TOWARDS A VISUAL-HULL BASED MULTI-AGENT SURVEILLANCE SYSTEM

Pilar Callau Uson, Kaori Hagihara, Diego Ruiz and Benoît Macq

Communications and Remote Sensing Laboratory, Ecole Polytechnique de Louvain,UCL, Belgium

ABSTRACT

We present a multi-view three dimensional intelligent surveillance system. We use a multi-agent framework to identify the behaviors of individuals in the scene. Detection and interpretation are performed completely in 3D space. A moving train coach is monitored by eight fish-eye cameras. Segmentation masks extracted from the undistorted images are fed to a distributed 3D reconstruction algorithm producing an octree-based description of the volume at each frame. Voxel-based algorithms extract connected-regions and their descriptions from consecutive models. The set of regions is mapped to a set of agents. We achieve dynamically consistent high-level interpretations by combining probabilistic models of human behaviors and intelligent reasoning.

Index Terms— Fish-eye, 3D-reconstruction, volumetric, event-detection, HMM-DBN

1. INTRODUCTION

In order to enforce the conformity of people behaviour with society rules, state operators and private companies are increasing the number and size of areas to monitor.

Nowadays, recorded sequences are mostly used in a posteriori basis to arrest and convict people. Technical and economical constraints forbid the online monitoring of each video sequence by a human operator. Our society would benefit of robust intelligent surveillance systems able to raise alarms to human operators in real time. Then, each operator may handle a group of surveillance systems and react to raised alarms while the action is taking place.

Even with the increase in computer power, image acquisition, transmission and analysis is still expensive in terms of equipment. Therefore, the number of cameras is restricted to the minimum by using fish eye lenses. Their wide angle of view maximizes the volume seen by each camera at the cost of spatial resolution.

Our goal is to design a multi-agent framework to identify the behaviors of individuals in the scene, where detection and interpretation are performed completely in 3D space. Until now most of the surveillance work has been done using 2D techniques [1], in which the main problems are to resolve occlusions and tracking of multiple people.

The authors would like to thank Region Wallone for the support through STRADA and BOSS projects.

2. SYSTEM DESCRIPTION

Eight fish-eye cameras monitor the volume of interest. Segmentation masks extracted from the undistorted images are fed to a distributed 3D reconstruction algorithm producing an octree-based description of the volume at each frame. Voxel-based algorithms extract connected-regions and related features from consecutive models, which are the inputs to an intelligent event detection system.

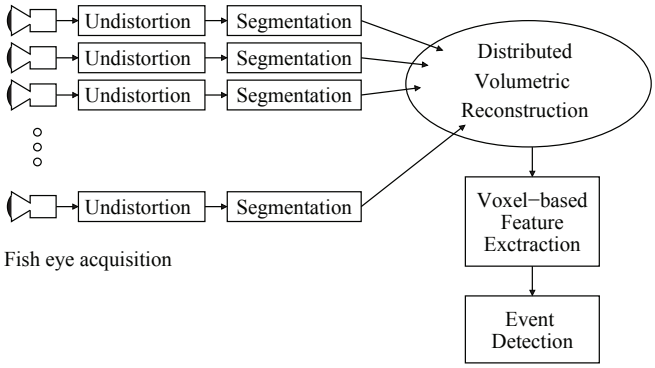

Fig. 1. General overview of the system

2.1. Fish-eye acquisition and calibration

Our 3D reconstruction system computes the projection of each voxel into each image. The current implementation uses a pinhole projection model:

$$\tilde{m} = P\tilde{X_w} \tag{1}$$

where $\tilde{X_w}$ and $\tilde{m}$ are respectively the homogeneous world and image coordinates of a 3D point, and $P = A(R|T)$ is the camera projection matrix which consists of intrinsic matrix A and extrinsic matrix $(R|T)$. Since we use fish-eye lenses, which causes strong distortions, computation of P is not straightforward. Our calibration is performed in two steps.

U.S. Government Work Not Protected by U.S. Copyright

The first step corrects image distortion by projecting images from the omnidirectional camera model to the pinhole camera model. This 2D to 2D projection provides intrinsic pinhole camera parameters. We use D.Scaramuzza algorithm to calibrate Omni-directional cameras [2]. It provides efficient results for our fisheye lenses as shown in figure 2. We observe that distortion is corrected and that the curved chess pattern is straightened.

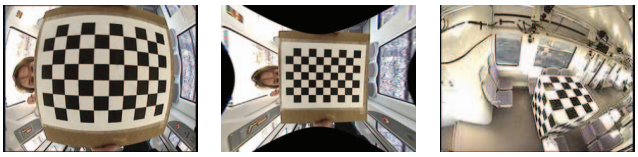

Fig. 2. Left and Middle: Example of undistortion effects Right: Extrinsic calibration target inside train

The second calibration step is the computation of extrinsic parameters. It uses a cubic object with chess painting on its faces as shown in figure 2. This object is designed to be visible from every camera, so that one can define a common 3D coordinate system shared by all cameras. We use a basic least-square method.

2.2. Distributed volumetric 3D reconstruction

The 3D reconstruction algorithm has been fully described in [3]. It uses a distributed and scalable volumetric architecture based on an efficient exploitation of inter-frame redundancy and an efficient merging of partial models. The architecture is composed of acquisition nodes reconstructing partial models from multiple views and of a master node merging partial models. The master node updates local copies of the partial models with non-redundant information from the acquisition nodes. Then it merges the partial models to produce the volumetric description of the scene. Each voxel is described by a single feature coding visibility, occupancy and subdivision of space. By adding visibility to the voxel description, we allow each camera to see only part of the volume of interest. We achieve fifteen reconstructions per second and less than 130 ms latency with eight cameras and only twelve cores.

We use a post-filtering step to remove noise due to visibility [3]. Our algorithm computes connected regions, their size and their visibility. Each region is a group of *non-outside* voxels for which there is a six-neighbourhood path between any two voxels composed of voxels of the region. The size of a region is the number of voxels composing it. The visibility of a region is the maximum number of partial models seeing a voxel of the region.

Classical visual hull implementations are based on the hypothesis that the target is entirely visible by all the cameras. Our hypothesis is that the target is seen by the highest number of partial models. We sort connected regions by their visibility and size, and keep the most visible ones as long as their size is above a threshold. The maximum number of regions kept is a configuration parameter. Our filter also erases connected regions that are seen by a single partial model. Our system ensures that a subject is correctly modeled even if different body parts are seen by different subset of cameras as long as the visible parts in each camera are correctly segmented.

The system had to be adapted in order to use the segmented undistorted fish eye images as input. As seen in figure 2, the undistorted images present regions with undefined pixel values near the borders. These regions are set to black by default. The segmentation algorithm interprets them as background. Voxel projecting onto those regions are erased by the Visual Hull. In order to prevent this, we define voxels projecting entirely inside an undefined region as invisible by the corresponding camera. Other cameras will help us decide if these voxels should belong or not to the model.

2.3. Voxel-based feature extraction

We extract spatio-temporal features out of the reconstructed models in order to drive the event detection system. The precision of the extracted information depends on the quality of the reconstructed models. Our system computes upper approximations of true shapes from the voxel-based reconstructions. Their quality depends on numerous factors including the number of cameras, their resolution, the octree resolution, the quality of the segmentation algorithm and the size of the volume of interest.

We compute the centre of gravity $\mathbf{cg}^j(t) = (cg^j(t)_x, cg^j(t)_y, cg^j(t)_z)^T$ and the axis-aligned minimal bounding box $\mathbf{BB}^j(t)$ for each connected region $R^j(t)$. Consequently, time derivative of these parameters are computed; the velocity of the center of gravity $\mathbf{v}^j(t) = \frac{d}{dt}\mathbf{cg}^j(t) = (v^j(t)_x, v^j(t)_y, v^j(t)_z)^T$ and the variation in size along each axis $\Delta\mathbf{BB}^j(t)$ of the minimal bounding box. Axis x and z are horizontal, and axis y is vertical to the ground in our system. These parameters are computed by a single traversal of the volumetric scene. They consume almost no CPU time.

Complex features can be also extracted from the sequence of volumetric models, e.g. Fourier descriptors [4] , 3D shape contexts [5] or motion capture data [6]. If the feature extraction is too complex for the master, our distributed architecture exploiting inter-frame redundancy allows a quick transfer of the reconstructed models towards another cluster.

2.4. Behaviour analysis

Behaviour understanding involves the analysis and recognition of motion patterns, and the production of high-level description of actions and interactions. In this section we develop a compositional analysis of individual behaviour by combining probabilistic models of human behaviors and

intelligent reasoning to achieve dynamically consistent high-level interpretations.

2.4.1. Multi-agent framework

We propose a multi-agent system, where the concept of agent is used to represent each tracked person. The concept of agent $\mathbf{Ag}_i(t)$ is used to represent the extracted features $\mathbf{EF}_i(t)$ of a person i and his inferred action $A_i(t)$ at time t.

$$\mathbf{EF}_i(t) = \left(\mathbf{cg}^j(t)^T, \mathbf{BB}^j(t)^T, \mathbf{v}^j(t)^T, \Delta\mathbf{BB}^j(t)^T\right)^T$$

where,

$$A_i(t) \in SA = \{\, STAND,\, WALK,\, RUN,\, SIT\,DOWN,\, GET\,UP,\, SEAT,\, DUCK,\, FALL\,DOWN,\, GO\,DOWN,\, LIE,\, BANG\}$$

The features are directly transferred from regions to agent when there is a clear one to one correspondence between regions and agents. This correspondence is established by comparing the features of agents at time $t-1$ with the features of the regions extracted at time t. This rule allows the system to create a new agent when a person enters the monitoring volume, to keep track of the person and to delete the agent when the person leaves.

A connected region may correspond to several agents if their silhouettes intersect in all the cameras in which they are visible (e.g. body contact). The system can distinguish agents being lost and temporally inconsistent by referring to their eventual status LEAVING or ENTERING the monitoring volume after computing a basic correlation. The new features of those agents are estimated from those at the time $t-1$ and from those of the corresponding region at the time t.

2.4.2. Action inference

Many research in individual behaviour recognition focus on a hierarchical structure to map features to high level actions. This decomposition is generally not deterministic, as can be stated from the many approaches proposed in literature using different statistical [7][8] or rule-based models [9].

Probabilistic graphical models as Bayesian networks are a widely used solution to approach this problem as they allow a more sophisticated analysis of data with spatio-temporal variability. They offer a good trade-off between system complexity and performance while providing a good framework for coping with small training data sets and the addition of novel behaviours to extend the system.

We have pursued a classic Hidden Markov Model dynamic Bayesian network (HMM-DBN) [10] that includes both prior knowledge of each action $Pr(action)$ and features $\mathbf{EF}_i(t)$, to probabilistically represent and infer the action $A_i(t)$ of individual agents and integrate these in time. The conditional probability distributions of the observations given an action $Pr(\mathbf{EF}_i(t)|A_i(t))$ are learnt from a set of annotated training sequences. This parameter learning is reinforced by a hierarchical analysis of the relational constraints between agent parameters and defined actions. The following is our DBN structure at time t.

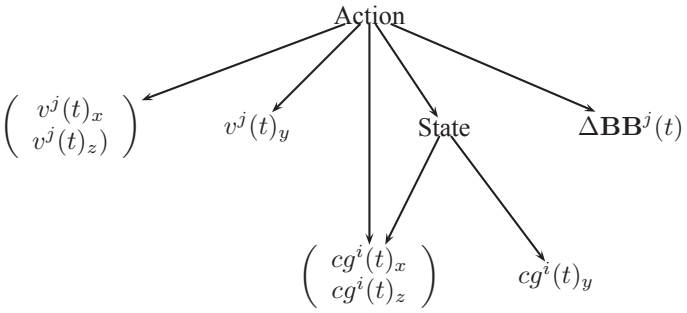

Fig. 3. DBN structure

Thus, some middle level states are created by grouping semantically equivalent actions to determine which computer vision features have stronger causal relationships with the defined actions. Moreover, action transition probabilities are also optimized in the learning stage.

Inference in the DBN consists on selecting the action with the highest probability given the observations:

$$A_i(t) = \max_{a \in SA} \frac{Pr(\mathbf{EF}_i(t)|a) * Pr(a)}{Pr(\mathbf{EF}_i(t))}$$

3. RESULTS

First, we present the 3D reconstruction of two subjected people taken in a moving train coach. Since fast variations of lighting conditions make automatic foreground segmentation difficult, the input masks to our reconstruction system were generated manually. We use an octree of maximum depth six and a volume of interest of size $2.8 * 2.5 * 4.0m^3$, which corresponds to the monitored volume of the coach. Each camera observes only part of the volume of interest. Figure 4 shows one of the volumetric description outputs of the 3D distributed reconstruction system using eight cameras. The system found two connected regions.

Next, we present the performance of our individual behavior analysis algorithm. We took 54 sequences (aproximately 20 minutes duration in total), in which actors play the defined actions SA in another system. This system was built interior, and a standard segmentation technique was sufficient enough to do full automatic 3D reconstruction and then feature extraction was performed. We defined alarm events on training sequences and evaluated with test sequences.

Table 1 shows the efficiency of our system.

However, infinite possibility of kinematic movenets arise a trade-off between the detection accuracy and the amount

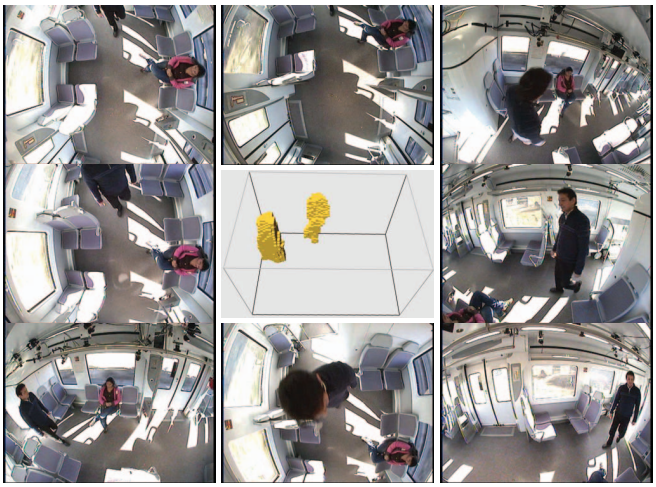

Fig. 4. Eight synchronized images at a frame and the corresponding reconstructed scene

Event	detection rate (%)
FALL DOWN	95.6
BANG	89.1
Average of the rest of defined actions	84

Table 1. Detection rate at each event from our system

of training data. Thus, our future work is collecting more suitable sequences to improve our model.

4. CONCLUSIONS

Fish-eye cameras and voxel visibility allow us to monitor part of a coach of a moving train. The distributed reconstruction algorithm automatically selects the appropriate subset of cameras to model each volume element. Furthermore, voxel visibility allows us to filter the output volumetric description, which is decomposed into connected regions.

We successfully map the set of regions to a set of active agent by using spatio-temporal information. Features extracted from the regions are transfered to the agents. We used a classic Hidden Markov Model dynamic Bayesian network (HMM-DBN) to infer individual actions from agent's features.

We plan to extend the system in order to model person-to-person interactions. The concept of agent can be extended to hold interaction parameters like the relative distance and speed difference with other agents. Our aim is to compute the probability of normal or abnormal interaction situation based on image features and the individual actions of the involved agents.

5. REFERENCES

[1] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 34, no. 3, pp. 334–352, 2004.

[2] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," *Proceedings of the Fourth IEEE International Conference on Computer Vision Systems*, 2006.

[3] D. Ruiz and B. Macq, "Exploitation of inter-frame redundancy for real time volumetric reconstruction of arbitrary shapes," *IEEE Journal of Selected Topics in Signal Processing (JSTSP), special issue on Distributed Processing in Vision Networks*, vol. 2, no. 4, pp. 556–567, August 2008.

[4] D. V. Vranic and D. Saupe, "3d shape descriptor based on 3d fourier transform," in *Proceedings of the EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services*, Budapest, Hungary, September 2001, pp. 271–274.

[5] M. Kortgen, G. J. Park, M. Novotni, and R. Klein, "3d shape matching with 3d shape contexts," *In the 7th Central European Seminar on Computer Graphics*, April 2003.

[6] Y. Sun, M. Bray, A. Thayananthan, B. Yuan, and P.H.S. Torr, "Regression-based human motion capture from voxel data," in *British Machine Vision Conference (BMVC06)*, 2006, p. I:277.

[7] Nam T. Nguyen, Hung H. Bui, Svetha Venkatesh, and Geoff West, "Recognising and monitoring high-level behaviours in complex spatial environments," in *In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR*, 2003, pp. 620–625.

[8] Yuri A. Ivanov and Aaron F. Bobick, "Recognition of visual activities and interactions by stochastic parsing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 852–872, 2000.

[9] L Xin and T Tan, "Ontology-based hierarchical conceptual model for semantic representation of events in dynamic scenes," in *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, 2005, pp. 57–64.

[10] L.R. Rabiner et al., "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.