Multimed Tools Appl (2014) 73:977–993
DOI 10.1007/s11042-012-1184-z

Smart video sensors for 3D scene reconstruction of large infrastructures

Oscar Ripolles · José E. Simó · Gines Benet · Roberto Vivó

Published online: 25 July 2012

Abstract This paper introduces a new 3D-based surveillance solution for large infrastructures. Our proposal is based on an accurate 3D reconstruction using the rich information obtained from a network of intelligent video-processing nodes. In this manner, if the scenario to cover is modeled in 3D with high precision, it will be possible to locate the detected objects in the virtual representation. Moreover, as an improvement over previous 2D solutions, having the possibility of modifying the view point enables the application to choose the perspective that better suits the current state of the scenario. In this sense, the contextualization of the events detected in a 3D environment can offer a much better understanding of what is happening in the real world and where it is exactly happening. Details of the video processing nodes are given, as well as of the 3D reconstruction tasks performed afterwards. The possibilities of such a system are described and the performance obtained is analyzed.

Keywords Surveillance · Smart video sensors · Tracking · 3D reconstruction

O. Ripolles (✉) · J. E. Simó · G. Benet · R. Vivó
Inst. Universitario de Automatica y Informatica Industrial,
Universidad Politecnica de Valencia, Camino de Vera s/n, Valencia, Spain
e-mail: oripolles@ai2.upv.es

J. E. Simó
e-mail: jsimo@disca.upv.es

G. Benet
e-mail: gbenet@disca.upv.es

R. Vivó
e-mail: rvivo@dsic.upv.es

1 Introduction

Surveillance solutions commonly resort to simple client applications to present the obtained images. Advanced solutions are capable of offering images from multiple cameras at the same time, so that one single application can offer images from a large number of cameras. The main problem appears as the average attention span of a person under these circumstances is very limited. In this sense, psychological research has demonstrated that attention deteriorates markedly after 15–20 min [9].

The use of intelligent vision systems where image analysis is performed can reduce human effort. Thus, visual or sonour stimuli can be used to focus the staff attention to a specific camera which, for any reason, requires supervision. Previous solutions based on image analysis were also capable of detecting different types of event and alert the surveillance staff accordingly. Nevertheless, a 3D context can be used to construct a world view with the captured data and extend it with distances, perspective correction, or other features. This richer context model is subsequently applied to derive a more accurate analysis about the events that happen. Thus, having a 3D reconstruction can improve the context model and apply more reliable filtering, segmentation or occlusion handling. Having 3D feedback can improve the context modeling as it would be possible to find interpretation errors and impossible situations that, without 3D information, would not be possible to detect.

In this paper we present a multi-camera control and surveillance system which can reconstruct the events that happen in a large infrastructure. The data is presented through a 3D visualization platform where all the information collected from the different cameras can be displayed at the same time, offering new means to interact with the data. The 3D environment has been modeled with high accuracy to assure a correct simulation of the scenario. The 3D reconstruction is feasible thanks to specifically-designed cameras which offer the necessary metadata in an efficient manner. The embedded software of these cameras includes acquisition, segmentation, labeling, tracking and low-level classification, as well as extraction of additional features for each detected object. Figure 1 presents an overview of the system, composed of a set of smart nodes distributed over a larger infrastructure.

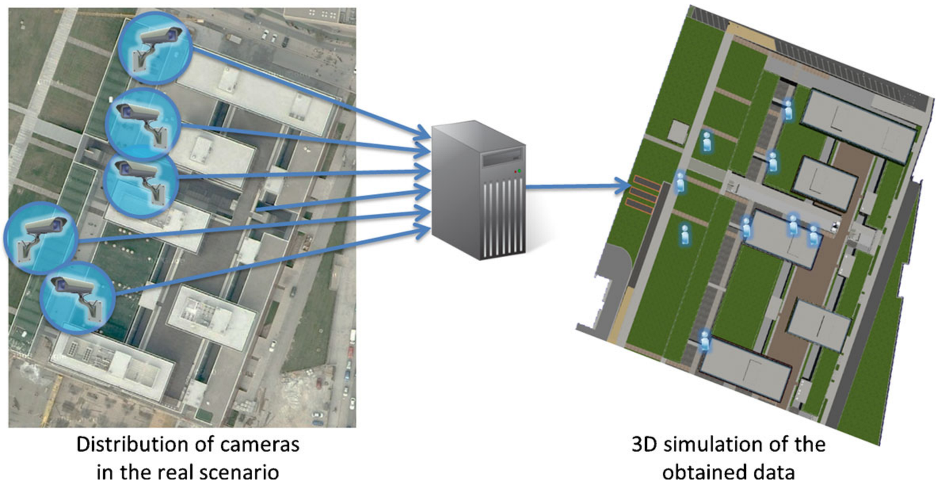


Fig. 1 Overview of the proposed solution

These nodes provide metadata to the main computer which is in charge of the 3D reconstruction. As the load of performing the visual analysis is done on each camera, we can see this system as a distributed system where the 3D reconstruction is performed on the end-user machine. In addition, managing the 3D context simplifies camera deployment as the registration within the system is fast and simple. Once the camera is correctly located on the scenario, the correlation between the cameras is automatically calculated.

Our aim is to exploit distributed embedded processing to improve the quality of the information displayed and of the Human-Machine Interface (HMI) itself. Collecting all the information coming from a network of smart cameras in one single 3D view can open new possibilities for surveillance. Each camera in our system is not regarded as an independent observer, but the assembly of cameras in a 3D simulation acts as an omni-directional vision sensor. This allows, for example, for automatic point of view selection (e.g. activated by activity or movement). This mixed reality simulation is useful to represent many images in a single one with indications of what is happening and where it is happening. Thus, the system could track a suspect automatically among different cameras or offer a bird's-eye view of the scenario to show all the events that are happening in a single view. Nevertheless, this 3D representation is not intended to be a substitute of the real video sources but to serve as a tool to simplify surveillance tasks. In this sense, automatic video production is one of the applications of our proposed solution. Thus, as an example, having a 3D simulation can allow the system to choose the better series of real cameras to follow a person. Other possible benefits could be: filtering erroneous data, filtering erroneous orientation information, predicting the trajectory of a detected object, showing non-visible objects, controlling a perimeter or performing multi-camera tracking without the limitations of vision-based approaches. In Section 3 we will address how these potential improvements can be obtained with our proposal.

This paper is organized as follows. In Section 2 the embedded architecture is presented and some of its main features are discussed. In Section 3 we describe the 3D reconstruction process for luggage, people and groups. In Section 4 the presented framework is analyzed and the obtained results are discussed. Finally, Section 5 concludes our work and outlines future work.

2 Smart nodes description

In this section we will detail the hardware and software included in the smart cameras that our proposed framework uses. These smart nodes have been developed as part of the SENSE research project [13]. A SENSE node can be understood as an embedded video subsystem, which acquires and processes video images extracting features in a modal way.

The video board has been developed using two Blackfin DSP processors (a BF561 and a BF537processor) for videoprocessing tasks and a Spartan III FPGA, which provides a way of configuring hardware outline. Blackfin core modules provides a PPI (Parallel Peripheral Interface) port to connect digital cameras following YUV, YCbCr and RGB standards. The camera programming can be done through I^{2}C interface using the Serial Camera Control Bus (SCCB) standard. The current choice for the camera is to use an Omnivision OV7660 VGA because image processing for higher resolution than VGA is too heavy for these Blackfin DSP processors. One constraint for this system is that it works with a YUV 422 interface, which affects segmentation process.

The architecture has been designed so that the main image analysis tasks are performed in the BF561 processor. The BF537 processor has two main tasks involving the video node: compressing the incoming frames into JPEG format and encoding the information given by the BF561 processor to obtain XML messages. The preliminary designs for this system included FIFO buffers to store intermediate data. This approach had the advantage of being very easy to implement. However, there were two main disadvantages on this system:

1. High latency. When a phase runs faster than the following one, the interconnection buffer gets full and the data has to wait for longer queues.
2. Data destruction. The last phases need data stored on the first phases. However, due to the large number of intermediate processes, when the last phases access the data stored on the first ones this data has been replaced.

To manage information sharing among phases, a structure has been created to store the information relative to a single frame that is collected through the different steps. In the current implementation of the SENSE nodes there are three parallel execution units that cover all the processing steps. In this way, at most, we must keep in memory information regarding three different frames.

The reader is referred to [14] for more information on the hardware and software characteristics of the SENSE nodes. In the rest of this Section we will describe the different image-based processes performed in the BF561 processor. Figure 2 displays the main steps, including some sample images that can clarify each of the tasks.

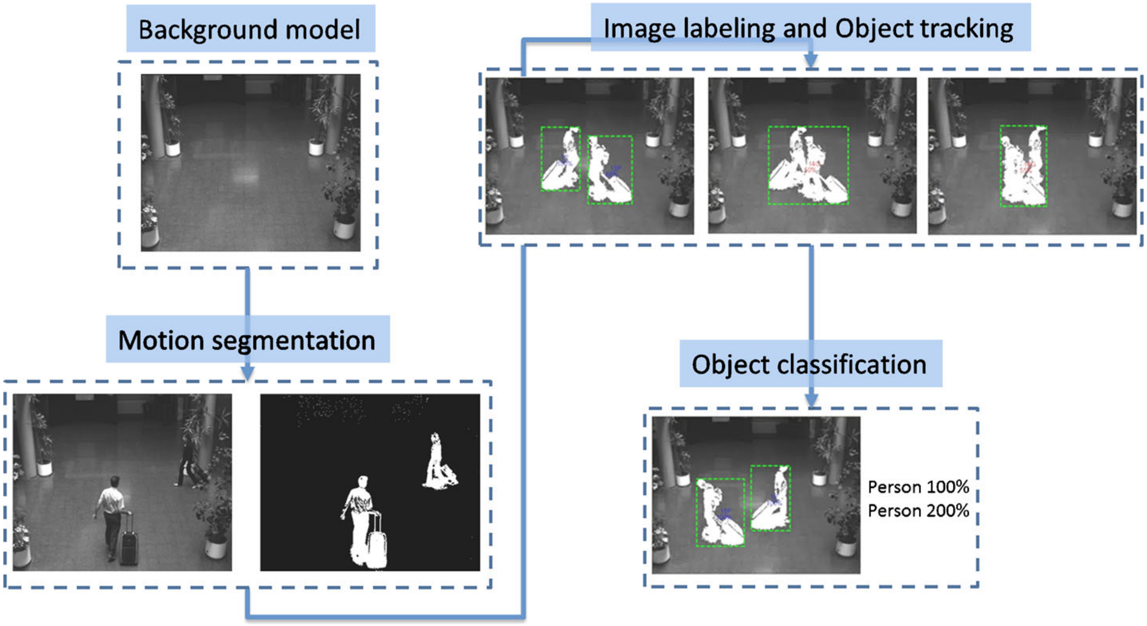


Fig. 2 Diagram and sample results of the main image analysis steps

2.1 Image acquisition

To acquire an image from the camera we just need a function to trigger the *DMA receive* operation. In order to simplify the system design, the FPGA attaches the time stamp[1] of each frame as a watermark on the first bytes of each image. This way, both processors receive exactly the same information from each frame, so there is no need to perform additional synchronization processes.

2.2 Motion segmentation and background update

From a general perspective, objects in a captured frame are detected by comparing the incoming images with an empty scene model known as *background model*. This model represents the scene without any foreground object. For each pixel on the image, the YUV color space is used to implement a very efficient color segmentation process.

The process of creating the background model consists in acquiring several images and performing a median operation for each pixel array to get the most probable value. As there are limited memory resources we have an upper bound for the number of images that we can store and, thus, a stationary foreground object could interfere the environment model. The solution that we have adopted has been twofold. On the one hand, we expand the acquisition of the images over time by including a stride parameter on the background creation algorithm. On the other hand, the image is divided into several blocks, so that only one block is updated at the same system cycle.

2.3 Image labeling

The image labeling phase recovers the binary image created by the segmentation process and produces the first high level data of the scene. Foreground objects are found, labeled, and some features are obtained. In addition, to improve the quality of the labeling process, Gaussian noise is filtered as well as objects that do not reach an established size threshold. The algorithm that we have used is a modified version of the one presented by Chang and Cheng [3]. This algorithm is based on the contour tracing technique and allows us to obtain the perimeter, bounding box, area and centroid of the objects while doing the labeling process.

2.4 Local object tracking

The tracking algorithm is based on the superposition method: bounding boxes of the objects from current scene are compared with the boxes from the previous scene. If the boxes are overlapped, a relationship between the objects is considered. This is a low level method and it is not intended for performing complex tracking of people.

[1] Despite using the term *time stamp*, it is not intended for timing purposes, but only for aligning features with its corresponding compressed image.

It has been implemented to solve segmentation errors and enrich the results of the object classification phase. Four situations can occur:

- an object from the previous scene matches an object from the current scene. In these cases we just spread all the information of the previous object to the current one.
- an object from the previous scene matches several objects from the current one. This situation can be produced by a real segregation of objects or by a bad segmentation. We have introduced an *inertia* period in order to have information from more frames to decide whether it is a real segmentation or an error.
- several objects from the previous scene match a single object from the current scene. This situation can occur either when several objects merge into a group or, again, when segmentation was not correctly solved. In either case, the system will never consider the bad segmentation case and it will create a grouped object.
- several objects from the previous scene match several objects from the current scene. This is the most complex situation. The system simply merges all the objects involved in the multiple match.

As a final consideration, we must assume that the segmentation is not going to perform perfectly on any situation. The system has to be able to reduce the impact of wrong segmentations. Consequently, we must keep in mind that merging objects is always preferable to keeping them split up.

2.5 Object classification

In the current software of the SENSE nodes, objects are classified as *Group*, *Person* or *Luggage*. A study based on nearly 3,000 objects was performed in order to obtain the statistical distributions of the features of the objects. From this study, it was concluded that the features that best distinguish the three classes are:

- the dispersion, known as *dispersedness* [8]. It relates the square of the perimeter with the area of the object, so that the bigger the dispersedness of a blob, the more dispersed (and less compact) its pixels are.
- the *extent* or *filling factor*, which is the relationship between the area of the object and the area of its bounding box. This value gives a measure of how filled is the area of the bounding box of the object. Luggage usually has bigger *extent* values than people because of its physical properties.
- the *number of heads*, considering each maximum of the upper silhouette to be a head. To improve results, a low-pass filter is used although there are some errors (like raised arms) which cannot be filtered out. For this reason, the vertical projection is used, which represents the number of pixels of a column. Thus, only those maximums which have a sufficiently large vertical projection (compared to the height of the silhouette) are considered to be heads.

The number of heads has proved to be very efficient to discriminate in a first step between *Group* of people from a single *Person* or *Luggage*. If the object has more than one head, then it is considered to be a *Group*. If we find only one head, then the statistical distributions are used to distinguish between *Person* and *Luggage*. More precisely, for one headed objects, the mean of the membership results given by the *extent* and *dispersedness* features is obtained.

3 3D scene reconstruction

It is possible to find in the literature some proposals for 3D reconstruction based on computer graphics techniques [11, 12], vanishing points [6, 15] or trained systems [17]. In our case, our approach uses computer graphics techniques which, following the same basic ideas as previous works, improve the obtained results by using the information from the intelligent nodes to obtain a better reconstruction if compared with previous solutions.

Our proposal uses the rich XML generated by the SENSE node to populate a 3D environment. The 3D HMI application receives images and features from the BF537 processor via its Ethernet device. In our current framework the images are discarded. We must underline that, for the correct performance of our proposal, the only requisites affect 3D modeling. We must have a very precise model of the scenario and we must also know the position and orientation of the cameras with a very accurate precision.

As indicated before, the 3D reconstruction using 2D information is done by means of computer graphics techniques using a 3D rendering engine. The idea is to estimate the position of the objects using collisions information. The process is very simple, although the results obtained will not be correct if cameras are not located and oriented properly. As an example, Fig. 3 presents a diagram of how this process works. While visualizing the 3D scenario, the synthetic camera is located at point A, while the image plane that matches the perspective view from the synthetic camera is located at point B. On this plane we display the real-world video sequence and the information obtained from the smart nodes. Thus, to locate the object in the 3D scenario we simply trace a ray that starts at A, goes through B and collides on the 3D scenario. We then can locate a 3D model representing the object at point C.

As we indicated before, some authors have previously used similar techniques, although they commonly use very rough 3D scenarios and create a simple textured

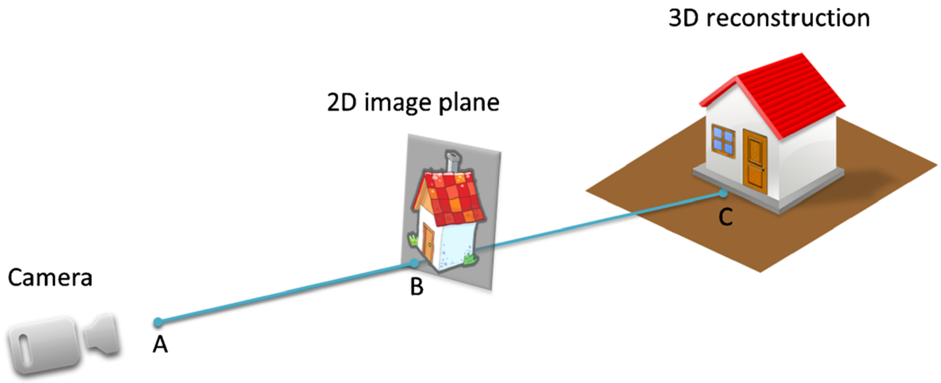


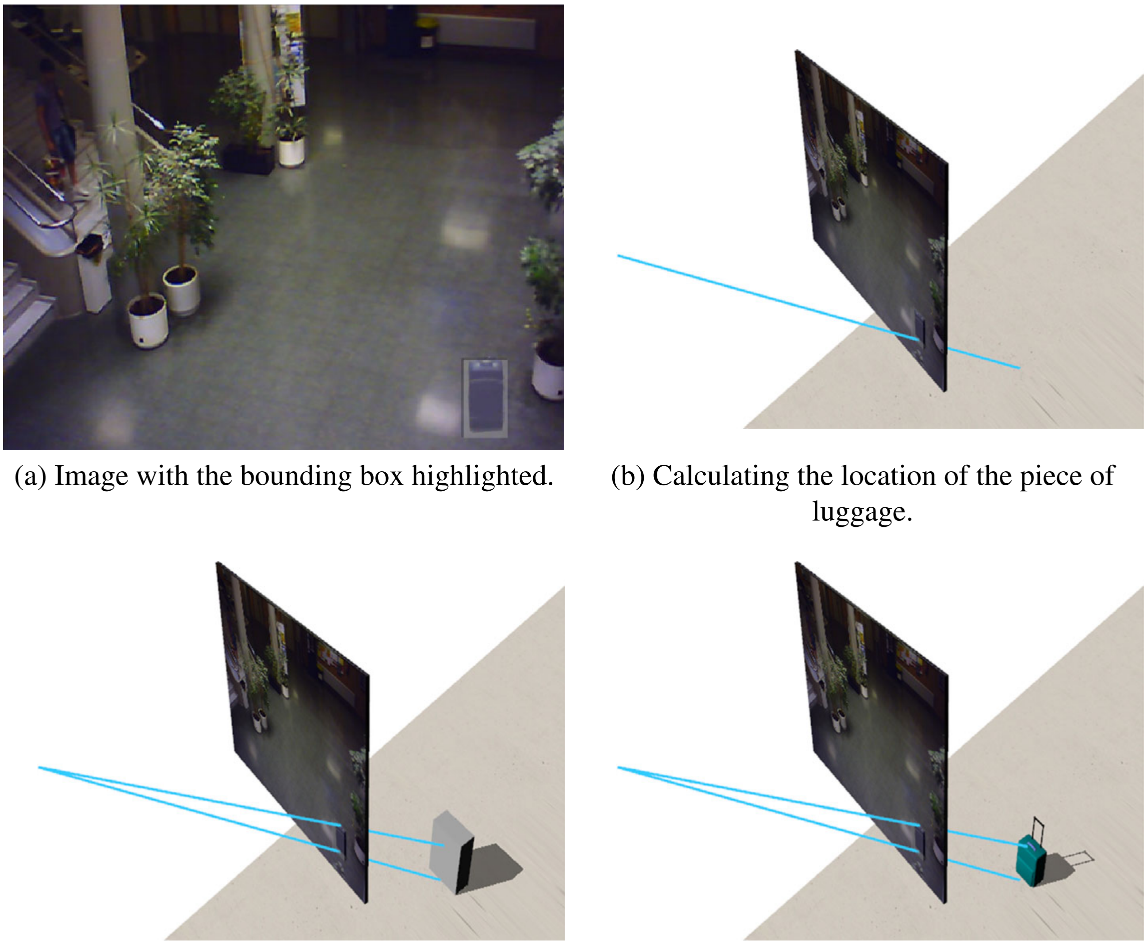
Fig. 3 Object location by casting a ray from *A* (optical center) through *B* (object location on the image plane) which intersects the floor in *C* (calculated position on the 3D scenario)

polygon at point C to represent the object with an image from the video source [5, 11, 12]. The main difference with previous approaches is the fact that we do not only use a point of the bounding box to locate the objects, but richer information to obtain a much better approximation of the objects. From the information that the smart camera outputs, our 3D reconstruction approach uses:

- the classification percentages
- the location and size of the bounding box
- the heads information
- the contour points of the object
- the velocity from the tracking algorithm, which indicates moving direction

Thus, although based on the idea presented in Fig. 3, a more complex process is performed to simulate the objects detected by the SENSE cameras: luggage, people and groups. The classification information of the SENSE node is used to choose which reconstruction process to apply.

Figure 4 presents an example of the described process performed when modeling a piece of luggage, illustrating the three main steps. Firstly, we start by casting a



(a) Image with the bounding box highlighted.

(b) Calculating the location of the piece of luggage.

(c) Calculating the height of the piece of luggage.

(d) Final appearance.

Fig. 4 Example of reconstruction of a piece of luggage

ray towards the scenario through the center of the bottom line of the bounding box. The collision gives us the location of the object within the scenario. Secondly, to assess the size of the suitcase, we create a box centered at the collision point and oriented toward the camera. The size of this box is slightly bigger than a regular suitcase to avoid errors in the following step. The third step consists in casting another ray through the center of the top line of the bounding box. This collision indicates the height of the piece of luggage. The width and length of the object are given proportionally.

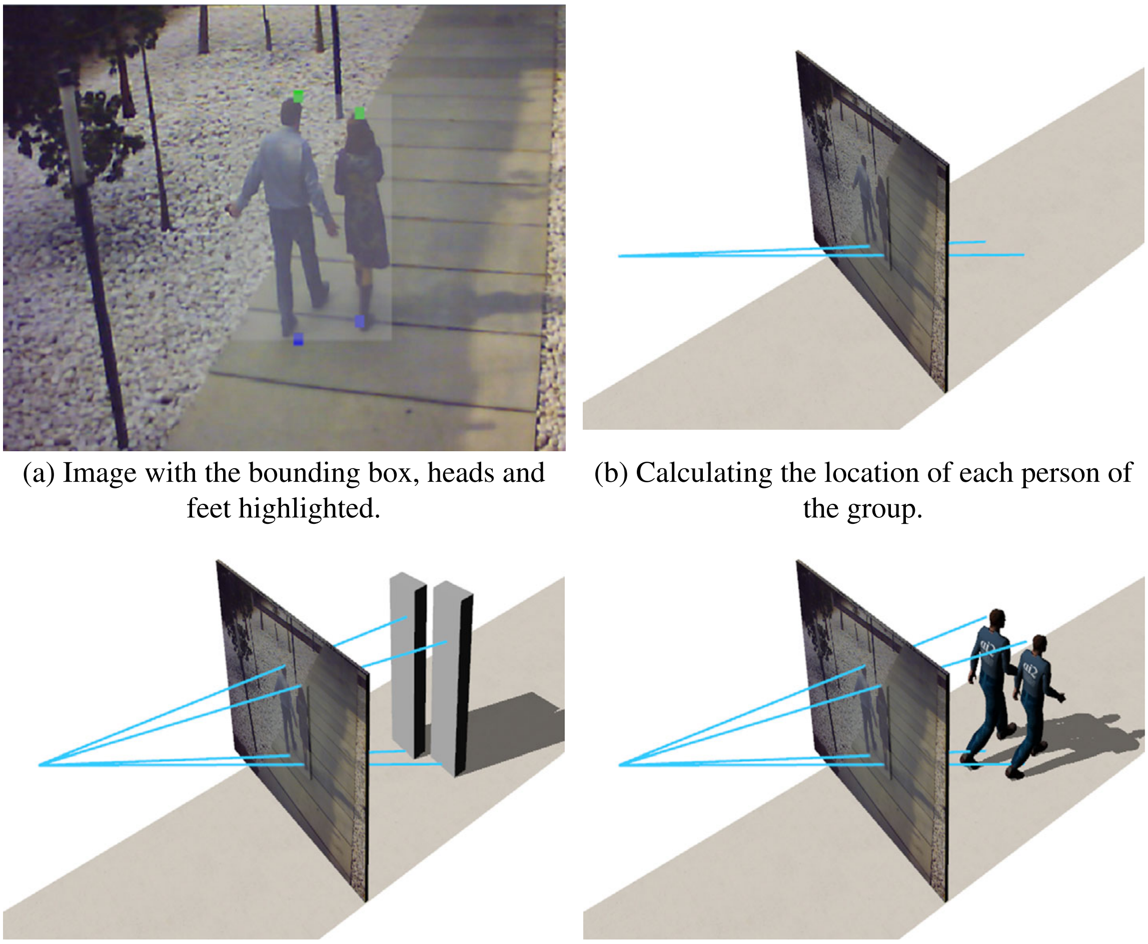
The reconstruction of a single person follows a process similar to that described above. Nevertheless, reconstructing a group of people entails a much more complex process. The main problem of a group is that each person might be at a different distance from the camera and, thus, using the information on the bounding box is not sufficient for approximating their actual position and size. The algorithm followed in this case is based on correctly locating the head and feet of each person on the input image. The head of each person is given in the XML offered by the SENSE camera. For obtaining the feet position, we simply use the head and the silhouette information included in the XML, so that we just find which point of the lower silhouette belongs to the column where the head is located.

Once we have the head and the feet position of each person, we firstly cast a ray through the feet to locate the people on the scenario. Then, as we did with the luggage, we create a box on each collision and trace new rays toward the heads. The collisions obtained with these new rays indicate the height of each person. Then, we can locate the 3D model on the scenario and give it the correct height. This process is depicted in Fig. 5. Finally, we must mention that the orientation of the people in our 3D simulation is obtained with the velocity parameter that the tracking algorithm of the SENSE node outputs within the XML information.

The main advantage of the method that we have described is the fact that it is camera-independent. Once the camera has been correctly located on our 3D scenario, the process to infer the object size or position is the same for every camera. It is true that some authors propose different techniques to infer these values using 2D information only, but they entail some manual calibration that can be very complex. In this sense, we underline that in our proposed framework we just need to know the real position of the camera to successfully locate it on the 3D scenario and start processing the XMLs.

The presented 3D framework is very promising and presents new possibilities for the interaction with data coming from surveillance cameras. As indicated in the Introduction Section, the detailed solution can be suitable for:

- filtering erroneous data. As we are capable of knowing the height and width of the objects in the real world, we can discard any object that exceeds some values. Moreover, in the results Section we will see how the classification results of the SENSE nodes can be improved.
- filtering erroneous orientation information. The SENSE nodes provide velocity information which is used to infer the orientation. As in our simulation we can know the actual position of the objects, we can adjust the orientation to what is actually happening in the scenario.
- predicting the trajectory of a detected object, so that the system can maintain the animation between the reception of the XML of two consecutive frames or even when there is a delay of several seconds.



(a) Image with the bounding box, heads and feet highlighted.

(b) Calculating the location of each person of the group.

(c) Calculating the height of each person of the group.

(d) Final appearance.

Fig. 5 Example of reconstruction of a group of people

- showing non-visible objects. Our system would be capable of displaying objects that are not visible from the cameras, using the aforementioned trajectory prediction. In this sense, we could display information even where there is no camera covering the area or when an object is occluded.
- following an object among the cameras, simplifying the task to the surveillance staff. This automatic production of the scene visualization can choose the better view-point according to the current situation.
- controlling a perimeter, knowing when and where an object entered a delimited area. We could also define where a person can access/leave the area in order to detect anomalous behaviour.
- performing multi-camera tracking without the limitations of vision-based approaches. Tracking across non-overlapping views is really difficult as the appearance of an object can be very different in each view. Moreover, object observations can be widely separated in time and space [7]. A 3D simulation can greatly reduce the problems posed by this kind of systems. We do not require overlapped views of the scenario; with just the exact position and orientation of the cameras we are capable of inferring the rest of information.

4 Experimental results obtained

The HMI we propose is based on the OpenSceneGraph (OSG) rendering engine [10], which includes some interesting features for visualizing 3D information and also for detecting and managing collisions. The human models are rendered using Cal3D, which is a skeletal based 3D character animation library written in C++ [2]. The implementation of Cal3D within OSG is available through the OSGCal extension. Finally, it is worth mentioning that all this software is open source.

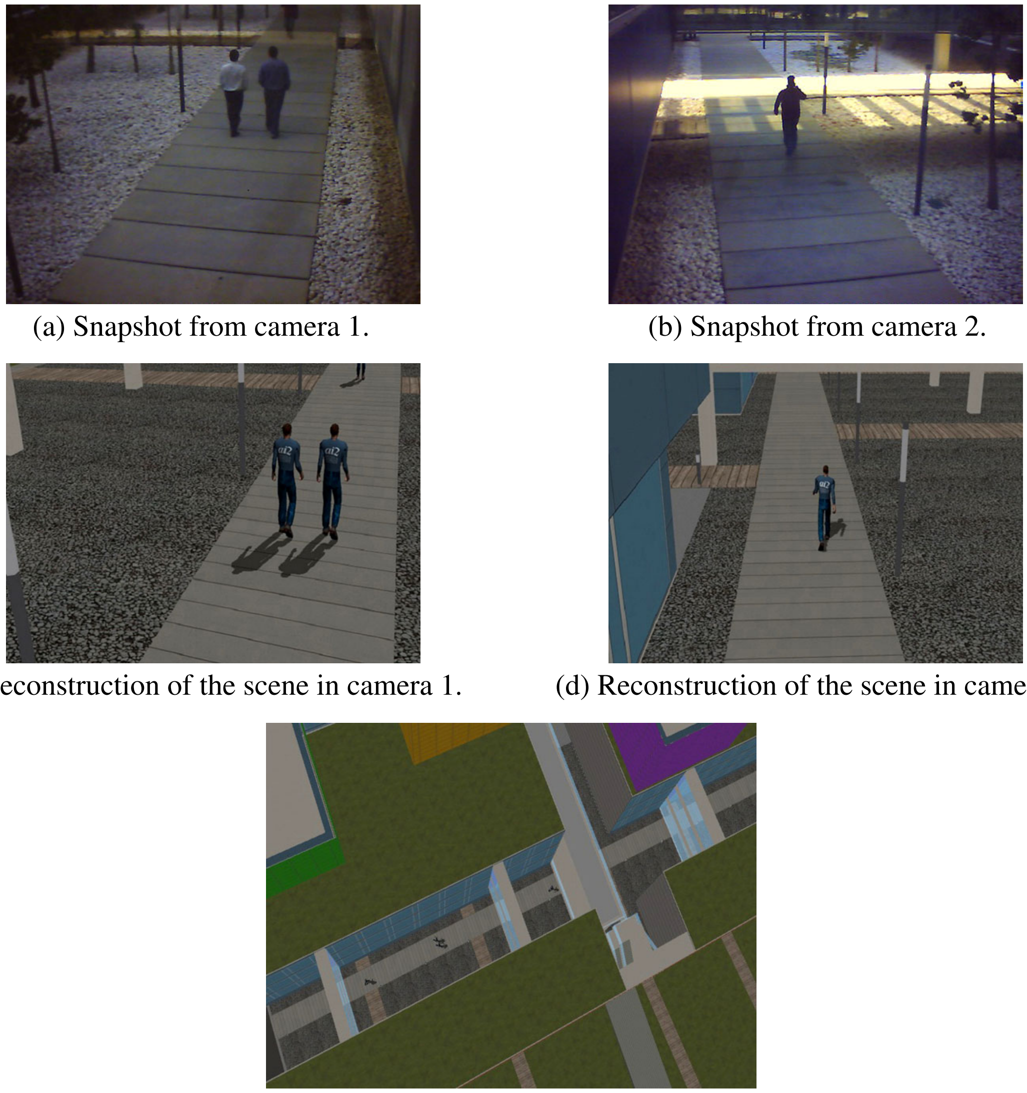
In this section we present some results that have been obtained using Windows 7 on a PC with a 2.8 GHz. processor, 4 GB RAM and an nVidia GeForce GTX 260 graphics card with 1 GB RAM. In these experiments, the BF561 processor was configured to run at a 600 Mhz. clock frequency and the camera was configured to provide 640 × 480 images at 4 frames per second. Throughout the rest of this section we will use recording data from the Polytechnical City of Innovation on the Campus of the Universidad Politécnica de Valencia in Spain. This large infrastructure has been chosen for testing purposes. A 3D model has been developed by professional modelers using all the CAD information available and photographs from the buildings.

4.1 3D reconstruction performance

We want to analyze the performance of our 3D reconstruction proposal from two different perspectives. On the one hand, Figs. 6 and 7 offer some visual results of our 3D reconstruction approach. In Fig. 6 we present two snapshots obtained from two cameras that we have deployed on our scenario. The real images shown in Fig. 6a and b are simulated in Fig. 6c and d. In addition, Fig. 6e presents a different point of view where the 4 persons detected in the cameras are depicted in a single view. As commented before, contextualizing the events detected can help surveillance staff. We also want to comment again that our framework is capable of reconstructing the scenario despite not having overlapped views. Regarding Fig. 7, the trajectories followed by two different people are depicted. In this figure we want to underline how it is possible to visualize in a simple way trajectories followed by persons at different heights within the building.

On the other hand, we have tested the performance of the 3D reconstruction and the visualization platform. The amount of cameras is simulated, ranging from 10 to 80 cameras and, to consider a case with high computational cost, we have decided that each camera was detecting and tracking 5 different objects (2 people, 2 pieces of luggage and 1 group of 2 people).

Table 1 presents the temporal costs of the 3D representation considering separately the time needed for XML parsing, scene update and visualization. From these results we can conclude that the scene reconstruction is a costly process, as we must perform several ray-casting operations per object. Nevertheless, in this test we have considered a very crowded scenario where all the areas covered by the cameras have a dense population. Thus, even for the worst scenario where 400 objects are detected, processed and depicted, the system is capable of working at 9 frames per second. This value is greater than the throughput offered by the SENSE cameras which was limited to 4 frames per second.



(a) Snapshot from camera 1.

(b) Snapshot from camera 2.

(c) Reconstruction of the scene in camera 1.

(d) Reconstruction of the scene in camera 2.

(e) Bird's-eye view of the whole scenario.

Fig. 6 Original images and synthetic reconstructions

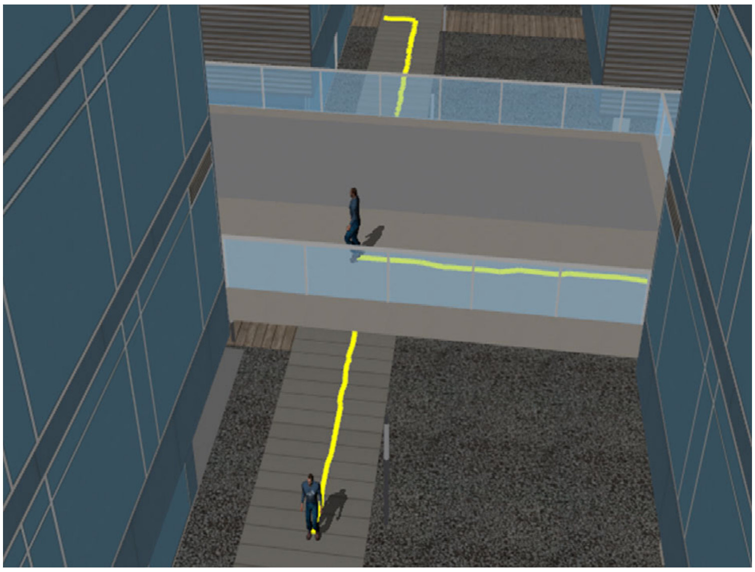


Fig. 7 Trajectories detected and visualized in our synthetic environment

Table 1 Performance obtained with our 3D simulation framework (in seconds)

# of cameras	10	20	30	40	50	60	70	80
# of objects detected	50	100	150	200	250	300	350	400
XML parsing	0.005	0.009	0.014	0.019	0.024	0.029	0.033	0.038
Scene reconstruction	0.008	0.017	0.024	0.032	0.042	0.049	0.058	0.066
Visualization	0.010	0.010	0.011	0.011	0.011	0.011	0.012	0.012
Total	0.023	0.037	0.0409	0.061	0.077	0.090	0.103	0.116

4.2 SENSE classification accuracy

In a previous article, the SENSE classification accuracy was tested [1]. Using a real sequence of 3,000 frames a total of 643 persons, 202 groups and 390 pieces of luggage were detected and classified. The classifications results obtained have been summarised in Table 2.

The main conclusion of these results is that the accuracy of the classificator was of 92 % for people, 81 % for groups of people and 99 % for pieces of luggage. We must note that the classification performed by the SENSE nodes is made at a low level due to its limited computational power. Nevertheless, using our 3D reconstruction and the wealth of information provided by the cameras we can improve the classification.

Our 3D reconstruction process can detect false classifications automatically by detecting objects with a size or position that cannot be correct. It is not possible for the system to correct the classification, but it can depict some information in order to warn the user about the issue. We have analyzed the frames used in the aforementioned test and we can conclude that:

- people are commonly mis-classified as a group. This error can be easily solved by using our approach where heads and feet are employed, as the final height of the person can be used to filter out mistaken information. In those cases were a person is classified as luggage, the size can be a hint to discard erroneous values, although we must consider that it is possible to have large piece of luggage.
- groups are never classified as luggage. Objects belonging to this class are easily confused with instances of class person, and it is easy to see that, depending on how people is arranged inside the group (for instance, people in groups may occlude one to each other and appear as a person), a group of people is quite similar to a single person. When a group of people is classified as a single person we can have two situations: (1) the SENSE node may not be able to classify the rest of people and it considers them noise or (2) the SENSE node may think that the whole group is a single person, thus resulting in a very large person which would also be filtered by our system.

Table 2 Classification results obtained with the implemented software during a real experiment

Object class	Classified as		
	Person (%)	Group (%)	Luggage (%)
Person	91.9	19.3	0.3
Group	6.8	80.7	0.5
Luggage	1.3	0.0	99.2

- luggage classification is very accurate, although sometimes it can be mistaken for a person or a group. In either case the 3D reconstruction is capable of detecting the false positives.

As a consequence, we can say that the 3D reconstruction is a very accurate tool for assessing the classification information. Nevertheless, some erroneous classification will remain undetected as there are cases in which our idea of what is "normal size" for people or luggage may be wrong. But it is not the case of the infrastructure that we consider, and these situations were not detected in the sequence used for the tests.

5 Conclusions

In this paper we have proposed a new solution for the automatic reconstruction of a scenario by gathering the information coming from many cameras into one single 3D view. The visual surveillance system that we propose includes a 3D feedback which offers a very detailed simulation of the scenario. Nevertheless, the success of the system completely depends on a correct detection and classification of the objects on the scenario and, moreover, on a fast performance of these algorithms in order to assure a real-time simulation of the events that are happening in the real world. Potential application areas range from home monitoring to security and surveillance in public or corporate buildings.

We have described the hardware and software of the SENSE vision system. In addition, the computer graphics techniques that enable the 3D reconstruction have also been presented. The performance of the whole system has been analyzed so that it can offer real-time simulations for 400 objects. Moreover, we have also studied how the 3D reconstruction can help the system filter erroneous classification data. In this sense, it would be interesting to develop some communication mean between the camera and the 3D application. If the 3D application could send information to the smart nodes, it could indicate them when a bad classification has happened or when an area of the image has a lot of erroneous information and needs a new background calculation.

It is worth mentioning that the classification routines can be trained to detect any object type once its main characteristics are extracted. At the moment the SENSE node was programmed to work in an airport-like scenario, although we believe that the current classification (person, group and luggage) is very general and may be applied to different environments. In this sense, people detection is always interesting and luggage (which can be suitcases, backpacks or simply bags) can also be interesting as they can pose a threat (a bomb for example) or just as a matter of managing lost objects. Nevertheless, if necessary, the embedded system can be re-oriented and adapted to the final scenario. The work presented in this paper tried to use the embedded nodes as-is to obtain a 3D reconstruction and explore its possibilities.

Some possible applications of the proposed framework have been presented. As future work, we would like to develop a reasoning and decision-making unit that exploits the global knowledge about objects to detect events [16]. If we can model the different situations to detect, this module could trigger alarms and select which information is offered to the surveillance staff, in order to avoid them being overwhelmed with the vast amount of events that can happen at the same time. From a different perspective, we would also like to test different visualization platforms such as autostereoscopic display or CAVEs [4]. The main objective is to study the adequateness of these platforms for the surveillance of large infrastructures.

Acknowledgements This work has been partially supported by the ViCoMo project (ITEA2 project IP08009 funded by the Spanish MICINN with project TSI-020400-2011-57), the Spanish Government (TIN2009-14103-C03-03, DPI2008-06737-C02-01/02 and DPI 2011-28507-C02-02) and European FEDER funds.

We would like to thank the Multimedia services of ASIC at the Universidad Politécnica de Valencia (Spain) for providing the 3D model of the CPI.

References

1. Atienza-Vanacloig V, Rosell-Ortega J, Andreu-Garcia G, Valiente-Gonzalez J (2008) People and luggage recognition in airport surveillance under real-time constraints. In: 19th international conference on pattern recognition, pp 1–4
2. Cal3D (2011) http://gna.org/projects/cal3d/. Accessed 19 July 2012
3. Chang F, Chen CJ (2003) A component-labeling algorithm using contour tracing technique. In: 7th int. conference on document analysis and recognition, pp 741–745
4. Cruz-Neira C, Sandin DJ, DeFanti TA, Kenyon RV, Hart JC (1992) The cave: audio visual experience automatic virtual environment. Commun ACM 35:64–72
5. Fleck S, Busch F, Biber P, Strasser W (2006) 3D surveillance a distributed network of smart cameras for real-time tracking and its visualization in 3D. In: Conference on computer vision and pattern recognition workshop (CVPRW06), p 118
6. Hoiem D, Efros AA, Hebert M (2005) Automatic photo pop-up. ACM Trans Graph 24: 577–584
7. Javed O, Shah M (2008) Automated multi-camera surveillance: algorithms and practice. Springer, New York
8. Lipton A, Fujiyoshi H, Patil R (1998) Moving target classification and tracking from real-time video. In: Proceedings of IEEE workshop on applications of computer vision, vol 1, pp 8–14
9. Lloyd DH (1968) A concept of improvement of learning response in the taught lesson. In: Visual education, pp 23–25
10. Osfield R, Burns D (2011) OpenSceneGraph. http://www.openscenegraph.org. Accessed 19 July 2012
11. Rieffel EG, Girgensohn A, Kimber D, Chen T, Liu Q (2007) Geometric tools for multicamera surveillance systems. In: IEEE int. conf. on distributed smart cameras
12. Sebe I, Hu J, You S, Neumann U (2003) 3D video surveillance with augmented virtual environments. In: ACM SIGMM workshop on video surveillance, pp 107–112
13. SENSE Consortium (2006) Smart embedded network of sensing entities. Web page: http://www.sense-ist.org (European Commission: IST Project 033279). Accessed 19 July 2012
14. Sánchez J, Benet G, Simó JE (2012) Video sensor architecture for surveillance applications. Sensors 12(2):1509–1528
15. Vouzounaras G, Daras P, Strintzis M (2011) Automatic generation of 3D outdoor and indoor building scenes from a single image. Multimedia Tools Appl. doi:10.1007/s11042-011-0823-0
16. Yan W, Kieran D, Rafatirad S, Jain R (2011) A comprehensive study of visual event computing. Multimedia Tools Appl 55:443–481
17. Zúñiga M, Brémond F, Thonnat M (2006) Fast and reliable object classification in video based on a 3D generic model. In: Proceedings of the international conference on visual information engineering (VIE2006), pp 26–28

Oscar Ripolles is a PhD researcher at the Universidad Politécnica de Valencia, Spain. He received his degree in Computer Engineering in 2004 and his PhD in 2009 at the Universitat Jaume I in Castellon, Spain. His research interests include geometry optimization, hardware programming and virtual environments.

José E. Simó got the Industrial Engineering degree in 1990 from the Polytechnic University of Valencia and in 1997 the PhD degree in the same University. Since 1990 he has participated in numerous research projects at a national and European level, mainly related to real-time systems, architectures for process control and artificial intelligence. He has made numerous scientific publications in journals and specialized conferences. He is currently an Associate Professor at the Polytechnic University of Valencia and his research focuses on the architectures of autonomous systems, cyber-physical systems and mobile robotics.

Gines Benet received the Industrial Engineering degree in 1980 in the Polytechnic University of Valencia and in 1988 the PhD degree in the same University. Since 1984 he is teaching at the Polytechnic University of Valencia and his research focuses on embedded systems development, processing information in real time real, intelligent instrumentation and hardware-software co-design. He has made numerous publications in specialized journals and conferences and participated in research projects at national and European level.

Roberto Vivó is an industrial engineer from Polytechnic University of Valencia where he is a professor in the area of languages and systems. He is director of the Institute of Automation and Industrial Informatics (ai2) of the UPV. He is director of this university institute since 2005. He has been president of the Spanish Congress of Computer Graphics (CEIG) and executive member of Eurographics Spanish Section.