Tracking and Pursuing Persons with a Mobile Robot

Stefan Feyrer and Andreas Zell
Wilhelm-Schickard-Institute, Department of Computer Architecture
University of Tübingen
Köstlinstr. 6, 72074 Tübingen, Germany
{feyrer, zell}@informatik.uni-tuebingen.de

Abstract

In this paper we address the problem of detecting, tracking, and pursuing persons with a mobile robot in real-time. Our system consists of two cooperatively interacting parts: A visual gaze holding process using an active stereo vision system and a component controlling the trajectory of the robot. The detection process is based on an approach which combines color, motion, and contour information. Additionally, facial features and depth information are integrated to increase the robustness of detection. For navigation a potential field method is used taking into account the path of the person going ahead as well as a local environment map built by means of laser range sensor data. Consequently, stationary and dynamic obstacles can be avoided during the process of pursuit. The system is able to operate at approximately human walking rates.

1. Introduction

There is a large variety of possible application scenarios for mobile service robots. However, all these scenarios have one thing in common: They all require a man-machine interface that allows the user to communicate and to interact with the robot in an easy and natural way. Recognition and generation of natural speech as well as the visual perception of humans and human action are basic tasks in this area.

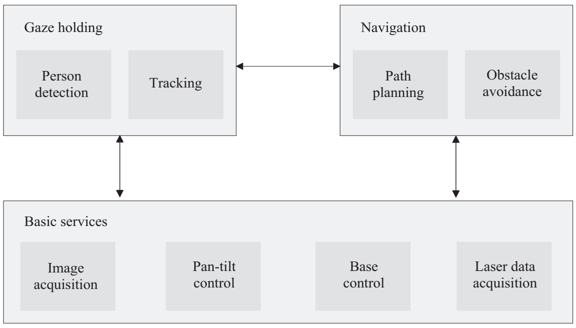
It is important for a mobile service robot to perform the visual perception not only in a passive manner, but also to integrate it actively in the current overall task. In our system the visual detection and tracking process is combined with a navigation task aiming to pursue the person currently focused by the vision system. Potential applications of this combination are e.g. the guidance of the robot by a human user to a certain operational area or to get the robot familiar with a previously unknown environment.

When using a mobile platform, many problems – especially motion detection – become substantially different and often more difficult in comparison with stationary systems. Furthermore, a number of constraints have to be taken into account. First, the system has to be highly robust with regard to scene structure and changing illumination. Second, it is important that the processing speed allows a natural reactive behaviour. This constraint becomes even more restrictive when we consider the limited processing power of a mobile robot. For this reason, the use of less general and sophisticated but more simple and efficient algorithms becomes necessary. In addition, to achieve real-time performance, image resolution has to be reduced respectively.

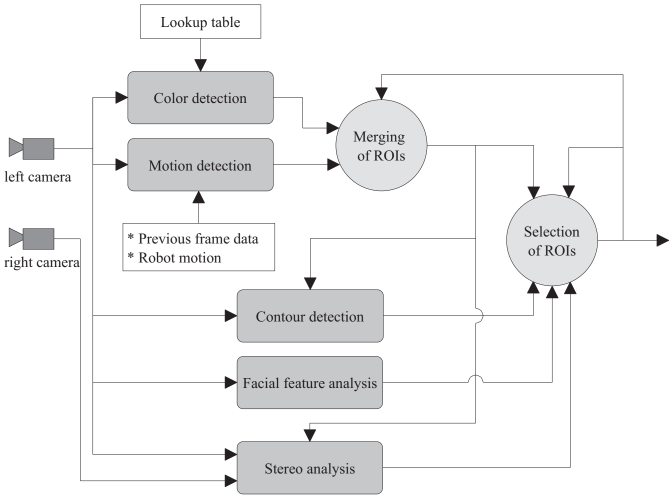
2. System architecture

The system consists of two main components, both based on a set of basic services providing low-level access to image and laser sensor data as well as to the actuators of the robot (Fig. 1a). The first component comprises the complete visual gaze holding process, i.e. face detection and pan-tilt control. The second module is responsible for path planning and obstacle avoidance. The modules are realized as two separate processes in order to increase the speed of the gaze holding loop. Coordination of the two components is realized via a fast interprocess communication protocol.

Many person and face detection systems are based on one single cue at the detection level, mainly motion [10], [12] or color [4], [8]. The claim for systems working in natural and complex environments, however, has effected a shift to systems using a combination of multiple cues to segment persons from the background [2], [11]. In our approach we also combine several basic cues. Fig. 1b shows the structure of the face detection component. Two main sub-components are the extraction of skin-colored areas and the determination of image regions containing significant object movements in the scene. A fusion algorithm combines related ROIs (regions of interest) that are erroneously detected as separate blobs. Besides geometrical and structural cues from color and motion detection, contour and depth information as well as information about facial



(a)



(b)

Figure 1. (a) System overview. (b) Structure of the detection component.

features is used in the following selection process, where all cues are combined according to a probabilistic framework.

3. Detection of persons

In the past decade a considerable amount of work has been done in the field of visual detection of humans and human motion (see [1] for an overview) and a wide variety of approaches have been proposed. The combination of cues we use in our system has been chosen from the aspect of a complex real-world application scenario, real-time constraints and the limited processing power of our robot.

3.1. Detection of skin-colored regions

Skin-colored image pixels are identified using the HSI color model. Since only pure color information is necessary for classification, we ignore the intensity values. Within the resulting two-dimensional HS-subspace skin color values are defined by means of a lookup table which we have derived from manual segmentation of more than 50 image sequences recorded with 15 people under varying lighting conditions. To avoid the time-consuming calculation of the HSI values, the lookup table is transformed back to the RGB color space. To deal with changing illumination, we apply a simple linear color constancy algorithm before we perform the classification step.

From the clustered skin-color regions we extract the size, the height-width ratio of the bounding box, and the skin color probability of the blob. These values are passed to the selection module.

3.2. Motion detection

The problem of motion detection with a non-stationary camera is to distinguish between the stationary background and objects moving in the scene. Knowing the camera motion parameters and the depth value of each image pixel, compensation for ego-motion is possible. The quality of the results, however, strongly depends on the temporal exactness of the camera motion information, which is getting really poor for high camera speeds. Therefore, we compensate only for the relatively slow and well determinable motion of the robot base but not for the motion of the pan-tilt unit, which can achieve speeds of up to 300°/s. Motion detection itself is done by subtracting two successive frames grabbed with temporarily stationary camera with respect to the robot. This approach requires a saccadic camera motion but reduces the false detection rate considerably. A thresholding and clustering algorithm yields a list of image areas in which significant grey level changes have occurred. Information in terms of motion parameters of the objects moving in the scene is not derived.

Compensation for ego-motion is performed according to [7]. First, the position and orientation change of the camera between two successive frames is computed in terms of a homogeneous transformation matrix T with rotation parameters r_{ij} and translation parameters t_k. We can easily derive this matrix since robot motion is controlled by the system itself and therefore known by the program. Then, according to (1) a predicted image is calculated and matched with the actually grabbed image.

$$x_2 = f \cdot \frac{r_{11}x_1 + r_{12}y_1 + fr_{13} + f\frac{t_x}{Z}}{r_{31}x_1 + r_{32}y_1 + fr_{33} + f\frac{t_z}{Z}}$$
$$y_2 = f \cdot \frac{r_{21}x_1 + r_{22}y_1 + fr_{23} + f\frac{t_y}{Z}}{r_{31}x_1 + r_{32}y_1 + fr_{33} + f\frac{t_z}{Z}} \quad (1)$$

x_1 and y_1 are the image coordinates of the original image, x_2 and y_2 refer to the predicted image. f denotes the focal length of the camera lens. The value Z, which denotes the distance between the camera and the objects in the scene for each pixel, is derived by the stereo analysis. Calculation of distance values for all pixels increases the computational complexity and therefore requires a reduction of image resolution at this point.

3.3. Integration of contour information

For determining the boundaries of the head of a person we use an algorithm that follows the method of *active rays* [3]. Starting at a reference point within the object boundaries, which is provided here by the color segmentation module, grey levels are examined along straight lines passing through this point. An energy minimization yields a chain of contour points approximating the object boundaries. Since this method can only deal with convex objects, we have to extract the shoulder lines separately. Subsequently, the extracted contour is normalized in size and compared with a statistically determined average head-shoulder contour.

3.4. Detecting facial features

Several approaches to feature-based face localization have been described [9], [13]. For simplicity reasons we use a fast and simple method, which determines the position of the eyes and the mouth by searching for grey-level minima within the resolution-reduced ROI provided by the color detection component. To achieve independence of the distance between the person and the robot, the resolution within the considered ROI is set to a fixed value. Minima lying within the ROI but outside the skin-colored area are rejected. Thus, the influence of dark hair, clothes, or background objects is reduced. Once the minima have been determined, a set of geometric constraints are applied taking into account the relative position of the features and their absolute position with respect to the bounding box of the area. This yields a confidence value describing the likelihood of a ROI for containing a face in frontal view.

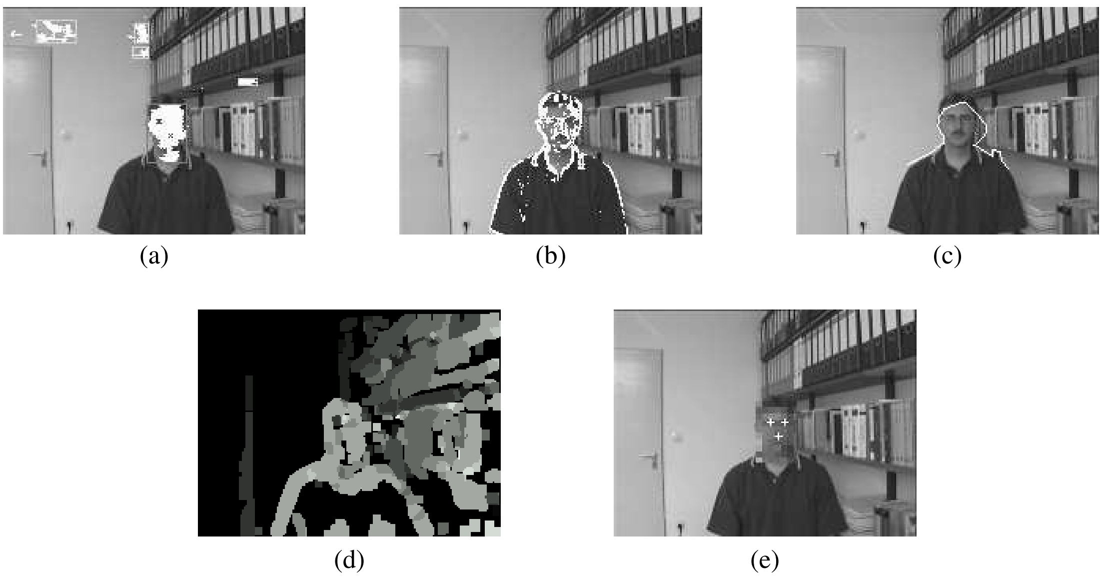


Figure 2. Results of the different detection steps. (a) Color detection. (b) Motion detection. (c) Contour detection. (d) Depth analysis. (e) Facial feature detection.

In order not to reject ROIs containing slightly rotated faces or profile views of heads, the evaluation function of the selection process is influenced by the facial feature module only in case of a face being detected in frontal view.

3.5. Stereo analysis

Our stereo algorithm is based on a standard stereo geometry. Because of the high accelerations of the pan-tilt unit it is necessary to have a fast working auto-calibration algorithm which recalibrates the external camera parameters in certain time intervals automatically.

To calibrate the stereo system we first extract feature points by means of the Moravec feature detector. The pixel offsets determined by analyzing correspondences are approximated by a linear regression line which is then used for the subsequent correlation analysis. Thus, vertical offset errors and rotational errors with respect to the optical axes can be compensated.

There are a number of different techniques for correlation analysis. For performance reasons we have chosen a method which correlates normalized grey level areas in the following way [5]:

$$s = \max(0, 1 - c), \tag{2}$$

where

$$c = \frac{\sum_{i,j}((L(x+i, y+j) - \bar{L}) - (R(x+dx+i, y+dy+j) - \bar{R}))^2}{\sqrt{(\sum_{i,j}(L(x+i, y+j) - \bar{L})^2) \cdot (\sum_{i,j}(R(x+dx+i, y+dy+j) - \bar{R})^2)}} \tag{3}$$

L and R denote the left resp. the right image matrix, $\bar{L}$ and $\bar{R}$ are the average grey level values of the respective correlation window.

By restricting the search to an interval heuristically depending on the disparity of the previous pixel and by dropping areas with insufficient grey level variance we further reduce computation time.

3.6. Merging and selection of ROIs

Often faces in the image are covered by more than one extracted ROI. Especially the eye line often separates two ROIs. Therefore a fusion algorithm is applied, which traverses a neighbourhood graph of ROIs. Each combination of ROIs is evaluated according to the selection criteria. Those combinations gaining the highest rating are merged.

The final selection of the winner ROI is done by joining the different evaluation results. Therefore a confidence factor p is calculated which represents the likelihood of a ROI for containing a face:

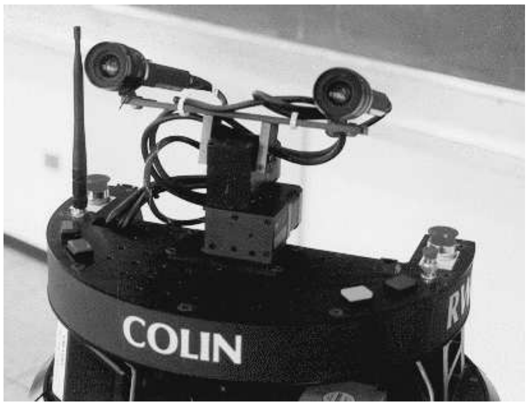


Figure 3. Camera platform with pan-tilt-unit.

$$p = e^{-\frac{1}{2}(X - \mu)C^{-1}(X - \mu)} \tag{4}$$

Elements of the feature vector X are the width-height ratio of the ROI, the skin-color probability, the change in size, the change in depth and the contour matching value. Except for the size and depth change all elements are statistically independent, so that the covariance matrix reduces to a block diagonal matrix. The mean values μ_i and the covariance coefficients c_{ij} have been calculated using a sample set of correct detections. The influence of the size and depth change rate depends on the confidence value of the winner ROI in the previous frame. Thus, the adhesive effect of erroneously selected ROIs is reduced.

Humans are not always in motion. Furthermore we cannot expect that a person faces the cameras throughout the tracking process. Therefore the confidence factor must not be reduced if no motion or no frontally oriented face could be detected. For this reason the results of the motion and facial feature analysis are not integrated via the probabilistic framework described above but by non-negative additive values.

4. Tracking and navigation

The camera platform (Fig. 3a) has two degrees of freedom described by a pan and a tilt angle. As already mentioned, we realized a saccadic gaze control based on simple linear equations to keep the detected face centered in the image. A sequence of images grabbed during tracking is shown in Fig. 3b.

For navigation we use a potential field method [6]. It works on an artificial potential field U, which is defined as the sum of an attractive potential U_{attr} pulling the robot towards a goal position and a repulsive potential U_{rep} push-

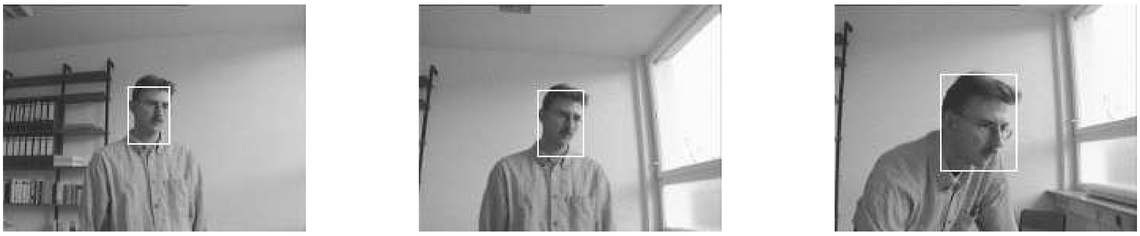


Figure 4. Sequence of images during tracking.

ing the robot away from the obstacles. Since we have no global obstacle map, the distance values ρ_k provided by the range sensors are directly included in the calculation of the repulsive potential:

$$U_{rep} = \sum_{k=1}^{r} U_k$$

$$U_k = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho_k} - \frac{1}{\rho_0}\right)^2 \sin\varphi_k & \text{for} \quad \rho_k \le \rho_0 \\ 0 & \text{else} \end{cases} \tag{5}$$

$$U_{attr} = \frac{1}{2}\xi\rho_{goal}^2 \tag{6}$$

ρ_0 denotes the distance of obstacle influence, ρ_{goal} is the current distance to the goal. η and ξ are scaling factors. The resulting direction of motion is determined by an artificial force induced by the potential field. To avoid dead end situations the path positions of the person are remembered and the goal position is determined dynamically. Thus, in case of the current position not being reachable, the remembered path positions can be brought in to determine a useful goal.

5. Experimental results

The described algorithms run on a RWI B21 mobile robot (Fig. 5). The robot is equipped with two dualprocessor 200MHz PentiumPro PCs. Image acquisition is done by two CCD color cameras mounted on a pan-tilt-unit. Distance data is acquired via the built-in 2D laser scanner.

We achieve a processing frequency of 8 frames per second at an image resolution of 192x144 pixels in case of a stationary robot base. When the robot is moving, the performance decreases to 6 frames per second due to the time-consuming compensation for ego-motion. Nevertheless even during motion, tracking and pursuit of humans is possible as long as they are not moving unusually quickly. The velocity varies up to a maximum of 0.5 m/s depending on the environment and on the distance to the person.

The performance of face detection in terms of correct detection rates strongly depends on scene structure and scene dynamics. Analysis of several image sequences recorded during tracking showed that we get higher recognition rates when we add motion analysis to simple color detection. Integration of contour and face structure information leads to a relatively modest improvement of detection performance. Tab. 1 exemplarily shows the rates of correctly detected heads vs. other segmentation clusters like e.g hands for one selected image sequence.

To illustrate the behaviour of pursuit, the path of a person walking backwards along marking lines on the floor and the path of the pursuing robot are depicted in Fig. 6.

Figure 5. The mobile service robots Robin and Colin.

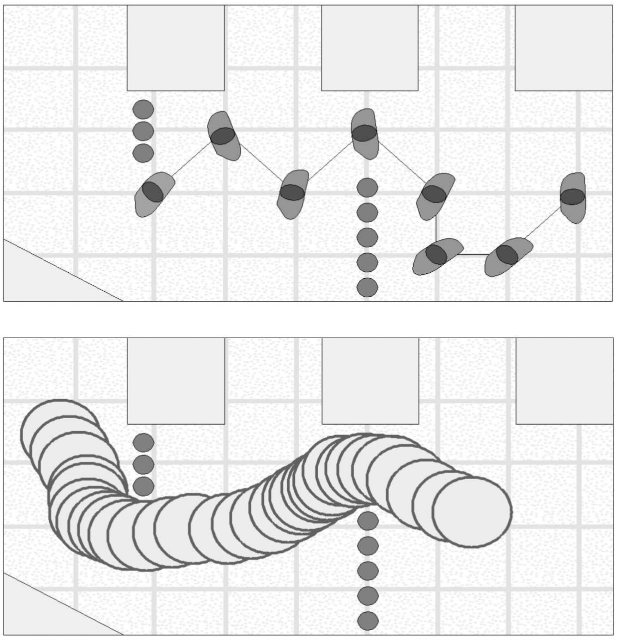


Figure 6. Path of a person and path of the pursuing robot.

The path of the robot is drawn manually from trace data recorded on the real robot.

Table 1. Head recognition rates with respect to a selected sample image sequence.

	Contour/feature detection off	Contour/feature detection on
Motion detection off	92.5%	95.4%
Motion detection on	97.9%	98.9%

6. Conclusion

We have presented a mobile robot system which is able to follow a person in real-time. At the detection level several cues are combined to achieve a sufficient level of robustness. The active vision unit cooperates with a navigation component, which is responsible for obstacle avoidance and the pursuing task. Experiments have shown that the system successfully handles even difficult situations like highly dynamic environments as well as narrow doorways.

The main limitation of the current system is that the person has to face the robot constantly and therefore has to walk backwards. To overcome this restriction future work will be on the realization of a more flexible detection module, which, after an initial phase, will then be able to track the person even when the face has been turned away.

7. References

[1] J. K. Aggarwal and Q. Cai, ”Human Motion Analysis: A Review,“ *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428-440, 1999.

[2] J. L. Crowley, F. Berard, ”Multi-Modal Tracking of Faces for Video Communications,“ *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 640-645, 1997.

[3] J. Denzler and H. Niemann, ”Active Rays: A New Approach to Contour Tracking,“, *Int. Journal of Computing and Information Technology*, vol. 4, no. 1, pp. 9-16, 1996.

[4] P. Fieguth and D. Terzopoulos, ”Color-Based Tracking of Heads and Other Mobile Objects at Stereo Frame Rates,“ *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 21-27, 1997.

[5] P. Fua, ”A Parallel Stereo Algorithm That Produces Dense Depth Maps and Preserves Image Features,“ *Machine Vision and Applications*, vol. 6, no. 1, pp. 35-49, 1993.

[6] O. Khatib, ”Real-Time Obstacle Avoidance for Manipulators and Mobile Robots,“ *Int. Journal of Robotics Research*, vol. 5, no. 1, pp. 90-98, 1986.

[7] D. Murray and A. Basu, ”Motion Tracking with an Active Camera,“ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 449-459, 1994.

[8] Y. Raja, S. J. McKenna, and S. Gong, ”Tracking and Segmenting People in Varying Lighting Conditions,“ *Proc. 3rd Int. Conf. on Automatic Face and Gesture Recognition*, pp. 228-233, 1998.

[9] D. Reisfeld and Y. Yeshurun, ”Preprocessing of Face Images: Detection of Features and Pose Normalization,“ *Computer Vision and Image Understanding*, vol. 71, no. 3, pp. 413-430, 1998.

[10] K. Rohr, ”Towards Model-Based Recognition of Human Movements in Image Sequences,“ *CVGIP: Image Understanding*, vol. 59, no. 1, pp. 94-115, 1994.

[11] C. Schlegel, J. Illmann, H. Jaberg, M. Schuster, and R. Wörz, ”Vision Based Person Tracking with a Mobile Robot,“ *Proc. 9th British Machine Vision Conference*, pp. 418-427, 1998.

[12] A. Shio and J. Sklansky, ”Segmentation of People in Motion,“ *Proc. IEEE Workshop on Visual Motion*, pp. 325-332, 1991.

[13] K. C. Yow and R. Cipolla, ”Feature-based Human Face Detection,“ *Image and Vision Computing*, vol. 15, pp. 713-735, 1997.