# Using Stereo Vision and Fuzzy Systems for Detecting and Tracking People

Rui Paúl *, Eugenio Aguirre, Miguel García-Silvente, and Rafael Muñoz-Salinas

Department of Computer Science and A.I., E.T.S. Ingeniería Informática,
University of Granada, 18071 Granada, Spain.
Department of Computing and Numerical Analysis, E.P.S.,
University of Córdoba, Córdoba, Spain.
{ruipaul,eaguirre,m.garcia-silvente}@decsai.ugr.es
rmsalinas@uco.es

**Abstract.** This paper describes a system capable of detecting and tracking various people using a new approach based on stereo vision and fuzzy logic. First, in the people detection phase, two fuzzy systems are used to assure that faces detected by the OpenCV face detector actually correspond to people. Then, in the tracking phase, a set of hierarchical fuzzy systems fuse depth and color information captured by a stereo camera assigning different confidence levels to each of these information sources. To carry out the tracking, several particles are generated while fuzzy systems compute the possibility that some generated particle corresponds to the new position of people. The system was tested and achieved interesting results in several situations in the real world.

**Key words:** People Tracking, Stereo Vision, Fuzzy systems, Particle Filtering, Color Information

## 1 Introduction and Related Work

People detection and tracking can be done in various ways and with different kind of hardware. When computer vision is used, the system analyzes the image and searches for cues that provide important information in the detection of people. Those cues could be, for instance, morphological characteristics of the human body [1]. Due to illumination change problems some authors have opted to use dynamic skin color models [2].

In this work stereo vision has been used so 3D information could be extracted from the images. This information is relatively invariable with respect to illumination changes. In [3], the authors present a system capable of detecting and tracking several people. Their work is based on a skin detector, a face detector and the disparity map provided by a stereo camera. In the work of Grest and

* This work is supported by the FCT Scolarship SFRH/BD/22359/2005, Spanish MCI Project TIN2007-66367 and Andalusian Regional Government project P09-TIC-04813.

Koch [4] a particle filter [5] is also used to estimate the position of the person and create color histograms of the face and breast regions of that person and stereo vision to compute the real position of the person in the room. However, stereo and color were not integrated in the tracking process and they use cameras positioned in different parts of a room rather than one stereo camera. Moreno *et. al.* [6] present a system able to detect and track a single head using the Kalman filter. They combined color and stereo information but head color does not provide enough information to distinguish among different users. In [7] and [8], the authors present an approach to detect and track several people using *plan-view maps*. They use information provided by an *occupancy map* and a *height map* using the Kalman filter.

In our approach the problem is solved using a new approach based on a particle filter which generates particles that are evaluated by means of fuzzy logic. Although we also use depth and color information as sources of information, they are supplied to several hierarchically organized fuzzy systems. People tracking is done by generating different particles in the image and then computing their possibility to be part of a previous detected person using a fuzzy system approach. We opted for using fuzzy logic [9] in order to have the possibility of dealing with uncertainty and vagueness in a flexible manner so we can avoid possible restrictions when representing imprecision and uncertainty with probabilistic models. Furthermore, when using linguistic variables and rules to define the behavior of our system, it turns out to be more understandable and similar to the way humans represent and process knowledge.

## 2 People Detection and Tracking

Our system is based on a stereo camera (BumbleBee model) that allows us to extract not only visual information from the images but also depth information about most of the objects appearing in the images. By combining these two different types of information it is possible to achieve a more robust tracking than when using only one of them. If one of them fails, it is possible to keep track of a person by using the other and vice-versa.

### 2.1 People Detection

The detection of people begins with a face detector phase. This is done by using the face detector available in the OpenCV library [10], that is free to use and download. Although this detector is free, fast and able to detect people with different morphological faces, false positives can be found. The classifier outputs the rectangular region(s) of the faces detected in our RGB image. In order to reject possible false positives each of the detected face(s) have to pass two tests to assure that it belongs to a person.

The first test use the concept of the projection of the model of a person. Taking into account the usual size of a person we can estimate the projection of a person in our camera image, according to his or her distance to the camera and

knowing the intrinsic parameters of the camera. From now on we will call the projection of the model of a person as $R_p$ (standing for Region of Projection). Fig.1 shows the region of projection in a stereo image and its corresponding reference image.

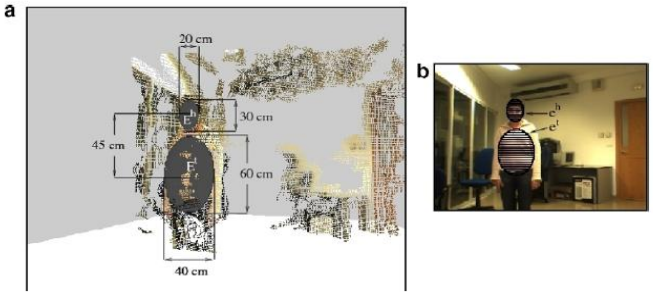


**Fig. 1.** (a) Model employed. (b) Projection of the model on the reference image

The goal of this test is to check whether inside $R_p$ there are enough pixels respecting three conditions: they have to belong to the foreground (if they belong to the background they cannot be considered as being part of a person), they have disparity information (if there is a person in $R_p$ then there should be a high number of pixels containing depth information) and they are not occluded (if most of the pixels inside $R_p$ are occluded then $R_p$ represents a region where visual and depth information, important for the tracking process, is not sufficient and consequently trustable).

These three measures are fuzzified by three linguistic variables labeled as $ForegroundPixels$, $StereoPixels$ and $NonOccludedPixels$, respectively (see Fig.2). Using these three variables as input variables to the Fuzzy System 1 shown by Table 1, the fuzzy output $DetectedFace$ is computed. Fuzzy System 1 and the rest of the fuzzy systems shown in this work use the Mamdani inference method. The defuzzified value of $DetectedFace$ indicates the possibility, from 0 to 1, whether region $R_p$ is worth to contain a true positive face. If this value is higher than $\alpha_1$, the detected face passes to the second and last test. The second test also checks whether $R_p$ may contain a true positive face. However the idea is different now. If there is a person in that region, then pixels inside $R_p$ should have approximately the same depth. Therefore the Fuzzy System 2 receives, as input, the difference between the average depth of $R_p$ and the depth of the detected face as seen in Eq.1.

$$d = |Z - \frac{\sum_{j=1}^{n}(z_j)}{n}|. \tag{1}$$

where $d$ is the difference we want to compute, $Z$ the actual depth of the detected face, $z_j$ the depth of the $j$ pixel inside $R_p$ and $n$ the total number of pixels inside $R_p$. This value is fuzzified by the linguistic variable $AverageDifference$.

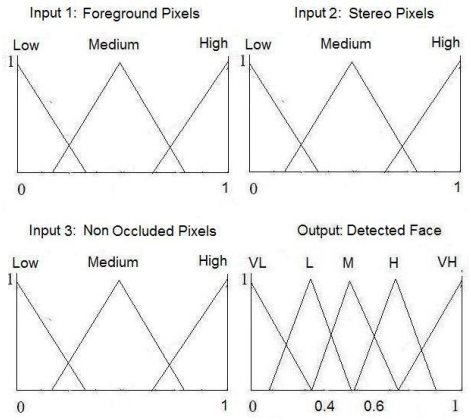


**Fig. 2.** Fuzzy sets for detecting faces with variables ForegroundPixels (ratio), StereoPixels (ratio), NonOccludedPixels (ratio) and DetectedFace

**Table 1.** Rules for Fuzzy System 1.

| IF | | | THEN |
|---|---|---|---|
| ForegroundPixels | StereoPixels | NonOccludedPixels | DetectedFace |
| High | High | High | Very High |
| High | High | Medium | High |
| High | High | Low | Medium |
| High | Medium | High | High |
| ... | ... | ... | ... |
| Low | Low | Low | Very Low |

Fuzzy System 2 also receives the standard deviation of those pixels, fuzzified by the linguistic variable *StandardDeviation*, and the depth at which the face was detected, fuzzified by the linguistic variable *Depth*. Depth of the detected face is used to compute the confidence that we should assign to the values of the other variables. At farther distances, the uncertainty is higher. The output variable *SimilarDepth* is computed by Fuzzy System 2 and its defuzzified value is a value between 0 and 1 corresponding to the possibility that $R_p$ contains pixels with depth similar to the depth of the detected face. In Fig.3 linguistic variables $AverageDifference$, *StandardDeviation*, *Depth* and *SimilarDepth* (output) are shown. In Table 2 it is possible to find examples of the rules defined for Fuzzy System 2.

Finally, if this value is higher than $\alpha_2$, we assume that a person was detected and we assign a tracker for him or her. The values for parameters $\alpha_1$ and $\alpha_2$ have been experimentally tuned.

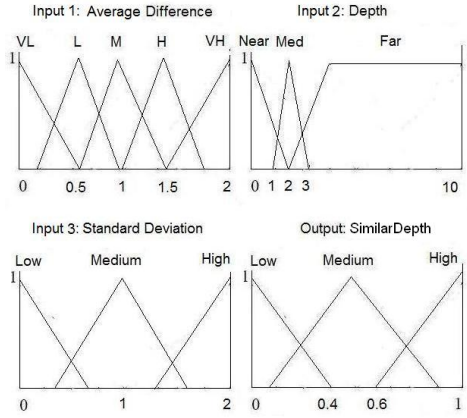


**Fig. 3.** Fuzzy sets for detecting faces with variables AverageDifference (meters), Depth (meters), StandardDeviation (meters) and SimilarDepth

**Table 2.** Rules for Fuzzy System 2.

| IF | | | THEN |
|---|---|---|---|
| AverageDifference | Depth | StandardDeviation | SimilarDepth |
| VL | Far | Low | High |
| VL | Far | Medium | High |
| VL | Far | High | Medium |
| L | Far | Low | High |
| ... | ... | ... | ... |
| VH | Near | High | Low |

The rules and linguistic variables defined for other fuzzy systems in Section 2.2 are similar to the ones of Figures 2, 3 and Tables 1, 2 so that they are omitted in order to not exceed the allowed number of pages of this paper.

### 2.2 People Tracking

As we said before, a tracker is created for each person detected. The order of the trackers goes from the person that is closer to the camera to the person that is farther. The tracking process is divided into two parts. In the first one, a particle filter approach is used to generate and evaluate possible new positions for the person being tracked. This particle filter is based on the condensation algorithm, but particles are evaluated by means of fuzzy logic rather than using a probabilistic approach. After some experiments, 50 particles were considered to be sufficient to keep track of people without compromising performance. In the second one, the average position of all particles is computed. This average is an weighted average, based on the value of possibility $PossibilityP(i)$ of each previously generated particle. The average position $PersonPos_{(t)}$ is the new

position of the person in 3D. We consider that the position of the person is his or her face position. There are as many trackers as people being tracked. So, this phase is repeated as many times as the number of people being tracked.

The generation of a new particle, is based on the previous position of the average particle in the previous frame $PersonPos_{(t-1)}$. The idea is to generate most particles in the surroundings of the previous and a few farther as people are not expected to move fast from frame to frame (the frame rate is 10 fps). The propagation model of the particles is based on the previous position of the person plus some $\delta$ value that follows a gaussian distribution with parameters $N(\mu = 0\,m, \sigma = 0.1\,m)$. After generating the set of particles we begin the process of evaluating the possibility ($PossibilityP(i)$) that each particle corresponds to the tracked person. The observation model for each particle is based on the output of different fuzzy systems as shown in Fig.4. We use a two layer fuzzy system approach to take into account the confidence level of the outputs of some of the fuzzy systems. This situation will be explained later when each of the fuzzy system is described. Finally, the overall result for each particle is given by $PossibilityP(i) = OutFS_5 * OutFS_6 * OutFS_7$ where $OutFS_i$ stands for the "ith" Fuzzy System defuzzified output and is a value between 0 and 1 (see Fig.4).

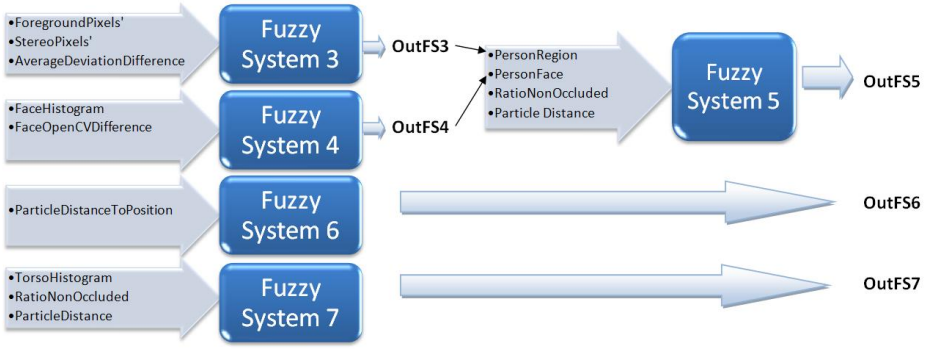


**Fig. 4.** Fuzzy Systems used to evaluate de overall quality of each generated particle. For each fuzzy system, the input linguistic variables are specified.

The goal of "Fuzzy System 3" is to evaluate the region of projection of some person $R_p(P_i)$ (see Fig.1) according to the depth of the current particle being evaluated ($P_i$). This evaluation will take into consideration only aspects related with the possibility that some object, similar to a person, is located in that region. The first step is to compute the area of $R_p(P_i)$. After obtaining this information we define three linguistic variables: $ForegroundPixels'$, $StereoPixels'$ and $AverageDeviationDifference$. $ForegroundPixels'$ and $StereoPixels'$ are defined in a similar way to $ForegroundPixels$, $StereoPixels$ at Section 2.1. $AverageDeviationDifference$ gives us information about the difference be-

tween the depth of $P_i$ and the depth average of all pixels inside $R_p(P_i)$. This value is also fused with the standard deviation for those pixels. The reason for defining this variable is that, all pixels inside $R_p(P_i)$, should have approximately the same depth as $P_i$ and should have approximately the same depth between them, as long as they belong to some person or object. These values will be the input to Fuzzy System 3 that will output a deffuzified value between 0 and 1. The higher amount of foreground, disparity pixels and lower difference in average and standard deviation, the closer the output is to 1. A value closer to 1 means that, in the area represented by $R_p(P_i)$, it is likely to have some object that could hypothetically be a person.

The scope of "Fuzzy System 4" is to evaluate face issues related to the person being tracked. We define two linguistic variables called $FaceHistogram$ and $FaceOpenCVDistance$. The first one contains information about the similarity between the face region of $R_p(P_i)$ and the face histogram of the person being tracked. As people from frame to frame (at a 15 fps frame rate) do not tend to move or rotate their face so abruptly, the histograms should be similar. We use the elliptical region of the face to create a color model [11]. We then measure the difference between the face histogram of region of $R_p(P_i)$ and the face histogram of the person being tracked. This difference is based on a popular measure between two color distributions: the Bhattacharyya coefficient [12]. This method gives us the similarity measure of two color models in the range $[0, 1]$. Values near 1 mean that both color models are identical. Values near 0 indicate that the distributions are different. An important feature of this method is that two color models can be compared even if they have been created using a different number of pixels. The second linguistic variable measures the distance between $P_i$ and the position of the nearest face to $P_i$ detected by the OpenCV face detector. Although OpenCV is not 100% accurate, most of time this information can be worth as it can tell if there is really a face near $P_i$. The deffuzified output of this fuzzy system is also a number between 0 and 1 where 1 is an optimal value.

The deffuzified outputs of "Fuzzy System 3" and "Fuzzy System 4" are then provided as input for another fuzzy system that we call "Fuzzy System 5". This fuzzy system allows us to measure the confidence of the outputs of Fuzzy Systems 3 and 4 based on occlusion and depth information. We define four linguistic variables called $PersonRegion$, $PersonFace$, $RatioNonOccluded$ and $ParticleDistance$ to compute the final output for Fuzzy System 5. $PersonRegion$ and $PersonFace$ have five linguistic labels Very Low, Low, Medium, High and Very High distributed in a uniform way into the interval $[0, 1]$ in a similar way to the membership functions of $AverageDifference$ shown by Fig. 3. Their inputs are the defuzzified outputs of "Fuzzy System 3" and "Fuzzy System 4" respectively. $RatioNonOccluded$ contains information about the ratio of non occluded pixels inside $R_p(P_i)$. The higher the number of non occluded pixels, the more confidence we have on the output values. In other words, the more pixels we can use from $R_p(P_i)$ to compute foreground, depth, average information and histogram the more trustable the outputs of "Fuzzy System 3" and "Fuzzy System 4". Finally $ParticleDistance$ has information about the distance of the particle

evaluated ($P_i$). As errors in stereo information increase with distance, the farther the particle is located, the less trustable it is in means of depth information. The defuzzified output of "Fuzzy System 5" ($OutFS_5$) is also a number between 0 and 1. Higher values indicate a region with higher possibility to contain a person.

With respect to "Fuzzy System 6", this fuzzy system's goal is to evaluate whether $P_i$ is likely to be the person being followed taking into consideration the distance to the previous location of the person (in the frame before). Due to the frame rate used, people from frame to frame are not expected to move significantly. Therefore, we define only one variable called $ParticleDistanceToPosition$ that contains information about the 3D distance between the 3D position of $P_i$ and the 3D position of the currently tracked person ($PersonPos_{(t-1)}$). The deffuzified output will be, once again, a value between 0 and 1 represented by $OutFS_6$. An output equal to 1 means that $P_i$ is located exactly in the same place where $PersonPos_{(t-1)}$ was located.

The last fuzzy system ("Fuzzy System 7") is related with torso information. Identically to "Fuzzy System 4" we also define a variable that translates the similarity between the torso histogram information of $R_p(P_i)$ and the histogram information of the torso of the person being tracked. This variable is called $TorsoHistogram$. We also use for this fuzzy system, the variables $RatioNonOccluded$ and $ParticleDistance$ analogously to "Fuzzy System 5". When doing this, we are adding a measure of confidence for the output which after its deffuzification is called $OutFS_7$ and has a value between 0 and 1.

As said before, all these outputs are multiplied and result on a final value between 0 and 1. Then an weighted average of the 3D position $PersonPos_{(t)}$ is computed by taking into consideration all the possibility values for the set of particles. A particle that has a possibility value closer to 1 will weight much more than one with a possibility value of 0. Its $R_p(P_i)$ is also added to an occlusion map, so the following trackers and the people detection's algorithm know, that there is already a person occupying that region. This occlusion map is reset every time a new frame is processed. The face and torso histograms are also updated.

## 3 Experimental Results

The system was tested in various scenarios and with different people. Videos were recorded with a resolution of 320x240 pixels and the system was tested with an Intel Core 2 Duo 3.17 Processor (only one processor is used for processing). The achieved operation frequency of our system was about 10 Hz in average depending on the number of people being tracked. As each tracked person implies a new tracker, processing time increases in average by 50 ms for each added tracker. We consider up to 4 people for the system to perform in real time with this kind of camera and processor.

We recorded 15 videos with 1, 2 and 3 people moving freely in a room. The set of videos provided over 15 minutes of recording with various people interacting freely. We could observe that, when either disparity or color information were not completely reliable, the system still kept track of the people. The average

accuracy rate for tracking people in the test set was over 90%. This result was achieved after tuning different settings of the fuzzy systems. We think an higher rate could be achieved if these values keep being tuned.

In Fig.5 we show four frames taken from one of those videos, with both reference image and disparity image shown for each frame. In the disparity image, lighter areas represent shorter distances to the camera. In Fig.5(a) it is possible to see that the system detected person A (ellipse 1) while person B was not detected. In Fig.5(b) we can see that person B was detected (ellipse 2) since most of the pixels were visible. In Fig.5(c) it is possible to see that, although depth information for both people was very similar, the system could still keep an accurate track for each of the people. The reason for achieving this accuracy relies on color information that compensated the similarity of depth information. Finally in Fig.5(d) it is possible to see that, for person A, although part of his body was occluded, the system could still achieve an accurate tracking, based on disparity information rather than color information.

We would also like to mention that, when people cross their paths, the system manages to keep track of each person by making use of both depth and color information. However, situations in which two people dressed with the same colors and located at the same distance got very close, could originate that the system would confuse both targets. This issue is expected to be solved in a near future by providing more information sources to the system.

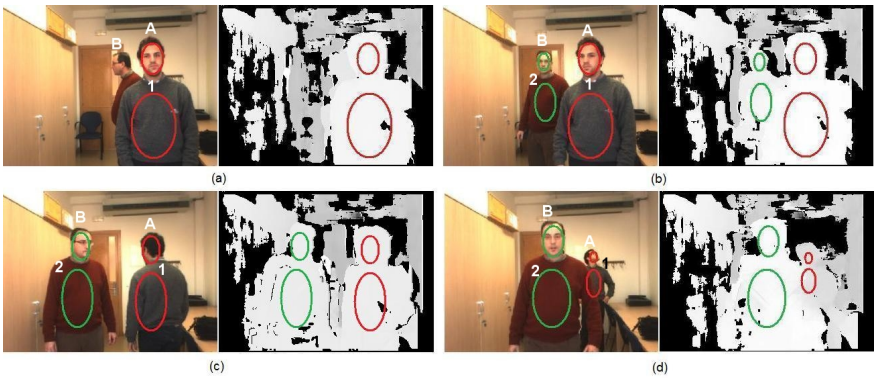


**Fig. 5.** Different frames taken from a video with 2 people being tracked

## 4 Conclusions and Future Work

The system proposed proved to work in real life situations, where people were interacting freely and occluded each other sometimes. The system was capable of detecting and tracking people based on fuzzy logic as it has proven in the past that it is an interesting tool for treating uncertainty and vagueness. A particle

filter is used to generate particles that are evaluated using fuzzy logic instead of probabilistic methods. As we know, information supplied by sensors is commonly affected by errors, and therefore the use of fuzzy systems help us to deal with this problem. In our case, as stereo information is not 100% accurate, we may sometimes rely more on color information and solve that problem. On the other hand, we can easily manage unexpected situations as, for instance, sudden illumination changes, by giving more importance to stereo information. By setting up linguistic variables and rules that deal with this problem we achieved an efficient way of solving it. Also, when using fuzzy systems to represent knowledge, the complexity in understanding the system is substantially lower as this kind of knowledge representation is similar to the way the human being is used to represent its own knowledge. Furthermore, it allows an easy way of adding new features, just by adding more variables or fuzzy systems.

In this work, rules and linguistic variables are defined after testing different values in different experiments. As a future work, we would like to build a system capable of learning and therefore adjusting these parameters automatically.

## References

1. Hirai, N., Mizoguchi, H.: Visual tracking of human back and shoulder for person following robot. IEEE/ASME International Conference on Advanced Intelligent Mechatronics vol. 1, 527–532 (2003)
2. Sigal, L., Sclaroff, S., Athitsos, V.: Skin color-based video segmentation under time-varying illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence 26, 862–877 (2003)
3. Darrell, T., Gordon, G., Harville, M., Woodfill, J.: Integrated person tracking using stereo, color, and pattern detection. International Journal of Computer Vision 37, 175–185 (2000)
4. Grest, D., Koch, R.: Realtime multi-camera person tracking for immersive environments. IEEE Sixth Workshop on Multimedia Signal Processing. 387–390 (2004)
5. Isard, M., Blake, A.: CONDENSATION-conditional density propagation for visual trackings. International Journal of Computer Vision. 29, 5–28 (1998)
6. Moreno, F., Tarrida, A., Andrade-Cetto, J., Sanfeliu, A.: 3D real-time head tracking fusing color histograms and stereovision. International Conference on Pattern Recognition. 368–371 (2002)
7. Harville, M.: Stereo person tracking with adaptive plan-view templates of height and occupancy statistics. Image and Vision Computing. 2, 127–142 (2004)
8. Muñoz-Salinas R., Aguirre E., García-Silvente M.: People Detection and Tracking using Stereo Vision and Color. Image and Vision Computing, 25, 995–1007 (2007)
9. Yager, R. R., Filev, D.P.: Essentials of Fuzzy Modeling and Control. John Wiley & Sons, Inc. (1994)
10. Intel, OpenCV: Open source Computer Vision library, `http://www.intel.com/research/mrl/opencv/`
11. Birchfield, S.: Elliptical head tracking using intensity gradients and color histograms. IEEE Conf. Computer Vision and Pattern Recognition, 232–237 (1998)
12. Kailath, T.: The divergence and Bhattacharyya distance measures in signal selection. IEEE Transactions on Communication Technology 15, 52–60 (1967)