People Detection and Tracking using Stereo Vision and Color

DECSAI-SI-2005-04

Authors:

Rafael Muñoz-Salinas

Eugenio Aguirre

Miguel García-Silvente

1.1 Abstract

People detection and tracking are important capabilities for applications that desire to achieve a natural human-machine interaction. Although the topic has been extensively explored using a single camera, the availability and low price of new commercial stereo cameras makes them an attractive sensor to develop more sophisticated applications that take advantage of depth information. This work presents a system able to visually detect and track multiple persons using a stereo camera placed at an under-head position. This camera position is especially appropriated for human-machine applications that requires interacting with people or to analyze human facial gestures. In an initial phase, the environment (background objects) is modelled using a height map that will later be used to easily extract the foreground objects and to search people among them using a face detector. Once a person has been spotted, the system is capable of tracking him while is still looking for more people. Tracking is performed using the Kalman filter to estimate the next position of each person in the environment. Nonetheless, when two or more people become close to each other, information about the color of their clothes is also employed to track them more efficiently. The system has been extensively tested and the results show that the use of color greatly reduces the errors of the tracking system. Besides, the low computing time required for the detection and tracking process makes it suitable to be employed in real time applications.

1.2 Introduction

The topic *human-machine interaction* (HMI) has drawn a lot of attention in the last decade. The objective is to create intelligent systems capable of extracting information about the context or about the actions to perform through a natural interaction with the user, for example, through their gestures or voice. One fundamental aspect in this sense is people detection and tracking, existing a extensive literature about the topic (15; 26; 36; 40). People detection techniques are frequently based on the integration of different information sources such as: skin color, face detectors, motion analysis, etc.

Although people detection and tracking with a single camera is a well explored topic, the use of stereo technology for this purpose concentrates now an important interest. The availability of commercial hardware to solve the low-level problems of stereo processing, as well as the lower prices for these types of devices, turns them into an appealing sensor to develop intelligent systems. Stereo vision provides a type of information that brings several advantages when developing human-machine applications. On one hand, the information regarding disparities becomes more invariable to illumination

changes than the provided by a single camera. That is a very advantageous factor for the development of background estimation techniques (7; 10; 19). Furthermore, the possibility to know the distance from the camera to the person could be of great assistance for tracking as well as for a better analysis of his gestures.

In this work a system able to detect and track multiple persons with a stereo camera placed at an under-head position is presented. In a first phase, the system creates a background model of the environment using a height map. It can be constructed even in the presence of moving objects in the scene (like people passing by). Using this structural map, foreground objects are easily detected and those that are potential candidates to people are analyzed using a face detector. Once an object is identified as a person, the system keeps track of him as well as of the rest of detected people while still looking for more people. In order to achieve a robust tracking, the Kalman filter has been employed to estimate the position of each person in the next image captured. Nonetheless, when people become close to each other, the estimation of the position becomes unreliable. Thus, information about the color of the person is combined with the predicted position to achieve a more robust tracking.

The remainder of this paper is structured as follows. Section 1.3 explains some of the more relevant related works and exposes the main differences with of our approach. Section 1.4 explains the basis of the background modelling and foreground extraction techniques. In Sect. 1.5 it is shown how people detection and tracking are performed. Section 1.6 presents the experimentation carried out and Section 1.7 draws the conclusions and possible future work.

1.3 Related Works

Among the first projects related to people detection and tracking using stereo vision we find the one by Darrel et al (8). They present an interactive display system capable of detecting and tracking several people. Person detection is based on the integration of the information provided by three modules: a skin detector, a face detector and the disparity map provided by a stereo camera. First, independent objects (*blobs*) detected in the disparity map are treated as candidate to people. Then, the color of the image is analyzed to identify those areas that could be related to skin. Finally, a face detector is applied on selected regions of the camera image. These three items are merged in order to detect and track several people. However, two main drawbacks can be pointed out on their approach. First, the system requires the face of the users to be visible in order to appropriately track them. Second, as the system relies on a predefined color model to detect skin, a degradation on the tracking performance can be expected when the

illumination conditions differ significantly from the training ones (28). A dynamic skin color model (35) could have solved this problem.

Grest and Koch (16) developed a system able to detect and track a single-person using stereo vision. It allows the user to navigate in a virtual environment by just walking through a room using virtual reality glasses. The user is detected using the face detector proposed by Viola and Jones (39). Once the user is located, both a color histogram of the face region and a color histogram of the chest region are created and employed by a particle filter to estimate his position in the next image. Then, stereo information assists in determining the real position of the person into the room that is employed to calculate the corresponding position in the virtual environment. In their work, the stereo processing is performed using the information gathered by different cameras located at different positions of the room. As in the previous case, the main limitation is that the system requires the face of the user to be visible in the image to perform the tracking.

A very interesting method to locate and track multiple persons in stereo images using occupancy maps is presented by Harville (20). Before the detection process takes place, a model of the environment is created through a sophisticated image analysis method (19). Once the background image is created, objects that do not belong to it are easily isolated. Then, both an occupancy map and a height map are created. The information from both maps is merged to detect people through the use of simple heuristics. Person tracking is performed using the Kalman filter combined with deformable templates. The stereoscopic system used in his work is located three meters above the ground in a fixed slanting position. The main draw back of his approach is that the simple detection heuristics employed may lead the system to incorrectly detect as people new objects in the scene whose dimensions are similar to human beings (as indicated by the author). That is the case of coat placed on a hanger or a big box got into the room.

Hayashi et al (21) present a people detection and tracking system especially designed for video surveillance. The environment is modelled using an occupancy map an the stereo points are projected as *"variable voxels"* to deal with the stereo errors. People detection is performed in the occupancy map based on a simple heuristic that assumes that a person is a peak in the map whose height is into a normal range. As in the previous case, there can be expected many false positives because of the simplicity of the detection scheme.

Tanawongsuwan presents in (37) the initial steps for designing a robotic agent able to follow a person and recognizing his gestures. His system employs a basic technique to find the arms and head of a person combining the information provided by a skin color filter, a movement detector and depth. Once a person has been located, Hidden Markov Models are used to recognize his gestures among a set of previously learned ones. Nevertheless, he does not deal in depth with the detection problem and assumes that there

is a person if three skin colored blobs (corresponding to head and hands) are found.

1.3.1 Our Approach

A system able to detect and track multiple persons is presented in this work. It is specially designed for situations in which the camera must be placed at an under-head position. The position of the camera is a problem-dependent issue related with the purpose of the system. Several authors have mounted their cameras in the ceiling to perform people detection in Ambient Intelligence domains (18; 36). Others have used slating cameras in overhead positions (4; 20; 21) to achieve the same functionality. Nevertheless, in most of the works that seek interacting with people, the position of the camera is usually lower than them (8; 30; 37). This approach is mainly supported by two facts. On one hand, this camera configuration allows to see the faces and arms of the people and thus, be able to analyze their facial expressions and gestures. On the other hand, studies in the human-robot field reveals that people tends to feel threaten by big robots (13). However, the main drawback of low camera positions is that occlusions between people are more frequent that in elevated ones.

Most of people detection and tracking systems have an initial phase where a background model is created. It is employed to easily detect the moving objects in the environment (foreground). Techniques usually employed for that purpose consist in creating a background image (pixel by pixel) using several images of the scene, if possible, without motion elements in them (15; 20; 26; 36). The simplest approach is to use the average of the sequence in each pixel as the background value. Others authors have used the median value and even Kalman filters to perform the update process. In this paper, a height map of the environment (built using stereo information) is employed as background model. This technique has been previously employed by Darrell et al (7) to estimate the trajectory of moving people in a room. The background model is created in their work using several stereo cameras placed at different locations of the room. Height maps bring several advantages over traditional techniques to create background models. Firstly, because of stereo information is used instead of intensity values, the background model created is more invariant to sudden illumination changes. Secondly, the background model can be simultaneously created and employed by different devices placed at different locations. In fact, height maps have been widely used in mobile robotics to describe the environment and plan trajectories on them (5; 9; 34; 38). Therefore, they seem to be specially appropriated for mobile devices like autonomous robots (2; 29) or mobile stereoscopic systems (4). Once the height map of the environment is created, the foreground is modelled as an occupancy map that registers the position of the moving objects in the environment.

There can be found mainly two approaches for people detection when using stereo vision in the related literature. The first one is considering a person as an object in an occupancy map with sufficient weight (18; 20; 21). This approach is commonly used when the camera is placed at elevated positions. As we have previously commented, the main problem of that approach is that objects with dimensions similar to human beings that enters in the scene can be incorrectly detected as people. The second approach consists in looking for faces in the camera image (8; 16; 30). This approach seems to be more appropriate when low camera positions are employed. However, if no additional information is employed, this approach is sensible to the false positives of the face detector. The system proposed in this work combines these two approaches to avoid the drawbacks of each one them. An object detected in the foreground occupancy map is considered as person if it has appropriate dimensions (human being dimensions) and if it is detected a face on it. To speed up computation, the face detector is only applied on selected regions of the image where it seems possible to find faces.

When a foreground object is identified as a person, the system starts to track him in the occupancy map. While tracking him, the system is able to keep tracking the rest of people previously detected and is still looking for new people. To track each person, the system combines information about his position (predicted using the Kalman filter (17)) with information about the color of his clothes. Both pieces of information are dynamically combined in the following way. If a person is far from others, relying on the prediction of his position is safe. However, if the person is close to others, the prediction about his future position is not reliable because he could change his trajectory to avoid a collision or to interact with other people. In that case, information about the color of his clothes is employed to enhance the tracking process. It is important to remark that the face detector is only employed to detect people. However, once a person is detected, the tracking process does not employ the face detector. Thus, the person does not need to look at the camera to be tracked.

1.4 Environment Map Building

In this section, the basis of the stereo processing, background modelling and foreground extraction are presented. The section is structured in three parts. Subsection 1.4.1 explains the basis of stereo calculation and how the 3D points captured by the stereo camera are translated to another reference system more appropriate for our purposes. Then, Subsection 1.4.2 explains the technique employed to create the background height map using the translated 3D points. Finally, Subsect. 1.4.3 explains how the height map is used to extract the foreground objects.

1.4.1 Stereo Processing

A commercial stereo camera (32) has been employed in this work. It can capture two images from slightly different positions (stereo pair) that are transferred to the computer to calculate a *disparity image* I_d 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 p_{cam} of a pixel (u, v) in I_d. Let us denote by $P_{cam} = \{p^0_{cam}, \ldots, p^{np-1}_{cam} |\ p^i_{cam} = (X^i_{cam}, Y^i_{cam}, Z^i_{cam})^T\}$ the set of three dimensional points captured by the camera that are calculated using Eq. 1.1. Where, f is the focal length of the cameras, b is the baseline distance between the cameras and d the disparity value of the pixel (u, v) in the image I_d.

$$\begin{aligned} Z_{cam} &= \frac{fb}{d} \\ X_{cam} &= \frac{uZ_{cam}}{f} \\ Y_{cam} &= \frac{vZ_{cam}}{f} \end{aligned} \tag{1.1}$$

The three-dimensional positions p^i_{cam} calculated by Eq. 1.1 (affected by typical stereo errors (1; 33)) are referred to the stereo camera reference system. In our case it is centered at the right camera. However, this reference system may be changed from one application to another, i.e., the stereo camera can be placed at different positions and with different orientations. Hence, it is preferable for our purposes to translate the position of the points captured to a *"world"* reference system placed at ground level and parallel to it. Knowing the position and orientation of the camera in relation to the floor plane, it is possible to calculate the linear transformation matrix T that translates the points p^i_{cam} into $p^i_w = (X^i_w, Y^i_w, Z^i_w)^T$ using Eq. 1.2. For more information about three-dimensional transformations the interested reader is referred to (11).

$$p_w = T p_{cam}. \tag{1.2}$$

Figure 1.1(a) shows an example of an scene captured with our stereo camera (the image corresponds to the right camera). Figure 1.1(b) shows the three-dimensional reconstruction of the scene captured using the points detected by the stereo camera. The *"world"* and camera reference systems have been superimposed in the Figure 1.1(b).

As it can be seen in Figure 1.1(b), the number of points acquired by an stereo camera can be very high (they are usually referred to as point cloud). For that reason, many authors perform a reduction of the amount of information by orthogonally projecting them into a 2D plan-view map (18; 20; 21). This decision is also supported by the fact that people do not tend to be overlapped in the floor plane as much as they are in the original captured images. Therefore, the detection and tracking process is more reliable in the 2D projection.

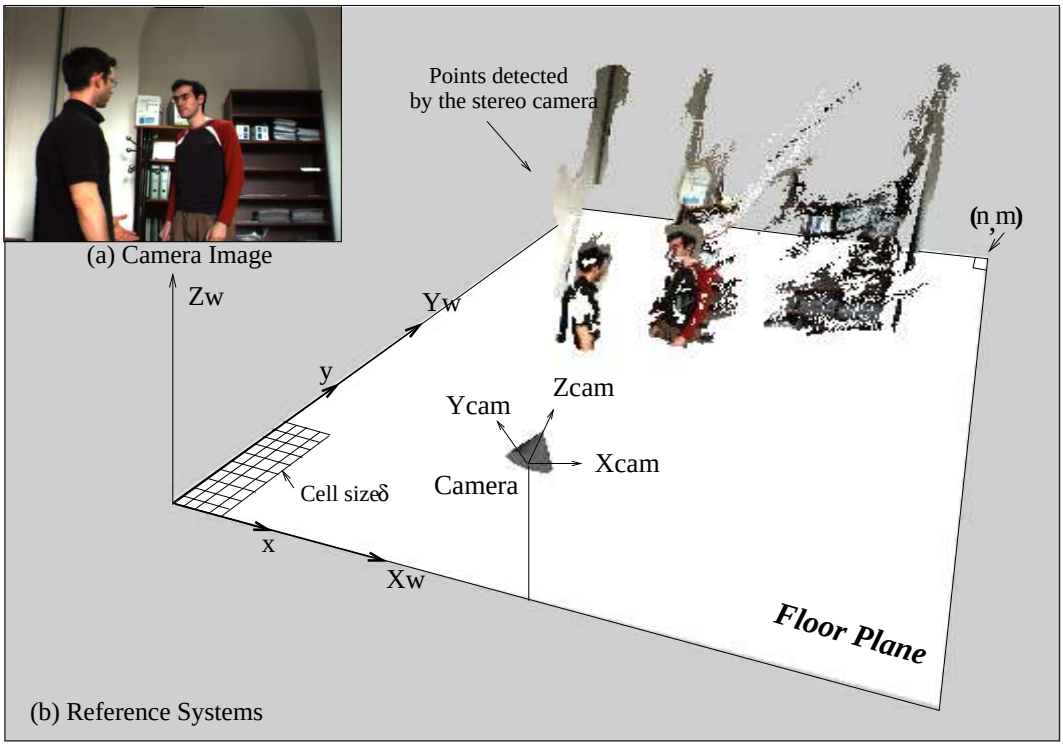


Figure 1.1: (a) Image of the right camera captured with the stereo system. (b) Three-dimensional reconstruction of the scene showing the reference systems employed.

A plan-view map divides a region of the floor plane into a set of nxm cells of fixed size δ. In this work, the cell $(x, y) = (0, 0)$ coindices with the *"world"* positions $(0, 0, Z_w)$ (see Fig. 1.1(b)). Hence, the cell (x^i, y^i) in which a three-dimensional point p_w^i is projected can be calculated as:

$$x^i = (X_w^i/\delta) \ ; \ y^i = (Y_w^i/\delta) \tag{1.3}$$

Every time a stereo pair is captured, the set of points that are projected on each cell is calculated as:

$$P_{(x,y)} = \{i \mid x^i = x \wedge y^i = y \wedge Z_w^i \in [h_{min}, h_{max}]\}.$$

Where $[h_{min}, h_{max}]$ is a height range that has two purposes. On one hand, the superior limit h_{max} avoids using points from the ceiling or from objects hanging from it (i.e. lamps). On the other hand, the inferior limit h_{min} excludes from the process low points that could not be relevant for a particular application (floor points or even the legs of people) and thus helps to reduce the computing time. The height range $[h_{min}, h_{max}]$ should be such that, at least, the head and shoulders of the people to detect should fit in

it. The rest of points p_w^i whose projection is outside the limits of the plan view map are not considered.

The selection of δ must be made taking into account several aspects. A high value helps to decrease the computational effort and the memory used. The side effect is that it causes a loss of precision in the estimation of the position. On the other hand, a low value increases the precision (up to the limit imposed by the errors of the stereo computation (1; 33)) but also the computational requirements. Ismail et al. in (18) propose the use of $\delta = 0.2$ cm while Harville in (20) uses $\delta \in [2, 4]$ cm. We have selected a value of $\delta = 1$ cm which according to our experimentation is an adequate balance between both requirements (precision and memory).

1.4.2 Background Modelling

Because people can be considered movable elements in the environment, it is very helpful to separate the points that belong to the environment (background) from those that do not (foreground). Our approach for background modelling differs from others in that it is based on the creation of a geometrical height map of the environment $\hat{\mathcal{H}}$ (7), instead of directly modelling the intensity values from the camera image. $\hat{\mathcal{H}}$ is a plan-view map that indicates in each cell $\hat{\mathcal{H}}_{(x,y)}$ the maximum height of the points projected in it. We might think of $\hat{\mathcal{H}}$ as a representation of the surface of the environment over which foreground objects move. To avoid including as part of the background objects momentarily passing by, $\hat{\mathcal{H}}$ is created aggregating several instantaneous height maps $\mathcal{H}^t$ with a robust estimator as the median:

$$\hat{\mathcal{H}}_{(x,y)} = median(\mathcal{H}_{(x,y)}^{t=t_0}, \ldots, \mathcal{H}_{(x,y)}^{t=t_0+\Delta t}). \tag{1.4}$$

Each cell of An instantaneous $\mathcal{H}^t$ is calculated as:

$$\mathcal{H}_{(x,y)}^t = \begin{cases} max(Z_w^j \mid j \in P_{(x,y)}) & if\ P_{(x,y)} \neq \emptyset \\ h_{min} & if\ P_{(x,y)} = \emptyset \end{cases} \tag{1.5}$$

Figure 1.2 shows the evolution of the height map $\hat{\mathcal{H}}$ from a set of 13 instantaneous height maps $\mathcal{H}^t$ captured at time intervals of $\Delta t = 400$ ms. Due to space reasons, the Fig. 1.2 shows only the status of the map at the time instants $t = \{0, 1600, 4000, 5200\}$ ms. Figures 1.2(a-d) (upper row) show the images captured by the right camera. Figures 1.2(e-h) (middle row) are the corresponding instantaneous height maps $\mathcal{H}^t$. Dark areas represent the highest zones and white areas represent the lowest ones h_{min}. Finally, Figs. 1.2(i-l) (lower row) show the evolution of the height map $\hat{\mathcal{H}}$ as more instantaneous height maps are employed to calculate it. Notice that $\hat{\mathcal{H}}$ has been created in the presence of people moving in the environment (their positions has been marked in the instantaneous height maps). As it can

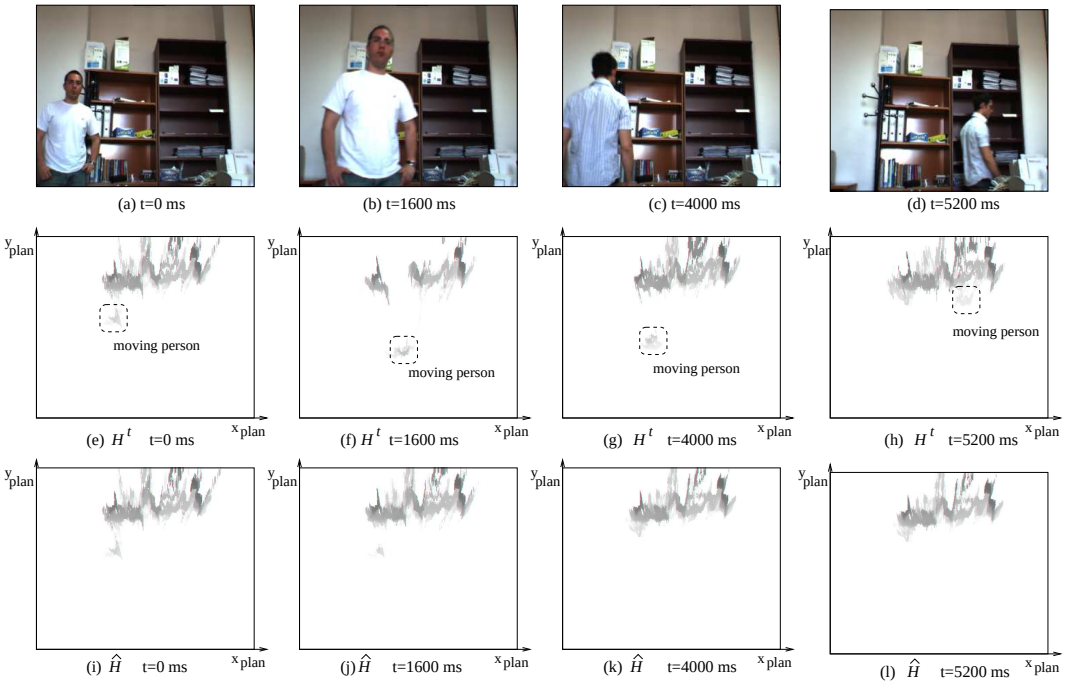


Figure 1.2: Creation of the height map. Upper row (a,b,c,d) shows the images in instants $\{0, 1600, 4000, 5200\}$ ms. Central row (e,f,g,h) shows the instantaneous height maps $\mathcal{H}^t$ for each one of the upper images. Lower row (i,j,k,l) shows the evolution of the height map $\hat{\mathcal{H}}$ created as the median of the height maps $\mathcal{H}^t$ created until that moment

be seen, at the beginning $\mathcal{H}^{t=0} = \hat{\mathcal{H}}$ and thus the moving person appears in the height map. But as the time goes and more instantaneous height maps are employed to create $\hat{\mathcal{H}}$, it tends to truly represent the motionless characteristics of the environment. To create these maps we have used $h_{min} = 0.5$ m and $h_{max} = 2.1$ m.

$\hat{\mathcal{H}}$ can be periodically updated in order to dynamically be adapted to the changes in the environment. Nonetheless, to avoid old data influence in the update process and to save memory, the number of instantaneous height maps $\mathcal{H}^t$ employed to create $\hat{\mathcal{H}}$ must be limited to the more recent ones. According to our experimentation, $\hat{\mathcal{H}}$ can be appropriately updated using the last 10 instantaneous height maps. The frequency employed to update the height map should be smaller than the employed to detect and track people. In this way, it is possible to avoid including as part of the background people momentarily standing by or other moving objects. The update frequency employed in our experiments has been set to 0.1 Hz.

1.4.3 Foreground Extraction

The foreground points present in each captured stereo pair can be easily isolated using the height map $\hat{\mathcal{H}}$. To model the foreground we have employed a plan view-map $\mathcal{O}$ that is called *occupancy map*. $\mathcal{O}$ registers in each cell $\mathcal{O}_{(x,y)}$ the amount of points belonging to the foreground that are projected in it. Lets denote by

$$F_{(x,y)} = \{i \mid i \in P_{(x,y)} \wedge Z_w^i > \hat{\mathcal{H}}_{(x,y)}\},$$

to the set of points detected in a stereo pair, projected on the cell (x, y) and whose height is above the height indicated in $\hat{\mathcal{H}}_{(x,y)}$, i.e., the foreground points detected over the background surface that represents $\hat{\mathcal{H}}$. Each cell of the occupancy map is calculated as:

$$O_{(x,y)} = \sum_{j \in F_{(x,y)}} \frac{(Z_{cam}^j)^2}{f^2} \tag{1.6}$$

The idea is that each foreground point increments the cell in with it is projected by a value proportional to the surface that it occupies in the real scene (20). Points closer to the camera correspond to small surfaces and vice versa. If the same increment is employed for every cell, the same object would have a lower sum of the areas the farther it is located from the camera. This scale in the increment value will compensate the difference in size of the objects observed according to their distance to the camera. Figure 1.3(b) shows the occupancy map $\mathcal{O}$ of the scene in the Figure 1.3(a) using the height map $\hat{\mathcal{H}}$ from Fig. 1.2(l). The darker values represent the areas with higher occupancy density. The image has been manually retouched to make the occupied areas visible. As it can be seen, there are small dark dots in the upper area of Fig. 1.3(b) that are caused by errors of the stereo correlation process. However, the person that stands in the scene is clearly projected in the occupancy map as a connected group of cell with high occupancy level.

The next step in our processing, is to identify the different objects present in $\mathcal{O}$ that could correspond to human beings. For that purpose, $\mathcal{O}$ is processed with a closing operator in order to link possible discontinuities in the objects caused by the errors in the stereo calculation. Then, objects are detected as groups of connected cells. Those objects whose area is similar to the area of a human being and whose sum of cells (occupancy level of the object) is above a threshold θ_{occ} are employed in the next phase for people detection and tracking. This test is performed in a flexible way so that it is possible to deal with the stereo errors and partial occlusions. Figure 1.3(c) shows the unique object detected in the occupancy map of Fig. 1.3(b) after the above mentioned process.

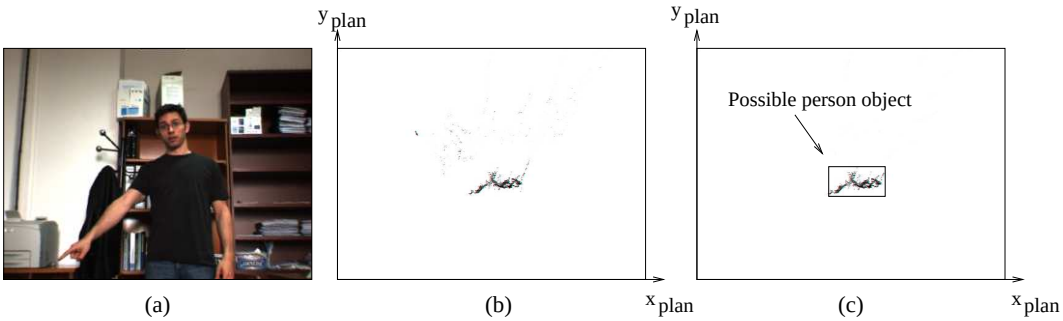


Figure 1.3: (a) Right image of the pair in an instant, the environment with an object not in background. (b) Occupancy map $\mathcal{O}$ corresponding to the environment. (c) Framed information corresponding to the object, detected using $\mathcal{O}$

1.5 People Detection and Tracking

People detection and tracking are performed as separate processes. Every time a new scene is captured, the system must decide first, which one of the objects detected in $\mathcal{O}$ corresponds to each one of the people that are being tracked (an assignment problem). Then, the system applies a face detector on the remaining objects in order to detect new people. Notice that the face detector is only applied on those objects that have not been detected as people yet. That approach allows to manage false negatives detections of the face detector, i.e., if the face detector fails in detecting a person once, he could be detected in the next image.

Tracking people in that work consist in: (i) predicting their future positions according to its past movement using the Kalman filter (17) and, (ii) solving the assignment problem using the predicted positions as well as information about the color of the objects. Therefore, a color model of each object detected in $\mathcal{O}$ is created to be compared with the color model of each person. The color model of a person is he one created when he was first detected that is updated each time the person is tracked. Both sources of information (position and color) are combined dynamically to achieve a robust assignment in the following way. When a person is far from others, the system gives more importance to the prediction about its position. However, when a person is near others the system uses also information about the color of his clothes to enhance the tracking.

In the next subsections these processes are explained in detail. Subsection 1.5.1 explains how the color model of each object is created. Later, in Subsect. 1.5.2 it is shown how people detection is performed. And finally, Subsect. 1.5.3 explains how these pieces of information are fused to perform the tracking.

1.5.1 Color Modelling

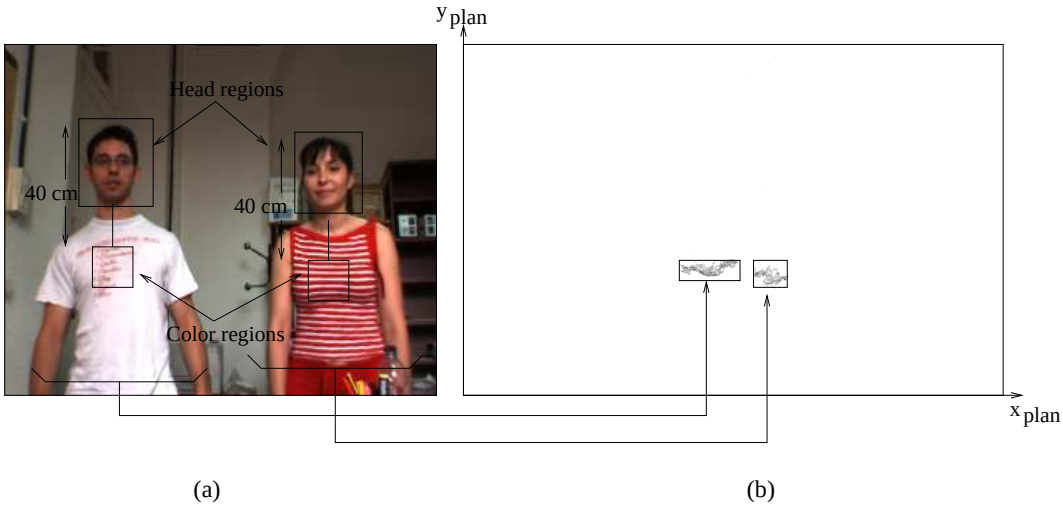


Figure 1.4: (a) Scene with two objects in it. The bottom black boxes are the regions used for creating their corresponding color models (b) Occupancy map $\mathcal{O}$ of the scene

A color model $\hat{q}$ of each object is created to assist the tracking process. The color model aims to capture information about the color of the clothes of the people in the scene. Thus, the pixels around what it should be the chest of a person are employed. For that purpose, the position of the chest p_{chest} in the camera image is estimated as 40 cm below the top of the head region. The head region is roughly estimated using the center of mass of the object in $\mathcal{O}$ and the disparity image I_d. The color model $\hat{q}$ is created using the pixels in a region around p_{chest} whose size varies accordingly to the distance of the object to the camera. When the object is far from the camera, the region employed is smaller to avoid including pixels from the background and it becomes bigger when the object is near to the camera.

Figure 1.4(a) shows an image where there are two objects in the environment. The regions employed to create the color models are depicted as boxes in the Fig. 1.4(a). Figure 1.4(b) shows the occupancy map of the scene in Fig. 1.4(a).

The color model $\hat{q}$ of each object is modelled as an histogram using technique described by Comaniciu et al (6). The HSV space (12) has been selected to represent color information. The histogram $\hat{q}$ is comprised by $n_h n_s$ bins for the hue and saturation. However, as chromatic information is not reliable when the value component is too small or too big, pixels on this situation are not used to describe the chromaticity. Because these "color-free" pixels might have important information, the histogram is also populated with n_v bins to capture its illuminance information. The resulting

histogram is composed by $m = n_h n_s + n_v$ bins.

Let $\{x_i^*\}_{i=1...n}$ be the locations of the pixels employed to create the color model. We define a function $b : \Re^2 \rightarrow \{1...m\}$ which associates to the pixel at location x_i^* the index $b(x_i^*)$ of the histogram bin corresponding to the color of that pixel. The color density distribution for each bin $\hat{q}_u$ of the region x^* is calculated in the following way:

$$\hat{q}_u = K \sum_{i=1}^{n} w(x_i^*) \kappa[b(x_i^*) - u]. \tag{1.7}$$

The weighting function w gives more relevance to pixels near the central point of the region x^* (p_{chest}) and thus reduces the influence of background pixels that might be incorrectly included. Function κ represents the Kronecker delta function. Finally, K is a normalization constant calculated by imposing the condition $\sum_{u=1}^{m} \hat{q}_u = 1.$, from where

$$K = \frac{1}{\sum_{i=1}^{n} w(x_i^*)} \tag{1.8}$$

since the summation of the Kronecker delta functions is equal to 1.

Once the color model $\hat{q}$ of an object is created, it can be compared with other color model $\hat{p}$ using the Bhattacharyya coefficient (3; 23). In the case of the continuous distributions it is defined as:

$$\rho(q, p) = \int \sqrt{q(u)p(u)} du, \tag{1.9}$$

and for the discrete distribution of our color models it can be expressed as:

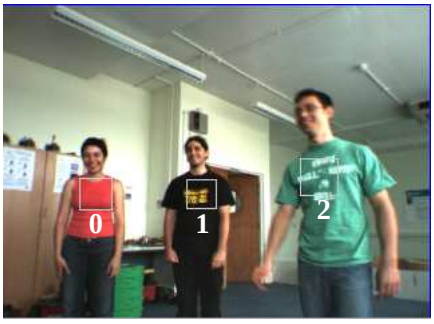
$$\rho(\hat{q}, \hat{p}) = \sum_{u=1}^{m} \sqrt{\hat{q}_u \hat{p}_u}. \tag{1.10}$$

The value $\rho(\hat{q}, \hat{p})$ gives a measure of the similarity of two color models in the range $[0, 1]$ where 1 means that both color models are identical and it decreases as the they differ. Figure 1.5(a) shows three people (objects) in the scene. The boxes on each person indicates the regions employed to create their color models. Figure 1.5(b) shows as a table the values $\rho(\hat{q}, \hat{p})$ for each pair of color models using a total of $m = 30$ bins.

1.5.2 People Detection

Our approach for people detection consists in analyzing if an object detected in $\mathcal{O}$ shows a face in the camera image. As previously indicated, the detection process is performed only on the remaining objects after tracking of known people has been completed.

Face detection is a process that can be time consuming if applied on the entire image, thus, it is only applied these on regions of the camera image



(a)

		0	1	2
0		1	0.339	0.359
1		0.339	1	0.094
2		0.359	0.094	1

(b)

Figure 1.5: (a) Scene with three objects in it. (b)$\rho(\hat{q}, \hat{p})$ of the color models of the objects

where the head of each object should be (head region). As the human head has an average width and height, the system analyzes first if the head region of an object has similar dimensions. If the object does not pass this test, the face detector is not applied on it. This test is performed in a flexible manner so that it can handle stereo errors and people with different morphological characteristics can pass it. If the object passes the test, the corresponding region in the image analyzed to detect if it contains a face. This reduction the search region where to apply the face detector brings two main advantages. First, it reduces the computational time as smaller regions are analyzed. Second, it reduces the number of false positives as stated in (24).

The face detector provided by the OpenCv's Library (22) has been selected to detect if there is any face in the head region of the objects. It is not in the scope of this paper to develop face detection techniques since there is plenty literature about it (41). The face detector employed is based on the face detector of Viola and Jones (39) which was later improved by Lienhart (27). The implementation is trained to detect both frontal and lateral views of human faces and works on gray level images.

Figure 1.6(a) shows a scene where there is a person that has entered and has hanged his coat. Figure 1.6(d) shows the occupancy map $\mathcal{O}$ of that scene. As it can be noticed, two objects are detected, the person and the coat. Using the procedure explained before, it is detected that the size of the upper part of the two objects are similar to human's heads. Hence, the regions of the image that should contain their faces (Figs. 1.6(b) and 1.6(c)) are processed by the face detector indicating that there is a face only in Fig. 1.6(b).

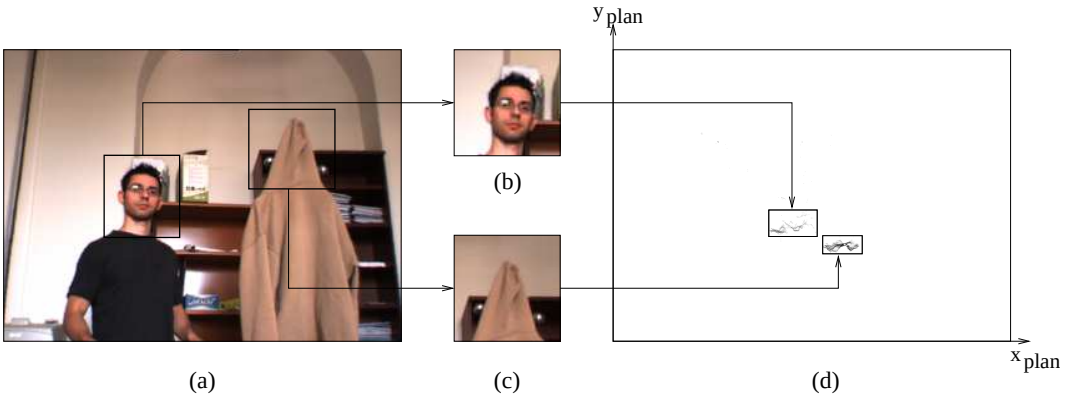


Figure 1.6: Example of people detection (a) Image captured by the camera (b-c) Images of the upper part of the objects detected. (d) Occupancy map of the scene.

1.5.3 People Tracking

Once an object has been identified as a person by the procedure explained before, it is necessary to keep track of him in the following images. The tracking problem can be seen as an assignment problem, i.e., relate a person that is being tracked with an object currently detected in $\mathcal{O}$. The problem has been solved using the Kuhn's well-known Hungarian Method for solving optimal assignment problems (25).

Let us denote by $\Upsilon = (\Phi, \Pi)$ the set of n objects detected in $\mathcal{O}$. The set $\Phi = \{obj^1, ..., obj^n\}$; $obj^i = (x^i_{obj}, y^i_{obj})$ denotes the location of their center of masses in the plan-view map, and $\Pi = \{\hat{p}^1, ..., \hat{p}^n\}$ their corresponding color models, being $\hat{p}^i$ the color model of the object obj^i.

Let also denote by $\Psi = (\varphi, \Omega)$ the set of m people detected and being tracked. The set $\varphi = (s^1 ... s^m); s^j = (x^j_p, y^j_p, v^j_x, v^j_y)$, denotes their positions and velocities, and $\Omega = (\hat{q}^1, ..., \hat{q}^m)$, their color models created when their faces were detected.

The assignment problem consists in determining the optimal assignment of currently detected objects Υ, to the people being tracked Ψ. In order to use the Hungarian method, it is necessary to calculate the probability value (or cost) of assigning the object obj^i to the person s^j. In this work, this probability value is calculated accordingly to two features. The first one is the difference between the position of the object and the predicted position for the person. The second one is the similitude between the color models of the object and the person by Eq. 1.10.

Kalman filter is employed to predict the new position $s^j_{pred} = (x^j_p, y^j_p)$ of each person in the plan-view maps using a linear model of his movement:

$$x(t+1) = x(t) + v_x t; \; y(t+1) = y(t) + v_y t,$$

where v_x and v_y are the velocities of the person in the plan-view map. Although it is a basic movement model, it is able to successfully predict the position of people when images are captured at short time intervals. Figure 1.7 shows the tracking results of a real sequence of 9 seconds captured at 7 Hz where a person walks 6.93 meters at steady speed. The solid line represents the observed path of a person moving in the environment while the dashed line represents the predictions of the Kalman filter. As it can be observed, the prediction model is able to estimate the trajectory of the person with a low error rate. Figure 1.8 shows the estimation errors. Notice that the maximum error does not exceed 15 cm, and it occurs when the person is turning.

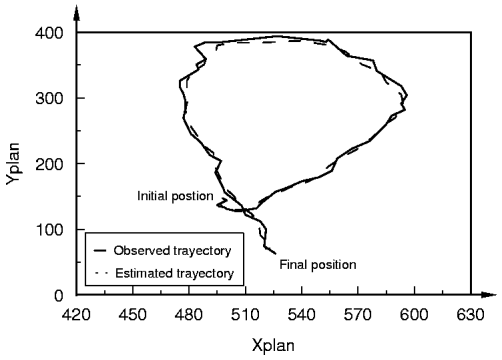


Figure 1.7: Example of trajectory of a person. Lines show both the observed positions and the positions estimated by the Kalman filter.

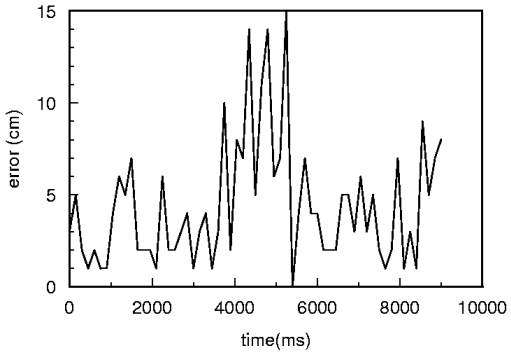


Figure 1.8: Errors in the predictions while tracking a person

In order to combine both pieces of information (color and position) into a single probability value, Eq. 1.11 has been employed. It calculates the probability value of the object obj^i to be assigned to the person s^j in the range $[0, 1]$. The Eq. 1.11 assigns values close to 1 to indicate a high probability and vice versa. The first term of this equation measures the distance between the position of the object obj^i (x_o^i, y_o^i) and the prediction for the person s^j (x_p^j, y_p^j). Values close to 1 indicate that the object is near the predicted position of the person. The parameters σ_x^j and σ_y^j are a measure of the uncertainty associated to the position predicted for s^j and are given by the Kalman filter prior uncertainty matrix. They decrease when the person does not move and increase when the person moves (in proportion to the speed) or when the person can not be tracked in a image (indicating that his position has a higher uncertainty).

The second term of the Eq. 1.11 compares the color models of the object and the person via Eq. 1.10. Finally, the parameter $\alpha_m^j \in [0, 1]$ is used to weight independently the importance of each term.

$$S(o^i, \hat{p}^i, s^j_{pred}, \hat{q}^j) = (1 - \alpha^j_m) e^{-\left(\frac{(x^i_o - x^j_p)^2}{2(\sigma^j_x)^2} + \frac{(y^i_o - y^j_p)^2}{2(\sigma^j_y)^2} \right)} + \alpha^j_m \rho(\hat{q}^i, \hat{p}^j) \quad (1.11)$$

When a person is far from others, the errors in the predicted position are less important because they are less likely to cause an error in the assignment process. Hence, a low value for α^j_m can be selected. However, relying only on position information is inappropriate when an person becomes close to others. In that case, the person could change its trajectory to avoid a collision or start an interaction with another (think of two people that approach each other to shake hands). Therefore, their predicted positions would be erroneous and even confuse the system, i.e., the identity of a person would be incorrectly assigned to the object corresponding to another person. In these cases, it is desirable a higher value for α^j_m. Nevertheless, if two people are wearing clothes of similar colors, using color information is also unreliable. To couple with all these situations, α^j_m is dynamically calculated for each person as:

$$\alpha^j_m = (1 - \rho(\hat{q}^j, \hat{q}^k)) * e^{-\left(\frac{(d^k_x)^2}{2(\sigma^j_x)^2} + \frac{(d^k_y)^2}{2(\sigma^j_y)^2} \right)}. \quad (1.12)$$

The pair (d^k_x, d^k_y) represents the displacement in the plan-view map from the person s^j to the nearest person in the scene s^k whose color model is $\hat{q}^k$. If their color models are similar then the term $(1 - \rho(\hat{q}^j, \hat{q}^k))$ tends to 0 and thus decreases α^j_m so the tracking of s^j is mostly based on his predicted position. However, if the similitude between their color models is low, α^j_m is more affected by the exponential term. The exponential term tends to have low values when the person s^j is far from the nearest person in the scene s^k (so tracking is based mostly on position). Nevertheless, it tends to increase as the s^j is nearer to s^k. It also increases as the uncertainty about the position of s^j (σ^j_x, σ^j_y) increases, reflecting that his predicted position is not reliable.

Figure 1.9 shows the trajectories of two people (s^0 and s^1) in a real test and Fig. 1.10 the evolution of α^0_m and α^1_m (in the particular case of only two people in the scene $\alpha^0_m = \alpha^1_m$). In that test, the similitude between the color models of both people was $\rho(\hat{q}^0, \hat{q}^1) = 0.12$. When s^0 and s^1 begin to move, they are far from each other so α^0_m and α^1_m are high and the tracking is mostly based on information about the predictions of their positions. While they approach each other, the distance decreases so α^0_m and α^1_m increase and the tracking is more based on color information.

In order to obtain a robust tracking system it must deal with occlusions. If the person is partially visible, it will be detected as an object if the sum of its cells in $\mathcal{O}$ is higher than the threshold θ_{occ} (previously explained in Sect. 1.4.2). The value θ_{occ} is selected so that a person could be projected

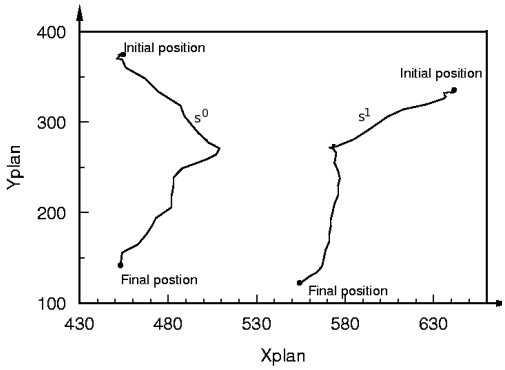


Figure 1.9: Trajectory of two people (s^0 and s^1) moving within the environment.

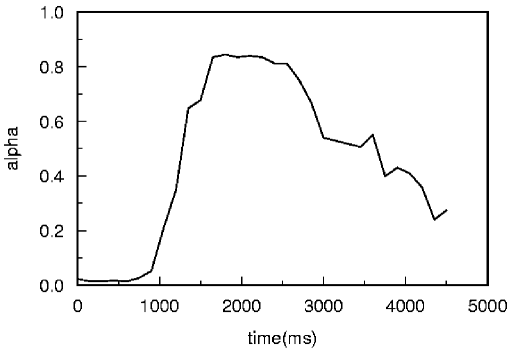


Figure 1.10: Values of the parameter α_m^0 and α_m^1 for the people s^0 and s^1 while they are moving

as an object in case of partial occlusion and dealing with stereo errors. However, if a person s^j is very occluded, it is not extracted any object from $\mathcal{O}$ for that person. In that situation the system must take care of not to assign an incorrect object obj^i to that person. For that reason, the system only consider valid an assignment when the probability value, given by Eq. 1.11, overcomes a threshold value θ_{assig}. If it does not happen for the person s^j, the system keep predicting his position and still looking for it for a maximum number of times. If the person s^j remain unseen for too long time, the system assumes that the person has definitively left the scene and deletes him from Ψ.

When the assignment problem has been solved, both the Kalman filter and the color model of each *"assigned"* person are updated using the information of its corresponding objects. The color model $\hat{q}^j$ of the person s^j is updated using the color model $\hat{p}^i$ of its corresponding object obj^i as:

$$\hat{q}_u^j = (1 - \alpha_{cu})\hat{q}_u^j + \alpha_u \hat{p}_u^i, \tag{1.13}$$

where α_{cu} weights the contribution of the new observation to the update process (31).

1.6 Experimentation

During the explanation of the model we have shown examples of its performance. A broader experimentation has been done to test detection and tracking of different people under different illumination conditions and different distances from the vision system. To perform the stereo process we have used $320x240$ sized images and sub-pixel interpolation to enhance the precision in the stereo calculation.

The operation frequency of our system is about 5 Hz on a 3.2 Ghz Pentium IV laptop computer running with Linux. Nearly the eighty percent of the computing time is dedicated to image capturing and stereo computation (about 130 ms) and the rest to detection and tracking (about 40 ms). It indicates that the proposed method for people detection and tracking itself is fast enough to be used in real time applications.

The face detector used has been configured to detect people at distances ranging from $0,5$ to $2,5$ meters. Although it could be configured for detecting people at larger distances, we have noticed that it substantially increments the time required to analyze an image. However, once a person has been located, it can be tracked up to distances of 5 meters. At higher distances, the errors of the stereo system employed are so high that people can not be correctly tracked.

A set of 18 color-with-depth video sequence, captured at 7 Hz, have been recorded in order to test the performance of the tracking process and the influence of color in it. The total time of all the sequences sum $10'23''$ and they were recorded for camera heights ranging between $0,5$ m and 1.2 m. Some of them were recorded using an stereo camera of $f = 6$ mm and the others using an stereo camera of $f = 4$ mm. The number of people in each sequence is different and it varies from 2 to 4. In the sequences, people perform several types of interactions: walk at different distances, shake hands, cross their paths, jump, run, embrace each other and even quickly swap their positions trying to confuse the system.

To evaluate the success of the system in tracking people, we have manually count the number of potentially *"conflicting"* situations that takes place in each video sequence. A conflicting situation is considered when: (i) most of the body of a person is out of the camera image, (ii) a person is almost totally occluded by another person and (iii) two or more people collide, embrace or touch each other. Figure 1.11 shows some images of one of the video sequences. An example of the conflicting situation (i) can be seen in Fig. 1.11(a). Examples of the conflicting situation (ii) can be observed in Figs 1.11(b,c,g), and examples of the conflicting situation (iii) can be seen in Figs 1.11(e,h).

Each video sequence has been processes twice: one time using color information and a second time without using it (i.e., setting $\alpha_m^j = 0$ for all the people). The processed video sequences can be downloaded in *avi* format at
http://decsai.ugr.es/∼salinas/humanrobot.htm. After processing them, the results have been examined and we have count the number of times that the system was able to successfully detect the object associated to each person when a conflicting situation had finished. Table 1.1 summarizes the results obtained. Column *#people* indicates the number of people in the test, f the focal length of the camera used and *#conflicts* the number of all the conflicting situations counted in these video sequences. Column *#nc*

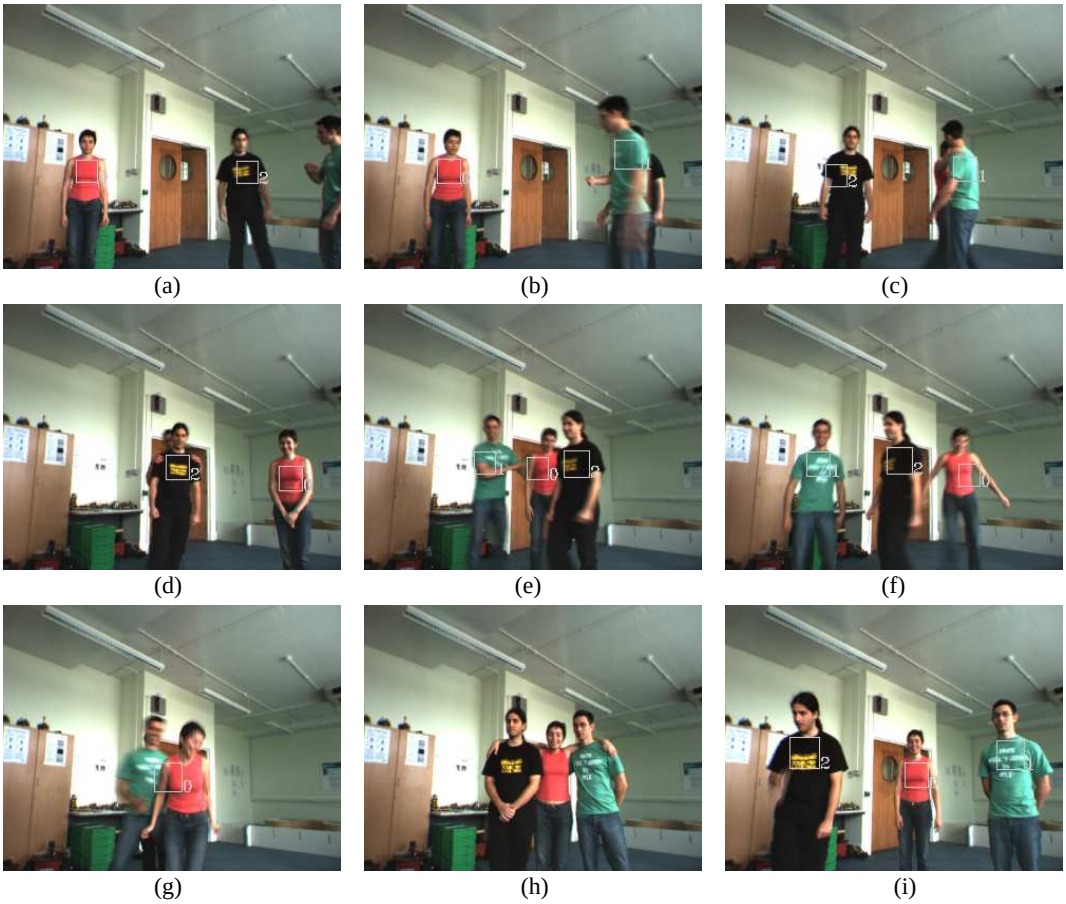


Figure 1.11: Images of a sequence employed to test the system.

indicates the success of the system in tracking the people in the conflicting situations without using color information. Finally, column $\#c$ indicates the success of the system when color information is employed.

At the light of the results showed in Table 1.1, it can be pointed out that the use of color information is a powerful clue when tracking people. We have also observed that for the $f = 6$ mm stereo camera, more than 3 people makes the tracking process unreliable because of the excessive occlusions that occurs when people is near the camera. Similarly, the maximum number of people for appropriate tracking with the $f = 4$ mm camera is 4. It can be observed that as the number of people increases, the system is more prone to fail (specially when color information is not employed). It must also be mentioned, that no false positives where detected on the video sequences. Thus, the combination of face detection and object detection in the occupancy map seems to be a very appropriate method for accurate people detection.

#people	f	#conflicts	#nc	#c
2	6 mm	30	86%	100%
3	6 mm	17	58%	82%
3	4 mm	52	69%	100%
4	4 mm	13	50%	100%

Table 1.1: Success of the tracking system when using and not using color

Additionally, we have observed that an important advantage of using color is that despite the system can incorrectly assigns a person to the object of another person in a image, the error can be corrected in the next image if both people are wearing clothes of different colors. This is because the color comparison is continuously done and the correct assignment can be done in the next image (while the confused people still close). This correction is not possible when tracking is only based on position information. Nevertheless, if the colors of the clothes of the people in the scene are similar, the system mostly use information about their predicted positions and it is more likely to fail when people are close each other.

1.7 Conclusions and Future Work

We have presented a system able to detect and track multiple people using an stereo camera placed at under-head positions. The method proposed is especially indicated for applications that require analyzing the user gestures and facial expression because of the position of the camera.

The system uses a height map built using depth information to model the environment that can be created and updated even in the presence of moving people in it. The background model obtained is more robust to sudden illumination changes than approaches based on intensity values because of the use of depth information (7).

Each time a new stereo pair is captured, an occupancy map that registers the position of the foreground objects is created. Foreground objects with dimensions similar to human beings are considered as potential candidates to people and a color model of each one of them is created.

Then, the system performs the tracking of the already known people. Tracking is considered as an assignment problem, i.e., assign known people to the objects detected in the occupancy map. To calculate the best assignment scheme, the Hungarian Method (25) has been employed. For that purpose it is necessary to calculate a probability value of each object detected to be a known person. This probability value is calculated combining information about a color model of each person and its predicted position using the Kalman filter. The combination is dynamically done is the following way. When a person is far from others, the probability value is mostly based on his predicted position. Nevertheless, when he comes closer to oth-

ers, position information becomes unreliable. Thus, the system takes into account information about the color of his clothes to prevent confusing him with others. The degree of confidence assigned to color information when tracking a person is based on the similarity between the colors of his clothes and the colors of the clothes of the people in his surroundings. Once the assignment problem is solved, both the Kalman filter and the color models of the tracked people are updated.

The remaining objects not belonging to any known person are examined using a face detector. The aim is to detect the new people that enters in the scene. Instead of employing the face detector on the entire camera image, it is only applied on selected regions where is more probable to find the face in each object. This technique allows to greatly reduce the computing time required and helps to avoid false positives of the face detector (24).

The system has been extensively tested on 18 color-with-depth video sequences (summing a total time of $10''23'$) where several people move freely in the environment. The video sequences have been processed twice, with and without color, in order to analyze the influence of color in the tracking process. The results show that the use of color allows to greatly reduce the number of errors of the tracking system. The proposed system is able to track multiple-persons up to distances of 5 meters with very high success rate without using complex three dimensional models to describe the scene. Besides, the time required to perform the detection and tracking is only 40 ms, what induces as to think that it could be suitable for real time applications. It is also important to remark that the no false detections where registered in the tests performed.

As future work, we consider of interest the use of multiscale techniques for creating the plan-view maps (14) and avoiding the use of a unique cell size. This could allow to manage the errors of the stereoscopic system in a more flexible way. We also consider that it could be interesting the use of additional features that could discriminate between people wearing similarly colored clothes. In particular, the use of face identification techniques seem interesting for that purpose. Finally, we find that the system is specially appropriated for being employed on mobile devices. Therefore, as future work we also plan to include the system into a bigger architecture that controls and autonomous mobile robot (2; 29) and testing it suitability for human-machine applications that requires to operate in real time and in moving conditions.

Bibliography

[1] R. Mohan adn G. Medioni and R. Nevatia. Stereo error detection, correction, and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:113 – 120, 1989.

[2] E. Aguirre and A. González. Fuzzy behaviors for mobile robot navigation: Design, coordination and fusion. *International Journal of Approximate Reasoning*, 25:255–289, 2000.

[3] F. Aherne, N. Thacker, and P. Rockett. The Bhattacharyya Metric as an Absolute Similarity Measure for Frecuency Coded Data. *Kybernetica*, 32:1–7, 1997.

[4] AA. Argyrs and MI. Lourakis. Three-dimensional tracking of multiple skin-colored regions by a moving stereoscopic system. *Applied Optics*, 43:366–378, 2004.

[5] W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thurn. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 144:3–55, 1999.

[6] D. Comaniciu, V. Ramesh, and P. Meer. Real-Time Tracking of Non-Rigid Objects using Mean Shift. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 142–149, 2000.

[7] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb. Plan-view trajectory estimation with dense stereo background models. In *Eighth IEEE International Conference on Computer Vision (ICCV 2001)*, volume 2, pages 628 – 635, 2001.

[8] T. Darrell, G. Gordon, M. Harville, and J. Woodfill. Integrated Person Tracking Using Stereo, Color, and Pattern Detection. *Int. Journ. Computer Vision*, 37:175–185, 2000.

[9] C. Eldershaw and M. Yim. Motion planning of legged vehicles in an unstructured environment. In *IEEE International Conference on Robotics and Automation (ICRA'2001)*, volume 4, pages 3383 – 3389, 2001.

[10] C. Eveland, K. Konolige, and R.C. Bolles. Background Modelling for Segmentatation of Vide-Rate Stereo Sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 266–271, 1998.

[11] J. Foley, A. Van Dam, S. Feiner, and J. Hughes. *Computer Graphics: Principles and Practice*. Addison Wesley, 1990.

[12] J.D. Foley and A. van Dam. *Fundamentals of Interactive Computer Graphics*. Addison Wesley, 1982.

[13] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42:143–166, 2003.

[14] M. García-Silvente, J.A. García, J. Fdez-Valdivia, and A.Garrido. A new edge detector integrating scale-spectrum information. *Image and Vision Computing*, 15:913– 923, 1997.

[15] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999.

[16] D. Grest and R. Koch. Realtime multi-camera person tracking for immersive environments. In *IEEE 6th Workshop on Multimedia Signal Processing*, pages 387–390, 2004.

[17] M.S. Grewal and A.P. Andrews. *Kalman Filtering. Theory and Practice*. Prentice Hall, 1993.

[18] I. Haritaoglu, D. Beymer, and M. Flickner. Ghost 3d: detecting body posture and parts using stereo. In *Workshop on Motion and Video Computing*, pages 175 – 180, 2002.

[19] M. Harville, G. Gordon, and J. Woodfill. Foreground segmentation using adaptive mixture models in color and depth. In *IEEE Workshop on Detection and Recognition of Events in Video*, pages 3–11, 2001.

[20] Michael Harville. Stereo person tracking with adaptive plan-view templates of height and occupancy statistics. *Image and Vision Computing*, 2:127–142, 2004.

[21] K. Hayashi, M. Hashimoto, K. Sumi, and K. Sasakawa. Multiple-person tracker with a fixed slanting stereo camera. In *6th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 681–686, 2004.

[22] Intel. OpenCV: Open source Computer Vision library. *http://www.intel.com/research/mrl/opencv/*.

[23] T. Kailath. The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology*, 15:52 – 60, 1967.

[24] H. Kruppa, M. Castrillon-Santana, and B. Schiele. Fast and Robust Face Finding via Local Context. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2003.

[25] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

[26] W. Liang, H. Weiming, and L. Tieniu. Recent developments in human motion analysis. *Pattern Recognition*, 36:585–601, 2003.

[27] R. Lienhart and J. Maydt. An Extended Set of Haar-Like Features for rapid Object detection. In *IEEE Conf. on Image Processing*, pages 900–903, 2002.

[28] B. Martinkauppi, M. Soriano, and M. Pietikainen. Detection of skin color under changing illumination: a comparative study. In *12th International Conference on Image Analysis and Processing*, pages 652 – 657, 2003.

[29] R. Muñoz-Salinas, E. Aguirre, M. García-Silvente, and M. Gómez. A multi-agent system architecture for mobile robot navigation based on fuzzy and visual behaviours. *To appear in Robotica*, 2005.

[30] K. Nickel, E. Seemann, and R. Stiefelhagen. 3D-Tracking of Head and Hands for Pointing Gesture Recognition in a Human-Robot Interaction Scenario. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR'04)*, pages 565–570, 2004.

[31] K. Nummiaro, E. Koller-Meier, and L. Van Gool. An adaptive color-based particle filter. *Image and Vision Computing*, 21:99–110, 2003.

[32] PtGrey. Bumblebee. *http://www.ptgrey.com/products/bumblebee/index.html*.

[33] J.J. Rodriguez and J.K. Aggarwal. Stochastic analysis of stereo quantization error. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:467 – 470, 1990.

[34] K. Sabe, M. Fukuchi, J.-S-Gutmann, T. Ohashi, K. Kawamoto, and T. Yoshigahara. Obstacle avoidance and path planning for humanoid robots using stereo vision. In *IEEE International Conference on Robotics and Automation (ICRA'04)*, volume 1, pages 592 – 597, 2004.

[35] L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:862 – 877, 2004.

[36] L. Snidaro, C. Micheloni, and C. Chiavedale. Video security for ambient intelligence. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 35:133 – 144, 2005.

[37] R. Tanawongsuwan. Robust Tracking of People by a Mobile Robotic Agent. Technical Report GIT-GVU-99-19, Georgia Tech University, 1999.

[38] S. Thompson and S. Kagami. Stereo vision terrain modeling for non-planar mobile robot mapping and navigation. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 6, pages 5392 – 5397, 2004.

[39] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 511–518, 2001.

[40] C.R. Wren, A. Azarbayejani, T. Darrell, Trevor, and A.P. Pentland. Pfinder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:780–785, 1997.

[41] M.H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.