ACTIVE STEREO VISION-BASED MOBILE ROBOT NAVIGATION FOR PERSON TRACKING

V. Enescu, G. De Cubber, K. Cauwerts, S. A. Berrabah, H. Sahli
Vrije Universiteit Brussel
Pleinlaan 2, B-1050 Brussels, Belgium

M. Nuttin
Katholieke Universiteit Leuven
Celestijnenlaan 300B, B-3001 Heverlee-Leuven, Belgium

Keywords: active vision, person following, color tracking, particle filter, proposal distribution, mean shift, measurement model update, robot navigation, fuzzy logic behavior-based navigation, hybrid behavior-based navigation.

Abstract: In this paper, we propose a mobile robot architecture for person tracking, consisting of an active stereo vision module (ASVM) and a navigation module (NM). The first tracks the person in stereo images and controls the pan/tilt unit to keep the target in the visual field. Its output, i.e. the 3D position of the person, is fed to the NM, which drives the robot towards the target while avoiding obstacles. As a peculiarity of the system, there is no feedback from the NM or the robot motion controller (RMC) to the ASVM. While this imparts flexibility in combining the ASVM with a wide range of robot platforms, it puts considerable strain on the ASVM. Indeed, besides the changes in the target dynamics, it has to cope with the robot motion during obstacle avoidance. These disturbances are accommodated by generating target location hypotheses in an efficient manner. Robustness against outliers and occlusions is achieved by employing a multi-hypothesis tracking method - the particle filter - based on a color model of the target. Moreover, to deal with illumination changes, the system adaptively updates the color model of the target. The main contributions of this paper lie in (1) devising a stereo, color-based target tracking method using the stereo geometry constraint and (2) integrating it with a robotic agent in a loosely coupled manner.

1 INTRODUCTION

In the past, robot navigation was commonly based upon data coming from classical sensory equipment like ultrasonic and infrared sensors or laser range scanners. This approach is in sharp contradiction with nearly all biological examples (e.g. humans) where vision is the primary sensing modality. This biological example inspired scientists (Beardsley et al., 1995; Davison and Murray, 1998) to tackle the visual navigation problem and, during the last decade, visual navigation has gained significant importance.

The main problem in setting up a global control architecture for a mobile robot with an active vision control loop is that the frequency of the robot control loop (and certainly that of an eventual manipulator installed on the mobile agent) differs from the frequency of the vision control loop. This also leads to a second problem: the reusability of the developed control architecture on different robotic platforms. Due to the difficulties with the timing between different loops in the systems, most researchers (Davison and Murray, 1998) tune their control processes such that they work well for one specific robot. Unfortunately, the high coupling between the visual and the robot control loop yields a robot-dependent control architecture.

In this paper, we set up a global system architecture for a visually guided robot, which is independent from the specific robot hardware and kinematics. We do this by separating the visual processing (ASVM), the navigation control (NM) and the robot motion controller (RMC). As an application for such a robotic system, we chose the person following task, for which several problems have to be solved: person tracking, coping with the erratic motion of the target, stereo head control, 3D position estimation, robot navigation and the robot motion control. In the following, we address each of these problems.

For most tracking applications a Kalman filter is used, as it is a reliable and efficient tool. As a disadvantage, the Kalman filter can not handle multi-modal distributions as present in our problem. Therefore, we resort to particle filtering, a Monte Carlo method able to maintain multiple hypotheses about the target state in the presence of non-linearity and non-Gaussian dis-

Enescu V., De Cubber G., Cauwerts K., A. Berrabah S., Sahli H. and Nuttin M. (2005).
ACTIVE STEREO VISION-BASED MOBILE ROBOT NAVIGATION FOR PERSON TRACKING.
In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, pages 32-39
DOI: 10.5220/0001187300320039

turbances.

In general, the motion of the target in a target tracking scheme is modeled using a predefined model such as a constant speed or a constant acceleration model (Strens and Gregory, 2003). This leads to problems when humans need to be tracked, as they could have both models as well as unpredictable motions. Therefore, to cater for the erratic target motion, we propose an effective mechanism for generating target location hypotheses in the particle filter. Based on the current estimate of the system state, a PID controller determines the control signal to be applied to the pan/tilt stereo head to keep it aligned with the target.

For the 3D position estimation, some authors (Ping et al., 2001) use scaling as means to retrieve depth information, while others (Ghita and P.-F., 2003) use the depth from defocus measure. The most popular approach is however to make use of a stereo setup to estimate the distance to a target (Arsenio and Banks, 1999; Schlegel et al., 2000; Kuniyoshi and Rougeaux, 1999; Wilhelm et al., 2004). This is also the method we use here.

In the context of robot navigation, many algorithms have been proposed to solve the path planning problem, ranging from simple potential field methods (Koren and Borenstein, 1991) to biologically inspired neural networks (Franz and H.-A., 2000). We opted for a behavior based control architecture for the navigation module.

The RMC requires careful consideration of the robot kinematics and dynamics. As we wanted to build a system which is easily portable from one robot system to another, we decoupled the platform-dependent RMC from the ASVM and NM. The RMC is therefore not considered part of the system architecture and is not pursued further.

The remainder of this paper is organized as follows. First, an overview of the system architecture is given in Section 2. Then, in Section 3, the active vision module is extensively explained. Here the topics of color histogram matching, stereo geometry, the particle-filter based target tracking and camera control are discussed. The navigation module is introduced in Section 4, after which, in Section 5, we present some results. Finally, we conclude the paper in Section 6.

2 OVERVIEW

To achieve the task of person following, two main problems need to be solved:

- The vision system has to track the target person
- The robot has to navigate to the target person without bumping into obstacles

In fact, these two problems can both be considered as a coordinate system alignment problem. This can be explained using Figure 1. The objective of the visual tracking system is to align the coordinate system of the stereo camera such that the Z-axis points straight at the target, thus minimizing the relative pan and tilt angles of the stereo vision system, α and β respectively. On the other hand, the objective of the robot is to move towards the goal, and hence to align the robot coordinate system such that the Z_R-axis points straight at the target. This is in general not the case due to the movement of the target person, the inertia of the robot and because the robot has to avoid obstacles on its way. Thus, most of the time, $\alpha_R > 0$, where $\alpha_R = \angle(O_R Z_R, OZ)$. To be able to navigate in a complex environment with obstacles, a behavior based robot navigation was adopted, where one behavior leads the robot to the target, whereas another behavior enables the robot to avoid obstacles.

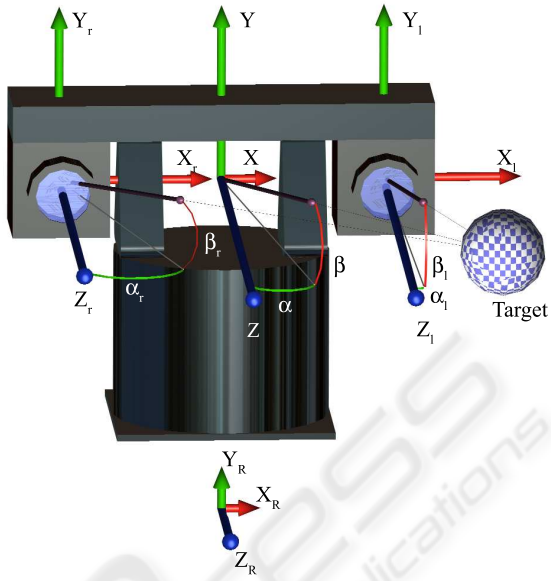

Figure 1: Definition of the coordinate systems: $(OXYZ)$ is the stereo head coordinate system, $(O_{l,r}X_{l,r}Y_{l,r}Z_{l,r})$ is the left/right camera coordinate system, and $(O_R X_R Y_R Z_R)$ is the robot camera coordinate system.

The general system architecture which integrates all the capabilities discussed above is sketched in Figure 2. On the left, one can observe the ASVM, which receives its input from the two cameras installed on the stereo head. At the heart of ASVM is a particle filter-based visual tracker which generates at each time step hypotheses (particles) about the 3D target position (in spherical coordinates) relative to the (XYZ) frame. These hypotheses are "projected" onto the image plane, resulting in a set of candidate target regions within the left and right images. For each pair of candidate regions, we compute two color histograms which we compare with

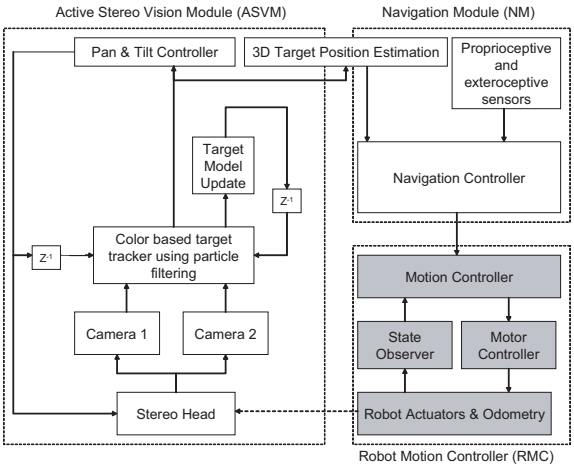

Figure 2: Overview of the system architecture

the base color histograms (left and right) serving as model for the tracked person. The likelihood of each hypothesis is then quantified by the matching degree between these histograms. The outcome of the tracking process is an estimate of the target position in the form of a probabilistic mixture of the target hypotheses. This estimate consists of the azimuth ($\hat{\alpha}$), elevation ($\hat{\beta}$), and range ($\hat{\lambda}$) of the target and its use is threefold. First, it serves for updating the base color histograms in view of coping with changing illumination conditions. Second, the target pose estimate is fed to the pan & tilt controller, which employs two PID controllers to ensure smooth and robust stereo camera control. Finally, the estimate is used to recover the absolute 3D position (relative to the robot frame) of the target person. This position estimator links the ASVM to the NM (see Figure 2). There, a navigation controller will distill a heading direction and speed from the absolute pan angle (i.e. $\alpha_R + \hat{\alpha}$), the target range ($\hat{\lambda}$) and the input from the proprioceptive and exteroceptive sensors. We have tested different behavior-based navigation controllers: a simple dual-behavior fuzzy logic-based navigation controller and a more elaborate hybrid architecture consisting of a deliberative and a reactive part. The final output on this level of the robot navigation module, a heading direction and a speed setpoint, is compatible with most robotic platforms, no matter what their kinematics and dynamics are. What follows further are robot-specific modules, indicated by the shaded blocks in Figure 2, which are not part of the presented architecture.

3 ACTIVE VISION

The ASVM accomplishes the following tasks:

1. tracking the 3D position of a person over time by means of the color properties of a region of his body; the color model of the target is updated in time to account for the variations in illumination;
2. control of the stereo head such that the person stays always in the field of view of the stereo head;

These tasks are detailed in the sequel of this section.

3.1 Dynamic model

The state of the target at time k is described by the vector $\mathbf{x}_k = (\alpha_k, \beta_k, \lambda_k)$ containing the spherical coordinates of the target with respect to the frame $(OXYZ)$ attached to the stereo head. α is the azimuth angle relative to OZ, β is the elevation angle relative to the plane (OXY), and λ is the target range.

Since a person may move in an unpredictable way, we adopt a weak state evolution model (inspired by (Pérez et al., 2004)) for the stereo head-target system. More specifically, we assume that the state vector components evolve according to mutually independent Gaussian random walk models, which we augment with uniform components to capture the possibly erratic motion of the target. Thus the state evolution model can be written as

$$p(\alpha_k|\alpha_{k-1}) = \varphi_1 \mathcal{N}(\alpha_k; \alpha_{k-1} + cu^p_{k-1}, \sigma_1^2) + (1-\varphi_1)\mathcal{U}(\alpha_k; -\alpha_m, \alpha_m) \quad (1)$$

$$p(\beta_k|\beta_{k-1}) = \varphi_2 \mathcal{N}(\beta_k; \beta_{k-1} + cu^t_{k-1}, \sigma_2^2) + (1-\varphi_2)\mathcal{U}(\beta_k; -\beta_m, \beta_m) \quad (2)$$

$$p(\lambda_k|\lambda_{k-1}) = \varphi_3 \mathcal{N}(\lambda_k; \lambda_{k-1}, \sigma_3^2) + (1-\varphi_3)\mathcal{U}(\lambda_k; \lambda_{min}, \lambda_{max}) \quad (3)$$

where u^p_k and u^t_k are the pan and tilt control inputs, c is a known coefficient, $\{\varphi_i\}_{i=1}^3 \in (0,1)$ are known mixing coefficients, $\mathcal{N}(x; \mu, \sigma^2)$ denotes a Gaussian distribution of variable x, mean μ, and variance σ^2, and $\mathcal{U}(x; x_{min}, x_{max})$ signifies that x is uniformly distributed between x_{min} and x_{max}. Note that $\sigma_{1,2,3}$, α_m, β_m, λ_{min} and λ_{max} are known by design. Since α_k, β_k, λ_k are independent variables, it follows that the state evolution distribution factorizes as :

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = p(\alpha_k|\alpha_{k-1})p(\beta_k|\beta_{k-1})p(\lambda_k|\lambda_{k-1}). \quad (4)$$

3.2 Stereo Geometry

The geometry of the stereo vision system is sketched in Figure 1. We track the 3D position of the target relative to the frame XYZ, (α, β, λ), using color measurements in the image plane. Hence, we need to find a relationship between (α, β, λ) and the 2D position of the point where the target projects on the image plane, for each image in a stereo pair. To this end, a

first step is to compute the azimuth and elevation angles of the target with respect to the coordinate frame attached to each camera, i.e. (α_l, β_l) for the left image and (α_r, β_r) for the right image. As this derivation is identical for α and β, we only present the solution for the azimuth angles here. From (Vieville, 1997, p. 30), we have that

$$\alpha_{l,r} = \arctan\left[u_0 + f\left(\tan(\alpha) \mp \frac{b}{2\lambda\cos(\alpha)}\right)\right] \quad (5)$$

where u_0 is the optical center of the camera, b is the baseline and f is the focal length.

Now, we shall relate $(\alpha_l, \beta_l, \alpha_r, \beta_r)$ to the position of the target projection in each image. Let $\mathbf{p}_l = (u_l, v_l)$ and $\mathbf{p}_r = (u_r, v_r)$ denote the position of the target projection on the left image $\mathbf{I}_l$ and on the right image $\mathbf{I}_l$, respectively. Given the geometry of the image formation, the following relations hold:

$$u_{l,r} = f \cdot D_u \cdot \tan(\alpha_{l,r}) \quad (6)$$

$$v_{l,r} = f \cdot D_v \cdot \tan(\beta_{l,r}) \quad (7)$$

where D_u and D_v represent the number of pixels per meter in horizontal and vertical direction, respectively. Thus, starting from (α, β, λ), we can determine $\mathbf{p}_l$ and $\mathbf{p}_r$ based on (5), (6), and (7). Let

$$(\mathbf{p}_l, \mathbf{p}_r) = \text{T2S}(\alpha, \beta, \lambda)$$

(T2S stands for "3D to stereo") be the function corresponding to these transformations. It is useful to also define $S2T()$, the inverse function of T2S():

$$(\alpha, \beta, \lambda) = \text{S2T}(\mathbf{p}_l, \mathbf{p}_r), \quad (8)$$

Alternatively, we refer to T2S as "projection" and to S2T as "back-projection."

3.3 Color-based measurement model

Initially, at time $k = 0$, the projections of the target on the image planes are delineated manually and described by means of two elliptical regions with the half axes H_u and H_v. Let these regions be denoted by $\mathcal{R}_{l,0} = \mathcal{R}(\mathbf{p}_{l,0}, 1)$ (in the left image) and $\mathcal{R}_{r,0} = \mathcal{R}(\mathbf{p}_{r,0}, 1)$ (in the right image), where $\mathcal{R}(\mathbf{p}, s)$ is an elliptical region of center $\mathbf{p}$ and scale factor s with respect to the initial ellipse (H_u, H_v). Subsequently, tracking the object throughout the stereo image sequence localizes the object at time k within the regions $\mathcal{R}_{l,k} = \mathcal{R}(\mathbf{p}_{l,k}, s_k)$ and $\mathcal{R}_{r,k} = \mathcal{R}(\mathbf{p}_{r,k}, s_k)$, where s_k is the scale at time k. Note that the scale of the object in the image is inverse proportional with λ and therefore can be estimated by

$$s_k = \lambda_0/\lambda_k, \quad (9)$$

where λ_0 is initial target range and is found by back-projecting $\mathbf{p}_{l,0}$ and $\mathbf{p}_{r,0}$.

The appearance of a target confined to the image region $\mathcal{R}(\mathbf{p}, s)$ is described by means of a spatially-weighted color (RGB) histogram (Comaniciu et al., 2003) with B bins:

$$h_{\mathcal{R}}(u) = c\sum_{\mathbf{r}\in\mathcal{R}} \phi\left(\frac{\|\mathbf{r}-\mathbf{p}\|}{H}\right)\delta[b(\mathbf{r})-u],\ u = 1, ..., B$$

where c is a constant such that $\sum_{u=1}^{B} h(u) = 1$, $b(\mathbf{r})$ is a function mapping the color of the point $\mathbf{r}$ into a color bin, $H = s\sqrt{H_u^2 + H_v^2}$, and ϕ is the kernel function

$$\phi(r) = \begin{cases} 1 - r^2, & r < 1 \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

The target model consists of the color histograms of the elliptic regions $\mathcal{R}_{l,0}$ and $\mathcal{R}_{r,0}$. For simplicity, let these histograms be denoted by

$$q_l(u) \triangleq h_{\mathcal{R}_{l,0}}(u), \quad q_r(u) \triangleq h_{\mathcal{R}_{r,0}}(u) \quad (11)$$

For $k > 0$, the similarity between the target model q and the color model h of a target candidate (in one image) is assessed using the Bhattacharya distance, defined as

$$d[q, h] = \sqrt{1 - \rho[q, h]}, \quad (12)$$

where $\rho[q, h] = \sum_{u=1}^{B}\sqrt{q(u)h(u)}$.

The Bayesian estimation paradigm entails specifying the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$ of the state $\mathbf{x}_k$ given the measurement $\mathbf{z}_k$. Note that, in our case, the measurement consists of color stereo images, $\mathbf{z}_k = \{\mathbf{I}_{l,k}, \mathbf{I}_{r,k}\}$. Dropping the time index k for convenience, the likelihood can be expressed as

$$p(\mathbf{I}_l, \mathbf{I}_r|\mathbf{x}) = p(\mathbf{I}_l, \mathbf{I}_r|\alpha, \beta, \lambda) = p(\mathbf{I}_l|\Delta, \mathbf{I}_r)p(\mathbf{I}_r|\Delta), \quad (13)$$

where $\Delta \triangleq (\mathbf{p}_l, \mathbf{p}_r, s)$ and $(\mathbf{p}_l, \mathbf{p}_r) = \text{T2S}(\mathbf{x})$. For the partial likelihood $p(\mathbf{I}_r|\Delta)$, we use the formulation from (Perez et al., 2002)(Nummiaro et al., 2003):

$$p(\mathbf{I}_r|\Delta) = p(\mathbf{I}_r|\mathcal{R}(\mathbf{p}_r, s)) \propto \exp\left\{-\frac{d^2[q_r, h_{\mathcal{R}_r}]}{2\sigma_r^2}\right\}, \quad (14)$$

where d is given by (12) and σ_r^2 is a design parameter which plays the role of a measurement error variance.

The image correlation likelihood $p(\mathbf{I}_l|\Delta, \mathbf{I}_r)$ quantifies the matching between the Bhattacharya distance in the left and right image and is modeled here as a Gaussian function of the distance difference:

$$p(\mathbf{I}_l|\Delta, \mathbf{I}_r) = p(\mathbf{I}_l|\mathcal{R}(\mathbf{p}_l, s), \mathcal{R}(\mathbf{p}_r, s), \mathbf{I}_r) \propto \exp\left\{-\frac{(d[q_l, h_{\mathcal{R}_l}] - d[q_r, h_{\mathcal{R}_r}])^2}{2\sigma_c^2}\right\} \quad (15)$$

where σ_c is a design parameter. Plugging (14) and (15) into (13) yields

$$p(\mathbf{I}_l, \mathbf{I}_r|\mathbf{x}) \propto \exp\left\{-\frac{d^2[q_r, h_{\mathcal{R}_r}]}{2\sigma_r^2} - \frac{(d[q_l, h_{\mathcal{R}_l}] - d[q_r, h_{\mathcal{R}_r}])^2}{2\sigma_c^2}\right\} \quad (16)$$

3.4 Particle filter algorithm

For non-linear, non-Gaussian and multi-modal models, as the one described here, the particle filter (Arulampalam et al., 2002) provides a Monte Carlo solution to the recursive filtering equation $p(\mathbf{x}_k|\mathbf{z}_{1:k}) \propto p(\mathbf{z}_k|\mathbf{x}_k) \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ necessary for tracking. Starting with a weighted particle set $\{(\mathbf{x}_{k-1}^{(i)}, \tilde{w}_{k-1}^{(i)})\}_{i=1}^{N}$ approximately distributed according to $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$, the particle filter proceeds by predicting new samples from a suitably chosen proposal distribution which may depend on the old state and the current and previous measurements, i.e. $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_{1:k})$. To maintain a consistent sample, the new particle weights are set to

$$w_k^{(i)} \propto \frac{p(\mathbf{z}_k|\mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_{1:k})} \tilde{w}_{k-1}^{(i)}. \qquad (17)$$

After weight normalization, the new particle set $\{(\mathbf{x}_k^{(i)}, \tilde{w}_k^{(i)})\}_{i=1}^{N}$ is then approximately distributed according to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. The particles are resampled according to their weights to avoid degeneracy.

Particle filters suffer from the curse of dimensionality, i.e., as the dimension of the state-space increases an exponentially increasing number of particles is required to maintain the same estimation accuracy. To mitigate this phenomenon, we choose a proposal density which biases the generation of the particles towards the most-likely 3D location, while it maintains predictive particles to handle the background clutter and recover from failure or temporary occlusion. More specifically, $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{1:k})$ is a mixture between the state evolution distribution, $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, and a Gaussian distribution whose mean $(\hat{\alpha}_k^{ms}, \hat{\beta}_k^{ms}, \hat{\lambda}_k^{ms})$ is derived via stereo mean-shift tracking and back-projection:

$$q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}) = (1-\gamma) p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}) + \gamma \mathcal{N}_3(\mathbf{x}_k) \quad (18)$$

where $\gamma \in (0, 1)$ is a mixing factor. The mean-shift algorithm (Comaniciu et al., 2003) minimizes (12), thereby finding a highly possible location of the target in the image. The coefficient γ expresses our belief in the mean-shift derived 3D target hypotheses, sampled according to the density

$$\begin{aligned} \mathcal{N}_3(\mathbf{x}_k) = \mathcal{N}(\alpha_k; \hat{\alpha}_k^{ms}, \sigma^2) \mathcal{N}(\beta_k; \hat{\beta}_k^{ms}, \sigma^2) \\ \cdot \mathcal{N}(\lambda_k; \hat{\lambda}_k^{ms}, \sigma^2). \end{aligned} \qquad (19)$$

The vector $(\hat{\alpha}_k^{ms}, \hat{\beta}_k^{ms}, \hat{\lambda}_k^{ms})$ is obtained as follows: starting from the mean target positions in the left and right image at time $k-1$, $\hat{\mathbf{p}}_l(k-1)$ and $\hat{\mathbf{p}}_r(k-1)$, we find via mean-shift the positions of the target at time k, respectively $\hat{\mathbf{p}}_l^{ms}(k)$ and $\hat{\mathbf{p}}_r^{ms}(k)$, in the current stereo images $\mathbf{I}_l$ and $\mathbf{I}_r$. Further, from these two

Given the sample set $S_{k-1} = \{(\mathbf{x}_{k-1}^{(i)}, \tilde{w}_{k-1}^{(i)})\}_{i=1}^{N}$ at time $k-1$, obtain $S_k = \{(\mathbf{x}_k^{(i)}, \tilde{w}_k^{(i)})\}_{i=1}^{N}$ as follows:

1. **Importance sampling**. For $i = 1, \ldots N$, sample $\mathbf{x}_k^{(i)}$ based on $\mathbf{x}_{k-1}^{(i)}$ and $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{1:k})$:
   - mean shift tracking and back-projection:
   $$\hat{\mathbf{p}}_l^{ms} = \text{MeanShift}\,(\hat{\mathbf{p}}_l(k-1), \mathbf{I}_l)$$
   $$\hat{\mathbf{p}}_r^{ms} = \text{MeanShift}\,(\hat{\mathbf{p}}_r(k-1), \mathbf{I}_r)$$
   $$(\hat{\alpha}^{ms}, \hat{\beta}^{ms}, \hat{\lambda}^{ms}) = \text{S2T}(\hat{\mathbf{p}}_l^{ms}, \hat{\mathbf{p}}_r^{ms})$$
   - sample $u \sim \mathcal{U}(u; 0, 1)$
   - if $u > \gamma$, sample $\mathbf{x}_k^{(i)} = (\alpha^{(i)}, \beta^{(i)}, \lambda^{(i)})$ as follows:
   $$\alpha^{(i)} \sim p(\alpha_k|\alpha_{k-1}^{(i)}) \quad \text{[see eq. (1)]}$$
   $$\beta^{(i)} \sim p(\beta_k|\beta_{k-1}^{(i)}) \quad \text{[see eq. (2)]}$$
   $$\lambda^{(i)} \sim p(\lambda_k|\lambda_{k-1}^{(i)}) \quad \text{[see eq. (3)]}$$
   else, sample $\mathbf{x}_k^{(i)} = (\alpha^{(i)}, \beta^{(i)}, \lambda^{(i)})$ as follows:
   $$\alpha^{(i)} \sim \mathcal{N}(\alpha_k; \hat{\alpha}^{ms}, \sigma^2) \quad \text{[see eq. (19)]}$$
   $$\beta^{(i)} \sim \mathcal{N}(\beta_k; \hat{\beta}^{ms}, \sigma^2) \quad \text{[see eq. (19)]}$$
   $$\lambda^{(i)} \sim \mathcal{N}(\lambda_k; \hat{\lambda}^{ms}, \sigma^2) \quad \text{[see eq. (19)]}$$
   - project $\mathbf{x}_k^{(i)}$ to the image locations $\mathbf{p}_l^{(i)}, \mathbf{p}_r^{(i)}$:
   $$(\mathbf{p}_l^{(i)}, \mathbf{p}_r^{(i)}) = \text{T2S}(\alpha^{(i)}, \beta^{(i)}, \lambda^{(i)}), \quad s^{(i)} = \lambda_0/\lambda^{(i)}$$
   - compute $w_k^{(i)}$, the unnormalized weight of $\mathbf{x}_k^{(i)}$, according to (17,4,16,18); the likelihood $p(\mathbf{I}_l, \mathbf{I}_r|\mathbf{x}_k^{(i)}) = p(\mathbf{I}_l, \mathbf{I}_r|\mathbf{p}_r^{(i)}, \mathbf{p}_l^{(i)}, s^{(i)})$ (16) is computed based on the histograms
   $$h_r^{(i)} \triangleq h_{\mathcal{R}_r}(u), \text{ where } \mathcal{R}_r = \mathcal{R}(\mathbf{p}_r^{(i)}, s^{(i)}) \subset \mathbf{I}_r$$
   $$h_l^{(i)} \triangleq h_{\mathcal{R}_l}(u), \text{ where } \mathcal{R}_l = \mathcal{R}(\mathbf{p}_l^{(i)}, s^{(i)}) \subset \mathbf{I}_l$$
2. **Weight normalization**: $\tilde{w}_k^{(i)} = w_k^{(i)} / \sum_{i=1}^{N} w_k^{(i)}$
3. **Estimation**. Compute the mean state of the set S_k, the scale estimate, and the mean target positions in the left and right image:
   $$\hat{\mathbf{x}}_k = (\hat{\alpha}_k, \hat{\beta}_k, \hat{\lambda}_k) = \textstyle\sum_{i=1}^{N} \tilde{w}_k^{(i)} \mathbf{x}_k^{(i)}, \;\; \hat{s}_k = \lambda_0/\hat{\lambda}_k,$$
   $$\hat{\mathbf{p}}_l(k) = \textstyle\sum_{i=1}^{N} \tilde{w}^{(i)} \mathbf{p}_l^{(i)}, \quad \hat{\mathbf{p}}_r(k) = \textstyle\sum_{i=1}^{N} \tilde{w}^{(i)} \mathbf{p}_r^{(i)}$$
4. **Target model update**. Compute the occlusion/outlier indicator $o = p(\mathbf{I}_l|\hat{\mathbf{p}}_l, \hat{s}) + p(\mathbf{I}_r|\hat{\mathbf{p}}_r, \hat{s})$ as the sum of likelihoods (defined by(14)) of the elliptical regions of scale $\hat{s}$, centered at $\hat{\mathbf{p}}_l$ and $\hat{\mathbf{p}}_r$ respectively; if o exceeds a threshold th_1, we proceed to the target model update (see Section 3.5).
5. **Selective resampling**: if the effective sample size
   $$N_{eff} = \left[\textstyle\sum_{i=1}^{N} (\tilde{w}_k^{(i)})^2\right]^{-1}$$
   is below a threshold th_2, apply a systematic resampling step - see (Arulampalam et al., 2002).

Figure 3: Particle filter algorithm

image locations, we can determine the 3D location $(\hat{\alpha}_k^{ms}, \hat{\beta}_k^{ms}, \hat{\lambda}_k^{ms})$ by back-projection (8).

The outline of the particle filter algorithm for color-based stereo target tracking is presented in Figure 3.

3.5 Target model update

Let $\hat{\mathbf{p}}_l$ and $\hat{\mathbf{p}}_r$ denote the estimates of the centers of the target regions in the left and right image, respectively. If the sum of likelihoods (14) of $\hat{\mathbf{p}}_l$ and $\hat{\mathbf{p}}_r$ are higher than a threshold, it means that there is no outlier or occlusion at the estimated target position in the image. Therefore, we can update the target model to cope with illumination variations resulting in appearance changes. The target models, $q_l(u)$ and $q_r(u)$, are updated as in (Nummiaro et al., 2003):

$$q_l(u) = (1-a)q_l(u) + ah_{\hat{\mathcal{R}}_l}(u), \qquad (20)$$

$$q_r(u) = (1-a)q_r(u) + ah_{\hat{\mathcal{R}}_r}(u), \qquad (21)$$

where $u = 1, ..., B$, $\hat{\mathcal{R}}_l = \mathcal{R}(\hat{\mathbf{p}}_l, \hat{s})$, $\hat{\mathcal{R}}_r = \mathcal{R}(\hat{\mathbf{p}}_r, \hat{s})$, $\hat{s}$ is the scale estimate, and $a \in (0,1)$ is a factor weighting the color model of the target at the estimated positions $\hat{\mathbf{p}}_r$ and $\hat{\mathbf{p}}_l$. This evokes a forgetting process whereby the contribution of a specific frame decreases exponentially in time.

3.6 Camera control

The control scheme refers here only to the pan (azimuth) angle α. The control of the tilt (elevation) angle β can be done in the same manner. We use a discrete Proportional-Integral-Derivative (PID) controller given by

$$u_k^p = K_p e_k + K_i \frac{T_s}{T_i} \sum_{i=0}^{k} e_i + K_d \frac{T_d}{T_s}(e_k - e_{k-1}) + u_0^p$$

where e_k is the estimate of the azimuth angle at time k as delivered by the particle filter, $e_k = \hat{\alpha}_k$. The parameters K_p, K_i, K_d, T_i, T_d are design constants and T_s is the sampling period.

4 ROBOT NAVIGATION

Two behavior-based architectures for robot navigation are presented here. The general idea behind behavior-based approaches is to decompose a task in simpler tasks that are easier to implement and test. The challenge of this approach remains in how to combine these different subtasks such that the global task is executed in a robust manner. The robot navigation problem can be subdivided into two main parts:

- How to reach the goal location?
- How to avoid obstacles?

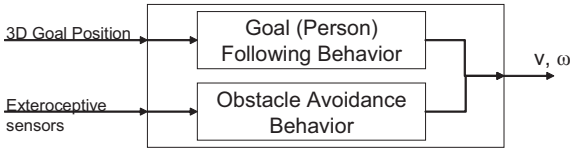

Figure 4: Fuzzy-logic behavior based navigation controller

The presented solutions preserve this ambivalent structure by providing a behavior-based navigation strategy where two main behaviors process each one of the questions raised above.

The robot navigation module produces as output a heading direction for the robot and a speed setpoint, usable on any mobile robotic platform. The speed setpoint depends directly on the distance to the target person: if the robot is far away, it needs to accelerate in order not to loose the person; when it approaches the target, it must move with more caution to not hurt the human. When the robot comes within one meter of the target person, it will stop automatically, for security reasons.

4.1 Fuzzy-logic behavior based navigation controller

In this setup, depicted in Figure 4, each of the behaviors consists of a fuzzy logic controller relating the input commands, i.e. the sensory data, to output commands for the robot actuators. For the person-following or goal-seeking behavior, the input comes from the active stereo vision system, delivering the 3D position of the target person, whereas the obstacle avoidance behavior uses exteroceptive sensor data to find a path without colliding with obstacles.

4.2 Hybrid behavior based navigation controller

We also exploit a hybrid architecture used for moving in human-centered environments (Nuttin et al., 2003). This architecture consists of a deliberative and a reactive part. In this way, the advantages of both approaches are combined. The robot is able to reason about how to reach a certain goal position, taking a priori knowledge about the environment into account if this is available. At the same time, it is able to react very quickly to unmodeled obstacles in the environment, by adopting a more direct coupling between sensors and actuators. A multi-agent framework in which behaviors can be specified conveniently was developed for this goal, as in (Waarsing et al., 2003).

Figure 5 depicts the proposed architecture. The navigation module as a whole calculates the linear

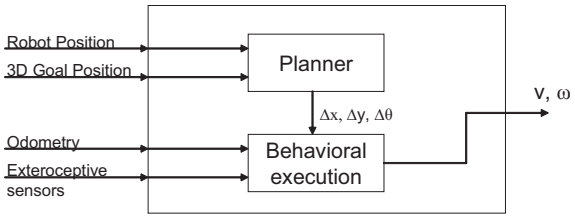

Figure 5: Hybrid behavior based navigation controller

Figure 6: The Nomad200 and wheelchair with the active stereo vision system

Figure 7: Target tracking results: the white ellipse indicates the goal which is tracked, the small circles represent the different particles of the particle filter. The columns show (1,2) stereo head tracking, (3) robot advancing to the target.

velocity and heading direction v and ω of the robot, given the current robot location (x, y, θ) and its uncertainty, the robot's global goal given by the active vision module, the measured ranges from the exteroceptive sensors, and the odometry values. During navigation, a global planner and a behavioral execution unit co-operate.

5 RESULTS AND DISCUSSION

The overall control strategy was tested on a Nomad200 robot, which is a pure laboratorial robot used for testing purposes. As such, the experimental results which are shown in this section, are obtained with this robot. For more real-world applications, we make use of a mobile wheelchair platform. Both mobile robot platforms are shown in Fig. 6. For this research project, we developed a totally independent stereo vision platform consisting of a PC, with a small LCD screen. On top of the platform, a Biclops stereo head is installed, carrying two high-resolution Pulnix color cameras. The whole system is totally self sustainable as it runs on its own power resources (six 10Ah batteries). The stereo vision platform can be seen on top of the Nomad robot in Figure 6, while a model of the stereo vision subsystem is shown in Figure 1.

The results of the person following application are illustrated in Figure 7. With a number of $N = 70$ particles and $B = 8 \times 8 \times 8$, the system is able to run in real-time. As can be noticed, the tracker succeeds to aim the stereo head towards the target person, withstanding illumination changes and even though the movement of this person was not easily predictable. The robot navigates towards this person while avoiding the obstacles on its way to come to a stop 2 meters in front of the person.

To assess the ASVM's performance as to the target range estimation, we conducted an experiment where a colored object is rotated with constant angular speed in a plan parallel with the ground and at a height corresponding to that of the stereo head. In this case, the range varies in a sinusoidal manner (see Figure 8). At the beginning, there is a short stationary period necessary for the ASVM to center the target in its field of view. Note that the target range is tracked quite accurately, with a small lag.

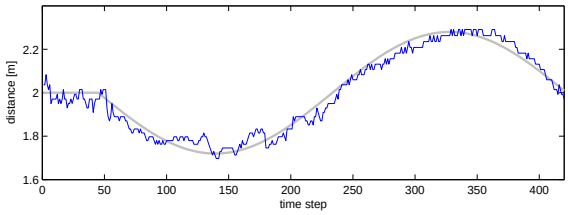

Figure 8: Estimated target range (blue line) vs. ground truth (gray line).

6 CONCLUSION

We have presented the active stereo vision (ASVM) and navigation (NM) modules of a mobile robot system designed for person following. The ASVM controls a stereo head for tracking a target by means of a color-based particle filter, robust to illumination variations, erratic target motions, and short occlusions. To enforce the stereo constraint (the target regions in the stereo images are correlated through the stereo head-target 3D geometry), the measurement process is formulated in the image plane, whereas the system dynamics is based on the 3D position of the target. Keeping the target in the ASVM's field of view is achieved by adjusting the pose of the stereo head via a PID pan/tilt controller. Further, the estimate of the 3D target position is fed to the NM, which consists of a behavior-based navigation controller. Two different navigation controllers were presented. Finally, the concept was demonstrated by implementing it on both a Nomad200 and a wheelchair platform.

ACKNOWLEDGMENT

This research has been conducted within the framework of the Inter-Universitary Attraction-Poles program number IAP 5/06 Advanced Mechatronic Systems, funded by the Belgian Federal Office for Scientific, Technical and Cultural Affairs.

REFERENCES

Arsenio, A. M. and Banks, J. L. (1999). People detection and tracking by a humanoid robot. Technical report, MIT.

Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions of Signal Processing*, 50(2):174–188.

Beardsley, P., Reid, I., Zisserman, A., and Murray, D. (1995). Active visual navigation using non-metric structure. In *5th International Conference on Computer Vision*, pages 58–64, Cambridge, MA, USA.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:564–577.

Davison, A. and Murray, D. (1998). Mobile robot localization using active vision. In *5th European Conference on Computer Vision*, volume 2, pages 809–825, Freiburg, Germany.

Franz, M.-O. and H.-A., M. (2000). Biomimetic robot navigation. *Robotics and Autonomous Systems*, 30:133–153.

Ghita, O. and P.-F., W. (2003). Real time 3d estimation using depth from defocus. *Vision, MVA (SME)*, 16(3):1–6.

Koren, Y. and Borenstein, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation. In *IEEE Conference on Robotics and Automation*, pages 1398–1404, Sacramento, California.

Kuniyoshi, Y. and Rougeaux, S. (1999). A humanoid vision system for interactive robots. In *1st Asian Symposium on Industrial Automation and Robotics*, pages 13–21.

Nummiaro, K., Koller-Meier, E., and Gool, L. V. (2003). An adaptive color-based particle filter. *Image and Vision Computing*, 21:99–110.

Nuttin, M., Vanhooydonck, D., Demeester, E., Brussel, H. V., Buijsse, K., Desimpelaere, L., Ramon, P., and Verschelden, T. (2003). A robotic assistant for ambient intelligent meeting rooms. In *1st European Symposium on Ambient Intelligence (EUSAI)*, pages 304–317, Veldhoven, The Netherlands. http://www.mech.kuleuven.be/pma/research/mlr.

Perez, P., Hue, C., Vermaak, J., and Gangnet, M. (2002). Color-based probabilistic tracking. In *European Conf. Computer Vision (ECCV)*, volume 1, pages 661–675.

Pérez, P., Vermaak, J., and Blake, A. (2004). Data fusion for visual tracking with particles. *Proc. IEEE*, 92:495–513.

Ping, H., Sahli, H., Colon, E., and Baudoin, Y. (2001). Visual servoing for robot navigation. In *3rd International Conference on Climbing and Walking Robots: Clawar 2001*, pages 255–264, Karlsruhe, Germany.

Schlegel, C., Illmann, J., Jaberg, H., Schuster, M., and Worz, R. (2000). Integrating vision based behaviors with an autonomous robot. *Videre: Journal of Computer Vision Research*, 1(4):32–60.

Strens, M.-J.-A. and Gregory, I.-N. (2003). Tracking in cluttered images. *Image Vision and Computing*, 21(10):891–911.

Vieville, T. (1997). *A few steps towards 3D active vision*. Springer, Berlin.

Waarsing, B., Nuttin, M., and Brussel, H. V. (2003). A software framework for control of multi-sensor, multi-actuator systems. In *International Conference on Advanced Robotics (ICAR)*, Coimbra, Portugal.

Wilhelm, T., Bohme, H.-J., and Gross, H.-M. (2004). A multi-modal system for tracking and analyzing faces on a mobile robot. *Robotics and Autonomous Systems*, 48:31–40.