Spatially constrained model for mean shift

M.J. Lucena
Departamento de Informática
University of Jaén
Campus Las Lagunillas Edif. A3
23071 - Jaén, Spain
mlucena@ujaen.es

J.M. Fuertes
Departamento de Informática
University of Jaén
Campus Las Lagunillas Edif. A3
23071 - Jaén, Spain
jmf@ujaen.es

N. Pérez de la Blanca
Departamento de Ciencias de la Computación e I.A.
University of Granada
Daniel Saucedo Aranda s/n
18071 - Granada, Spain
nicolas@ugr.es

ABSTRACT

This paper presents a multiple model real-time tracking technique based on the mean-shift algorithm. The proposed approach incorporates spatial information from several connected regions into the histogram-based representation model of the target, and enables multiple models to be used to represent the same object. The use of several regions to capture the color spatial information into a single *combined* model, allow us to increase the object tracking efficiency. We use a model selection function that takes into account both the similarity of the model with the information present in the image, and the target dynamics. In the tracking experiments presented, our method successfully coped with lighting changes, occlusion, and clutter.

Keywords: Non-rigid object tracking, target representation and localization.

1 INTRODUCTION

Object tracking has been studied and applied to numerous computer vision problems which include vehicle tracking, surveillance, medical diagnosis, actor animation, tracking multiple people, and face detection and animation. In this paper we are mainly interested in people tracking although our approach is general. The most recent survey of the state of the art on this topic is given in [8].

The goal of a tracking process is to estimate the state of the object in a time t, represented by a vector $\mathbf{X}_t$, given the set $\mathbf{Z}$ of measurements taken from the sequence of images in times $t-1, t-2, \ldots$

A straightforward way to derive a distribution model $p(\mathbf{X}_t|\mathbf{Z})$, is by using histogram analysis [2, 6, 4]. To do this, the current frame is searched for a region, a fixed-shape variable-size window, whose content best matches a reference color model. Comaniciu et al. [4] proposed a tracking algorithm in which a reference target model is represented by a h-bin histogram that approximates the probability density function (pdf) in the feature space. Maximization is then performed by a mean-shift procedure. Collins [10] improves the algorithm using two different kernels, one for scale and another for motion, what allows more stable tracking on targets with fast motion from or towards the camera.

When the target moves outdoors, noise, shadows and lighting changes can appear which significantly alter the color distributions in the image sequence. In order to deal with these difficulties, Porikli&Tuzel[9] introduce a mean-shift based model update technique with an adaptive change detection method. We take a different approximation, modelling all the appearance information in terms of probability distributions. In this case, a single pdf will be insufficient for modeling and tracking the object reliably. We suggest the use of a set of models which are switched according to a probabilistic rule [7].

The other important point in the mean shift algorithm is the use of the spatial information in the appearance model. The classic formulation [4] encodes spatial information using radially symmetric kernels and therefore it becomes easy for the tracker to get confused with other objects having the same feature distribution but different spatial configurations of features. Several contributions have been done to overcome this problem [1, 11]. In these cases the spatial information is encoded in complex statistical appearance models (spatiograms) using the pixels of the tracking region. In our case the focus is different, we use a more complex tracking region defined as the union of several connected regions, overlapping or not, each one having an appearance model defined by its color histogram. This choice looks more suitable for the tracking objects defined by several regions of homogeneous color.

The aim of this research is to develop a technique for real-time object tracking under variable lighting conditions in a cluttered scene. This paper proposes a generalization of the classical mean-shift tracking model to enable the handling of spatial restrictions, us-

ing several histograms associated to different parts of the tracking region, instead of using only one summarizing all the color information.

The paper is organized as follows: Section 2 introduces the combined model which is used to incorporate spatial information into the histogram-based model; experimental results are shown in Section 3; and Section 4 presents our conclusions.

2 PROPOSED METHOD

2.1 Tracking

The pdfs that characterize the target model and the target candidate are given by *h*-bin histograms, extracted from ellipsoidal regions of the image.

$$\begin{aligned} \text{target model:} \quad & \hat{\mathbf{q}} = \{\hat{q}_u\}_{u=1\ldots h} \quad \sum_{u=1}^{h} \hat{q}_u = 1 \\ \text{target candidate:} \quad & \hat{\mathbf{p}}(\mathbf{y}) = \{\hat{p}_u(\mathbf{y})\}_{u=1\ldots h} \quad \sum_{u=1}^{h} \hat{p}_u = 1 \end{aligned}$$

Where $\mathbf{y}$ is the spatial location of the target candidate. A similarity function $\hat{\rho}(\mathbf{y})$ is defined, whose local maxima in the image indicate the presence of objects having representations similar to $\hat{\mathbf{q}}$. The tracking process is then defined as a search procedure for those maxima. In order to avoid the computational cost associated with a gradient-based optimization, Comaniciu et al. regularize the similarity function by masking the region of interest with an isotropic Epanechnikov kernel in the spatial domain [3] and apply a mean-shift technique. Previously, the region of interest which defines the target is normalized to a unit circle, by independently rescaling the horizontal and vertical dimensions of the original ellipsoid.

The similarity function defines a distance between the target model and the candidates. The distance between two discrete distributions is defined as:

$$d(\mathbf{y}) = \sqrt{1 - \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}]} \tag{1}$$

where the similarity function will be denoted by:

$$\hat{\rho}(\mathbf{y}) \equiv \rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}] = \sum_{u=1}^{h} \sqrt{\hat{p}_u(\mathbf{y})\hat{q}_u} \tag{2}$$

which is the sample estimate of the Bhattacharyya coefficient between $\mathbf{p}$ and $\mathbf{q}$ [5].

2.2 Adding Spatial Information

One of the main drawbacks presented by histogram-based models is the absence of any spatial information. However, it is sometimes interesting to be able to include spatial restrictions in the object model. For example, when attempting to track a person walking, it is possible to identify two areas with typically different histograms, corresponding to the target's torso and head, with a specific spatial relationship between both of these. In this section, we extend the target model in order to incorporate this kind of information.

Our combined model Q is a set comprising m regions, overlapping or not, each of which is characterized by a distribution $\hat{\mathbf{q}}^i$, together with the offset value $\Delta\mathbf{y}^i$ of each region with respect to the location where the model is centered, $\mathbf{y}$. The new target candidate $P(\mathbf{y})$ is given by an analogous expression:

$$\begin{aligned} \text{Combined model:} \quad & Q = \{\hat{\mathbf{q}}^i, \Delta\mathbf{y}^i\}_{i=1\ldots m} \\ \text{Combined candidate:} \quad & P(\mathbf{y}) = \{\hat{\mathbf{p}}^i(\mathbf{y} + \Delta\mathbf{y}^i)\}_{i=1\ldots m} \end{aligned} \tag{3}$$

The *Bhattacharyya coefficient* corresponding to Q for a given location $\mathbf{y}$ of the image is therefore given by the following expression:

$$\rho[P(\mathbf{y}), Q] = \frac{1}{m}\sum_{i=1}^{m} \rho[\hat{\mathbf{p}}(\mathbf{y} + \Delta\mathbf{y}^i), \hat{\mathbf{q}}^i] \tag{4}$$

The following algorithm is used to estimate, in the new frame, the location $\mathbf{y}_1$ of the maximum value of the *Bhattacharyya coefficient*, starting from the location $\mathbf{y}_0$ estimated for the previous frame:

Algorithm: Given the combined target model Q, comprised by m regions, and its location $\mathbf{y}_0$ in the previous frame:

1. For each i from 1 to m:

   (a) Compute the weights $\{\omega_j^i\}_{j=1\ldots n_i}$ [4]:

   $$\omega_j^i = \sum_{u=1}^{h} \sqrt{\frac{\hat{q}_u^i}{\hat{p}_u^i(\mathbf{y}_0 + \Delta\mathbf{y}^i)}}\,\delta[\beta(\mathbf{x}_j^i) - u] \tag{5}$$

   (b) Obtain $\mathbf{y}_1^i$ from:

   $$\mathbf{y}_1^i = \frac{\sum_{j=1}^{n_i} \mathbf{x}_j^i \omega_j^i k'(\mathbf{x}_j^i)}{\sum_{j=1}^{n_i} \omega_j^i k'(\mathbf{x}_j^i)} \tag{6}$$

2. The next location of the target is computed as

   $$\mathbf{y}_1 = \arg\min_{\mathbf{y} \in \{\mathbf{y}_1^1, \ldots \mathbf{y}_1^m\}} \{\rho[P(\mathbf{y}), Q]\} \tag{7}$$

3. If $||\mathbf{y}_1 - \mathbf{y}_0|| < \varepsilon$, return $\mathbf{y}_1$ and stop.
   Otherwise, set $\mathbf{y}_0 \leftarrow \mathbf{y}_1$ and go to Step 1.

Where n_i is the number of pixels of the i–th region, $\mathbf{x}_j^i$ are the locations of their pixels, normalized to the unit circle. $\beta(\mathbf{x})$ represents the associated bin at the pixel located on $\mathbf{x}$, h is the histogram size, and δ is the

Kronecker delta function. $k(\mathbf{x})$ is the Epanechnikov kernel function, and $k'(\mathbf{x})$ represents its derivative.

Values $\hat{q}_u^i$ and $\hat{p}_u^i$ are defined by the expressions:

$$\hat{q}_u^i = C_q \sum_{j=1}^{n_i} k(\mathbf{x}_j^i)\delta[\beta(\mathbf{x}_j^i) - u] \tag{8}$$

$$\hat{p}_u^i(\mathbf{y}) = C_p \sum_{j=1}^{n_i} k(\mathbf{x}_j^i - \mathbf{y})\delta[\beta(\mathbf{x}_j^i) - u] \tag{9}$$

where C_q and C_p are normalization constants obtained by imposing the conditions $\sum_u \hat{q}_u^i = 1$ and $\sum_u \hat{p}_u^i = 1$ for each i.

When the number of regions $m = 1$, our combined model is reduced to the model proposed by Comaniciu et al.

2.3 Scale Selection

Different approaches for dealing with scale changes have been proposed. These techniques define a scale factor σ, which allows to adjust the bandwidth of the kernel profile, i.e. the size of the ellipsoid where the target candidate histogram is computed.

The approach proposed in [4] works as follows: Given the scale σ_{prev} of the previous frame, we run the target localization algorithm three times, with $\sigma_i = \sigma_{prev} + i \cdot \Delta\sigma$, for $i \in \{-1, 0, 1\}$. The new scale σ_{new} is then computed as

$$\sigma_{new} = \gamma \cdot \sigma_{opt} + (1 - \gamma) \cdot \sigma_{prev}, \tag{10}$$

where σ_{opt} is the value of σ_i which gives the best Battacharyya coefficient. In our experiments, we have chosen the default values proposed in [4] $\gamma = 0.1$ and $\Delta\sigma = 0.1\sigma_{prev}$.

In order to employ the scale selection technique proposed in [10], some modifications to the tracking algorithm are needed:

- The kernel function $k(x)$ is replaced by a Difference-of-Gaussian filter $H_x(x, s)$, where s is a scale parameter, as defined in [10].
- Expression (6) must be replaced by:

$$\mathbf{y}_1^i = \frac{\sum_s H_s(s) \sum_{j=1}^{n_i} \mathbf{x}_j^i \omega_j^i H_x'(\mathbf{x}_j^i, s)}{\sum_s H_s(s) \sum_{j=1}^{n_i} |\omega_j^i H_x'(\mathbf{x}_j^i, s)|} \tag{11}$$

  where $-n \leq s \leq n$ defines a range of scales σ_s, where $\sigma_s = \sigma_{prev} \cdot b^s$, with $n = 2$, $b = 1.1$, and $H_s(s) = 1 - (s/n)^2$.

- A mean-shift step is applied to estimate scale for every region, using the following equation:

$$s_i' = \frac{\sum_s \sum_{j=1}^{n_i} H_x(\mathbf{x}_j^i, s) \omega_j^i s}{\sum_s \sum_{j=1}^{n_i} H_x(\mathbf{x}_j^i, s) \omega_j^i} \tag{12}$$

  Being s' the scale value s_i' which gives the best Bhattacharyya coefficient, the scale for the next step is computed as $\sigma_{new} = \sigma_{prev} \cdot b^{s'}$.

2.4 Model Selection

In order to prevent loss of the target due to changes in object orientation and lighting conditions, we will extend the approach proposed in [7] to deal with scale changes. We define $\mathbf{M}$ as a multiple model, comprising a set of n combined models, corresponding to several histograms extracted from images generated by the object under different orientations and lighting conditions:

$$\mathbf{M} = \{Q_0, Q_1, \ldots, Q_{n-1}\} \tag{13}$$

We can thus run the target localization algorithm for each Q_i, proposed in Sections 2.2 and 2.3, and obtain a set $\mathbf{Y}$ of image locations and scales:

$$\mathbf{Y} = \{(\mathbf{y}_0, \sigma_0), (\mathbf{y}_1, \sigma_1), \ldots, (\mathbf{y}_{n-1}, \sigma_{n-1})\} \tag{14}$$

and a set $\mathbf{B}$ of coefficients:

$$\mathbf{B} = \{b_0, b_1, \ldots, b_{n-1}\}, \tag{15}$$

where each ($\mathbf{y}_i$, σ_i) represent the location and scale where the model Q_i maximizes the Bhattacharyya coefficient (4), and each $b_i = \rho[P(\mathbf{y}_i), Q_i]$, computed at scale σ_i, represents the actual value of the coefficient itself.

For each frame, we then need to select the combined model in $\mathbf{M}$ which best fits the observed image. Selecting the Q_i with the largest Bhattacharyya coefficient may increase the risk of distractions with image regions having similar histograms to the ones present in our model. In order to avoid this, we can weight the estimated values, giving more importance to the ones with the greatest coherence with the dynamics observed for the target so far. With this goal in mind, we will define a probability distribution based on the displacement between location $\mathbf{y}_i$ and scale σ_i estimated by the tracker, and predicted location $\bar{\mathbf{y}}$ and scale $\bar{\sigma}$ for the target, given by some dynamic model for the object to be tracked. In addition, we will consider that each of the combined models $Q_i \in \mathbf{M}$ presents an *a priori* probability $p(Q_i)$ of appearing in each image of the sequence.

We will define the probability of each Q_i, given $\mathbf{B}$, as:

$$p(Q_i/\mathbf{B}) = \frac{b_i \cdot p(Q_i)}{\sum_j \left(b_j \cdot p(Q_j)\right)} \tag{16}$$

and the probability of each Q_i, given $\mathbf{Y}$, as:

$$p(Q_i/\mathbf{Y}) = \frac{p(\bar{\mathbf{y}}-\mathbf{y}_i, \bar{\sigma}-\sigma_i)\cdot p(Q_i)}{\sum_j \Big(p(\bar{\mathbf{y}}-\mathbf{y}_j, \bar{\sigma}-\sigma_j)\cdot p(Q_j)\Big)} \quad (17)$$

In our case, we suppose that the $(\bar{\mathbf{y}}-\mathbf{y}_i)$ and $(\bar{\sigma}-\sigma_i)$ values follow a multivariate zero-mean Gaussian distribution, i.e. $p(\bar{\mathbf{y}}-\mathbf{y}_i, \bar{\sigma}-\sigma_i) \sim N(0;\sigma_x,\sigma_s)$. According to Bayes' rule,

$$p(\mathbf{Y}/Q_i) = \frac{p(Q_i/\mathbf{Y})\cdot p(\mathbf{Y})}{p(Q_i)} \quad (18)$$

and

$$p(\mathbf{B}/Q_i) = \frac{p(Q_i/\mathbf{B})\cdot p(\mathbf{B})}{p(Q_i)} \quad (19)$$

and also

$$p(\mathbf{B},\mathbf{Y}/Q_i) = \frac{p(Q_i/\mathbf{B},\mathbf{Y})\cdot p(\mathbf{B},\mathbf{Y})}{p(Q_i)} \quad (20)$$

We assume that $\mathbf{B}$ and $\mathbf{Y}$ are statistically independent given Q_i, i.e.

$$p(\mathbf{B},\mathbf{Y}/Q_i) = p(\mathbf{B}/Q_i)\cdot p(\mathbf{Y}/Q_i) \quad (21)$$

Replacing expressions (18), (19) and (20) in (21), we obtain the following equation:

$$p(Q_i/\mathbf{B},\mathbf{Y}) = \frac{p(Q_i/\mathbf{B})\cdot p(Q_i/\mathbf{Y})}{p(Q_i)}\cdot C \quad (22)$$

where $C = \frac{p(\mathbf{B})\cdot p(\mathbf{Y})}{p(\mathbf{B},\mathbf{Y})}$ is a constant term.

Finally, substituting $p(Q_i/\mathbf{B})$ and $p(Q_i/\mathbf{Y})$ for expressions (16) and (17) in Equation (22), and discarding the constant C, we obtain the following expression for $S(i)$:

$$\left[\frac{b_i\cdot p(\bar{\mathbf{y}}-\mathbf{y}_i, \bar{\sigma}-\sigma_i)\cdot p(Q_i)}{\sum_j \Big(b_j\cdot p(Q_j)\Big)\cdot \sum_k \Big(p(\bar{\mathbf{y}}-\mathbf{y}_k, \bar{\sigma}-\sigma_k)\cdot p(Q_k)\Big)}\right] \quad (23)$$

used to select the best model Q^* for each frame, which is the model Q_i with highest $S(i)$ value.

In practice, the proposed framework consists of two phases: first, the *mean shift* algorithm is applied to each Q_i comprising the multiple model. Next, the specific model which best matches the observation is selected.

In order to avoid processing all the models for each frame, specially when their number is high, we can represent the multiple model by means of a graph, where each node corresponds with a simple model Q_i. Two nodes will be connected if they can appear in two consecutive frames of a typical sequence. As we deal with video sequences, we can assume that the model changes must be gradual, and so only those nodes which are sufficiently similar in the graph will be connected. Therefore, it will be only necessary to examine (in each frame) the neighbors of the actual model Q^*, enabling us to restrain the computational requirements of the technique.

Sequence	*Frames*	*Resolution*	*Ground truth*
eps	208	320×240	Yes
paddle	501	320×240	Yes
player	142	352×288	No

Table 1: Sequences used in the experiments.

3 EXPERIMENTAL RESULTS

In order to evaluate his performance, the method proposed in this article has been applied to various sequences. In this section, some representative results are presented on mpeg-compressed sequences (Table 1), for some of which we have ground truth data. The RGB color space was taken as the feature space, quantized into $8\times8\times8$ bins. For scale selection, we have employed the technique proposed in [4] (see Section 2.3) to all the sequences, except for the *paddle* sequence, in which we have used the Collins approach [10].

In the images showing the tracking results, small squares are superimposed in the upper-left corner of each frame indicating the active simple model in the corresponding multiple model. All of these are shown in white, except for the one corresponding to the model chosen in each frame, which is displayed in gray.

3.1 Dynamical Model

In order to estimate the expected location $\bar{\mathbf{y}}^{t+1}$ and scale $\bar{\sigma}^{t+1}$ of the target in frame $t+1$ needed to compute $p(Q_i/\mathbf{B},\mathbf{Y})$, we have defined a simple dynamical model, that suffices to our purposes:

$$\begin{aligned} \bar{\mathbf{y}}^{t+1} &= \mathbf{y}^t + \mathbf{d}^t \\ \mathbf{d}^t &= \lambda\cdot(\mathbf{y}^t - \mathbf{y}^{t-1}) + (1-\lambda)\cdot\mathbf{d}^{t-1} \\ \bar{\sigma}^{t+1} &= \sigma^t \end{aligned} \quad (24)$$

where $\mathbf{y}^t$, σ^t represent the location and scale of the target estimated by the tracking algorithm at time t, and $\mathbf{d}^0$=0. In our experiments, we have used a value for λ of 0.5. In the case of a specially adapted tracker for a certain object, the dynamical model can be replaced by a more suitable one, or learnt from examples.

3.2 Performance Metrics

In order to evaluate our method we have employed several localization metrics to the results obtained from the sequences from which we have ground

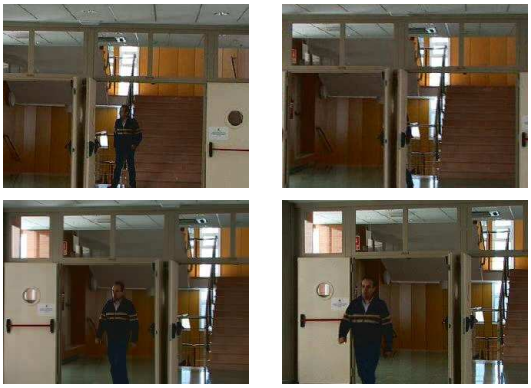

Figure 1: *Eps* sequence: Frames 41, 81, 121 and 161.

	OAR	ADC	BAP	OTE
Non Combined	28.36	34.24	30.51	90.50
Combined	67.44	77.49	70.47	10.10

Table 2: Performance results obtained for the *eps* sequence using combined and non-combined models.

truth data. Being $\mathbf{g} = (g_x, g_y)$ the real location of the object, $\mathbf{e} = (e_x, e_y)$ the target location estimated by the tracker, A_C the area corresponding to the object, and A_T the area estimated by the tracker:

- Object Area Recall: $OAR = 100 \cdot \dfrac{|A_T \cap A_C|}{|A_T|}$
- Box Area Precision: $BAP = 100 \cdot \dfrac{|A_T \cap A_C|}{|A_C|}$
- Area Dice Coefficient: $ADC = 100 \cdot \dfrac{2 * |A_T \cap A_C|}{|A_T| + |A_C|}$
- Object Tracking Error: $OTE = ||\mathbf{e} - \mathbf{g}||$

All of the mentioned metrics are averaged over the whole sequence.

3.3 Experiments

The *eps* sequence (Figure 1) was taken with a home video camera, showing a person coming down some stairs and walking along in poor lighting conditions. In this example, the target goes behind a door at around frame 80, and is occluded for several frames. As we can see, the non combined model can't overcome this occlusion. Figure 4 shows the Bhattacharyya coefficient between combined and non-combined models and the target estimated location along the sequence. When the target disappears, combined model coefficient decreases, while the non-combined one remains stable. This suggests that the first one falls into a local maximum.

As can be seen in Table 2, the best results are obtained by the combined models. The other tracking result is significantly worse, because the target is missed.

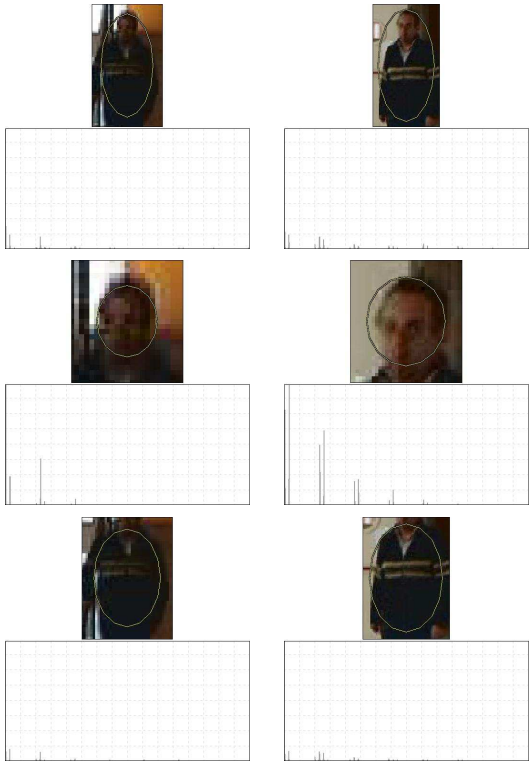

Figure 2: Models used for the *eps* sequence, with their corresponding weighted histograms.

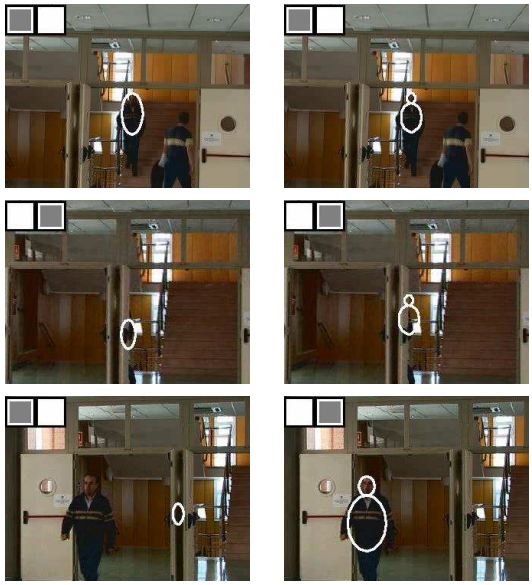

Figure 3: Results obtained with the *eps* sequence, using non-combined models (left) and combined models (right). In the upper-left corner, the active model is shown in gray.

 ISBN 978-86943-15-2

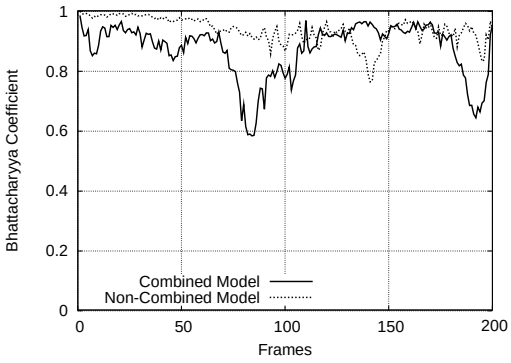

Figure 4: Bhattacharyya coefficient between the combined and non-combined models and the object's estimated location in the *eps* sequence.

Figure 5: *Paddle* sequence: Frames 86, 171, 256 and 341.

	OAR	ADC	BAP	OTE
Non Combined	29.39	80.70	40.94	22.02
Combined	55.20	97.15	69.45	6.77

Table 3: Performance results obtained for the *paddle* sequence using combined and non-combined models.

The *paddle* sequence (Figure 5) lasts 20 seconds, and shows a person playing paddle, moving, changing position and moving out of the shade. In this experiment, we have employed the Collins approach (see Section 2.3) to estimate target location and scale. As we can see, almost all of the models track correctly the target, with the multiple combined model behaving best (even when the target is inclined).

The performance results for the *paddle* sequence are shown in Table 3. As we can see, the combined models perform better than the non-combined ones.

The *player* sequence (Figure 8) represents several soccer players. It was taken from a television news broadcast and shows various players from the same team, moving quickly and with sudden zoom and panning movements. In this case, three simple models have been employed, three combined models, and the associated multiple models. The results are shown

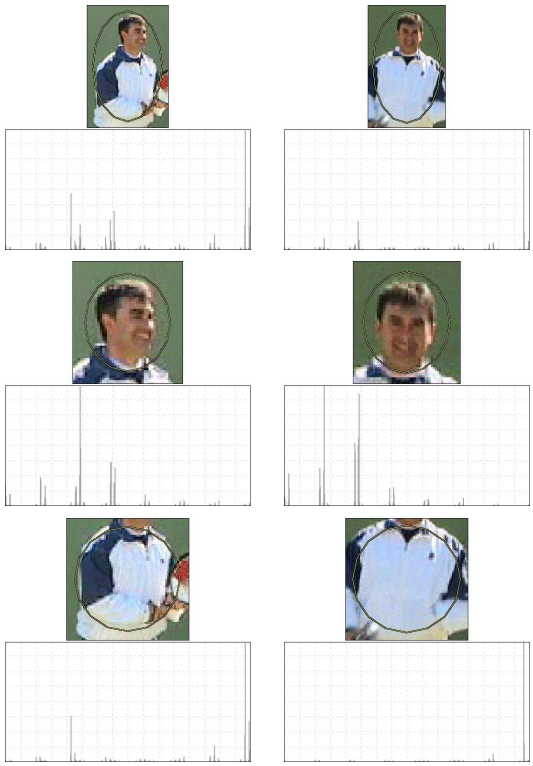

Figure 6: Models used for the *paddle* sequence, with their corresponding weighted histograms.

Figure 7: Results obtained with the *paddle* sequence, using non-combined models (left) and combined models (right). In the upper-left corner, the active model is shown in gray.

Figure 8: *Player* sequence: Frames 26, 51, 76 and 101.

Figure 9: Models used for the *player* sequence, with their corresponding weighted histograms.

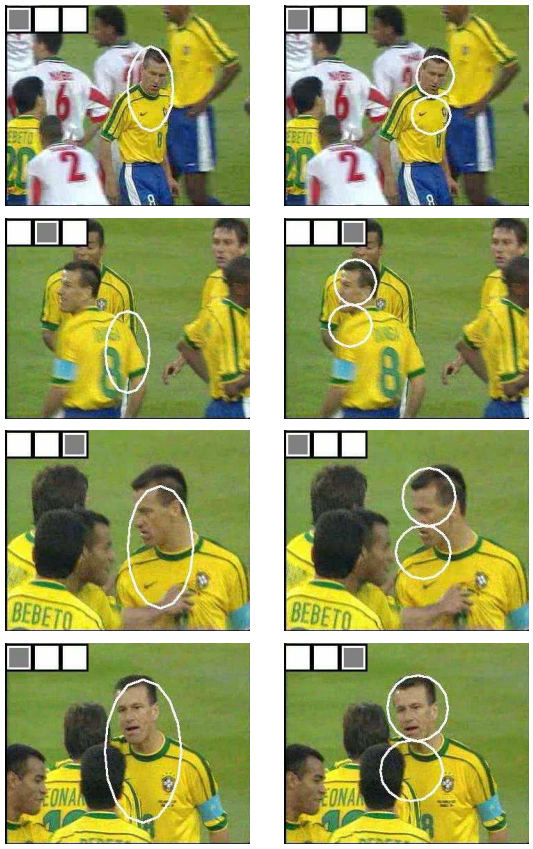

Figure 10: Results obtained with the *player* sequence, using non-combined models (left) and combined models (right). In the upper-left corner, the active model is shown in gray.

in Figure 10, and reveal once again how the combined multiple model behaves best, enabling the target to be tracked even when there are partial occlusions or when similar objects appear on the scene. Figure 11 shows that, like in the *eps* experiment, the Bhattacharyya coefficient of the combined model is generally lower than the non-combined model one. This suggests again that the combined model is less prone to fall into local maxima.

The experiments shown in this paper run at over 20 frames/second (40 in some cases) on a Pentium IV 3GHz desktop computer, except the ones that use the Collins scale selection technique (see Section 2.3), which run at 10 frames/second, due to the higher computational requirements of that approach.

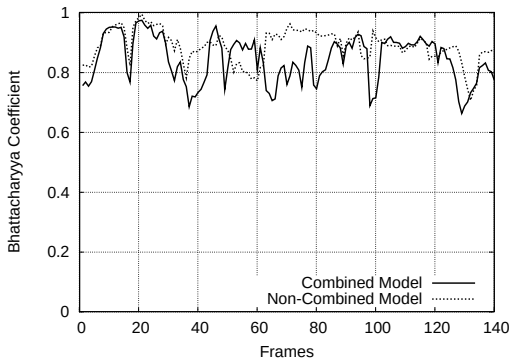

Figure 11: Bhattacharyya coefficient between the combined and non-combined models and the object's estimated location in the *player* sequence.

4 CONCLUSIONS

A combined tracking model has been presented, which enables spatial information to be incorporated into the histogram-based models, increasing the robustness of

the tracking process. On this, a distance measure has been defined which can be used in a minimization process based on the mean-shift algorithm.

The proposed combined models provided better results in our experiments than the classic approaches. Although the proposed technique involve a higher computational cost, this is low enough for them to be used in real time.

Our tracking scheme can be extended in several ways in order to be more flexible and applicable in real world situations: extending the combined model to deal with rotations and articulated motion, integrating into the tracking scheme more information sources, such as edges or optical flow.

5 ACKNOWLEDGMENTS

This work has been financed by Grant TIC-2005-1665 from the Spanish Ministry of Science and Technology.

REFERENCES

[1] S. Birchfield and S. Rangarajan. Spatiograms versus histograms for region-based tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 1158–1163, 2005.

[2] G. R. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, 2, 1998.

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[4] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577, 2003.

[5] T. Kailath. The divergence and Bhatacharyya distance measures in signal selection. *IEEE Transactions on Comm. Technology*, 15:52–60, 1967.

[6] T. L. Liu and H. T. Chen. Real-time tracking using trust-region methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):397–402, 2004.

[7] M. Lucena, J. M. Fuertes, and N. Perez de la Blanca. Real-time tracking using multiple target models. *Lecture Notes in Computer Science*, 3522:20–27, 2005.

[8] T. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding.*, 104:90–126, 2006.

[9] Faith Porikli and Oncel Tuzel. Human Body Tracking by Adaptive Background Models and Mean-Shift Analysis. Technical Report TR-2003-036, Mitsubishi Electric Research Laboratories, July 2003.

[10] R. T. Collins. Mean-shift Blob Tracking through Scale Space. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 234–240, 2003.

[11] Q. Zhao and H. Tao. Object tracking using color correlogram. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 263–270, 2005.