Robotic Vision:

Technologies for Machine Learning and Vision Applications

José García-Rodríguez
University of Alicante, Spain

Miguel Cazorla
University of Alicante, Spain

Information Science
REFERENCE

Managing Director:	Lindsay Johnston
Editorial Director:	Joel Gamon
Book Production Manager:	Jennifer Yoder
Publishing Systems Analyst:	Adrienne Freeland
Development Editor:	Christine Smith
Assistant Acquisitions Editor:	Kayla Wolfe
Typesetter:	Erin O'Dea
Cover Design:	Nick Newcomer

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Robotic vision: technologies for machine learning and vision applications / Jose Garcia-Rodriguez and Miguel A. Cazorla Quevedo, editors.
pages cm
Summary: "This book offers comprehensive coverage of the current research on the fields of robotics, machine vision, image processing and pattern recognition that is important to applying machine vision methods in the real world"-- Provided by publisher.
Includes bibliographical references and index.
ISBN 978-1-4666-2672-0 (hardcover) -- ISBN 978-1-4666-2703-1 (ebook) -- ISBN 978-1-4666-2734-5 (print & perpetual access) 1. Computer vision. 2. Pattern recognition systems. 3. Image processing. 4. Robotics--Human factors. I. Garcia-Rodriguez, Jose, 1970- II. Cazorla Quevedo, Miguel A., 1970-
TA1634.R63 2013
629.8'92637--dc23
2012029113

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

Chapter 15
Learning Robot Vision for Assisted Living

Wenjie Yan
University of Hamburg, Germany

Elena Torta
Eindhoven University of Technology, The Netherlands

David van der Pol
Eindhoven University of Technology, The Netherlands

Nils Meins
University of Hamburg, Germany

Cornelius Weber
University of Hamburg, Germany

Raymond H. Cuijpers
Eindhoven University of Technology, The Netherlands

Stefan Wermter
University of Hamburg, Germany

ABSTRACT

This chapter presents an overview of a typical scenario of Ambient Assisted Living (AAL) in which a robot navigates to a person for conveying information. Indoor robot navigation is a challenging task due to the complexity of real-home environments and the need of online learning abilities to adjust for dynamic conditions. A comparison between systems with different sensor typologies shows that vision-based systems promise to provide good performance and a wide scope of usage at reasonable cost. Moreover, vision-based systems can perform different tasks simultaneously by applying different algorithms to the input data stream thus enhancing the flexibility of the system. The authors introduce the state of the art of several computer vision methods for realizing indoor robotic navigation to a person and human-robot interaction. A case study has been conducted in which a robot, which is part of an AAL system, navigates to a person and interacts with her. The authors evaluate this test case and give an outlook on the potential of learning robot vision in ambient homes.

DOI: 10.4018/978-1-4666-2672-0.ch015

INTRODUCTION

The phenomenon of population ageing is becoming a serious problem of this century. According to the estimate of the U.S. Census Bureau, the American population aged over 65 will grow from 13% to 20% until 2030 (Hootman & Helmick, 2006). In Europe, more than 20% of the population will be beyond 60 by 2020 (Steg, Strese, Loroff, Hull, & Schmidt, 2006) and by 2050 this group will even exceed 37% (OECD, 2007). Ageing societies would benefit from the design of “intelligent” homes that provide assistance to the elderly (Steg et al., 2006). In this context the research field of robotics is focusing attention on AAL systems which refer to a set of technological solutions that permit the elderly population to maintain their independence at home for a longer time than would otherwise be possible (O’Grady, Muldoon, Dragone, Tynan, & O’Hare, 2010). Ambient homes will not only react passively, like turning on lights when the lighting condition changes, but they will also provide active help via home electronics, motorized actuators or - in the future - socially assistive robots. They can assist the elderly effectively in everyday tasks such as communication with the external world or the ambient system and can provide medicine and health check reminders in a proactive fashion.

A number of research topics are involved in the design of the functionalities of a socially assistive robot. Among them, robotic navigation and human-robot interaction are particularly relevant. Robotic navigation in ambient homes, in particular mutual positioning between the robot and a person, is an important task for a robot that strongly influences the quality of human-robot interaction. A robot should find a way to approach a target person after localization and go to the person without colliding with any obstacles, which is very challenging due to the complexity of real-home environments and the possible dynamical changes. A vision-based system is a potential way to tackle those challenges. Compared with other kinds of sensors, a vision system can provide far more information, good performance and a wide scope of usage at reasonable cost. A robot can perform different tasks and adapt its behavior by learning new features if equipped with sophisticated vision algorithms.

Human-robot interaction is a very broad research field. Therefore, in the context of this book chapter, we understand it as the study of how robots can communicate interactively with users. Computer vision algorithms are essentials for achieving this because they can be used to acquire feedback related to the user state during interaction. Unlike an industrial robot, that, in most cases, runs preprogrammed behaviors without being interactive, service robots should be able to adapt their behavior in real time for the purpose of achieving natural and easy interaction with the user. This requires the generation of appropriate verbal and non-verbal behaviors that allow the robot to participate effectively in communication. Vision algorithms can gather information about the user’s attention, emotion and activity, and allow the robot to evaluate non-verbal communication cues of the user. The benefits of non-verbal communication cues become apparent when the conversation is embedded in a context, or when more than two persons are taking part in a conversation. Particularly, head gestures are important for a smooth conversation, because cues signaled by head gestures are used for turn taking. But head gestures serve many more purposes; they influence the likability of the observer, communicate the focus of attention or the subject of conversation, and they can influence the recollection of the content of conversation (Kleinke, 1986).

In this chapter we aim at introducing the reader to the computer vision techniques used in a robotics scenario for Ambient Assisted Living (AAL) in the context of the European project KSERA: Knowledgeable SErvice Robots for Aging. Our project develops the functionalities of a socially assistive robot that acts as an intelligent interface between the assisting environment and the elderly person. It combines different vision-based methods for simultaneous person and robot localization, robot navigation, human-robot interaction with online face recognition and head pose estimation, and adapt those techniques in ambient homes. The system is able to detect a person robustly by using different features, navigate a robot towards the person, establish eye contact and assess whether the person is giving attention to the robot.

This chapter is organized as follows: The section "Related Works" presents a brief review of the state of the art of algorithms and technology related to robotics and AAL. Section "Methods" provides insight in the computer vision algorithms developed and applied in order to increase the skills of a socially assistive robot in Ambient Homes. Section "Case Study" describes a detailed case study of AAL systems, which combines the different robotic vision techniques for allowing the humanoid robot Nao (Louloudi, Mosallam, Marturi, Janse, & Hernandez, 2010) to navigate towards a person in a cluttered and dynamically changing environment and to interact with the person. Section "Conclusion and Future Research Directions" summarizes the findings providing an outlook on the potential of learning robot vision in ambient home systems and outlines the possible developments of the algorithms and methods introduced in this chapter.

RELATED WORKS

This section provides an overview of the state-of-the-art vision technologies related to our work, which combines localization, navigation, face detection and head pose estimation algorithms for use in an assisted living home environment. We show that these vision algorithms are efficient to let a robot help a person in an AAL environment.

Simultaneous Person and Robot Localization

Person tracking based on vision is a very active research area. For instance, stereo vision systems (Muñoz-Salinas, Aguirre, & Garcá-Silvente, 2007; Bahadori, Iocchi, Leone, Nardi, & Scozzafava, 2007) use 3D information reconstructed by different cameras to easily distinguish a person from the background. Multiple ceiling-mounted cameras are combined (Salah, et al., 2008) to compensate for the narrow visual field of a single camera (Lanz & Brunelli, 2008), or to overcome shadowing and occlusion problems (Kemmotsu, Koketsua, & Iehara, 2008). While these multi-camera systems can detect and track multiple persons, they are expensive and complex. For example, the camera system has to be calibrated carefully to eliminate the distortion effect of the lenses and to determine the correlations between different cameras. A single ceiling-mounted camera is another possibility for person tracking. West, Newman and Greenhill (2005) have developed a ceiling-mounted camera model in a kitchen scenario to infer interaction of a person with kitchen devices. The single ceiling-mounted camera can be calibrated easily or can be used even without calibration. Moreover, with a wide-angle view lens, for example a fish-eye lens, the ceiling-mounted camera can observe the entire room. Occlusion is not a problem if the camera is mounted in the center of the ceiling and the person can be seen at any position within the room. The main disadvantage of the single ceiling-mounted camera setup is the limited raw information retrieved by the camera. Therefore, sophisticated algorithms are essential to track a person.

There are many person detection methods on computer vision area. The most common technique for detecting a moving person is background subtraction (Piccardi, 2004), which finds the person based on the difference between an input and a reference image. Furthermore, appearance-based models have been researched in recent years. For instance, principal component analysis (PCA) (Jolliffe, 2005) and independent component analysis (ICA) (Hyvärinen & Oja, 2000) represent the original data in a low dimensional space by keeping major information. Other methods like scale-invariant feature transformation (SIFT) (Lowe, 2004) or a speeded-up robust feature

(SURF) (Bay, Tuytelaars, & Van Gool, 2006) detect interest points, for example using Harris corner (Harris & Stephens, 1988) for object detection. These methods are scale- and rotation invariant and are able to detect similarities in different images. However, the computation complexity of these methods is high and they perform poorly with non-rigid objects. Person tracking based on body part analysis (Frintrop, Königs, Hoeller, & Schulz, 2010; Hecht, Azad, & Dillmann, 2009; Ramanan, Forsyth, & Zisserman, 2007) can work accurately, but requires a very clear body shape captured from a front view. A multiple camera system has to be installed in a room environment to keep obtaining the body shape. The color obtained from the clothes and skin can be a reliable tracking feature (Comaniciu, Ramesh, & Meer, 2000; Muñoz-Salinas, Aguirre, & Garcá-Silvente, 2007; Zivkovic & Krose, 2004), but this has to be adapted quickly when the clothes or the light condition changes. The Tracking-Learning-Detection algorithm developed by Kalal, Matas and Mikolajczyk (2010) works for an arbitrary object, however, it requires an initial pattern to be selected manually, which is not possible in a real AAL setting.

Navigation as Part of HRI

People tend to attribute human-like characteristics to robots and in particular, to socially assistive robots (Siino & Hinds, 2004), (Kahn et al., 2012). Therefore, when the robot behavior does not match prior expectations, humans tend to experience breakdowns in human-robot interaction (e.g., Mutlu and Forlizzi, 2008). As a consequence, mobile robots that share the same space with humans need to follow societal norms in establishing their positions with respect to humans (Kirby, 2010). By doing so, they will likely produce a match between their mobile behavior and people's expectations (Syrdal, Lee Koay, & Walters, 2007). Models of societal norms for robot navigation are mostly derived from the observation of human-human interaction scenarios. For instance, Nakauchi and Simmons (2002) developed a control algorithm that allows a robot to stand in line using a model of personal space (PS) derived from observation of people standing in line. Similarly, Pacchierotti and Christensen (2007) define the robot's behavior in an avoidance scenario based on human-human proxemic distances derived from the work of Hall (1966) and Lambert (2004).

Navigation in dynamic and cluttered environments, such as ambient homes, in the presence of a human is still a challenge because the robot needs to cope with dynamic conditions while taking into account the presence of its human companion. Traditionally, two distinct approaches have been proposed to tackle this issue. The first approach includes societal norms in the navigation algorithm at the level of global path planning. As an example, Kirby (2010) described social conventions like personal space and tending to the right, as mathematical cost functions that were used by an optimal path planner to produce avoidance behaviors for a robot that were accepted by users. On the same line, Sisbot et al. (2010) introduce a human aware motion planning for a domestic robot that, besides guaranteeing the generation of safe trajectories, allows the robot to reason about the accessibility, the vision field and the preferences of its human partner when approaching him/her. The second approach to robotic navigation in the presence of a human includes societal norms at the level of reactive behaviors. As an example, Brooks and Arkin (2006) proposed a behavior-based control architecture for humanoid robot navigation that takes into account the user's personal space by introducing a proportional factor for mapping human-human interpersonal distances to human-robot distances. Along the same line, Torta et al. (2011) present a behavior-based navigation architecture that defines the robot's target positions through the solutions of a Bayesian filtering problem which takes into account the user's personal space. On the contrary of Brooks and Arkin, the model of the personal space was derived by means of a psychophysical experiment.

Face Detection and Head Pose Estimation

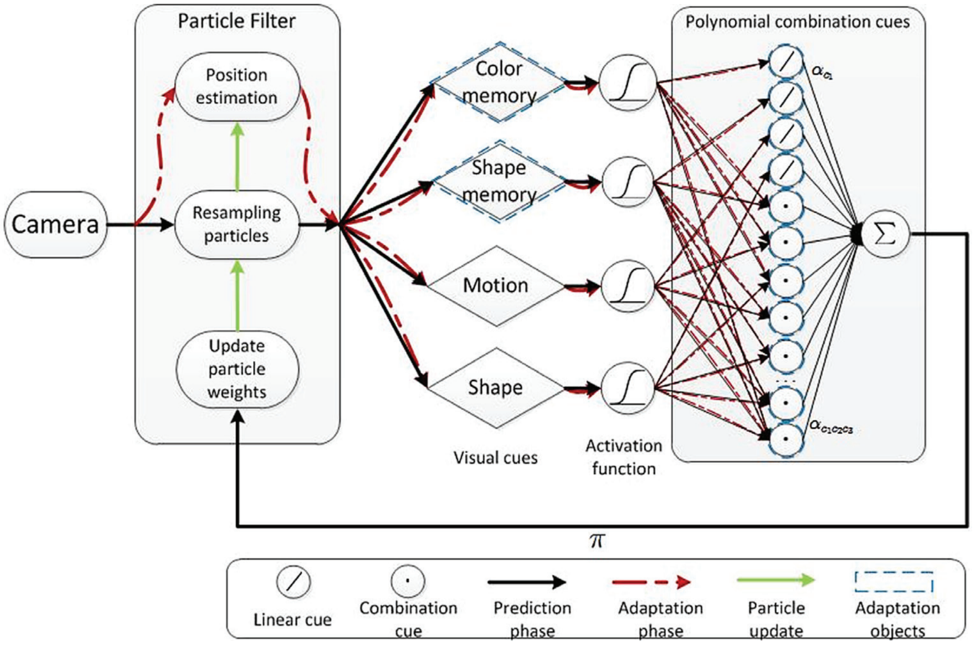
One of the most well known face detection methods for real-time applications is the one introduced by Viola and Jones (2002), which uses Haar-like features for image classification. Their work inspired subsequent research that extended the original method improving its accuracy. Some of them extended the spatial structure of the Haar-like features. For example Lienhart and Maydt (2002) have further developed the expression of the integral image that allows calculating Haar-like features which are rotated with respect to the original. Their method can cope better with diagonal structures. A different way to extend the original Haar-like features is presented by Mita, Kaneko and Hori (2005). In their work they join multiple threshold classifiers to one new feature, that they call "Joint Haar-like feature". By using this co-occurrence their method needs less Haar-like features for reaching the same accuracy. Beside the Viola and Jones method, other technologies and algorithms have been developed to solve the face detection problem. For instance, (Osuna, Freund, & Girosi, 1997) propose the use of support vector machines which yields a higher accuracy but is more computational expensive. Principal component analysis (Belhumeur, Hespanha, & Kriegman, 1997) was also used as well as convolution neural networks (Lawrence, Giles, Tsoi, & Back, 1997). The latter has the ability to derive and extract problem specific features.

Head gestures are specifically interesting for measuring the attention or engagement of the user related to what the robot is telling. Mutlu, Hodgins and Forlizzi (2006) have shown that the amount of gaze of the robot to the listener in a story telling context relates to the amount of information subjects remembered about the story. An advanced application of head pose estimation is joint attention. Joint attention entails reciprocal eye contact, but it can also be used to signal the subject of speech. Both interlocutors (the user and the robot) in this case focus on the same subject. This serves a communicative purpose (Kaplan & Hafner, 2006); by using the estimation of gaze direction, the robot is able to infer the object that the user refers to. For example, the user might ask the robot to "pick up that cup", while looking at the cup on the table. The system can now control the robot to pick up the cup on the table, and not the one on the mantelpiece. Yücel and Salah (2009) proposed a method for establishing joint attention between a human and a robot. A more advanced application that requires the robot to establish and maintain a conversation is turn taking during a conversation with its human partner (Kendon, 1967). To give the floor to the conversation partner, people often look at the partner just before finishing their turn. This can also be implemented in a robot by allowing the robot to estimate the gaze direction of the user. This allows the robot to use these cues to time its turn correctly. As robots are joining humans in everyday tasks, head pose estimation based on computer vision has seen a revival of interest. Different vision-based head pose estimation systems have been compared and summarized in Murphy-Chutorian and Trivedi (2009). Voit, Nickel and Stiefelhagen (2007) developed a neural network-based model for head pose estimation which has been further developed by van der Pol, Cuijpers and Juola (2011) and will be discussed in the "Head Pose Estimation" section.

METHODS

Here we present methods for realizing robotic navigation in an AAL environment as well as human-robot interaction. We first describe a hybrid neural probabilistic model for localizing a robot and a person using a ceiling-mounted camera. Then, we present a behavior-based model for robotic navigation which integrates the information provided by the localization algorithm with the robot's own sensor readings. A real-time face detection model is employed using the robot's camera to allow the robot to make eye contact

Figure 1. The person and robot localization algorithm using input data from the ceiling mounted camera. The weights of particles are assigned with a polynomial combination of visual cues.



with a person. The robot determines whether the user is paying attention to it with the head pose estimation method. The details of each model will be described in the following sections.

Simultaneous Person and Robot Localization

Inspired by a model of combining different information for face tracking (Triesch & Malsburg, 2001), we combine different visual information to detect and localize a person's position reliably. The system can track a person with or without motion information, and it is robust against environmental noise such as moving furniture, changing lighting conditions and interaction with other people. The work flow (Figure 1) can be split into two parts: prediction and adaptation.

In the prediction phase, each particle segments a small image patch and evaluates this patch using pre-defined visual cues. Four cues are used: 1) color memory cue based on a color histogram, 2) shape memory cue based on SURF features, 3) motion cue based on background subtraction and 4) fixed shape cue based on a neural network. When a person is detected in the image patch by for example the motion cue, the value of this cue will increase until it reaches a maximum value. The higher the visual cues' values are, the more likely is the target person present inside the image patch. We generate some extra polynomial combination cues using a Sigma-Pi network architecture (Weber & Wermter, 2007) to increase the weights when multiple cues are active at the same time. The output of the evaluation will be set to the particle filters, which provides robust object tracking based on the history of previous observations. Two particle filters are employed to estimate the position of a person and a robot. Particle filters are an approximation method that represents a probability distribution of an agent's state s_t with a set of particles $\{i\}$ and weight values $\pi^{(i)}$, which is usually integrated in partially observable Markov decision processes (POMDPs) (Kaelbling, Littman & Cassandra, 1998). A POMDP model consists of unobserved states of an agent *s*, in our case the *x,y* position of the observed person based on the image frame, and observations of the agent *z*. A transition model $P(s_t \mid s_{t-1}, a_{t-1})$ describes the probability that the state changes from s_{t-1} to s_t according to the executed action a_{t-1}.

If the agent plans to execute the action a_{t-1} in state s_{t-1}, the probability of the next state can be predicted by the transitions model $P(s_t \mid s_{t-1}, a_{t-1})$ and validated by the observation $P(z_t \mid s_t)$. Hence, the agent's state can be estimated then as:

$$P(s_t \mid z_{0:t}) = \eta P(z_t \mid s_t) \int P(s_{t-1} \mid z_{0:t-1}) \\ P(s_t \mid s_{t-1}, a_{t-1}) ds_{t-1}, \tag{1}$$

where η is a normalization constant, $P(z_t \mid s_t)$ is an observation model and $P(s_t \mid z_{0:t})$ is the belief of the state based on all previous observations. This distribution can then be approximated with weighted particles as:

$$P(s_t \mid z_{0:t}) \approx \sum_i \pi_{t-1}^{(i)} \delta(s_t - s_{t-1}^{(i)}), \tag{2}$$

where π denotes the weight factor of each particle with $\sum \pi = 1$ and δ denotes the Dirac impulse function. The higher the weight value, the more important this particle is in the whole distribution. The mean value of the distribution can be computed as $\Sigma_i \pi_{t-1}^{(i)} s_t$ and may be used to estimate the state of the agent if the distribution is unimodal.

In the person tracking system, the person's position is represented by the *x*- and *y*- coordinates in the image, i.e. $s = \{x, y\}$. The direction of a person's motion is hard to predict, because for example, an arm movement during rest could be wrongly perceived as a body movement into the corresponding direction. Hence, we do not use direction of movement information, but describe the transition model $P(s_t \mid s_{t-1}^{(i)}, a_{t-1})$ of the person with a Gaussian distribution:

Figure 2. Vision-based localization. Left: in the initial state particles are uniformly distributed in the image space. Blue circles represent particles describing the robot's pose while yellow circles represent particles for describing the person's location. Right: after initialization particles converge to the true person and robot locations

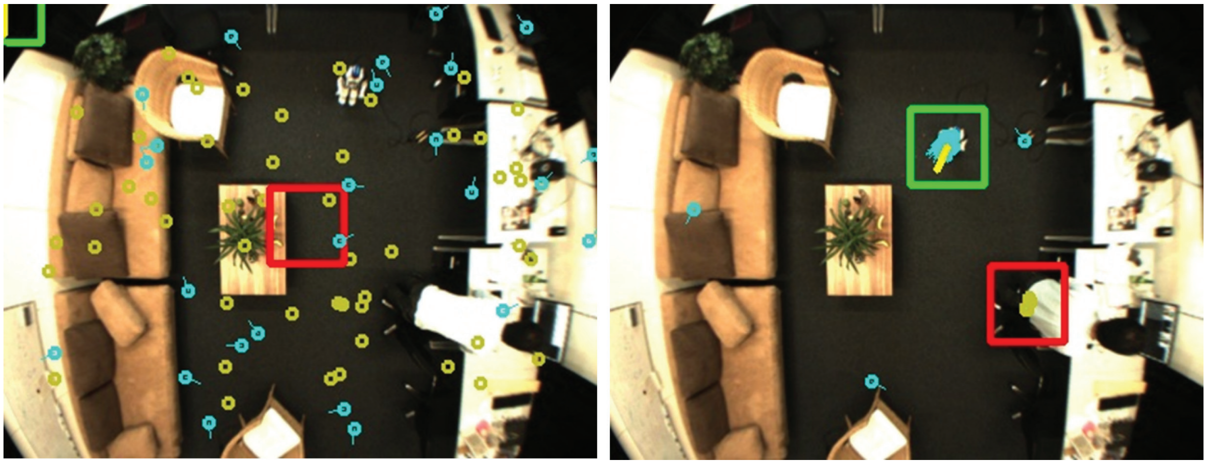
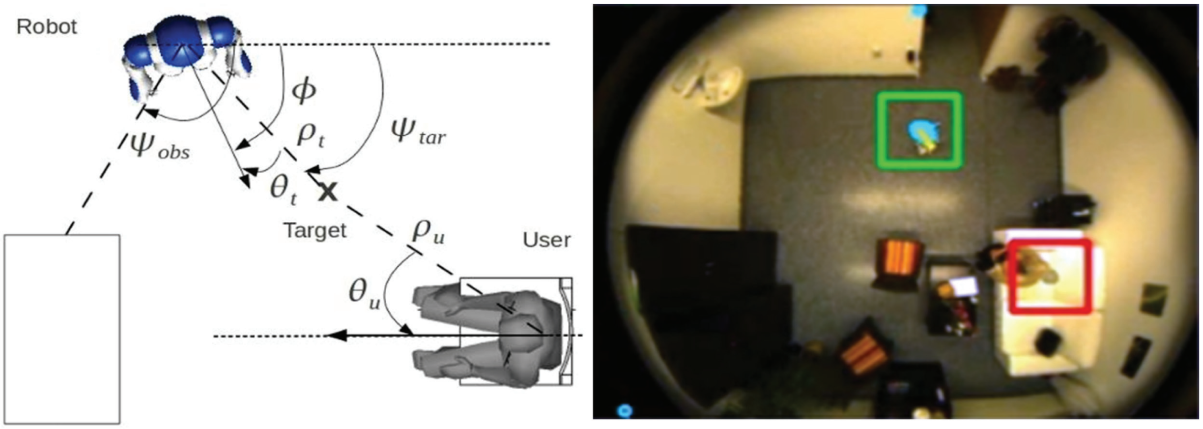


Figure 3. The equations of navigation rely on the visually obtained information from the localization module and on the information of the robot's proximity sensors. Left: relevant angles for the navigation algorithm. Right: view from the ceiling mounted camera.



$$P(s_t \mid s_{t-1}^{(i)}, a_{t-1}) = \frac{1}{\sqrt{2\pi\sigma(a_{t-1})^2}} e^{-\frac{(s_{t-1}^{(i)}-s_t^{(i)})^2}{2\sigma(a_{t-1})^2}}, \tag{3}$$

where $\sigma(a_{t-1})^2$ is the variance of the Gaussian related to the action, a_{t-1}, $s_{t-1}^{(i)}$ are the previous states and $s_t^{(i)}$ is the current states. In case of a moving person, the action a_{t-1} is a "binary" variable containing only information whether the person is moving or not. The variance $\sigma(a_{t-1})^2$ will be set larger when motion is detected, which allows particles to move further, and set to a small value to "stick" the particles on the current position when no motion is detected.

On the other hand, since we know precisely which actions the robot executes, the transition model $P(s_t \mid s_{t-1}^{(i)}, a_{t-1})$ of the robot can be built based on the robot's physical motion behavior. Therefore for the robot localization, the robot's state consists of three coordinate information: the *x*,*y* position and the orientation ϕ, i.e. $s = \left\{x, y, \phi\right\}$. As shown in Figure 2, the particles of the robot in cyan do not only encode the position coordinate as the person's particles, but also have the orientation information visualized by a short line. We can calculate the expected position of the robot x', y' and o' based on the designed feed-forward model and add Gaussian noise, as described by Equation (3).

When the person's and the robot's position are estimated, the particles will be resampled and their position will be updated (green arrows in Figure 1). After that, in the adaptation phase, the weight factor $\pi^{(i)}$ of particle *i* will be computed with a weighted polynomial combination of visual cues, inspired by a Sigma-Pi network (Weber & Wermter, 2007). The combination increases the effective number of cues and thereby leads to more robustness. The activities of the different visual cues are set as the input of the Sigma-Pi network and the particle weights are calculated as:

$$\begin{aligned}\pi^{(i)} = {} & \sum_{j=1}^{4} \alpha_{c_j}^{l}(t) A_{c_j}(s_{t-1}^{(i)}) + \\ & \sum_{\substack{j,k=1\\ j>k}}^{4} \alpha_{c_j c_k}^{q}(t) A_{c_j}(s_{t-1}^{(i)}) A_{c_k}(s_{t-1}^{(i)}) + \\ & \sum_{\substack{j,k,l=1\\ j>k>1}}^{4} \alpha_{c_j c_k c_l}^{c}(t) A_{c_j}(s_{t-1}^{(i)}) A_{c_k}(s_{t-1}^{(i)}) A_{c_l}(s_{t-1}^{(i)}),\end{aligned} \tag{4}$$

where $A_c(s_{t-1}^{(i)}) \in \left[0,1\right]$ is the activation function that signals activity of cue *c* at the state s_{t-1} (i.e. the position) of particle *i*. The activities of visual cues are generated via activation functions and scaled by their reliabilities α_c. We use a sigmoid activation function:

$$A(x) = \frac{1}{1 + e^{-(g \cdot x)}}, \tag{5}$$

Here, *x* is the function input and *g* is a constant scale factor. Through the polynomial combination of cues represented by a Sigma-Pi network, the weights of particles are computed. The coefficient of the polynomial cues, i.e. the network weights $\alpha_{c_j}^{l}(t)$ denote the linear reliabilities, $\alpha_{c_j c_k}^{q}(t)$ the quadratic combination reliabilities and $\alpha_{c_j c_k c_l}^{c}(t)$ are the cubic. Compared with traditional multi-layer networks, the Sigma-Pi network contains not only the linear input but also the second-order correlation information between the input values. The reliability of some cues, like motion, is non-adaptive, while others, like color, need to be adapted on a short time scale. This requires a mixed adaptive framework, as inspired by models of combining different information (Bernardin, Gehrig, & Stiefelhagen, 2008; Weber & Wermter, 2007). An issue is that an adaptive cue will be initially unreliable, but when adapted may have a high quality in predicting the person's position. To balance the changing qualities between the different cues, the reliabilities will be evaluated with the following equation:

$$\alpha_c(t) = (1 - \varepsilon)\alpha_c(t-1) + \varepsilon(f(s_t') + \beta), \tag{6}$$

where ε is a constant learning rate and β is a constant value. $f(s_t')$ denotes an evaluation function and is computed by the combination of visual cues' activities:

$$f_c(s_t') = \sum_{i \neq c}^{n} A_i(s_t') A_c(s_t'), \tag{7}$$

where s_t' is the estimated position and *n* is the number of the reliabilities. In this model *n* is 14 and contains 4 linear, 6 quadratic and 4 cubic combination reliabilities. We use a Hebbian-like learning rule to adapt the reliabilities. When the cue *c* is active together with several others, the function $f_c(s_t')$ is large, which leads to an increase of the cue's reliability α_c. For details of each visual cue please refer to (Yan, Weber, & Wermter, 2011).

Behavior-Based Robot Navigation

The general view of behavior-based robotics states that complex behaviors can be generated by the coordination of simpler ones. In the case of mobile robot's navigation each simple behavior solves a navigational subtask without the need of high level world representation (Arkin, 1998; Althaus, Ishiguro, Kanda, Miyashita, & Christensen,

Figure 4. Misalignments of the robot with respect to the person's position

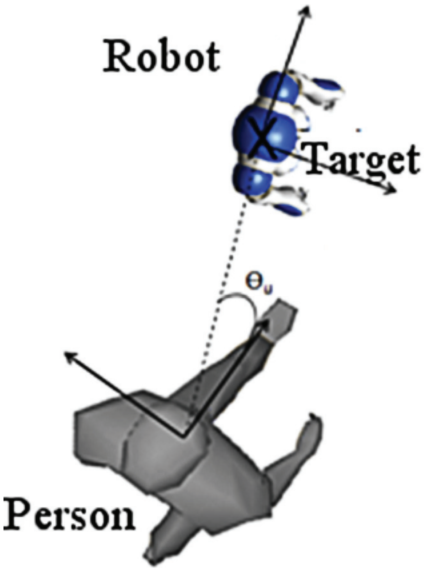
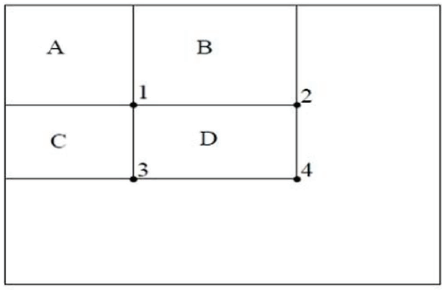
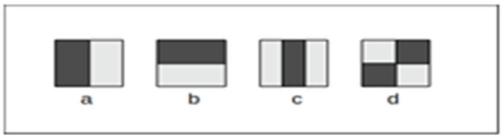


Figure 5. Left: integral image; right: haar-like feature





2004). Behavior-based robotics provides real-time adaptation to dynamically changing environments and can be adopted by robots with very limited sensory capabilities, such as humanoid robots (Bicho, 1999). These characteristics make this navigation framework convenient for applications in ambient homes.

There are several frameworks for behavior-based robotic navigation but here we focus on the dynamical system approach to mobile robot navigation (Schöner, Dose, & Engels, 1995; Bicho, 1999). This choice is due to the fact that the equations of the navigation algorithm do not rely on a complete world representation but on the visually obtained estimates of the user and robot positions, which can be obtained from the localization model. A behavior can be described by means of a behavioral variable that, in our work, is chosen to be the robot's heading direction $\phi(t)$, and by the temporal evolution of it. The evolution is controlled by a non-linear dynamical equation that can be generally expressed as:

$$\omega = \frac{d\phi(t)}{dt} = F(\phi(t)), \tag{8}$$

where $F(t)$ defines how the value of the behavioral variable $\phi(t)$ changes over time (Bicho, 1999), (Althaus et al., 2004). Multiple behaviors are aggregated by means of a weighted sum:

$$\frac{d\phi(t)}{dt} = \sum_{i=1}^{m} w_i f_i(\phi(t)) + d, \tag{9}$$

where *m* represents the number of behaviors that are needed for the robot to accomplish its task. The term $f_i(\phi(t))$ represents the force produced by the *i*th behavior and w_i represents the weight associated to the *i*th behavior. The term *d* represents a stochastic term that is added to guarantee escape from repellers generated by bifurcation in the vector field (Monteiro & Bicho, 2010). Attractor and repulsive functions, $f_i(\phi(t))$, are modeled with opposite signs. We can identify two basic behaviors whose coordination brings the robot from a generic location in the environment to a target location. The process of reaching a target point is represented by an attractor dynamic whose expression is:

$$f_1(t) = -\sin(\phi(t) - \psi_{tar}(t)), \tag{10}$$

where the term $(\phi(t) - \psi_{tar}(t))$ accounts for the angular location of the target with respect to the robot at time *t* and can be obtained by the visual estimation reported in the Section "Simultaneous Person and Robot Localization". The attractor dynamic acts to decrease the difference between $\phi(t)$ and ψ_{tar}; a graphical representation of those angles is visible in Figure 3.

The ability to obtain collision-free trajectories is encoded by a repulsive dynamic whose mathematical expression is given by:

$$f_2 = e\underbrace{\left(-\frac{1}{\beta_2}(d_{obs}(t)-R)\right)}_{term\ A} \underbrace{(\phi(t)-\psi_{obs}(t))e^{\left(-\frac{(\phi(t)-\psi_{obs}(t))^2}{2\sigma_{obs}^2}\right)}}_{term\ B}. \tag{11}$$

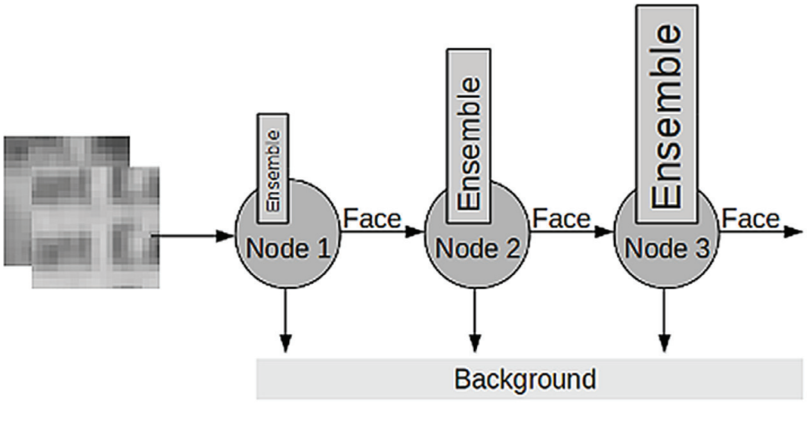
It generates a force which decays exponentially with the detected distance between the robot and the obstacle through the term A and with the angular separation between obstacle and robot thorough the term B. The detected distance between the robot and the obstacle at time *t* is represented by the term $d_{obs}(t)$, while the direction of the obstacle with respect to the robot at time *t* is encoded in the term $(\phi(t)-\psi_{obs}(t))$. The location of the obstacle with respect to the robot and the distance from the obstacles can be obtained from the robot's proximity sensors. The coefficients β_2 and σ_{obs} determine the range at which the repulsion strength acts. The repulsion force acts to increase the terms $(\phi(t)-\psi_{obs}(t))$ and $d_{obs}(t)$. A graphical visualization of $\psi_{obs}(t)$ is visible in Figure 3.

Referring to the stage of behaviors coordination reported in Equation(9), it is possible to obtain collision-free trajectories, if the weight w_2 associated with the repulsion force is greater than the weight w_1 associated with the attractor force. The aforementioned equations allow the robot to move towards a person and to avoid collisions on the way based on the localization estimate of the person and the robot and on the information of the robot's proximity sensors. We suppose to fix the target's position with respect to the user reference frame at a point in front of him located in the user's personal space described by:

$$X = (\rho_u \cos(\theta_u), \rho_u \sin(\theta_u)), \tag{12}$$

where ρ_u and θ_u represent the target point expressed in polar coordinates. While the coordinates of the target point are fixed with respect to the user reference frame, their expression with respect totherobotreferenceframe $(\rho_u \cos(\theta_u), \rho_u \sin(\theta_u))$ changes as the user moves. Therefore, knowing the position of the user with respect to the robot, allows us to derive the position of the target point with respect to the robot and from there to derive the motion equation that brings the robot in a position for facing the user.

Figure 6. Face detection ensemble



Algorithm 1. AdaBoost Algorithm

```
Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$
for $t = 1, \ldots, T$ do
  1. Normalize the weights, $w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$ so that $w_t$ is a probability distribution.
  2. For each feature, $j$, train a classifier $h_j$ which is restricted to using a single feature. The error is
     evaluated with respect to $w_i$, $\epsilon_j = \sum_i |h_j(x_i) - y_i|$.
  3. Choose the classifier, $h_t$ with the lowest error $\epsilon_j$.
  4. Update the weithgs: $w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$ where $e_i = 0$ if example $x_i$ is calssified correctly, $e_i = 1$
     otherwise, and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.
end for
The final strong classifier is:
```

$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \leq \frac{1}{2}\sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases} \quad \text{where } \alpha_t = \log \frac{1}{\beta_t}$$

Once the robot has reached its target point, its orientation might not be suitable to start interacting. For example it could not be facing the user since the equations we described before do not control the robot's final orientation (see Figure 4). For this reason, once the robot stops, it looks for the face of the person and adjusts its body orientation according to the detected location of the face in its visual field. This passage requires face detection techniques then the user interaction can start. This is described in the next section.

Human Robot Interaction

A fundamental prerequisite for smooth human-robot interaction is joint attention. Joint attention means that both the person and the robot jointly focus their attention on a single object. The same applies to eye contact, where human and robot mutually attend to each other. For this to happen the robot must first be able to localize a person's face and then estimate where a person is paying attention to. The feedback about the user's estimated head pose can be used to modify the robot's behavior with the purpose of achieving effective communication. In particular, if the robot wants to deliver a message to a person, and this person is not paying attention, the robot should attract attention using verbal or non-verbal cues. As soon as the person pays attention, the robot can deliver the message while monitoring whether the person is still paying attention to the robot. We focus on describing two typical tasks of non-verbal interaction between robot and user: building up eye-contact through face detection and estimating user's attention using head pose estimation. We apply computer vision methods to the robot's head camera to realize these functions.

Face Detection

Face detection can serve many different purposes in human-robot interaction and one of them refers to the correction of the robot's alignment. With this model, the robot is able to align its orientation with respect to the user's face position in the robot's visual field. Different computer vision methods have been developed for face detection and one of the most well-known was proposed by Viola and Jones (2002). This algorithm is based on

Haar-like features, threshold classifiers, AdaBoost and a cascade structure.

Haar-like features (Figure 5 right) are digital image features whose shapes are similar to Haar-Wavelets. A simple rectangle Haar-like feature can be described as the difference of the sum of the pixels within two rectangle areas (Viola & Jones, 2002). In order to speed up the computation, an intermediate representation of an image is processed which is called integral image (Figure 5 left). The transformation of an integral image can be performed with following equations:

$$\begin{aligned} s(x,y) &= s(x,y-1) + i(x,y) \\ ii(x,y) &= ii(x-1,y) + s(x,y), \end{aligned} \tag{13}$$

where (x,y) indicates the pixel position of an image, $s(x,y)$ is the cumulative row sum with $s(x,-1)=0$, $i(x,y)$ represents the value of the pixel at location (x,y) in the initial image, $ii(x,y)$ represents the value of the integral image at location (x,y) with $ii(-1,y)=0$.

The calculation of the Haar-like features can be simplified by computing four array references. As shown in Figure 5 left, the value of an integral image at location 1 is the sum of the pixels in rectangle *A* and the value at location 2 is the sum of pixels in rectangle (*A*+*B*). Similarly, the sum within area *D* equals then 4+1-(2+3) (Viola & Jones, 2002). Since the set of rectangle Haar-like features is overcomplete (Viola & Jones, 2002), an AdaBoost algorithm (Freund & Schapire, 1995) is employed to select a small number of significant features. A set of threshold classifiers are built using these Haar-like features. Each classifier has only a low detection rate therefore they are also called weak classifiers. To improve the classification results, the classifiers are combined with a cascade structure which rejects most of the background within few classification steps. The detailed AdaBoost algorithm is shown in Algorithm 1(Viola & Jones, 2002).

Once a face is detected in the robot's camera, we use a closed-loop control mechanism for centering the user's face in its visual field. Our humanoid robot's head has two degrees of freedom namely yaw and pitch. We control the yaw and the pitch angle to minimize the distance in the vertical and horizontal direction between the detected face and a relevant point in the image (X_t, Y_t). The information we gather from the face detection module is denoted as (X_0, Y_0) and is

Figure 7. Face tracking is based on the face's location in the robot's visual field. The tracking algorithm tends to minimize the distance between the position of the face in the robot's visual field and a relevant point in the image which in the case shown is the center of the visual field $(X_t = 0, Y_t = 0)$.

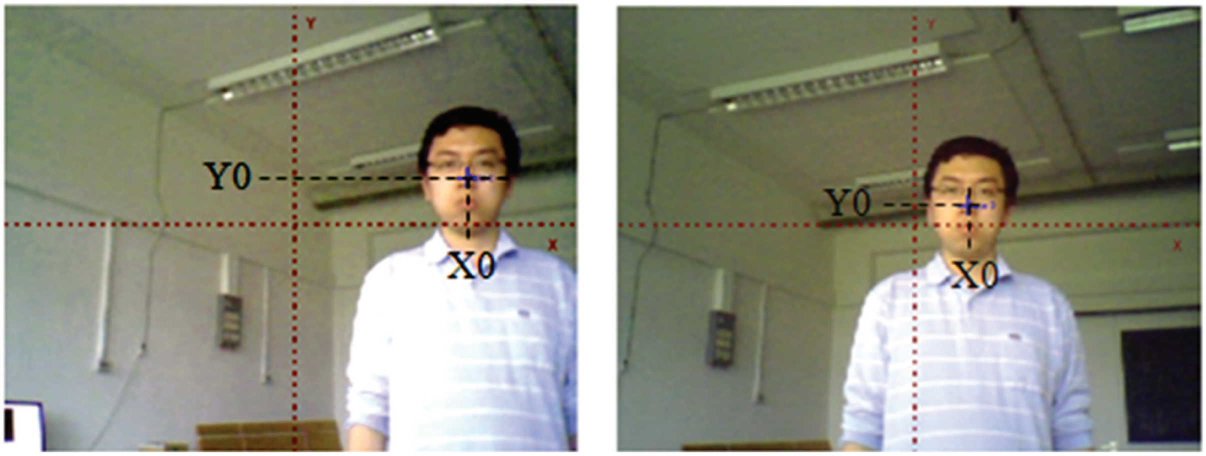
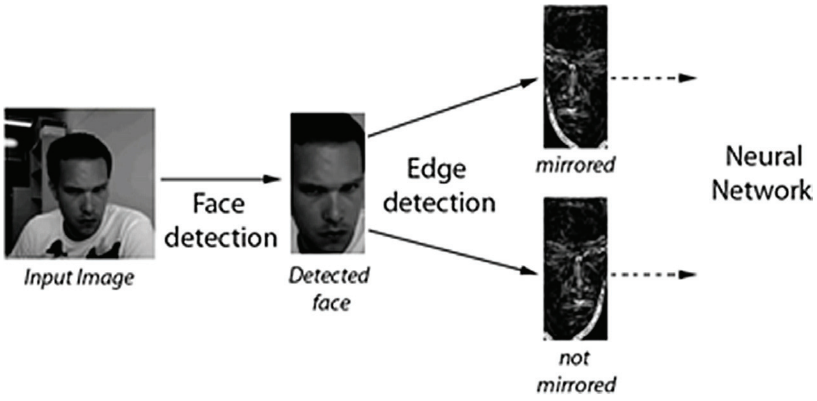


Figure 8. Face detection and image processing stages. Head Pose estimation requires face detection and image preprocessing, in particular edge detection, to the data acquired from the robot's camera.



reported in Figure 7. We generate control commands with a simple proportional action as described in the following equation:

$$\begin{aligned} C_{yaw} &= k_{yaw}(X_0 - X_t) \\ C_{pitch} &= k_{pitch}(Y_0 - Y_t), \end{aligned} \tag{14}$$

where k_{yaw} and k_{pitch} represent the strength of the proportional control action relative to the yaw and pitch angles of the robot's head.

Head Pose Estimation

Currently, several commercial methods are available for estimating a person's head pose (e.g. the face API, face.com). These methods are usually optimized for situations where a person is sitting behind his desktop computer: high-resolution images of nearby faces. In robotics the image quality is typically limited: (1) Small humanoid robots like Aldebaran's Nao typically have limited processing capacity available, because of the weight limitations. As a result the cameras do not have optics to improve image quality. (2) It is possible to process images on a fast remote machine, but this poses strong constraints on the bandwidth of the wireless connection. In practice, only low-resolution images are transmitted with sufficient refresh rate. (3) The robot, if autonomous, operates in hugely varying lighting conditions. Close inspection of Figure 7, for example, reveals large color differences although the scenes are very similar to the human eye. (4) In addition, a robot walking on the floor is never very close to the user. Thus, a person's face only covers a small part of the image (see Figure 7). The first two constraints are of a technical nature and can be remedied with more expensive equipment. The third and fourth constraints, however, are due to the different role a humanoid robot has when interacting with a person. Thus, an improved method is required to estimate head pose from a limited amount of visual information. Because of these reasons a neural network-oriented head pose estimation solution was developed based on the work by Voit et al. (2007), and further developed by van der Pol et al. (2011).

Image Processing

The image patch of the face as detected by the Viola and Jones' face detection method is preprocessed before feeding it into the neural network. The images are converted to black and white and scaled and cropped to 30 pixels wide and 90 pixels tall images containing only the facial region. The image aspect ratio is rather tall than wide, because this way the image contains only parts of the face even when it is rotated over large angles. After adjusting the image to the appropriate size, the image is filtered with a Laplacian edge detection filter. This filter is a translation- and rotation-invariant detector for contrast. The Laplacian works like an edge detector and behaves similarly to a combination of Sobel filters or a Canny edge detector. To be able to average over different data gathered from the neural network, the mirrored image is also passed through to the next level, as well as a cutout one pixel bigger in all directions and its mirrored image.

Neural Network

Neural networks are computationally expensive to train, but they are efficient after training. Other solutions often detect certain features within the face to calculate the rotation by using a three-dimensional model of the head (Murphy-Chutorian & Trivedi, 2009). This requires a high-resolution image for robust detection of the fine textures that define these features. Therefore, it also becomes very dependent on the lighting condition. Our neural network-based approach does not use these features, nor does it rely on a three-dimensional model.

The network is trained using a training set of 5756 images some of them coming from the database created by Gourier, Hall and Crowley (2004). Multiple networks are trained for both yaw (horizontal orientation) and pitch (vertical orientation) angles (see Figure 9). The Levenberg-Marquardt training method is used to train the two-layer, feed-forward neural network. Note that the roll angle cannot be estimated by the neural network, because the face detection algorithm is only able to detect faces that are upright. While training neural networks, especially complex ones like this, chances are high to end up at a local minimum. Therefore, the output of different networks is likely to differ. This method uses the best ten networks to increase performance. The output of the networks is multiple estimates of pitch and yaw angles. After averaging over all of these values the estimate of the head pose is retrieved (see Figure 10).

Caveats

Currently, the utilized face detection cascade for Viola and Jones' object (2002) detection is limited to detect a face when most of the face is visible excluding side views. Consequently, our method currently works for faces rotated less than 90 degrees from looking straight at the camera, in any direction. Head pose is only an estimator of gaze direction as human gaze is also determined by the orientation of the eyes. Nonetheless, head pose is a good estimator for gaze, because when people attend to something for some time they always turn their heads. The performance of our method in real life heavily depends on the training set that was used because the variation of, say, lighting condition in the training set affects the robustness against varying lighting conditions during implementation. We find an average error for the yaw estimation of about 7 degrees. For the pitch angle the average error is approximately 11

Figure 9. Pitch, yaw, and roll angles

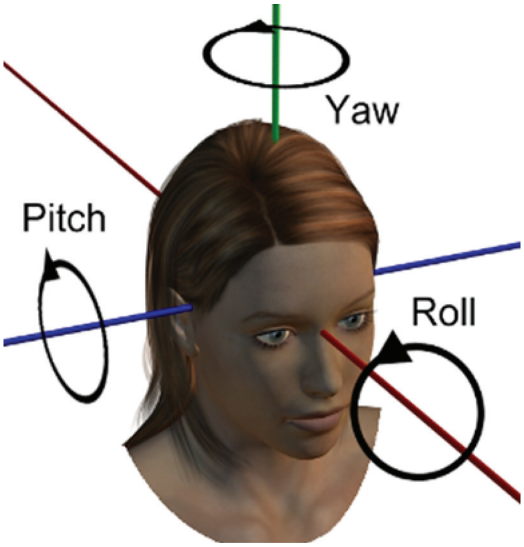
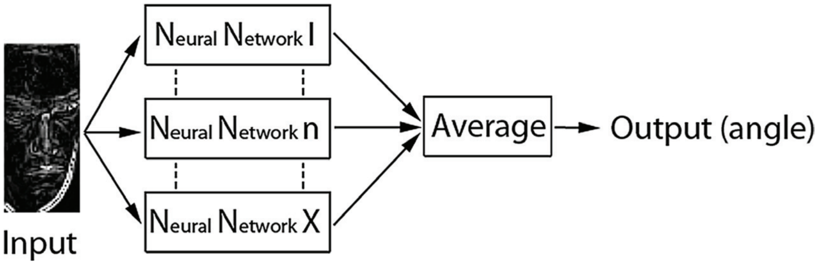


Figure 10. Two-layer feed forward neural network for head pose estimation. Image shows individual networks that combined give an estimate of the yaw angle of rotation.



degrees. These results are sufficiently precise for social interaction and they show that our method works from a low vantage point with varying lighting conditions and low resolution images.

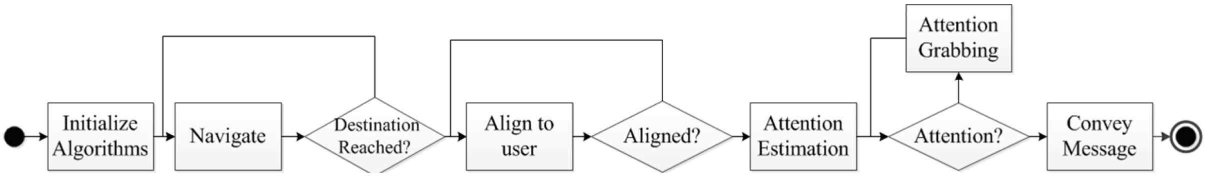
CASE STUDY

Effective assistance refers to both, the ability of the robot to interact with the environment and the ability of the robot to interact with the user as those two aspects are tightly coupled. Think about a robot that is able to move safely in a cluttered environment but that is not able to know where its human companion is or where (s)he is looking at. Then the inclusion of such a robot in ambient homes would not provide any benefit because it would not be able to communicate with the person and remind the user to check some health parameters. On the other hand, in case of a static robotic assistant, the user would be obliged to go to the robot. But in this situation how can a robot function as a reminder assistant if the person herself needs to be active and remember to approach the robot?

These two examples motivate that navigation capabilities of a mobile robot and its ability to acquire information about the user are coupled and they are essential for designing robotics applications for ambient homes. Effectiveness of interaction with the environment and the user can only be achieved by the acquisition of real time

Figure 11. The case study represents tests of the KSERA system done in the Schwechat (Vienna) Nursing Home. For privacy reason we have modified the faces to be unrecognizable.

Figure 12. The actions of various systems components are coordinated with finite element state machines. The figure represents the action sequence that the robot needs to accomplish for going to a person and interacting with her.



feedback on the environment's and the user's conditions. This feedback can be obtained by the joint application of the computer vision methods presented in the previous sections. Therefore this section presents a case study related to the introduction of a humanoid robot in an AAL application and provides a concrete example of how the computer vision techniques introduced in the previous sections can be integrated for employing a robot in an assisting environment (Figure 11). The robot should navigate autonomously towards a person, establish eye contact, check whether the user is focused on the robot and then ask the user to perform the measurement of his oxygen level. The test case refers to currently ongoing testing of the KSERA system in Schwechat Vienna.

In our case study, we run experiments with different users and one of our test cases is illustrated in Figure 12. This test case consists of two parts: (1) navigate safely towards a person and (2) human robot interaction using nonverbal communication. The event flow for achieving points (1) and (2) is represented as a state machine diagram. At first all the components are initialized, and then the robot navigates towards the person until it reaches its target point. At that moment it tries to establish eye contact with the user modifying its body pose. It then checks the focus of attention of the user and if the user is paying attention it asks the user to measure his oxygen level. We start by letting the robot approach a localized person. Two groups of particles with different setups for person and robot localization are initialized at

Figure 13. Simultaneous person and robot tracking. Left: initial position, Right: robot's movement towards a person.

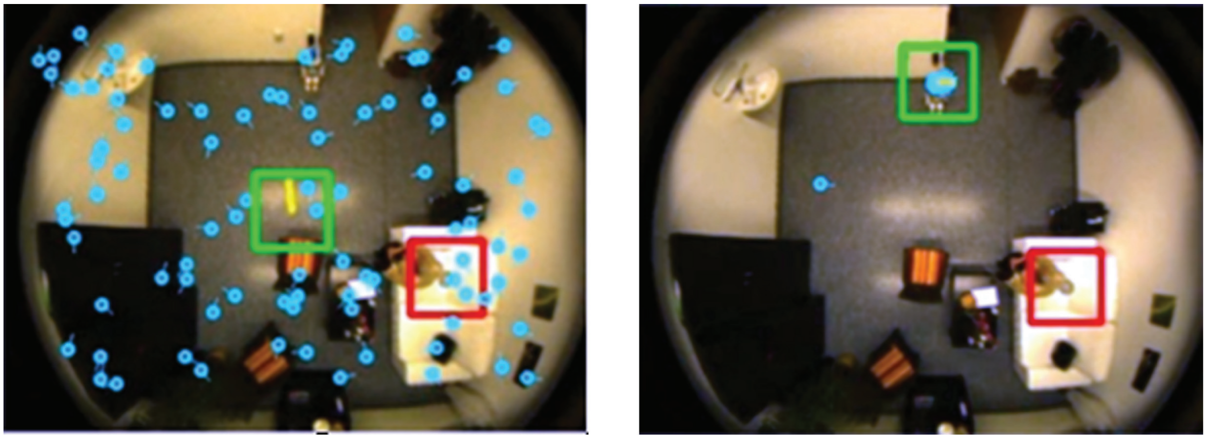
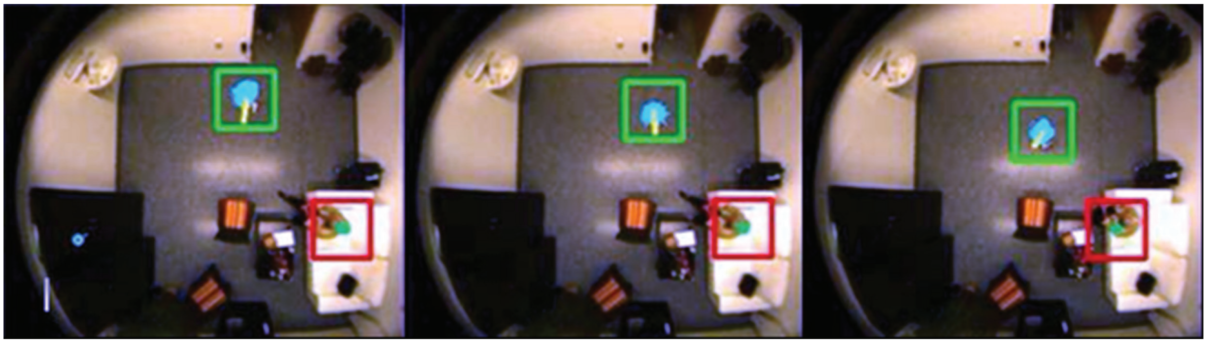


Figure 14. Robot's movements towards a person. After moving for a short distance the robot stops in a proper location for approaching the user as identified by (Torta et al., 2011).



random position in the image space at first (see Figure 13 left).

The person is first localized using motion information. The weights of particles close by the user increase and the particle cloud converges to a single position (Figure 13 right). Meanwhile, the shape feature as well as the color histogram adapts to store features of the localized user, which enables the system to localize the person even when motion is missing. The person localization has been tested and evaluated according to the CLEAR MOT Metrics (Keni & Rainer, 2008). Since only a single person is tracked in the system, based on our goal design, the frame number of misses m and of false positives f_p has been counted and the multiple object tracking accuracy (MOTA) has been calculated. The test results are shown in Table 1 and for details please see (Yan, Weber, & Wermter, 2011).

For the robot's particles a precise feed-forward motion model has been built according to the robot's behavior. When the robot moves, the particles move also with different orientation. The particles with wrong orientation will fly away from robot's position and only the particles with correct angles can survive which helps the system to estimate robot's orientation. The robot moves from its initial position to the final position in front of the user based on the navigation Equation (8) and Equation (9). Readers interested

Figure 15. Face of the person in the robot's visual field before and after the robot's alignment. Direct eye contact can be built after the alignment.

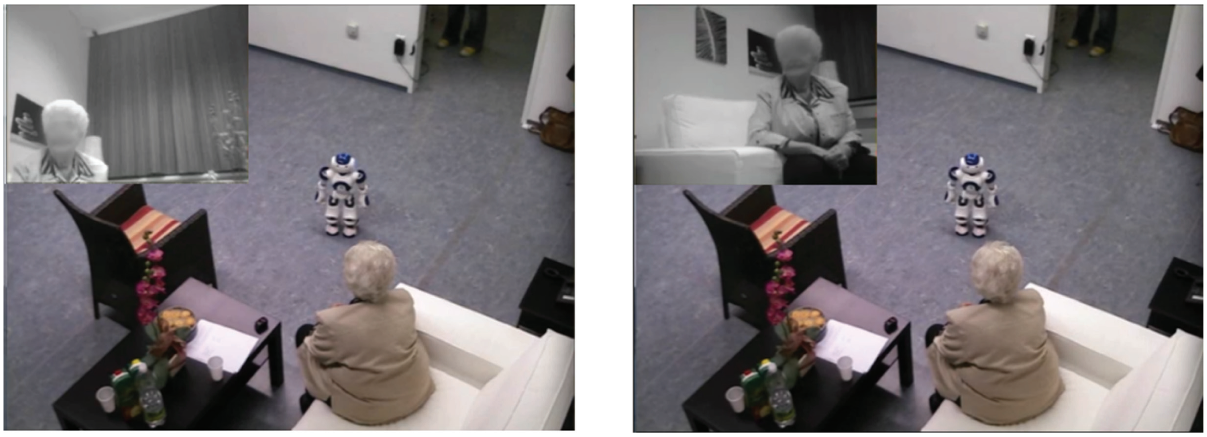


Table 1. Experimental results (Yan, Weber, & Wermter, 2011)

Name	**Total Frame**	m	fp	**MOTA (%)**
Person moving scenario 1	2012	19	22	97.96
Person moving scenario 2	2258	169	12	91.98
Person moving and sitting scenario 1	1190	78	21	91.68
Person moving and sitting scenario 2	980	22	130	84.18
Change environment scenario 1	1151	89	30	89.66
Change environment scenario 2	1564	157	141	80.94
Change light condition in night scenario	160	17	59	52.5
Change light condition in day scenario	540	0	3	99.45
Distracter person scenario 1	1014	48	35	91.81
Distracter person scenario 2	700	57	26	88.14
Distracter person scenario CLEAR 07	2122	188	52	88.68
Total	13691	844	531	89.96

in more complex navigation trajectories can refer to videos reported on the KSERA website http://www.ksera-project.eu. Personal space models have been defined with HRI psychophysical tests and are reported in Torta, Cuijpers, Juola and van der Pol (2011).

Once the robot reaches its target point, its final orientation might be inappropriate for initiating the interaction, as can be seen in Figure 14, because the robot is not able to establish eye contact with the user. In this case the robot can look for the user's face using the algorithm reported in section "Human Robot Interaction" with face detection and head pose estimation and adjusts its body's orientation and head angle to make eye contact with the user. As can be seen in Figure 15, at the beginning the face of the user is not centered in the robot's visual field, but after applying the face tracking algorithm the robot centers the user's face thus aligning to him. The person's attention can be monitored by applying the head pose estimation method as described in section "Head Pose Estimation". If the person is not paying attention to the robot, the robot will generate actions to grab user's attention until the user focuses on the robot. Then the robot conveys a message and the test case ends.

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

The chapter gives an overview of vision algorithms used in a typical scenario of ambient assisted living. We have focused our attention on robot's navigation towards a simultaneously localized person and on human-robot interaction. We discussed challenges for robots in ambient homes as well as the benefits of computer vision for these applications compared to systems with different sensors. A hybrid probabilistic algorithm is described for localizing the person based on different visual cues. The model is to some extent indicative of a human's ability of recognizing objects based on different features. When some of the features are strongly disturbed, detection recovers by the integration of other features. The particle filter parallels an active attention selection mechanism, which allocates most processing resources to positions of interest. It has a high performance of detecting complex objects that move relatively slowly in real time.

We described a sound method for face detection, the Viola and Jones method. We used the coordinates of the user's face in the visual field for correcting the robot's orientation for facing the user. Moreover, we illustrated a head pose estimation method based on the use of multiple neural networks. Once trained, the head pose estimation is computationally inexpensive, requires only low quality images and is robust to non-optimal lighting conditions. This makes our approach, compared to other methods for head pose estimation, especially useful in robotics applications for ambient assisted living.

A case study has been included that provides a concrete example of how the computer vision techniques can be integrated for employing a robot in an assisting environment. The experiments have been conducted and the evaluation shows that intelligent computer vision algorithms using a distributed sensory network (camera and robot) can be merged for achieving more robust and effective robot behavior and improve human-robot interaction in an AAL environment significantly. Our research focus is currently on the localization of a single person and robot navigation based on computer vision technology. However, in a real home situation, multiple persons may appear in a room at the same time and the robot should be able to distinguish them. Hence, in future research we will attempt to extend our model for localizing multiple persons at the same time using the ceiling mounted camera. A sophisticated person recognition model would also be employed in this case to distinguish the target person from the rest based on visual cues obtained from the robot's camera. Intention recognition would be another interesting direction for improving human-robot interaction and for defining the robot's proactive behavior. Fundamental information for intention estimation can come from the ceiling camera and the person's localization method. Another approach is to add a neural-based model for facial emotion recognition so as to understand whether the user is happy or sad and then adapt the robot's interactive behavior. In addition, the elaboration of a novel robot navigation method without camera calibration would be a useful improvement of robotic applications in domestic environments. Camera calibration is essential to eliminate the distortion effect of the camera lens and to ensure the quality of coordinate transformation from the camera view to the real world, but makes the system hard to install by persons without professional knowledge. Therefore, we are considering a model based on neural planning that can learn room mapping from the person's spatial knowledge and plan the robot's movement based on the learned map.

In general, our results show that - although many solutions exist to particular detailed problems like face recognition, navigation and localization - robotic applications in domestic environments like AAL require a level of integration that currently does not exist. This research is only the first step in addressing this issue.

ACKNOWLEDGMENT

The research leading to these results is part of the KSERA project (http://www.ksera-project.eu) funded by the European Commission under the 7th Framework Programme (FP7) for Research and Technological Development under grant agreement n^{o}2010-248085.

REFERENCES

Althaus, P., Ishiguro, H., Kanda, T., Miyashita, T., & Christensen, H. (2004). Navigation for human-robot interaction tasks. *2004 IEEE International Conference on Robotics and Automation, ICRA'04*, Vol. 2, (pp. 1894-1900).

Arkin, R. C. (1998). *Behavior-based robotics*. Cambridge, MA: MIT press.

Ba, S., & Odobez, J. (2004). A probabilistic framework for joint head tracking and pose estimation. *17th International Conference on Pattern Recognition, ICPR 2004*, Vol. 4, (pp. 264-267).

Bahadori, S., Iocchi, L., Leone, G., Nardi, D., & Scozzafava, L. (2007). Real-time people localization and tracking through fixed stereo vision. *Applied Intelligence*, *26*, 83–97. doi:10.1007/s10489-006-0013-3

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In Leonardis, A., Bischof, H., & Pinz, A. (Eds.), *Computer Vision - ECCV 2006* (*Vol. 3951*, pp. 404–417). Lecture Notes in Computer Science. doi:10.1007/11744023_32

Belhumeur, P., Hespanha, J., & Kriegman, D. (1997). Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(7), 711–720. doi:10.1109/34.598228

Bernardin, K., Gehrig, T., & Stiefelhagen, R. (2008). Multi-level particle filter fusion of features and cues for audio-visual person tracking. In Stiefelhagen, R., Bowers, R., & Fiscus, J. (Eds.), *Multimodal Technologies for Perception of Humans* (*Vol. 4625*, pp. 70–81). Lecture Notes in Computer Science. doi:10.1007/978-3-540-68585-2_5

Bicho, E. (1999). *Dynamic approach to behavior-based robotics.* PhD thesis, University of Minho.

Brooks, A. G., & Arkin, R. C. (2006). Behavioral overlays for non-verbal communication expression on a humanoid robot. *Autonomous Robots*, *22*, 55–74. doi:10.1007/s10514-006-9005-8

Comaniciu, D., Ramesh, V., & Meer, P. (2000). Real-time tracking of non-rigid objects using mean shift. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2, (pp. 2142-2150).

Dautenhahn, K. W., Koay, K., Nehaniv, C., Sisbot, A., Alami, R., & Siméon, T. (2006). How may I serve you? A robot companion approaching a seated person in a helping context. *1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, (pp. 172-179).

Freund, Y., & Schapire, R. (1995). A desicion-theoretic generalization of on-line learning and an application to boosting. In Vitányi, P. (Ed.), *Computational Learning Theory* (*Vol. 904*, pp. 23–37). Lecture Notes in Computer Science. doi:10.1007/3-540-59119-2_166

Frintrop, S., Königs, A., Hoeller, F., & Schulz, D. (2010). A component-based approach to visual person tracking from a mobile platform. *International Journal of Social Robotics*, *2*, 53–62. doi:10.1007/s12369-009-0035-1

Gourier, N., Hall, D., & Crowley, J. (2004). Facial features detection robust to pose, illumination and identity. *2004 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 1, (pp. 617-622).

Hall, E. (1966). *The hidden dimension*. New York, NY: Doubleday.

Harris, C., & Stephens, M. (1988). A combined corner and edge detector. *Alvey Vision Conference*, Vol. 15, (p. 50). Manchester, UK.

Hecht, F., Azad, P., & Dillmann, R. (2009). Markerless human motion tracking with a flexible model and appearance learning. *IEEE International Conference on Robotics and Automation, ICRA '09*, (pp. 3173-3179).

Hootman, J., & Helmick, C. (2006). Projections of US prevalence of arthritis and associated activity limitations. *Arthritis and Rheumatism*, *54*(1), 226–229. doi:10.1002/art.21562

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, *13*(4-5), 411–430. doi:10.1016/S0893-6080(00)00026-5

Jolliffe, I. (2005). Principal component analysis. In Everitt, B., & Howell, D. (Eds.), *Encyclopedia of statistics in behavioral science*. New York, NY: John Wiley & Sons, Ltd. doi:10.1002/0470013192.bsa501

Kaelbling, L., Littman, M., & Cassandra, A. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, *101*(1-2), 99–134. doi:10.1016/S0004-3702(98)00023-X

Kahn, J. P., Kanda, T., Ishiguro, H., Gill, B. T., Ruckert, J. H., Shen, S., et al. (2012). Do people hold a humanoid robot morally accountable for the harm it causes? *The Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction* (pp. 33-40). New York, NY: ACM.

Kalal, Z., Matas, J., & Mikolajczyk, K. (2010). P-N learning: Bootstrapping binary classifiers by structural constraints. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (pp. 49-56).

Kaplan, F., & Hafner, V. V. (2006). The challenges of joint attention. *Interaction Studies: Social Behaviour and Communication in Biological and Artificial Systems*, *7*, 135–169. doi:10.1075/is.7.2.04kap

Kemmotsu, K., Koketsua, Y., & Iehara, M. (2008). Human behavior recognition using unconscious cameras and a visible robot in a network robot system. *Robotics and Autonomous Systems*, *56*(10), 857–864. doi:10.1016/j.robot.2008.06.004

Kendon, A. (1967). Some functions of gaze-direction in social interaction. *Acta Psychologica*, *26*, 22–63. doi:10.1016/0001-6918(67)90005-4

Keni, B., & Rainer, S. (2008). Evaluating multiple object tracking performance: The CLEAR MOT metrics. *EURASIP Journal on Image and Video Processing*, *2008*, 10.

Kirby, R. (2010). *Social robot navigation.* PhD Thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

Kleinke, C. (1986). Gaze and eye contact: A research review. *Psychological Bulletin*, *100*(1), 78. doi:10.1037/0033-2909.100.1.78

Lambert, D. (2004). *Body language*. London, UK: Harper Collins.

Lanz, O., & Brunelli, R. (2008). An appearance-based particle filter for visual tracking in smart rooms. In Stiefelhagen, R., Bowers, R., & Fiscus, J. E. (Eds.), *Multimodal Technologies for Perception of Humans* (*Vol. 4625*, pp. 57–69). Lecture Notes in Computer Science. doi:10.1007/978-3-540-68585-2_4

Lawrence, S., Giles, C., Tsoi, A., & Back, A. (1997). Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks*, *8*(1), 98–113. doi:10.1109/72.554195

Lienhart, R., & Maydt, J. (2002). An extended set of Haar-like features for rapid object detection. *2002 International Conference on Image Processing*, Vol. 1, (pp. 900-903).

Louloudi, A., Mosallam, A., Marturi, N., Janse, P., & Hernandez, V. (2010). *Integration of the humanoid robot nao inside a smart home: A case study*. The Swedish AI Society Workshop.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*, 91–110. doi:10.1023/B:VISI.0000029664.99615.94

Mita, T., Kaneko, T., & Hori, O. (2005). Joint haar-like features for face detection. *10th IEEE International Conference on Computer Vision*, Vol. 2, (pp. 1619-1626).

Monteiro, S., & Bicho, E. (2010). Attractor dynamics approach to formation control: Theory and application. *Autonomous Robots*, *29*, 331–355. doi:10.1007/s10514-010-9198-8

Muñoz-Salinas, R., Aguirre, E., & Garcá-Silvente, M. (2007). People detection and tracking using stereo vision and color. *Image and Vision Computing*, *25*(6), 995–1007. doi:10.1016/j.imavis.2006.07.012

Murphy-Chutorian, E., & Trivedi, M. (2009). Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(4), 607–626. doi:10.1109/TPAMI.2008.106

Mutlu, B., & Forlizzi, J. (2008). Robots in organizations: The role of workflow, social, and environmental factors in human-robot interaction. *The 3rd ACM/IEEE International Conference on Human Robot Interaction* (pp. 287-294). New York, NY: ACM.

Mutlu, B., Forlizzi, J., & Hodgins, J. (2006). A storytelling robot: Modeling and evaluation of human-like gaze behavior. *6th IEEE-RAS International Conference on Humanoid Robots*, (pp. 518 -523).

Nakauchi, Y., & Simmons, R. (2002). A social robot that stands in line. *Autonomous Robots*, *12*, 313–324. doi:10.1023/A:1015273816637

O'Grady, M., Muldoon, C., Dragone, M., Tynan, R., & O'Hare, G. (2010). Towards evolutionary ambient assisted living systems. *Journal of Ambient Intelligence and Humanized Computing*, *1*, 15–29. doi:10.1007/s12652-009-0003-5

OECD. (2007). *OECD demographic and labour force database*. Organisation for Economic Cooperation and Development.

Oskoei, A., Walters, M., & Dautenhahn, K. (2010). *An autonomous proxemic system for a mobile companion robot*. AISB.

Osuna, E., Freund, R., & Girosi, F. (1997). Training support vector machines: An application to face detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (p. 130).

Pacchierotti, E., Christensen, H., & Jensfelt, P. (2007). Evaluation of passing distance for social robots. *The 15th IEEE International Symposium on Robot and Human Interactive Communication* (pp. 315-320). IEEE.

Piccardi, M. (2004). Background subtraction techniques: A review. *2004 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4, (pp. 3099-3104).

Ramanan, D., Forsyth, D., & Zisserman, A. (2007). Tracking people by learning their appearance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*, 65–81. doi:10.1109/TPAMI.2007.250600

Salah, A., Morros, R., Luque, J., Segura, C., Hernando, J., & Ambekar, O. (2008). Multimodal identification and localization of users in a smart environment. *Journal on Multimodal User Interfaces*, *2*, 75–91. doi:10.1007/s12193-008-0008-y

Schöner, G., Dose, M., & Engels, C. (1995). Dynamics of behavior: Theory and applications for autonomous robot architectures. *Robotics and Autonomous Systems*, *16*(2-4), 213–245. doi:10.1016/0921-8890(95)00049-6

Siino, R. M., & Hinds, P. (2004). *Making sense of new technology as a lead-in to structuring: The case of an autonomous mobile robot* (pp. E1–E6). Academy of Management Proceedings.

Sisbot, E. A., Marin-Urias, L. F., Broquère, X., Sidobre, D., & Alami, R. (2010). Synthesizing robot motions adapted to human presence. *International Journal of Social Robotics*, *2*, 329–343. doi:10.1007/s12369-010-0059-6

Steg, H., Strese, H., Loroff, C., Hull, J., & Schmidt, S. (2006). *Europe is facing a demographic challenge Ambient Assisted Living offers solutions.*

Syrdal, D. S., Lee Koay, K., & Walters, M. L. (2007). A personalized robot companion? - The role of individual differences on spatial preferences in HRI scenarios. *RO-MAN 2007 - The 16th IEEE International Symposium on Robot and Human Interactive Communication* (pp. 1143-1148). IEEE.

Torta, E., Cuijpers, R., Juola, J., & van der Pol, D. (2011). Design of robust robotic proxemic behaviour. In Mutlu, B. A., Ham, J., Evers, V., & Kanda, T. (Eds.), *Social Robotics* (*Vol. 7072*, pp. 21–30). Lecture Notes in Computer Science Berlin, Germany: Springer. doi:10.1007/978-3-642-25504-5_3

Triesch, J., & Malsburg, C. (2001). Democratic integration: Self-organized integration of adaptive cues. *Neural Computation*, *13*(9), 2049–2074. doi:10.1162/089976601750399308

van der Pol, D., Cuijpers, R., & Juola, J. (2011). Head pose estimation for a domestic robot. *The 6th International Conference on Human-Robot Interaction, HRI '11*, (pp. 277-278).

Viola, P., & Jones, M. (2002). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137–154. doi:10.1023/B:VISI.0000013087.49260.fb

Voit, M., Nickel, K., & Stiefelhagen, R. (2007). Neural network-based head pose estimation and multi-view fusion. In Stiefelhagen, R., & Garofolo, J. (Eds.), *Multimodal Technologies for Perception of Humans* (*Vol. 4122*, pp. 291–298). Lecture Notes in Computer Science. doi:10.1007/978-3-540-69568-4_26

Voit, M., Nickel, K., & Stiefelhagen, R. (2008). Head pose estimation in single- and multi-view environments - results on the clear'07 benchmarks. In Stiefelhagen, R., Bowers, R., & Fiscus, J. (Eds.), *Multimodal Technologies for Perception of Humans* (*Vol. 4625*, pp. 307–316). Lecture Notes in Computer Science. doi:10.1007/978-3-540-68585-2_29

Walters, M., Dautenhahn, K., Boekhorst, R., Koay, K., Syrdal, D., & Nehaniv, C. (2009). An empirical framework for human-robot proxemics. In Dautenhahn, K. (Ed.), *New frontiers in human-robot interaction* (pp. 144–149).

Weber, C., & Wermter, S. (2007). A self-organizing map of sigma-pi units. *Neurocomputing*, *70*(13-15), 2552–2560. doi:10.1016/j.neucom.2006.05.014

West, G., Newman, C., & Greenhill, S. (2005). Using a camera to implement virtual sensors in a smart house. *From Smart Homes to Smart Care: International Conference on Smart Homes and Health Telematics,* Vol. 15, (pp. 83-90).

Yamaoka, F., Kanda, T., Ishiguro, H., & Hagita, N. (2010). A model of proximity control for information-presenting robots. *IEEE Transactions on Robotics*, *26*(1), 187–195. doi:10.1109/TRO.2009.2035747

Yan, W., Weber, C., & Wermter, S. (2011). A hybrid probabilistic neural model for person tracking based on a ceiling-mounted camera. *Journal of Ambient Intelligence and Smart Environments*, *3*(3), 237–252.

Yücel, Z., Salah, A., Merigli, C., & Mericli, T. (2009). Joint visual attention modeling for naturally interacting robotic agents. *24th International Symposium on Computer and Information Sciences*, (pp. 242-247).

Zivkovic, Z., & Krose, B. (2004). An EM-like algorithm for color-histogram-based object tracking. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, (pp. 798-803).