploits low quality disparity maps that have been extracted by a real-time disparity estimation algorithm [10]. The algorithm employs two appearance-based representation methods for describing the object texture. The first one combines luminance with disparity information and the second one employs Local Steering Kernel (LSK) descriptors [11].

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement number 287674 (3DTVS). This publication reflects only the author's views. The European Union is not liable for any use that may be made of the information contained therein.

The paper is organized as follows. Section 2.1 presents the fusion of color and disparity information for texture representation. Section 2.2 presents the object texture description based on LSKs. Section 2.3 describes the prediction the object position separately in the left and right channel. Section 2.4 describes the selection of the stereo object pair final position. Section 3 presents the experimental evaluation of the proposed method. Finally, conclusions are drawn in section 4.

2. ALGORITHM DESCRIPTION

Tracking commences with manual initialization of the regions of the object projections (regions of interest - ROIs) on the first frame of the left and right video channels. In the following, we define as stereo ROI pair the object ROIs on the left and right frame that correspond to the same time instance. Then, tracking proceeds in two successive steps. First, candidate object ROIs are extracted individually for the left and right channel and, then, the results are merged in order to extract the final stereo ROI pair.

2.1. Luminance-disparity based texture representation

Disparity is the most essential information the stereo systems provide, as it provides information about the object relevant distance from the camera, i.e., it grows when the object approaches the camera. In the proposed algorithm, disparity information is combined with luminance information in 2-dimensional color-disparity histograms (2D-CDH), for object texture representation. Since the available disparity maps contain a significant amount of noise, we employ coarse color-disparity histograms with 16 bins in each dimension. The color bins widths are selected uniformly in the range $[0, 255]$, while the selection of the disparity bins is performed as follows:

1. Set the first bin width from 0 to the minimum disparity value of the first frame.
2. Set the sixteenth bin width from the maximum disparity value in the first frame to the maximum disparity value of the entire video.
3. Set the width of the second to fifteenth bins uniformly in the range from the minimum to the maximum disparity value of the first frame.

For each object ROI, three 2D-CDHs are extracted, $\mathbf{H}_R$, $\mathbf{H}_G, \mathbf{H}_B \in \Re^{16\times16}$ for the red, green and blue component, respectively in the RGB color space.

Generally, 2D-CDHs are sensitive to illumination variations, changes in the object view point and object displacement with respect to the camera position, therefore they vary throughout the video duration. However, by considering that these changes are small between two consecutive frames, 2D-CDHs can be exploited for a coarse discrimination of the object ROI from the background, as will be described in Subsection 2.3.

2.2. Local Steering Kernel based texture representation

Local Steering Kernels (LSKs) are local image texture descriptors that measure the similarity of a pixel with its $P \times P$ surrounding pixels, taking into account both pixel value and pixel distance information:

$$k(\mathbf{y}_l - \mathbf{y}) = \frac{\sqrt{\det(\mathbf{C}_l)}}{2\pi} \cdot \exp\left\{-\frac{(\mathbf{y}_l - \mathbf{y})^T \mathbf{C}_l (\mathbf{y}_l - \mathbf{y})}{2}\right\}, \quad (1)$$

$l = 1, \dots, P^2$, where $\mathbf{y}, \mathbf{y}_l \in \mathcal{Z}^{+2}$ are the vectors of the center pixel and the neighboring pixel coordinates, respectively. $\mathbf{C}_l$ is the covariance matrix of the pixels' gradients. The LSK object representation is invariant to small object changes in appearance that occur between two consecutive video frames. In the proposed algorithm, a more detailed object texture representation is determined by an object model, consisting of the object LSK representation in the first video frame and k additional object LSK representations, presenting representative object instances in previous frames. A separate object model is defined for each video channel. LSKs are more discriminative texture descriptors than 2D-CDHs, therefore they are employed for a more detailed object search, as it will be described in Subsection 2.3.

2.3. Single-channel candidate object ROIs extraction

In the first step of the algorithm, candidate object ROIs are extracted individually for the left and right channel. At first, the object translation for frame t to frame $t + 1$ is performed. Since this stage is not crucial for the tracking performance, a simple first order Kalman filter [12] is employed. Centered at the predicted position $\hat{\mathbf{y}}_{t+1}$, a search region is defined with dimensions $S_x \times S_y$ proportional to the object dimensions $Q_x \times Q_y$, $S_x \times S_y = aQ_x \times aQ_y$. The constant parameter a regulates the search region size and takes small values (e.g. $a = 1.5$), for slow/smooth object movements and larger values (e.g. $a = 2$) for faster/complicated object movements.

The search region determines the area in which the new object position will be searched. This object position may be determined by exhaustive search, however exhaustive search is computationally demanding. Therefore, subsampling of the search region is performed, by selecting randomly only $n << (S_x - Q_x + 1) \cdot (S_y - Q_y + 1)$ candidate object ROIs, according to:

$$\mathbf{Y}_{t+1} = \{\mathbf{y}^1_{t+1}, \dots, \mathbf{y}^n_{t+1}\} \sim N(\hat{\mathbf{y}}_{t+1}, \mathbf{\Sigma}), \quad (2)$$

where $\mathbf{\Sigma} = \text{diag}[S_x/m, S_y/m]$. A typical value for m is $m = 4$.

After the candidate object ROIs are determined, a coarse search for the new object position is performed based on 2D-CDHs. More specifically, the 2D-CDHs of the candidate object ROIs are computed and compared to the 2D-CDHs of the object ROI at frame t, by column-stacking the 2D-CDHs and applying cosine similarity. Then, 80% of the candidate object ROIs with the lowest 2D-CDH similarity to the object at frame t are discarded and a more detailed search is performed to the remaining candidate object ROIs, based on LSK similarity to the object model defined in Subsection 2.2. For the i-th candidate object ROI, its LSK similarity is computed by the average cosine similarity to the LSK representations of the object model, according to:

$$v_i = \lambda \frac{c_{i1}^2}{1 - c_{i1}^2} + \frac{1-\lambda}{k} \sum_{j=2}^{k} \frac{c_{ij}^2}{1 - c_{ij}^2} \in [0, +\infty), \qquad (3)$$

where c_{ij} is the cosine similarity between the LSK representations of the i-th candidate object ROI and the j-th LSK representation of the object model. λ is a parameter that determines the weight of the LSK similarity to the object LSK representation in the first frame. A typical value for λ is 0.5.

This procedure is repeated for the left and right channels. The candidate object ROI with the highest LSK similarity to the object model is considered the best candidate object ROI for the corresponding channel. Finally, the new object position is extracted by combining the results of the two channels, as will be explained in the following Subsection.

2.4. Stereo ROI pair extraction

After the best object ROI from the left and right channel are extracted, disparity information is exploited, in order to take the final decision on the new object position. Let us denote by $\mathbf{y}^l$, $\mathbf{y}^r$, the best object ROI position in the left and right channel, respectively, with corresponding LSK similarities $v^l_{\mathbf{y}^l}$, $v^r_{\mathbf{y}^r}$ and mean disparity values $\bar{d}_{\mathbf{y}^l}$, $\bar{d}_{\mathbf{y}^r}$ pixels. Two candidate stereo ROI pairs are extracted. The first (second) stereo ROI pair is generated from $\mathbf{y}^l$ ($\mathbf{y}^r$) by selecting the object ROI in the right (left) channel displaced horizontally $\bar{d}_{\mathbf{y}^l}$ ($\bar{d}_{\mathbf{y}^r}$) pixels to the right (left). This selection of the stereo ROI pairs ensures the stereo consistency of the of the object displacement in the two channels. The final decision of the new object position $\mathbf{y}_{t+1}$ is taken according to the LSK similarities of the generated ROI pairs to the object models:

$$\mathbf{y}_{t+1} = \mathbf{y}^{j'} = \arg\max_{\mathbf{y}^j} \left\{ \frac{1}{2} \left(v^r_{\mathbf{y}^j} + v^l_{\mathbf{y}^j} \right) \right\}, \quad j = r, l. \qquad (4)$$

Finally, the object models are updated with the new object ROI pair every time the maximum LSK similarity drops under a predetermined threshold.

3. EXPERIMENTAL RESULTS

The performance of the proposed algorithm was tested in two videos captured by a commercial stereo camera. The video resolution was 1920×1080 pixels per channel. The method employed for extracting the disparity maps is described in [10]. The initialization of the tracking algorithm was accomplished with the object detector described in [11]. The proposed framework takes into consideration the information obtained from both the right and left videos, leading to a stereo-consistent representation of the tracking result, i.e., the drawn bounding boxes in the left and right frames can be viewed in a 3-D display monitor as a single stereo bounding box. The significance of the incorporation of disparity information in the stereo tracking algorithm is examined by comparing the performance of the stereo tracker to the performance of three state of the art appearance based single channel trackers: CH tracker [13], which is based on color histogram information and particle filtering, L1 tracker [14], which is based on sparse representation of the object and CT tracker [15], which performs real-time compressive tracking.

The task in the first stereo sequence (Figure 1) is to track a rigid object that moves smoothly, with small changes in appearance due to scaling (the object moves towards and away from the camera), changes in the view angle and partial occlusion (by the hands). We notice that only the stereo tracker and the PF tracker were able to track the object in the entire video duration, however the stereo tracker performs more accurate tracking. In the second experiment (Figure 2), the task is to track a rigid object (a helmet) with constant changes in the view angle, small scale variations and partial occlusion. We notice that the stereo tracker and the CT tracker where able to keep track of the object in the entire video duration. The PF tracker was able to track the object correctly only in the left channel, since in the right channel the tracker kept reducing the object size.

4. CONCLUSION

A novel stereo object tracking was presented, that employs a coarse and a more detailed appearance-based representation of the object texture. The proposed algorithm can be applied to any video captured from a stereo camera, in any environmental setting, requiring no knowledge about the camera calibration parameters. The proposed framework achieves a stereo-consistent representation of the tracking result, i.e., the drawn bounding boxes in the left and right frames can be displayed in a 3D monitor as a single stereo bounding box. Experimental results showed the ability of the proposed stereo tracker to track rigid objects under appearance changes and partial occlusion.

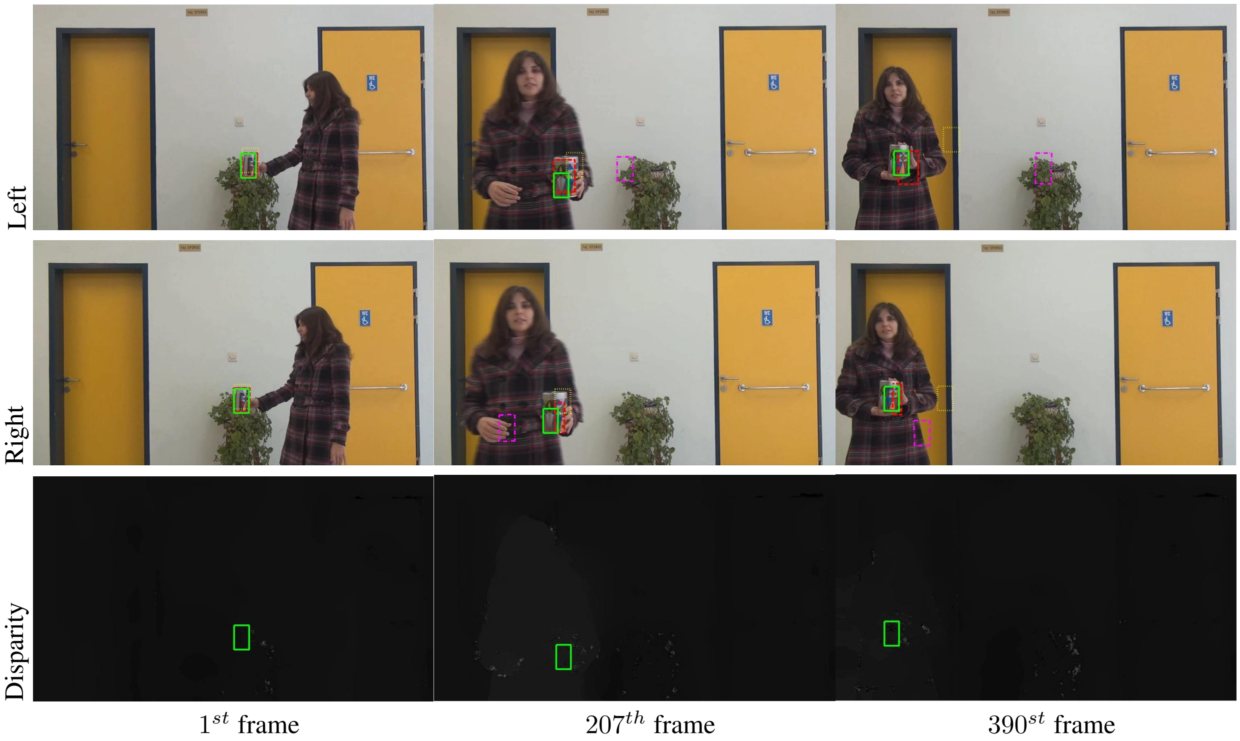

Fig. 1. Tracking results of a rigid object with small changes in appearance and partial occlusion. Solid bounding box: stereo tracker, dotted bounding box: CT tracker, dashed bounding box: PF tracker, dash-dot bounding box: L1 tracker.

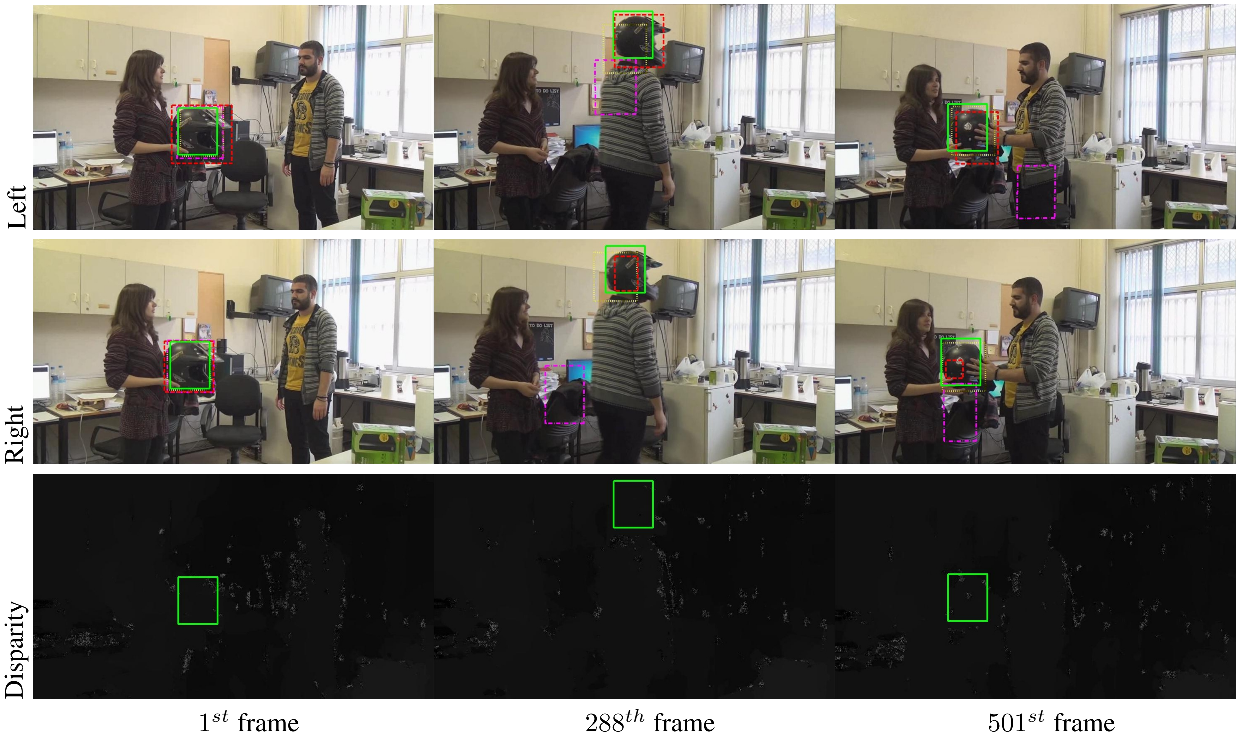

Fig. 2. Tracking results of a rigid object with constant changes in view angle. Solid bounding box: stereo tracker, dotted bounding box: CT tracker, dashed bounding box: PF tracker, dash-dot bounding box: L1 tracker.

5. REFERENCES

[1] A.E. Johnson, S.B. Goldberg, Yang Cheng, and L.H. Matthies, "Robust and efficient stereo feature tracking for visual odometry," in *IEEE International Conference on Robotics and Automation*, May 2008, pp. 39 –46.

[2] J. Wang, G. Bebis, and R. Miller, "Robust video-based surveillance by integrating target detection with tracking," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, June 2006, pp. 137–145.

[3] Ling Cai, Lei He, Yiren Xu, Yuming Zhao, and Xin Yang, "Multi-object detection and tracking by stereo vision," *Pattern Recognition*, vol. 43, no. 12, pp. 4028 – 4041, 2010.

[4] T. Darrell, G. Gordon, M. Harville, and J. Woodfill, "Integrated person tracking using stereo, color, and pattern detection," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 175–185, 2000.

[5] Eduardo Parrilla, Jaime Riera, Juan-R. Torregrosa, and Jos-L. Hueso, "Handling occlusion in object tracking in stereoscopic video sequences," *Mathematical and Computer Modelling*, vol. 50, no. 56, pp. 823 – 830, 2009.

[6] Michael Harville, "Stereo person tracking with adaptive plan-view templates of height and occupancy statistics," *Image and Vision Computing*, vol. 22, no. 2, pp. 127 – 142, 2004.

[7] Rafael Muñoz Salinas, Eugenio Aguirre, and Miguel García-Silvente, "People detection and tracking using stereo vision and color," *Image and Vision Computing*, vol. 25, no. 6, pp. 995–1007, 2007.

[8] Feng Tang, M. Harville, Hai Tao, and I.N. Robinson, "Fusion of local appearance with stereo depth for object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2008, pp. 1 –8.

[9] Emanuele Trucco and Alessandro Verri, *Introductory Techniques for 3-D Computer Vision*, vol. 93, Prentice Hall, 1998.

[10] Sergey Kosov, Thorsten Thormhlen, and Hans-Peter Seidel, "Accurate real-time disparity estimation with variational methods," in *Advances in Visual Computing*, vol. 5875 of *Lecture Notes in Computer Science*, pp. 796–807. 2009.

[11] H.J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, Sept. 2010.

[12] G. Welch and G. Bishop, "An introduction to the kalman filter," in *University of North Carolina at Chapel Hill, Tech. Rep. TR95041*, 2000.

[13] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Transactions on Image Processing*, vol. 13, pp. 1434–1456, 2004.

[14] Xue Mei and Haibin Ling, "Robust visual tracking using l_1 minimization," in *IEEE 12th International Conference on Computer Vision*, Oct. 2009, pp. 1436 –1443.

[15] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, "Real-time compressive tracking," in *European Conference on Computer Vision*, Oct. 2012.