第 29 卷 第 3 期
2003 年 5 月

自 动 化 学 报
ACTA AUTOMATICA SINICA

Vol. 29, No. 3
May, 2003

Color Features for Tracking Non-Rigid Objects[1)]

Katja Nummiaro[1] Esther Koller-Meier[2] Luc Van Gool[1,2]

[1] (*Katholieke Universiteit Leuven, ESAT/PSI-VISICS, Kasteelpark Arenberg* 10, 3001 *Heverlee, Belgium*)

[2] (*Swiss Federal Institute of Technology* (*ETH*), *D-ITET/BIWI, Gloriastrasse* 35, 8092 *Zurich, Switzerland*)

(E-mail: Katja. Nummiaro@esat. kuleuven. ac. be)

Abstract Robust real-time tracking of non-rigid objects is a challenging task. Color distributions provide an efficient feature for this kind of tracking problems as they are robust to partial occlusion, are rotation and scale invariant and computationally efficient. This article presents the integration of color distributions into particle filtering, which has typically been used in combination with edge-based image features. Particle filters offer a probabilistic framework for dynamic state estimation and have proven to work well in cases of clutter and occlusion. To overcome the problem of appearance changes, an adaptive model update is introduced during temporally stable image observations. Furthermore, an initialization strategy is discussed since tracked objects may disappear and reappear.

Key words Particle filtering, color distribution, Bhattacharyya coefficient

1 Introduction

With the recent improvements in computer technology, real-time automated visual surveillance has become a popular research area[1~3]. Nowadays surveillance cameras are installed in many security-sensitive areas such as railway stations, parking blocks, airports, banks or public building lobbies to improve the safety.

In this context, we focus on object tracking using color distributions in conjunction with a particle filter. First of all, color histograms have many advantages for tracking non-rigid objects as they are robust to partial occlusion, are rotation and scale invariant and are calculated efficiently. Particle filtering[4~7] on the other hand has been proven very successful for non-linear and non-Gaussian estimation problems and is reliable in cases of clutter and during occlusions. The novelty of the proposed tracker lies in the original mixture of efficient components that together yield a reliable and fast system for tracking non-rigid objects.

In general, tracking methods can be divided into two main classes specified as bottom-up or top-down approaches. In a bottom-up approach the image is segmented into objects which are then used for the tracking. For example blob detection[8] can be used for the object extraction. In contrast, a top-down approach generates object hypotheses and tries to verify them using the image. Typically, model-based[5,6] and template matching approaches[9] belong to this class. The proposed color-based particle filter follows the top-down approaches, in the sense that the image content is only evaluated at the hypothetical positions.

The related mean shift tracker by Comaniciu *et al.*[9] which is a nonparametric density gradient estimator, also uses color distributions. By employing multiple hypotheses and a model of the system dynamics our proposed method can track objects more reliably in cases of clutter and occlusions. Jepson *et al.*, McKenna *et al.* and Raja *et al.*[10~12] have already discussed adaptive models, but these approaches employ Gaussian mixture models while

1) Support by the European IST project STAR (IST 2000-28764) and by the GOA/VHS+project financed by the Research Fund of Katholieke Universiteit Leuven, Belgium
Received November 28, 2002; in revised form April 9, 2003
收稿日期 2002-11-28; 收修改稿日期 2003-04-09

we use color histograms together with multiple hypotheses. Isard *et al.*[13] have already employed color information in particle filtering by using Gaussian mixtures. In comparison, our target model has the advantage of matching only objects that have a similar histogram, whereas for Gaussian mixtures objects that contain one of the colors of the mixture will already match. Recently, Pérez *et al.*[14] introduced an approach that also uses color histograms and a particle filtering framework for multiple object tracking. The two independently proposed methods[14,15] differ in the initialization of the tracker, the model update, the region shape and the observation of the tracking performance. A detailed comparison of the color-based particle filter to the mean shift tracker and a combination between the mean shift tracker and Kalman filtering[16] is described in [17].

The remainder of this article is organized as follows. In Section 2 we briefly describe particle filtering and in Section 3 we indicate how color distributions are used as object models. The integration of the color information into the particle filter is explained in Section 4 and Section 5 describes the model update. As tracked objects may disappear and reappear an initialization based on an appearance condition is introduced in Section 6. In Section 7 we present some experimental results and finally, in Section 8, we summarize our conclusions.

2 Particle filtering

Particle filters[5,6] offer a probabilistic framework for dynamic state estimation. They approximate the posterior density of the current object state X_t conditioned on all observations $\{\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_t\}$ up to time t by a weighted sample set $S = \{(\boldsymbol{s}^{(n)}, \pi^{(n)}) \mid n=1,2,\cdots,N\}$. Each sample $\boldsymbol{s}$ represents one hypothetical state of the object, with a corresponding discrete sampling probability π, where $\sum_{n=1}^{N} \pi^{(n)} = 1$.

The evolution of the sample set is described by propagating each sample according to a system model. Each element of the set is weighted in terms of the observations and N samples are drawn with replacement, by choosing a particular sample with probability $\pi^{(n)} = p(\boldsymbol{z}_t \mid \boldsymbol{X}_t = \boldsymbol{s}_t^{(n)})$. The mean state of an object is estimated at each time step by

$$E[S] = \sum_{n=1}^{N} \pi^{(n)} \boldsymbol{s}^{(n)} \tag{1}$$

Particle filtering provides a robust tracking framework in case of clutter and occlusion, as it models uncertainty. It can keep its options open and consider multiple state hypotheses simultaneously.

3 Color distribution model

We want to apply a particle filter in a color-based context. Color distributions are used as target models as they achieve robustness against non-rigidity, rotation and partial occlusion. Suppose that the distributions are discretized into m-bins. The histograms are produced with the function $h(\boldsymbol{x}_i)$, that assigns the color at location $\boldsymbol{x}_i$ to the corresponding bin. In our experiments, the histograms are typically calculated in the RGB space using $8 \times 8 \times 8$ bins. To make the algorithm less sensitive to lighting conditions, the HSV color space could be used instead with less sensitivity to V (e.g. $8 \times 8 \times 4$ bins).

We determine the color distribution inside an upright elliptic region with half axes H_x and H_y. To increase the reliability of the color distribution when boundary pixels belong to the background or get occluded, smaller weights are assigned to the pixels that are further away from the region center by employing a weighting function

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

where r is the distance from the region center. Thus, we increase the reliability of the color distribution when these boundary pixels belong to the background or get occluded. It is also possible to use a different weighting function, for example the Epanechnikov kernel[9].

The color distribution $p_y = \{p_y^{(u)}\}_{u=1,2,\cdots,m}$ at location $\mathbf{y}$ is calculated as

$$p_y^{(u)} = f \sum_{i=1}^{I} k\left(\frac{\| \mathbf{y} - \mathbf{x}_i \|}{a}\right)\delta[h(\mathbf{x}_i) - u] \tag{3}$$

where I is the number of pixels in the region, δ is the Kronecker delta function, the parameter $a = \sqrt{H_x^2 + H_y^2}$ is used to adapt the size of the region, and the normalization factor

$$f = \frac{1}{\sum_{i=1}^{I} k\left(\frac{\| \mathbf{y} - \mathbf{x}_i \|}{a}\right)} \tag{4}$$

ensures that $\sum_{u=1}^{m} p_y^{(u)} = 1$.

In a tracking approach, the estimated state is updated at each time step by incorporating the new observations. Therefore, we need a similarity measure which is based on color distributions. A popular measure between two distributions $p(u)$ and $q(u)$ is the Bhattacharyya coefficient[18,19]

$$\rho[p,q] = \int \sqrt{p(u)q(u)}\,\mathrm{d}u \tag{5}$$

Considering discrete densities such as our color histograms $p = \{p^{(u)}\}_{u=1,2,\cdots,m}$ and $q = \{q^{(u)}\}_{u=1,2,\cdots,m}$ the coefficient is defined as

$$\rho[p,q] = \sum_{u=1}^{m} \sqrt{p^{(u)} q^{(u)}} \tag{6}$$

The larger ρ is, the more similar the distributions are. For two identical normalized histograms we obtain $\rho = 1$, indicating a perfect match. As distance between two distributions we define the measure

$$d = \sqrt{1 - \rho[p,q]} \tag{7}$$

which is called the Bhattacharyya distance.

4 Color-based particle filtering

The proposed tracker employs the Bhattacharyya distance to update the a priori distribution calculated by the particle filter. Each sample of the distribution represents an ellipse and is given as

$$s = \{x, y, \dot{x}, \dot{y}, H_x, H_y, \dot{a}\}$$

where x, y specify the location of the ellipse, $\dot{x}, \dot{y}$ the motion, H_x, H_y the length of the half axes and $\dot{a}$ the corresponding scale change. As we consider a whole sample set the tracker handles multiple hypotheses simultaneously.

The sample set is propagated through the application of a dynamic model

$$\mathbf{s}_t = A\mathbf{s}_{t-1} + \mathbf{w}_{t-1}$$

where A defines the deterministic component of the model and $\mathbf{w}_{t-1}$ is a multivariate Gaussian random variable. In our application we currently use a first order model for A describing a region moving with constant velocity $\dot{x}, \dot{y}$ and scale change $\dot{a}$. Expanding this model to second order is straightforward.

To weight the sample set, the Bhattacharyya coefficient has to be computed between the target histogram and the histogram of the hypotheses. Each hypothetical region is specified by its state vector $\mathbf{s}^{(n)}$. Both the target histogram q and the candidate histogram $p_{\mathbf{s}^{(n)}}$ are calculated from Eq. 3 where the target is centered at the origin of the elliptic region.

As we want to favor samples whose color distributions are similar to the target model, small Bhattacharyya distances correspond to large weights:

$$\pi^{(n)} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{d^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(1-\rho[p_{s^{(n)}}, q])}{2\sigma^2}} \tag{10}$$

that are specified by a Gaussian with variance σ. During filtering, samples with a high weight may be chosen several times, leading to identical copies, while others with relatively low weights may not be chosen at all. The programming details for one iteration step are given in Fig. 1.

Given the sample set S_{t-1} and the target model

$$q^{(u)} = f \sum_{i=1}^{I} k\left(\frac{\| x_i \|}{a}\right) \delta[h(x_i) - u],$$

perform the following steps.

1. **Select** N samples from the set S_{t-1} with probability $\pi_{t-1}^{(n)}$

   a) calculate the normalized cumulative probabilities c'_{t-1}

   $$c_{t-1}^{(0)} = 0$$

   $$c_{t-1}^{(n)} = c_{t-1}^{(n-1)} + \pi_{t-1}^{(n)}$$

   $$c_{t-1}^{\prime(n)} = \frac{c_{t-1}^{(n)}}{c_{t-1}^{(N)}}$$

   b) generate a uniformly distributed random number $r \in [0, 1]$

   c) find, by binary search, the smallest j for which $c_{t-1}^{\prime(j)} \geqslant r$

   d) set $s_{t-1}^{\prime(n)} = s_{t-1}^{(j)}$

2. **Propagate** each sample from the set S'_{t-1} by a linear stochastic differential equation

   $$s_t^{(n)} = A s_{t-1}^{\prime(n)} + w_{t-1}^{(n)}$$

   where $w_{t-1}^{(n)}$ is a multivariate Gaussian random variable

3. **Observe** the color distributions

   a) calculate the color distribution

   $$p_{s_t^{(n)}}^{(u)} = f \sum_{i=1}^{I} k\left(\frac{\| s_t^{(n)} - x_i \|}{a}\right) \delta[h(x_i) - u]$$

   for each sample of the set S_t

   b) calculate the Bhattacharyya coefficient for each sample of the set S_t

   $$\rho[p_{s_t^{(n)}}, q] = \sum_{u=1}^{m} \sqrt{p_{s_t^{(n)}}^{(u)} q^{(u)}}$$

   c) weight each sample of the set S_t

   $$\pi_t^{(n)} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(1-\rho[p_{s_t^{(n)}}, q])}{2\sigma^2}}$$

4. **Estimate** the mean state of the set S_t

   $$E[S_t] = \sum_{n=1}^{N} \pi_t^{(n)} s_t^{(n)}$$

Fig. 1 An iteration step of the color-based particle filter

The left image in Fig. 2 shows the application of the color-based particle filter for a surveillance application. To illustrate the distribution of the sample set, the right image displays the Bhattacharyya coefficient for a rectangular region around the tracked face. The samples (black points) are located around the maximum of the Bhattacharyya coefficient which represents the best match to the target model. As can be seen, the calculated mean state (white point) of the sample distribution corresponds well to the maximum and consequently the localization of the face is accurate.

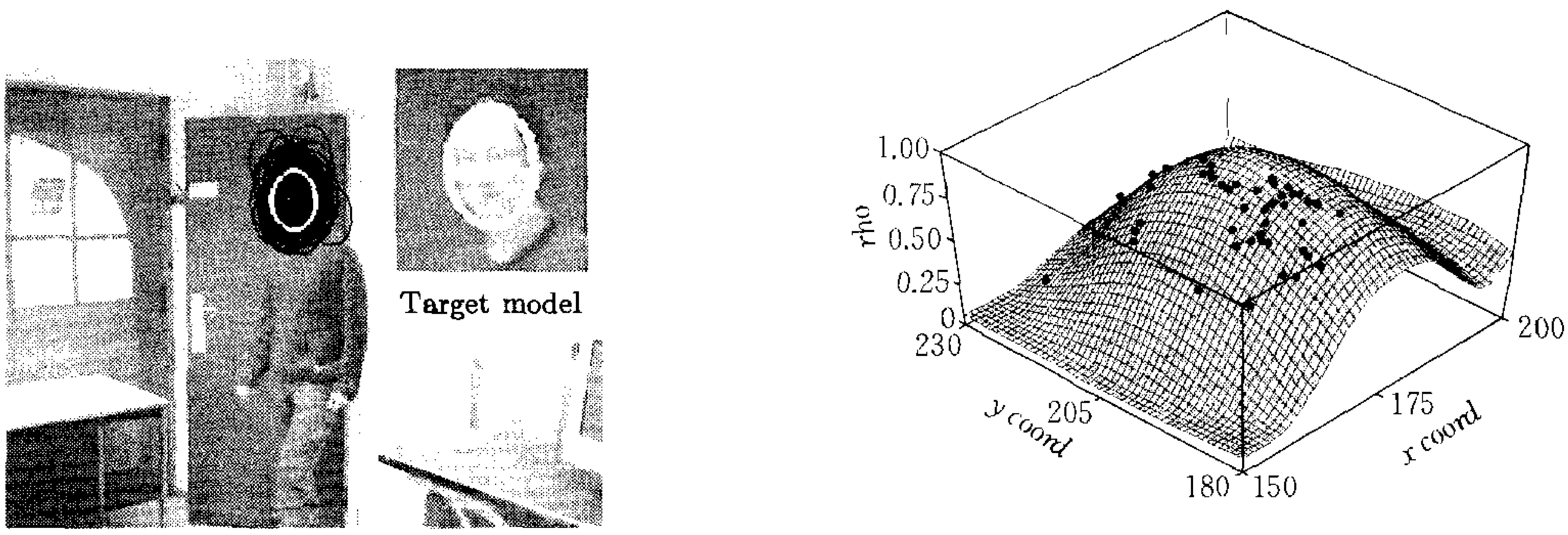


Fig. 2 The left image shows the different samples (black ellipses) as well as the mean state (white ellipse) with respect to a chosen target model for a surveillance application. The right image illustrates the surface plot of the Bhattacharyya coefficient of a small area around the face of the person. The black points indicate the centers of the ellipses of the sample set while the white point represents the mean location which is positioned close to the maximum of the plot

5 Target model update

Illumination conditions, the visual angle, as well as the camera parameters can influence the quality of the color-based particle filter. To overcome the resulting appearance changes we update the target model during slowly changing image observations. By discarding image outliers— where the object is occluded or too noisy—it can be ensured that the model is not updated when the tracker has lost the object. So, we use the update condition

$$\pi_{E[S]} > \pi_T \tag{11}$$

where $\pi_{E[S]}$ is the observation probability of the mean state $E[S]$ and π_T is a threshold. We have found that $\pi_T = 0.9$ has proven to be effective for our applications.

The update of the target model is implemented by the equation

$$q_t^{(u)} = (1-\alpha) q_{t-1}^{(u)} + \alpha p_{E[S_t]}^{(u)} \tag{12}$$

for each bin u where α weights the contribution of the mean state histogram $p_{E[S_t]}$. Thus, we evoke a forgetting process in the sense that the contribution of a specific frame decreases exponentially the further it lies in the past. A similar approach is often used for model updates in figure-background segmentation algorithms[20,21].

To summarize, one single target model is used, respectively adapted, for the whole sample set of the particle filter. We have also considered to use different target models for each sample but the computational cost increases while the results are not significantly better. Furthermore, some samples could adapt themselves to a wrong target.

6 Initialization

For the initialization of the particle filter, we have to find the initial starting values x, y, H_x and H_y. There are three possibilities depending on the prior knowledge of the target object: manual initialization, automatic initialization using a known histogram as target model or an object detection algorithm that finds interesting targets. Whatever the choice, the object must be fully visible, so that a good color distribution can be calculated.

If the target histogram $q = \{q^{(u)}\}_{u=1,2,\cdots,m}$ is known, we can place samples strategically at positions where the target is expected to appear, like shown in Fig. 3. The tracker should detect the object when it enters the field of view of the camera. In this case, the Bhattacharyya coefficient in the vicinity of the object position should be significantly higher than the average coefficient of the background. Therefore, we first calculate the mean value μ and the standard deviation σ of the Bhattacharyya coefficient for elliptic regions over all the positions of the background:

$$\mu = \frac{1}{I}\sum_{i=1}^{I}\rho[p_{x_i}, q] \tag{13}$$

$$\sigma^2 = \frac{1}{I}\sum_{i=1}^{I}(\rho[p_{x_i}, q] - \mu)^2 \tag{14}$$

and then define an appearance condition as

$$\rho[p_{s_t^{(n)}}, q] > \mu + 2\sigma \tag{15}$$

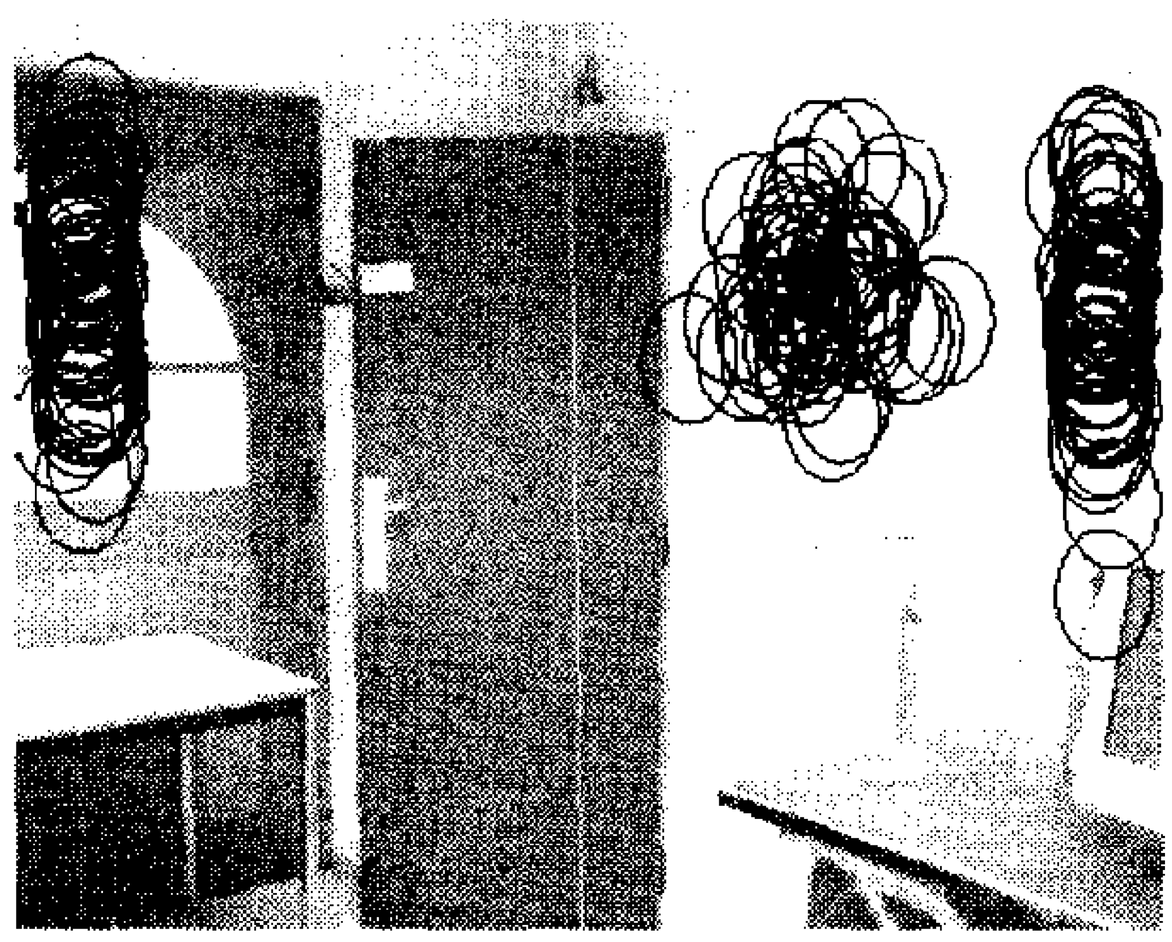


Fig. 3 The particle filter samples can be initially placed at positions where the known human head is most likely to appear, like doors and image borders

This indicates a 95% confidence that a sample does not belong to the background. If more than $b \cdot N$ of the samples fulfill the appearance condition during initialization, we consider the object to be found and start tracking. The parameter b is called the 'kick-off fraction'.

Likewise, the same condition is used to determine if an object is lost during the tracking. If the number of positive appearances is smaller than $b \cdot N$ for a couple of frames, the tracker returns into the 'initialization' mode. In our experiments a value of $b = 0.1$ has been proven sufficient.

In several experiments, the goal was to track faces, and we used an automatic object detection algorithm based on Support Vector Machines[22] for the initialization.

7 Results

This section shows several surveillance sequences which demonstrate the efficiency of the color-based particle filter to track non-rigid objects. We first consider the tracking results of a person and a car in front of a parking place for the public PETS2001 dataset. As presented in Fig. 4 both targets are tracked robustly while the car sequence is especially interesting as scale changes and out-of-plane rotations have to be handled. The results are illustrated by the tracked trajectories and the mean state of every 25th frame. In Fig. 5 the corresponding computing times for the tracked person and the car respectively, with $N=100$ samples are plotted. As can be seen, the proposed algorithm runs comfortably in real-time without any special optimization on a normal 800MHz PC. The calculating time of the color-based particle filter is dependent on the number of samples, the size of the elliptic object region and the number of bins for the histogram, whereas the weighting of the sample set is the most time consuming part of the overall computation.

In Fig. 6 students in front of the Katholieke Universiteit Leuven are being tracked in a sequence of 222 frames. The experiment illustrates the robustness of the proposed initialization strategy. As can be seen in frame 101, the target is temporarily lost as it is completely occluded by the tree, but can be recovered using the appearance condition given in Eq. 15. The switching between the 'initialization' and 'tracking' modes is shown in Fig. 7. A threshold (see the dashed line) which is fixed by the 'kick-off fraction' controls the transition between the two modes. Besides the first occlusion which comes from the tree, the target is two more times hidden by other passing students. As those disappearances are short, the tracker is able to recover simply by propagating the samples according to the system model. The switch to the 'initialization' mode only needs to be done when the object does not reappear for a couple of frames. The last decrease of positive appearances in the plot is forced by leaving the cameras field of view.

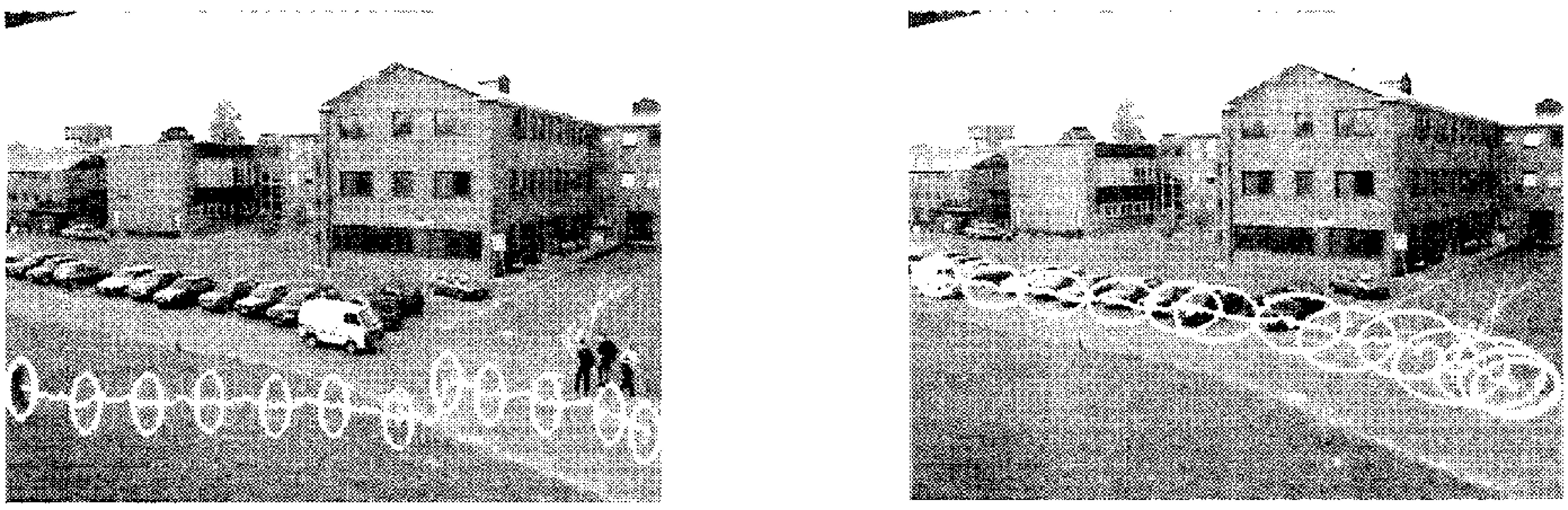


Fig. 4 The left image illustrates the tracking of a person while the right image shows the results of a tracked car for the PETS2001 sequence. The tracked trajectories and the mean state are displayed for every 25th frame

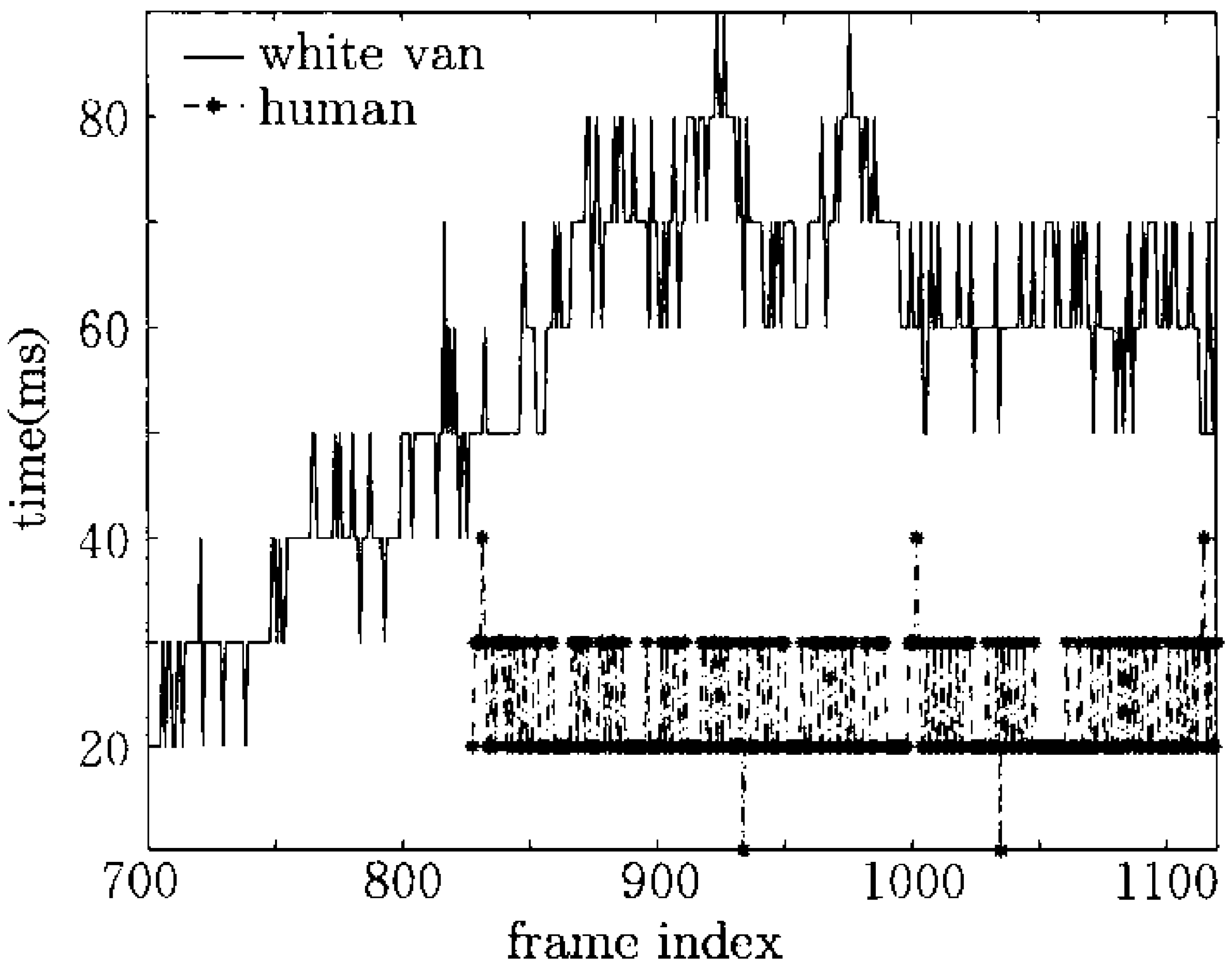
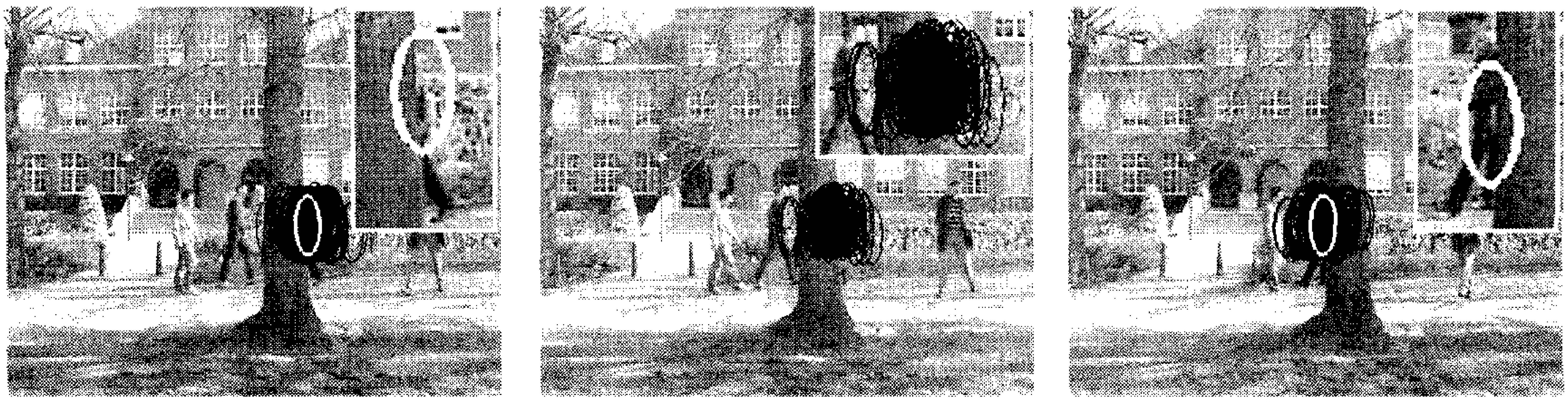


Fig. 5 Computing times for the tracked person and the car respectively, with $N=100$ samples



frame 98 frame 101 frame 112

Fig. 6 The effect of the initialization strategy is shown

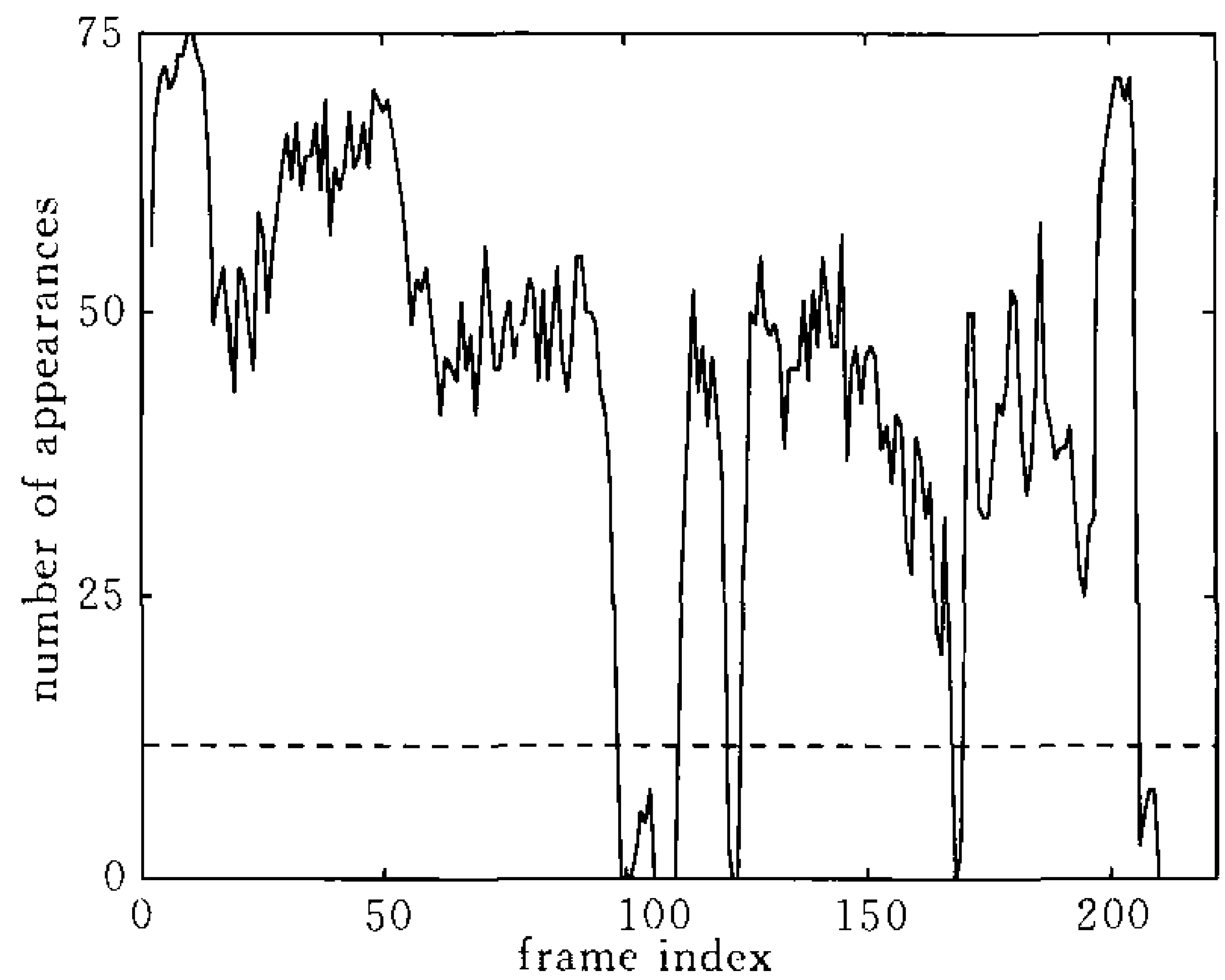


Fig. 7 The number of samples in each frame that fulfill the appearance condition

To demonstrate the importance of the model update we regard another result sequence for the PETS2001 dataset. In Fig. 8 the results of a driver assistance application, where the color-based particle filter has to handle rapid movements of the target and the camera, are shown. Different viewing angles of the tracked car make the experiment quite difficult. Nevertheless, the scale changes as well as the out-of-plane rotations can be handled well by applying an adaptive target model. For a more detailed comparison the mean length of the horizontal half axis H_x of the target are plotted in Fig. 9 for the two approaches. The target is better localized and can be tracked through the whole sequences by applying the model update as the scale is estimated more accurately. For more sequences which illustrate the model update please have a look at [23].

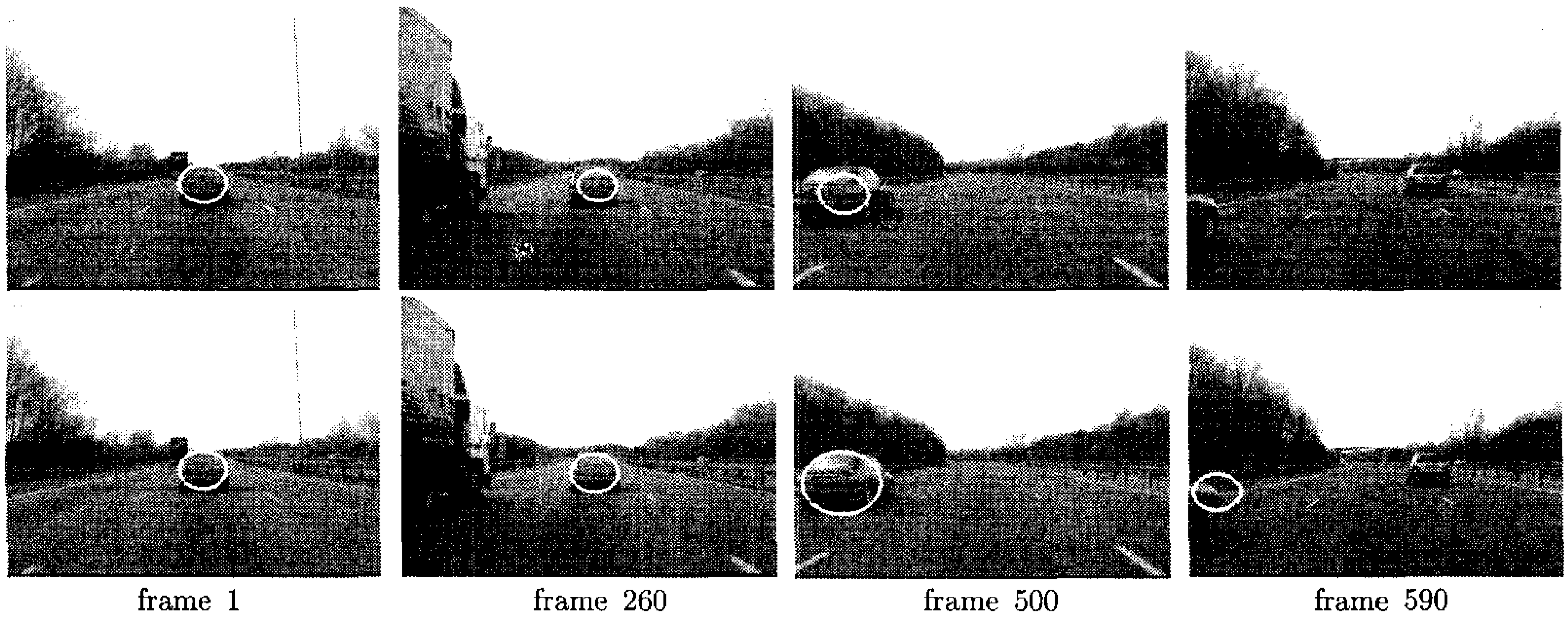


Fig. 8 The traffic sequence of the PETS2001 dataset illustrates the importance of an adaptive target model in cases of large scale changes and out-of-plane rotations. In the top row the tracking results without a model update are presented while in the bottom row an update is applied

We also tested the color-based particle filter with a real application in a factory. The proposed tracker is employed for face tracking in an Augmented Reality environment for training, on-line documentation and planning purposes. In Fig. 10 the result images of a typical scene in a Siemens factory for such a service and training application are shown. The tracked person carries out several operations on different machines, resulting large out-of-plane rotations of the head and unpredictable locations in front of a highly textured background. The color-based particle filter handles all requirements successfully over this

450 frame sequence with only 100 samples. For reasons of anonymity, the face of the tracked person in the resultant images is pixelized, but the tracker was running on the original images.

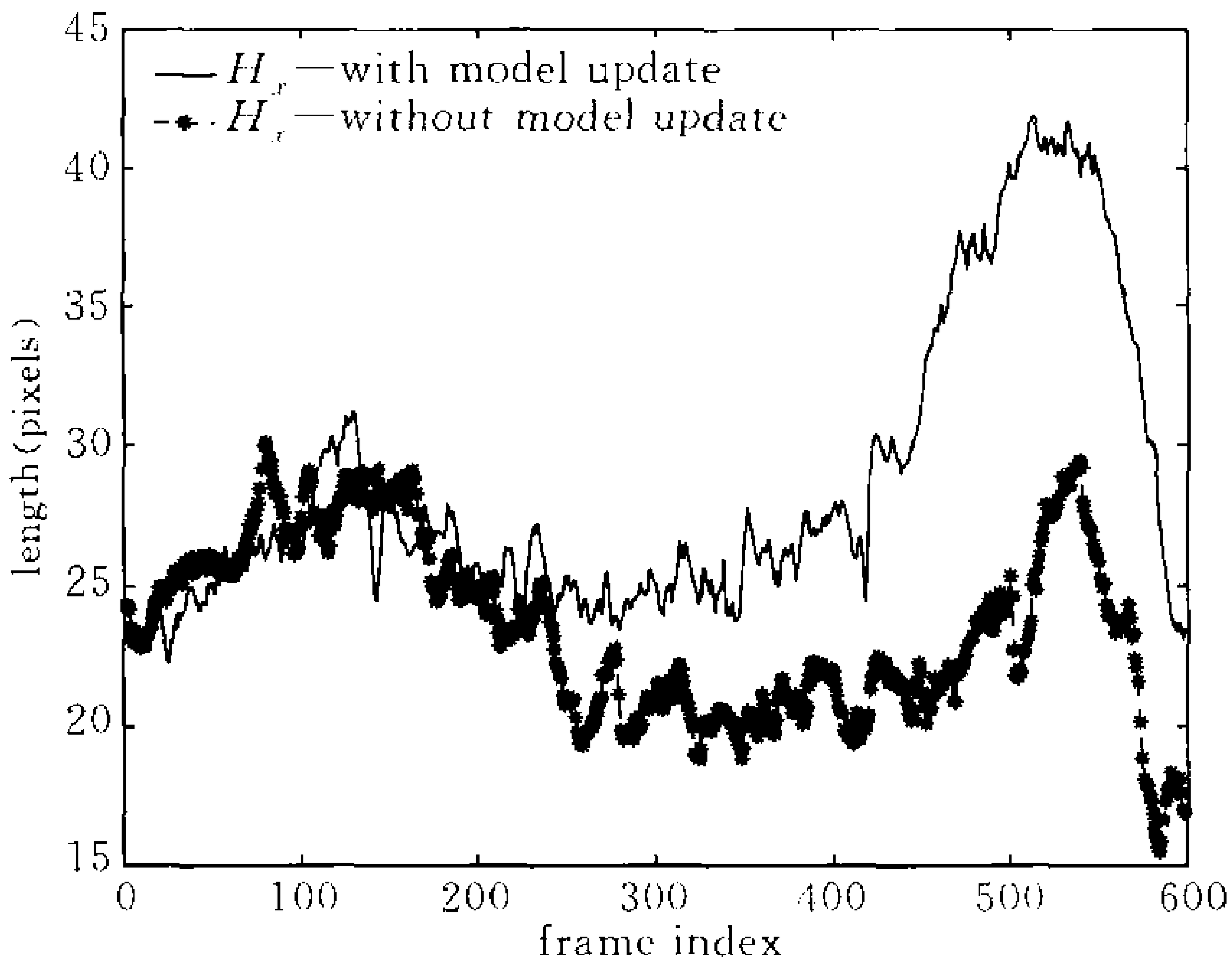


Fig. 9 The scale changes of the horizontal half axes H_x of the mean object state are plotted for the traffic sequence to illustrate the effectiveness of the model update. The improvement of an adaptive target model allows a more accurate representation of the object as the scale is estimated correctly

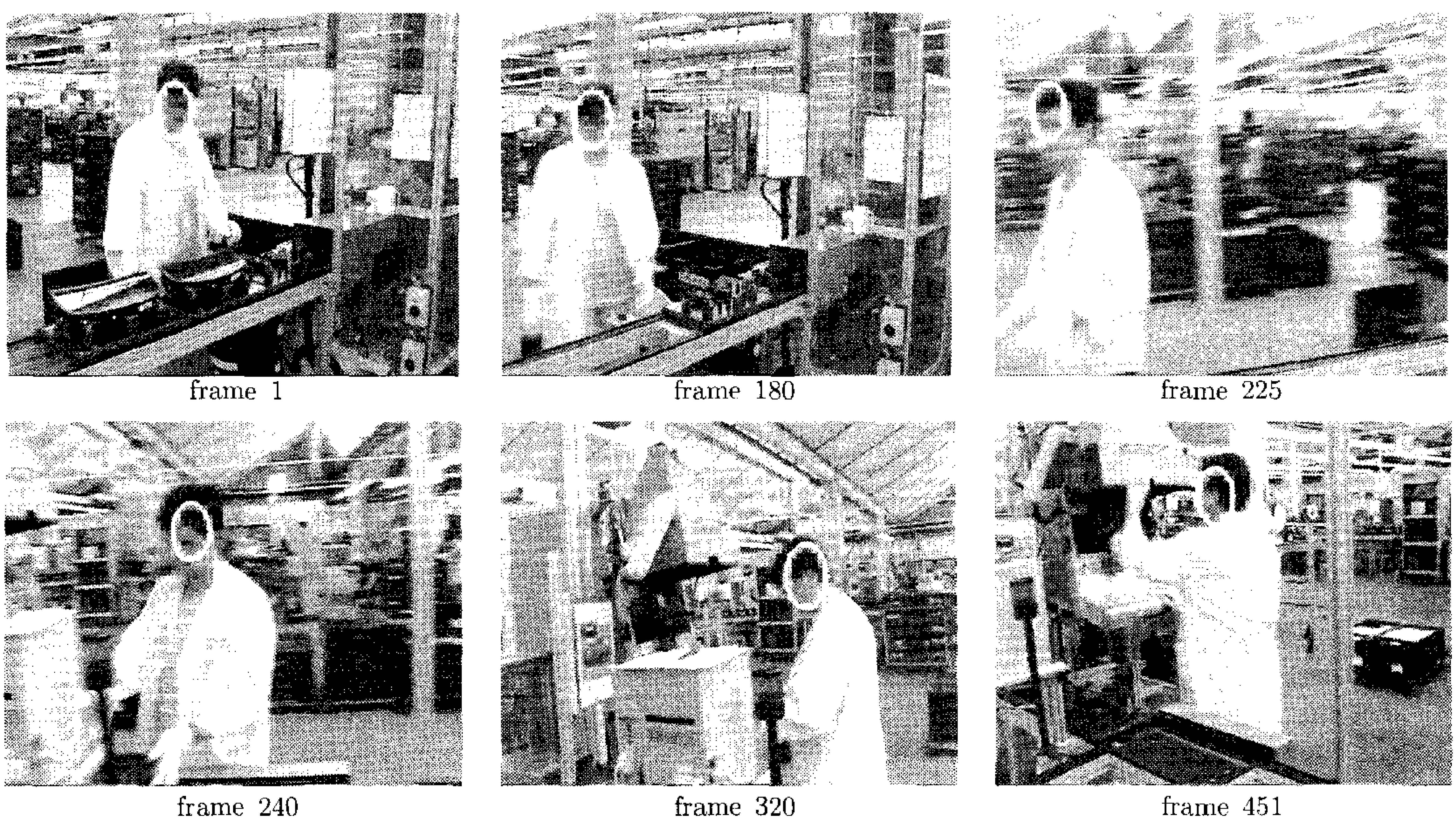


Fig. 10 Head tracking for an Augmented Reality application. Despite large out-of-plane rotations and unpredictable behavior, the proposed tracker follows the head successfully during the sequence. $N = 100$ samples have been used

8 Conclusions

The proposed tracking method adds an adaptive appearance model based on color distributions to particle filtering. The color-based tracker can efficiently and successfully handle non-rigid and fast moving objects under different appearance changes. Our approach focuses on achieving a reliable tracking without having to determine the exact parameters of the non-rigid transformations. Moreover, as multiple hypotheses are processed, objects can be tracked well in cases of occlusions or clutter. The proposed algorithm runs comfortably in real time without any special optimization.

The object model is represented by a weighted histogram which takes into account both the color and the shape of the target. The number of bins in the histogram should be optimized with respect to the noise of the camera, as too many bins can otherwise pose a problem. In these cases, a different similarity measure could be considered that also takes into account neighboring bins. In addition, further improvements can be achieved by using a different weighting function for the histograms to put more emphasis on the shape of the object, i. e. to utilize some prior knowledge of the expected object silhouette to calculate the weighted histogram.

A straightforward kinematic system model is currently used to propagate the sample set. By incorporating more a priori knowledge, for example by employing a learned motion model, the quality of the tracking could be further improved. The application of an adaptive model always implies a trade-off between an increasing sensitivity to extended occlusions and a more reliable tracking under appearance changes.

Our research interests now focus on multiple camera systems that can exchange information about the state of the objects that they track.

References

1 Beymer D, McLauchlan P, Coifman B, Malik J. A real-time computer vision system for measuring traffic parameters. *Computer Vision and Pattern Recognition*, 1997. 495～501

2 Greiffenhagen M, Ramesh V, Comaniciu D, Niemann H. Statistical modeling and performance characterization of a real-time dual camera surveillance system. *Computer Vision and Pattern Recognition*, 2000. 335～342

3 Segen J, Pingali S. A camera-based system for tracking people in real time. In: International Conference on Pattern Recognition, 1996. 63～67

4 Gordon N, Salmond D. Bayesian state estimation for tracking and guidance using the bootstrap filter. *Journal of Guidance, Control and Dynamics*, 1995, **18**(6): 1434～1443

5 Isard M, Blake A. Contour tracking by stochastic propagation of conditional density. *European Conference on Computer Vision*, 1996. 343～356

6 Isard M, Blake A. CONDENSATION—Conditional density propagation for visual tracking. *International Journal on Computer Vision*, 1998, **1**(29): 5～28

7 Kitagawa G. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 1996, **5**(1): 1～25

8 Koller D, Weber J, Malik J. Robust multiple car tracking with occlusion reasoning. In: European Conference on Computer Vision, 1994. 189～196

9 Comaniciu D, Ramesh V, Meer P. Real-time tracking of non-rigid objects using mean shift. *Computer Vision and Pattern Recognition*, 2000. 142～149

10 Jepson A, Fleet D, El-Maraghi T. Robust online appearance models for visual tracking. *Computer Vision and Pattern Recognition*, 2001. 415～422

11 McKenna S, Raja Y, Gong S. Tracking colour objects using adaptive mixture models. *Image and Vision Computing*, 1999, **17**(2): 225～231

12 Raja Y, McKenna S, Gong S. Tracking and segmenting people in varying lighting conditions using colour. In: International Conference on Face and Gesture Recognition, 1998. 228～233

13 Isard M, MacCormick J. BraMBLe: A Bayesian multiple-blob tracker. In: International Conference on Computer Vision, 2001. 34～41

14 Pérez P, Hue C, Vermaak J, Gangnet M. Color-based probabilistic tracking. In: European Conference on Computer Vision, 2002. 661～675

15 Nummiaro K, Koller-Meier e, Van Gool L. A color-based particle filter. In: First International Workshop on Generative-Model-Based Vision, in conjunction with ECCV'02, 2002. 53～60

16 Comaniciu D, Ramesh V. Mean shift and optimal prediction for efficient object tracking. In: International Conference on Image Processing, 2000. 70～73

17 Nummiaro K, Koller-Meier E, Van Gool L. An adaptive color-based particle filter. *Image and Vision Computing*, 2002, **21**(1): 99～110

18 Aherne F, Thacker N, Rockett P. The Bhattacharyya metric as an absolute similarity measure for frequency coded data. *Kybernetika*, 1997, **32**(4): 1～7

19 Kailath T. The divergence and Bhattacharyya distance measures in signal selection. *IEEE Transactions on Communication Technology*, 1967, **15**(1): 52～60

20 Friedman N, Russell S. Image segmentation in video sequences: A probabilistic approach. *Uncertainty in Artificial Intelligence*, 1997. 175～181

21 Halevy G, Weinshall D. Motion of disturbances: Detection and tracking of multi-body non-rigid motion. *Machine Vision and Applications*, 1999, **11**(1): 122～137

22 Burges C. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 1998. 121～167

23 Nummiaro K, Koller-Meier E, Van Gool L. Object tracking with an adaptive color-based particle filter. *Symposium for Pattern Recognition of the DAGM*, 2002. 353～360

Katja Nummiaro Doctor candidate in image processing at the Katholieke Universiteit Leuven ESAT, Belgium since 2001. She received her master degree in surveying from Helsinki University of Technology, Finland 1998. After that she worked in the European Laboratory for Particle Physics (CERN), Switzerland, before joining K. U. Leuven. Her research interests include real-time object tracking, particle filtering and multi-camera applications.

Esther Koller-Meier Received her master degree in computer science in 1995 from the Swiss Federal Institute of Technology (ETH). At the beginning of 2000 she obtained her Ph. D. degree from the Department of Electrical Engineering of the same university. Currently, she is working as postdoctoral research within the Computer Vision Group of the ETH. Her research interests include object tracking, gesture analysis, multi-camera systems and robot vision.

Luc Van Gool Received his master degree in electrical-mechanical engineering and his Ph. D. degree at the Katholieke Universiteit Leuven in 1981 and 1991, respectively. Currently, he is a professor at the Katholieke Universiteit Leuven in Belgium and the ETH in Zurich, Switzerland. He leads computer vision research at both sites, where he also teaches. He has been a program committee member of several major vision conferences. His main interests include 3D reconstruction and modeling, object recognition, grouping and segmentation, tracking and optical flow, robot navigation and registration. In 1998, he received the David Marr Prize together with Marc Pollefeys and Reinhard Koch and an EITC Prize from the European Commission. He was the coordinator of several European projects and is currently involved in a number of European and national projects in both Belgium and Switzerland. He is a cofounder of the company, Eyetronics.

利用颜色的非刚性物体跟踪方法

Katja Nummiaro[1] Esther Koller-Meier[2] Luc Van Gool[1,2]

[1](*Katholieke Universiteit Leuven*, *ESAT/PSI-VISICS*, *Kasteelpark Arenberg* 10, 3001 *Heverlee*, 比利时)

[2](*Swiss Federal Institute of Technology* (*ETH*), *D-ITET/BIWI*, *Gloriastrasse* 35, 8092 *Zurich*, 瑞士)

(E-mail: Katja. Nummiaro@esat. kuleuven. ac. be)

摘 要 提出了一个利用颜色特征实时跟踪非刚性物体的方法. 首先,建立了一个颜色分布模型,该模型对部分遮挡具鲁棒性,对放缩和旋转具不变性,且计算简单. 对非刚体物体的实时鲁棒跟踪是一个非常有挑战性的课题,本文提出了利用颜色特征实时跟踪非刚体物体的方法. 首先,建立了一个颜色分布模型,该模型对部分遮挡具有鲁棒性,对放缩具有不变性,而且计算简单. 然后,采用粒子滤波的方法将颜色分布模型集成到一个动态状态估计的概率框架中. 为了处理光照变化等引起的外貌变化,进一步引入自适应模型更新过程. 同时,本文还讨论了初始化策略用以处理跟踪物体的消失或消失后再出现的情况.

关键词 粒子滤波,颜色分布,Bhattacharyya 系数

中图分类号 TP391.41