Propagation of Pixel Hypotheses for Multiple Objects Tracking

Haris Baltzakis and Antonis A. Argyros

Institute of Computer Science, Forth
{xmpalt,argyros}@ics.forth.gr
http://www.ics.forth.gr/cvrl/

Abstract. In this paper we propose a new approach for tracking multiple objects in image sequences. The proposed approach differs from existing ones in important aspects of the representation of the location and the shape of tracked objects and of the uncertainty associated with them. The location and the speed of each object is modeled as a discrete time, linear dynamical system which is tracked using Kalman filtering. Information about the spatial distribution of the pixels of each tracked object is passed on from frame to frame by propagating a set of pixel hypotheses, uniformly sampled from the original object's projection to the target frame using the object's current dynamics, as estimated by the Kalman filter. The density of the propagated pixel hypotheses provides a novel metric that is used to associate image pixels with existing object tracks by taking into account both the shape of each object and the uncertainty associated with its track. The proposed tracking approach has been developed to support face and hand tracking for human-robot interaction. Nevertheless, it is readily applicable to a much broader class of multiple objects tracking problems.

1 Introduction

This paper presents a novel approach for multiple object tracking in image sequences, intended to track skin-colored blobs that correspond to human hands and faces. Vision-based tracking of human hands and faces constitutes an important component in gesture recognition systems with many potential applications in the field of human-computer and/or human-robot interaction.

Some successful approaches for hand and face tracking utilize ellipses to model the shape of the objects on the image plane [1–5]. Typically, simple temporal filters such as linear, constant-velocity predictors are used to predict/propagate the locations of these ellipses from frame to frame. Matching of predicted ellipses with the extracted blobs is done either by correlation techniques or by using statistical properties of the tracked objects.

In contrast to blob tracking approaches, model based ones [6–11] do not track objects on the image plane but, rather, on a hidden model-space. This is commonly facilitated by means of sequential Bayesian filters such as Kalman or

G. Bebis et al. (Eds.): ISVC 2009, Part II, LNCS 5876, pp. 140–149, 2009.

particle filters. The state of each object is assumed to be an unobserved Markov process which evolves according to specific dynamics and which generates measurement predictions that can be evaluated by comparing them with the actual image measurements.

Model based approaches are commonly assumed to be more suitable to track complex and/or deformable objects whose image projections cannot be modeled with simple shapes. Human hands, especially when observed from a short distance, fall in this category. Despite the fact that standard Bayesian filtering does not explicitly handle observation-to-track assignments, the sophisticated temporal filtering which is inherent to model based approaches allows them to produce better data association solutions. This is particularly important for multiple objects tracking, where it is common for tracked objects to become temporarily occluded by other tracked or non-tracked objects.

Among model-based approaches, particle filtering [12] has been successfully applied to object tracking, both with edge-based [12] and kinematic [7, 8] imaging models. With respect to the data association problem, particle filtering offers a significant advantage over other filtering methods because it allows for different, locally-optimal data association solutions for each particle which are implicitly evaluated through each particle's likelihood. However, as with any other model-based approach, particle filters rely on accurate modeling, which in most cases leads to an increased number of unknown parameters. Since the number of required particles for effective tracking is exponential to the number of tracked parameters, particle filter based tracking is applicable only to problems where the observations can be explained with relatively simple models.

In this paper we propose a blob-tracking approach that differs significantly from existing approaches in (a) the way that the position and shape uncertainty are represented and (b) the way that data association is performed. More specifically, information about the location and shape of each tracked object is maintained by means of a set of pixel hypotheses that are propagated from frame to frame according to linear object dynamics computed by a Kalman filter. Unlike particle filters which correspond to object pose hypotheses in the model space, the proposed propagated pixel hypotheses correspond to single pixel hypotheses in the observation space. Another significant difference is that, in our approach, the distribution of the propagated pixel hypotheses provides a representation for the uncertainty in both the position and the shape of the tracked object. Moreover, as it will be shown in the following sections, the local density of pixel hypotheses provides a meaningful metric to associate observed skin-colored pixels with existing object tracks, enabling an intuitive, pixel-based data association approach based on the joint-probabilistic paradigm.

The proposed approach has been tested in the context of a human-robot interaction application involving detection and tracking of human faces and hands. Experimental results demonstrate that the proposed approach manages to successfully track multiple interacting deformable objects, without requiring complex models for the tracked objects or their motion.

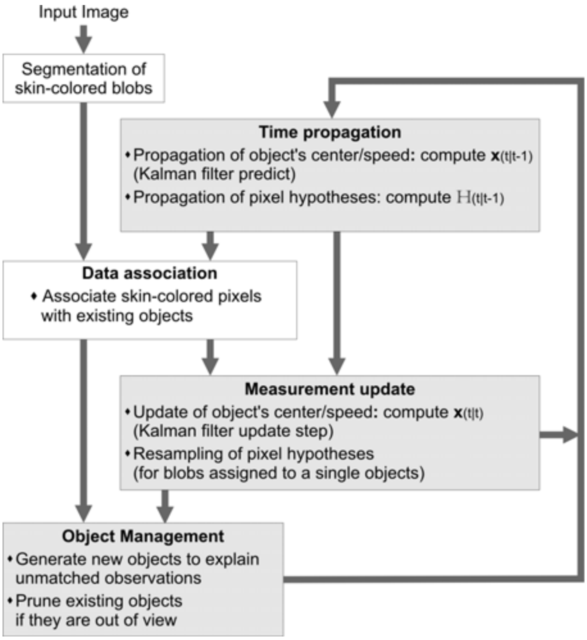

Input Image
Segmentation of skin-colored blobs
Time propagation
• Propagation of object's center/speed: compute x(t|t-1) (Kalman filter predict)
• Propagation of pixel hypotheses: compute H(t|t-1)
Data association
• Associate skin-colored pixels with existing objects
Measurement update
• Update of object's center/speed: compute x(t|t) (Kalman filter update step)
• Resampling of pixel hypotheses (for blobs assigned to a single objects)
Object Management
• Generate new objects to explain unmatched observations
• Prune existing objects if they are out of view

Fig. 1. Block diagram of the proposed approach

2 Problem Description and Methodology

A tracking algorithm must be able to maintain the correct labeling of the tracked objects, even in cases of partial or full occlusions. Typically, this requirement calls for sophisticated modeling of the objects' motion, shapes and dynamics (i.e. how the shape changes over time). In this paper we present a blob tracker that handles occlusions, shape deformations and similarities in color appearance without making explicit assumptions about the motion or the shape of the tracked objects. The proposed tracker uses a simple linear model for object trajectories and the uncertainty associated with them. Moreover, it does not rely on an explicit model for the shape of the tracked object. Instead, the shapes of the tracked objects and the associated uncertainty is represented by a set of pixel hypotheses that are propagated over time using the same linear dynamics as the ones used to model the object's trajectory.

An overview of the proposed approach is illustrated in Fig. 1. The first step in the proposed approach is to identify pixels that are likely to belong to tracked objects. In the context of the application under consideration, we are interested in tracking human hands and faces. Thus, the tracker implemented in this paper tracks skin-colored blobs[1]. To identify pixels belonging to such objects we em-

[1] The proposed tracking method can also be used to track blobs depending on properties other than skin color.

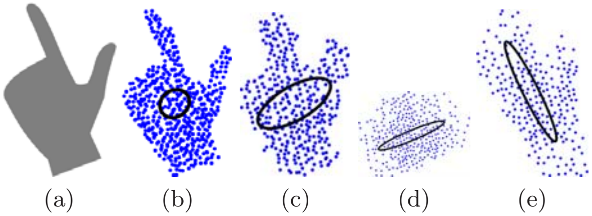


Fig. 2. Object's state representation. (a) Observed blob (b)-(e) Examples of possible states. Ellipses represent iso-probability contours for the location of the object (i.e. the first two components of $\mathbf{x_t}$). Dots represent the pixel hypotheses.

ploy a Bayesian approach that takes into account their color as well as whether they belong to the foreground or not. Image pixels with high probability to belong to hand and face regions are then grouped into connected blobs using hysteresis thresholding and connected components labeling, as in [3]. Blobs are then assigned to objects which are tracked over time. More specifically, for each tracked object two following types of information is maintained:

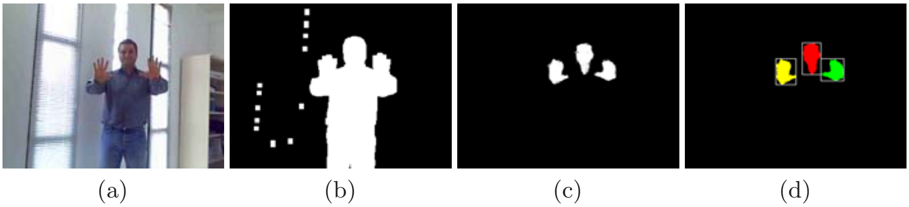
- The location and the speed of the object's centroid, in image coordinates. This is encoded by means of a 4D vector $\mathbf{x}(t) = [c_x(t), c_y(t), u_x(t), u_y(t)]^{\mathrm{T}}$, where $c_x(t)$ and $c_y(t)$ are the image coordinates of the object's centroid at time t and $u_x(t)$ and $u_y(t)$ are the horizontal and vertical components of its speed. A Kalman filter is used to maintain a Gaussian estimate $\hat{\mathbf{x}}(t)$ of the above-described state vector and its associated 4×4 covariance matrix $\mathbf{P}(t)$.
- The spatial distribution of the object's pixels. This is encoded by means of a set $\mathbb{H} = \{(x_i, y_i) : i = 1 \ldots N\}$ of N pixel hypotheses that are sampled uniformly from the object's blob and propagated from frame to frame using the dynamics estimated by the Kalman filter.

The representation described above is further explained in Fig. 2. Figure 2(a) depicts the blob of a hypothetical object (a human hand in this example). Figures 2(b)-(e) depict four possible states of the proposed tracker.

The distribution of the propagated pixel hypotheses provides the metric used to associate measured evidence to existing object tracks. During the data association step, observed blob pixels are individually processed one-by-one in order to associate them with existing object tracks.

After skin-colored pixels have been associated with existing object tracks, the update phase follows in two steps: (a) the state-vector (centroid's location and speed) is updated using the Kalman filter's measurement-update equations and (b) pixel hypotheses are updated by resampling them from their associated blob pixels. The resampling step is important to avoid degenerate situations and to allow the object hypotheses to closely follow the blobs shape and size.

Finally, track management techniques are employed to ensure that new objects are generated for blobs with pixels that are not assigned to any of the existing tracks and that objects which are not supported by observation are eventually removed from further consideration.



(a) (b) (c) (d)

Fig. 3. Blob detection. (a) Initial image, (b) foreground pixels, (c) skin-colored pixels, (d) resulting skin-colored blobs.

3 The Proposed Tracking Method

In this section we provide a detailed description of the proposed multiple objects tracking method.

3.1 Segmentation of Skin-Colored Foreground Blobs

The first step of the proposed approach is to detect skin-colored regions in the input images. For this purpose, a technique similar to [3, 13] is employed. Initially, background subtraction [14] is used to extract the foreground areas of the image. Then, for each pixel, $P(s|c)$ is computed, which is the probability that this pixel belongs to a skin-colored foreground region s, given its color c. This can be computed according to the Bayes rule as $P(s|c) = P(s)P(c|s)/P(c)$, where $P(s)$ and $P(c)$ are the prior probabilities of foreground skin pixels and foreground pixels having color c, respectively. Color c is assumed to be a 2D variable encoding the U and V components of the YUV color space. $P(c|s)$ is the prior probability of observing color c in skin colored foreground regions. All three components in the right side of the above equation can be computed based on offline training.

After probabilities have been assigned to each image pixel, hysteresis thresholding is used to extract solid skin color blobs and create a binary mask of foreground skin-colored pixels. A connected components labeling algorithm is then used to assign different labels to pixels that belong to different blobs. Size filtering on the derived connected components is also performed to eliminate small, isolated blobs that are attributed to noise.

Results of the intermediate steps of this process are illustrated in Fig. 3. Figure 3(a) shows a single frame extracted out of a video sequence that shows a man performing various hand gestures in an office-like environment. Fig. 3(b) shows the result of the background subtraction algorithm and Fig. 3(c) shows skin-colored pixels after hysteresis thresholding. Finally, the resulting blobs (i.e. the result of the labeling algorithm) are shown in Fig. 3(d).

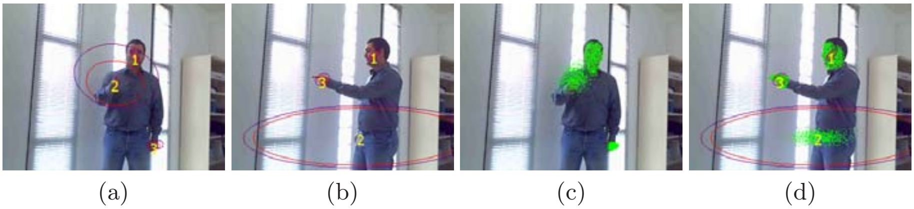

(a) (b) (c) (d)

Fig. 4. Tracking hypotheses over time. (a), (b) uncertainty ellipses corresponding to predicted hypotheses locations and speed, (c), (d) propagated pixel hypotheses.

3.2 Tracking Blob Position and Speed

The dynamics of each tracked object are modeled by means of a linear dynamical system which is tracked using the Kalman filter [15, 16]. The state vector $\mathbf{x}(t)$ at time t is given as $\mathbf{x}(t) = (c_x(t), c_y(t), u_x(t), u_y(t))^{\mathrm{T}}$ where $c_x(t)$, $c_y(t)$ are the horizontal and vertical coordinates of the tracked object's centroid, and $u_x(t)$, $u_y(t)$ are the corresponding components of the tracked object's speed.

The Kalman-filter described above is illustrated in Figures 4(a) and 4(b) which show frames extracted from the same sequence as the one in Fig. 3. The depicted ellipses correspond to 95% iso-probability contours for the predicted location (smaller, red-colored ellipses) and speed (larger, purple-colored ellipses) of each tracked object's centroid. As can be verified, objects that move rapidly (e.g., object 2 in Fig. 4(a)) or objects that are not visible (e.g., object 2 in Fig. 4(b)) have larger uncertainty ellipses. On the other hand, objects that move slowly (e.g., faces) can be predicted with more certainty.

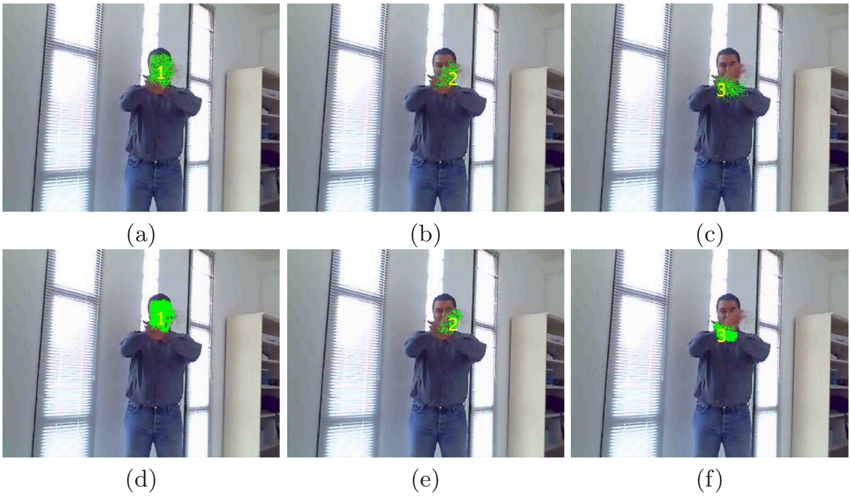
3.3 Pixel Hypotheses Propagation

Pixel hypotheses are propagated using the predicted state estimate $\hat{\mathbf{x}}(t|t-1)$ and the predicted error covariance $\mathbf{P}(t|t-1)$ of the Kalman filter discussed in the previous section. More specifically, each pixel hypothesis (x_i, y_i) in $\mathbb{H} = \{(x_i, y_i) : i = 1 \ldots N\}$ is propagated in time by drawing a new sample from

$$N\left(\begin{bmatrix} x_i + \hat{u}_x(t|t-1) \\ y_i + \hat{u}_y(t|t-1) \end{bmatrix}, \mathbf{P_h}(t|t-1)\right) \tag{1}$$

where $\hat{u}_x(t|t-1)$ and $\hat{u}_y(t|t-1)$ are the predicted velocity components (i.e. third and forth element of $\hat{\mathbf{x}}(t|t-1)$ and $\mathbf{P_h}(t|t-1)$ is the top left 2×2 submatrix of $\mathbf{P}(t|t-1)$.

Figures 4(c) and 4(d) depict the predicted pixel locations (i.e. pixel hypotheses) that correspond to the object tracks shown in Figs. 4(a) and 4(b), respectively. As can be verified, tracks with larger uncertainty ellipses correspond to less concentrated pixel hypotheses. On the other hand, propagated pixel hypotheses tend to have higher spatial density for object tracks that are predictable with higher confidence.



(a) (b) (c)

(d) (e) (f)

Fig. 5. Three objects merged into a single blob. Predicted pixel locations for each of the three objects (1st row), pixels finally assigned to each object (2nd row).

3.4 Associating Pixels with Objects

The purpose of the data association step is to associate observations with existing object tracks. In this paper, data association is performed on a pixel basis rather than a blob basis; i.e. each observed skin-colored pixel is individually associated to existing tracks. This permits pixels that belong to the same blob to be associated with different object tracks.

The metric used to provide the degree of association between a specific skin-colored pixel with image coordinates (x, y) and a specific object track o_i is assumed to be equal to the local density of the propagated pixels hypotheses of this track at the location of this specific pixel. More specifically, to estimate the degree of association $A(\mathbf{p}, o_i)$ between pixel $\mathbf{p}$ and track o_i, we make use of the following metric:

$$A(\mathbf{p}, o_i) = \alpha_i \frac{C^P_{N(\mathbf{p})}}{C_{N(\mathbf{p})}}, \tag{2}$$

where $N(\mathbf{p}) = \{\mathbf{p_k}, \|\mathbf{p} - \mathbf{p_k}\| \leq D\}$ is a neighborhood of pixel $\mathbf{p}$, $C^P_{N(\mathbf{p})}$ is the number of propagated pixel hypotheses of object track o_i within $N(\mathbf{p})$ and $C_{N(\mathbf{p})}$ is the total number of pixels in $N(\mathbf{p})$. α_i is a normalizing factor ensuring that the sum of all data association weights of (2) remains constant for each track over time. An 8-neighborhood ($D = \sqrt{2}$) has proven sufficient in all experiments.

After pixels have been associated with tracked objects, weighted means (according to $A(\mathbf{p}, o_i)$) are computed for each tracked object and used for the Kalman filter update phase. Pixel hypotheses are also resampled from the weighted distribution of the observed pixels. The above-described data association scheme follows the joint-probabilistic paradigm by combining all potential association candidates in a single, statistically most plausible, update.

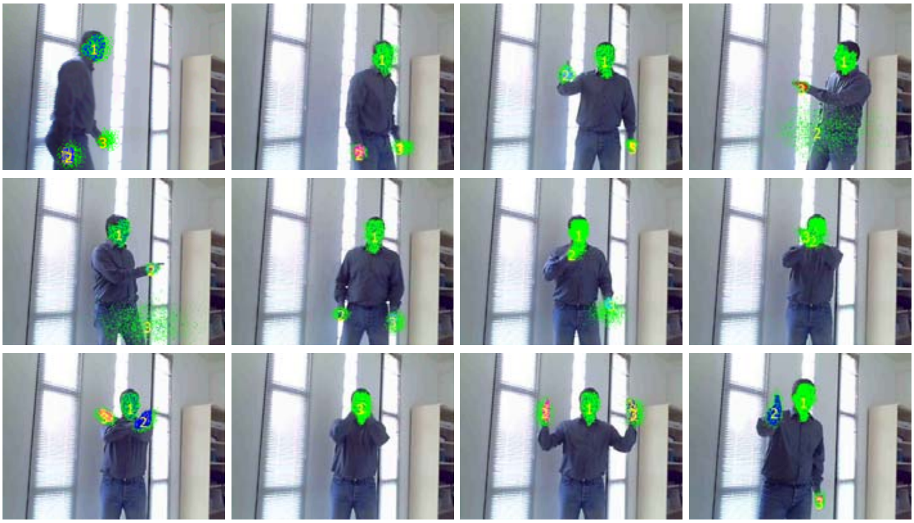


Fig. 6. Tracking results for twelve segments of the office image sequence used in the previous examples. In all cases the algorithm succeeds in tracking the three hypotheses.

A notable case that is often encountered in practice, is when all pixels of a single blob are assigned to a single track and vice versa (i.e. no propagated pixel hypotheses are associated with pixels of other blobs). In this case, resampling of pixel hypotheses is performed by uniformly sampling blob pixels. This permits pixel hypotheses to periodically re-initialize themselves and exactly-follow the blob position and shape when no data association ambiguities exist.

Figure 5 demonstrates how the proposed tracking algorithm behaves in a case where three objects simultaneously occlude each other, leading to difficult data association problems. The top row depicts the predicted pixel locations for each of the three valid tracks. The bottom row depicts the final assignment of blob pixels to tracks, according to the density of the predicted pixel hypotheses.

4 Experimental Results

Figure 6 depicts the tracker's output for a number of frames of the image sequence comprising the running example used in Figs 3, 4 and 5. As can be observed, the tracker succeeds in keeping track of all the three hypotheses despite the occlusions introduced at various fragments of the sequence.

The proposed tracker comprises an important building block of a vision-based, hand- and face-gesture recognition system which is installed on a mobile robot. The purpose of the system is to facilitate natural human-robot interaction while guiding visitors in large public spaces such as museums and exhibitions. The performance of the system has been evaluated for a three-weeks time in a large public place. Figure 7 depicts snapshots of three different image sequences captured at the installation site. Despite the fact that the operational requirements

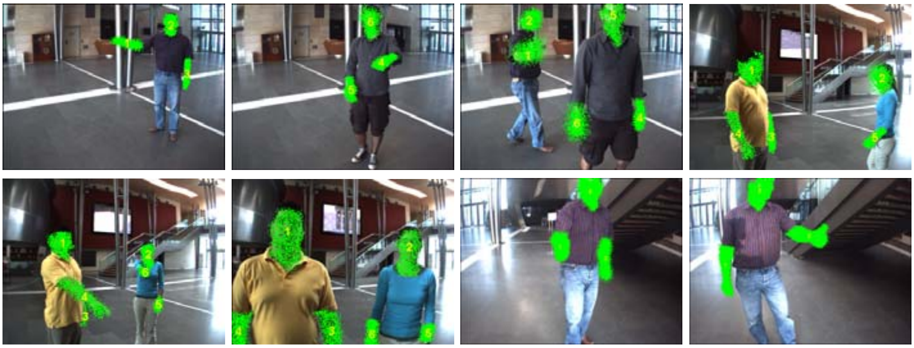


Fig. 7. Tracking results from a real-world application setup

of the task at hand (i.e. unconstrained lighting conditions, unconstrained hand and face motion, varying and cluttered background, limited computational resources) were particularly challenging, the tracker operated for a three weeks time with results that, in most cases, were proved sufficiently accurate to provide input to the hand- and face-gesture recognition system of the robot. During these experiments the algorithm ran on a standard laptop computer, operating at 640 × 480 images. At this resolution, the algorithm achieved a frame rate of 30 frames per second. Several video sequences obtained at the actual application site are available on the web[2].

5 Conclusions and Future Work

In this paper we have presented a novel approach for tracking multiple objects. The proposed approach differs from existing approaches in the way used to associate perceived blob pixels with existing object tracks. For this purpose, information about the spatial distribution of blob pixels is passed on from frame to frame by propagating a set of pixel hypotheses, uniformly sampled from the original blob, to the target frame using the object's current dynamics, as estimated by means of a Kalman filter. The proposed approach has been tested in the context of face and hand tracking for human-robot interaction. Experimental results show that the method is capable of tracking several deformable objects that may move in complex, overlapping trajectories.

Acknowledgments

This work was partially supported by the EU-IST project INDIGO (FP6-045388) and the EU-IST project GRASP (FP7-IP-215821).

[2] http://www.ics.forth.gr/~xmpalt/research/handfacetrack_pixelhyps/index.html

References

1. Birk, H., Moeslund, T., Madsen, C.: Real-time recognition of hand alphabet gestures using principal component analysis. In: Proc. Scandinavian Conference on Image Analysis, Lappeenranta, Finland (1997)
2. Wren, C.R., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: Real-time tracking of the human body. IEEE Trans. Pattern Analysis and Machine Intelligence 19, 780–785 (1997)
3. Argyros, A.A., Lourakis, M.I.A.: Real-time tracking of multiple skin-colored objects with a possibly moving camera. In: Proc. European Conference on Computer Vision, Prague, Chech Republic, pp. 368–379 (2004)
4. Argyros, A.A., Lourakis, M.I.A.: Vision-based interpretation of hand gestures for remote control of a computer mouse. In: ECCV Workshop on HCI, Graz, Austria, pp. 40–51 (2006)
5. Usabiaga, J., Erol, A., Bebis, G., Boyle, R., Twombly, X.: Global hand pose estimation by multiple camera ellipse tracking. Machine Vision and Applications 19 (2008)
6. Rehg, J., Kanade, T.: Digiteyes: Vision-based hand tracking for human-computer interaction. In: Workshop on Motion of Non-Rigid and Articulated Bodies, Austin Texas, pp. 16–24 (1994)
7. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: IEEE Conference on Computer Vision and Pattern Recognition 2000, Proceedings, vol. 2, pp. 126–133 (2000)
8. Sidenbladh, H., Black, M.J., Fleet, D.J.: Stochastic tracking of 3d human figures using 2d image motion. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 702–718. Springer, Heidelberg (2000)
9. Stenger, B., Mendonca, P.R.S., Cipolla, R.: Model-based hand tracking using an unscented kalman filter. In: Proc. British Machine Vision Conference (BMVC), vol. 1, pp. 63–72 (2001)
10. Shamaie, A., Sutherland, A.: Hand tracking in bimanual movements. Image and Vision Computing 23, 1131–1149 (2005)
11. Stenger, B., Thayananthan, A., Torr, P.H.S., Cipolla, R.: Model-based hand tracking using a hierarchical bayesian filter. IEEE Trans. Pattern Analysis and Machine Intelligence 28, 1372–1384 (2006)
12. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. Int. Journal of Computer Vision 29, 5–28 (1998)
13. Baltzakis, H., Argyros, A., Lourakis, M., Trahanias, P.: Tracking of human hands and faces through probabilistic fusion of multiple visual cues. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 33–42. Springer, Heidelberg (2008)
14. Grimson, W.E.L., Stauffer, C.: Adaptive background mixture models for real time tracking. In: Proc. IEEE Computer Vision and Pattern Recognition (CVPR), Ft. Collins, USA, pp. 246–252 (1999)
15. Kalman, R.E.: A new approach to linear flitering and prediction problems. Transactions of the ASME–Journal of Basic Engineering 82, 35–42 (1960)
16. Bar-Shalom, Y., Li, X.: Estimation and Tracking: Principles, Techniques, and Software. Artech House Inc., Boston (1993)