in: Proc. 55th Int. Scientific Colloquium, Ilmenau, Germany, 2010, pp. 606-611, Verlag ISLE 2010

APPEARANCE-BASED PERSON TRACKING AND 3D POSE ESTIMATION OF UPPER-BODY AND HEAD

Christoph Weinrich, Steffen Müller, Horst-Michael Gross

Ilmenau University of Technology, Neuroinformatics and Cognitive Robotics Lab
christoph.weinrich@tu-ilmenau.de

ABSTRACT

In the field of human-robot interaction (HRI), recognition of humans in a robot's surroundings is a crucial task. Besides the localization, the estimation of a person's 3D pose based on monocular camera images is a challenging problem on a mobile platform. For this purpose, an appearance-based approach, using a 3D model of the human upper body, has been developed end experimentally investigated. For a real time tracking, the state of the person is estimated by a particle filter tracker, which uses different observation models for evaluating pose hypotheses. The 6D body pose is modeled by 4 parameters for the torso position and orientation as well as 2 for the head pan and tilt. In order to achieve real time operation, a smooth fit value function simplifies the particle filter's convergence. Futhermore, a sparse feature based model eliminates the need for computationally expensive geometric transformations of the image, as required by conventional Active Appearance Models (AAM). The initialization problem of the pose tracker is overcome by integrating a Histograms of Oriented Gradients (HOG) detector.

Index Terms— Appearance model, visual person detection, body pose tracking

1. INTRODUCTION

The visual detection and tracking of human pose is a long-standing task with great importance to the human-robot interaction. For realizing socially acceptable navigation behaviors of a robot this pose information is essential. Since 2005 the Histograms of Oriented Gradients (HOG)[1] have been successfully used for visual 2D full length body detection at lower scale. Robustness to texture, color and illumination as well as invariance to the body pose characterize this method. Therefore, the orientation of a person is not perceptable by HOG detectors. Additionally, the HOG detector is very time consuming and not suitable real time tracking. The robust detections of HOG however seem to be useful for initialization and backup of other real time tracking methods.

For appearance based tracking Cootes [2] originally introduced Active Aprearance Models (AAMs) in 1998. These parametric models are originally used for face tracking and analysis, but the idea of analysis by synthesis might be useful also for upper body detection.

The contribution of this paper is the combination of HOG person detection with a 3D appearance model of the human upper-body to recognize the exact body pose and head orientation in real-time. In contrast to other body models, which rely on texture models of the target person, the presented appearance model initially relies on edge information only and learns the color model of the target person on the fly. By means of the combination of HOG and the appearance model, we are able to overcome the initialization problem of other 3D model tracking approaches as well as to distinguish people by different appearance once a model has been specialized.

The remainder of the paper first gives a brief survey on people detection and tracking approaches, before our approach is presented in detail. Some experimental results show the achived quality and open issues of our approach.

2. RELATED WORK

Visual person detection approaches can be categorized into *implicit* and *explicit* methods. On the one hand, implicit methods learn a background model and detect foreground objects like people or moving objects as a deviation from this model [3]. They require static cameras, which prevents application on a mobile robot. On the other hand, explicit methods detect people with some kind of person model. This model can consist of a representation of the complete body or only parts thereof – the best example being the face. The most prominent up-to-date face detection method is AdaBoost, which learns and applies a cascade of simple but very efficient image region classifiers [4]. Drawbacks of face detection become relevant, whenever the person does not face towards the robot or the face appears too small due to a large distance.

This work has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 216487 (CompanionAble Project) and from the Ph.D. Graduate School on Image Interpretation at Ilmenau University of Technology.

Full body models can be distinguished into *monolithic* and *part-based* models. Monolithic body representations (mostly discriminative models) use templates, active contours or other features. The former group uses huge numbers of static templates to capture high variances in people's poses [5]. Active contours try to overcome the rigidness of templates by fitting flexible structures to the human body [6]. They usually are computationally expensive and may be distracted by background structures. In the field of feature-based monolithic methods, Histograms of Oriented Gradients (HOG) and their derivations have established as state-of-the-art person detectors [1].

Existing part-based body models are mostly generative and feature-based. These models consist of different body parts (head, torso, limbs) and a representation of their spatial relationship making them adequate to capture different body poses. The body parts and relationships can either be modeled *explicitly* [7] or *implicitly* – including the well-known Implicit Shape Model [8]. The main disadvantage of these approaches is the high computational cost for detecting body parts. Therefore, methods with explicit modeling often are initialized manually or by predefined poses, assume little self-occlusion and favorable illumination conditions.

Among these presented methods several combinations exist. In [9] for example a monolithic HOG detector is combined with deformable sub parts, which increases detection performance given partial occlusion.

Once detection on individual images is possible, tracking of persons over time usually improves accuracy of these methods. Here, mostly probabilistic state estimation and tracking approaches are applied, often based on the Bayesian Filter. Unimodal state hypotheses thus are tracked with Kalman filters and its derivates, whereas for multimodal tracking typically particle filters are employed [10].

3. 3D UPPER BODY APPEARANCE MODEL

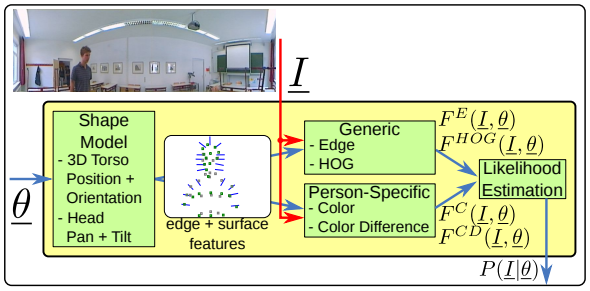


Fig. 1. *Structure of our 3D Appearance Model for evaluation of an image's likelihood given a hypothesis: a shape model defines feature positions, observation models compare image to expected appearance, yielding a likelihood*

The core of our approach is a 3D appearance model (Figure 1). It estimates the likelihood $P(\underline{\boldsymbol{I}}|\underline{\boldsymbol{\theta}})$, that the current image $\underline{\boldsymbol{I}}$ might be observed given the person-pose-hypothesis parameter vector $\underline{\boldsymbol{\theta}}$. To estimate $P(\underline{\boldsymbol{I}}|\underline{\boldsymbol{\theta}})$, the image $\underline{\boldsymbol{I}}$ is evaluated at several discrete feature positions: The shape model determines the positions of m=19 edge features (along the body's silhouette) and n=45 surface features, according to the pose parameters $\underline{\boldsymbol{\theta}}$. Using these feature positions, two generic and two person-specific observation models (see sections 3.2 and 3.3) evaluate the likelihood of the current image $\underline{\boldsymbol{I}}$. Finally the likelihood estimations of the parts are combined to get a global fit value approximation $P(\underline{\boldsymbol{I}}|\underline{\boldsymbol{\theta}})$.

In order to find the best pose parameters $\underline{\boldsymbol{\theta}}$, in the Active Appearance Model (AAM) [2] approach a local gradient descent on an image difference measure is applied, which relies on an adequate initialization. Contrary, in our approach the global parameter space is searched by a particle filter, which samples for initialization positions. During an inner loop a gradient ascent is applied to adapt the hypotheses, aiming to fit the image best and thus reduce the number of particles needed. Details on the adaptation process are given in section 4.

Furthermore, a fundamental difference to the AAM's texture model is that the image is only evaluated at discrete positions, by several difference measures. Due to efficiency reasons, we use no parametric model of the color and edge appearance, instead a likelihood estimation is required substituting the direct comparison in image space. This likelihood function is a weighted mixture of different aspects. On the one hand there are generic components to detect people in general, even if the model could not adapt to an individual yet. The generic models are a contour model of the upper body (edge model) as well as a HOG detector. Details on these are described in 3.2

On the other hand, there are person-specific components primarily analyzing the color at the 3D feature points projected by the shape model. In particular the direct color and color differences of feature point pairs are modeled and adapted to a detection online in order to specialize the model. Later on, this helps to reidentify persons to be tracked.

3.1. Shape Model

The Shape Model is used to model the person's geometric shape in 3D space. It is parameterized by the direction and distance to the camera (polar coordinates in top view), the height of the model (body size), the orientation angle towards the camera (heading direction), and the pan-tilt parameters of the head.

The model has been learned from a sequence of people wearing a flexible marker suit, standing upright in front of a camera. A non linear bundle adjustment method has been used for optimizing the feature's po-

sitions. Originally, also a principal component analysis has been applied to the 3D shapes to allow representation of individual body properties (breast, stomach, and shoulder measures). For reasons of efficiency, the current implementation neglects these parameters.

According to the model parameters $\underline{\boldsymbol{\theta}}$, knowing the camera geometry, the expected positions of particular edge and surface-features are projected into the camera's image plane. For the edge-features in addition to the positions $\underline{\boldsymbol{k}}_1, ..., \underline{\boldsymbol{k}}_m$, the gradient orientations $\gamma_1, ..., \gamma_m$ are modeled.

Due to the 3D nature of the body, self occlusion is of significant relevance. Thus for each surface-feature position $\underline{\boldsymbol{p}}_i$ a visibility value v_i is calculated, in order to handle self occlusion of the upper body. Due to the convex shape of a body, visibility is defined by the angle of the feature to the shape model center and the shape model center to the camera. Thus, no normal vector (as usually used for backface culling) has to be modeled and transformed each time an evaluation is necessary.

During the tracking process, for each particle the shape model defines the feature positions in the current image, at which evaluation by the observation models takes place. In the following, these likelihood estimators are described in detail.

3.2. Generic observation model components

Edge Model

Due to the great variance of texture and color of peoples clothes, the only invariant information can be found in the image gradients. The success of robust detection approaches, like HOG, prove the relevance of these features. The edge model compares expected edge orientation of the silhouette to the gradient orientations $\underline{\boldsymbol{I}}^P$ in the image, whereat the respective magnitudes $\underline{\boldsymbol{I}}^M$ of the image gradients are used as weights.

Since image gradients are placed very locally (see Fig. 2a), a slightly different position parameter $\underline{\boldsymbol{\theta}}$ will produce great changes in the response of the edge model likelihood. To get a smooth likelihood function, which will speed up convergence of the particles, some preprocessing is applied to the edge image, which is computed as described in the following.

In order to extract the horizontal and vertical edge images $\underline{\boldsymbol{I}^X}$ and $\underline{\boldsymbol{I}^Y}$, simple Roberts' Cross kernels are used:

$$\underline{\boldsymbol{I}^X} = \underline{\boldsymbol{I}} * \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{1}$$

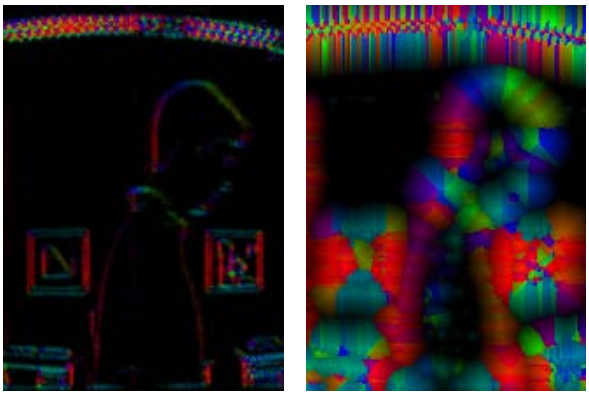
$$\underline{\boldsymbol{I}^Y} = \underline{\boldsymbol{I}} * \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \tag{2}$$

The resulting images can be used to calculate the edge gradients orientation (phase) $\underline{\boldsymbol{I}}^p$ and magnitudes $\underline{\boldsymbol{I}}^m$.

$$\underline{\boldsymbol{I}}^p = \tan^{-1}\left(\frac{\underline{\boldsymbol{I}}^Y}{\underline{\boldsymbol{I}}^X}\right) \tag{3}$$

$$\underline{\boldsymbol{I}}^m = \sqrt{\underline{\boldsymbol{I}}^{X^2} + \underline{\boldsymbol{I}}^{Y^2}} \tag{4}$$

To reduce noise and emphasize the relevant edges, a nonlinear filter is applied to the magnitude image $\underline{\boldsymbol{I}}^m$, suppressing low values and emphasizing the higher ones. Additionally, in order to get a smoothed gradient image, the gradient magnitudes are spatially spread out similar to the Chamfer algorithm [11], while edge orientation is taken from the highest magnitude in the surrounding pixels. This algorithm allows to propagate the edge information to arbitrary distance at constant time. Figure 2b shows the resulting gradient image, used for sampling the difference between edge orientation and the respective expectation.



(a) $\underline{\boldsymbol{I}}^{mm}, \underline{\boldsymbol{I}}^{mp}$ (b) $\underline{\boldsymbol{I}}^M, \underline{\boldsymbol{I}}^P$

Fig. 2. *Left: raw gradient image, Right: propagated edges; In both images orientation is coded by hue and the magnitude by intensity.*

For evaluating a particle's edge likelihood, for each edge feature point of the shape model, a Gaussian on the difference between expected measured orientation is applied and weighted by the magnitude at the feature's position. High likelihoods are resulting from correctly oriented image gradients directly situated at the expected shape model feature positions. The edge models likelihood $F^E\left(\underline{\boldsymbol{I}}, \underline{\boldsymbol{\theta}}\right)$ is calculated as the arithmetic mean of all m edge fit values. One might expect a product here, assuming independent observations at the feature positions. However, experiments showed that this assumption might not be correct and an average is more robust.

HOG Model

As already noticed, the HOG detector [1] is a very reliable source of information, which we do not want to miss. Caused by the expensive computational effort, we unfortunately only can run the HOG detector in a parallel process every second. We trained an HOG detector on upper body data, containing different body orientations. Due to Non-Maximum-Suppression, this should result in one bounding box for each detected person. To get a real-valued likelihood $F^{HOG}\left(\underline{\boldsymbol{I}}, \underline{\boldsymbol{\theta}}\right)$ for weighting the particles of the particle filter based tracker, the matching of the expected bounding box,

resulting from the shape model features (see above), to the closest HOG detection is used. Due to a fixed ratio of HOG detection bounding boxes, the area of bounding box overlap can not be applied directly. Instead we only use the vertical overlap and the horizontal distance (see Fig. 3). Additionally, it would be possible to include a real valued likelihood of the HOG detections as introduced by Felzenszwalb [9].

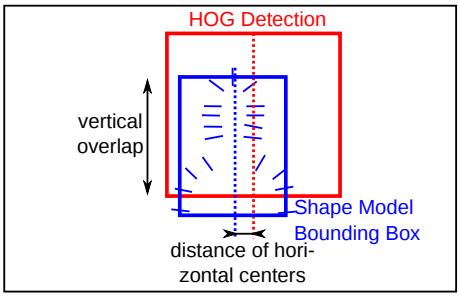


Fig. 3. *For calculation of HOG likelihood for a given hypothesis $\underline{\boldsymbol{\theta}}$, only the vertical overlap and the horizontal distance, between the bounding box detected by HOG and the expected bounding box of the Shape Model, are used*

3.3. Specific observation model components

While the HOG models are useful for detecting people initially, only the Edge Model is applicable at an adequate frame rate. For robust tracking, which is not possible purely based on edge orientation, and for identification of different people, more person dependent information is needed. The color of clothes, hair and skin are suitable. Once an HOG and edge based model could be fitted, the color of feature points can be adapted for the present person, and the information of color and color difference model now allows a tracking in real time, based on edges and color features. In particular, each time an unspecific color model trained on multiple people yields better likelihood than a person specific model, a new instance for the person specific trackers is created for that person. Models not matching for a longer time are deleted.

Color Model

The Color Model operates in the HSI color space. For each shape model feature point, a Gaussian distribution on the HSI color is specified. Its parameters are learned, using the images when the model has been fitted using the generic models. Therefore, a recursive Maximum Likelihood estimation is used. While different color spaces have been tested, we found that robustness on illumination changes and shadow casting are represented best in a decorrelated space like HSI. Reflecting that issue, the estimated variance regarding the intensity dimension usually is higher than for hue and saturation.

To get the color likelihood $F^C\left(\underline{\boldsymbol{I}}, \underline{\boldsymbol{\theta}}\right)$ for a given parameter vector $\underline{\boldsymbol{\theta}}$, again all feature positions i are evaluated and combined taking into account their current visibility v_i.

First, the Gaussians are evaluated with the current pixel color at feature position i, yielding $w_{H,i}$, $w_{S,i}$ and $w_{I,i}$ for the three color space channels.

One disadvantage of HSI space is the high noise of hue for high and low intensities. Therefore, a second step mixes w in a continuous manner. λ is the mixing function depending on the image intensity at the considered pixel. It is high for weak intensities, falling to zero for medial intensities and rising again for high intensities.

$$w_i = \left[\lambda(\underline{\boldsymbol{I}}_{I,i}) \cdot w_{I,i} + (1 - \lambda(\underline{\boldsymbol{I}}_{I,i})) \cdot \frac{w_{H,i} + w_{S,i} + w_{I,i}}{3}\right] \tag{5}$$

Once the color model responses w_i for all the feature positions are evaluated, they are combined, weighting by the visibility v_i.

$$F^C\left(\underline{\boldsymbol{I}}, \underline{\boldsymbol{\theta}}\right) = \exp\left(\frac{\sum_{i=1}^{m} v_i \cdot \ln(w_i)}{\sum_{j=1}^{m} v_j}\right) \tag{6}$$

Color Difference Model

As described above, the color model is adapted using only a few images. Typically, the person does not turn around, to show all the feature points for training the color. Induced by an unsymmetrically trained color model, the tracker will prefer orientations used while training the color model, since the according color features mostly match better to the observations, than unspecified color features. In order to allow a prediction of backside colors of a person, the general color difference model knows relationships of colors of different shape model feature points and helps adapting the unobserved features. Unfortunately, this intelligent update of color models was not yet implemented for the experimental tests described below, but it is expected that it improves the orientation estimation drastically.

3.4. Likelihood Estimation

Once the partial likelihoods have been determined, they have to be combined into a particle weight. Again the independence of the modalities (color, edge and HOG) can not be assumed, thus a pure product is not sufficient. In our implementation, a Gamma-operation, known from Fuzzy logic, is used. It is a compromise between product and arithmetic mean.

$$P\left(\underline{\boldsymbol{I}}|\underline{\boldsymbol{\theta}}\right) \approx \gamma \left(\sqrt[|M|]{\prod_{M\in\{E,HOG,C,CD\}} \alpha^M F^M\left(\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{I}}\right)}\right) + (1-\gamma)\left(\frac{1}{|M|}\sum_{M\in\{E,HOG,C,CD\}} \alpha^M F^M\left(\underline{\boldsymbol{\theta}}, \underline{\boldsymbol{I}}\right)\right) \tag{7}$$

It is aspired to learn the γ and α^M parameters of that combination function on training data.

Algorithm 1 Tracking

1: **for all** particle hypotheses $\underline{\boldsymbol{\theta}}_i$ **do**
2: Prediction using stochastic motion model
3: **repeat** // local optimisation
4: Importance Sampling using appearance likelihood $P\left(\underline{\boldsymbol{I}}|\underline{\boldsymbol{\theta}}\right)$ from observation model
5: gradient ascent
6: **until** convergence
7: **end for**
8: Resampling

4. TRACKING

The tracking is based on the condensation algorithm [12], a particle filter for state estimation. In our implementation, this state is the pose of persons, characterized by the parameter vector $\underline{\boldsymbol{\theta}}$. If there are multiple persons present, an additional dimension of the state space is the identitiy, that is the person specific model to be used.

Initially the particles are equally distributed within the pose space.

For each new image the particle positions $\underline{\boldsymbol{\theta}}$ are predicted using a stochastic human motion model (see alg. 1). This model considers restrictions of joints and limitations in space.

A conventional particle filter at this step would apply the importance sampling in order to weight the particles. Because of the high dimensional pose space and the limited number of particles, the likelihood of finding the optimal position in the state space is rather low. To prevent suboptimal particle distributions, a local optimization step has been introduced, which by means of a gradient ascent improves particle positions, before the final importance sampling is done. One may notice, that such a local optimization changes the belief distribution, but we are interested in the maximum at all. Too strong convergence can be compensated by a rather wide motion model. Two possible methods for the local optimization have been evaluated, a Particle Swarm Optimization (PSO) and deterministic gradient ascent, where each dimension is optimized iteratively. The PSO could not outperform the greedy gradient ascent and therefore, the more efficient gradient method has been chosen.

After that a common resampling is perfomed. Unfortunately, condensation algorithm converges all particles close to the maximum. In order to be able to detect new people, a defined rate of particles is randomly interspersed.

5. REAL TIME TRACKING AS SHARED TRACKING

A fundamental issue of the proposed tracker is the computational effort. As noticed above, the HOG model needs several seconds for one image. Because this would delay the complete tracker, we decided to run two instances in parallel (see Fig. 4). The first instance is running in real time, while only edge and person specific color models are used. Naturally, no robust tracking is possible initially.

On the second instance, which is running with a very slow frame rate and the delay of the HOG detector, the detections and local optimization yields good results. However, it is not really applicable for tracking. Once this slow instance has found a detection, the user specific models, which are shared with the other instance, are updated. From that point, the real time tracker can find people robustly.

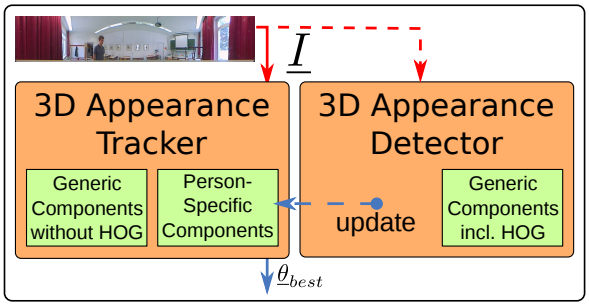


Fig. 4. *Structure of the shared overall tracker: a real time instance (left part) uses edges and color models, a more specific but significantly slower instance is utilizing the edge and HOG model, while updating the color models of the fast instance*

6. EXPERIMENTS

The validation data was captured, while a person moved within a $2m \times 4m$ rectangular area in front of the robot (see Fig. 6).

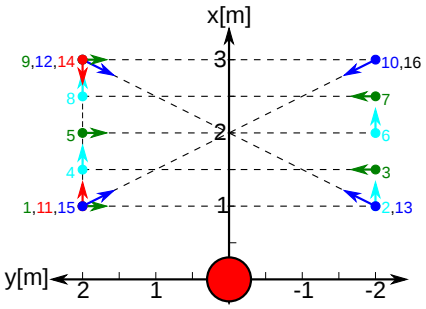


Fig. 6. *Bird's eye view of a test person's trace. Numbers give the sequential order*

The robot captured images $(1396px \times 260px)$ of its surrounding using an omnidirectional camera (SONY RPU C2512) at 6.5 frames per second. Furthermore, a laser scanner (Hokuyo URG-04LX) was applied to gain ground truth data of the person's position and orientation. The person's orientation follows from the positions, because of the predefined trace the person had to pass.

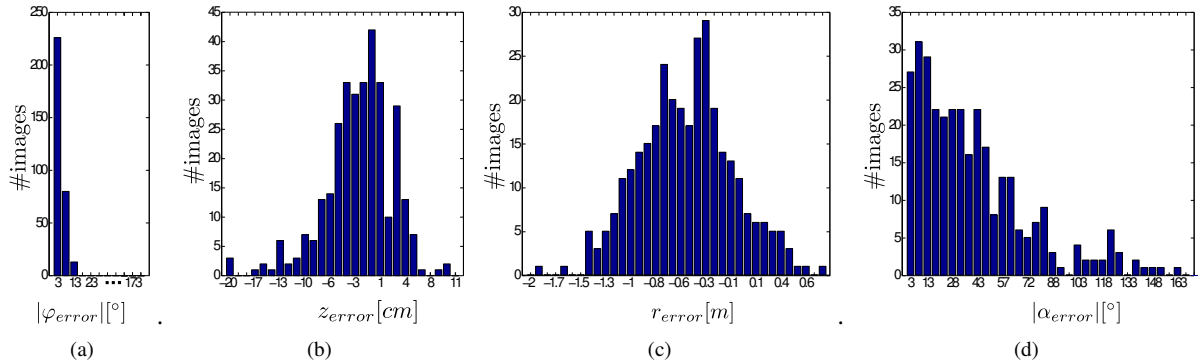


Fig. 5. Histograms on the errors in the cylindrical position dimensions (a-c) and the upper body orientation (d).

In order to evaluate the real time tracker, in a first pass, the HOG detector has been applied on the sequence and adapted the color model to its detections as described above. The image sequence consists of 592 images, but when the person turned around at the corner points no exact orientation data exists in the ground truth data. Thus these images are neglected for the evaluation, while 319 images remaine.

We could operate the system on a 2 GHz CPU in real time at 6.5 frames per second, which was the limitation due to the camera used.

The validation of the person's pose estimation is performed in cylindrical coordinates as also used for the pose parameters of the shape model. Figure 6 shows a few histograms on the errors in each of the position dimensions we were interested in.

The most informative value for validation is the estimation of the direction angle φ. This reflects the horizontal position in the onmidirectional image. The histogramm of the absolute direction error shows a clear tendency that people's position is correctly tracked. Higher errors occured when the person was very close to the robot and thus covered a wider area, which is not matchable to the laser scan accurately.

The estimation of the height z in the image has an impressive low variance of $\sigma_z = 4.76cm$. The errors here are caused by head-contour mismatches. Sometimes the model fitted to the shoulders instead of the head.

The evaluation of the distance estimation r is visualized in Figure 5(c). Its variance is $\sigma_r = 0.43m$, which is not as good as for the horizontal and vertical positions. The reason for that is the variance of the body shape. Our model assumes a rigid body structure, causing greater variance in depth by means of the inverse perspective.

The body orientation is not estimated robustly. As mentioned above, due to the not yet implemented color difference model, the initial angle can be detected rather good, but when the person turns and the other side is becoming visible, the prior in the color model causes the wrong adaptation. We are optimistic to overcome this drawback by the color difference model.

Conclusions

The proposed approach extends the generic person detection, via Histograms of Oriented Gradients (HOG), by parallel real-time 3D pose tracking and person identification based on specific color appearance models. We could overcome the limitation due to computational effort of the HOG by splitting the real time tracker from a slow instance operating the HOG.

7. REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, vol. 1, pp. 886–893.

[2] T.F.Cootes, G.J. Edwards, and C.J.Taylor, "Active appearance models," in *Proc. ECCV*, 1998, vol. 2, pp. 484–498.

[3] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. CVPR*, 1999, vol. 2, pp. 246–252.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. CVPR*, 2001, vol. 1, pp. 511–518.

[5] Dariu Gavrila and Vasanth Philomin, "Real-time object detection for "smart" vehicles," in *Proc. ICCV*, 1999, pp. 87–93.

[6] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, "Snakes: Active contour models," *IJCV*, vol. 1, no. 4, pp. 321–331, 1988.

[7] D. Ramanan, D.A. Forsyth, and A. Zisserman, "Strike a pose: tracking people by finding stylized poses," in *Proc. CVPR*, 2005, vol. 1, pp. 271–278.

[8] Bastian Leibe, Aleš Leonardis, and Bernt Schiele, "Robust object detection with interleaved categorization and segmentation," *IJCV*, vol. 77, no. 1-3, pp. 259–289, 2008.

[9] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. CVPR*, 2008, pp. 1–8.

[10] H.-M. Gross, A. Koenig, H.-J. Boehme, and Ch. Schroeter, "Vision-based monte carlo self-localization for a mobile service robot acting as shopping assistant in a home store," in *Proc. IROS*, 2002, vol. 1, pp. 256 – 262.

[11] Donald G Bailey, "An efficient euclidean distance transform," in *Proc. IWCIA*, 2004, pp. 394–408.

[12] Michael Isard and Andrew Blake, "Condensation - conditional density propagation for visual tracking," *IJCV*, vol. 29, pp. 5–28, 1998.