

Vision Based Person Tracking with a Mobile Robot*

Christian Schlegel, Jörg Illmann, Heiko Jaberg,
Matthias Schuster, Robert Wörz
Research Institute for Applied Knowledge Processing (FAW)
PO-Box 2060, D - 89010 Ulm, Germany
<last-name>@faw.uni-ulm.de

Abstract

We address the problem of detecting and tracking people with a mobile robot. The need for following a person with a mobile robot arises in many different service robotic applications. The main problems of this task are real-time-constraints, a changing background, varying illumination conditions and a non-rigid shape of the person to be tracked. The presented system has been tested extensively on a mobile robot in our everyday office environment.

1 Introduction

In this paper we address the problem of detecting and tracking people in natural environments in real-time. The person to be followed introduces itself to the system during a startup step. Without any predefined model of a person, the system builds an internal representation of the person which then enables it to track and follow the person in real-time even in dynamic environments. While following the person the representations are continuously adapted to account for changing illumination conditions and varying shapes of the person due to the non-rigidness of their body. The system is used to enhance the man-machine-interface of an autonomous mobile robot and to provide basic mechanisms to build applications in the field of service robotics, e.g. guiding persons through office buildings. These applications impose certain requirements on the implemented methods which include the use of moving cameras mounted on moving platforms, tracking of people differing in their looks and real-time operation in a changing environment.

The main contribution of our method is the integration of a fast color-blob based approach with a sophisticated but computational expensive contour-based approach. By restricting the contour matching to color-based preselections, real-time performance can be achieved, which in these applications requires to provide new heading and distance information of the tracked person several times per second. The internal representation



Figure 1: Startup procedure

*This work is part of project C3 (<http://www.uni-ulm.de/SMART/Projects/c3.e.html>) of the SFB 527 sponsored by the Deutsche Forschungsgemeinschaft

generated during the startup phase consists of the color distribution and the contour of the person. The color model is used to segment sequencing images. While the color-based method provides rough candidate regions at a high rate, these are further investigated by the edge-based approach by matching the contour model against the candidate regions. During each cycle the internal models are updated to cope with contour variations due to the non-rigidness and rotations of the person's body and to account for changing illumination conditions.

The approach was tested in a natural indoor environment on a real robot. The results show that by combining different representations, it is possible to track and follow persons in real-time in natural, dynamic environments using off-the-shelf hardware. Using a combination of two different representations our system becomes more robust and faster compared to previous methods.

The remainder of this paper is organized as follows. After discussing some previous work conducted in this area in section 2, in section 3 the approach we use is proposed. In section 4 we describe the hardware used to implement our experimental system and the implementation of the algorithm on our robot. Section 5 presents our experimental results, followed by a discussion of further research issues.

2 Related Work

Much work in the field of person detection and tracking is based on vision systems using stationary cameras [10]. Most of them originate from virtual reality applications where one person moves in front of a stationary background. Moving objects are detected by subtracting two frames [9, 4]. These approaches often require that the tracked person is never occluded by other objects [1]. Therefore they are typically not suitable for person following on a mobile robot since at the same time the robot tracks the person it has to move into the person's direction to stay close enough.

Many approaches can be classified regarding the kind of models that are used. Complex models consider a 3D model of the body and even take into account kinematic constraints but require three static cameras [6]. Simpler 2D models are used in [9] and [10]. The basic idea is to use deformable models to represent the boundary of the segmented body. Even those approaches assume one or more static cameras. In [8] spatio-temporal patterns resulting from the regular motion of a walking person are used. This approach requires a steadily moving person which is an unwanted restriction for our application. In [2] an approach is described, where the shape model is derived from a set of training shapes, of which each shape is represented by a parameterised cubic B-spline. In use shape parameters are estimated by using a Kalman filter.

Several approaches use color-based image segmentation for object-ground separation. In [1] a person is represented by a collection of colored blobs. Each blob is described by a 3d-normal-distribution in YUV-color space. Since the background is also modelled by a normal distribution, a simple color classification is sufficient for object-ground separation. While the color-blob representation of the person is quite useful, they require a smoothly changing image background which can not be guaranteed on a moving platform. In [3] a similar approach is presented which dynamically adapts color templates to cope with changing illumination conditions. Another color-based system which is able to track colored blobs in real-time is presented in [11]. It is implemented on a mobile robot which

locates a person in dynamic environments but requires to wear a uni-colored shirt with one of four predefined colors. Since their approach does not rely on static environments and is quite robust and may be run in real-time, it is very promising for our purposes. Drawbacks are that the approach is not adaptive to illumination changes and since it relies only on the color and does not consider any shape, it requires that the person can be uniquely identified only by its color. Therefore the person's color may not occur in the background.

A model-based approach for object tracking using two-dimensional geometrical models can be found in [7]. A binary contour image resulting from the application of a Canny edge filter is matched against two-dimensional binary contour model using the Hausdorff distance. A successive match leads to an update of the contour model by a logical combination of the dilated current model with the current edge image. Thus this approach adapts to continuous deformation of the person's image during movement and overcomes the difficulties, which person tracking imposes on rigid model-based approaches. The main drawback of this approach is its computational complexity. Without any restrictions in the search space it seems not to be suitable for real-time application.

3 Color- and contour-based person tracking

Because the difficulties imposed by person tracking cannot be overcome easily with a single approach, we propose a combination of a fast color-based with a robust contour-based approach. The color-based part provides regions of interest at which to look for the person by matching extracted contours against a contour model. Our internal representations are based on color histograms and adaptive contour models to be able to track three-dimensional, non-rigid objects which may have a non-uniform color and texture distribution. After discussing two color-based approaches, we present the combination with the contour-based object detection.

3.1 Color blob tracking with a modified NCC-Method

We first extended the approach in [11] to allow tracking of any uni-colored shirt after presentation during a startup phase. Additionally an adaptive filter to compensate for changing illumination conditions is added.

3.1.1 Specifying a target color

At startup the person introduces itself to the robot as shown in figure 1. As a first step the intensity information is removed from the color values.

$$\hat{r} = \left\lfloor \alpha \frac{R}{R + G + B} \right\rfloor, \hat{g} = \left\lfloor \alpha \frac{G}{R + G + B} \right\rfloor \quad (1)$$

The resulting image is scaled by α and discretized to the interval $[0, \dots, 255]$ to form the normalized color components (NCC). The NCC values of each pixel within a region of interest (ROI) are measured (fig. 2). Then the thresholds (r_l, r_h, g_l, g_h) are determined based on the density distribution of the ROI. The distribution resembles the shape of a gaussian function. Under the assumption of a normal distribution of the color values, we determine the thresholds by computing the mean μ and the variance σ of each color using



Figure 2: Region of interest (ROI) for measuring the thresholds



Figure 3: Resultant binary image after color matching

the equations $r_{l/h} = \mu_r \mp \sigma_r$ and $g_{l/h} = \mu_g \mp \sigma_g$. These thresholds form a rectangular area in the NCC color space which now specifies the color distribution to be tracked.

3.1.2 Tracking

The following steps are repeated on every frame. First, for each pixel at location (x, y) in the image frame the NCC values $\hat{r}_{x,y}$ and $\hat{g}_{x,y}$ are computed and compared to the target color values using the following criteria:

$$p_{x,y} = \begin{cases} 1, & \text{if } r_l < \hat{r}_{x,y} < r_h \text{ and } g_l < \hat{g}_{x,y} < g_h \\ 0, & \text{else} \end{cases} \quad (2)$$

The resulting binary image contains a 1 for all those pixel whose NCC values are within the rectangular area in the NCC color space defined by the startup step (fig. 3). Now a binary median filter is applied to delete noise and small false detections. Since the object to be tracked is dominant in the field of view, we choose the largest blob and compute its center of gravity which determines the position of the object.

3.1.3 Adaptive filter

Every k frames new color thresholds are computed to adapt to new illumination conditions. We use a simple adaptive filter $s_{t+1} = (1 - \alpha)\tilde{s} + \alpha s_t$ which computes the new thresholds at time step $t + 1$ from the old values s_t and the measured new values \tilde{s} with a weighting factor α . The pixels used for computing the new thresholds are determined by a mask generated from the actual binary image. A closing process is applied to this mask to close gaps. The resulting mask selects those pixel of the NCC image which contribute to the calculation of the new thresholds.

3.2 Color histogram based person tracking

One main restriction of the NCC-method is its requirement of uni-colored shirts. To overcome this problem we introduced color-histograms to describe the distribution of color-components. This color-distribution is characteristic for the color to be tracked. Color values which occur frequently in the object get a high value in the histogram. High histogram values indicate that the color value belongs to the object to a high degree.

3.2.1 Histogram extraction and backprojection

Based on an image representation in the NCC color space we compute a two dimensional histogram h_{2D} where $h_{2D}(\hat{r}, \hat{g})$ specifies the number of image pixels with the color value (\hat{r}, \hat{g}) . The resulting histogram is subsequently normalized by mapping the frequency values to the interval $[0, \dots, 255]$. Figure 4 shows such a normalized 2D-histogram computed from the color bands (\hat{r}, \hat{g}) . The extracted color histogram represents a color model which specifies how frequently certain colors occur within the person's image. To detect and locate the person in the image, the histogram values are backprojected on the image. The pixel value $p(x, y)$ in the backprojection image with the color value (\hat{r}, \hat{g}) is set to the histogram value of (\hat{r}, \hat{g}) : $p_{\hat{r}, \hat{g}} = h_{2D}(\hat{r}, \hat{g})$.

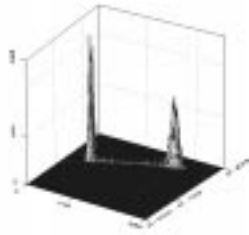


Figure 4: Normalized 2D-Histogram of color bands (\hat{r}, \hat{g})

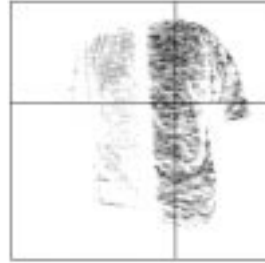


Figure 5: Backprojection image

The resulting backprojection image specifies the frequency that the image point $p(x, y)$ belongs to the tracked person. Subsequently pixels with low probability values are eliminated. The target position is estimated by a weighted center of gravity in the backprojection image. Figure 5 exhibits the backprojection computed from an image of a person wearing a two-colored shirt. The cross shows the center of gravity, which can be used to determine the direction where the tracked person is expected.

3.2.2 Systemoverview

An overview is given in figure 6. During the startup-phase the color histogram is extracted and the color parameters are estimated. In the tracking phase the histogram backprojection is used to estimate the target direction without explicitly segmenting the image.

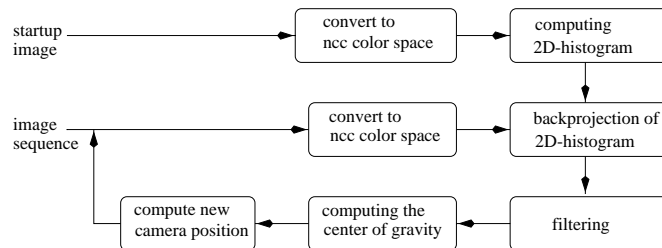


Figure 6: Color-histogram based approach

3.3 Contour-based approach

While the presented color-based approaches are fast and fairly robust against shape changes of the person, there are situations where using only color is not sufficient. Sometimes persons wear similar colored dresses or cannot be discriminated from the background because there are objects which have the same colors. Therefore we developed a method based on the approach in [7], that allows to track non-rigid objects. The model of an object consists of a number of edge pixels describing the contour.

3.3.1 Detection of the seeked object

In the first step the RGB-image is converted to a greylevel image which is fed into a Canny operator [5] to generate a binary edge image. Let I_t be the binary edge image taken at time step t and M_t be the model at time step t represented by a binary edge image. The model may undergo certain transformations $g(M_t)$. We allow just translations in the image space. Then the seeked object is detected by matching the current model $g(M_t)$ against the next image I_{t+1} . To estimate the similarity between model and edge image we use the generalized Hausdorff-distance [7] as a distance measure.

$$h_k(g(M_t), I_{t+1}) = K_{p \in M_t}^{th} \min_{q \in I_{t+1}} \|g(p) - q\| \quad (3)$$

Minimizing h_k over all transformations $g(\cdot)$ provides the translation of the model M_t which leads to the best match. Let g^* be the transformation which minimizes formula 3 and d be the minimal distance.

$$d = \min_{g \in G} h_k(g(M_t), I_{t+1}) \quad (4)$$

Descriptively this means that at least K points of the transformed model $g(M_t)$ lie at most at a distance d away from any point of the image I_t .

3.3.2 Contour Model Generation and Update

The initial contour model is generated by a startup step. All edges within a predefined rectangular area are defined to belong to the initial contour model of the presented person.

To be able to handle shape changes of a non-rigid object the model has to be updated. The new model M_{t+1} is built from the points of the image I_{t+1} whose distance to a point of the transformed model $g(M_t)$ does not exceed a threshold δ . The parameter δ controls which shape changes are allowed within one timestep.¹

$$M_{t+1} = \{q \in I_{t+1} \mid \min_{p \in M_t} \|g^*(p) - q\| \leq \delta\} \quad (5)$$

3.4 Combination of color-based and contour-based approach

Since the contour-based approach is computational too expensive, we combine it with the color-based one. The color-based approach is used for detecting regions where it is reasonable to apply the contour-based approach. Figure 7 depicts our concept. Depending on

¹This shouldn't be confused with the translation of the same shape within the field of view which is not restricted

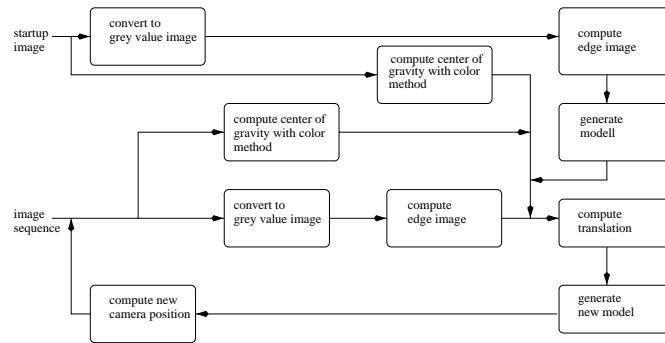


Figure 7: Combined color- and contour-based approach

the appearance of the person and the color distribution of the environment, the appropriate color-based approach is chosen. The quality of the obtained color model can be detected by the covariance of the calculated model. If the person's dress consists of multiple colors the color-histogram based approach is chosen, the color-blob method otherwise.



Figure 8: Mask image



Figure 9: Model update

To update the contour model, the result of the color segmentation is used to mask the edge image. A binary image representing the blob belonging to the detected person is first dilated (figure 8 shows the dilated blob image) and a rectangular region is added for the person's head. This considers the characteristic shape of the head and shoulder part when generating the initial model though the color distribution is generated only from the body. This binary mask is used to eliminate edge pixels belonging to the background. Figure 9 shows the updated model.

4 Integration on the robot

To implement the behaviour person-following on our mobile robot control commands for the robots actuators have to be generated. After detecting and locating the sought person within the image we use a very simple camera model to implement a reactive following-behaviour. The video cameras of our robot are mounted on a pan-tilt unit (ptu) which owes two degrees of freedom. The ptu is used to implement a gaze holding behaviour, i.e. steadily holding the person in the center of the image. To compute the control commands for the ptu, we assume a known focal length f and size of the CCD-Chip. A camera's

field of view γ is computed as $\gamma_x = 2 * \arctan(\frac{ccdsizex}{2*f})$ horizontally and vertically.

The ptu-angles are modified by $\alpha_x = \frac{x_B - x}{x_B} \gamma_x$ where x_B denotes the number of columns and x the estimated center of gravity of the person in the image. These directions are used as input for other modules which are responsible for generating collision free motions.

For holding a given distance between target and robot we use a method based on the disparity between two simultaneously shot images. The conjugate pair we use are the centers of gravity of the segmented color blobs in both images. Our model is shown in figure 4. The distance z can be computed to $z = \frac{b*f}{x_l - x_r}$. This is precise enough for a simple distance-estimation in order to decide whether the robot has to accelerate or decelerate. If the person is lost more than three times in sequence a stop command is sent to the motion control. After redetection the motion control gets new goal information.

The robustness of the method significantly depends on the startup illumination of the target. A startup illumination that is on average to the environment provides that the tracking method is rather robust against limited illumination fluctuations.

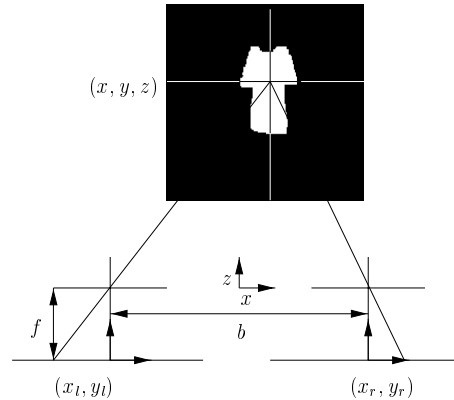


Figure 10: Distance calculation

5 Real-World Implementation / Experiments

Our approach has been implemented and tested extensively on image sequences taken by a mobile robot and also in our everyday office environment where the robot had to follow autonomously a person to different places. The robot is equipped with two color video cameras which have several automatic image acquisition features that were activated in our experiments. The focal length of the camera was set to 5.75 mm which corresponds to a field of view of horizontal 48.8 degrees by 37.6 degrees vertical. The size of the grabbed image was 768 by 576 pixels. These images were subsampled by factor 4 in both directions to speed up the processing resulting in an image size of 192 by 144 pixels. All computations have been performed by a 200 MHz Pentium Pro system with 128 MB memory running linux. We have tested the system in different natural environments. A robotics laboratory equipped with workstations and typical laboratory equipment, a cafeteria filled with chairs, tables, plants and a lot of people and an ordinary office. The illumination conditions and the scene background differed very much between different experimental setups.

5.1 Experimental results with the modified NCC-method

The color-blob tracking method is a fast and effective method for tracking humans in indoor environments providing the target color is distinctive from the environment. The tracking process performs well within a distance to the target from 1m to 5m. There are

no assumptions about the movement of a person as long as the person moves in the field of view of the camera. Tracking is performed with 5-6 Hz when running with all the other modules of the robot software architecture. This is fast enough since the robot drives not faster than 400 mm/s during tracking of persons. Limitations are the requirement to wear a uni-colored shirt whose color does not occur in the background. Due to the definition of NCC, colors laying on the main diagonal of the rgb cube are mapped to the same point in NCC space. For example, grey values all have similar NCC values and therefore cannot be distinguished without using additional information like their intensity values.

5.2 Experimental results with the color-histogram method

The color-histogram based object detection was implemented to overcome the color-blob method's problem with multi-colored dresses. The color histogram extracted from an image containing a person clothed with a red-green shirt is shown in figure 4. Figure 5 shows the backprojected image. Processing time for a 192×144 pixel image is 0,09 seconds (11 Hz) when running solely on the system. When it is running with the other modules 7 frames/s can be processed. The persons clothes have to be colored. Black and especially white clothes are unsuitable, because our typical environment mainly consists of white colored walls. It is fairly difficult to detect persons who wear colors that occur frequently in the background. The person's clothing colors should be significant i.e. the color distribution should have significant peaks. The person's motion is not restricted in any way as long as it stays inside the field of view. We found that the optimal distance between camera and person in our setup is between 1m and 5m.

5.3 Experimental results with the combined method



Figure 11: Person detection and model extraction using the combined method

In figure 11 the application of the combined approach is shown. The left image is the original image shown as greylevel image. The image in the middle shows the result of the Canny edge detector and the right image contains the new contour model used for the next frame. Using only the contour method on 192×144 sized images requires 83 seconds to detect the person. The combined method which selectively applies the contour-based approach only takes 1 second, which is still sufficient for our purposes. The contour method does not require homogeneous illumination but supposes significant contrast between the tracked person and the background of the scene.

6 Conclusion

We have presented a vision-based approach to person-following on a mobile robot. Experiments conducted in real-world environments show that our system is able to track and follow persons robustly in real-time. Only little requirements are made to the appearance of people. We do not impose any further restrictions on persons to be tracked since we do not use any particular a-priori model. Instead models are extracted at a startup phase when the target person is presented to the system. Using adaptive representations, the approach successfully handles difficulties such as continuous deformation of the person's shape during movement. Since we do not rely on tracking specific features we avoid difficulties such as the occlusion of features by other parts of the person or by the environment. To a certain extent the system is adaptive to changing illumination conditions.

References

- [1] A. Azarbayejani, C. Wren, and A. Pentland. Real-time 3-d tracking of the human body. In *Proceedings of IMAGE'COM 96*, 1996.
- [2] A. Baumberg. Hierarchical shape fitting using an iterated linear filter. *Image and Vision Computing*, 16:329–335, 1998.
- [3] S.A. Brock-Gunn, G.R. Dowling, and T.J. Ellis. Tracking using colour information. In *3rd ICCARV*, pages 686–690, 1994.
- [4] Q. Cai, A. Mitchie, and J.K. Aggarwal. Tracking human motion in an indoor environment. In *2nd International Conference on Image Processing*, Oct. 1995.
- [5] J. Canny. Finding edges and lines in images. Technical Report AI-TR-720, MIT Artificial Intelligence Lab, 1983.
- [6] D.M. Gavrila and L.S. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *Int. Workshop on Face and Gesture Recognition*, 1995.
- [7] D. Huttenlocher, J.J. Noh, and W.J. Ruckli. Tracking non-rigid objects in complex scenes. TR92-1320, Department of Computer Science, Cornell University, 1992.
- [8] S. Niyogu and E. Adelson. Analyzing and recognizing walking figures in xyz. In *Proc. IEEE Computer Society Conference in Computer Vision and Pattern Recognition*, pages 469–474, 1994.
- [9] C. Richards, C. Smith, and N. Papanikolopoulos. Detection and tracking of traffic objects in IVHS vision sensing modalities. In *Proc. Fifth Annual Meeting of ITS America*, 1995.
- [10] M. Sullivan, C. Richards, C. Smith, O. Masoud, and N. Papanikolopoulos. Pedestrian tracking from a stationary camera using active deformable models. In IEEE Industrial Electronics Society, editor, *Proc. of Intelligent Vehicles*, pages 90–95, 1995.
- [11] C. Wong, D. Kortenkamp, and M. Speich. A mobile robot that recognizes people. In *IEEE International Joint Conference on Tools with Artificial Intelligence*, 1995.