Cube : A multi-view localization framework for 3D object analysis in computer vision applications

Steven Verstockt, Glenn van Wallendael, Sarah de Bruyne, Chris Poppe, Peter Lambert and Rik van de Walle

Proceedings of Computer Graphics International (CGI), Short papers, 2010.

To refer to or to cite this work, please use the citation to the published version:

Steven Verstockt, Glenn van Wallendael, Sarah de Bruyne, Chris Poppe, Peter Lambert and Rik van de Walle (2010). Cube : A multi-view localization framework for 3D object analysis in computer vision applications. Proceedings of Computer Graphics International (short papers).

Cube: a multi-view localization framework for 3D object analysis in computer vision applications

Steven Verstockt*†, Glenn Van Wallendael*, Sarah De Bruyne*,
Chris Poppe*, Peter Lambert* and Rik Van de Walle*

* Department of Electronics and Information Systems
Multimedia Lab, Ghent University – IBBT
Gaston Crommenlaan 8, bus 201,
B-9050 Ghent-Ledeberg, Belgium
Email: steven.verstockt@ugent.be

† University College West Flanders, Ghent University Association
Graaf Karel de Goedelaan 5
8500 Kortrijk, Belgium

***Abstract*—Precise object localization and volume estimation from a single viewpoint is a very difficult task. Furthermore, analysis of the real motion in only one view is (quasi)-impossible. To accomplish these valuable object analysis steps, information of multiple cameras is analyzed using a new localization framework. The framework merges the single-view detection results of the multiple cameras by homographic projection onto multiple horizontal and vertical planes, which slice the scene. The crossings of these slices create a 3D grid of virtual sensor points. Using this grid and subsequent spatial and temporal 3D clean-up filters, information about 3D object location, size, and motion can be instantly extracted from the video data. The novel aspect in the proposed framework is the 3D grid creation, which is a 3D extension of Arsic's multiple plane homography. By using the grid, more accurate localization and object modelling is possible. Also the use of dynamic camera maps, and spatial and temporal 3D filters, provides a more reliable object analysis.**

***Index Terms*—3D object localization, motion analysis, plane slicing, homography, 3D filtering, multi-view video analysis, dynamic camera maps**

I. Introduction

Single-view object analysis algorithms are able to accurately detect moving objects and visually track them during an event. Unfortunately, due to visibility problems, e.g. occlusion and shadows, and due to limitations in 2D → 3D reconstruction, crucial information about the object location, size, and motion trajectory is hard to retrieve. In order to retrieve these valuable object characteristics, we propose a new multi-view localization framework that detects the 3D position and volume of an object in an accurate way.

Multi-view analysis of moving objects has already been studied by many authors and the majority of the existing work relies on homographic projection [1] of camera views. For example, in [2-4] homography is used to project single-view detections of moving objects onto a common ground plane, where intersecting regions indicate object positions. Although these approaches seem promising, problems arise when objects do not touch the ground plane or when objects are split up in several parts. Further, these methods suffer with a high number of false positives in overcrowded scenes. Applying the homographic transformation in multiple horizontal layers, as is done in the work of Arsic [5], Khan [6], and Lai [7], solves the majority of the reported problems and provides a more precise 3D reconstruction. To further improve these approaches, we propose a 3D extension to the horizontal plane slicing. By also slicing in the two vertical directions, more accurate localization and volume estimation is achieved.

Fig. 1 shows the building blocks of our multi-view localization framework. First, the framework detects the moving objects in each single view. Since the focus of this paper is on the framework, and not on the detection of objects itself, this first step is not covered in this paper. Secondly, the single-view detection results of the available cameras are projected by homography [1] onto horizontal and vertical planes which slice the scene. Next, the multi-view plane slicing algorithm accumulates the multi-view detection results on each of the horizontal and vertical planes. Then, a grid of virtual multi-camera sensors, i.e. the Cube, is created at the crossings of these planes. At each sensor point of the 3D grid, the detection of the horizontal and vertical planes that cross in that point are accumulated. Finally, 3D spatial and temporal filters clean-up the grid and remove the remaining noise.

The remainder of this paper is organized as follows. Section 2 presents the homographic projection and multi-view plane slicing algorithm. Subsequently, Section 3 describes in detail all the steps required to complete the 3D object localization, i.e. the grid accumulation and clean-up post-processing. Further, in Section 4, experimental results for 3D body modelling are shown. Section 5 ends this paper with conclusions.

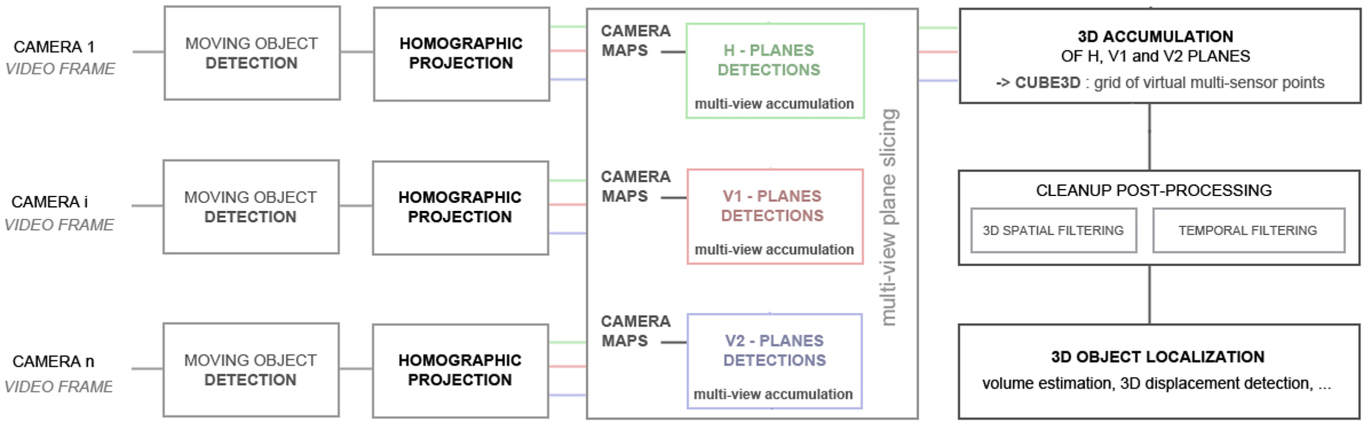

Fig. 1. General scheme of our multi-view localization framework

II. Homographic projections and multi-view plane slicing

A. Homographic projections

Core of our localization framework is the planar homography constraint [2, 3], i.e. a multi-view localization technique that we translated into the 3D structure of the Cube. This constraint implies that only points corresponding to ground plane locations of detected moving regions will consistently warp to foreground (FG) points in every view. Points that violate this assumption result in mapping to a skewed location on the projection plane. It must be pointed out that this homography constraint is not limited to the real ground plane. The constraint also holds for the homographic projection of virtual planes parallel and orthogonal to the real ground plane, i.e. the planes that form the basis of the Cube. The creation of these planes is based on basic geometry and is fairly straightforward. Before going further into detail on this, we discuss the basics of homographic projection, which forms the basis of our extended multi-view plane slicing.

To project the FG views, i.e. the moving parts of the single-view camera images, to a common (ground) plane, the homography matrix Hom of each camera is needed. The matrix can be calculated offline using techniques such as the 4-point-based Direct Linear Transform [1] or even methods for self-calibration can be used. The proposed framework uses the former method. The homographic projection of single-view detection results, i.e., the second step in the localization framework, uses the homography matrix to map each FG point (x, y) onto the point (x', y') in the common ground plane. To know in how many views (x', y') is foreground, (x', y') is incremented for each mapping.

As soon as the mapping of FG views is finished, object locations are detected on the common (horizontal) ground plane $Plane_H$ by finding points (x', y') that are FG, i.e. corresponding to a value of 1, in more than half of the views F_v which can see the location (Eq. 1). Points that do not meet this requirement are labeled as background (BG), corresponding to a value of 0.

Since object regions are not always visible in all n_V views or only partly visible, the localization will be influenced by the number of cameras having the position under surveillance. By making a camera map $nCAM_H$ of the ground plane, that contains the number of cameras that are able to monitor the specific position, appropriate detection criteria can be determined. The higher the value of a position on the map, the more cameras that monitor that position.

B. Multiple plane homography

By detecting the presence of FG points on different virtual planes orthogonal and parallel to the ground plane, the precise 3D locations of objects can be retrieved. To be able to do this, the homography matrices of these virtual planes need to be known. Since selecting calibration points for each of these planes is too time-consuming and error-prone, a technique is needed to automatically generate the homography of these planes. From the few horizontal multiple plane strategies that have recently been proposed in literature [5-7], the computational low multilayer homography by Arsic et al. [5] is most interesting to compute the homography matrices of the virtual planes. As in [5], the computational effort for creating the multiple plane homographies is kept at a minimum. Starting from eight calibration points, the homography is computed for the six reference planes connecting these points, i.e. two horizontal and four vertical planes. All other homographies, for the planes parallel to those calibration planes, are computed by basic geometry. As soon as the homography of each view to the common plane is known, the accumulative object detection on that plane is performed. The accumulative detections (Eq. 2 and Eq. 3) for the two vertical directions are similar to the horizontal detection described in (Eq. 1). Also here, camera maps are needed to determine appropriate detection criteria.

The plane slicing virtually cuts the scene in the horizontal and vertical directions. Each of these cuts represents an accumulated detection plane, i.e. a weighted combination of the projected multi-view detection results to that plane. By observing these detection planes, it is already possible to get

$$Plane_H[x',y'] \rightarrow \begin{cases} FG, & \text{if } \sum\limits_{v=0:n_v} F_v(Hom^{-1}_{F_v \rightarrow H}[x',y'])) > \dfrac{nCAM_H[x',y']}{2} \\ BG, & \text{otherwise} \end{cases} \tag{1}$$

$$Plane_{V1}[y',z'] \rightarrow \begin{cases} FG, & \text{if } \sum\limits_{v=0:n_v} F_v(Hom^{-1}_{F_v \rightarrow V1}[y',z'])) > \dfrac{nCAM_{V1}[y',z']}{2} \\ BG, & \text{otherwise} \end{cases} \tag{2}$$

$$Plane_{V2}[x',z'] \rightarrow \begin{cases} FG, & \text{if } \sum\limits_{v=0:n_v} F_v(Hom^{-1}_{F_v \rightarrow V2}[x',z'])) > \dfrac{nCAM_{V2}[x',z']}{2} \\ BG, & \text{otherwise} \end{cases} \tag{3}$$

$$Cube[x',y',z'] \rightarrow \begin{cases} FG, & \text{if } Plane_H[x',y'] + Plane_{V1}[y',z'] + Plane_{V2}[x',z'] \geq 2 \\ BG, & \text{otherwise} \end{cases} \tag{4}$$

an idea about the object's location and size. But in order to ensure accurate localization and to automatically provide easy-interpretable data for motion analysis, further processing of these plane detections is needed. For this reason the 3D Cube, which is described in Sect. 3, is created.

III. Grid Accumulation and Clean-up Filtering

The grid accumulation combines the detection results of the horizontal and vertical planes. At the crossings of these planes, a grid, i.e. the $Cube$ (Fig. 2), is formed. The $Cube$ consists of virtual 3D sensor points (x, y, z) at which the detection results, i.e. the corresponding FG value 1 and BG value 0, of the horizontal $Plane_Z$ and vertical planes $Plane_{V1}$ and $Plane_{V2}$ are accumulated. Sensor points with a value equal or greater than two are labeled as FG (Eq. 4). This constraint implies that only points that are detected as FG on at least two of the three planes, which cross in that point, are considered as FG. As such, misdetection in one of the planes is filtered out and only stable detections stay. This way, the multi-plane fusion methodology eliminates errors caused by misclassification during the previous steps.

For higher detection robustness, the framework finalizes with a clean-up filtering step in the spatial and temporal domain. In the spatial domain, it filters out noisy FG points in the $Cube$ with a set of weighted 3D median filters and fills up holes in FG regions with a 3D filling operator. The temporal filtering on its turn removes objects which have no overlap with detected objects in the previous or subsequent Cube.

IV. Experimental Results

In order to verify the proposed method, the generic framework is used to extract the body model of a person (Fig. 3. For each test person, his length, hip width, and hip perimeter were manually measured, i.e. the volume ground truth GT_l, GT_{hw}, and GT_{hp} respectively. Also, the ground position GT_P of the test person was determined. By comparing these ground truth values with the values derived from the framework, i.e. l, hw, hp, and P, we obtain the volume and position error VE and PE for each person. As expressed in Eq. 5, the volume error equals the sum of the differences between the hip perimeter, the length and the hip width of the GT and the detected region. The position error is the Euclidean distance between GT_P and P. By averaging VE and PE over all n_t tests, using the formulas shown in Eq. 6, the average volume error AVE and the average position error APE are determined. As the results in Table 1 indicate, the proposed algorithm yields good results for all the test persons. The obtained AVE and APE of 11 and 7 centimeter (cm) are very satisfactory.

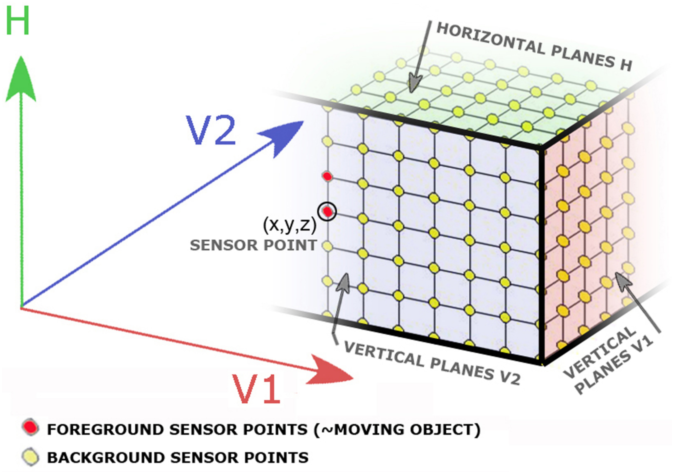

Fig. 2. Grid accumulation of horizontal and vertical planes.

$$\begin{aligned} VE &= |GT_{hp} - hp| + |GT_l - l| + |GT_{hw} - hw| \\ PE &= \sqrt{(GT_{P_x} - V_{P_x})^2 + (GT_{P_y} - V_{P_y})^2} \end{aligned} \tag{5}$$

$$\begin{aligned} AVE &= 1/n_t \sum\nolimits_{i=1}^{n_t} VE_i \\ APE &= 1/n_t \sum\nolimits_{i=1}^{n_t} PE_i \end{aligned} \tag{6}$$

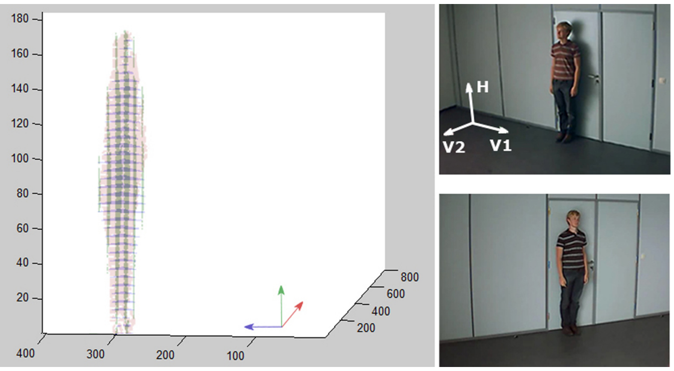

Fig. 3. Cube experiment for body modeling.

Person	**Volume** $l; hp; hw$ (cm)	**VE**	**Position** P_x, P_y (cm)	**PE**
1	187 ; 112 ; 47	**14**	291 ; 26	**8**
GT	(185 ; 120 ; 51)		(285 ; 21)	
2	178 ; 108 ; 47	**6**	277; 17	**9**
GT	(179 ; 112 ; 46)		(285 ; 20)	
3	172 ; 134 ; 53	**13**	282 ; 24	**5**
GT	(176 ; 128 ; 50)		(285; 28)	
4	186 ; 111; 44	**10**	278; 25	**7**
GT	(184 ; 116; 47)		(285 ; 23)	
	AVE =	**11**	**APE =**	**7**

TABLE I
BODY MODELLING RESULTS

V. Conclusion

In this work we have proposed a multi-view framework for 3D object localization, which is mainly based on 3D extensions to homographic plane slicing by Arsic [5]. The framework merges single view detection results of multiple cameras by homographic projection onto multiple horizontal and vertical planes which slice the scene under surveillance. At the crossings of these slices, we create a 3D grid of virtual sensor points. At each of these points, the detection results of the crossing planes are accumulated and compared to a dynamic camera map, which is also one of the novel aspects in our work. Using this grid and the subsequent spatial and temporal 3D clean-up filters, information about 3D object location, size, and motion can be extracted very accurately from the video data. This information can be very useful in a broad range of aplications, e.g. motion capturing for computer graphics animations [8], object tracking and abnormal event detection for smart video surveillance [9], and body modelling and dynamics analysis for medical diagnosis [10].

The novel 3D grid, dynamic camera maps, and spatial and temporal 3D filters, which extend existing 2D filter concepts, provide accurate localization and opens the door to more reliable object analysis. Experimental results confirm these findings and indicate that the multi-view localization framework performs very well. By further optimizing and extending the framework, like for example with the motion history image technique of Davis [11], we believe that many more valuable characteristics of moving objects and their events can be obtained from the framework at relatively low cost.

Acknowledgment

The research activities as described in this paper were funded by Ghent University (UGent), the Interdisciplinary Institute for Broadband Technology (IBBT), University College West Flanders, the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), and the EU.

References

[1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, 2004, pp. 87-131.

[2] S. M. Khan and M. Shah, *A Multi-view Approach to Tracking People in Crowded Scenes using a Planar Homography Constraint*, 9th European Conference on Computer Vision, vol. 4, 2006, pp. 133-146.

[3] S. Park and M. Trivedi, *Homography-based Analysis of People and Vehicle Activities in Crowded Scenes*, IEEE Workshop on Applications of Computer Vision, 2007, pp. 51-51.

[4] S. Verstockt, S. De Bruyne, C. Poppe, P. Lambert, R. Van de Walle, *Multi-view Object Localization in H.264/AVC Compressed Domain*, 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, 2009, pp. 370-374.

[5] D. Arsic, E. Hristov, N. Lehment, B. Hornler, B. Schuller, G. Rigoll, *Applying multi layer homography for multi camera person tracking*, ACM/IEEE International Conference on Distributed Smart Cameras, 2008, pp. 1-9.

[6] S. M. Khan, M. Shah, *Tracking Multiple Occluding People by Localizing on Multiple Scene Planes*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, pp. 505-519.

[7] P.-L. Lai, A. Yalmiz, *Efficient object shape recovery via slicing planes*, IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[8] T. Molet, R. Boulic, D. Thalmann, Human Motion Capture Driven by Orientation Measurements, Presence, vol. 8, 1999, pp. 101-115

[9] I. Ivanov, F. Dufaux, T. M. Ha, T. Ebrahimi, Towards Generic Detection of Unusual Events in Video Surveillance, 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, 2009, pp.61-66

[10] J.J. Wang, S. Singh, Video analysis of human dynamics: a survey, Real-Time Imaging, vol. 9, 2003, pp. 321-346

[11] J. W. Davis, Hierarchical Motion History Images for Recognizing Human Motion, IEEE Workshop on Detection and Recognition of Events in Video, 2001, pp. 39-46