

Human Detection in Indoor Environments Using Multiple Visual Cues and a Mobile Robot

Stefan Pszczółkowski and Alvaro Soto

Pontificia Universidad Catolica de Chile
Santiago 22, Chile
spp@ing.puc.cl, asoto@ing.puc.cl

Abstract. In order to deploy mobile robots in social environments like indoor buildings, they need to be provided with perceptual abilities to detect people. In the computer vision literature the most typical solution to this problem is based on background subtraction techniques, however, in the case of a mobile robot this is not a viable solution. This paper shows an approach to robustly detect people in indoor environments using a mobile platform. The approach uses a stereo vision system that yields a *stereo pair* from which a *disparity image* is obtained. From this disparity image, interesting objects or *blobs* are segmented using a *region growing algorithm*. Afterwards, a color segmentation algorithm is performed on each blob, searching for human skin color areas. Finally, a probabilistic classifier provides information to decide if a given skin region corresponds to a human. We test the approach by mounting the resulting system on a mobile robot that navigates in an office type indoor building. We test the system under real time operation and different illumination conditions. The results indicate human detection accuracies over 90% in our test.

Keywords: human detection, human-computer interaction, face detection.

1 Introduction

An important field in Robotics is *Socially interactive robots* [6], which consists in providing robots with the ability to interact with other agents. To effectively interact socially, robots have to separate possible agents from the rest of the scene. Then, they have to discriminate which of these candidate agents they can interact with. The most common separation is between humans and other objects, like furniture, doors, and decorations.

Serving as a contribution towards the development of socially interactive robots, this paper shows an approach to robustly detect people in indoor environments. Our goal is to mount our system on a mobile robot navigating through an indoor environment, therefore, the use of traditional background subtraction techniques is not a viable solution.

In our case, our approach is based on information provided by a stereo vision system to perform the initial segmentation of candidate humans. We use the fact

that for each person in the scene, its depth is roughly constant and appears on the disparity image as a uniform intensity area. After this initial segmentation, we use color cues to detect skin color pixels that feed a probabilistic classifier that provides the final detection of humans.

The rest of the paper is organized as follows: Chapter 2 reviews relevant previous work on human detection using computer vision techniques. Chapter 3 discusses the main details of our approach to detect people. Chapter 4 shows the results of applying our methodology to real data in real time. Finally, chapter 5 presents the main conclusions of this work.

2 Previous Work

Human detection and tracking are important topics of research in the computer vision literature. Applications for these topics include surveillance, elderly assistance, human-robot interaction, and pedestrian counting, among others [9]. The state of the art in this area can be divided into two main categories: i) Methods that require background subtraction as a first step to detect the interesting objects. ii) Methods that perform the detection using moving cameras. Our method belongs to this last category, hence, we concentrate the review here in methods that do not rely in background subtraction techniques, for a more extensive review see [9].

The work in [11] proposes a method based on geometrical structures. It uses the fact that the relative positions of various body parts are common to all humans. On each input image, patches at multiple locations and scales are compared to previously stored templates. Then a threshold is used to classify a patch as a human or a non-human. Recognition rates between 83% and 90% are presented for this method.

The method proposed by [12] deals with the detection of pedestrians from video. The algorithm scans a detector over two consecutive frames of a video sequence and extracts simple rectangular features by evaluating motion and appearance filters. The detector is a cascade of classifiers that is trained using AdaBoost. A static detector only with appearance information is also presented. Results with low false positive rates and detection rates of about 80% are shown.

The work in [10] proposes a method for human detection in video sequences for outdoor surveillance. The technique computes optic flow of several human and non-human motion examples and trains a support vector machine (SVM) with radial basis function (RBF) kernel using these examples. The classifier can be applied to new input video at multiple positions and scales, followed by pruning of detections with large overlap. Good recognition performance for walking people are shown, even in the presence of other moving objects.

Recently, [3] describes a method that uses grids of Histograms of Oriented Gradient (HOG) descriptors for building a support vector machine classifier. It divides the image in small spatial parts (cells) and finds the histograms of edge orientations over all the pixels of the cell. The combined histogram entries form the feature representation after local contrast normalization in overlapping

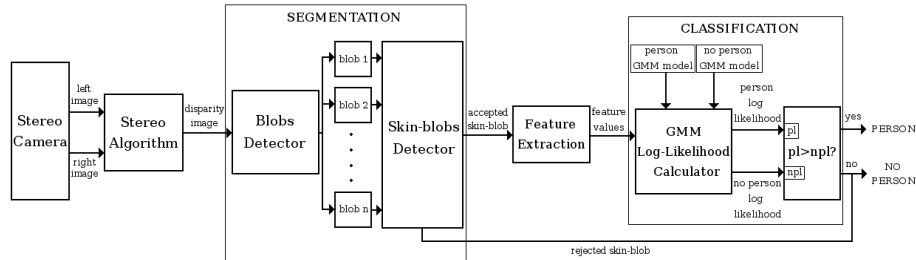


Fig. 1. System diagram. The images obtained from the stereo cameras are passed to the segmentation process, where the stereo blobs and face candidates are detected. Then features are extracted from the face candidates to feed a classifier that distinguishes between person and not person.

descriptor blocks. The inclusion of four different normalizations on each HOG improves performance from 84% to 89%.

3 Our Approach

In this chapter we show the different parts of our approach and how they are integrated to effectively detect people. Figure 1 shows an overview of the approach. It consists of a stereo vision system that yields a *stereo pair* from which a *disparity image* is obtained. From this disparity image, interesting objects or *blobs* are segmented using a *region growing algorithm*. Afterwards, a color segmentation algorithm is performed on each blob, searching for human skin color areas. Then, over each blob a feature extraction process provides information to a probabilistic classifier that finally distinguishes if a given skin region corresponds to a human. In the next, we explain each of these steps in detail.

3.1 Stereo-Based Human Segmentation

The process we developed to make human segmentation is based on the idea that for each person in the scene, its depth is roughly constant and appears on the disparity image as a uniform-intensity area. Based on that, one can separate humans from the background by finding these areas.

To obtain the stereo pair we use the SVS library [8] and the disparity image is calculated using the library implementation of the *Area Correlation Method*. Then, a Breadth First Search (BFS) *region growing* algorithm is performed over the disparity image. This algorithm iteratively looks around each pixel searching for neighbors with similar gray intensity and connecting them. This process yields several regions of connected pixels known as *blobs*. Empirically, we set that blobs smaller than 3.26% of the total image area are filtered out.



Fig. 2. Skin pixel criteria applied to some images. Note that some colors of the T-shirts in the first image are more difficult to filter out due to its similarity to some skin pigmentations.

3.2 Skin-Color Based Segmentation

We want to detect humans, hence, a useful visual cue is skin color. The procedure to obtain skin color blobs consists in searching for skin pixels inside the blobs detected by the stereo vision algorithm (stereo blobs). To effectively classify between *skin* pixels and *non-skin* pixels we used a transformation of the RGB values into a “log color-opponent” space [4]. This space can directly represent the approximate hue of skin color:

$$\log Val_1 = \ln(G); \log Val_2 = \ln(R) - \ln(G); \log Val_3 = \ln(B) - \frac{\ln(R) + \ln(G)}{2} \quad (1)$$

We classify a pixel as *skin*, if it meets the following criteria:

$$\log Val_1 \in [3.5, \infty); \quad \log Val_2 \in [0.05, 0.8]; \quad \log Val_3 \in [-1.25, 0] \quad (2)$$

We set these intervals by sampling 32000 pixels of both *skin* and *non-skin* classes and searching for the optimal thresholds that separate the classes. Figure 2 shows an example of the typical segmentations obtained with this scheme.

Given that our goal is to provide human detection capabilities to a social robot, we focus in detecting people that is standing and facing the robot. Also, due to the biological constraint that humans have their heads in the upper part of their body, we just search for skin pixels in the upper half of the stereo blobs. Here, we find the image row r_{max} and the image column c_{max} with maximum number of skin pixels. If the pixel located at (r_{max}, c_{max}) is a skin pixel, a Breadth First Search region growing algorithm starting on that point finds a *skin-colored blob* and its bounding box; if is not, then a search for a skin pixel is performed over its neighbors. If one of the neighbors is a skin pixel, the region

growing algorithm is done with that neighbors as a starting point, but if none of the neighbors is a skin pixel, then the candidate is rejected.

Finally, over the skin colored blobs obtained, we apply a size based rejection test. Given that the size of the expected color blobs depends of the distance of a potential person from the camera, we use training data to find an adaptive rejection threshold. This threshold depends on the average gray intensity of the corresponding pixels in the disparity image. The following equation shows the relation found between minimum blob area (*minArea*) and average gray intensity (*dispAvg*):

$$\text{minArea} = \frac{1.672 \cdot \text{dispAvg}^2 - 340.3 \cdot \text{dispAvg} + 19310}{2} \quad (3)$$

The numerator in Eq. 3 is obtained by sampling the area of 66 skin-colored blobs known to be real faces versus the average gray intensity of their containing stereo blobs, and fitting the best second-degree curve on the samples. Then, as to ensure that most of the samples are higher than the curve, we set a conservative tolerance for the threshold of 50% below the curve (denominator).

3.3 Feature Extraction

The skin segmentation shown in section 3.2 can yield several types of candidates. For example, for two people very close to each other and producing one blob, we will extract the two faces, but for a person who is waving, we will extract its face and its hand. This originates the need to differentiate between these candidate face blobs. To accomplish this, we extract color features from the color blobs to perform a classification between *face* and *non-face*.

In the computer vision literature there is a large amount of work about different features that can be extracted from a segmented area. We try several of them like Hu moments [7] and Flusser moments [5]. None of these features yielded acceptable results, mainly because skin-colored blobs have a very noisy and non-uniform shape, thus, they did not separate the *face* and *non-face* classes well. Finally, the feature set that provide us the best results is summarized in 3 criteria, that, applied to each RGB channel makes a total of 9 features:

- Normalized Standard Deviation

$$\sigma_{norm}^k = \frac{\sqrt{\frac{1}{N_s} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} (x_{ij}^k - \bar{x}^k)^2}}{\bar{Y}^2}$$

- Normalized Contrast

$$I_{norm}^k = \frac{\sum_{i=1}^8 \sum_{j=1}^8 (i-j)^2 \cdot P^k[i, j]}{\bar{Y}^2}$$

- Normalized Uniformity or Energy

$$E_{norm}^k = \sum_{i=1}^8 \sum_{j=1}^8 (P^k[i, j])^2 \cdot \bar{Y}^2$$

where N_i , N_j and N_s are the number of rows, columns and skin pixels of the skin image's bounding box, $k \in \{R, G, B\}$, x_{ij}^k is the value at position (i, j) of the color channel, \bar{x}^k is the mean over all x_{ij}^k on each color channel, $P^k[i, j]$ is the value of the 8x8 co-occurrence matrix [2] at position (i, j) for each color channel, and \bar{Y} is the mean luma component (Y) value of the YCbCr transformation of the skin-colored blob pixels. This value is defined as:

$$\bar{Y} = \frac{1}{N_s} \sum_{(i,j) \in \text{skinBlob}} 0.299 * R_{(i,j)} + 0.587 * G_{(i,j)} + 0.114 * B_{(i,j)}$$

Note that all these features are chromatic, hence, they overcome the problem introduced by the very noisy and non-uniform shape of the skin blobs.

σ_{norm}^k deals with the fact that in faces, pixel colors have great variability, because of the presence of hair, eyes, nose, and mouth. In hands, the pixel colors are, generally, less variable. In addition, the standard deviation penalizes the high values the variance takes with very illuminated images, in order to make this feature less responsive to that illumination.

I_{norm}^k and E_{norm}^k deals with the presence of hair, eyes, nose, and mouth in faces by taken into account contrast and uniformity. In these face areas, changes in pixel values tend to increase the contrast and decrease the uniformity. In contrast, the less variable nature of hands pixels tend to decrease the contrast and increase uniformity.

The mean squared luma value \bar{Y}^2 is intended to be a normalization value that balances the difference between very illuminated and obscure pictures.

To further improve the extraction of features, we perform an exhaustive feature selection process over all possible subsets of the 9 features. This process receives a feature set and evaluates the average number of blobs that are correctly classified, according to the classification criteria and training set described in section 3.4. This evaluation is performed using a 10-fold cross validation over the entire training data. The results of the best 3 set of features are summarized in Table 1. In our final system we use the set $\{\sigma_{norm}^R, I_{norm}^R, I_{norm}^G\}$.

Table 1. Feature selection results on training data

Feature Set	Average Correctly Classified Points
$\{\sigma_{norm}^R, I_{norm}^R, I_{norm}^G\}$	93.4%
$\{\sigma_{norm}^R\}$	91.1%
$\{\sigma_{norm}^R, \sigma_{norm}^B\}$	84.1%

3.4 Classification

Our approach classifies between *person* and *non-person* classes. The *person* class corresponds to all the skin-colored blobs found to be faces and the *non-person* class correspond to all other skin-colored blobs.

The classifier uses Gaussian Mixture Models (GMMs) to learn each class. These GMMs are trained using the MATLAB toolbox described in [1] with 1000



Fig. 3. Example of the operation of the system in an indoor office environment

examples for each class, taking care of having sufficiently different examples, that is, a “face” training set with different skin tonalities and a “non-face” training set representative of our intended scenario. The GMMs are normalized to the $[0,1000]$ range. Every skin color blob is classified according to the likelihood ratio test between the *person* and *non-person* models, using a null threshold.

4 Results

4.1 Person Detection

The first experiment is oriented to measure the performance of the system under two illumination conditions and different distances between the camera and the people being detected.

The test data consists of 100 frames with two people each, corresponding to 20 frames at each of 5 different distances. Experimentally, the maximum distance at which the stereo algorithm begins to capture a person blob is approximately between 360 and 310 cms, and the minimum distance is approximately 130 cms. We test the algorithm in a distance range from 150 to 350 cms. Figure 3 shows an example of the system operating in the office building environment.

The percentage of the people correctly detected as person are shown in figure 4. It is possible to see that, for the corridor location, the system performs better if the people are not too far or too close from the camera. For the office location with good illumination, the algorithm has performances that exceed 95%, except for the case of 250 cms, where a reflex in one of the faces made the feature values lie in a region where both person and no-person likelihoods are very similar.

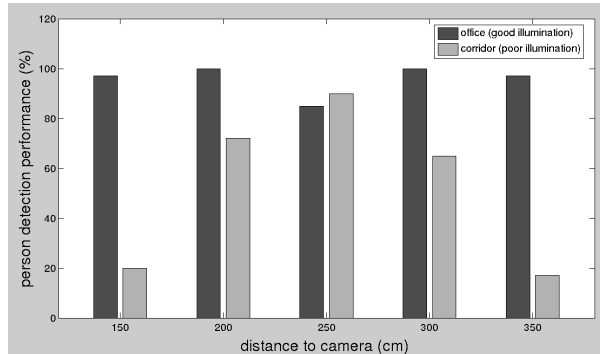


Fig. 4. Detection performance for people at different distances in two environments. Dark bars show performance in a well lighted room. Light bars show performance in a poor lighted corridor.

4.2 GMM Classifier

The second experiment is oriented to illustrate the robustness of the face classifier independently of problems in the initial stereo based segmentation. Therefore the setting of this experiment is similar to the previous one, but it only considers the frames where the stereo algorithm manages to segment the people correctly and at least one face candidate appears inside any of the blobs.

The test data consists of 200 frames with 1 of 5 people appearing in each. This corresponds to 10 frames for each people, acquired at 4 different distances. For this experiment, we consider distances from 150 to 300 cms and the results are shown in figure 5. It can be seen that a good performance (over 90%) is achieved, except for people at 300 cms in the office location. This is due to a moderate amount of false positives (12%). The result shows the robustness of the color based face detector to changes in illumination.

4.3 Real Video

To test our system mounted on a robot under real time operation, we run the system while the robot navigates in a indoor environment. The average robot velocity is around 0.5m/s and the system frame rate is around 2.5 fps. Figure 6 shows a map of the environment and the trajectory followed by the robot.

During its trajectory, the robot encounters 15 people. Table 2 summarizes the test results. One person was not detected because during several consecutive frames appeared standing at a distance that exceeds the detection range of the system. Once this person walked towards the robot, the poor frame rate made the system not to capture an image of this person. The two false positives are due to the arm of a person close to a wooden furniture and with clothes with a similar color to skin.

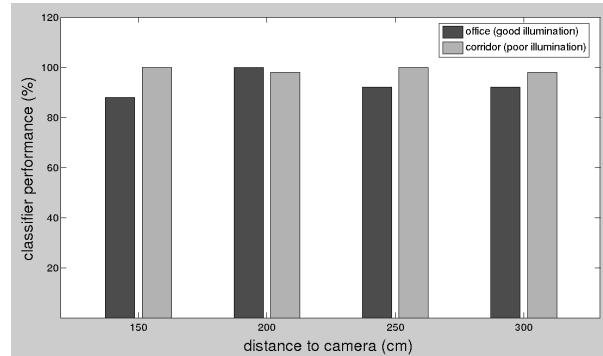


Fig. 5. Classifier performance for skin blobs of people located at different distances in two environments. Dark bars show performance in a well lighted room. Light bars show performance in a poor lighted corridor.

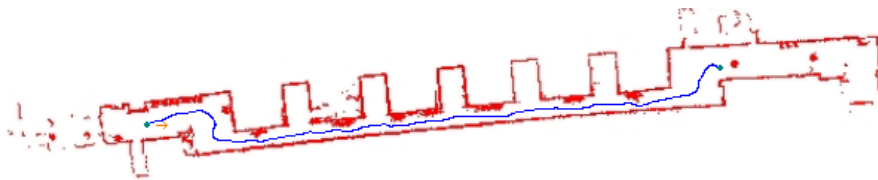


Fig. 6. Robot trajectory in our office environment. The map was obtained with our robot.

Table 2. System performance in the real video test. In the test, the robot meet 15 people.

People Correctly Detected	14
Missed People	1
False Positives	2

5 Conclusions

We have presented a person detection system mounted a moving platform, that is able to operate in real time in indoor buildings. Promising results have been obtained for this system with performance over 90%. The inclusion of a robust skin pixel criteria and an illumination-invariant set of color features are also important contributions of this work.

The results indicate that the stereo vision based segmentation process is vulnerable to changes in illumination conditions and distance from the camera. The color based face segmentation process presents a better result respect to changes in illumination and some vulnerability respect to distance. In the case of our application distance to the camera is not a relevant issue because we have a mobile

platform, however, we still need to further explore the illumination problems of the stereo system.

As future area of research, we can mention: the addition of a pan-tilt mechanism to add target tracking capabilities, the inclusion of probabilistic priors to improve the classification results, and finally, the need to increase the processing frame rate of our system to be able to deal with more crowded scenarios.

Acknowledgments

This work was partially funded by FONDECYT grant 1070760 and CONICYT project ACT-32. We would like to thank Domingo Mery for the valuable comments.

References

1. Baggenstoss, P.M.: Statistical modeling using gaussian mixtures and HMMs with MATLAB. Naval Undersea Warfare Center, Newport, RI (2002), <http://www.npt.nuwc.navy.mil/Csf/htmldoc/pdf>
2. Castleman, K.R.: Digital image processing. Prentice-Hall, Englewood Cliffs (1996)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–893. IEEE Computer Society Press, Los Alamitos (2005)
4. Darrell, T., Gordon, G., Harville, M., Woodfill, J.: Integrated person tracking using stereo, color, and pattern detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 601–608. IEEE Computer Society Press, Los Alamitos (1998)
5. Flusser, J., Suk, T.: Pattern recognition by affine moment invariants. *Pattern Recognition* 26(1), 167–174 (1993)
6. Fong, T., Nourbakhsh, I., Dautenhahn, K.: A survey of socially interactive robots. *Robotics and Autonomous Systems*, Special issue on Socially Interactive Robots 42(3-4), 143–166 (2003)
7. Hu, M.K.: Visual pattern recognition by moment invariants. *IRE Trans. Info. Theory* IT(8), 179–187 (1962)
8. Konolige, K.: Small vision system: Hardware and implementation. In: Eighth Symposium on Robotics Research, pp. 111–116 (1997)
9. Ogale, N.A.: A survey of techniques for human detection from video. Master's thesis, University of Maryland (May 2006)
10. Sidenbladh, H.: Detecting human motion with support vector machines. In: ICPR 2004, vol. 2, pp. 188–191 (2004)
11. Utsumi, A., Tetsutani, N.: Human detection using geometrical pixel value structures. In: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, p. 39. IEEE Computer Society Press, Los Alamitos (2002)
12. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, pp. 734–741. IEEE Computer Society Press, Los Alamitos (2003)