Virtu4D: a Real-time Virtualization of Reality

Mourad Boufarguine Malek Baklouti
Vincent Guitteny
Thales Security Solutions and Services - Research Departement, FRANCE
surname.name@thalesgroup.com

Frederic Precioso
ETIS ENSEA/CNRS/Univ. Cergy-Pontoise
95014 Cergy-Pontoise, FRANCE
frederic.precioso@ensea.fr

Abstract

In video surveillance systems, when dealing with dynamic complex scenes, processing the information coming from multiple cameras and fusing them into a comprehensible environment is a challenging task.
This work addresses the issue of providing a global and reliable representation of the monitored environment aiming at enhancing the perception and minimizing the operator's effort. The proposed system Virtu4D is based on 3D computer vision and virtual reality techniques and takes benefit from both the "real" and the "virtual" worlds offering a unique perception of the scene.
This paper presents a short overview of the framework along with the different components of the design space: Video Model Layout, Video Processing and Immersive Model Generation. The final interface gathers the 2D information in the 3D context but also offers a complete 3D representation of the dynamic environment allowing a free intuitive 3D navigation.

1. Introduction

The ineffectiveness of the traditional video surveillance systems has sparked demand for a shift in the security paradigm as researchers are looking for new approaches to enhance situation awareness in video surveillance applications and monitoring systems. In a classical way (Figure 1), video surveillance systems by means of distributed architectures of fixed cameras represent the streams mosaically in a control viewer and rely on human capabilities to analyze them which needs expert eyes and can be very tiring.
Some studies about the effectiveness of human monitoring of surveillance video, carried out by the US National Institute of Justice [8] concluded that "*such a task[..manually detecting events in surveillance video], even when assigned to a person who is dedicated and well-intentioned, will not support an effective security system. After only 20 minutes of watching and evaluating monitor screens, the attention of most individuals has degenerated to well below acceptable levels. Monitoring video screens is both boring and mesmerizing. There are no intellectually engaging stimuli, such as when watching a television program*".

Breaking up with the common matrix representation to enhance the perception and minimize the operator's effort

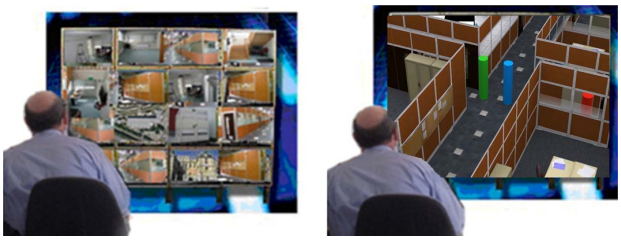


Figure 1. Virtu4D versus the classical video surveillance systems. Left: classical video surveillance representation. Right: real time 3D virtualization.

is becoming a fundamental issue.
Thus, extensive research has been carried out in order to tackle this issue [19, 5, 7]. Existing approaches tend at combining video with the 3D environment to provide coherence between video streams and spatial context. Wang et al. [22] present a comprehensive survey comparing the different techniques related to contextualized videos.
Contextualizing the video in the 3D environment needs to address several issues for virtual reality research community, mainly related to Video Processing Method, Model Processing Method, Video-Model Layout Design and Video Visualization (Navigation Design), identified by Wang et al. as video design space.

The work in this paper is presented referring to the structure of design space proposed by Wang et al. [22] and is in line with the ongoing research concerns. It addresses the issue of providing a global and reliable representation of the monitored environment.
Indeed, we consider that the increase of computational capabilities together with the emergence of affordable stereoscopic vision systems have opened new opportunities for the development of innovative approaches based on 3D computer vision and virtual reality techniques. Considering stereoscopy provides the powerful context required to build the perceptual space we propose.

Summary of the achievements and main contributions

We are presenting in this paper our approach to answer the issue of visual data representation for surveillance systems. Based on 3D computational techniques, we propose to make, in real-time, a virtual copy of the complex dynamic scene observed by video cameras. The goal behind this approach is to generate a real-time unique perception of the scene and thus making the surveillance task more intuitive and natural.

Figure 2 outlines the main components of the proposed system. The major design dimensions addressed in this work are Video Model Layout, Video Processing and Immersive Model Generation. This is consistent with the components highlighted in the unified framework proposed by Wang. The proposed structure contains the following components:

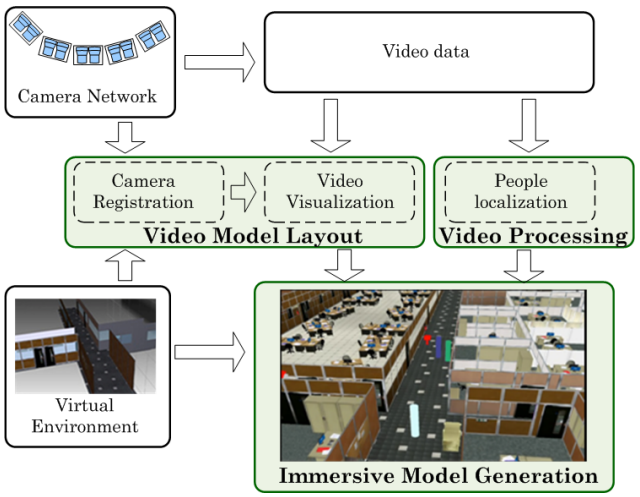


Figure 2. Overview of Virtu4D process

- Video Model Layout: addresses the issue of video data representation within the 3D context. Contextualizing the video in the 3D environment highlights some crucial issues in virtual reality research, mainly: camera registration and video visualization.
- Video Processing: aims at extracting salient information from the video data. In this step, the system detects, extracts and locates persons in order to generate a 3D dynamic virtual view of the environment.
- Immersive Model Generation: offers the final interface. The final interface gathers the 2D information (video streams) in the 3D context but also offers a complete 3D representation of the dynamic environment allowing a free intuitive 3D navigation.

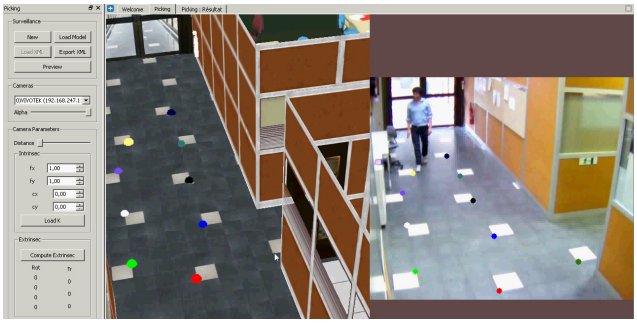
By generating a real-time unique perception of the scene, Virtu4D system provides a flexible interface with a natural visual information display. The remainder of the paper describes each of these components.

2. Video Model Layout: Network Registration and Video Visualization Issue

This section details our camera registration approach and brings forward the limitations of the existing contextualized video approaches.

2.1. Camera network registration

Camera registration (extrinsic calibration) seems obvious in the case of a single camera. However, when deal-



(a) Picking correspondences between (left) 3D model and (right) IP camera frame: matches points are represented with colored 3D spheres.

(b) Camera registration in (left) Augmented Reality mode with alpha blended video on 3D model and (right) Mixed Reality mode, the registered camera is represented by a red 3D object and the image plane in the sensor visual field of view.

Figure 3. Registration software based on model to image correlation.

ing with distributed camera network the current techniques raise some issues mainly related to the definition of a single absolute referential.

The camera registration is usually estimated by a correlation between a pattern and the image provided by camera sensor. The pose-camera corresponds then to the translation and rotation of the camera in the 3D scene referential. When dealing with a camera network, it is hard to find a unique referential in the scene that can be observed by each camera. The use of grid calibration pattern, more than being impractical, implies a prior knowledge on the absolute positions of each (moving) object in a common absolute referential.

Some works have proposed to overcome this issue by making use of natural human motion or planar trajectory estimation of one or more objects in motion to recover the relative pose of several stationary cameras.

These techniques usually suffer from the restriction that the cameras should have a common area of the scene. The calibration object should be viewed simultaneously by all the cameras for a one-step calibration. Otherwise, the calibration should be processed in multiple steps where each calibration result has to be merged and transformed in the absolute coordinate frame. In both cases, the area covered by each camera should intersect with each other's which makes the approach not well suited for surveillance systems with uncovered areas.

In our approach, the designed system allows a fast camera registration by correlating the 3D scene model with the camera image planes. The method relies on manually matching corresponding points by picking the virtual model and the image, as shown in Figure 3(a). The pose camera can then be estimated from a few correspondences and allows the different virtual reality representations, as presented in Figure 3(b). The Mixed Reality (MR) (geo-localized textures billboards) is generally used for 3D virtual navigation since it does not constraint the pose of camera for user, however the Augmented Reality one allows to check the accuracy for 3D people localization with calibrate merge between video-streams and synthetic environment.

The last challenge concerns the video streams visualization from the referenced cameras. The registration step provides information about the cameras such as pose and perspective parameters. Augmenting dynamically the 3D virtual environment with videos provided from the cameras is then possible making use of the latter parameters.

2.2. Video visualization

Current approaches consider the video stream content as texture patterns and map it onto the 3D model. In [19], Sawhney et al. propose to project video stream contents, from multiple cameras, onto a 3D model of the complex outdoor scene considered. They have had a very interesting idea of considering video stream content as texture patterns to be mapped onto the 3D model: They combine static texture mapping for the areas not covered by video cameras and dynamic video projection for the areas covered by cameras. They named this process as *"Flashlight video system"* and provided hence a real-time 3D visualization system of the scene while also detecting and tracking moving objects in the model space. Similar approaches have been developed in [16]. Haan et al. [5] also consider projecting video streams as texture maps as in Sawnhey et al. approach but they avoid a complete reconstruction in the model space. Their approach aims at guiding the user into correct view space reconstructions still considering video streams as texture maps but projecting them onto 3D canvas modeling the scene. Hence, they can provide an intuitive visualization and navigation surveillance system based on the egocentric point of view of the operator. This method can be seen as a "view dependent rendering". Figure 4(a) shows a projection example. One can notice that the projection viewed from a near camera position is well aligned and accurate. However, projecting the acquired image from the camera viewpoint in the 3D model as a texture presents many limitations. Probably, the most significant one is that it depends highly of the user point of view. Our own system is in line with all these works. However, as mentioned by Girgensohn2007 et al. [7] and well illustrated by Wang et al. [22], the two afore-

mentioned methods suffer from projection issues. Indeed, projecting the video stream of a 3D object onto a 3D environment model can lead to serious semantic issues when the projected model area contains corners or broken walls, e.g. open doors. The projected video may be even harder to perceive and interpret because the video image is broken into multiple parts. Other problems can occur, such as scaling, distortion [22]. As illustrated in figure 4(b), the rendering of the video appears warped when the user view point moves away from the camera axis.

(a) Viewed from a near camera position, the projection is well aligned.

(b) The projection is warped when viewed away from the camera axis.

Figure 4. View-dependant projective texture mapping.

To overcome these issues, we propose to embed the video streams in a canvas object that we keep fixed in the optical axis of the camera. We propose then to extract and translate the dynamic information from the real environment to the virtual one. We get hence rid of all the problems arising from the projective mapping.
The proposed approach restores the positions and the movements of the persons in the generated 3D representation and endeavors to remain accurate to the reality. This needs to detect persons in the flow of images and infer their 3D locations accurately.

3. Video Processing: Real-time people extraction and 3D localization

Detecting persons in video frames is a challenging issue due to the large variability in their appearance, clothing, pose, background and intrinsic camera parameters. Usual methods tries to develop automatic person detectors based on local features extracted from a single image. Other detectors try to include motion features to provide a stronger cues for person presence [2].
The second issue raised in our application is person localization. Although locating detected persons using a monocular approach is conceivable, using only one view quickly shows its limits. In fact, inferring the location using a monocular approach usually needs to add hypothesis such as the equation of the floor (usually considered not sloped) or hypothesis regarding the height of the persons (which obviously cannot be generalized). In this way, people localization suffers from being very sensitive to the quality of the detected bounding box which needs to be accurate (in the foot or/and head). Monocular approaches are thus not very reliable due to shadowing and occlusion. This led us think of multiplying the camera point of view for a more accurate estimation.
Applying stereoscopy provides the opportunity to overcome the aforementioned technical problems, as no hypothesis regarding the floor or the height of the person are needed. It has also the advantage to overcome the technical issues related to network synchronization [15]. Using synchronous image pairs and using the simplification led by the epipolar geometry, stereo vision algorithms provide an accurate estimation of the 3D location of the objects.
A three staged approach is implemented in our framework: (1) **extraction**, (2) **classification** and (3) **localization**.
The extraction step aims at extracting the moving objects in the scene representing potential persons. These candidates regions are then classified in people/not-people regions using a real-time boosting-based classifier, in the second step. Finally, based on the 2D detection of people in a frame, the depth map is estimated still in real-time using our parallel implementation on GPU of a belief propagation algorithm.

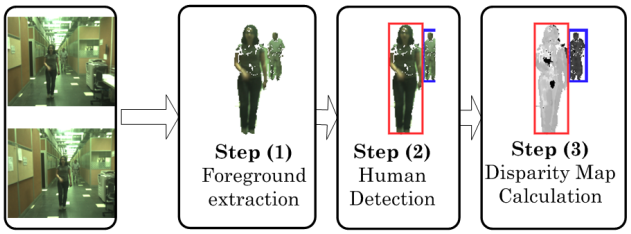


Figure 5. People detection and localization framework in Virtu4D

3.1. Foreground Extraction

Detecting humans within the images requires scanning the whole image, with different scales, in order to locate potential person candidate regions. This process can be computationally very expensive. The foreground extraction process targets to reduce the search space for potential human detection, assuming they are moving. For that purpose, moving objects are separated from the background. Most of the approaches from the earliest as in Polana et al. [18] for instance to more recent ones [7] extract video blocks corresponding to the objects of interest in the data.

In this paper, we consider a statistical color image segmentation modeled by an adaptative mixture of Gaussians. This approach has been proposed first by Grimson et al [9]. This online per-pixel and low-level method makes the assumption of a color difference between moving objects and the background, similarly to a human eye responding to different wavelengths of light. The final algorithm developed

in Virtu4D comes up with some improvements regarding this reference method:

- a variable number of Gaussians to model the background. Each pixel is modeled by a mixture of a variable number of Gaussians depending on the complexity of the background thus optimizing computation time and avoiding concurrence between Gaussians.
- Shadows and sudden luminosity changes may lead to a false foreground map. A shadowed pixel is characterized by a small color distortion and a lower brightness compared to the background model. A highlighted pixel, however, is characterized by a low color distortion and a higher brightness compared to the background model of the pixel [10]. We introduce a conic shape model to tackle this issue as in [11].

Figure 6 presents the foreground extraction results through different frames. One can notice that the search region is highly focused which significantly reduces the number of person region candidates and thus minimizes tests we would further potentially need to process. Thanks to

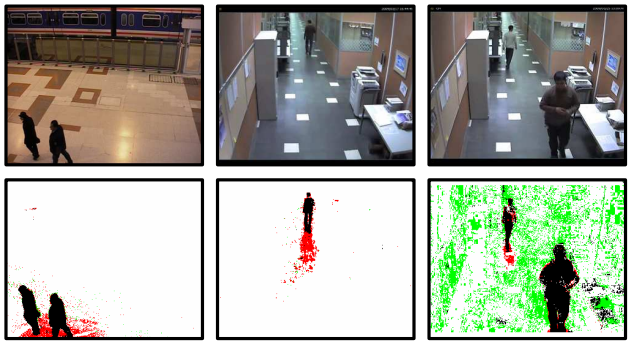


Figure 6. Some foreground extraction results. Foreground elements are in black, shadows are in red and highlights are in green. Left: foreground extraction of the 241^{th} frame of PETS'2006 Dataset [14]. Center and right: foreground extraction results in a MPEG-4 medium quality video from our camera network. Right: sudden highlight detection.

the foreground extraction, a set of candidate pedestrians is then spotted in different images to further classification as possible person detection.

3.2. Person boosting-based classifier

The deformable shape of the human body, combined with illumination and pose variations contribute to the complexity of person detection task.
The leading approaches for this task use boosting classification techniques which consists in a linear combination of weak classifiers to build a "strong" classifier. The most well known example of applying these algorithms is the face detection method proposed by Viola and Jones [] where the weak classifiers are defined by a tuple of (Haar-filter response, the size of the filter, the threshold over the response for which the face is detected). Other weak classifiers than Haar-like features [17] have been proposed as Histograms of Gradient (HoG) or even SIFT descriptors [4, 3]. More recently, these boosting techniques have shown their potential to address person detection in images through the works of Laptev [13], based on HoG features and weighted Fisher linear discriminant similarity measure between HoG, and Felzenszwalb [6] which enriches Laptev approach by assembling local HoG detectors of body parts, according to geometric constraints to form the final human model. For its good low complexity properties, we focus here on Laptev approach.
In all these approaches, each weak classifier (Haar-filters, HoG...) is evaluated for all possible parameter settings and scanning the whole image. In our work, based on the results of previous step (foreground extraction), we can restrict scanning and evaluation of the weak classifiers on candidate regions which speed up a lot this classification process. As in Laptev method, we use a structure of cooperating Weighted Fisher Linear Discriminant (WFLD), for which parameter settings are generated using the well known Adaboost method on a database of 700 true and 7000 false samples taken from our camera network. The performance of the overall strong classifier is improved by iteratively applying to the next weak classifier a reweighted version of the training data in order to emphasize data which were wrongly classified by the previous weak classifier.
The features are histograms of oriented gradients (HoG) computed separately in subdivided parts of the rectangular sub-image of candidate region (24x9 pixels).
The detection is finally achieved on the region of interest through different scales. As a single pedestrian is likely to be detected multiple times, we generate a mean bounding box that surround the final detected pedestrians.

3.3. Depth Map Estimation

Once a person is located in an image plan, depth map estimation allows recovering his/her 3D location in the scene. Depth map estimation has been extensively studied in the stereo vision field [20]. Stereo matching algorithms are generally classified into two classes, local algorithms and global algorithms based on the cost computation method. Local approaches are based on a correlation criterion over a local window containing the element to match. While being fast, these methods generally do not perform well on occluded or low textured regions.
To overcome this, global approaches minimize an overall cost function that involves all the pixels of the image. Though, such algorithms are known to be computationally expensive and hence may cause a performance bottleneck for a real-time system. To overcome this major problem, we

take profit of a parallel implementation of the beliefs propagation algorithm on a programmable graphic hardware [1]. The beliefs propagation is a recursive inference algorithm that minimizes an energy function to estimate the disparity field. In our case, the energy function is formulated as a sum of two terms (Eq 1). The first term is a data driven energy. The second one enforces smoothing depth estimation. Let P be the set of pixels in an image and L be the disparity values set. A disparity field d assigns a disparity value $d_p \in L$ to each pixel $p \in P$. To measure the quality of a disparity field d, we consider the global energy function

$$E(d) = \sum_{p \in P} D_p(d_p) + \sum_{(p,q) \in N} U_{p,q}(d_p, d_q) \quad (1)$$

D_p and $U_{p,q}$ are the *data cost* and *smooth cost* respectively. The data cost encodes the log-likelihood function. The smooth cost encodes the prior distribution. N is the set of neighboring pixels couples.
To compute the data cost, we use a truncated absolute difference as the matching cost. We aggregate this cost over a square window with constant disparity (Eq 2).

$$D_p(d_p) = \sum_{(x,y) \in N(p)} min(|I_L(x,y) - I_R(x - d_p, y)|, T) \quad (2)$$

$N(p)$ is a p-centered square window. I_L and I_R are respectively the left and the right images. T is a threshold. The smooth cost is also computed using a truncated absolute difference with threshold λ (Eq 3).

$$U_{p,q}(d_p, d_q) = min(|d_p - d_q|, \lambda) \quad (3)$$

The minimization of this energy function is achieved recursively by passing "messages" between neighboring pixels. A pixel p sends to each of its four neighbors $q_i, i \in \{1,2,3,4\}$ a message $m^k_{p \to q_i}$ at every iteration k (Eq 4). Each message is a vector, with each component being proportional to how likely the pixel p "believes" that the pixel q_i will have the corresponding disparity. After convergence, we compute the "beliefs" vector for each pixel p (Eq 5), and we select the disparity that corresponds to the component of the beliefs vector with the minimum value (Eq 6).

$$m^k_{p \to q_i}(d_{q_i}) = \min_{d_p}(D_p(d_p) + U_{p,q_i}(d_p, d_{q_i}) + \sum_{j \in \{1,2,3,4\}, j \neq i} m^{k-1}_{q_j \to p}(d_p)) \quad (4)$$

$$b_p(d_p) = D_p(d_p) + \sum_{i \in \{1,2,3,4\}} m^K_{q_i \to p}(d_p) \quad (5)$$

$$d^*_p = arg \min_{d_p \in L} b_p(d_p) \quad (6)$$

Given the person 2D location within the image plane and $\tilde{d}$ the mean value of the disparities within the person blob,

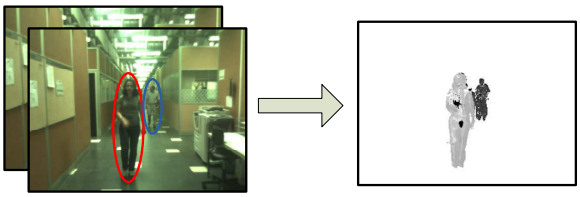


Figure 7. Depth map estimation result.

the 3D location with respect to the stereo sensor referential is recovered by triangulation following the equation 7. The parameters f and b corresponds respectively to the focal length of the sensor and the stereo baseline.

$$Z = \frac{f * b}{\tilde{d}} \quad (7)$$

The implementation of this algorithm on a programmable graphic hardware using NVIDIA CUDA technology [1] unlocks the processing power of the GPU to offer accurate and real-time estimation of the depth map. The table 1 reproduces the classification by the benchmark Middlebury of our results compared to some algorithms listed in the benchmark. It is noteworthy that our algorithm is the 23^{rd} place in the overall standings, and 2^{nd} in the ranking algorithms real time.

Table 1. Classification by the benchmark Middlebury

Algorithm	Rank	% false matching			
		Tsukuba	Venus	Teddy	Cones
AdaptingBP [12]	2.8	1.11	0.10	4.22	2.48
RealtimeBP [23]	21.9	1.49	0.77	8.72	4.61
Our implementation	**23.2**	**1.59**	**1.13**	**12.6**	**6.27**
RealTimeGPU [21]	26.8	2.05	1.92	7.23	6.41
BP+MLH [20]	32.5	4.17	1.96	10.2	4.93

Figure 7 shows the result of the depth map estimation coupled with foreground extraction on a frame containing two persons. The grey level in the disparity map is inverse proportional to the depth of the person. The 3D location of persons within a global referential can then be computed using the registration parameters of the camera network.

4. Immersive Model Generation

Video surveillance systems often provide multiple interfaces/views to help monitoring and understanding different situations. DOTS system [7] provides mainly two user interfaces. The first, a 2D interface "multi-stream video player", displays multiple video streams (camera bank) along with a timeline and a floor plan. The second is a 3D viewer that displays billboards representing detected people within a 3D model of the surveillance area.

Virtu4D provides the user with three main complementary ”views” in only one user interface. First, the 3D environment model and information from the sensors network are fused in one 3D dynamic virtual representation. Detected and localized persons are represented by dynamic avatars to enrich the environment model. Second, video streams are added to the 3D virtual world. Their locations follow the physical locations of the cameras, allowing an increasing understanding of observed situation and the spatial relationships between streams. A third view allows the user to watch 2D video from one camera in augmented reality mode.

4.1. Virtual Reality Mode

Virtu4D system takes benefit from both the ”real” and the ”virtual” worlds offering a unique perception of the scene. Detected persons are represented by 3D avatars (cylinders in Figure 8, humanoid models in 9) in the virtual world. The 3D locations of the avatars reflect the actual humans' positions in the real scene. This representation mode gives a simple and computable duplicate of the real world making possible the further exploitation of data in ”operational” algorithms like people tracking, detection of abnormal behaviors in crowd, etc. Furthermore, the generated scene is freely navigable without any constraints with respect to the position of the observer.

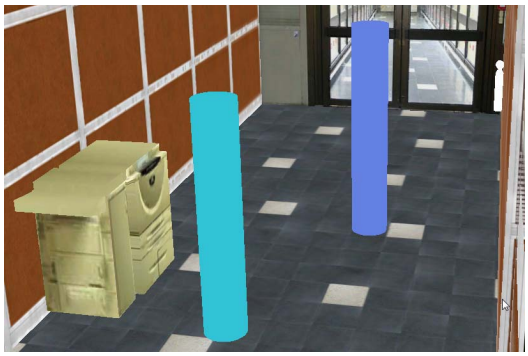


Figure 8. Virtual reality representation.

4.2. Mixed Reality Mode

The proposed system does not break totally with the classical surveillance systems. Indeed, live video streams are also reported in the 3D visualization as video walls in front of the registered cameras (Figure 10). This allows recovering the amount of information that image processing algorithms fail to extract from the video streams, due to either the medium/poor quality and low resolution of cameras or the high complexity of the observed scene. The operator can then switch smoothly between the virtual representation and the real one by navigating within the 3D environment.

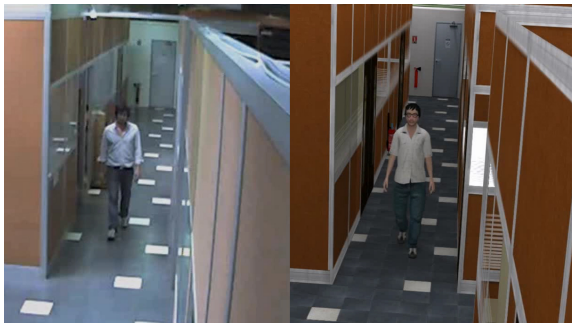


Figure 9. Humanoid avatar

(a) (b)

Figure 10. Two successive frames in the mixed reality representation. The 3D trajectories help the observer understand the scene.

4.3. Augmented Reality Mode

In Virtu4D system, switching between the 3D and the 2D representation is straightforward. Within the same user interface, when moving the observation point to meet the physical location of a camera, the observer can watch the corresponding video stream. This view is obtained by placing the 3D viewer in the place of a camera. Using alpha blending between the video wall and the dynamic 3D model, we get merged visualization between video and 3D model.

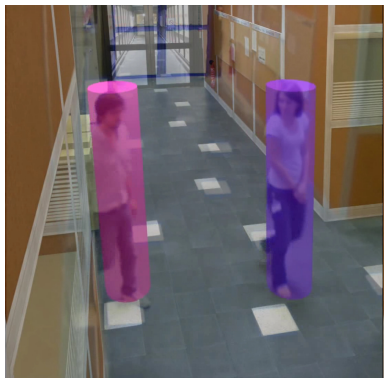


Figure 11. Augmented reality representation

5. Conclusion

We presented in this paper a new surveillance system that provides a coherent live visualization of a dynamic complex scene captured by passive video cameras. Using a stereo vision algorithm coupled with foreground extraction and human detection, the disconnected moving elements captured by each camera are gathered into a single animation accurately to the reality, hence allowing an unique perception of the scene. This virtual copy allows a free intuitive navigation in the environment and hence making the supervision systems easier.

Future work will focus on pushing further the virtualization of the real world in the perspective of reducing the need of video streams to better understand the situations. Such achievement can be done by adding a re-identification module that will attribute the same ID to the same person whenever he/she reappears in a supervised zone. Such module will allow people tracking even with a surveillance system containing non-supervised zones.

References

[1] M. Boufarguine, M. Baklouti, V. Guitteny, and S. Couvet. Real-time dense disparity estimation using cuda's api. In *International Conference on Computer Vision Theory and Applications*, February 2009. 6

[2] Q. Cai, A. Mitiche, and J. K. Aggarwal. Tracking human motion in an indoor environment. In *In 2nd Intl. Conf. on Image Processing*, pages 215–218, Washinton, D.C., October 1995. 4

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *International Conference on Computer Vision & Pattern Recognition*, pages 886–893, June 2005. 5

[4] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *European Conference on Computer Vision*, pages 7–13, May 2006. 5

[5] G. de Haan, J. Scheuer, R. de Vries, and F. Post. Egocentric navigation for video surveillance in 3d virtual environments. In *IEEE workshop on 3D User Interfaces*, March 2009. 2, 3

[6] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2008. 5

[7] A. Girgensohn, D. Kimber, J. Vaughan, T. Yang, F. Shipman, T. Turner, E. Rieffel, L. Wilcox, F. Chen, , and T. Dunnigan. Dots: support for effective video surveillance. In *the 15th international Conference on Multimedia (MULTIMEDIA '07)*, Augsburg, Germany, September 2007. 2, 3, 4, 6

[8] M. W. Green. Appropriate and effective use of security technologies in u.s. schools. Technical Report 97-IJ-R-072, National Institute of Justice, September 1999. 1

[9] W. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 22–29, June 1998. 4

[10] T. Horprasert, D. Harwood, and L. S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. In *International Conference on Computer Vision*, pages 1–19, September 1999. 5

[11] J.-S. Hu and T.-M. Su. Robust background subtraction with shadow and highlight removal for indoor surveillance. *EURASIP Journal on Applied Signal Processing*, 2007:108, January 2007. 5

[12] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR06*, pages 15–18, 2006. 6

[13] I. Laptev. Improvements of object detection using boosted histograms. In *British Machine Vision Conference*, volume 3, pages 949–958, 2006. 5

[14] V. Manohar, M. Boonstra, V. Korzhova, P. Soundararajan, D. Goldgof, R. Kasturi, P. Soundararajan, D. Goldgof, R. Kasturi, S. Prasad, H. Raju, R. Bowers, and J. Garofolo. Pets vs. vace evaluation programs: A comparative study. In *9th IEEE International Workshop on PETS*, pages 1–6, June 2006. 5

[15] A. Nakazawa, H. Kato, and S. Inokuchi. Human tracking on distributed vision agents. In *ICPR*, 1998. 4

[16] U. Neumann, Y. Suya, H. Jinhui, J. Bolan, and L. JongWeon. Augmented virtual environments (ave): dynamic fusion of imagery and 3d models. In *Proceedings. IEEE Virtual Reality*, pages 61–67, March 2003. 3

[17] C. Papageorgiou and T. Poggio. Trainable pedestrian detection. In *International Conference on Image Processing*, pages 35–39. IEEE, October 1999. 5

[18] R. Polana and R. Nelson. Low level recognition of human motion (or how to get your man without finding his body parts). In *In Proc. of IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects*, pages 77–82, 1994. 4

[19] H. S. Sawhney, A. Arpa, R. Kumar, S. Samarasekera, M. Aggarwal, S. Hsu, D. Nister, and K. Hanna. Video flashlights: real time rendering of multiple videos for immersive model visualization. In E. Association, editor, *the 13th Eurographics Workshop on Rendering*, volume 28, Pisa, Italy, June 2002. S. Gibson and P. Debevec, Eds. ACM International Conference Proceeding Series. 2, 3

[20] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision*, 47(1-3):7–42, 2002. 5, 6

[21] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nister. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. In *3DPVT06*, pages 798–805, 2006. 6

[22] Y. Wang, D. Krum, E. Coelho, and D. Bowman. Contextualized videos: Combining videos with environment models to support situational understanding. In *IEEE Transactions on Visualization and Computer Graphics*, Oct. 2007. 2, 3, 4

[23] Q. Yang, L. Wang, and R. Yang. Real-time global stereo matching using hierarchical belief propagation. In *BMVC06*, page III:989, 2006. 6