ARS ADVANCED ROBOTIC SYSTEMS INTERNATIONAL

A Monocular Pointing Pose Estimator for Gestural Instruction of a Mobile Robot

Jan Richarz[2, now 1], Andrea Scheidig[2], Christian Martin[2], Steffen Müller[2] and Horst-Michael Gross[2]

[1] Intelligent Systems Group, Robotics Research Institute, University of Dortmund, Germany
[2] Department of Neuroinformatics and Cognitive Robotics, Ilmenau Technical University, Germany
jan.richarz@udo.edu

Abstract: *We present an important aspect of our human-robot communication interface which is being developed in the context of our long-term research framework PERSES dealing with highly interactive mobile companion robots. Based on a multi-modal people detection and tracking system, we present a hierarchical neural architecture that estimates a target point at the floor indicated by a pointing pose, thus enabling a user to navigate a mobile robot to a specific target position in his local surroundings by means of pointing. In this context, we were especially interested in determining whether it is possible to accomplish such a target point estimator using only monocular images of low-cost cameras. The estimator has been implemented and experimentally investigated on our mobile robotic assistant HOROS. Although only monocular image data of relatively poor quality were utilized, the estimator accomplishes a good estimation performance, achieving an accuracy better than that of a human viewer on the same data. The achieved recognition results demonstrate that it is in fact possible to realize a user-independent pointing direction estimation using monocular images only, but further efforts are necessary to improve the robustness of this approach for everyday application.*

Keywords: *Human-Robot Interaction, Man-Machine-Interfaces, Gesture Recognition, Robotics*

1. Introduction and motivation

In recent years, a lot of research has been done to develop mobile robotic assistants that can interact with - and be controlled by - non-instructed users, making them suitable for application in everyday life. To achieve this, it is essential to integrate man-machine-interfaces that are natural and intuitive to use. In our ongoing long-term research framework PERSES (PERsonal SErvice Systems) we aim to develop such highly interactive mobile robotic assistants for a wide spectrum of demanding everyday life applications, like shopping assistants for supermarkets or home stores (Gross, H.-M. & Boehme , H.-J., 2000), (Gross, H.-M., Koenig, A., Boehme, H.-J. & Schroeter, C., 2002) or mobile information kiosks for public buildings or areas (Martin, C., Boehme, H.-J. & Gross, H.-M., 2004), (Martin, C., Schaffernicht, E., Scheidig, A. & Gross, H.-M., 2006).

From the human-robot interaction (HRI) point of view, such an interactive mobile service robot must be able to autonomously observe its operation area, to detect, localize, and contact potential users, to interact with them continuously, and to adequately offer its specific services considering the current status of the ongoing dialog. Specific service tasks we want to tackle in this research framework are to interactively guide users to desired areas, rooms or people within its operation area (guidance function) , or to follow the user as a smart user-oriented mobile assistant that is able to continuously observe the user and to immediately react on his/her instructions (service companion function).

Figure 1. Non-verbally commanding a mobile companion robot to a parking position defined by a pointing pose.

To be a really smart companion, a highly interactive robot should be able to analyze both the current user state described for example by gender, age, facial expression, or body language of its interaction partner, and to interpret his verbally or non-verbally given instructions. In this article, we will only focus on a particular aspect of HRI, the video-based recognition of pointing poses and estimation of pointing directions.

The estimation of a pointing direction offers the possibility for different interesting modes of interaction: Knowing where a user is pointing to might help to clarify his intentions or instructions, e.g. when commanding the robot to pick up or manipulate an object. It could also be used to divert the robot's attention to an object or a different interaction partner and, most obvious, to command a robot's movements, e.g. send it to a specific target position in the local surroundings of the user (Fig. 1). The latter is the application we are particularly interested in: Our intention is to command a mobile robot to approach a parking position indicated by pointing on the floor in its vicinity. I.e. the task is to estimate the pointing direction, use it to infer the location on the floor the user is referring to, and then navigate the robot to this location. Note that while the size of the valid area for pointing targets is constrained in this work (see Section 3.2 a), we do not assume the target to be near any object or marker, i.e. the user is able to command the robot to freely selected positions

Besides the methodical background of this recognition technique, we are presenting results of a series of experiments obtained with our mobile experimental robot HOROS (HOme RObot System).

HOROS' hardware platform is an extended Pioneer II based robot from ActiveMedia. It integrates an on-board PC (Pentium M, 1.6 GHz) and is equipped with a laser range-finder and sonar sensors (see Fig. 2). For the purpose of HRI, the robot was equipped with different interaction oriented modalities. This includes a tablet PC for touch-based interaction, speech recognition and speech generation. HOROS was further extended by a simple robot face which integrates an omnidirectional fisheye camera situated in the center of the head, a camera with a telephoto lens mounted on a tilting socket on the "forehead", and a wide-angle camera in one of the eyes. In answer to the manifold possible applications of this robot service companion, the two frontal cameras (eye and forehead) have very different fields of view: The task of the forehead camera is to obtain close-up views of objects, e.g. faces for person identification or facial expression recognition. In contrast to this, the eye camera is designated to deliver wide-angle images of the scene in front of the robot. Therefore, the cameras cannot be reliably combined to form a stereo vision system.

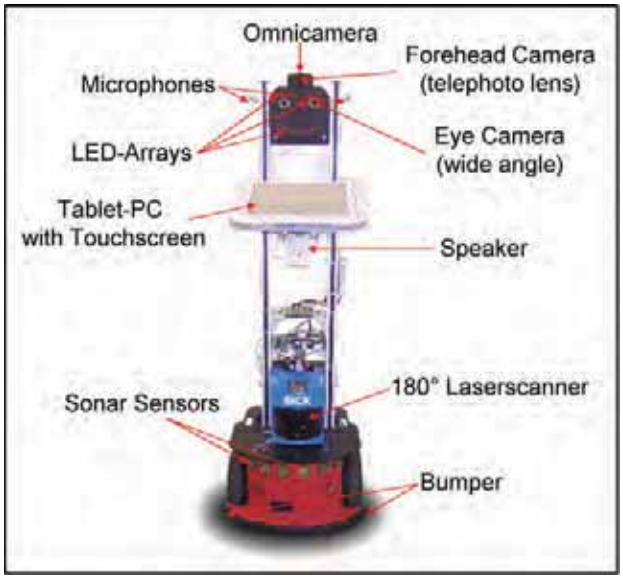

Figure 2. Equipment of the interaction-oriented mobile robot HOROS.

Since one objective of the PERSES-project is the development of a low-cost prototype of a mobile and interactive robot assistant, we are especially interested in vision technologies with a good price-performance ratio. Therefore, for the two frontal cameras, low-cost fixed-lens webcams were utilized. This forces us to develop powerful and robust recognition algorithms in order to compensate for the deficits of the hardware. In this context, we were interested if it would be possible to robustly estimate a target position on the floor from a pointing pose using only inexpensive hardware and monocular images.

The remainder of this article is organized as follows: First, in Section 2 we give a very brief overview over our multi-modal people detection and tracking system. A thorough description of this system is not subject of this article, but we show how it is exploited as a required prerequisite for the pointing direction estimator, which we present in detail in Section 3. Section 4 describes the experimental investigations we conducted in order to assess the overall performance and drawbacks of our estimator, and presents the results. We conclude with a summary in Section 5 and give an outlook on possible improvements we plan to examine in the near future.

2. Multi-modal people detection and tracking

A fundamental prerequisite for the video-based recognition of user instructions is a stable detection and tracking of the interaction partner in the local surroundings of the robot. Therefore, we recently developed a new approach for the integration of several sensor modalities and presented a multi-modal, probability-based people detection and tracking system and its application using the different sensory systems of our mobile interaction robot HOROS . This approach can be characterized by the fact

that all used sensory cues are concurrently processed and integrated into a robot-centered, local hypothesis map using a probabilistic aggregation scheme. Up to now we utilize the laser-range-finder, the sonar sensors, the omnidirectional and the frontal eye-camera of our experimental platform HOROS (see Fig. 2) as sensor inputs for our probabilistic tracker illustrated in Fig. 3. A detailed discussion of the advantages and drawbacks of the several sensory modalities, the mathematical details of the probabilistic aggregation scheme, and experimental results of this multi-modal and multi-person tracker are given in (Martin, C., Schaffernicht, E., Scheidig, A. & Gross, H.-M., 2006).

By turning the robot towards that tracker hypothesis with the largest weight (defined, e.g., by the smallest distance to the robot), the potential user can be directly localized in front of the robot, allowing the frontal cameras to evaluate if that person could be willing to interact with the robot. As a very simple criterion, we assume that a tracked person may be considered to be a user willing to interact if the upper part of his body is oriented towards the robot. This decision is taken by means of a Viola & Jones detector (a cascaded feature detector that uses Boosting to obtain strong classifiers from simple Haar-like box filters, see (Viola, P. & Jones, M., 2001) for details) - in this case a head-shoulder detector. If this proves to be true, in the next step the robot can try to recognize the user's instructions. In the case presented here, we are interested in estimating the target position of a pointing pose triggered by a preceding voice command, like the call "HOROS!", to attract the robot's attention.

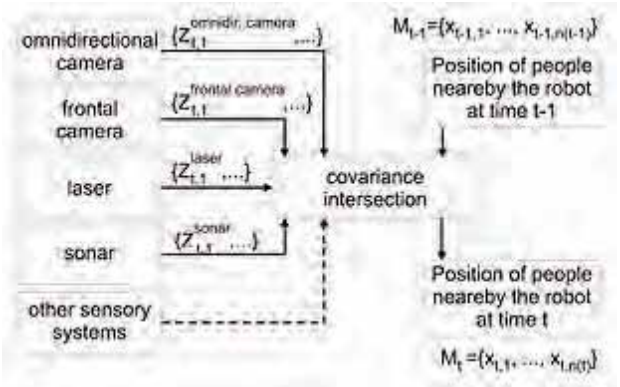

Figure 3. General architecture of the multi-modal tracking system presented in (Martin, C., Schaffernicht, E., Scheidig, A. & Gross, H.-M., 2006) : The observations $z_{t,i}^{omnidir.camera}$, $z_{t,i}^{frontal\_camera}$, $z_{t,i}^{laser}$ and $z_{t,i}^{sonar}$ of the different sensory cues modeled as Gaussian hypotheses are combined in a robot-centered local map M_t that contains a time varying number n(t) of object hypotheses $x_{t,j}$ moving around the robot. Hypotheses are fused by means of the Covariance Intersection rule (Julier, S. & Uhlmann, J., 1997).

3. Monocular pointing pose estimation

3.1 State-of-the-art in pointing pose estimation

Inter-human communication is based on many different facets. Speech, gestures, body pose, facial expression and many other aspects influence the way information is transferred, and the information itself. Many of these aspects are difficult to observe and distinguish or are even not yet understood completely. Even we humans, having cognitive skills superior to every technical system, sometimes fail to understand all of them and therefore misinterpret the intentions of our communication partner. This makes it particularly difficult to implement human-machine interfaces that are really natural. However, integrating some of these aspects into an interface helps to make it more intuitive and natural-looking.

Not only is interaction with the robot simplified for the human user, under the view of "social robotics", enriching a robot with the ability to show such "social skills" is considered even more essential. For example, when reviewing basic requirements and design principles for socially interactive robots, (Fong, T., Nourbakhsh, I. & Dautenhahn, K., 2003) state that "a socially interactive robot must proficiently perceive and interpret human activity and behavior. This includes detecting and recognizing gestures, monitoring and classifying activity, discerning intent and social cues, and measuring the human's feedback." So, if we want to design a robot service companion that is fully accepted as a competent interaction partner by its users, it must have the ability to react to natural human behaviour in a suitable manner.

One of the most important and informative aspects of nonverbal inter-human communication are gestures and poses. In particular, pointing poses simplify communication by linking speech to objects or locations in the environment in a well-defined way. Therefore, a lot of work has been done in recent years focusing on integrating gesture recognition into man-machine-interfaces.
However, most of this work concentrates on distinguishing a fixed symbolic set of gestures, creating a "command alphabet" for robot control. Figure 4 shows a non-exhaustive overview of viewpoints that can be used to describe and classify vision-based gesture recognition and pointing pose estimation approaches. They can be distinguished by the used camera configuration and image quality, the amount of preprocessing (like user-background segmentation) that is utilized, the way, features are extracted, encoded and represented, the applied recognition algorithm, the mechanism that triggers the recognition process, and – of course – the application fields they are suitable and intended for.

A good introduction and overview on the subject of gesture recognition for human-computer interaction (HCI) – including gesture taxonomy, different approaches for

spatial and temporal gesture modeling, and analysis – is given in (Pavlovic, V., Sharma, R.. & Huang, T., 1997).

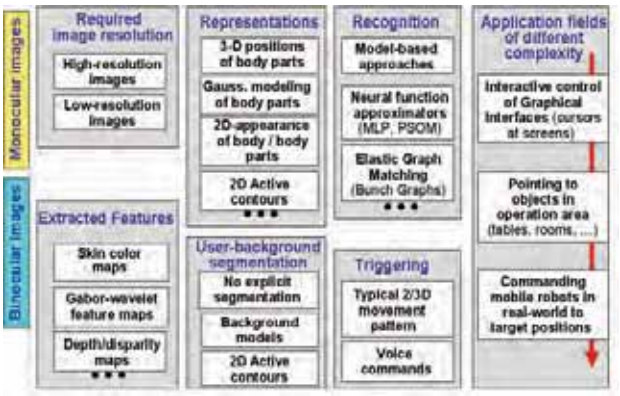

Figure 4. Overview of significant criteria and aspects to describe and distinguish vision-based gesture recognition and pointing pose estimation approaches.

(Rogalla, O., Ehrenmann, M., Zoellner, R., Becher, R. & Dillmann, R., 2002) presented a system that classifies hand postures for robot control. They use monocular high-resolution color images and extract a hand contour by means of skin color segmentation. This contour is sampled with a fixed number of sampling points, normalized and Fourier-transformed. The Fourier descriptors represent the feature vector that is classified using a model database and a distance measure.

(Triesch, J. & von der Malsburg, C., 2001) detect and classify hand postures in monocular images by using compound bunch graphs. No explicit segmentation is needed, since their system can cope with highly complex backgrounds. The features used are the responses of Gabor wavelets and color information at the graph nodes. Hand postures are classified using a distance measure to a model graph, taking into account deformation and scaling.

Up to now, there are only a few authors who tried to actually estimate a pointing direction out of a deictic gesture. (Jojic, N., Brumitt, B., Meyers, B., Harris, S. & Huang, T., 2000) did so by detecting a person using dense disparity maps (from a stereo system) and color information in low-resolution images. In their approach, after an explicit background subtraction using a statistical background model, a simple Gaussian mixture model of the body and outstretched arm is fitted to the person. If a pointing gesture is detected (when the angle between the main principal components of the arm and body "blob" exceeds a threshold), the pointing direction is determined from the largest principal component of the "arm-blob".

(Noelker, C. & Ritter, H., 1998) use low-resolution monochrome images from two infrared cameras. The images are Gabor-filtered and then a Local Linear Map (LLM) classifier calculates the 2D positions of landmarks (shoulder, elbow and hand of the pointing arm). This is done separately for the two camera images. A Parametrized Self-Organizing Map (PSOM) then estimates the 3D coordinates of these landmarks, making it possible to calculate a pointing direction. The approach is used to control a Virtual-Reality-System and therefore the working conditions for their system can be very restrictive.

(Nickel, K. & Stiefelhagen, R., 2003) classify dynamic gestures by means of Hidden Markov Models (HMM). They extract candidates for heads and hands using color and disparity information from a stereo camera system. These candidates are transformed into a user-centered polar coordinate system, yielding a three-dimensional feature vector. Tracking is performed by maximizing a product of three quality scores. For detection of pointing gestures, they model three gesture phases - *begin*, *hold* and *end* – and train a HMM for each of these phases. If a pointing gesture is recognized, the pointing direction is estimated by calculating the connecting line between the center of the head and the hand for the *hold*-phase.

(Hofemann, N. Fritsch, J. & Sagerer, G., 2004) identify pointing gestures and referred objects in monocular color images. Hand regions are extracted by skin-color segmentation and tracked over time with a Kalman filter. Activity recognition is achieved with a modified version of the CONDENSATION algorithm (Isard, M. & Blake, A., 1998): Activities are classified via matching with parametrized models. By adding a context area to each particle, the authors link a pointing gesture to a referred object and therefore can identify objects the user points at. However, they do not really estimate a pointing direction, links are only established because of spatial proximity in the image.

With our approach, we are interested to determine whether it is possible to accomplish a pointing position estimator using only monocular images of low-cost cameras as input data. No explicit background segmentation is utilized, and we use an appearance-based approach. Feature extraction is done by means of Gabor wavelets. A cascade of Multi-Layer Perceptron (MLP) neural function approximators serves as pointing direction estimator. The estimation process is triggered by a simple voice command, e.g. the call "HOROS!", so no explicit recognition of pointing poses (or distinction from non-pointing actions and meaningless gesticulation) is necessary. Note that this distinction is itself a non-trivial problem commonly referred to as "gesture spotting". Different approaches have been proposed to tackle this task. (Nehaniv, C. et al., 2005), for example, suggest using the interaction history and context knowledge to infer the type of gesture performed. In a way, (Hofemann, N. Fritsch, J. & Sagerer, G., 2004) realize this by assuming a pointing gesture when an object is present in a context area near

the hand. Other authors approach the problem by classifying hand shapes (e.g. Stoerring, M., Moeslund, T., Liu, Y & Granum, E., 2004) or trajectories (e.g. Nickel, K. & Stiefelhagen, R., 2003). By utilizing a voice command, we elude the gesture spotting problem. We do not believe this to be a serious constraint of our system, since it is natural behavior to use a speech utterance in order to catch the attention of an interaction partner.

We implemented this approach on our mobile robot HOROS, making it navigate to specified targets, thus enabling a user to control HOROS only by means of pointing. To the best of our knowledge, there are no other low-cost oriented approaches that are comparable to the one presented here.

3.2 System Overview

a) Pointing Area, Training Data and Ground Truth:
We encoded the target points on the floor as (r, φ) coordinates in a user-centered polar coordinate system. This requires a transformation of the target estimate into the robot's coordinate system (by simple trigonometry), but the estimation task becomes independent of the distance between user and robot. Moreover, we limited the valid area for targets to the half space in front of the robot with a value range for r from 1 to 3 m and a value range for φ from −120° to +120° . The 0° direction is defined as user-robot-axis, negative angles are on the user's left side. With respect to a predefined maximum user distance of 2 m, this spans a valid pointing area of approximately 6 by 3 m on the floor in front of the robot in which the indicated target points may lie.

Fig. 5 shows the configuration we chose for recording the training data. We used three markers (at distances of 1, 1.5 and 2 m from the robot) specifying different user positions, however, in Fig. 5 for reasons of clarity only the marker in 1.0 m distance is shown. Around each marker, three concentric circles with radii of 1, 2 and 3 m are drawn, being marked every 15°. Positions outside the specified pointing area are not considered. The subjects were asked to point to the markers on the circles in a defined order and an image was recorded each time (see Fig. 6, right). Pointing was performed as a defined pose, with outstretched arm and the user fixating the target point. All captured images are labeled with distance, radius and angle, thus representing the ground truth used for training and for the comparing experiments with human viewers (see Section 4.1).

This way, we collected a total of 900 images of 10 different interaction partners During preprocessing, the Regions of Interest (ROI - see next section) were calculated from manually marked starting points and then moved a few pixels in each direction to receive a slight variation of the data. This way, nine samples per training image were extracted, resulting in a sample database of 8,100 labeled images. This database was divided into a training subset and a validation subset containing two complete pointing series (i.e two sample sets each containing all possible coordinates (r, φ) present in the training set). The latter was composed from different persons and includes a total of 1620 images (18 samples for each valid (r, φ) coordinate). This leaves a training set of 6480 samples (72 samples for each valid (r, φ) coordinate).

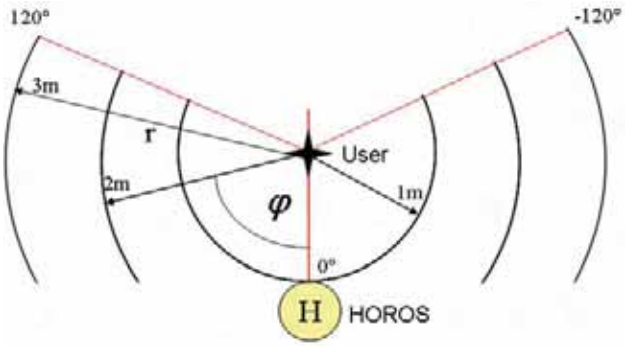

Figure 5. Configuration used for recording the ground truth training and test data. Here, for reasons of clarity only one of the marked positions in front of the robot (at 1m distance) to generate pointing poses to predefined target points is shown.

b) Preprocessing and Feature Extraction:
Since the users standing in front of the camera can have different height and distance, an algorithm had to be developed that can calculate a "normalized" region of interest, resulting in similar subimages for subsequent processing. We determined the ROI by using a combination of head-shoulder-detection (based on the Viola & Jones Detector cascade mentioned above), empirical factors, and the distance measurement from the multi-modal person tracker described before (Fig. 6).

The head-shoulder detector will typically yield a center of detection somewhere in the throat area of the user. Its coordinates are used as starting point.

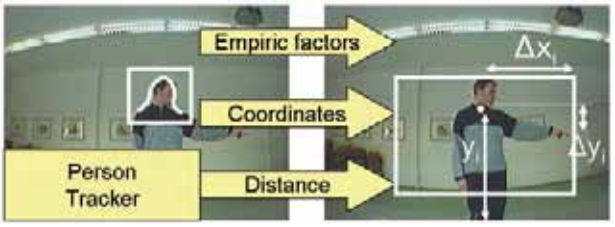

Figure 6. Example for an image provided by the low-cost eye-webcam. Moreover, this figure sketches how the region of interest (ROI) in the camera image is determined: a combination of empiric factors, head coordinates of the head-shoulder-detector and distance estimation given by the multi-modal tracker (see Section 2) is used to achieve a normalized ROI (right).

Since this center of detection will have different vertical image coordinates (y-coordinates) for different user heights given a constant user-camera distance, it can be used to implicitly include the user's height into the calculation. Next, the size of the ROI has to be determined based on the head-shoulder detection result. To this purpose, we measured the maximum distances between the center of detection and the tip of the pointing arm in both x and y direction for different subjects. We also determined these values for different user-camera distances for each subject.

Let $\Delta x_{i,d}$ and $\Delta y_{i,d}$ be the measured maximum distances between the center of detection and the tip of the pointing arm in pixels for subject i given user-camera-distance d. x_i and y_i are the coordinates of the center of detection yielded by the head-shoulder detector. We calculate

$$fac_{x,i,d} = \frac{\Delta x_{i,d}}{y_i}, \quad fac_{y,i,d} = \frac{\Delta y_{i,d}}{y_i} \tag{1}$$

and obtain two factors $fac_{x,i,d}$ and $fac_{y,i,d}$ specifying the maximum extension of the pointing arm for this particular user i and distance d dependent on the user's height in the image y_i. In other words, these factors describe how big the ROI rectangle must be to ensure it contains the complete pointing arm for this person and the pose for which they were measured. Since we chose examples with maximum possible extension of the arm given the valid pointing area – more precisely the cases 90° pointing direction with 3 m pointing distance (maximum extension in x-direction) and 0° pointing direction with 1 m distance (which means the user is pointing at the ground directly in front of himself, thus yielding the maximum extension in y-direction) – these two factors ensure that the calculated ROI *always* contains the complete pointing arm for this user, provided he is performing a valid pointing action.

As mentioned before, these values were calculated for different subjects and for different user-camera distances, namely 1, 1.5 and 2 meters. Since $fac_{x,i,d}$ did not vary strongly for the subjects i given distance d, we took the mean values $\overline{fac_{x,d}}$ for the calculation of the ROI width. By doing this, we assume that the ratio between height and arm length is the same for most humans, which is not true in general. But our experiments showed that this inexactness is minor compared to other error sources present in the system.

Thus, our ROI is $2 \cdot y_i \cdot \overline{fac_{x,d}}$ wide and has its center at the center of detection (x_i, y_i). Note that this width is dependent on the user´s height in the image y_i and this way compensates for different arm lengths due to different heights of users. We determined the values of $fac_{x,1.0}$ as 1.6, $fac_{x,1.5}$ as 1.4 and $fac_{x,2.0}$ as 1.2, respectively. Linear interpolation over the values for different distances d (with a valid value range from 1.0 to 2.0) yielded the following expression for the ROI width w:

$$w = y \cdot (1.2 + 0.4 \cdot (2.0 - d)). \tag{2}$$

Since $fac_{y,i,d}$ was almost exactly the same value for all subjects, we decided to discard this factor and simply fix the height of the ROI to the next whole-numbered width-to-height ratio, which is 3:2. However, it is not reasonable to center the ROI vertically at the center of detection because we would like to extract an image region containing the head of the person as uppermost and the tip of the pointing arm as lowermost part, while avoiding as much noninformative background as possible. So we have to shift the ROI downwards (remember that the center of detection is in the throat area). We do this by calculating an offset o_y using the following equation

$$o_y = (60 - 25 \cdot (d - 1.0)), \tag{3}$$

which was obtained in a similar way as described above: Determining the appropriate offsets for different subjects and different distances d, calculating the mean values over all subjects and then interpolating linearly over the distance d. Please note that these parameters depend highly on the camera used to record the images. Note also that we use the distance estimation d from the tracking system to simply scale the ROI size and offset linearly according to the user-camera distance. Although this algorithm is quite simple, it shows satisfactory results. Fig. 7 shows typical ROI extracted this way.

The cropped ROI is scaled to 81x81 pixels, and then an illumination correction and histogram equalization is applied. After this, the preprocessed image is Gabor-filtered (4 frequencies with 8 orientations each, absolute values of filter responses) using an equidistant 4x5 grid to extract a pose-describing feature vector as input for the first stage of the pointing estimator. For later stages, the ROI is modified again to create two subimages (using a modified version of the algorithm described above), one of them containing the pointing arm, the other one the head (Fig. 7, bottom). By doing this, the head pose of the instructor is directly integrated into the pointing pose estimation as additional information.

c) Architecture of the Classifier Cascade:

A series of experiments showed that it is not possible to tackle the function approximation problem with a single neural network estimating both radius and angle in one step. It also became clear that, while the radius estimation works quite well, it is more difficult to robustly estimate the angle. Therefore, we decided to use a cascade of neural classifiers and function approximators (typically three-layered MLPs trained by means of the RPROP learning rule (Riedmiller, M. & Braun, H., 1993)).

Figure 7. (Top) Captured ROI extracted with the described normalization algorithm for three instructors with different height (from 1.65 to 2 m) all performing the same pose. (Middle) Extracted ROI for different distances person-robot ranging from 1-2 m. (Bottom) Examples for sub-images extracted from the ROI containing both the pointing arm and the head pose. By using these as input data for the target point estimator, the head pose is integrated as additional information.

Fig. 8 gives an overview over the architecture of the developed target point estimator cascade. After extracting and preprocessing the ROI, a left/right MLP classifier (topology: 640-40-20-2, i.e. 640 input neurons, two hidden layers with 40 and 20 neurons, respectively, and 2 output neurons.) first determines whether the person is pointing to the left or to the right. Knowing this, that half of the input image that does not contain the pointing arm can be discarded. This way two smaller ROIs containing the head and body-arm regions (see Fig. 7 (bottom)) can be extracted. Each of these two input images is also Gabor-filtered (4 frequencies with 8 orientations, absolute values of filter responses) using an equidistant 5x5 grid resulting in 1,600 input features describing the head and arm pose sub-images. If the person is pointing to the left, the image is simply flipped. This allows us to use the same classifier for both directions.

In the following cascade stage the value for the pointing radius r is directly estimated by means of a first MLP function approximator (network topology: 1600-30-20-1 neurons, i.e. 1,600 input neurons, 30 neurons in the first hidden layer, 20 neurons in the second hidden layer, and 1 output neuron) with the single output neuron linearly coding the range from 1 to 3 m (output interval: 0 ...1.0). Since the estimation of the pointing angle φ is less accurate and prone to errors, this estimation is done later in the cascade to provide this stage as much supporting and simplifying information as possible. To that purpose, the arm and head ROIs are first classified into one of three coarse radius classes (see Fig. 8, bottom left). For each of these classes, there is a specialized MLP classifier assigning the input to a coarse angle class (network topology: 1600-30-20-10-3, i.e. 1,600 input and 3 output neurons, and 30, 20 and 10 neurons in the several hidden layers). Finally, within the respective coarse class, a finer estimation of φ is determined by highly specialized MLP function approximators (with slightly different topologies for the 9 subclasses, typically 1600-20-10-5-1) leading to the final target estimation [r, φ].

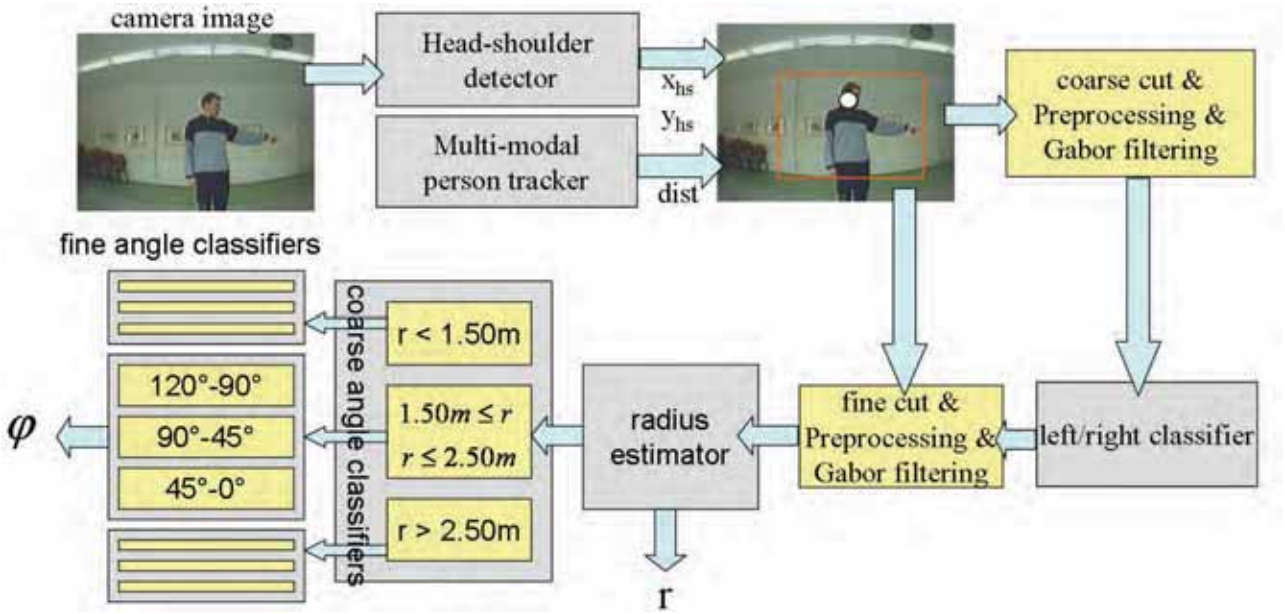

Figure 8. System overview of the target point estimator cascade. The Gabor-filtered subimage is first fed into a left/right - classifier. The result of this classifier enables it to extract the finer image ROIs shown in Fig. 7, bottom. In the following stage, the final pointing radius r is estimated, and the input is classified into one of three radius classes. For each class, a coarse angle estimator is trained, yielding a classification into one of three angle classes. The last stage yields the final angle estimate.

The cascade contains a total of 14 MLP networks (1x left/right, 1x radius, 3x coarse angles, 9x fine angles), but, due to the hierarchical architecture, only four of these classifiers have to be activated during one pass.

4. Experimental Results

4.1 Estimation Results of Human Viewers:

In order to get a reference value for the recognition performance of the estimator, in particular experiments we determined how accurately a human viewer could estimate the referred target point from a monocular image. Therefore, the images from the training and test data sets were presented to test viewers in random order using the graphical user interface shown in Fig. 9. The valid area for the pointing targets is specified by a circular arc. Subjects were told beforehand that the targets can only lie within this area. The circular arc is skewed perspectively to create a 3D impression and adapted in size according to the distance between the person in the image and the camera. The test person marked a guessed target point by clicking with the mouse pointer on the interface. The found coordinates were then transformed according to the given perspective and the distance of the person, yielding the estimated target coordinates r and φ. The estimates were then compared with the known image labels.

These comparative experiments were performed with 8 test viewers, resulting in 885 target estimates altogether. The achieved estimation accuracy is shown in Fig. 10.

On the top, the mean values and standard deviations of the angle estimates are shown versus the correct angle. Obviously, perfect estimates would lie on a straight line depicted by the dotted line in the image.

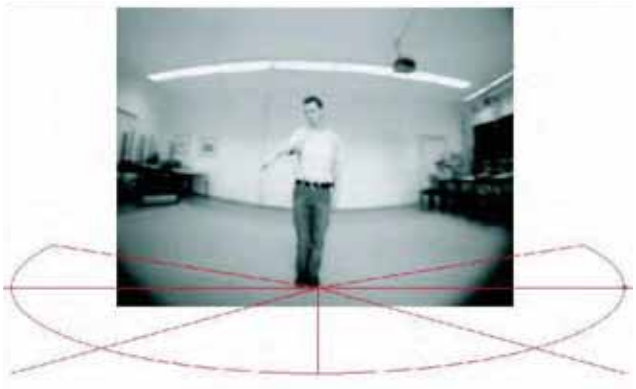

Figure 9. Graphical user interface for experiments with human viewers. The valid target area is marked by the circle segment.

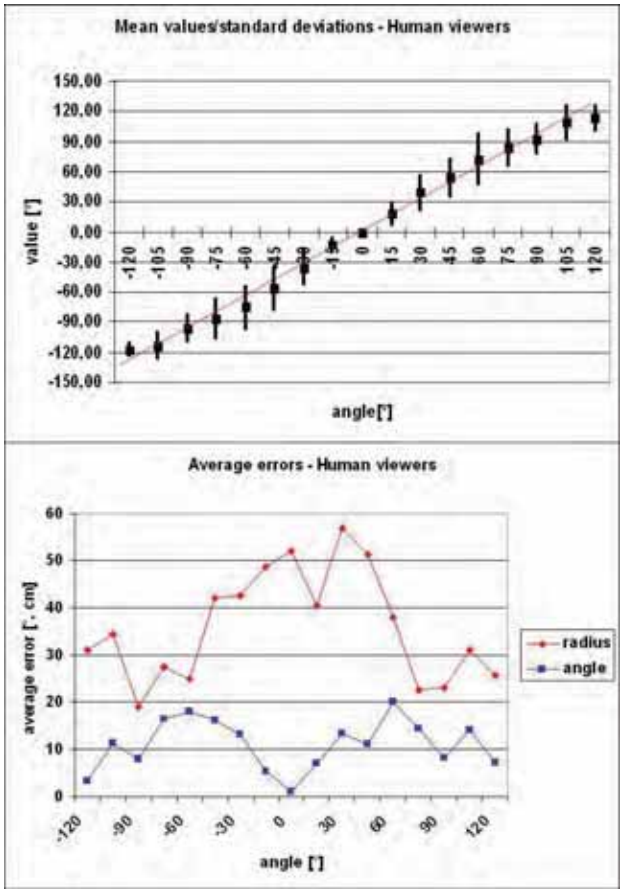

Figure 10. Estimation results of human viewers. (Top) Mean values and standard deviations of the angle estimate vs. the correct angle. (Bottom) Average errors of the radius and angle estimates vs. the correct angle.

The mean values of the estimates deviate slightly from this ideal case. It is noticeable that angle estimates between (+/-) 45° and 90° are persistently too large in magnitude. What's more, the standard deviations (depicted by the vertical lines) for these angles are significantly higher. So, it seems to be quite difficult for a human viewer to precisely estimate φ from the monocular images in this area.

At the bottom of Fig. 10, the average errors for the estimates of r and φ versus the correct angle are shown. For the radius r, the errors are significantly higher for small angle values compared to large angle values. The errors for φ behave inversely, being small for small angle values, then getting bigger with increasing angle value. For an explanation of this behavior, consider the situations depicted in Fig. 11.

The upper figure shows the top view of a person (P) standing in front of HOROS (H) (i.e. the camera) and performing a pointing action with a small angle φ_1 (left) and a large angle φ_2 (right). A change of pose by angle $\Delta\varphi$ results in a different length of the pointing arm's projection (l_1, l_2) into the image plane. Obviously, this difference

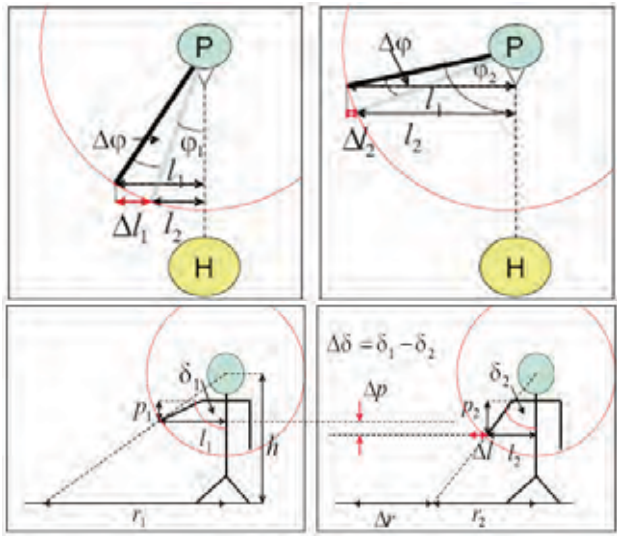

Figure 11. Geometrical explanation for the experimental results shown in Fig. 10 (Bottom). Top view (Top) and frontal view (Bottom) of a person (P) performing a pointing pose with different indicated angles φ.

Δl is bigger for small angles *φ*. Or in other words, as the value of *φ* gets larger, it becomes increasingly difficult to distinguish a pose change by *Δφ*, leading to larger estimation errors.

A similar explanation can be given for *r* (Fig. 11 bottom). This time, the projection of a frontal view of the user into the camera plane is shown. A change of the indicated radius *r* results in a significant change of the body pose. This change is measurable in the image by the values *Δδ*, *Δp* and *Δl*. Since the projections of *Δδ* and *Δl* into the image plane depend on *sinφ*, they get smaller for smaller values of *φ* given a constant change of radius *Δr*. Again, this means that changes of pose get harder to distinguish for small angles *φ*, thus yielding larger estimation errors and leading to the behavior visible in Fig. 10. For angles greater than 90°, the errors decrease again. This is due to the fact that pointing to a target behind one's position results in a significant change of the body pose: The shoulder and the face are turned backwards, which is clearly visible in the images.
Overall, in 50.1% of all cases, the human viewers estimated *φ* correctly within a tolerance of 10°. For *r*, 76.3% of all trials were within a tolerance of 50 cm. These results give a hint for valuating the following results of our neural estimator, keeping in mind that the presented data, the distorted monocular images, are very unfamiliar for a human.

4.2 Results of the Neural Estimator Cascade:
In the following experiment, static image sequences of different users performing pointing poses were used. These sequences were recorded using the same configuration (and ground truth) as for the training set (see section 3.2 a).

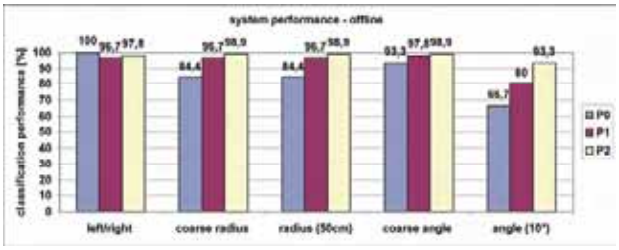

Figure 12. Classification results of the different stages of the estimator cascade for 3 test subjects. (From left to right:) left/right classifier, radius classifier (radius classes), radius estimate (tolerance 50cm), coarse angle classifier, angle estimate (tolerance 10°)

In each test image, the correct face position was labeled manually beforehand. By means of this step, the negative influence of positioning errors possibly generated by the automatic head-shoulder-detection could be completely eliminated. This way, the performance and properties of the developed ROI extraction algorithm and the neural estimator cascade could be analyzed without impairments by deficits of preceding subsystems. Fig.12 shows the classification results of each cascade stage for three test persons. For comparison, person P2 is from the training data set. All results mentioned in the following passages refer to the two remaining subjects not included in the training data, i.e. previously unseen by the system.

The left/right classifier yields classification rates of almost 100% for all subjects. This is especially important since further processing of the input image depends on the results of this stage, and misclassifications will lead to a totally erroneous target estimate. The radius estimator stage shows a good overall performance, with 84.4% and 96.7% of the samples within an allowed 50 cm tolerance and classified into the correct radius class. Compared to this, the angle estimator stages perform poorly: While the performance of the coarse angle classifier stage is very good for all subjects, the fine angle estimate is not, with only 66.7 or 80% of the samples within the 10° tolerance. These results show that the angle estimate is the major problem, limiting the performance and accuracy of the developed pointing direction estimator.

When comparing the results for P0 and P1 in Fig. 12, it is noticeable that the performance for P0 is significantly worse. This shows a drawback of the current system: P0 tended to perform the pointing gestures with the pointing arm not fully extruded, but slightly angled. Obviously, the neural estimator is quite sensitive to deviations from the pose it was trained for.

For comparison with Fig. 10, the diagram for the mean values and standard deviations of the angle estimate is given in Fig. 13 (top). The results are close to the optimal

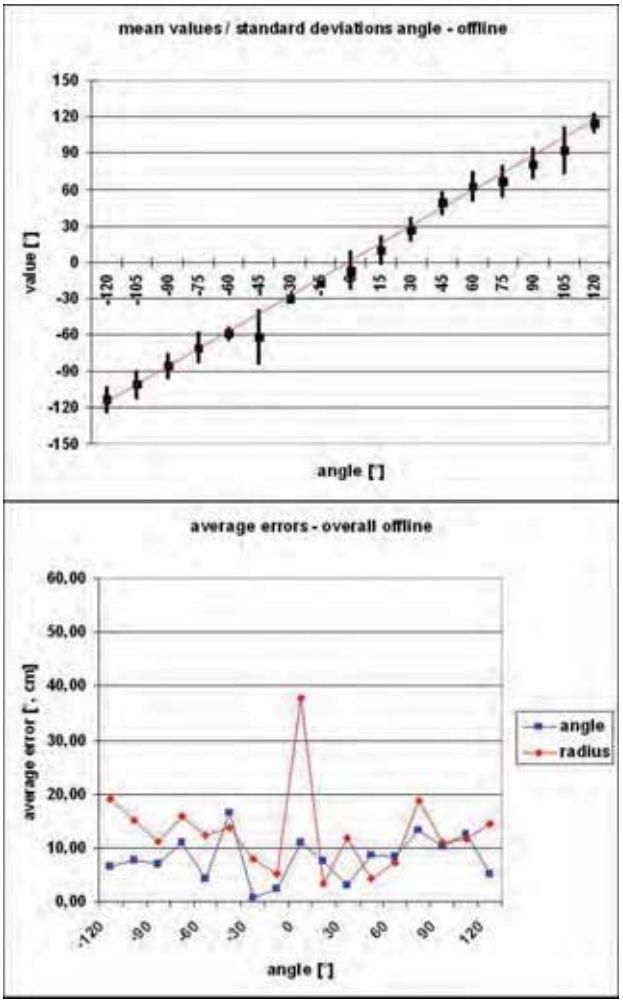

Figure 13. (Top) Mean values and standard deviations of the angle estimate vs. the correct angle determined on manually selected ROIs (off-line estimation). The ideal case is depicted by the dotted line. (Bottom) Average errors for radius and angle estimate vs. the correct angle.

straight line with small standard deviations for most angles. Fig. 13 (bottom) shows the average errors of the angle and radius estimates. The behavior of the angle estimate is quite similar to that observed in Fig. 10. The radius estimate behaves almost inversely to that observed before, apart from the large errors for 0°. So far, we have no complete explanation for this deviation from expected behavior, but we believe a correlation between the outputs r and φ of the detector cascade to be the reason.

Looking at Fig. 12 again, it can be seen that the neural estimator achieved a classification rate of 66.7% and 80% respectively for the fine angle estimate with a tolerance of 10°, and 84.4% or 96.7% for the radius estimate with a tolerance of 50 cm. This is significantly better than the results achieved by human viewers (50.1 / 76.3%). But of course, the latter are more reliable in the sense that they don't produce outliers and large errors.

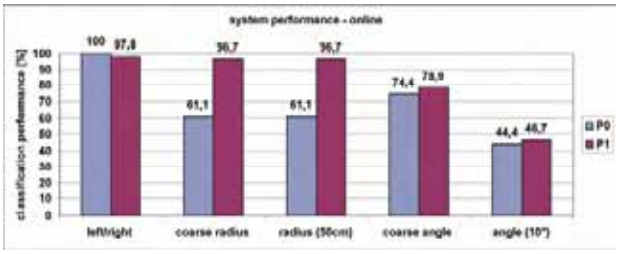

Figure 14. Online classification results of the different stages of the estimator cascade for two test subjects. In these experiments the head-shoulder detector was activated for positioning of the ROI.

When interpreting these results, we have to keep in mind that they were achieved off-line with a perfect head detection (and therefore a perfect ROI placement). Fig. 14 shows the performance of the classifier stages for the two test persons when the Viola & Jones detector is activated and used online for head-shoulder-detection. This detector reliably finds persons in the camera image, but its center of detection is usually not centered exactly on the persons's throat. In this case the recognition rate for the coarse radius becomes about 20% lower for person P0 and stays constant for P1, while the fine angle estimates (with a tolerance of 10°) get significantly worse for both persons (only 45%). This clarifies that of all possible error sources, the head shoulder detection is the most crucial: misplacements of a few pixels from the optimal position may already lead to greater errors in the final target estimate.

To determine the overall online performance and precision of the presented target point estimator while operating on the mobile robot HOROS , a random target pointing experiment was conducted finally: Standing at many different positions within the operation area, the instructor pointed to randomly selected target positions in his local surroundings, and the robot had to navigate from its current rest position to the estimated target position. From a total of 72 trials, only six (8.3%) were totally erroneous outliers. The remaining trials yielded an average position error of 59 cm. 28 of them (38.9%) were within 50 cm, 31 (43.1%) within 1 meter, and 7 (9.7%) within 1-2 m from the target point.

For a correct interpretation of these results, it should be taken into account that in this experiment all possible disturbances and localization errors superimposed: an imperfect person tracking and head-shoulder detection resulting in non-optimally placed ROIs, an erroneous target point estimation with many different reasons (changing background, badly executed pointing poses, image disturbances, etc.), and insufficiencies in the robot's navigation system resulting, for example, in an imperfect self-localization and motion planning to the given target points.

5. Summary, Discussion and Outlook

In this article, we presented a neural classifier cascade for appearance-based estimation of a referred target point on the floor from a pointing pose. Although we only use monocular image data of relatively poor quality, the system accomplishes a good target point estimation, achieving an accuracy better than that of a human viewer on the same data. The achieved performance rates demonstrate that it is in fact possible to realize a user-independent pointing pose estimation system using monocular images only, but further efforts are necessary to improve the robustness of this approach for everyday application.

There are several possible improvements to our system that need to be investigated in the near future: First, the used feature extraction (Gabor filtering using an equidistant grid) seems to be too simple for a robust target point estimation. Several more sophisticated methods for feature extraction and representation are imaginable that may lead to better results. For instance, a foreground extraction routine, e.g. based on active contours or shapes, could be applied, segmenting the pointing person from the background and thus limiting disturbing background influences.

Second, further efforts are necessary to improve the accuracy of the head-shoulder detection preceding the target point estimation. Possibly this can be achieved by combination with the active contours to compensate for the deficits of a simple input-driven detector. It is also imaginable to use the Viola-Jones head-shoulder detector only as first step of a cascaded detector. The detections it yields could then be used to restrict the search area for a more specialized and accurate detector concentrating on features that are hard to find robustly in the complete camera image.

An interesting approach that could potentially solve both problems mentioned above is described in (Treptow, A., Cielniak, G., and Duckett, T. 2005): The authors use a particle filter to find and track persons in thermal images. Each particle is labeled with a set of parameters that describe a simple body model. Edge features are used to evaluate how good the respective model fits to a person in the image, and the particles are weighted accordingly. Thus, the particle filter is used to optimally fit the model to a person in the image. This could help to solve the user-background segmentation problem and provides a good hint for ROI placement. Moreover, the model parameters could be used as additional cues for the classifier. But since we intend to utilize low-cost hardware, the usage of thermal cameras is not possible. We are currently evaluating whether a similar approach can be applied to normal monochrome images.

We are also thinking of implementing an explicit head pose or gaze estimator (instead of implicitly including head pose by using a head sample as additional classifier input, see Fig. 7) and fuse the results of both classifiers to improve accuracy.

Moreover, so far we only evaluated the performance of our target point estimator on single images of the final pointing pose. An interesting question is whether the dynamic movement of the pointing arm to the final pose contains additional information that could be exploited to enhance the precision of the estimator. In a first attempt to integrate the temporal history of the pointing gesture, we utilized a Kalman filtering algorithm using several very simple system models. The observable states of the system (e.g. the pointing user) are simply the indicated coordinates r and φ. As sensor model, we used the outputs of the estimator cascade in combination with a large uncertainty for φ and a somewhat smaller uncertainty for r (the absolute values for these were derived according to the experiments described before and then slightly varied). In a first experiment, r and φ were assumed to be constant over time. Since this is not true for a dynamic pointing gesture, this model did not yield satisfactory results, as could be expected. We continued with models that allowed r and φ to change with constant and arbitrary velocities. This very simple temporal filtering (realizing a temporal lowpass filter) did not improve the overall accuracy significantly, but helped to stabilize the estimator outputs: The estimator showed a tendency to "jump over" certain value ranges - especially for the angle estimate φ – when used on single images. This tendency could be reduced. A more sophisticated filtering method, like a particle filter, might yield better results. Further investigations are required on this topic.

Another critical issue we have to tackle is the speed of the estimator in combination with the motion planner and navigator. In the current implementation of our demo application (which was in no way optimized for speed), the complete evaluation of a given pointing pose – including person detection, ROI calculation and extraction, calculation of the neural pointing pose estimator cascade, transformation of the estimator results into the robot´s world coordinate system and navigator path planning to the target point - takes about four to six seconds on HOROS from the starting command to the beginning of the robot´s movement. Although this seems an acceptable delay for the scenarios we considered, it is desirable to speed up the whole estimation and target interpretation process allowing an immediate reaction of the mobile robot to a given pointing pose command in real-world environments and tasks.

6. References

Fong, T., Nourbakhsh, I. & Dautenhahn, K., 2003. "A survey of socially interactive robots", in: Robotics and Autonomous Systems 42 (2003), pp. 143-166, Elsevier Science

Gross, H.-M. & Boehme, H.-J., 2000. "PERSES - a Vision-based Interactive Mobile Shopping Assistant", in: Proc. 2000 IEEE Intern. Conf. on Systems, Man and Cybernetics (IEEE-SMC 2000), pp. 80–85

Gross, H.-M., Koenig, A., Boehme, H.-J. & Schroeter, C., 2002. "Vision-Based Monte Carlo Self-localization for a Mobile Service Robot Acting as Shopping Assistant in a Home Store", in: Proc. 2002 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2002), pp. 256–26

Hofemann, N. Fritsch, J. & Sagerer, G., 2004. "Recognition of Deictic Gestures with Context", in Proc. Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM) -Symposium 2004, LNCS 3175, pp. 334-341

Isard, M. & Blake, A., 1998. "Condensation - conditional density propagation for visual tracking", Int. Journal on Computer Vision 29 (1998) pp. 5–28.

Jojic, N., Brumitt, B., Meyers, B., Harris, S. & Huang, T., 2000. "Detection and estimation of pointing gestures in dense disparity maps", in Proc. Int. Conf. on Automatic Face and Gesture Recognition, 2000, pp. 468-475.

Julier, S. & Uhlmann, J., 1997. "A nondivergent estimation Algorithm in the presence of unknown correlations", in: Proc. American Control Conference, Vol. 4, IEEE, 1997, pp. 2369–2373.

Martin, C., Boehme, H.-J. & Gross, H.-M., 2004. "Conception and realization of a multi-sensory interactive mobile office guide", in: Proc. IEEE Conf. on Systems, Man and Cybernetics (SMC), 2004, pp. 5368–5373.

Martin, C., Schaffernicht, E., Scheidig, A. & Gross, H.-M., 2006. "Sensor Fusion using a Probabilistic Aggregation Scheme for People Detection and Tracking", Robotics & Autonomous Systems 54 (2006), pp. 721-728, Elsevier Science.

Nehaniv, C., Dautenhahn, K., Kubacki, J., Haegele, M., Parlitz, C. & Alami, R., 2005. "A methodological approach relating the classification of gesture to identification of human intent in the context of human-robot interaction.", in Proc. 14th IEEE Int. Workshop on Robot an Human Interactive Communication (RO-MAN), pp. 371-377, 2005

Nickel, K. & Stiefelhagen, R., 2003. "Real-time recognition of 3D pointing gestures for human-robot-interaction", in Proc. Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM) - Symposium 2003, pp. 557-565.

Noelker, C. & Ritter, H., 1998. "Illumination independent recognition of deictic arm postures", in Proc. 24th Annual Conf. of the IEEE Industrial Electronics Society 1998, pp. 2006-2011.

Pavlovic, V., Sharma, R.. & Huang, T., 1997. "Visual interpretation of hand gestures for human-computer interaction: A review", in IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) 1997

Riedmiller, M. & Braun, H., 1993. "A direct adaptive method for faster backpropagation learning: the RPROP algorithm", in Proc. Int. Conference on Neural Networks (ICNN), San Francisco, 1993, pp. 586-591

Rogalla, O., Ehrenmann, M., Zoellner, R., Becher, R. & Dillmann, R., 2002. "Using gesture and speech control for commanding a robot assistant", in Proc. 2002 IEEE Int. Workshop on Robot and Human Interactive Communication (ROMAN 2002), pp. 454-459.

Stoerring, M., Moeslund, T., Liu, Y. & Granum, E., 2004. "Computer vision-based gesture recognition for an augmented reality interface.", in Proc. 4th IASTED Int. Conference on Visualization, Imaging and Image Processing, pp. 766-771, 2004

Treptow, A., Cielniak, G., and Duckett, T. 2005. "Active people recognition using thermal and grey images on a mobile security robot." In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)., pages 3610–3615.

Triesch, J. & von der Malsburg, C., 2001. "A system for person-independent hand posture classification against complex backgrounds", in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.23 No.12 (2001), pages 1448-1453.

Viola, P. & Jones, M., 2001. "Rapid object detection using a boosted cascade of simple features", in Proc. Conference of Computer Vision and Pattern Recognition (CVPR), 2001, vol. 1, pp. 511-518