J Math Imaging Vis
DOI 10.1007/s10851-011-0290-2

Stereo Video Surveillance Multi-agent System: New Solutions for Human Motion Analysis

Pere Marti-Puig · Sara Rodríguez · Juan F. De Paz · Ramon Reig-Bolaño · Manuel P. Rubio · Javier Bajo

Abstract This article presents a distributed agent-based system that can process the visual information obtained by stereoscopic cameras. The system is embedded within a global project whose objective is to develop an intelligent environment for location and identification within dependent environments that merges with other types of technologies. In this kind of environments, vision algorithms are very costly and require a lot of time to produce a response, which is highly inconvenient since many applications can require action to be taken in real time. A multi-agent system (MAS) can automate the process of analyzing images obtained by cameras, and optimize the procedure. This study presents a MAS that can process stereoscopic images to detect and classify people by combining a series of novel techniques.

The article shows in detail the combination of techniques used to perform the detection process. The process can be subdivided into human detection, human tracking, and human behavior understanding. With the addition of a case-based reasoning (CBR) model, the system can also incorporate reasoning capabilities. The system was tested under different conditions and environments.

Keywords Multi-agent systems · Stereo processing · Human detection · Case based reasoning

P. Marti-Puig · R. Reig-Bolaño
Department of Digital and Information Technologies, University of Vic, C. de la Laura 13, 08500 Vic (Barcelona), Spain

P. Marti-Puig
e-mail: pere.marti@uvic.cat

R. Reig-Bolaño
e-mail: ramon.reig@uvic.cat

S. Rodríguez (✉) · J.F. De Paz
Department of Informatics, University of Salamanca, Plaza de la Merced s/n, 37008 Salamanca, Spain
e-mail: srg@usal.es

J.F. De Paz
e-mail: fcofds@usal.es

M.P. Rubio
Engineering Graphics Department, University of Salamanca, Avd. Requejo no 33, 49001 Zamora, Spain
e-mail: mprc@usal.es

J. Bajo
Department of Informatics, Pontifical University of Salamanca, Compañía, 5, 37002 Salamanca, Spain
e-mail: jbajop@upsa.es

1 Introduction

One of the greatest challenges for the scientific community is to find more effective means of providing care for the growing number of people that make up the disabled and elderly sector [18]. The importance of developing new and more cost-effective methods for administering medical care and assistance to this sector of the population is underscored when we consider current tendencies. Artificial intelligent systems have been recently examined as potential medical care supervisory systems. Among those systems are, multi-agent systems (MAS) [1, 10, 23, 26] for elderly and dependent persons, providing continual support in the daily lives of these individuals; other examined systems are artificial vision systems, where we find medical image analysis and high level computer vision [47, 64]. The study of artificial vision, specifically stereoscopic vision, has been the object of considerable attention within the scientific community over the last few years. Image processing applications are varied and include aspects such as remote measurements, biomedical images analysis, character recognition, virtual

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 2/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

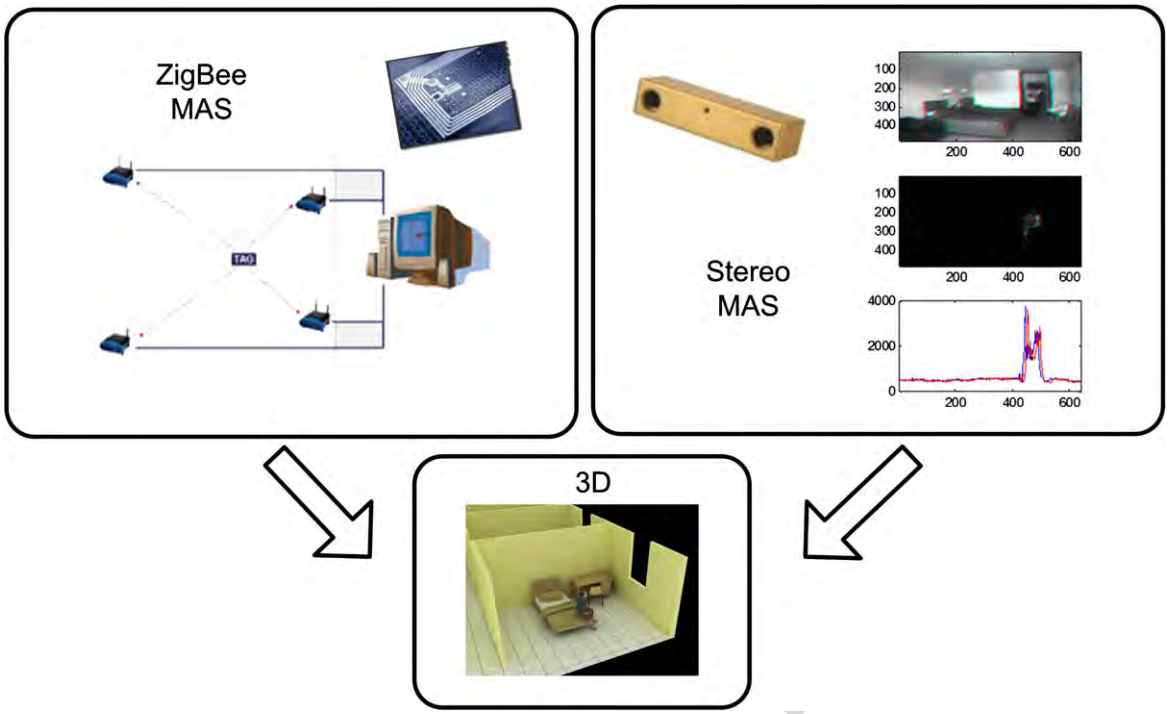

Fig. 1 System for the care and supervision of patients in dependent environments

reality applications, and enhanced reality in collaborative systems, among others. However, it is still an open trend and the use of multiagent-systems for improving the stereoscopic image data processing can help to construct effective intelligent environments.

The main topic of our research is part of a larger, global project whose objective is to develop an intelligent environment for the care and supervision of patients in dependent scenarios, providing an environment capable of automatically carrying out location, identification and patient monitoring tasks [23]. Such an environment would also allow medical personnel to supervise patients as well as to simulate situations. In order to reach this objective, artificial intelligence techniques, intelligent agents, wireless technologies and vision systems are used.

Within the larger scope of the project, the current attention is being focused on vision systems and 3D representation. Figure 1 shows the three main modules of our system. The upper-left module shows location wireless technologies. The upper-right module shows the vision system. The bottom module shows a representation of three-dimensional data. The study presented in this article focuses on the vision module, specifically the development of an agent-based distributed architecture that allows for the processing of visual information obtained by stereoscopic cameras.

For many years, the scientific community has demonstrated an increasing interest in the study of artificial vision. Image processing applications are varied and include such aspects as remote control, biomedical image analysis, character recognition, virtual reality applications, and enhanced reality in collaborative systems, among others. Although image analysis and people detection is a well-explored topic, the use of multi-agent technology in this area has become the focal point of important interest [17, 66]. The capabilities of commercial hardware to solve the low-level problems of stereo processing have turned it into an attractive sensor to develop intelligent systems. Stereo vision provides a type of information that offers several advantages in the development of human–machine applications based on artificial vision. The advantages come from the fact that stereo vision is based on a single physical point in the scene that projects to a unique pair of image locations in two observing cameras. Applications such as stereo-movies, post-production or 3D reconstruction are indeed all based on the same basic ingredient: finding the depth of a scene as viewed from several cameras [5, 40, 80].

This article presents a system that is capable of processing stereoscopic images and detecting people with a stereo camera, automatically identifying as states of interest a person who is standing or lying on the bed. The detector agent is based on robust and low complexity algorithms, with the additional advantage that they can be executed in real time with a low-cost hardware. The system was tested in a small indoor environment characterized by very different lighting conditions in which it had to track people who remained at a very low activity level for a long time. In addition, there

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 3/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

were many situations in which the subject being monitored suffered partial occlusion.

Many techniques used in foreground identification are based on an adaptive background. In our application we found that background updating techniques introduce many errors. After testing various strategies adopted from those found in previous publications, we decided to base the tracking method on a motion detection algorithm. For the detection of presence and location, the system can use the wireless technologies included in the location module [23, 29]. The module has a channel identification based on ZigBee [29]. Motion detectors can be very sensitive and are capable of detecting movements even when the patient is sleeping.

In addition to obtaining fast algorithms that can run in real time, we decided to use the lateral projection of motion estimations. These projections will be the input data to classify the detection of the patient state. The information disparity is estimated directly from the lateral projections. All these algorithms are not only simple enough, but also robust and suitable for real-time execution.

The remainder of this paper is structured as follows: Sect. 2 explains the background related to the key technologies involved in this study; Sect. 3 describes in detail the approach used in the vision module; Sect. 4 describes the experiments and results carried out within a specific case study; and Sect. 5 presents conclusions and further work.

2 Background

Computational stereo refers to the problem of determining three-dimensional structure of a scene, from two or more images, taken from distinct viewpoints. Stereo vision is employed in a wide range of applications, e.g. industrial inspection for 3-D objects, autonomous vehicles, robotics, image-based rendering and more [3, 40, 56, 80].

A point of interest is the detection of people in a stereo shot sequence; their positioning, their tracking in a room and the discrimination of their body pose in real time. Several approaches have been used for human motion analysis, like the ones referred in [53] and [54], however we will follow the taxonomy defined in [75] with three major tasks in the process of human motion analysis (namely human detection, human tracking and human behaviour understanding).

The next three subsections will focus on this taxonomy, while the last subsection will show a short background of multi-agent systems and reasoning models used in the process of human behaviour understanding.

2.1 Human Detection

Typical human detection strategies are based on one of the following techniques or a combination of them: background subtraction, temporal differencing and optical flow analysis.

The background subtraction technique attempts to detect moving regions in an image by differencing pixel by pixel between the current image and a reference background. This common approach has evolved with several methods incorporating dynamic update in the background [41] or [46], trying to adapt the background to the lighting conditions, or even small changes in the background scene. Moreover, one of them [9] had introduced the disparity information from stereo pair images to achieve a better result. They have been proved effective for background modelling: single Gaussians [44, 76] a mixture of Gaussians [36, 70], or even median [28] or minimum-maximum values [41], etc. However most of them have limited results when they have to deal with sudden changes in illumination, environmental changes (such as new objects appearing on scene, or changes in the positioning of objects or furniture), the existence of an extended period of immobility for the human individuals, or occlusions of people in the scene. All these situations are very common in our case study, as detailed in Sect. 4.

Many of the used background modes exploit intensity, disparity, edges or any combination of these magnitudes. The background is updated according to some statistical criteria. In our particular case, due to variable lighting scenarios, the backgrounds based on intensity have a highly variable pattern. From intensity, it is difficult to obtain an algorithm fast enough to follow a fast illumination change; this change incurs a long stream of errors. Another problem occurs when the patient being monitored stays still, almost motionless, for a long time, as when for instance, the patient is sleeping in bed. In these situations updated background models can cause any substantive part of the body to remain integrated into the background. When considering the use of the edges, the biggest problem we have found throughout our work is also due to the variability of illumination. Wall lamps introduce sets of lights and shadows that lead to false contours even on the smooth walls, while natural illuminations avoid these false contours. The information of disparity is nevertheless more stable but—as it is well known—it can only be exploited in the areas of the image with sufficient variability. In areas with smooth textures like portions of the image representing walls this magnitude cannot be evaluated. The disparity information is often helpful to distinguish the shadows and to segment two objects when there is partial overlap between them. However, when shadows appear on a portion of the image in which there is no disparity information, they are detected as part of the object of interest.

Another possible approach to the human detection is the calculation of temporal differencing, using two or three consecutive frames from the video stream, and obtaining the absolute difference between them. These methods are very adaptive to dynamic environments, but generally do a poor job of extracting all relevant feature pixels. Moreover, [20]

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 4/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

proposes a combination of adaptive background subtraction and three-frame differencing. In our case we have adopted an approach based exclusively on temporal differencing: we use this information in conjunction with presence sensors based on RFID and ZigBee technologies, from the ZigBee MAS of Fig. 1.

Optical flow [11] is the last common approach to human detection; it calculates an approximation to a vector map of the estimated movement of the pixels obtained from two consecutive frames. It can be used to detect moving objects independently in the presence of camera motion; however, most optical flow computation methods are computationally complex and cannot be applied to full-frame video streams in real time without specialized hardware. Sometimes, to complete the human detection module, an object detection part is required. This part separates the human from other moving objects based on their shape characteristics, their silhouette (when it is possible to extract) or even their motion related to a kinematics model. Note that this step might be unnecessary under some situations where the moving objects are known to be human.

Other possible approaches are based on skin colour detection. This approach cannot be used in our case, as our stereo pair images are monochromatic.

2.2 Human Tracking

The main objective of the stage for tracking people in motion is to generate their trajectories over time by tracking their plot to plot position. The output of this stage provides us with the image area occupied by the person and updates it frame by frame. Usually the representations of people and objects to track can be done through a set of representative points [74], or by primitive shapes (rectangles, circles or ellipses) [21], or by the surroundings and the complete silhouettes [77]. Other possibilities are the articulated models linked by articulation points [2]. According to the latter strategy, we can model a person's body through the head, trunk and extremities, associating them with a predetermined movement pattern, or patterns, based on skeletons which have been used for this as well. Of all options, this one was chosen to obtain the rectangles that encompass people, because it provides us with a good initial approach to the problem, which is a prior and necessary step in the rest of the representations.

In some cases the characterization is complemented by information on appearance characteristics (textures and color fundamentally) (appearance features). Appearance can be modeled parametrically with key parameters of probability densities [58], and non-parametrically with their histograms [21]. If the objects we follow do not change while tracking them, they can be replaced by templates. These additional parameters did not fit in our case, since we do not have color information and the texture cannot serve to discriminate between people and other objects in the whole scene. Other models incorporate information from multiple views of the objects to follow [55], using systems with multiple cameras at different points in the scene where the tracking is carried out. Neither case applied to our particular situation since we have a stereo camera situated in a corner of the room.

To summarize the different tracking methods at the workplace [78] proposes a taxonometric classification and provides a detailed description of the main approaches, which range from the simple deterministic systems based on tracking singular points [74], to systems based on silhouettes and their correspondence in tracking models [67].

In our case, we have carried out a bounding box tracking, based on the lateral histograms of image differences. With the use of stereo cameras, the location becomes straightforward when we have a dense disparity map.

2.3 Human Behavior Understanding

After successfully tracking the movement of the human subject from one frame to another in an image sequence, the problem of understanding human behaviours from the perspective of image sequences becomes apparent. Behaviour understanding involves action recognition and description and may be simply considered a classification problem of time varying feature data, i.e., matching an unknown test sequence with a group of labelled reference sequences representing typical human actions. However, when there is a need for a more complete description, other approaches are convenient: Dynamic Time Warping [45], Hidden Markov Models [35] or Neural networks [38]; followed by an action recognition step and semantic description phase.

Existing techniques can be grouped into the following types based on the nature of the algorithms used: Naive Bayes probabilistic models [34], Hidden Markov Models [35], Fuzzy Logic: K-NN (K-Nearest Neighbours), NN (K-Nearest Neighbours) [4], roles: Dynamic Time Warping [45], Sequential Minimal Optimization (SMO) [60], trees and decision rules CART (Classification and Regression Trees) [12] C4.5, C5.0/See5 [63] RIPPER [19], Neural networks [38].

For a set of general data it is not possible to determine in advance the classifier that will lead to better results, as there is a tendency to combine the outputs of several classifiers to generate a specific output. This technique is known as ensemble [79] although in a more generic mode it is called mixture of experts. At present, the mixture of experts technique is being used in various studies [39, 71]. Mixture of experts makes it possible to select and merge the outputs of various processes to generate a response that best suits the final value [49]. The mixtures used are usually

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

limited to selecting maximum values of the output or determining/calculating their weighted means to estimate the final value. In more complex combinations neural networks are applied to combine the outputs [57]. Basically these techniques of combining classifiers are grouped into Bagging (Bootstrapping aggregation) or Bagging classifiers, and Boosting [12] or Ada-Boosting [37]. However, these procedures do not allow a mixture to minimize the output error depending on several factors. For this reason it is necessary to include a procedure that can mix the output of several classifiers, according to the predictive ability of each one of them, by using a combination of their outputs.

2.4 MAS and CBR

Agents are autonomous software entities [14] able to interact with their surroundings, and highly capable of adapting to changes. Agents can communicate with other agents and work in a coordinated manner. For this reason, Multi-agent systems (MAS) facilitate the development of dynamic environments such as patients care and supervision. Moreover, agent-oriented methodologies provide a mechanism for modelling distributed, inter-operable and secure systems by taking social and organizational considerations into account and by integrating multiple devices, sensors and humans.

The use of deliberative BDI (Belief, Desire, Intention) agents [13, 65] is essential in the development of the system we are proposing. Apparently, the human visual system deals with a high level of specialization when it comes to classifying and processing the visual information that it receives, such as reconstructing an image by texture, shadow, depth, etc. Computationally, it is difficult to compete with such specialization and to separate from an image only the relevant information for any particular purpose. In response to this problem, we propose implementing a distributed agent-based architecture that will allow visual information contained in an image to be processed in real time. An agent-based distributed architecture, which runs on demand, allows code to be moved to places where actions are required. This allows run-time responses, autonomy, continuity of services and greater levels of flexibility and scalability than centralized architectures [7, 15].

Because the system is capable of generating knowledge and experience, the effort involved in programming multiple tasks will also be reduced since it would only be necessary to specify overall objectives, allowing the agents to cooperate and achieve the stated objectives.

The agents must be capable of both independent reasoning and joint analysis of complex situations in order to be able to achieve a high level of interaction with humans [8]. Although multi-agent systems already exist and are capable of gathering information within a given environment in order to provide medical care [23], there is still much work to be done. It is necessary to continue developing systems and technologies that focus on the improvement of services in general. These technologies can help to construct more efficient distributed systems capable of addressing new problems.

A case-based reasoning system (CBR) [1] embedded within a deliberative agent allows it to respond to events, to take the initiative according to its goals, to communicate with other agents, to interact with users, and to make use of past experiences to find the best plans to achieve goals. The learning capabilities of the CBR systems are due to their inherent structure, which is composed of four main phases [1]: *retrieval*, *reuse*, *revision* and *retention*. In the first phase, the most similar cases to the proposed problem are *retrieved* from the case base. Once a series of cases are extracted from the case base, they must be *reused* by the system. In this second phase, an adaptation of the selected cases is done to fit the current problem. After giving a solution to the problem, that solution is *revised* to check if the proposed alternative is a solution to the problem. If the proposal is confirmed as a solution, then it is *retained* by the system and could eventually serve as a solution to future problems.

This cycle is integrated within the activities of the BDI agent [22] and identifies the phases as tasks or roles that the agent should be able to perform. This makes up for one of the primary deficiencies in the BDI model, which involves the manner in which memory and past experiences are handled. In [22, 25] a method is presented for incorporating a CBR engine to the BDI model. The main idea in these studies is the use of the mechanisms provided by the deliberative BDI model, (namely; Beliefs, Desires and Intentions) to be able to obtain a representation of the case and initiate the CBR reasoning cycle. To integrate the CBR reasoning system within the structure of a deliberative BDI agent, [22] proposed a formula relating the case concept to the fundamental concepts of BDI. The relationship between CBR systems and BDI agents can be established by implementing cases as beliefs, intentions and desires, which leads to the resolution of the problem. As described in [22], in a CBR-BDI agent, each state is considered a belief; the objective to be reached may also be a belief. The intentions are plans of actions that the agent has to carry out in order to achieve its objectives. So an intention is an ordered set of actions; each change from state to state is made after carrying out an action (the agent remembers the action carried out in the past, when it was in a specified state, and the subsequent result). A desire will be any of the final states reached in the past (if the agent has to deal with a situation, which is similar to a past one, it will try to achieve a similar result to the previously obtained one).

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 6/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

3 Our Approach

As mentioned earlier, the study presented here focuses on the vision module of a MAS. The different processes are implemented over a distributed agent-based architecture, which allows it to run tasks in parallel using each service as an independent processing unit. The architecture would allow a stereoscopic image processing system to carry out its own phases, which could be distributed among the agents. Thus data gathering, preprocessing, filtering and reconstruction, as well as human form detection, could all be carried out in parallel. A description and initial proposal for this global architecture can be found in [66]. The system is comprised of a set of agents with defined roles that share information and services. The image analysis involves a complex process where each agent executes its task with the information available at each moment.

As shown in Fig. 2, the data obtained from the stereoscopic camera are entered into the system and shared between the agents that will use specific services to process the data (filtering, preprocessing, disparity analysis, etc.) [66]. The system outputs can be located on the high-density disparity map obtained from the distance between the camera and the objects, the numerical representation of these distances, their three-dimensional representation in real time, and/or the detection of human forms in the specified area.

A commercial stereo camera [66] was employed in this work because it can capture two images from slightly different positions (stereo pair) that are transferred to the computer to calculate a disparity image containing the points matched in both images. Knowing the extrinsic and intrinsic parameters of the stereo camera, it is possible to reconstruct the three-dimensional position.

People detection and stereo processing are treated as separate processes in this study. Every time a new image is captured, the system must first apply stereo processing to obtain the distances of the objects in the image. After that, the system can decide to apply the people detection to the same image.

Figure 2 shows the steps that occur in a typical sequence of processing images from the stereoscopic camera. Each of the phases depicted in Fig. 2, or parts of it, can be performed by the agents that constitute the system. This study focuses on the detection phase of multi-agent system, and more specifically on the detector agent responsible for carrying out this functionality to allow Human Detection, Human Tracking, and Human behavior understanding. Detection and tracking provide us with the information necessary for the final classification (Human Behavior Understanding). The classification process is detailed in Sect. 3.3 and shows how it is possible to apply multiple classifiers and then make a mixture of their outputs, yielding the final estimate. For this, we propose a CBR model that includes independent techniques to carry out this step.

The detector Agent is a CBR-BDI agent composed of a reasoning cycle that consists of four sequential phases: *retrieve*, *reuse*, *revise* and *retain*. The CBR system is completely integrated into the agents' architecture. The structure of the CBR system has been designed around the case concept. In order to initially construct the model case base starting from the available histogram data, the CBR stores the

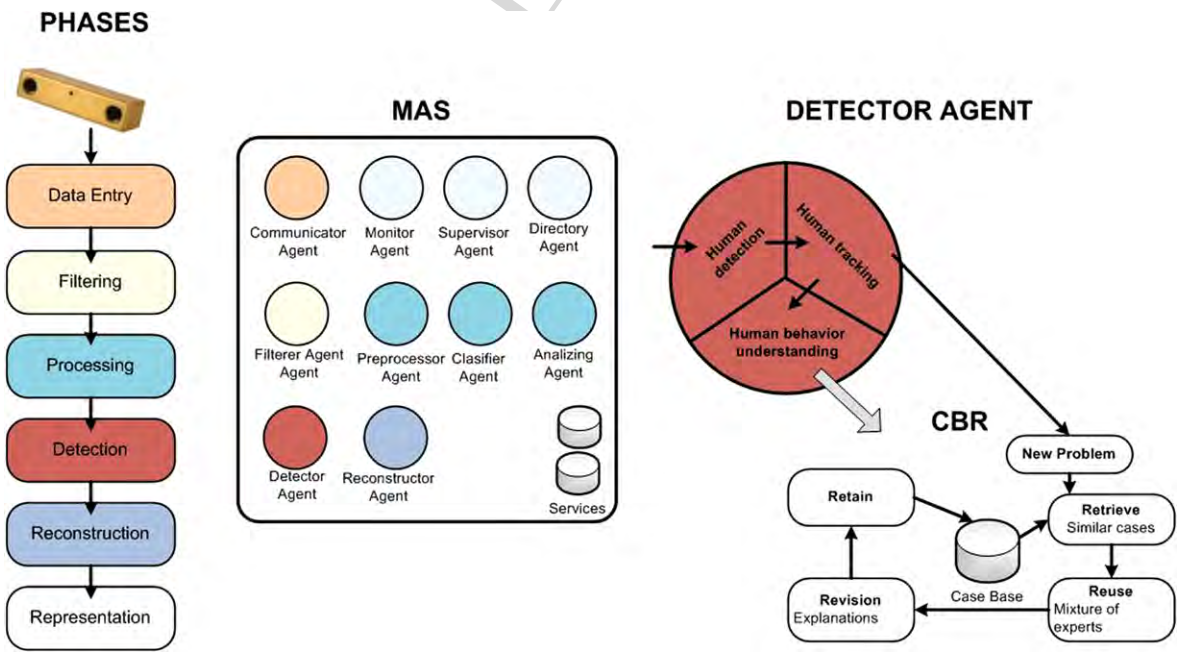

Fig. 2 Phases, MAS and CBR

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 7/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

histograms obtained with the human detection and tracking techniques.

That is, once the preprocessing for the dimensionality reduction and the extraction of relevant information has been completed, we proceed to the classification process. To perform the classification process we break up the information obtained from the horizontal and vertical histograms.

During the retrieval stage, we select the most similar cases to the present case, taking into account the type of illumination. This phase can be thought of as a data selection step that aims to retrieve the list of cases that might be most informative given a new sample to classify.

The adaptation of previous cases in order to solve a new problem is accomplished in the reuse stage (right bottom square in Fig. 2). A mixture of experts, explained below, is used in this stage.

In the revision stage, the expert is provided with useful data about the decision made by the system. The expert contrasts the initial prediction given by the system with other external information, such as patient history, in order to ascertain a revised prediction and a final classification.

Every time a new problem is solved, the internal structure of the CBR system is updated (retain stage). The new case is associated with its corresponding class and added to the case base. The case base is updated and the system marks the most similar cases selected for future classifications.

3.1 Human Detection

Most of the modules for people detection with monocular or stereo cameras are based on a background estimation and subtraction [9]. However, our approach is based on the measurement of frame differences; this is basically due to the better performance of this option compared to a background estimation system based on median values of successive frames. Other more sophisticated approaches to background estimation failed to reproduce in real-time calculation. After several trials on images captured under working conditions, we found that when creating a balance between the calculation complexity and the obtained results, the best results for the movement detection would be with a measure of frame differences. Therefore, we use a 2-frame absolute differences measure, which has better results than a single frame differences.

In order to better capture the movement, the information can be extracted from three consecutive frames instead of only two. Thus, $\mathbf{I}_t^R(m,n)$ represents the gray level right image intensity of the stereo pair at frame t—with m,n being the pixel coordinate indices—, and the right differences $\mathbf{D}_t^R(m,n)$ can be defined as:

$$\mathbf{D}_t^R(m,n) = |mean\{(\mathbf{I}_t^R(m,n) - \mathbf{I}_{t-1}^R(m,n)), (\mathbf{I}_{t-1}^R(m,n) - \mathbf{I}_{t-2}^R(m,n))\}| \quad (1)$$

Where $|.|$ represents the operation module. To simplify, the calculation of (1) is also obtained from:

$$\mathbf{D}_t^R(m,n) = k \cdot |\mathbf{I}_t^R(m,n) - \mathbf{I}_{t-2}^R(m,n)|, \quad (2)$$

where the constant parameter k takes the value $2/3$.

When using $k = 2/3$ and an image capture rate of 15 frames per second, the movement detection is obtained from images taken at a temporal distance of 133.3 ms. The same movement resolution could be obtained working at half of the image capture rate and taking the image differences of two time consecutive frames.

The same operations in (2) are then made with the stereo left images, to obtain the left absolute differences $\mathbf{D}_t^L(m,n)$.

3.2 Human Tracking

The objective for the human tracking phase is to automatically find the box that encloses the person being monitored. In order to accomplish this objective, the two frames of absolute differences of the stereo pair images are projected onto the vertical and horizontal axes. Sometimes these are called lateral histograms [31], and are calculated by summing the gray levels of the pixels in each of the columns and rows of the image differences.

In (3) we define the horizontal and vertical histograms of the right stereo image differences.

$$\mathbf{h}_t^R(m) = \sum_n \mathbf{D}_t^R(m,n), \qquad \mathbf{v}_t^R(n) = \sum_m \mathbf{D}_t^R(m,n) \quad (3)$$

The same definition is used for the horizontal and vertical histograms of left stereo differences $\mathbf{h}_t^L(m)$ and $\mathbf{v}_t^L(n)$.

With this operation the 2-D information required to perform the human tracking in the images is reduced to the 1-D discrete function, which is more suitable for real time processing. This information will then be classified using different statistical methods.

We take the mean of the right and left horizontal projections to obtain a single 1-D horizontal projection $\mathbf{h}_t(m)$, and we repeat the same process to obtain the vertical projection $\mathbf{v}_t(n)$.

In order to find the horizontal and vertical box sizes we establish a threshold on the simplified versions of the signals $\mathbf{h}_t(m)$ and $\mathbf{v}_t(n)$.

The simplification of $\mathbf{h}_t(m)$ and $\mathbf{v}_t(n)$ is performed by applying a 1-D morphological filtering. The main advantage of morphological operations is that they preserve the positions of the maximums, the minimums, and the edges of the original function. The mathematical morphology was first developed for black and white images, but has been extended to work with gray and color images as well as one-dimensional functions [68].

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 8/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

There are two basic morphological operations; erosion and dilation. These are computed using a structuring element Y. There are several structuring elements. Morphological operations are strongly dependent on the structuring element. We have selected a flat structuring element of a given length L that in the 1D context can be seen as a mobile window. Then the 1D erosion computes the minimum value of the part of the function inside a mobile window defined by the structuring element Y. Thus, the erosion of the function $f(m)$ by the structuring element Y, $\varepsilon_Y(f(m))$, decreases peaks and accentuates valleys. The dilation of $f(m)$, $\delta_Y(f)$ gives the maximum value of the part of the function inside the mobile template defined by Y, accentuating peaks and minimizing valleys. By combining dilation and erosion we can form other morphological filters. The opening and the closing are the basic morphological filters.

The morphological opening of $f(m)$ by the structuring element Y is denoted by $\gamma_Y(f)$ and is defined as the erosion of $f(m)$ by Y followed by the dilation by the same structuring element. Similarly, the morphological closing of $f(m)$ by the structuring element Y is denoted by $\varphi_Y(f)$ and is defined as the dilation of $f(m)$ by Y followed by erosion by Y. The filter output function is more simple than the input function because the details of the original function that are smaller than the length of the structuring element Y will disappear.

There is an order relation between the opening, the original function and the closing given by:

$$\gamma_Y(f(m)) \leq f(m) \leq \varphi_Y(f(m)) \tag{4}$$

The opening is always less than or equal to the original function while the closing is always greater or equal.

We have used the opening to simplify $\mathbf{h}_t(m)$ and $\mathbf{v}_t(n)$ and to obtain the threshold. Accordingly, by means of a closing we define $\mathbf{ch}_t(m)$ and $\mathbf{cv}_t(n)$ as:

$$\mathbf{ch}_t(m) = \varphi_Y(\mathbf{h}_t(m)) = \varepsilon_Y(\delta_Y(\mathbf{h}_t(m))) \tag{5}$$

$$\mathbf{cv}_t(n) = \varphi_Y(\mathbf{v}_t(n)) = \varepsilon_Y(\delta_Y(\mathbf{v}_t(n))) \tag{6}$$

The election of the length L—the unique parameter in a 1-D flat structuring element—is determined by taking into account the size of the elements to preserve in $\mathbf{ch}_t(m)$ and $\mathbf{cv}_t(n)$. The object of interest is the human to be detected. His or her size in the room scene must be preserved. The higher the length L of Y the simpler $\mathbf{ch}_t(m)$ and $\mathbf{cv}_t(n)$ are. A good level of simplification can be obtained by taking values of L that are between 5 and 12 parts of the maximum number of pixels of the image scene. As the used images have 640×480 pixels, we have chosen a flat structuring element Y of length 100 pixels although the system works well for a wide range around this value.

We decided that the movement in the x-direction, at frame t, is concentrated in the parts of $\mathbf{ch}_t(m)$ where, considering $\mathbf{ch}_t(0)$ and $\mathbf{ch}_t(M-1)$, the extreme values of $\mathbf{ch}_t(m)$ are greater than:

$$(1-\alpha)\max\{\mathbf{ch}_t(m)\} + \max\{\mathbf{ch}_t(0), \mathbf{ch}_t(M-1)\} \tag{7}$$

where α is a number between 0 and 1. We proceed in the same way for the *vertical* direction. The sections that are moving must to be greater than:

$$(1-\alpha)\max\{\mathbf{cv}_t(n)\} + \max\{\mathbf{cv}_t(0), \mathbf{cv}_t(N-1)\} \tag{8}$$

where $\mathbf{cv}_t(0)$ and $\mathbf{cv}_t(N-1)$ are the extreme values of $\mathbf{cv}_t(n)$. We have worked using $\alpha = 0.85$.

If in $\mathbf{D}_t^R(m,n)$ the number of pixels representing the movement is zero, the information provided by image t can be discarded.

Another advantage of working with these lateral histograms of right $\mathbf{h}_t^R(m)$ and $\mathbf{v}_t^R(n)$, and left views $\mathbf{h}_t^L(m)$ and $\mathbf{v}_t^L(n)$ is that we can rapidly measure a rough value of the disparity as it is represented in the results section.

The morphological operations can be performed with fast algorithms optimized as those described in [35], the lateral projection operations are not expensive in computing time.

3.3 Human Behavior Understanding

Finally, for the human behaviour understanding phase, in this first approach, the human detection agent will classify two positions: standing/walking and lying. For the classification process we used the lateral histograms of the image differences as a data classifier against a trained classifier.

To perform the final calculation, several classifiers were applied and then a mixture of their outputs was made to provide the final estimation. The final output is based on the minimization of the final error of classification. The starting point for creating the experts mixture is based on the calculation of the output based on the weighted mean of classifiers as shown in (9).

$$f(x_1, \ldots, x_n) = \sum_i w_i x_i \quad \text{with} \sum_i w_i = 1 \tag{9}$$

Where x_i represents values obtained by the experts and w_i the weight values. To set the weights value, we define the set of variables that affect the final estimation. In this case we have taken into account several factors to calculate the final weights: the estimation of error in calculating the average of the values estimated by experts, the variance of the outputs, and the hit rate. The following sections set out the relevance of each of these values. Generally, the factors affecting the final weights are denoted as p_i for both the expression in (9) and as shown in (10)

$$f(x_1, \ldots, x_n) = \sum_i (p_1^i + \cdots + p_n^i) x_i \tag{10}$$

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

The factors are defined so that they meet the condition (11)

$$\sum_i p_j^i = 1 \tag{11}$$

From the expression (10) we set out a series of variables that determine the relevance degree of each factor in calculating the estimated final value, thus obtaining (12). These variables will serve to minimize the final error.

$$f(x_1, \ldots, x_n) = \sum_i (w_1 p_1^i + \cdots + w_n p_n^i) x_i \tag{12}$$

In order for (12) to meet the definition of weighted sum, the condition defined in (13) must be given as

$$\sum_i (w_1 p_1^i + \cdots + w_n p_n^i) = 1 \tag{13}$$

Taking into account the expression (11) we can simplify (13) the expression as follows:

$$\begin{aligned} &w_1 p_1^1 + w_2 p_2^1 + \cdots + w_n p_n^1 + w_2 p_1^2 + w_2 p_2^2 + \cdots \\ &\quad + w_n p_n^2 + \cdots + w_1 p_1^n + w_2 p_2^n + \cdots + w_n p_n^n = 1 \end{aligned}$$

Separating the term as the following manner, it is clear that the term corresponding to the last line is simplified with the first term of the previous expressions, leaving only the terms: $w_1 p_1^1, w_1 p_1^2 \ldots$, the same thing happens with other terms.

$$\begin{aligned} &w_1 p_1^1 + w_2 p_2^1 + \cdots + w_n p_n^1 \\ &w_1 p_1^2 + w_2 p_2^2 + \cdots + w_n p_n^2 \\ &\vdots \\ &w_1 p_1^{n-1} + w_2 p_2^{n-1} + \cdots + w_n p_n^{n-1} \\ &w_1 (1 - p_1^1 - p_1^2 - \cdots - p_1^{n-1}) \\ &\quad + w_2 (1 - p_2^1 - p_2^2 - \cdots - p_2^{n-1}) \\ &\quad + \cdots + w_n (1 - p_n^1 - p_n^2 - \cdots - p_n^{n-1}) \end{aligned}$$

Simplifying even more, we finally get

$$w_1 + \cdots + w_n = 1 \tag{14}$$

The goal is to find the set of values of w_i that minimize the final error value in the estimation given the values of p and x. The calculation of this value is part of the definition of the mean square error to measure the level of error, which leaves us with the expression to minimize expression (15) subject to the indicated restrictions

$$\begin{aligned} &f(w_1, \ldots, w_n) = \sum_i ((w_1 p_1^i + \cdots + w_n p_n^i) x_i - y_i)^2 \\ &\text{s.t.} \quad 1 - w_1 - \cdots - w_n = 0 \end{aligned} \tag{15}$$

Applying Lagrange

$$\begin{aligned} f(w_1, \ldots, w_n, \lambda) = &\sum_i ((w_1 p_1^i + \cdots + w_n p_n^i) x_i - y_i)^2 \\ &- \lambda (1 - w_1 - \cdots - w_n)) \end{aligned} \tag{16}$$

From expression (16) we obtain the system of (17) that allows us to calculate the values of $w_1, \ldots, w_n$ that minimize the error of the final classification

$$\begin{aligned} \frac{\partial f}{\partial w_1} &= \frac{\partial \sum_i ((w_1 p_1^i + \cdots + w_n p_n^i) x_i - y_i)^2 - \lambda (1 - w_1 - \cdots - w_n)}{\partial w_1} \\ &\vdots \\ \frac{\partial f}{\partial w_n} &= \frac{\partial \sum_i ((w_1 p_1^i + \cdots + w_n p_n^i) x_i - y_i)^2 - \lambda (1 - w_1 - \cdots - w_n)}{\partial w_n} \\ \frac{\partial f}{\partial \lambda} &= \frac{\partial \sum_i ((w_1 p_1^i + \cdots + w_n p_n^i) x_i - y_i)^2 - \lambda (1 - w_1 - \cdots - w_n)}{\partial \lambda} \end{aligned} \tag{17}$$

Then we present the experts utilized to carry out the classification: SVM and MLP.

3.3.1 *Support Vector Machine*

The Support Vector Machine (SVM) is a supervised learning technique applied to the classification and regression of elements. SVM can be applied in a variety of fields such as chemistry, ambient intelligence, modelling and simulation, and data or text mining. The algorithm represents an extension of the linear models [73]. Originally developed for the classification of linearly separable problems, it basically consists of finding the straight line or hyper plane (in two or more dimensions) that makes it possible to separate the elements of a set. SVM can also separate different classes

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 10/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

of elements that cannot be separated linearly. To do so, it uses functions to map out the initial space of coordinates in a highly dimensional space. Because the dimensionality of the new space can be so high, it is not practical to calculate the hyperplanes that perform the linear separation. Instead, a series of non linear functions known as kernel Φ are used, where x_i is a vector with n-dimension, the idea is to convert the elements x_i in a highly dimensional space using the application of a feature function $\Phi(x)$. The following equation is used to perform the classification (18) [43].

$$class(x_k) = signe[w\Phi(x_k) + b] = signe\left(\sum_{i=1}^{m} \lambda_i y_i \Phi(x_i)\Phi(x_k) + b\right) \tag{18}$$

where x_i is a vector with n-dimension, the idea is to convert the elements x_i in a highly dimensional space using the application of a feature function $\Phi(x)$, λ_i is a Lagrange multiplier, and y_i is the output value for the pattern b constant. The calculation of these values is described in [39].

As we can see, there is a product $\Phi(x_i)\Phi(x_k)$ that, according to the dimensionality of the new space, can be very costly to calculate. For this reason, it is necessary to select a series of kernel functions that can operate in the original space to perform these calculations without requiring a heavy computational load.

To calculate the classifier $class(x_k)$ there are algorithms such as the Sequential Minimal Optimization (SMO) [61]. From the hyperplane calculated by SMO, we proceed to calculate the distance of each of the points to the hyperplane. These distances will be calculated to estimate the error in the calculation of the distance and to make the mixture of methods as described in the last paragraph of the subsection classification model. The distance is calculated according to (19)

$$d(x; w, b) = \frac{|w \cdot \Phi(x) + b|}{\|w\|} \tag{19}$$

For each of the complexes in which the input patterns are divided, we create a hyperplane through the application of SMO , therefore generating a set of hyperplanes denoted by $P = \{h_1, \ldots, h_n\}$

$P = \{h_1, \ldots, h_n\}$.

3.3.2 Neural Network

The reasoning memory used by the agent is defined by the following expression: $P = \{p_1, \ldots, p_n\}$ and is implemented by means of a MLP (Multilayer Perceptron) neural network. MLP is the most widely applied and researched artificial neural network (ANN) model. MLP networks implement mappings from input space to output space and are normally applied to supervised learning tasks [38]. A sigmoid function with a range of values in the interval [0, 1] was selected as the MLP activation function. It is used to classify the different states of the people detected in the room.

Entries for the neural network corresponding to the case elements are defined in Table 7. The output corresponds to x^y. Because the neurons exiting from the hidden layer of the neural network contain sigmoid neurons with values between [0, 1], the incoming variables are redefined so that their range falls between [0.2–0.8]. This transformation is necessary because the network cannot work with values that fall outside this range. The outgoing values are similarly limited to the range of [0.2, 0.8] with the value 0.2 corresponding to a non-attack and the value 0.8 corresponding to an attack. Training for the network is carried out by the error Backpropagation Algorithm [50].

When a previously trained network is already available for the set of data associated with the new case, the case classification process is carried out in the revise phase. If a previously trained network is not available, the training is carried out after the entire procedure has been completed, beginning with the cases related to the service and subnet mask, as shown in the above equation.

3.3.3 Relevant Factors

The detected relevant factors were based on the error during the estimation of the average value for each of the following types: the variance of the data and the hit rate. To calculate the average error we assume that $N \gg n$ because the total number of images to estimate, though unknown, is much greater than the set of images used during the training. The error for the mean is defined in terms of expression (20) calculated from the definition of error for estimating the average

$$e = \pm k \frac{S_c}{\sqrt{n}} \tag{20}$$

where k is defined from the stated confidence level, S_c is the quasi-variance and n is the number of elements in the sample.

The final value of the factor is set according to the ratio of the sample mean and the error

$$p_i^e = \bar{x}/e \tag{21}$$

We define a factor for each of the i classifiers, and for each classifier we define a different factor for each different class defined.

Another factor is based on the value obtained as an output for the classifier, taking into account the distance with respect to the average theoretical value of the class, the variance and the value provided by the classifier. Values

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

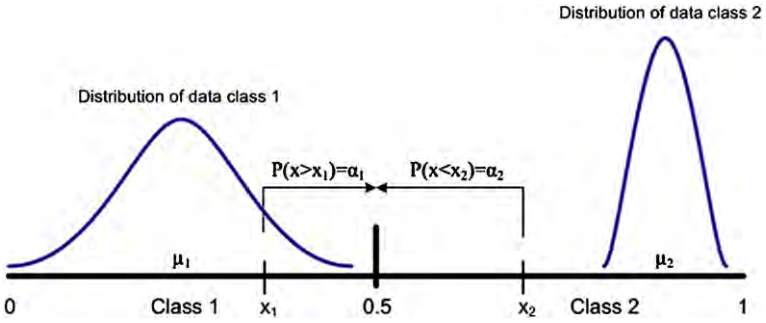

Fig. 3 Distribution of the values of a classifier for each class

corresponding to this class are standardized together with the value obtained by the classifier and then we calculate the probability that z is less than or greater than the k value obtained by standardizing the value of the classifier $P(z < k) = \alpha$. Based on the value of alpha we calculate the weight of this factor according to the original value. Figure 3 provides a graphical representation of the distribution of values obtained from a classifier for classes 1 and 2 (up and down), representing both the normal distribution of the mean μ_1 and μ_2. The values x_1 and x_2 correspond to the estimated value of the classifier for a particular pattern.

The value of the factor for class 1 and class 2 corresponds to as appropriate the value obtained with the case x_1 or x_2 respectively, and is defined by the expression (22). This factor changes for each classifier and for each case.

$$p(x_1) = \begin{cases} k \cdot P(z > \frac{x_1 - \mu_1}{\sigma_1}) & x_1 > \mu_1 \\ 1 & ioc \end{cases}$$
$$p(x_2) = \begin{cases} k \cdot P(z > \frac{x_2 - \mu_2}{\sigma_2}) & x_2 < \mu_2 \\ 1 & ioc \end{cases} \quad (22)$$

where k is a constant, x_1 is the value obtained by the classifier, μ_1 is the average for the values obtained by the classifier for class 1, and σ_1 the variance. We similarly define the variables for the second case.

The last factor taken into account is related to the hit rate for each method. The hit rate is defined by the number of correctly classified cases during the training and estimation phase. The value of the factor is constant for all cases given a particular classifier. Each of the factors is defined to meet (11). To calculate these values the following operation is performed:

$$p_j^i = p_j^i / (p_j^1 + \cdots + p_j^n) \quad (23)$$

where i corresponds to the classifier i.

3.3.4 *Classification Model*

The classification model was applied to the case study of the proposed mixture in Sect. 3.3. In the mixture, the classification models are applied according to the SMO and the MLP, weighted by the factors described in the Relevant factors section This process eventually results in the following model:

$$f(x_1, x_2) = (w_1 p_1 + w_2 p_2^i + w_3 p_3) x_1 + (w_1(1 - p_1) + w_2(1 - p_2^i) + w_3(1 - p_3)) x_2 \quad (24)$$

where w_i is calculated according to (17), the value p_1 is calculated according to (21), p_2^i is calculated from (22), and p_3 contains the hit rate. All these parameters are defined so as to meet (23), while the values of x_1 and x_2 correspond to the estimation calculated by SVM and the MLP respectively.

4 Experiments and Results

A broader experimentation was done to test the processing and detection of different people under different lighting conditions and different distances from the stereo vision system. We employed 640×480 sized images and sub-pixel interpolation to enhance the precision in the stereo calculation. The operation frequency of our system is near 10 Hz on a 3.2 GHz Pentium IV computer running on Windows XP. The camera has the following characteristics [62]: 640×480 pixel sensors, monochrome, 3.8 mm focal distance, capable of capturing 48 photograms per second, 120 mm line base, 6 pin IEEE-1394 (FireWire) interface connection. The images were taken from a height of 1.6 m with a 15 fps velocity, obtaining approximately 400M coded data in AVI and PGM format (16 bits per image).

The environment in which the system was developed is a hospital room. The rooms were small in size, containing one or two beds. The environment was subjected to very different lighting conditions that could change rapidly. For example, there was natural light through a window, and several possibilities for artificial lighting, including ceiling or wall lamps. There was also a door in front of the camera that could change the lighting condition of the scene if it were suddenly opened or closed. These variable conditions create different shades that can be combined in many ways and appear at different angles. Figure 4 shows the scene captured by the camera at six different illuminations. In this figure we

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 12/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

Fig. 4 A set of different illuminations of the room. *Top-down*, and *left* to *right*: wall lamp on with closed door; wall lamp on with open door; ceiling lamp on with closed door; ceiling lamp on with open door; natural light with closed door; and natural light with open door

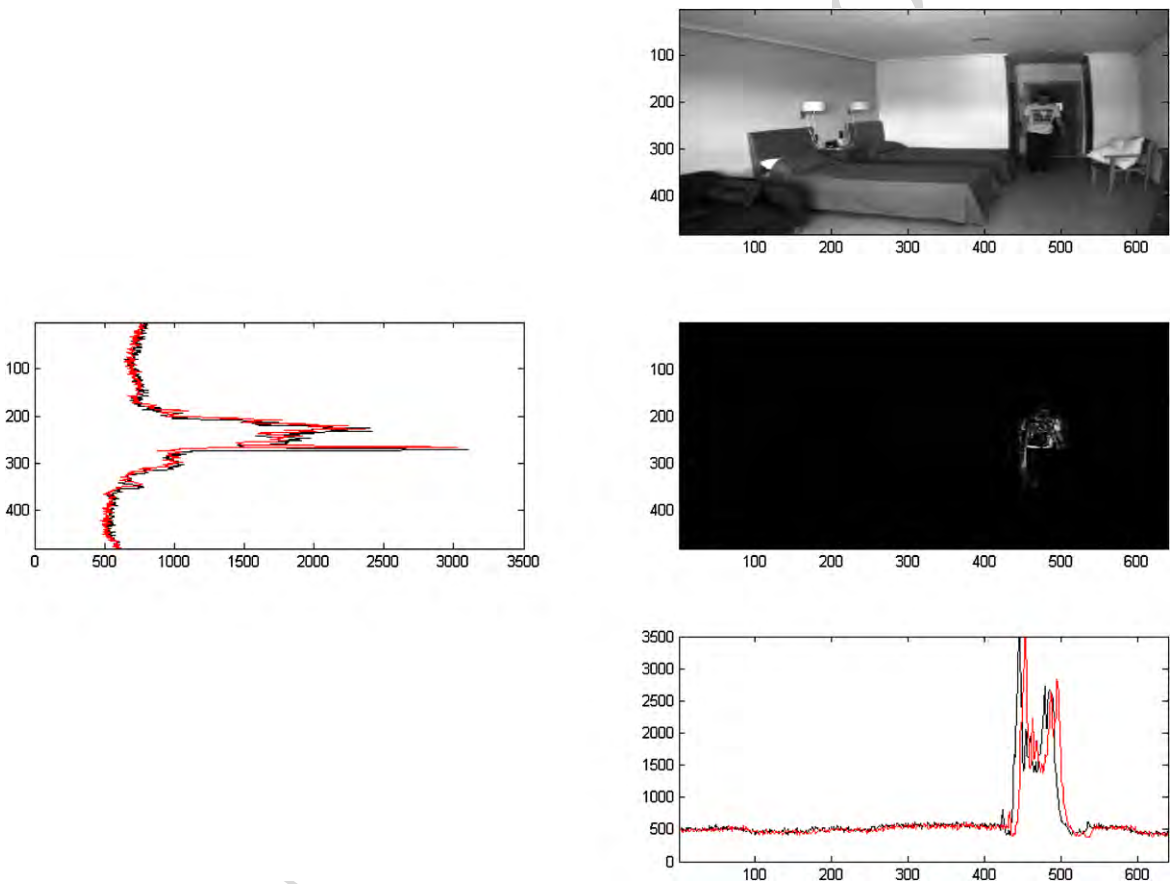

Fig. 5 Lateral histograms of a stereo difference. At the *upper right panel* the original stereo pair in a single image. At the *central right panel* the two frames absolute differences images. At the *lower right panel* the horizontal histogram, or horizontal projection of stereo differences (with *red* representing the right camera, and *blue* representing the left one). At the *left panel* the vertical histograms of stereo differences

can observe that the patterns of shadows and reflections can vary widely.

To perform our analysis, we selected 682 video sequence images captured under different conditions of lighting and human presence in the room. The different images contain one person under different lighting sources: 248 images with natural lighting (37%), 186 images with fluorescent ceiling light (27%), and 245 cases of incandescent wall lighting on the far wall of the image (36%). For each of these various lighting conditions, the individuals in the room were either

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 13/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

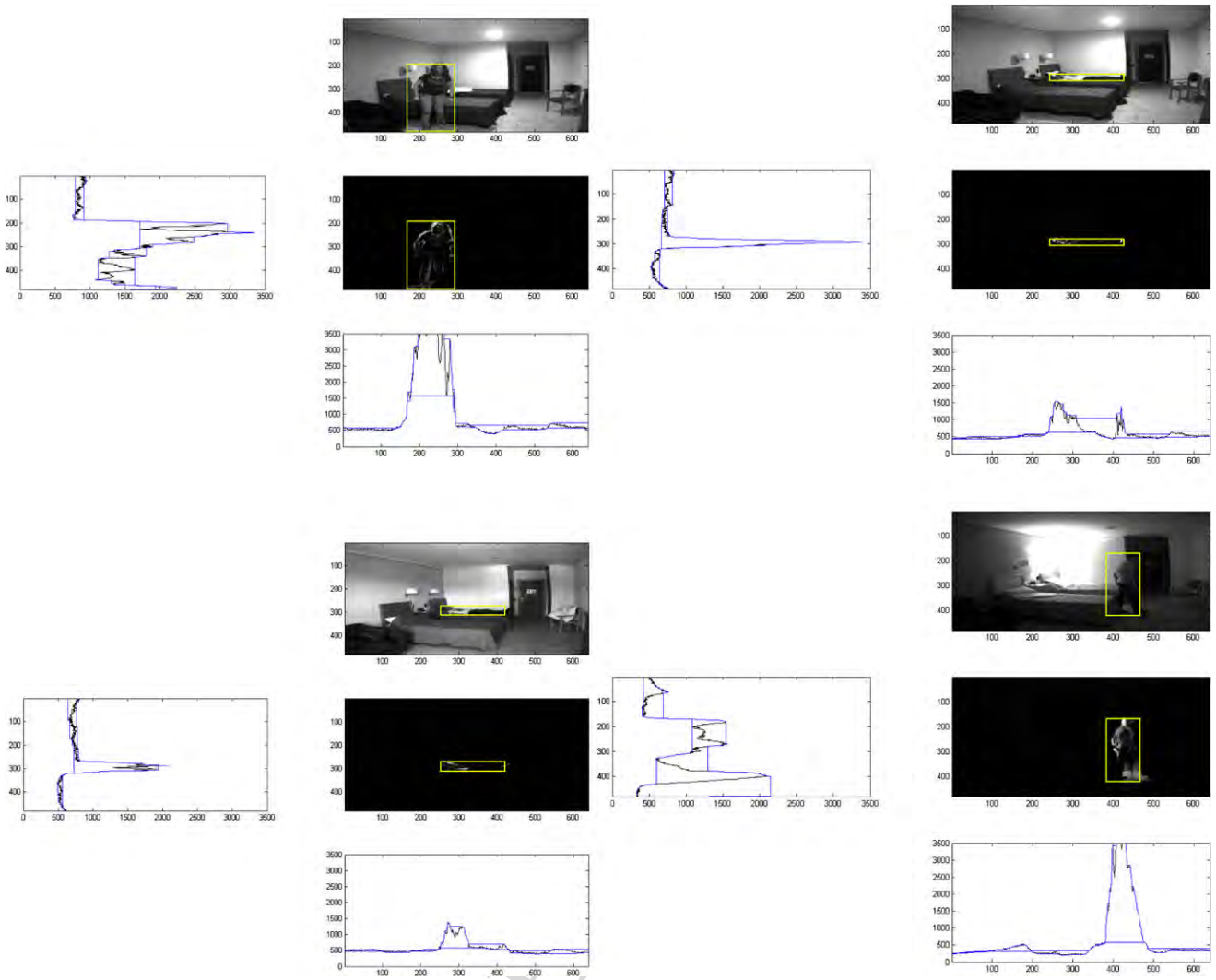

Fig. 6 Examples of surrounding people by a box under different positions and changing lighting conditions

lying down or standing in different areas of the room. Of the 682 images, 680 were correctly classified (lying down or standing). Two images were incorrectly classified, introducing skewed data into the system for the purpose of ensuring the proper functioning of the algorithms. The previous classification was used to train different supervised algorithms and the proposed CBR in order to analyze its functioning.

After the classification of the images it is necessary to recover the information associated with the input parameters used in the algorithms. Therefore, it is necessary, as a first step, to detect the motion of objects in space in order to classify the objects in the scene. The detection of motion is performed by calculating histograms. Figure 5 shows part of the partial calculations realized in the motion detection phases with the lateral histograms and their relationship to a particular scene. The upper part of the image shows the original scene in a gray level image with a person coming through the door. Just below is the output of the motion detector for the instant of capturing the original image, detecting, for this very instant, the parts in motion of the scene. The difference images in the scene are obtained simultaneously from each one of the sides of the stereo camera. The lateral histograms are calculated on these images and the figure represents the lateral histograms which are represented simultaneously using the color red for the right camera and the color blue for the left camera. The horizontal histograms appear in the lower part of the image while the lateral histograms on the left image. It is clear that by using only the information of the histograms we can locate the moving parts of the scene.

Lateral histograms can reduce the data dimensionality that the application works with, thus enabling us to work at the system runtime. The detection and tracking of the person is done using the morphological operations described in Sect. 3.2, using $\alpha = 0.85$.

The tracking process can be observed in Fig. 6 in which four different cases were considered. In this figure the im-

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 14/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

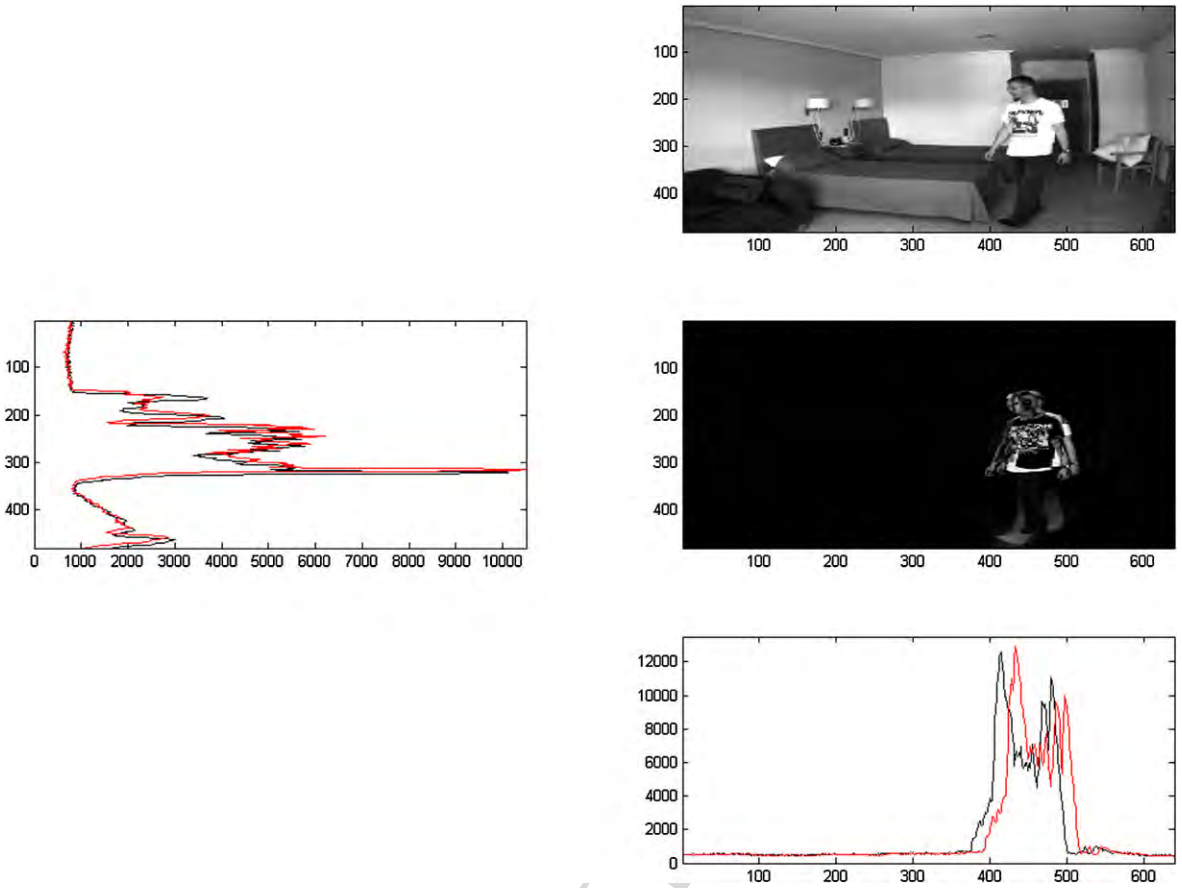

Fig. 7 Lateral histograms of a stereo difference. With high disparity, the person is close to the camera. At the *lower right panel* the horizontal histogram, or horizontal projection of stereo differences (*red* represents the right eye, and *blue* the left one)

age of each person is surrounded by a yellow box in different positions and in different lighting conditions. Looking at a set of four images, and using the same representation as in Fig. 5, the top left image represents the original image, the image below is image D_t. A yellow box encloses the area in which the person is detected. On the left, the vertical projection $\mathbf{v}_t(y)$ is represented in red, $\mathbf{cv}_t(y)$ in blue, and in green the parts of $\mathbf{v}_t(y)$ that are greater than the threshold and determine the vertical axis of the yellow square. Below $\mathbf{D}_t(x, y)$, $\mathbf{h}_t(x)$ is depicted in red, $\mathbf{ch}_t(x)$ in blue, and in green the parts of $\mathbf{h}_t(x)$ that are greater than the threshold determining the horizontal axis of the yellow square.

To have a direct and rough measure of disparity we calculated the maximum value of the cross correlation on the lateral histograms of right and left views.

In Fig. 7 we can observe a person close to the stereoscopic camera equipment, which is separated along the horizontal axis. We see that the horizontal histogram calculated for the right camera, in red, shows a similar but shifted profile with respect to what we obtain from the left camera, in blue.

From the cross correlation between both profiles we can estimate the horizontal component of the disparity.

The position of the maximum represents the disparity between points, and its value could represent the range of the distance, as we can see in Fig. 8.

After analyzing the previous preprocessing for dimensionality reduction and the extraction of relevant information, we proceed to perform the technical analysis of the classification. To perform the classification process we start with the information obtained from the horizontal and vertical histograms of the previous classification and proceed to evaluate the efficiency of the proposed classification technique. To evaluate the significance of the possible classification techniques used during the reuse phase, we performed a comparison between different classifiers following Dietterich's model 5×2-Cross-Validation Paired t-Test algorithm [33]. The value 5 in the algorithm represents the number of replications of the training process, and value 2 is the number of sets into which the global set is divided. Thus, for each of the techniques, the global dataset S was divided into two groups S_1 and S_2 as follows: $S = S_1 \cup S_2$ and $S_1 \cap S_2 = \phi$. Then, the learning and estimation processes

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 15/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

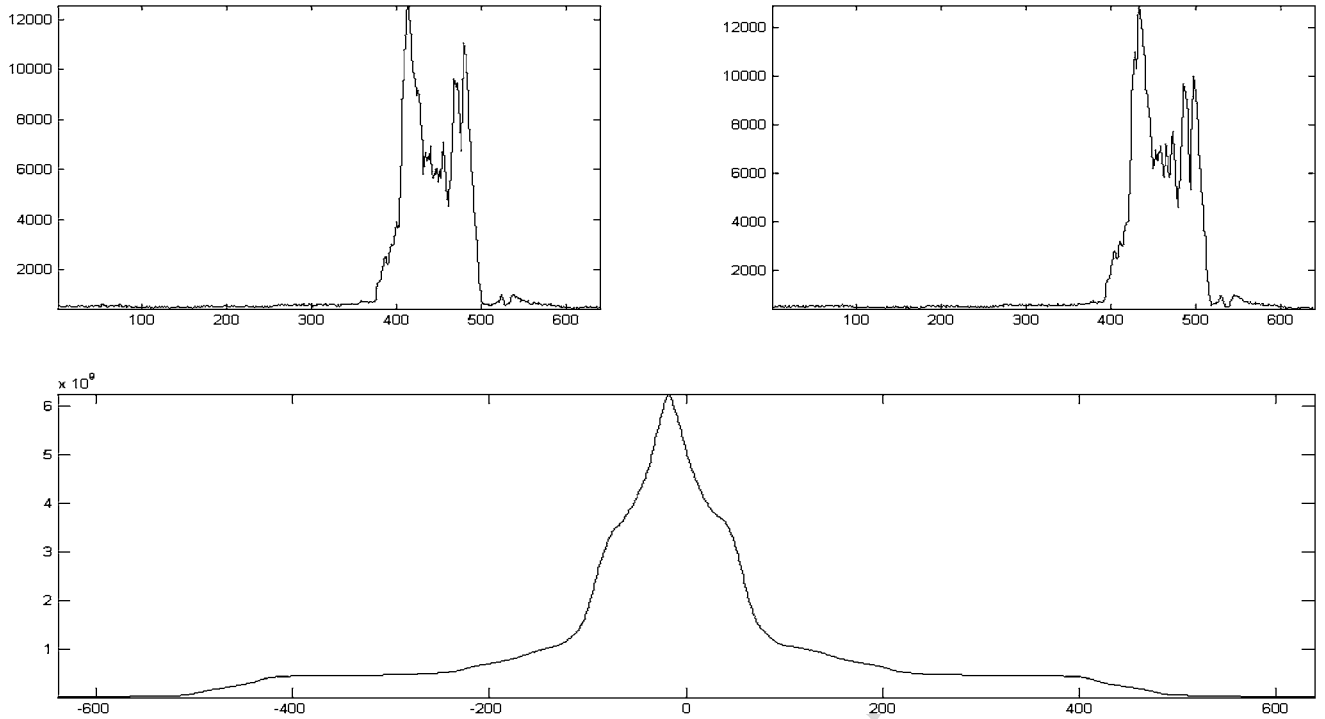

Fig. 8 *Left* and *right* horizontal histograms of difference images at *upper panels*, and Cross-correlation of the stereo horizontal histograms of Fig. 7, at *lower panel*. The position of the maximum is at $d = -21$

Table 1 Number of hits in the classification process together with the average and deviation

	Correctly classified										Average	Deviation
BayesNet	317	327	313	323	316	325	310	324	303	320	317.64	7.17
NaiveBayes	283	291	294	281	291	293	283	295	277	277	286.34	6.68
AdaBoostM1	312	334	315	319	319	320	313	335	311	325	320.10	8.16
Bagging	313	329	312	325	320	330	312	328	311	328	320.62	7.65
DecisionStump	310	316	306	312	309	315	307	311	298	308	309.12	4.83
JRip	314	322	310	326	318	335	318	326	312	339	321.75	9.11
Logistic	302	341	302	341	289	341	302	341	299	341	318.46	21.40
LogitBoost	314	333	317	334	321	333	312	333	314	336	324.43	9.40
MultiBoostAB	310	316	306	313	309	318	307	316	305	309	310.84	4.35
OneR	310	320	301	316	315	321	299	318	303	310	311.11	7.62
Stacking	169	172	165	176	157	184	165	176	163	178	170.15	7.76
CBR	333	335	330	336	333	335	335	333	313	332	331.37	6.39

were carried out. This process was repeated 5 times for each of the techniques, and involved the following steps: the classifier was trained using S_1 and then it was used to classify S_1 and S_2. In the second step, the classifier was trained using S_2 and then it was used to classify S_1 and S_2. The results obtained are shown in Table 5, where the columns represent the classifications obtained for S_1, S_2 (trained with S_1) and S_1, S_2 (trained with S_2) for each of the 5 repetitions. The rows in Table 1 show the different classifiers used during the classification process. The columns represent the number of hits obtained during the learning process. The last two columns represent the mean and the standard deviation. The proposed system presents the highest hit rate against the other methods if we consider the final mean obtained. The deviation in the number of hits is also low so we can conclude that the hit rate is more constant than the rest of methods.

The analysis of the cross validation is completed using the Dietterich's 5 × 2-Cross-Validation Paired t-Test [65]. The significance levels obtained in the test are shown in Table 2. As can be observed, the results are very similar to those previously shown in Table 2. In this case, the only technique that provides results similar to CBR is LogitBoost

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 16/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

Table 2 *t*-test among different statistical techniques

	BayesNet	NaiveBayes	AdaBoostM1	Bagging	DecisionStump	J48	JRip	Logistic	LogitBoost	MultiBoostAB	OneR	Stacking	CBR
BayesNet	1.00	0.00	0.57	0.09	0.00	0.12	0.21	1.00	0.08	0.01	0.02	0.00	0.00
NaiveBayes	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AdaBoostM1	0.57	0.00	1.00	0.71	0.00	0.10	0.63	1.00	0.15	0.00	0.02	0.00	0.00
Bagging	0.09	0.00	0.71	1.00	0.00	0.28	0.91	1.00	0.11	0.00	0.01	0.00	0.00
DecisionStump	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.91	0.00	0.21	0.19	0.00	0.00
JRip	0.21	0.00	0.63	0.91	0.00	0.30	1.00	1.00	0.48	0.00	0.01	0.05	0.00
Logistic	1.00	0.00	1.00	1.00	0.91	1.00	1.00	1.00	1.00	1.00	0.76	1.00	0.59
LogitBoost	0.08	0.00	0.15	0.11	0.00	0.41	0.48	1.00	1.00	0.00	0.00	0.67	0.01
MultiBoostAB	0.01	0.00	0.00	0.00	0.21	0.00	0.00	1.00	0.00	1.00	0.68	0.00	0.00
OneR	0.02	0.00	0.02	0.01	0.19	0.00	0.01	0.76	0.00	0.68	1.00	0.00	0.00
Stacking	0.00	0.00	0.00	0.00	0.00	0.42	0.05	1.00	0.67	0.00	0.00	1.00	0.00
CBR	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.59	0.01	0.00	0.00	0.00	1.00

although the value obtained is lower than the significance level of 0.05.

Therefore the difference is considered to be significant. The logistic method cannot be considered significantly different from the CBR as the value obtained in the test is 0.59. However, noting the mean of both methods and comparing 331 with 318, while we can conclude that there does exist a difference between the two methods, we cannot do so statistically due to the high deviation within the Logistic method.

Figure 9 shows the information obtained in the classification process. As we can see, the method cannot estimate that it is different from any other method because of the high variability. The CBR system is estimated differently for all the methods with the exception of the logistic method. The theoretical significance level is shown by the variable significance.

Once the techniques used in the CBR are statistically verified, we proceed to analyze the evolution of the CBR system with the increase of case memory. To do this, we estimate the efficiency in the classification by studying the evolution of the number of errors from the increased size of the database. In Fig. 10, we can see the evolution of the hit rate with the increase in the base case. The x-axis represents the number of cases and the y-axis represents the hit rate.

The qualitative analysis of the errors is obtained for different training methods by training $N-1$ images and sorting the remaining image. The results show that of 682 test images we detected 17 classification errors, which corresponds to about 2.5% of the cases. Two of the errors correspond to errors forced to verify the system, and thus should not count.

If we analyze the errors we get different results in each type of illumination, based on the results using the CBR-based method. In the case of natural illumination we found a 1.2% error rate, regardless of the conditions of open or closed door., In the case of fluorescent illumination from the ceiling, we have an error rate below 1% which does not depend on the status of the door. But with incandescent illumination on the wall, the error rate soars to 5.3% of the cases.

5 Conclusions and Future Work

The article has focused on the combination of techniques used to perform a detection process, subdivided into: human detection, human tracking and human behavior understanding. The system proposes a novel solution based on the CBR paradigm, including advanced reasoning capacities.

In order to evaluate the system's capacity, a variety of tests were performed. The system improved the processing capability compared to other centralized systems, given that the distributed agents approach makes it possible to process tasks individually and with different techniques (filters, obtaining distance, human detection, human tracking and hu-

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

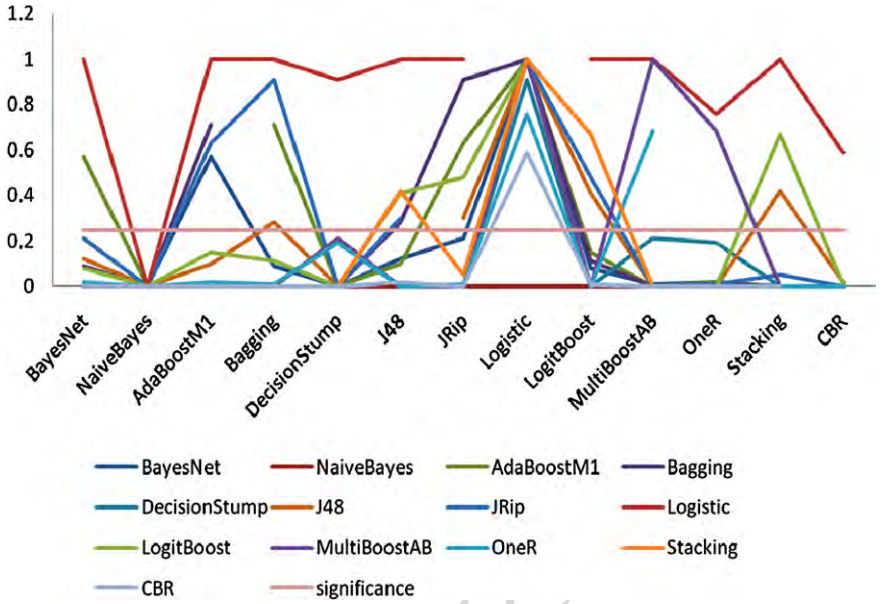

Fig. 9 Significance level obtained using the t-test

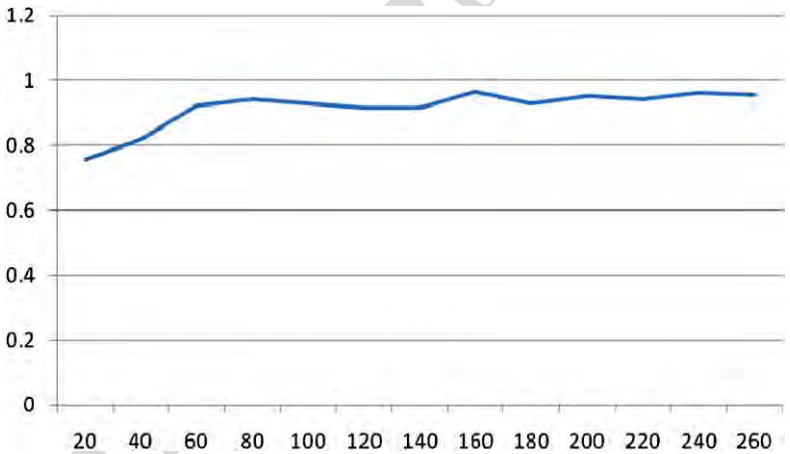

Fig. 10 Evolution of the hit rate with the increase in case memory

man behavior understanding). Because the system is perfectly modularized, the tasks can be carried out simultaneously and/or in a distributed manner. Due to its inherent nature, the implemented human detection techniques are not sensitive to human presence when the person remains static, standing or laying, during extended periods of time. Moreover, due to the camera position we can find that the person being monitored can suffer partial occlusions. Sometimes a person can be in the same position for long periods of time, when lying in bed for example, with a very low level of activity.

In the test sequences we worked with images with a single person in scene, as the main use of the system is to monitor people when they are alone in a room. However, the system could be adapted to detect more than one person in a scene, with the decomposition of the image into sub-images according to the activity detected in different regions.

The classification of the images varies depending on the illumination. With images of low contrast it becomes more difficult to make a proper detection and classification. The illumination issues are, by far, one of the trickiest questions relating to real-time real-world systems. The approach was designed to be independent of the illumination, however we found the worst results with wall lamp illumination, probably due to higher noise introduced by the large amplification gain produced by the low illumination. The best results correspond to scenes with fluorescent illumination or with natural tamed daylight.

Another critical point is the adjustment of the cameras. In our case we have used an automatic gain and an automatic focus adjustment for the stereo-pair camera. These settings work well for getting optimal visualization of the images, regardless of the changes in the scene, or the changes in the lighting conditions as we will have in the real working environment described, but they become hard conditions for the setting parameters of a vision system. Our solution is independent of these adjustments, as it works with differences

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF

« JMIV 10851 layout: Large v.1.3.2 file: jmiv290.tex (petras) class: spr-twocol-v1.2 v.2011/02/19 Prn:2011/04/26; 9:36 p. 18/19»
« doctopic: OriginalPaper numbering style: ContentOnly reference style: mathphys»

of close related time frames. These frames will have very similar settings.

As lines of future work, the proposed system can be enhanced by choosing different human tracking, human detection or classification techniques. Moreover, application of our approach is not restricted to the shown environment and can also be employed in other dynamic environments. The proposed approach can be extended to applications where stereo data are processed, e.g. movies, gaming, mobile robot navigation [56] or industrial applications [3].

Acknowledgements This research has been partially supported by the MICINN project TIN 2009-13839-C03-03 and by the grant *Ajuts per a estades de recerca fora de Catalunya 2009-10* from the University of Vic.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Commun. **7**, 39–59 (1994)
2. Aggarwal, J.K., Cai, Q.: Human motion analysis: a review. Comput. Vis. Image Underst. **73**(3), 428–440 (1999)
3. Aguilar, J.J., Torres, F., Lope, M.A.: Stereo vision for 3D measurement: accuracy analysis, calibration and industrial applications. Measurement **18**(4), 193–200 (1996)
4. Aha, D., Kibler, D.: WInstance-based learning algorithms. Mach. Learn. **6**, 37–66 (1991)
5. Alahi, A., Vandergheynst, P., Bierlaire, M., Kunt, M.: Cascade of descriptors to detect and track objects across any network of cameras. Comput. Vis. Image Underst. **114**(6), 624–640 (2010)
6. Angulo, C., Tellez, R.: Distributed intelligence for smart home appliances. Red Española de Minería de Datos (2004)
7. Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. Comput. Intell. **20**, 693–709 (2004)
8. Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leonel, R., Nardi, D., Oddi, A., Pecora, F., Rasconi, R.: RoboCare: pervasive intelligence for the domestic care of the elderly. Intell. Artif. **1**(1), 16–21 (2004)
9. Bahadori, S., Iocchi, L., Leone, G.R., Nardi, D., Scozzafava, L.: Real-time people localization and tracking through fixed stereo vision. In: Lecture Notes on Artificial Intelligence (LNAI), vol. 3533, pp. 44–54 (2005)
10. Bajo, J., de Paz, J.F., de Paz, Y., Corchado, J.M.: Integrating case-based planning and RPTW neural networks to construct an intelligent environment for health care. Expert Syst. Appl. **36**(3), Part 2, 5844–5858 (2009)
11. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. Int. J. Comput. Vis. **12**(1), 42–77 (1994)
12. Breiman, L., Fried, J.H., Olshen, R.A., Stone, C.J.: Classification and regression trees. Wadsworth International Group (1984)
13. Brenner, W., Wittig, H., Zarnekow, R.: Intelligent Software Agents: Foundations and Applications. Springer, New York (1998)
14. Camarinha-Matos, L., Afsarmanesh, H.: Design of a virtual community infrastructure for elderly care. In: Proc. of 3rd IFIP Working Conference on Infrastructures for Virtual Enterprises: Collaborative Business Ecosystems and Virtual Enterprises, p. 635 (2002)
15. Camarinha-Matos, L.M., Afsarmanesh, H.: A comprehensive modeling framework for collaborative networked organizations. J. Intell. Manuf. **18**(5), 529–542 (2007)
16. Canny: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**(6), 679–698 (1986)
17. Castanedo, F., García, J., Patricio, M.A., Molina, J.M.: Designing a visual sensor network using a multi-agent architecture. In: Advances in Soft Computing, vol. 55, pp. 430–439 (2009). doi:10.1007/978-3-642-00487-2_46
18. CIA: The World FactBook. ISSN 1553-8133 (2010)
19. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the 12th International Conference on Machine Learning. Morgan Kaufmann, San Mateo (1995), pp. 115–123
20. Collins, R.T., Lipton, A.J., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., Wixson, L.: A system for video surveillance and monitoring. Tech. Rep., CMU-RI-TR-00-12. Carnegie Mellon Univ., Pittsburgh, PA (2000)
21. Comaniciu, D., Ramesh, V., Andmeer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**, 564–575 (2003)
22. Corchado, J.M., Laza, R.: Constructing deliberative agents with case-based reasoning technology. Int. J. Intell. Syst. **18**(12), 1227–1241 (2003)
23. Corchado, J.M., Bajo, J., De Paz, Y., Tapia, D.I.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems. Elsevier, Amsterdam (2006)
24. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intell. Syst. **23**(2), 19–25 (2008)
25. Corchado, J.M., Glez-Bedia, M., de Paz, Y., Bajo, J., de Paz, J.F.: Concept, formulation and mechanism for agent replanification: MRP architecture. In: Computational Intelligence. Blackwell Publishers, Malden (2008)
26. Corchado, J.M., Glez-Bedia, M., de Paz, Y., Bajo, J., de Paz, J.F.: Replanning mechanism for deliberative agents in dynamic changing environments. Comput. Intell. **24**(2), 77–107 (2008)
27. Corchado, J.M., Bajo, J., Tapia, D.I., Abraham, A.: Using heterogeneous wireless sensor networks in a telemonitoring system for healthcare. IEEE Trans. Inf. Technol. Biomed.. **14**(2), 234–240 (2009). Special Issue: Affective and Pervasive Computing for Healthcare. doi:10.1109/TITB.2009.2034369. ISSN:1349-4198
28. Cucchiara, R., Grana, C., Piccardi, M.: Prati, A. Detecting moving objects, ghosts, and shadows in video streams. IEEE Trans. Pattern Anal. Mach. Intell. **25**(10), 1337–1342 (2003)
29. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference Computer Vision and Pattern Recognition, USA, pp. 886–893 (2005)
30. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: European Conference on Computer Vision (ECCV), vol. 2, pp. 428–441 (2006)
31. Davies, E.R.: Lateral histograms for efficient object location: speed versus ambiguity. Pattern Recognit. Lett. **6**(3), 189–198 (1987)
32. Dhond, U.R., Aggarwal, J.K.: Structure from stereo—a review. IEEE Trans. Syst. Man Cybern. **19**(6) (1989)
33. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation 1895–1923 (1998)
34. Duda, R.O., Hart, P.: Pattern Classification and Scene Analysis. Wiley, New York (1973)
35. Eddy, S.R.: Hidden Markov models. Curr. Opin. Struct. Biol. **6**(3), 361–365 (1996)
36. Elgammal, H.M., Harwood, D., Davis, L.S.: Non-parametric model for background subtraction. In: Proc. of the 6th European Conference on Computer (ECCV). Springer, London (2000), pp. 751–767
37. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Thirteenth International Conference on Machine Learning (1996), pp. 148–156

Journal ID: 10851, Article ID: 290, Date: 2011-04-26, Proof No: 1, UNCORRECTED PROOF