18

Adaptive Real-Time Image Processing for Human-Computer Interaction

Bogdan Kwolek
Rzeszow University of Technology
Poland

1. Introduction

There is a need for computers to be capable to interact naturally with the user, in similar way as human-human interaction takes place. Adding perceptual abilities to computers would enable computers and humans to work jointly more in partner manner. A user-friendly computing system should be aware of the person's context so that it can respond appropriately to user's intentions and anticipate his or her needs. To achieve such functionality, the system needs to integrate a wide variety of visual functions, like localization, object tracking and recognition, action recognition, categorization, interactive object learning.

Human-computer interaction (HCI) is a research discipline that is concerned with the design, implementation and evaluation of interactive computing systems for human use. Most of the research in this area focuses on developing of user-friendly interfaces. Although existing interfaces are designed with human user in mind, many of them are far from being user-friendly end effective. So, if we put in a computer between two people communicating in a natural way, the effective bandwidth and naturalness of the communication will be reduced. Thus, a very important issue is how to accomplish synergism between man and the computer.

The idea of ambient intelligence is a promising area of human-computer interaction. An ambient intelligence environment is sensitive to the presence of people and responsive to their needs. It consists of devices that are integrated into environment and work in concert to support people in carrying out their everyday life activities. The unobtrusive layout should be adaptive, which means that it should change in response to user needs. Instead of single and predefined model, such a system acquires its knowledge via interaction with the user and the usage of multiple modules and pathways. To achieve goals of that kind the research interest has shifted from generic computer vision systems towards vision modules able to solve more specific tasks. However, multi-module vision systems running in real-world scenarios and integrating several different vision behaviors are only sparsely reported in the literature.

The objective of this work is to present the methods and techniques for construction of vision systems that can perform tasks oriented towards human-computer interaction. Particular emphasis is placed on algorithms using images acquired in realistic settings and through employing both static and dynamic information to allow knowledge acquisition and

interpretation. It presents efficient and adaptive algorithms, which can operate over long period of time, in varying illumination conditions, with complex background. It is devoted to adaptive object tracking along with unsupervised learning algorithms. It presents adaptive observation models for probabilistic tracking. Particle filters built on such adaptive models are presented. A template based tracking, where the target is represented by a collection of rectangular regions is discussed. Every such a region votes in a common map reflecting the possible positions and the scales of the object. The mentioned above algorithms were tested using various test sequences. The attention is focused on human head/face, one of the most important features in tasks consisting in people tracking and action recognition. The efficiency of the algorithms is demonstrated in several real scenarios, among others in tasks consisting in person following via a real mobile agent, which is equipped with an on-board camera. The mentioned above algorithms constitute a vision system with adaptation and learning capabilities, which can operate on images with complicated background, taken in highly varying illumination conditions.

2. Current Research

Most of the research in human-computer interaction is connected with analysis of the dynamics of human face and body. HCI research devoted to human face is mainly concentrated on face detection, face recognition, tracking of location of the head/face, tracking of head pose, eye tracking, recognition of head gestures, recognition of facial expressions. Research work that is connected with the human body focuses on human detection, gesture as well as recognition of gestural commands, body motion analysis, human tracking, 3D body tracking. Although great progress has been achieved in human machine interaction, most researches still treat the above ingredients separately and the complete systems are reported only sparsely in the relevant literature. Little work has been done in order to endow computers with ability to see where humans are. Here, we evoke some recent surveys, then we present some methods that have been successfully applied in practice, and finally expand the discussion to areas not covered in previous surveys.

Extensive surveys have been published in several HCI fields such as face detection (Yang et al., 2002), face recognition (Zhao et al., 2003), facial expression analysis (Fasel and Luettin, 2003), gesture recognition (Pavlovic et al., 1997, Mitra and Acharya, 2007), human motion analysis (Gavrila, 1999, Aggarwal and Cai, 1999, Wang et al., 2003). A survey presenting the use of vision in HCI, particularly in the area of head tracking can be found in work (Porta, 2002, Kisacanin, 2005). The authors of work (Jaimes and Sebe, 2007) give an overview of multimodal human machine interfaces.

One of the methods that was applied in practice is eye tracking. The study of eye movements pre-dates the widespread use of personal computers. Eye tracking in the field of human-computer interaction has demonstrated modest growth both as a means of studying the suitability of computer interfaces and as a means of interfacing with the computer. An eye tracker is a device measuring both eye positions and eye movements. The commercially available eye tracking systems are mounted on the participant's head or remotely in front of the participant (Duchowski, 2002). Typically they acquire reflections of the infrared light from both the retina and cornea. The problem of constraining the relationship between the eye and the user is one of the obstacles for incorporation of eye tracking in broader scope of the practice use. Great progress has been made in reducing this barrier, but from the user

perspective the existing solutions remain far from optimal (Jacob et al., 2003). Some recent advances in integrating computer interface and eye tracking make possible a mapping of fixation points to visual stimuli (Crowe et al., 2000, Reeder et al. 2001). The gaze tracker proposed in work (Ji and Zhu, 2004) can perform robust and accurate gaze estimation without calibration through the use of procedure identifying the mapping from the pupil parameters to the coordinates of the screen. The mapping function can generalize to other participants not attending in the training. A survey of work related to eye tracking can be found in (Duchowski, 2002).

The Smart Kiosk System (Rehg et al., 1997) uses vision techniques to detect potential users and decide whether the person is a good candidate for interaction. It utilizes face detection and tracking for gesture analysis when a person is at a close range. CAMSHIFT (Bradski, 1998) is a face tracker that has been developed to control games and 3D graphics through predefined head movements. Such a control is performed via specific actions. When people interact face-to-face they indicate of acknowledgment or disinterest with head gestures. In work (Morency et al., 2007) a vision-based head gesture recognition techniques and their usage for common user interface is studied. Another work (Kjeldsen, 2001) reports successful results of using face tracking for pointing, scrolling and selection tasks. An intelligent wheelchair, which is user-friendly to both the user and people around it by observing the faces of both user and others has been proposed in work (Kuno et al., 2001). The user can control it by turning his or her face in the direction where he or she would like to turn. Owing to observing the pedestrian's face it is able to change the collision avoidance method depending on whether or not he or she notices the wheelchair. In related work (Davis et al., 2001) a perceptual user interface for recognizing predefined head gesture acknowledgements is described. Salient facial features are identified and tracked in order to compute the global 2-D motion direction of the head. A Finite State Machine incorporating the natural timings of the computed head motions has been utilized for modeling and recognition of commands. An enhanced text editor using such a perceptual dialog interface has also been described.

Ambient intelligence, also known as Ubiquitous or Pervasive Computing, is a growing field of computer science that has potential for great impact in the future. The term ambient intelligence (AmI) is defined by the Advisory Group to the European Community's Information Society Technology Program as "the convergence of ubiquitous computing, ubiquitous communication, and interfaces adapting to the user". The aim of AmI is to expand the interaction between human beings and information media via the application of ubiquitous computing devices, which encompass interfaces creating together a *perceptive computer environment* rather than one that relies exclusively on active user input. These information media will be available through new types of interfaces and will allow drastically simplified and more intuitive use. The combination of simplified use and their ability to communicate will result in increased efficiency of the contact and interaction. One of the most significant challenges in AmI is to create high-quality, user-friendly, user-adaptive, seamless, and unobtrusive interfaces. In particular, they should allow to sense far more about a person in order to permit the computer to be more acquainted about the person needs and demands, the situation the person is in, the environment, than current interfaces can. Such devices will be able to either bear in mind past environments they operated in, or proactively set up services in new environments (Lyytinen and Yoo, 2002). Particularly, this includes voice and vision technology.

Human computer interaction is very important in multimedia systems (Emond, 2007) because the interaction is basic necessity of such systems. Understanding the meaning of the user message and also the context of the messages is also of great importance for development of practical multimedia interfaces (Zhou et al., 2005).

Common industrial robots usually perform repeating actions in an exactly predefined environment. In contrast to them service robots are designed for supporting jobs for people in their life environment. These intelligent machines should operate in dynamic and unstructured environment and provide services while interaction with people who are not especially skilled in a robot communication. In particular, service robots should participate in mutual interactions with people and work in partnership with humans. In work (Medioni, 2007) visual perception for personal service robot is discussed. A survey of the research related to human-robot interaction can be found in (Goodrich and Schultz, 2007, Fong et al., 2003). Such a challenging research is critical if we allow robots to become part of our daily life.. Several significant steps towards a natural communication have been done, including use of spoken commands and task specific gestural commands in order to convey complex intent. However, meaningful progress in the development is required before we can accomplish communication that feels effortless. Automating the use of human-machine interfaces is also a substantial challenge.

3. Particle Filtering for Visual Tracking

Assume that a dynamic system is described by the following state-space model

$$\begin{aligned} \mathbf{x}_t &= f(\mathbf{x}_{t-1}, \mathbf{u}_t), \ t = 1,2,..., \\ \mathbf{z}_t &= h(\mathbf{x}_t, \mathbf{v}_t), \ t = 0,1,..., \end{aligned} \tag{1}$$

where $\mathbf{x}_t \in R^n$ denotes the system state, $\mathbf{z}_t \in R^m$ is the measurement, $\mathbf{u}_t \in R^n$ stands for the system noise, $\mathbf{v}_t \in R^m$ express the measurement noise, and n and m are dimensions of $\mathbf{x}_t$ and $\mathbf{z}_t$, respectively. The sequences $\{\mathbf{u}_t\}$ and $\{\mathbf{v}_t\}$ are independent and identically distributed (i.i.d.), independent of each other, and independent of the initial state $\mathbf{x}_0$ with a distribution $\mathbf{p}_0$. For nonlinear models, multi-modal, non-Gaussian or any combination of these models the particle filter provides a Monte Carlo solution to the recursive filtering equation $p(\mathbf{x}_t \mid \mathbf{z}_{1:t}) \propto p(\mathbf{z}_t \mid \mathbf{x}_t) \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}) d\mathbf{x}_{t-1}$, where $\mathbf{z}_{1:t} = \{\mathbf{z}_1, ..., \mathbf{z}_t\}$ denotes all observations from time 1 to current time step t. It approximates $p(\mathbf{x}_t \mid \mathbf{z}_{1:t})$ by a probability mass function

$$\hat{p}(\mathbf{x}_t \mid \mathbf{z}_{1:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta(\mathbf{x}_t - \mathbf{x}_t^{(i)}) \tag{2}$$

where δ is the Kronecker delta function, $\mathbf{x}_t^{(i)}$ are random points and $w_t^{(i)}$ are corresponding, non-negative weights representing a probability distribution, i.e.

$\sum_{i=1}^{N} w_t^{(i)} = 1$. If weights are i.i.d. drawn from an importance density $q(\mathbf{x}_t \mid \mathbf{z}_{1:t})$, their values should be set according to the following formula:

$$w_t^{(i)} \propto \frac{p(\mathbf{x}_t^{(i)} \mid \mathbf{z}_{1:t})}{q(\mathbf{x}_t \mid \mathbf{z}_{1:t})} \tag{3}$$

Through applying the Bayes rule we can obtain the following recursive equation for updating the weights

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{z}_t)} \tag{4}$$

where the sensor model $p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)})$ describes how likely it is to obtain a particular sensor reading $\mathbf{z}_t$ given state $\mathbf{x}_t^{(i)}$, and $p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})$ denotes the probability density function describing the state evolution from $\mathbf{x}_{t-1}^{(i)}$ to $\mathbf{x}_t^{(i)}$. In order to avoid degeneracy of the particles, in each time step new particles are resampled i.i.d. from the approximated conditional density. The aim of the re-sampling (Gordon, 1993) is to eliminate particles with low importance weights and multiply particles with high importance weights. It selects with higher probability particles that have a high likelihood associated with them, while preserving the asymptotic approximation of the particle-based posterior representation. Without re-sampling the variance of the weight increases stochastically over time (Doucet et al., 2000). Given $w_{t-1}^{(i)} = 1/N$, the weighting function is simplified to the following form:

$$w_t^{(i)} \propto \frac{p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{z}_t)} \tag{5}$$

If a filtering algorithm takes the prior $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ as the importance density, the importance function reduces to $q(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{z}_t) = p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})$, and in consequence the weighting equation takes the form $w_t^{(i)} \propto p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)})$. This simplification leads to bootstrap filter (Gordon, 1993) and a variant of a well-known particle filter in computer vision, namely CONDENSATION (Isard and Blake, 1998).

The generic particle filter operates recursively through selecting particles, moving them forward according to a probabilistic motion model that is dispersed by an additive random noise component, then evaluating against the observation model, and finally resampling particles according to their weights in order to avoid degeneracy. The algorithm is as follows:

1. Initialization. Sample $\mathbf{x}_0^{(1)},...,\mathbf{x}_0^{(N)}$ i.i.d. from the initial density $\mathbf{p}_0$
2. Importance Sampling/Propagation. Sample $\mathbf{x}_t^{(i)}$ from $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}^{(i)}),\ i = 1,...,N$

3. Updating. Compute $\hat{p}(\mathbf{x}_t \mid \mathbf{z}_{1:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta(\mathbf{x} - \mathbf{x}_t^{(i)})$ using normalized weighs:

$$w_t^{(i)} = p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)}),$$
$$w_t^{(i)} = w_t^{(i)} / \sum_{j=1}^{N} w_t^{(j)}, \qquad i = 1,...,N$$

4. Resampling. Sample $\mathbf{x}_t^{(1)},...,\mathbf{x}_t^{(N)}$ i.i.d. from $\hat{p}(\mathbf{x}_t \mid \mathbf{z}_{1:t})$

5. $t \leftarrow t+1$, go to step 2.

The particle filter converges to the optimal filter if the number of particles grows to infinity. The most significant property of the particle filter is its capability to deal with complex, non-Gaussian and multimodal posterior distributions. However, the number of particles that is required to adequately approximate the conditional density grows exponentially with the dimensionality of the state space. This can cause some practical difficulties in applications such as articulated body tracking (Schmidt et al., 2006). In such tasks we can observe weakness of the particle filter consisting in that the particles do not cluster around the true state of the object as the time increases, and instead they migrate toward local maximas in the posterior distribution. The track of the object can be lost if particles are too diffused. If the observation likelihood lies in the tail of the prior distribution, most of the particles will become meaningless weights. If the system model is inaccurate, the prediction done on the basis of the system model may be too distant from the expected state. In case the system noise is large and the number of particles is not sufficient, poor predictions can also be caused by the simulation in the particle filtering. To deal with the mentioned difficulties different enhancements to presented above algorithm have been proposed, among others algorithms combining extended Kalman filter/unscented Kalman filter with generic particle filter (Merve, 2001). Such particle filters incorporate the current observation to create the more appropriate importance density than the generic particle filter, which utilizes the prior as the importance density.

4. Head Tracking Using Color and Ellipse Fitting in a Particle Filter

Most existing vision-based tracking algorithms give correct estimates of the state in a short span of time and usually fail if there is a significant inter-frame change in object appearance or change of lighting conditions. These methods generally fail to precisely track regions that share similar statistics with background regions.

Service robots are designed for supporting jobs for people in their life environment. These intelligent machines should operate in dynamic and unstructured environment and provide services while interaction with people who are not especially skilled in a robot communication. A kind of human-machine interaction, which is very interesting and has some practical use is following a person by a mobile robot. This behavior can be useful in several applications including robot programming by demonstration and instruction, which in particular can contain tasks consisting in a guidance a robot to specific place, where the user can point to object of interest. A demonstration is particularly useful at programming of new tasks by non-expert users. It is far easier to point towards an object and demonstrate a track, which robot should follow, than to verbally describe its exact location and the path of movement (Waldherr at al., 2000). Therefore, robust person tracking is important prerequisite to achieve the mentioned above robot skills. However, vision modules of the mobile robot impose several requirements and limitations on the use of known vision

systems. First of all, the vision module needs to be small enough to be mounted on the robot and to derive enough small portion of energy from the battery, which would not cause a significant reduction of working time of the vehicle. Additionally the system must operate at an acceptable speed (Waldherr at al., 2000).
Here, we present fast and robust vision based low-level interface for person tracking (Kwolek, 2004). To improve the reliability of tracking using images acquired from an on-board camera we integrated in probabilistic manner the edge strength along the elliptical head boundary and color within the observation model of the particle filter. The adaptive observation model integrates two different visual cues. The incorporation of information about the distance between the camera and the face undergoing tracking results in robust tracking even in presence of skin colored regions in the background. Our interface has been used to conduct several experiments consisting in recognizing arm-postures while following a person via autonomous robot in natural laboratory environment (Kwolek, 2003a, Kwolek, 2003b).

4.1 State space and dynamics

The outline of the head is modeled in the 2D-image domain as a vertical ellipse that is allowed to translate and scale subject to a dynamical model. The object state is given by $\mathbf{x} = \{x, \dot{x}, y, \dot{y}, s_y, \dot{s}_y\}$, where $\{x, y\}$ denotes the location of the ellipse center in the image, $\dot{x}$ and $\dot{y}$ are the velocities of the center, s_y is the length of the minor axis of the ellipse and $\dot{s}_y$ is the rate at which s_y changes.
Our objective is to track a face in a sequence of images coming from an on-board camera. To achieve robustness to large variations in the object pose, illumination, motion, etc. we use the first-order auto-regressive dynamic model $\mathbf{x}_t = A\mathbf{x}_{t-1} + \mathbf{v}_t$, where A denotes a deterministic component describing a constant velocity movement and $\mathbf{v}_t$ is a multivariate Gaussian random variable. The diffusion component represents uncertainty in prediction.

4.2 Shape and color cues

As demonstrated in (Birchfield, 1998) the contour cues can be very useful to represent the appearance of the tracked objects with distinctive silhouette when a model of the shape can be learned off-line and then adapted over time. The shape of the head is one of the most easily recognizable human parts and can be reasonably well approximated by an ellipse. Therefore a parametric model of the ellipse with a fixed aspect ratio equal to 1.2 is utilized to calculate the likelihood. During tracking the oval shape of each head candidate is verified using the sum of intensity gradients along the head boundary. The elliptical upright outlines as well as masks containing interior pixels have been prepared off-line and stored for the use during tracking. The contour cues can, however, be sensitive to disturbances coming from cluttered background, even when detailed models have been used.
When the contour information is poor or is temporary unavailable color information can be very useful alternative to extract the tracked object. Color information can be particularly useful to support a detection of faces in image sequences because the color as a cue is computationally inexpensive (Swain and Ballard, 1991), robust towards changes in orientation and scaling of an object being in movement. The discriminative ability of color is especially worth to emphasize if a considered object is occluded or is in shadow, what can

be in general significant practical difficulty using edge-based methods. Robust tracking can be accomplished using only a simple color model constructed in the first frame and then accommodated over time. One of the problems of tracking on the basis of analysis of color distribution is that lighting conditions may have an influence on perceived color of the target. Even in the case of constant lighting conditions, the apparent color of the target may change over a frame sequence, since other objects can shadow the target.

Color localization cues can be obtained by comparing the reference color histogram of the object of interest with the current color histogram. Due to the statistical nature, a color histogram can only reflect the content of images in a limited way (Swain and Ballard, 1991). In particular, color histograms are invariant to translation and rotation of the object and they vary slowly with the change of angle of view and with the change in scale. Additionally, such a compact representation is tolerant to noise that can result from imperfect ellipse-approximation of a highly deformable structure and curved surface of face causing significant variations of the observed colors.

A color histogram including spatial information can be calculated using a 2-dimensional kernel centered on the target (Comaniciu et al., 2000). The kernel is used to provide the weight for color according to its distance from the region center. In order to assign smaller weights to the pixels that are further away from the region center a nonnegative and monotonic decreasing function $k:[0,\infty)\rightarrow R$ can be used (Comaniciu et al., 2000). The probability of particular histogram bin *u* at location $\mathbf{x}$ is calculated as

$$d_{\mathbf{x}}^{(u)} = C_r \sum_{l=1}^{L} k\left(\left\| \frac{\mathbf{x}-\mathbf{x}_l}{r} \right\|^2 \right) \delta\left[h(\mathbf{x}_l)-u\right] \tag{6}$$

where $\mathbf{x}_i$ are pixel locations, L is the number of pixels in the considered region, constant r is the radius of the kernel, δ is the Kronecker delta function, and the function $h:R^2\rightarrow\{1,...,K\}$ associates the bin number. The normalization factor C_r ensures that $\sum_{u=1}^{K} d_{\mathbf{x}}^{(u)} = 1$. This normalization factor can be precalculated (Comaniciu et al., 2000) for the utilized kernel and assumed values of r. The 2-dimensional kernels have been prepared off-line and then stored in lookup tables for the future use. The color representation of the target has been obtained by quantizing the ellipse's interior colors into *K* bins and extracting the weighted histogram. To make the histogram representation of the tracked head less sensitive to lighting conditions the HSV color space has been chosen and the V component has been represented by 4 bins while the H and S components obtained the 8-bins representation.

To compare the histogram Q representing the tracked face to the histogram I obtained from the particle position we utilized the metric $\sqrt{1-\rho(I,Q)}$, which is derived from Bhattacharyya coefficient $\rho(I,Q)=\sum_{u=1}^{K}\sqrt{I^{(u)}Q^{(u)}}$. The work (Comaniciu et al., 2000) demonstrated that the utilized metric is invariant to the scale of the target and therefore is superior to other measures such as histogram intersection (Swain and Ballard, 1991) or Kullback divergence.

Based on Bhattacharyya coefficient we defined the color observation model as $p(\mathbf{z}^C \mid \mathbf{x}) = (\sqrt{2\pi}\sigma)^{-1} e^{-\frac{1-\rho}{2\sigma^2}}$. Owing to such weighting we favor head candidates whose color distributions are similar to the distribution of the tracked head. The second ingredient of the observation model reflecting the edge strength along the elliptical head boundary has been weighted in a similar manner $p(\mathbf{z}^G \mid \mathbf{x}) = (\sqrt{2\pi}\sigma)^{-1} e^{-\frac{1-\phi_g}{2\sigma^2}}$, where ϕ_g denotes the normalized gradient along the ellipse's boundary.

4.3 Probabilistic integration of cues

The aim of probabilistic multi-cue integration is to enhance visual cues that are more reliable in the current context and to suppress less reliable cues. The correlation between location, edge and color of an object even if exist is rather weak. Assuming that the measurements are conditionally independent given the state we obtain the equation $p(\mathbf{z}_t \mid \mathbf{x}_t) = p(\mathbf{z}_t^G \mid \mathbf{x}_t) \cdot p(\mathbf{z}_t^C \mid \mathbf{x}_t)$, which allows us to accomplish the probabilistic integration of cues. To achieve this we calculate at each time t the L2 norm based distances $D_t^{(j)}$, between the individual cue's centroids and the centroid obtained by integrating the likelihood from utilized cues (Triesch et al., 2001). The reliability factors of the utilized cues $\alpha_t^{(j)}$ are then calculated on the basis of the following leaking integrator $\xi\dot{\alpha}_t^{(j)} = \eta_t^{(j)} - \alpha_t^{(j)}$, where ξ denotes a factor that determines the adaptation rate and $\eta_t^{(j)} = 0.5 \cdot (\tanh(-aD_t^{(j)}) + b)$. In the experiments we set $a = 0.3$ and $b = 3$. Using the reliability factors the observation likelihood has been determined as follows:

$$p(\mathbf{z}_t \mid \mathbf{x}_t) = \left[p(\mathbf{z}_t^G \mid \mathbf{x}_t)\right]^{\alpha_t^{(1)}} \cdot \left[p(\mathbf{z}_t^C \mid \mathbf{x}_t)\right]^{\alpha_t^{(2)}}, \quad 0 \le \alpha_t^{(j)} \le 1. \tag{7}$$

4.4 Adaptation of the color model

The largest variations in object appearance occur when the object is moving. Varying illumination conditions can influence the distribution of colors in an image sequence. If the illumination is static but non-uniform, movement of the object can cause the captured color to change alike. Therefore, tracker that uses a static color model is certain to fail in unconstrained imaging conditions. To deal with varying illumination conditions the histogram representing the tracked head has been updated over time. This makes possible to track not only a face profile which has been shot during initialization of the tracker but in addition different profiles of the face as well as the head can be tracked. Using only pixels from the ellipse's interior, a new color histogram is computed and combined with the previous model in the following manner $Q_t^{(u)} = (1-\gamma)Q_{t-1}^{(u)} + \gamma I_t^{(u)}$, where γ is an accommodation rate, I_t denotes the histogram of the interior of the ellipse representing the estimated state, Q_{t-1} is the histogram of the target from the previous frame, whereas $u = 1,\ldots,K$.

4.5 Depth cue

In experiments, in which a stereovision camera has been employed, the length of the minor axis of the considered ellipse has been determined on the basis of depth information. The system state presented in Subsection 4.1 contains four variables, namely location and speed. The length has been maintained by performing a local search to maximize the goodness of the observation match. Taking into account the length of the minor axis resulting from the depth information we considered smaller and larger projection scale of the ellipse about two pixels. Owing to verification of face distance to the camera and face region size heuristics it is possible to discard many false positives that are generated through the face detection module.

4.6 Face detection

The face detection algorithm can be utilized to form a proposal distribution for the particle filter in order to direct the particles towards most probable locations of the objects of interest. The employed face finder is based on object detection algorithm described in work (Viola et al., 2001). The aim of the detection algorithm is to find all faces and then to select the highest scoring candidate that is situated nearby a predicted location of the face. Next, taking the location and the size of the window containing the face we construct a Gaussian distribution $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_t)$ in order to reflect the face position in the proposal distribution. The formula describing the proposal distribution has the following form:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_t) = \beta p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_t) + (1-\beta) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \tag{8}$$

The parameter β is dynamically set to zero if no face has been found. In such a situation the particle filter takes the form of the CONDENSATION (Isard and Blake, 1998).

4.7 Head tracking for human-robot interaction

The experiments described in this Section were carried out with a mobile robot Pioneer 2DX (ActivMedia Robotics, 2001) equipped with commercial binocular Megapixel Stereo Head. The dense stereo maps are extracted in that system thanks to small area correspondences between image pairs (Konolige, 1997) and therefore poor results in regions of little texture are often provided. The depth map covering a face region is usually dense because a human face is rich in details and texture, see Fig. 1b. Owing to such a property the stereovision provides a separate source of information and considerably supports the process of approximating the tracked head with an ellipse of proper size.

A typical laptop computer equipped with 2.5 GHz Pentium IV is utilized to run the software operating at images of size 320x240. The position of the tracked face in the image plane as well as person's distance to the camera are written asynchronously in block of common memory, which can be easily accessed by Saphira client. Saphira is an integrated sensing and control system architecture based on a client server-model whereby the robot supplies a set of basic functions that can be used to interact with it (ActivMedia Robotics, 2001). During tracking, the control module keeps the user face within the camera field of view by coordinating the rotation of the robot with the location of the tracked face in the image plane. The aim of the robot orientation controller is to keep the position of the tracked face at specific position in

the image. The linear velocity has been dependent on person's distance to the camera. In experiments consisting in person following a distance 1.3 m has been assumed as the reference value that the linear velocity controller should maintain. To eliminate needless robot rotations as well as forward and backward movements we have applied a simple logic providing necessary insensitivity zone. The PD controllers have been implemented in the Saphira-interpreted Colbert language (ActivMedia Robotics, 2001).

To test the prepared software we performed various experiments. After detection of possible faces, see Fig. 1. a, b, the system can identify known faces among the detected ones, using a technique known as eigenfaces (Turk and Pentland, 1991). In tracking scenarios consisting in realization of only a rotation of mobile robot, which can be seen as analogous to experiments with a pan-camera, the user moved about a room, walked back and forth as well as around the mobile robot. The aim of such scenarios was to evaluate the quality of ellipse scaling in response of varying distance between the camera and the user, see Fig. 1. e, h. Our experimental findings show that owing to stereovision the ellipse properly approximates the tracked head and in consequence, sudden changes of the minor axis length as well as ellipse's jumps are eliminated. The greatest variability is in horizontal motion, followed by vertical motion. Ellipse's size variability is more constrained and tends towards the size from the previous time step. By dealing with multiple cues the presented approach can track a head reliably in cases of temporal occlusions and varying illumination conditions, see also Fig. 1. c, even when person undergoing tracking moves in front of the wooden doors or desks, see also Fig. 1. c - h. Using this sequence we conducted tracking experiments assuming that no stereo information is available. Under such an assumption the system state presented in Subsection 4.1 has been employed. However, considerable ellipse changes as well as window jitter have been observed and in consequence the head has been tracked in only the part of the sequence.

Fig. 1. Face detection in frame #9 (a), depth image (b), #44 (c), #45 (d), #69 (e), #169 (f), #182 (g), #379 (h)

Figure 2. demonstrates some tracking results that were obtained in experiments consisting in person following via the mobile robot. As we can see, the tracking techniques described above allow us to achieve the tracking of the person in real situations, under varying illumination.

The tracker runs with 400 particles at frame rates of 13-14 Hz. The face detector can localize human faces in about 0.1 s. The system processes about 6 frames per second when the information about detected faces is used to generate the proposal distribution for the particle filter. The recognition of single face takes about 0.01 s. These times allow the robot to follow the person moving with a walking speed.

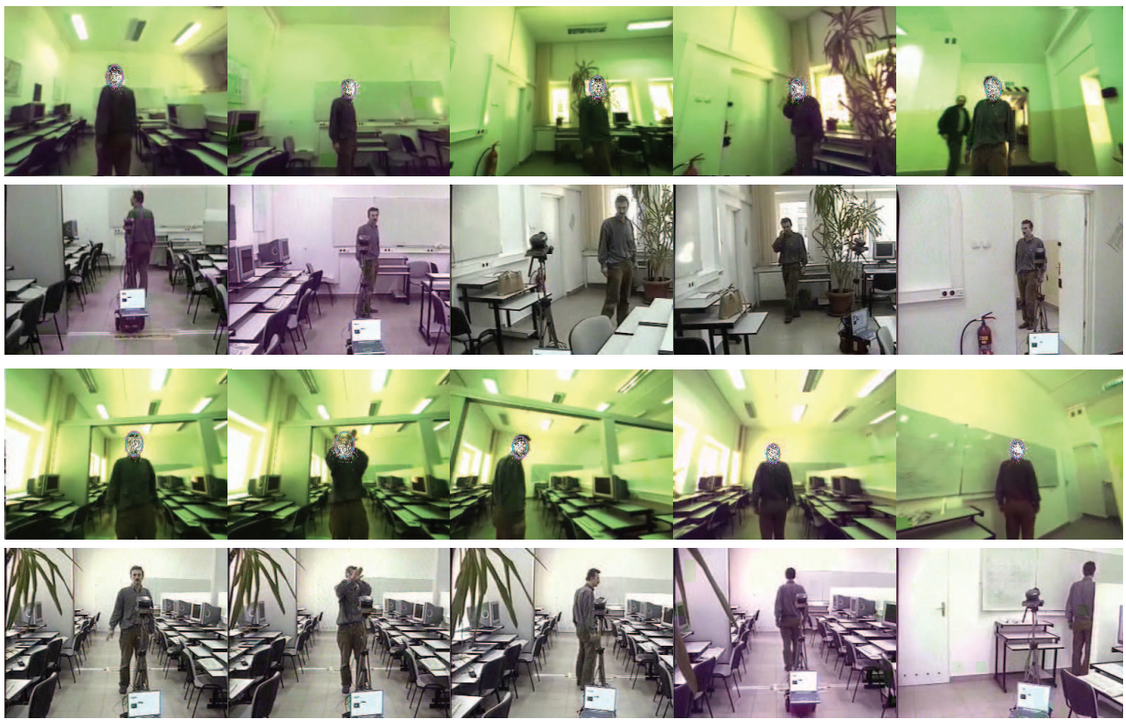

Fig. 2. Person following with a mobile robot. In 1-st and 3-rd row some images from on-board camera are depicted, whereas in 2-nd and 4-th row the corresponding images from an external camera are presented

5. Face Tracking for Human-Computer-Interaction

5.1 Adaptive models for particle filtering

Low-order parametric models of the image motion of pixels laying within a template can be utilized to predict the movement in the image plane (Hager and Belhumeur, 1998). This means that by comparing the gray level values of the corresponding pixels within region undergoing tracking, it is possible to obtain the transformation (giving shear, dilation and rotation) and translation of the template in the current image (Horn, 1986). Therefore, such models allow us to establish temporal correspondences of the target region. They make region-based tracking an effective complement to tracking that is based on classifier distinguishing between foreground and background pixels. In a particle filter the usage of change in transformation and translation $\Delta\boldsymbol{\omega}_{t+1}$ arising from changes in image intensities within the template can lead to reduction of the extent of noise $\mathbf{v}_{t+1}$ in the motion model. It can take the form (Zhou and Chellappa, 2004): $\boldsymbol{\omega}_{t+1} = \hat{\boldsymbol{\omega}}_t + \Delta\boldsymbol{\omega}_{t+1} + \mathbf{v}_{t+1}$.

5.2 Head tracking for human-robot interaction

Let $I_{x,t}$ denote the brightness value at the location $\mathbf{x} = \{x_1, x_2\}$ in an image I that was acquired in time t. Let $\mathcal{R}$ be a set of J image locations $\{\mathbf{x}^{(j)} \mid j = 1,...,J\}$ defining a template. $Y_t(\mathcal{R}) = \{I_{x,t}^{(j)} \mid j = 1,2,...,J\}$ is a vector of the brightness values at locations $\mathbf{x}^{(j)}$ in the template. We assume that the transformations of the template can be modeled by a parametric motion model $g(\mathbf{x};\boldsymbol{\omega}_t)$, where $\mathbf{x}$ denotes an image location and $\boldsymbol{\omega}_t = \{\omega_t^{(1)}, \omega_t^{(2)},..., \omega_t^{(l)}\}$ denotes a set of l parameters. The image variations of planar objects that undergo orthographic projection can be described by a six-parameter affine motion models (Hager and Belhumeur, 1998):

$$g(\mathbf{x};\boldsymbol{\omega}) = \begin{bmatrix} a & d \\ c & e \end{bmatrix} \mathbf{x} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = A\mathbf{x} + \mathbf{u} \tag{9}$$

where $\boldsymbol{\omega} = (a, c, d, e, u_1, u_2)^T$. With these assumptions, the tracking of the object in time t can be achieved by computing $\boldsymbol{\omega}_{t+1}$ such that $Y_{t+1}(g(\mathcal{R};\boldsymbol{\omega}_{t+1})) = \hat{Y}_t(\mathcal{R})$, where the template $\hat{Y}_t(\mathcal{R})$ is in pose determined by the estimated state.

Given a set $S = \{\boldsymbol{\omega}_t^{(n)}, w_t^{(n)} \mid n = 1,..., N\}$ of weighted particles, which approximate the posterior distribution $p(\boldsymbol{\omega}_t \mid Y_{1:t})$, the maximum aposteriori estimate (MAP) of the state is calculated according to the following formula:

$$\hat{\boldsymbol{\omega}}_t = \arg\max_{\boldsymbol{\omega}_t} p(\boldsymbol{\omega}_t \mid Y_{1:t}) \approx \arg\max_{\boldsymbol{\omega}_t} w_t^{(n)} \tag{10}$$

The motion parameters in time $t+1$ take values according to the following formula:

$$\boldsymbol{\omega}_{t+1} = \hat{\boldsymbol{\omega}}_t + A_{t+1}[\hat{Y}_t(\mathcal{R}) - Y_{t+1}(g(\mathcal{R};\hat{\boldsymbol{\omega}}_t))] . \tag{11}$$

This equation can be expressed as follows: $\Delta\boldsymbol{\omega}_{t+1} = A_{t+1}\Delta\mathbf{y}_{t+1}$. Given N measurements we can estimate matrix A_{t+1} from matrices consisting of adjoined vectors $\Delta\boldsymbol{\omega}_{t+1}$ and $\Delta\mathbf{y}_{t+1}$ (Horn, 1986):

$$\Delta M_t = [\hat{\omega}_t^{(1)} - \omega_t^{(1)},..., \hat{\omega}_t^{(N)} - \omega_t^{(N)}] \tag{12}$$

$$\Delta Y_t = [\hat{Y}_t^{(1)} - Y_t^{(1)},..., \hat{Y}_t^{(N)} - Y_t^{(N)}] . \tag{13}$$

Using the least squares (LS) method we can find the solution for A_{t+1} (Horn, 1986):

$$A_{t+1} = (\Delta M_t \Delta Y_t^T)(\Delta Y_t \Delta Y_t^T)^{-1} . \tag{14}$$

Singular value decomposition of ΔY_t yields: $\Delta Y_t = UWV^T$. Taking q largest diagonal elements of W the solution for A_{t+1} is as follows: $A_{t+1} = \Delta M_t V_q W_q^{-1} U_q^T$. The value of q

depends on the number of diagonal elements of W that are below a predefined threshold value.
In the CONDENSATION algorithm we utilized the following motion model:

$$\boldsymbol{\omega}_{t+1} = \hat{\boldsymbol{\omega}}_t + \Delta\boldsymbol{\omega}_{t+1} + \mathbf{v}_{t+1}, \tag{15}$$

where $\mathbf{v}_{t+1}$ is zero mean Gaussian i.i.d. noise, independent of state and with covariance matrix Q which specifies the extent of noise.
When individual measurements carry more or less weight, the individual rows of $\Delta\boldsymbol{\omega} = A\Delta\mathbf{y}$ can be multiplied by a diagonal matrix with weighting factors. If the diagonal matrix is the identity matrix we obtain the original solution. In our approach such row weighting is used to emphasize or de-emphasize image patches according to number of background pixels they contain. The background pixels can be detected by a supplementing tracker, built on different cues, with different failure mode.

5.3 Appearance modeling using adaptive models

Our intensity-based appearance model consists of three components, namely, the *W*-component expressing the two-frame variations, the *S*-component characterizing the stable structure within all previous observations and *F*-component representing a fixed initial template. The model $A_t = \{W_t, S_t, F_t\}$ represents thus the appearances existing in all observations up to time $t-1$. It is a mixture of Gaussians (Jepson et al., 2001) with centers $\{\mu_{i,t} \mid i = w, s, f\}$, their corresponding variances $\{\sigma_{i,t}^2 \mid i = w, s, f\}$ and mixing probabilities $\{m_{i,t} \mid i = w, s, f\}$.

The update of the current appearance model A_t to A_{t+1} is done using the Expectation Maximization (EM) algorithm. For a template $\hat{Y}(\mathcal{R}_t, t)$ corresponding to the estimated state we evaluate the posterior contribution probabilities as follows:

$$o_{i,t}^{(j)} = \frac{m_{i,t}^{(j)}}{\sqrt{2\pi\sigma_{i,t}^2}} \exp\left[-\frac{\hat{I}_{x,t}^{(j)} - \mu_{i,t}^{(j)}}{2\sigma_{i,t}^2} \right] \tag{16}$$

where $i = w, s, f$ and $j = 1,2,\ldots,J$. If the considered pixel belongs to background, the posterior contribution probabilities are calculated using $\hat{I}_{x,1}^{(j)}$:

$$o_{i,t}^{(j)} = \frac{m_{i,t}^{(j)}}{\sqrt{2\pi\sigma_{i,t}^2}} \exp\left[-\frac{\hat{I}_{x,1}^{(j)} - \mu_{i,t}^{(j)}}{2\sigma_{i,t}^2} \right] \tag{17}$$

This prevents the slowly varying component from updating by background pixels. The posterior contribution probabilities (with $\sum_i o_{i,t}^{(j)} = 1$) are utilized in updating the mixing probabilities in the following manner:

$$m_{i,t+1}^{(j)} = \gamma o_{i,t}^{(j)} + (1-\gamma) m_{i,t}^{(j)} \quad | i = w,s,f \ , \tag{18}$$

where γ is accommodation factor. Then, the first and the second-moment images are determined as follows:

$$\begin{aligned} M_{1,i+1}^{(j)} &= (1-\gamma) M_{1,t}^{(j)} + \gamma o_{s,t}^{(j)} \hat{I}_{x,t}^{(j)} \\ M_{2,i+1}^{(j)} &= (1-\gamma) M_{2,t}^{(j)} + \gamma o_{s,t}^{(j)} (\hat{I}_{x,t}^{(j)})^2 \ . \end{aligned} \tag{19}$$

In the last step the mixture centers and the variances are calculated as follows:

$$\begin{aligned} \mu_{s,t+1}^{(j)} = \frac{M_{1,t+1}^{(j)}}{m_{s,t+1}^{(j)}}, \ \sigma_{s,t+1}^{(j)} &= \sqrt{\frac{M_{2,t+1}^{(j)}}{m_{s,t+1}^{(j)}} - (\mu_{s,t+1}^{(j)})^2} \\ \mu_{w,t+1}^{(j)} = \hat{I}_{x,t}^{(j)}, \ \sigma_{w,t+1}^{(j)} &= \sigma_{w,1}^{(j)} \\ \mu_{f,t+1}^{(j)} = \mu_{t,1}^{(j)}, \ \sigma_{f,t+1}^{(j)} &= \sigma_{f,1}^{(j)} \ . \end{aligned} \tag{20}$$

When the considered pixel belongs to background, the mixture center in the component expressing two-frame variations is updated according to:

$$\mu_{w,t+1}^{(j)} = \hat{I}_{x,l}^{(j)} , \tag{21}$$

where index l refers to last non-background pixel.

In order to initialize the model A_1 the initial moment images are set using the following formulas: $M_{1,1} = m_{s,1} I(\mathcal{R}, t_0)$ and $M_{2,1} = m_{s,1}(\sigma_{s,1}^2 + I(\mathcal{R}, t_0)^2)$. The observation likelihood is calculated according to the following equation:

$$p(Y_t \mid \boldsymbol{\omega}_t) = \prod_{j=1}^{J} \sum_{i=w,s,f} \frac{m_{i,t}^{(j)}}{\sqrt{2\pi\sigma_{i,t}^2}} \exp\left[-\frac{I_{x,t}^{(j)} - \mu_{i,t}^{(j)}}{2\sigma_{i,t}^2} \right] \tag{22}$$

In the particle filter we use a recursively updated mixture appearance model, which depicts stable structures in images seen so far, initial object appearance as well as two-frame variations. If a supplementing tracker with different failure mode is used, the update of slowly varying component is done using only pixels that are labeled as belonging to foreground. In pairwise comparison of object images we employ only non-background pixels and in case of background we use the last foreground pixels. Our probabilistic models differ from those proposed in (Zhou and Chellappa, 2004) in that we adapt models using information about background.

5.4 Face tracking using particle filter built on adaptive appearance models

Figure 3. shows tracking the face of a person on images acquired by a camera mounted on the computer monitor. The experiments have been done in a typical home environment. The camera was located in front of wooden doors or furniture. In such a scenario a typical color based tracker can have severe difficulties in following the face because the skin color form clusters, which overlap with colors of pixels belonging to wood. In the depicted images we can see that the jitter of the tracking window is relatively low. However, this gray-level image based tracker failed in frame #36.

Fig. 3. Face tracking using particle filter built on adaptive appearance models. Frames #1, #20, #25, #30, #35 (from left to right and from top to bottom)

Figure 4. depicts some tracking results which have been obtained in experiments with partial occlusions of the face undergoing tracking. In image #25 we can observe that despite of the considerable occlusion of the face, the template's location has not been shifted from desirable location. However, in images #45, #55 we can notice that in response to the occlusion of the face the template has been moved to left side. Therefore, the part of the template learned some background pixels. After the occluding book has been moved to the right a considerable jitter of the template has been observed in our experiment. Due to the mentioned above undesirable effects the tracking failed in some of our other experiments.
In Fig. 5. we present some tracking results that were obtained on another test sequence[1]. However, as we can see, the tracker failed in frame #33. In this experiment far larger tracking window has been utilized in order to cover the area of the face to be tracked.

6. Tracking using Multi-Part Object Representation

6.1 The algorithm

In work (Pérez et al., 2002) it has been demonstrated that multi-part object representation leads to better tracking. The authors proposed a method consisting in dividing the object to be

[1] Available on: http://research.microsoft.com/vision/cambridge/i2i

Fig. 4. Face undergoing tracking via particle filter built on adaptive appearance models. Frames #1, #15, #25, #45, #55 (from left to right and from top to bottom)

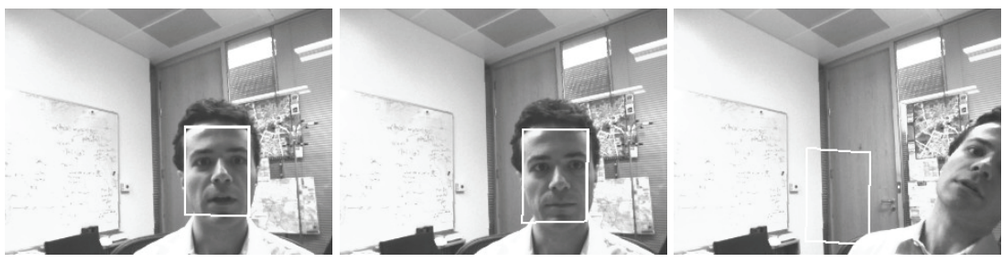

Fig. 5. Face tracking using particle filter built on adaptive appearance models. Frames #1, #32, #33 (from left to right and from top to bottom)

tracked into non-overlapping rectangular regions and then computing in each such a sub-region a color histogram. The observation model has been constructed under assumption that image data in different sub-regions are independent. The object likelihood was proportional to the exponential of negative sum of squared distances between histograms. In (Fieguth et al., 1997), rectangle sub-regions that are defined within the extent of the object to be tracked are employed to calculate the averaged color of pixels. The object tracking is achieved through an exhaustive search in a confidence region. An occlusion model has been developed to discriminate between good and spurious measurements.

Cognitive science states that complex entities are perceived as composition of simple elements. Objects are represented through such components and the relations between them (Ommer et al., 2006). One of the disadvantages of color histograms is the loss of spatial information. To incorporate such information in the object representation we divide the object template into adjoining cells, regularly spaced within the extent of the target to be tracked. We compute histograms within such regularly spaced cells using a fast method that has been proposed in (Porikli et al., 2005). Given the estimated position in the previous frame and the histograms within object at the estimated position we employ the chi-square test between such histograms and histograms extracted from cells within the template at candidate position.

The χ^2 test is given by: $\chi^2 = \sum_i ((h_{e,i} - h_{c,i})^2 / (h_{e,i} + h_{c,i})^2)$, where $h_{e,i}$ and $h_{c,i}$ represent the number of entities in the i-th bin of the histograms, and a low value for χ^2 indicates a good match. The distances are transformed into likelihoods through the usage of exponential function. We seek for the object in the new frame within a search window, which center is located at the previous object position. At each candidate position we compare the corresponding histograms and the result is utilized to vote for the considered position. Every cell votes in its own map. Then we combine the votes in a relevance map, which encodes hypothesis where the target is located in the image. In order to cope with partial occlusions we employ a simple heuristics aiming at detecting outliers. If the difference between corresponding histograms is below a certain level the score in the relevance map remains unchanged. The level for each cell is determined individually using the distances between histograms from the initial template and corresponding histograms from few first frames. Similar test is performed with respect to the actual medians.

6.2 Face tracking using multi-part representation

In Fig. 6 we can observe that this algorithm temporally failed in frame #35 and then recovered in later images. However, as we can see in frames #53 and #77 a considerable jitter accompanied the tracking of the face. The tracker is able to track the face under severe rotations, see frames #134 - #173. However, the tracking of the face ends with considerable jitter of the window, see frame #175.

Figure 7. shows some tracking results in case of partial occlusions of the face. In image #25 we can see that despite of the occlusion the window of the tracker has not been shifted from desirable location, see also Fig. 4. However, the occlusion of the face from the opposite side caused severe shifts of the tracked window, see also frame #48. In consequence, the tracker temporally failed in frame #55 and then recovered in the next frames.

In the tracking results presented in Fig. 8 we can see that the tracker can fail in some images. Its advantage is that it can deal with severe rotations of face undergoing tracking. However, in order to achieve such functionality, relatively large tracking windows should be employed in the course of the tracking. In the discussed experiments the template has been divided into four non-overlapping sub-regions. In order to take advantages of the tracker in the last sequence, the number of sub-regions should be something larger, say eight sub-regions.

7. Combined Face Tracker

Obtaining a collection consisting of both positive and negative examples for on-line learning is a complex task. In work (Zeki, 2001) the author argues that the human visual system consists of a number of interacting but still autonomously operating subsystems that process different object representations. Within such a mechanism, the subsystems can serve mutually in course of learning and bootstrapping of object representations. This inclined us to construct an object tracker consisting of two independent trackers with different failure modes, complementing each other and operating in co-training framework. The co-training approach has been employed in previous work (Levin et al., 2004), in which a tracker starts with a small training set and increases it by co-training of two classifiers, operating on different features.

In the combined algorithm we utilize the trackers presented in the last two Sections. Our

Fig. 6. Face tracking using multi-part object representation in case of occlusions. Frames #1, #15, #25, #45, #46, #48, #55, #65, #70 (from left to right and from top to bottom)

Fig. 7. Face tracking using multi-part object representation in case of occlusions.

Fig. 8. Face tracking using multi-part representation. Frames #1, #51, #86, #112, #126, #200

experiments demonstrated that even very simple combined algorithm based on the distance between the locations of windows, which have been determined via the employed trackers, leads to considerable improvement of tracking, see Fig. 9 and 10. As we already mentioned in the previous Section, in our combined tracker the adaptive appearance models are learned on-line using only foreground pixels, owing to the estimates determined by the complementary tracker. The information about distance between the locations of windows allows us to detect such pixels easily. What more, given such information we can accommodate the histograms of the object parts, which likely belong to the tracked object. In consequence, the drift of the part-based tracker is smaller. This in turn allows us to accommodate the histograms with rather lower risk that they will be updated via non-object pixels.
In the results depicted in Fig. 9 the size of the windows of both trackers has been estimated on the basis of depth map, which has been determined by the stereovision system. The depth map has also been utilized in labeling of the pixels for the adaptation. Assuming that the face is relatively flat we exclude pixels too distant from the distance of the template to the camera. Typically, in the combined face tracker the distance between the windows of the utilized trackers is small. In the part-based tracker we tried to determine the rotation of the template though searching for the best one using information on the similarities of the histograms. However, in the course of the tracking the orientation of the template estimated in such a way considerably fluctuates about the true template orientation. Therefore, better results were achieved using the orientation estimated by adaptive appearance model based tracker. This in turn allowed us to accommodate the histogram in experiments like these depicted in Fig. 11.

8. Conclusions

We have discussed adaptive algorithms for face tracking. Face tracking is one of the most important research directions in human machine interaction with many possible applications, for example see work (Tu et al., 2007). The algorithms were validated in various test sequences and demonstrated their great potential to be used in applications employing human face to complement the opportunities offered by the computer mouse.

Fig. 9. Face tracking via combined algorithm. Frames #1, #20, #25, #30, #35, #40 #59, #85, #104

Fig. 10. Tracking of the face undergoing occlusion using combined algorithm. Frames #1, #15, #25, #45, #55

The distance map provided by stereovision system offers great possibilities to construct robust algorithms for human-machine interaction. The proposed algorithms for adaptation and unsupervised learning were tested in real home/laboratory scenarios, and acknowledged great ability to deal with varying illumination conditions as well as complex background.

Fig. 11. Face tracking using combined algorithm. Frames #1, #51, #86, #112, #126, #135

9. References

Aggarwal, J. K.; Cai, Q. (1999). Human motion analysis: A review, *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428-440

Birchfield, S. (1998). Elliptical head tracking using intensity gradients and color histograms, *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 232-237

Bradski, G. (1998). Computer vision face tracking for use in a perceptual user interface, *Intel Technoloy Journal*, vol. 2, no. 2, pp. 12-21

Comaniciu, D.; Ramesh, V.; Meer P. (2000). Real-time tracking of non-rigid objects using Mean Shift, *Proc. Int. Conf. on Computer Vision and Pattern Recognition*, pp. 142-149

Crowe, E. C.; Narayann, N. H. (2000). Comparing interfaces based on what users watch and do, *Proc. Eye Tracking Research and Applications Symposium*, ACM, pp. 29-36

Davis, J.; Vaks, S. A. (2001). Perceptual user interface for recognizing head gesture acknowledgements, *ACM Workshop on Perceptual User Interfaces*, pp. 1-7

Doucet, A.; Godsill, A.; Andrieu, Ch. (2000). On sequential Monte Carlo sampling methods for bayesian filtering, *Statistics and Computing*, vol. 10, pp. 197-208

Duchowski, A. T. (2002). A breadth-first survey of eye tracking applications, *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 4, pp. 455-470

Emond, B. (2007). Multimedia and human-in-the-loop: interaction as content enrichment, *Proc. of the ACM Int. Workshop on Human-Centered Multimedia*, pp. 77-84

Fasel, B.; Luettin, J. (2003). Automatic facial expression analysis: A survey, *Pattern Recognition*, vol. 36, pp. 259-275

Fieguth, P.; Terzopoulos, D. (1997). Color-based tracking of heads and other mobile objects at video frame rates. *Proc. IEEE Conf. on Comp. Vision and Patt. Rec.*, pp. 21-27

Fong, T.; Nourbakhsh, I.; Dautenhahn, K. (2003). A survey of socially interactive robots, *Robotics and Autonomous Systems*, vol. 42, pp. 143-166

Gavrila, D. (1999). The visual analysis for human movement: A survey, *Computer Vision and Image Understanding*, vol. 13, no. 1, pp. 82-98

Goodrich, M. A.; Schultz, A. C. (2007). Human-robot interaction: A survey, *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203-275

Gordon, N.; Salmond, D.; Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEEE Trans. Radar, Signal Processing*, vol. 140, pp. 107-113

Hager, G. D.; Belhumeur, P. N. (1998). Efficient region tracking with parametric models of geometry and illumination, *IEEE Trans. on PAMI,* vol. 20, pp. 1025-1039

Horn, B. K. P. (1986). Robot vision, The MIT Press

Isard, M.; Blake A. (1998). CONDENSATION - conditional density propagation for visual tracking, *Int. Journal of Computer Vision*, vol. 29, pp. 5-28

Jacob, R. J. K.; Karn, K. S. (2003). Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. In: R. Radach, J. Hyona, and H. Deubel (eds.), The mind's eye: cognitive and applied aspects of eye movement research, Boston: North-Holland/Elsevier, pp. 573-605

Jaimes, A.; Sebe, N. (2007). Multimodal human computer interaction: A survey, *Computer Vision and Image Understanding*, no. 1-2, pp. 116-134

Jepson, A. D.; Fleet, D. J.; El-Maraghi, T. (2001). Robust on-line appearance models for visual tracking, *Int. Conf. on Comp. Vision and Pattern Rec.*, pp. 415-422

Ji, Q.; Zhu, Z. (2004). Eye and gaze tracking for interactive graphic display, *Machine Vision and Applications*, vol. 15, no. 3, pp. 139-148

Kisacanin, B.; Pavlovic, V.; Huang, T. S. (eds.) (2005). Real-time vision for human-computer interaction, Springer-Verlag, New York

Kjeldsen, R. (2001). Head gestures for computer control, *IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pp. 61-67

Konolige, K. (1997). Small Vision System: Hardware and implementation, *Proc. of Int. Symp. on Robotics Research*, Hayama, pp. 111-116

Kuno, Y.; Murakami, Y.; Shimada, N. (2001). User and social Interfaces by observing human faces for intelligent wheelchairs, *ACM Workshop on Perceptive User Interfaces*, pp. 1-4

Kwolek, B. (2003a). Person following and mobile robot via arm-posture driving using color and stereovision, In *Proc. of the 7th IFAC Symposium on Robot Control SYROCO*, *Elsevier IFAC Publications*, (J. Sasiadek, I. Duleba, eds), Elsevier, pp. 177–182

Kwolek, B. (2003b). Visual system for tracking and interpreting selected human actions, *Journal of WSCG*, vol. 11, no. 2, pp. 274-281

Kwolek, B. (2004). Stereovision-based head tracking using color and ellipse fitting in a particle filter, *European Conf. on Comp. Vision*, *LNCS,* vol. 3691, Springer, 2004, 192–204

Levin, A.; Viola, P.; Freund, Y. (2004). Unsupervised improvement of visual detectors using co-training, *Proc. Int. Conf. on Comp. Vision*, 626-633, vol. 1

Lyytinen, K.; Yoo, Y. J. (2002). Issues and challenges in ubiquitous computing, *Communications of the ACM*, vol. 45, no. 12, pp. 62-70

Medioni, G.; Francois, A. R. J.; Siddiqui, M.; Kim, K.; Yoon, H. (2007). Robust real-time vision for a personal service robot, *Computer Vision and Image Understanding*, Special issue on vision for HCI, vol. 108, pp. 196-203

Merve, R.; Freitas, N.; Doucet, A.; Wan, E. (2000). The unscented particle filter, *Advances in Neural Information Processing Systems*, vol. 13, pp. 584-590

Mitra, S.; Acharya, T. (2007). Gesture recognition: A survey, *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 3, pp. 311-324

Morency, L. P.; Sidner C. L.; Lee, Ch.; Darrell, T. (2007). Head gestures for perceptual interfaces: The role of context in improving recognition, *Artificial Intelligence,* vol. 171, no. 8-9, pp. 568-585

Ommer, B.; Buhmann, J. M. (2006). Learning compositional categorization models, *European Conf. on Computer Vision*, pp. III:316-329

Pérez, P.; Hue, C.; Vermaak, J.; Gangnet, M. (2002). Color-based probabilistic tracking, *European Conf. on Computer Vision*, LNCS, vol. 2350, pp. 661-675

Porikli, F. (2005). Integral histogram: A fast way to extract histogram in cartesian spaces, *IEEE Computer Society Conf. on Pattern Rec. and Computer Vision*, pp. 829-836

Porta, M. (2002). Vision-based interfaces: methods and applications, *Int. J. Human-Computer Studies*, vol. 57, no. 1, 27-73

Pavlovic, V.; Sharma, R.; Huang, T. (1997). Visual interpretation of hand gestures for human-computer interaction: A review, *IEEE Trans. on PAMI,* vol. 19, pp. 677-695

Rehg, J. M.; Loughlin, M.; Waters, K. (1997). Vision for smart kiosk, *Proc. IEEE Comp. Society Conf. on Computer Vision and Pattern Recognition*, pp. 690-696

Reeder, R. W.; Pirolli, P.; Card, S. K. (2001). WebEyeMapper and WebLogger: Tools for analyzing eye tracking data collected in web-use studies, *Int. Conf. on Human Factors in Computing Systems*, ACM Press, pp. 19-20

Schmidt, J.; Fritsch, J.; Kwolek, B. (2006). Kernel particle filter for real-time 3D body tracking in monocular color images, *IEEE Int. Conf. on Face and Gesture Rec.*, Southampton, UK, IEEE Comp. Society Press, pp. 567-572

Swain, M. J.; Ballard, D. H. (1991). Color indexing, *Int. J. of Computer Vision*, vol. 7, pp. 11-32

Triesch, J.; von der Malsburg, Ch. (2001). Democratic integration: Self-organized integration of adaptive cues, *Neural Computation*, vol. 13, pp. 2049-2074

Tu, J.; Tao, H.; Huang, H. (2007). Face as a mouse through visual face tracking, *Computer Vision and Image Understanding*, Special issue on vision for HCI, vol. 108, pp. 35-40

Turk, M. A.; Pentland, A. P. (1991). Face recognition using eigenfaces, *Proc. of IEEE Conf. on Comp. Vision and Patt. Rec.*, pp. 586-591

Viola, P.; Jones, M. (2001). Rapid object detection using a boosted cascade of simple features, *The IEEE Conf. on Comp. Vision and Patt. Rec.*, pp. 511–518

Wang, L.; Hu, W. M.; Tan, T. N. (2003). Recent developments in human motion analysis, *Pattern Recognition*, vol. 36, no. 3, pp. 585-601

Waldherr, S.; Romero S.; Thrun, S. (2000). A gesture-based interface for human-robot interaction, *Autonomous Robots*, vol. 9, pp. 151-173

Yang, M-H.; Kriegman, D.; Ahuja, N. (2002). Detecting faces in images: A survey, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34-58

Zeki, S. (2001). Localization and globalization in conscious vision, *Annual Review Neuroscience*, 24, pp. 57-86

Zhao, W.; Chellappa, R.; Phillips, P.; Rosenfeld, A. (2003). Face recognition: A literature survey, *ACM Computing Surveys,* vol. 35, no. 4, pp. 399-458

Zhou, S. K.; Chellappa, R.; Moghaddam, B. (2004). Appearance tracking using adaptive models in a particle filter, *Proc. Asian Conf. on Comp. Vision*

Zhou, M. X.; Wen Z.; Aggarwal, V. (2005). A graph-matching approach to dynamic media allocation in intelligent multimedia interfaces, *Proc. of ACM Conf. on Intelligent User Interfaces*, pp. 114-121

ActivMedia Robotics (2001). Pioneer 2 mobile robots

Human Computer Interaction: New Developments

Edited by Kikuo Asai

ISBN 978-953-7619-14-5
Hard cover, 382 pages
Publisher InTech
Published online 01, October, 2008
Published in print edition October, 2008

The book consists of 20 chapters, each addressing a certain aspect of human-computer interaction. Each chapter gives the reader background information on a subject and proposes an original solution. This should serve as a valuable tool for professionals in this interdisciplinary field. Hopefully, readers will contribute their own discoveries and improvements, innovative ideas and concepts, as well as novel applications and business models related to the field of human-computer interaction. It is our wish that the reader consider not only what our authors have written and the experimentation they have described, but also the examples they have set.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen