A Real-Time Head Tracker Supporting Human Computer Interaction

Bogdan Kwolek

Rzeszów University of Technology, W. Pola 2, 35-959 Rzeszów, Poland
bkwolek@prz.rzeszow.pl

Summary. This paper describes a fast and completely automatic algorithm for human face tracking. The tracked face is represented by a weighted histogram. The current histogram is compared to histograms at the particles' positions. The weight of each particle is determined on the basis of Bhattacharyya distance and intensity gradient along the ellipse's boundary. 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. The initialization of the tracker is realized by means of face detection. The detection is carried out using Haar-like features, followed by the verification of face distance to the camera and face region size heuristics.

1 Introduction

Fulfilling the idea of machines that interact face to face with people forces us to think in new ways about computers that could be used in daily life. Within the past decade, significant advances in machine learning and perception open up the possibility of understanding human actions. To obtain a high level interpretation of human actions one must first detect humans. There are a variety of approaches to human detection, mainly focusing on face detection [18].

The visual tracking of objects of interests has become an elementary task in many applications, including surveillance, human-machine interfaces, smart environments, and many more. However, the majority of available algorithms assume that the camera is mounted at a fixed location. Most existing vision-based tracking algorithms give correct estimates of the state in a short span of time and often fail if there is a significant inter-frame change in object appearance. These methods generally fail to precisely track regions that share similar statistics with background regions. To improve the reliability of tracking in such circumstances we integrated in probabilistic manner the edge strength along the elliptical head boundary and color within the observation model of the particle filter. Particle filters provide a means to track the state of an object even if the dynamics and observations are non-linear/non-Gaussian [6][7].

The incorporation of information about the distance between the camera and the face undergoing tracking results in robust tracking on the basis of images acquired from a moving camera even in presence of skin colored regions in the background. In order to initialize the tracker, or reinitialize the system if the tracking fails, we adopt the fast and efficient face detecting method of Viola and Jones [17]. The face detector finds the location and size of each region containing the frontal face in an input image. Next, using the face location, the eigenfaces algorithm [16] is utilized to identify the robot user.

In tracking techniques [1][2][4], the current frame is searched for a region whose colors content best matches a reference color model. The searching starts from the final location in the previous frame and proceeds iteratively to find the minimum distance to the reference color histogram. Global color reference models and Bhattacharyya coefficient as a similarity measure between the color distribution of the model and target candidates have been used in a particle filter-based tracker [10]. A histogram representation of the region of interest has been extracted in a rectangular window. In work [3] an ellipse is used to approximate the head outline during 2D tracking on the basis of a particle filter. Darrell, at al. [5] combine stereo and color via an intensity pattern classification method to track people. The CMU face detector [12] has been used to distinguish the frontal face from other body parts. Over the years various strategies for face detection have been proposed in the literature [18]. The Viola-Jones system [17] was the first for real-time frontal face detection.

The remainder of the paper is organized as follows. In the next section we briefly outline particle filtering. In section 3 we present all ingredients of our tracker and demonstrate how color and contour cues can be integrated to improve the performance of the tracker. Then we describe the face detection algorithm. Section 4 reports results which were obtained in experiments with a moving camera. Finally, some conclusions follow in the last section.

2 Particle Filtering for Visual Tracking

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 $\mathbf{x}_t$ and $\mathbf{z}_t$ denote the hidden state of the object of interest and the observation vector at discrete time t, respectively, whereas $\mathbf{z}_{1:t} = \{\mathbf{z}_1 ... \mathbf{z}_t\}$ denotes all the observations up to current time step. With this recursion we can calculate the posterior, given a dynamic model $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ describing the state propagation and an observation model $p(\mathbf{z}_t \mid \mathbf{x}_t)$ describing the likelihood that a state $\mathbf{x}_t$ causes the measurement $\mathbf{z}_t$. Starting with a weighted particle set $S = \left\{ (\mathbf{x}_{t-1}^{(n)}, \pi_{t-1}^{(n)}) \mid n = 1...N \right\}$ approximately distributed according to $p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1})$ the particle filter operates through predicting new particles from a proposal distribution. To give a new particle representation

$S = \left\{ (\mathbf{x}_t^{(n)}, \pi_t^{(n)}) \mid n = 1...N \right\}$ of the posterior density $p(\mathbf{x}_t \mid \mathbf{z}_{1:t})$ the weights of particles are set to $\pi_t^{(n)} \propto \pi_{t-1}^{(n)} p(\mathbf{z}_t | \mathbf{x}_t^{(n)}) p(\mathbf{x}_t^{(n)} | \mathbf{x}_{t-1}^{(n)}) / q(\mathbf{x}_t^{(n)} | \mathbf{x}_{t-1}^{(n)}, \mathbf{z}_t)$.

From time to time the particles should be resampled according to their weights to avoid degeneracy. The resampling 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 resampling the variance of the weight increases stochastically over time [6]. When the proposal distribution is chosen as the distribution conditioning the state at the previous time step, the importance function reduces to $q(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)}, \mathbf{z}_t) = p(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)})$ and in consequence the weighting equation takes the form $\pi_t^{(n)} \propto p(\mathbf{z}_t \mid \mathbf{x}_t^{(n)})$. This simplification leads to a variant of a well-known particle filter in computer vision, Condensation [7].

3 State Space and Observation Model

The observation model integrates two different visual cues. We construct a likelihood model for each of the cues. The motion model will be presented as the first topic in this section. The observation model in which the multiple cue integration takes place will be discussed in detail later. The model adaptation over time will be presented afterwards. An outline of face detection algorithm ends this section.

3.1 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 $\{x, \dot{x}, y, \dot{y}, s_y, \dot{s}_y\}$, 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 varies.

Our objective is to track a face in a sequence of images acquired from a moving camera. To achieve robustness to large variations in the object pose, illumination, motion, etc. we use the first-order auto-regressive dynamic model $\mathbf{x}_t = A\mathbf{x}_{t-1} + w_t$, where A is a deterministic component describing a constant velocity movement and w_t denotes a multivariate Gaussian random variable.

3.2 Shape and Color Cues

As demonstrated in [1][3], 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 quite well approximated by an ellipse. Therefore a parametric model of the ellipse with a fixed aspect ratio equal to 1.2 is utilized to verify the oval shape of head candidates. During tracking the oval shape of each head candidate is verified using the sum of intensity gradients along the ellipse's boundary.

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 helpful to support detection of faces in image sequences because color as a cue is computationally inexpensive [14], 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 partially occluded because edge-based methods can be ineffective.

A color histogram including spatial information can be extracted on the basis of a 2-dimensional kernel centered on the target [4]. The kernel weights the color of the pixel according to its distance from the kernel center. In order to assign smaller weights to the color of pixels that are further away from the center of the kernel a nonnegative and monotonic decreasing function $k : [0, \infty) \rightarrow R$ can be utilized [4]. The probability of particular histogram bin u at location $\mathbf{x} = \{x, y\}$ is determined by the following formula:

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

where $\mathbf{x}_j$ are pixel locations, L is the number of pixels in the considered kernel, 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 [4] 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 extracted 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 V component obtained the 4-bin representation while the remaining components of the HSV color space have been represented by 8 bins [9].

To compare the histogram Q representing the tracked face to a histogram I obtained from the particle configuration 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 [4] demonstrated that the utilized metric is invariant to the scale of the target and therefore is superior to other measures such as histogram intersection [14] or Kullback divergence. Using the Bhattacharyya coefficient we defined the color observation model as $p(\mathbf{z}^C \mid \mathbf{x}) = (\sqrt{2\pi}\sigma)^{-1} e^{-\frac{1-\rho}{2\sigma^2}}$. Thanks 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.

3.3 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 [15]. The reliability factors of the 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 $\eta_t^{(i)} = 0.5 * (\tanh(-aD_t^{(j)}) + b)$. In the experiments we set $a = 0.3$ and $b = 3$. Using the reliability factors the observation likelihood has been determined as follows:

$$p(\mathbf{z}_t \,|\, \mathbf{x}_t) = [p(\mathbf{z}_t^G \,|\, \mathbf{x}_t)]^{\alpha_t^{(1)}} \cdot [p(\mathbf{z}_t^C \,|\, \mathbf{x}_t)]^{\alpha_t^{(2)}} \quad 0 \leq \alpha_t^{(j)} \leq 1 \qquad (2)$$

3.4 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, a 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 calculated from the estimated state, $Q_{t-1}^{(u)}$ is the histogram of the target from the previous frame, whereas $u = 1...K$.

3.5 Depth Cue

The length of the minor axis of the considered ellipse has been determined on the basis of depth information. 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. Thanks 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.

3.6 Supporting the tracking through 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 [17]. Using a training set of positive and negative images the Real AdaBoost [13] has been utilized both to select features and to train a robust classifier. A 18 layer cascaded classifier has been trained on images of size 20x20 pixels to detect frontal faces in gray images. The detector has been trained on 1500 frontal faces. All training images were manually aligned by eyes position. 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:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_t) = \beta p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_t) + (1-\beta) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \tag{3}$$

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 [7].

4 Experiments

4.1 The system

The experiments described in this section were carried out with a mobile robot Pioneer 2DX [11] equipped with commercial binocular MegaPixel Stereo Head. The dense stereo maps are extracted in that system thanks to small area correspondences between image pairs [8] 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 a depth subimage in Fig. 1. a). Thanks 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 prepared in C/C++ and operating at images of size 320x240. 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 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 [11].

4.2 Experiments on Real-World Situations

To test the prepared software we performed various experiments with the moving camera. After detection of possible faces, see Fig. 1. a), the system can identify known faces among the detected ones. In tracking scenarios the user moved about a laboratory, 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),f). Our experimental findings show that thanks to stereovision the ellipse of proper size 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, see Fig. 1. b),c), and varying illumination conditions, see Fig. 1. e),f), even when person moves in front of skin-like colors of window-panes, see also Fig. 1. e). During a typical experiment with person following the user typically rounds the laboratory in 70 s and goes a distance about 35 m.

The tracker runs with 400 particles at frame rates of 12-13 Hz. The face detector can localize faces in images of size 320x240 in about 0.1 s. The full cascade consist of 820 weak classifiers. The first five stages of the cascade consists of 80 classifiers and the first ten stages is comprised of 250 classifiers. The recognition of single face takes about 0.01 s. These times allow the system to process about 6 frames per second when the information about detected faces is used to generate the proposal distribution for the particle filter.

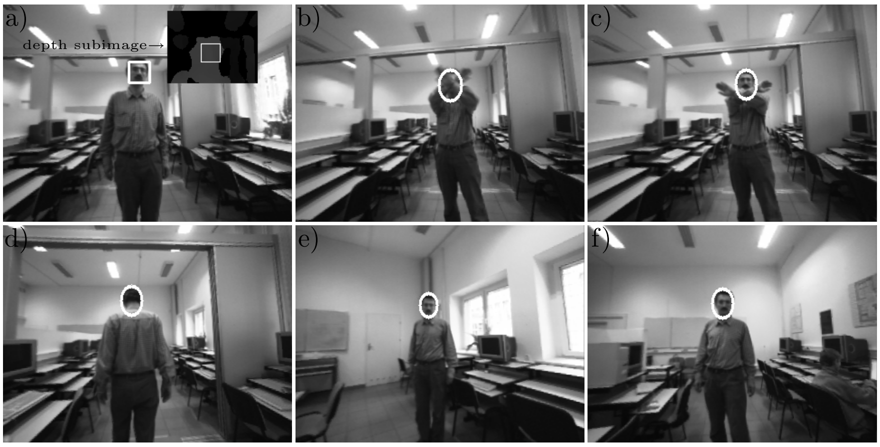

Fig. 1. Face detection and tracking, frames #9, #110, #111, #168, #317, #970

5 Conclusions

We have presented a vision module that robustly tracks and detects a human face. By employing shape, color, stereovision as well as elliptical shape features the proposed method can track a head in case of dynamic background. These features make it general enough to be useful for many human-machine as well as surveillance applications. Experimental results on tracking faces in long indoor video sequences demonstrate the robustness of the tracking system.

References

1. Birchfield S. (1998) Elliptical head tracking using intensity gradients and color histograms, The IEEE Conf. on Comp. Vision and Patt. Rec., 232–237
2. Bradski G. R. (1998) Computer vision face tracking as a component of a perceptual user interface, In Workshop on Applications of Computer Vision, 214–219
3. Chen Y., Rui Y., Huang T. (2002) Mode–based multi–hypothesis head tracking using parametric contours, In Proc. IEEE Int. Conf. on Aut. Face Rec., 112–117
4. Comaniciu D., Ramesh V., Meer P. (2000) Real–time tracking of non–rigid objects using Mean Shift, The IEEE Conf. on Comp. Vision and Patt. Rec., 142–149
5. Darrell T., Gordon G., Harville M., Woodfill J. (1998) Integrated person tracking using stereo, color, and pattern detection, Proc. of the Conf. on Comp. Vision and Patt. Rec., 601–609
6. Doucet A., Godsill S., Andrieu Ch. (2000) On sequential Monte Carlo sampling methods for bayesian filtering, Statistics and Computing, 10:197–208
7. Isard M., Blake A. (1998) CONDENSATION - conditional density propagation for visual tracking, Int. J. of Computer Vision, 29:5–28
8. Konolige K. (1997) Small Vision System: Hardware and implementation, Proc. of Int. Symp. on Robotics Research, 111–116
9. Kwolek B. (2004) Stereovision–based head tracking using color and ellipse fitting in a particle filter, 8th European Conf. on Computer Vision, 192–204
10. Perez P., Hue C., Vermaak J., Gangnet M. (2002) Color–based probabilistic tracking, European Conf. on Computer Vision, 661–675
11. Pioneer 2 mobile robots (2001) ActivMedia Robotics
12. Rowley H., Baluja S., Kanade T. (1996) Neural network–based face detection, Proc. of IEEE Conf. on Comp. Vision and Patt. Rec., 203–207
13. Schapire R., Singer Y. (1998) Improved boosting algorithms using confidence-rated predictions, Proc. 11th Ann. Conf. Computational Learning Theory, 80-91
14. Swain M. J., Ballard D. H. (1991) Color indexing, Int. J. of Computer Vision, 7:11–32.
15. Triesch J., Malsburg Ch. (2001) Democratic integration: Self–organized integration of adaptive cues, Neural Computation, 13:2049–2074
16. Turk M. A., Pentland A. P. (1991) Face recognition using eigenfaces, In Proc. of Conf. on Comp. Vision and Patt. Rec., 586–591
17. Viola P., Jones M. (2001) Rapid object detection using a boosted cascade of simple features, The IEEE Conf. on Comp. Vision and Patt. Rec., 511–518
18. Yang M. H., Kriegman D., Ahuja N. (2002) Detecting faces in images: A survey, IEEE Trans. on Pattern Analysis and Machine Intelligence, 24:34–58