OBJECT TRACKING BASED ON PARTICLE FILTERING WITH MULTIPLE APPEARANCE MODELS

Nicolas Widynski
Télécom ParisTech, CNRS LTCI, Paris, France
nicolas.widynski@telecom-paristech.fr

Emanuel Aldea, Séverine Dubuisson
University Pierre et Marie Curie, UPMC-LIP6, Paris, France
{emanuel.aldea, severine.dubuisson}@lip6.fr

Isabelle Bloch
Télécom ParisTech, CNRS LTCI, Paris, France
isabelle.bloch@telecom-paristech.fr

Keywords: Object tracking in video Sequences, Particle filter, Multiple appearance models.

Abstract: In this paper, we propose a novel method to track an object whose appearance is evolving in time. The tracking procedure is performed by a particle filter algorithm in which all possible appearance models are explicitly considered using a mixture decomposition of the likelihood. Then, the component weights of this mixture are conditioned by both the state and the current observation. Moreover, the use of the current observation makes the estimation process more robust and allows handling complementary features, such as color and shape information. In the proposed approach, these estimated component weights are computed using a Support Vector Machine. Tests on a mouth tracking problem show that the multiple appearance model outperforms classical single appearance likelihood.

1 INTRODUCTION

Using several features or sensors in a tracking particle filter based procedure has abundantly been studied in the literature. For instance, the authors in (Brasnett et al., 2007) propose to combine color, edge and texture information, (Maggio et al., 2005) use color and texture, and (Muñoz-Salinas et al., 2008) use color and a distance map. Integrating several modalities into the particle filter is still a challenging problem, since it requires to take into account the context to cope with different situations. In (Hotta, 2006; Xu and Li, 2005), a mixture density is used to model the likelihood, whose weights correspond to the probability for the considered modality to be the most relevant one. Confidence exponents in feature likelihoods are considered in (Brasnett et al., 2007; Erdem et al., 2010), and an uncertainty principle of a feature with a dispersion criterion of the computed likelihoods in (Maggio et al., 2005). However, most of existing methods require to evaluate a posteriori the parameters of the modality relevance, i.e. after the particle filter estimation, and therefore deliver a unique set a parameters for the particle cloud. This strategy may fail, since assigning the same weights to all the particles may induce error propagations.

In this article, we propose to decompose the likelihood into multiple appearance models. This problem is close to the changing appearance one proposed in (Nummiaro et al., 2002; Muñoz-Salinas et al., 2008). The main difference is that we explicitly consider several appearance models. This allows us to use several features in an original way. Furthermore, we consider a mixture likelihood model whose weights depend on the state and the current observation, and are then unique for each particle. As an original application, we propose to compute the weights with a Support Vector Machine, and to experiment the proposed model on a mouth tracking application.

2 PARTICLE FILTERING

Let us consider a classical filtering problem and denote by $\mathbf{x}_t \in \mathbb{X}$ the hidden state of a stochastic process at time t and by $\mathbf{y}_t \in \mathbb{Y}$ the measurement state. The non-linear Bayesian tracking consists in estimating the posterior filtering density function $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ through a non-linear transition function $\mathbf{x}_t = f_t(\mathbf{x}_{t-1}, \mathbf{v}_t)$ and a non-linear measurement function $\mathbf{y}_t = h_t(\mathbf{x}_t, \mathbf{w}_t)$. Particle filters, also known as sequential Monte-Carlo methods, are used to approximate the posterior distribution by a weighted sum of Dirac masses centered on hypothetical state realizations of the state $\mathbf{x}_t$, also called particles. For more details about particle filters techniques, see (Doucet et al., 2001).

In this paper, we focus on the design of the likelihood density function induced by h_t, which is of prime interest in Bayesian estimation methods, and especially in a particle filter framework, since it weights the particle cloud. As we will see in the next section, using several features or sensors usually helps, but imposes to properly model these different pieces of information, which can be redundant or conflicting. In Section 4, we propose a new method for integrating several features in the likelihood process, by partitioning the space of the object feature, leading to a mixture of likelihood density functions.

3 STATE OF THE ART

Using several features, several sensors, or several appearance models, are distinct concepts. In the particle filter framework, multi-sensors models and multi-features ones often lead to similar implementations, and are therefore not always clearly distinguished in the literature. Here we use the term multi-modalities to denote these two types of data. We describe next two popular models, which are suited in most cases, dealing with several features of several sensors.

In the following, $\mathbf{x}_t \in \mathbb{X}$ denotes the hidden state of a stochastic process at time t, and $\mathbf{y}_t = (\mathbf{y}_t^1, \ldots, \mathbf{y}_t^R)$ is a vector of R components, where $\mathbf{y}_t^r$ stands for the r^{th} modality.

The first model consists in factorizing the likelihood density as a mixture model, in which each component represents a modality: $p(\mathbf{y}_t|\mathbf{x}_t) = \sum_{r=1}^R \pi_t^r p(\mathbf{y}_t^r|\mathbf{x}_t)$, with $\{\pi_t^r\}_{r=1}^R$ the “relevance probability” of the modalities ($\sum_{r=1}^R \pi_t^r = 1$), i.e. the probability that the modality r is the one which describes the state $\mathbf{x}_t$. These probabilities are either fixed by hand (Xu and Li, 2005), or adaptive but with a fixed set a possible values (Hotta, 2006).

The second model introduces confidence or reliabilitiy in a modality. It considers a conditional independence between the modalities according to the state $\mathbf{x}_t$. Confidence indices $\{\alpha_t^r\}_{r=1}^R$ are then added in a *ad hoc* way as exponents of the marginal likelihoods $p(\mathbf{y}_t^r|\mathbf{x}_t), r = 1, \ldots, R$: $p(\mathbf{y}_t|\mathbf{x}_t) = \prod_{r=1}^R p(\mathbf{y}_t^r|\mathbf{x}_t)^{\alpha_t^r}$, with $\alpha_t^r \in [0,1]$. The values α_t^r represent the confidence in the modality r. Unlike the first model, indices α_t^r are independent of each other, which facilitates their update. They can be defined using different features, one can see for example (Brasnett et al., 2007; Erdem et al., 2010).

When the appearance of the object evolves during time, for example because of luminosity or pose changes, tracking algorithms using a correlation criterion between a reference model and a candidate must update the reference model to stay robust. The implementation of a model with a changing appearance consists in updating progressively the reference model, as it has notably been proposed in (Nummiaro et al., 2002; Muñoz-Salinas et al., 2008).

Here, instead of updating the reference model, we propose a different approach, that explicitly models several components which may be related to several appearances.

All methods described in this section aim at defining adaptive weights, of probability, confidence or model update. This adaptive feature enhances the models with more flexibility and robustness. However, the update is often difficult and therefore often performed in an heuristic way, by computing the values *a posteriori* according to a defined criterion. The strategy is therefore not directly included in the particle filter framework, and delivers a single parameter set for all the particles. Hence, errors can propagate and accumulate during time, thus definitely biasing or deteriorating the reference model. This may lead to unsuitable likelihoods and penalize the tracking task.

The model we propose defines the likelihood by a mixture of densities, in which each particle is associated with a different set of weights. Hence, this strategy does not suffer from the aforementioned problem. The originality comes from the fact that a weight is related to a decomposition of the state and the observation and not to a feature or a sensor.

4 MULTIPLE MODEL LIKELIHOOD

We propose in this section to define a multiple model likelihood. We consider that an appearance is a possible representation of an object, according to a considered feature. This modeling is useful when object appearance (color, shape,...) changes during time. For example, in a $3D$ face tracking problem, one may define several components, that we call postures, for which the probabilities are computed using the orientation of the head (*front*, *profile* and *behind*), and associated with a color likelihood, conditioned by the considered posture. The appearance corresponds to the reference model used by the color likelihood. In this case, the orientation criterion defines the type of posture that the object describes. By defining the joint likelihood by a mixture density, component weights are defined using the orientation of the head, and allows integrating in an original way complementary features in the joint likelihood density.

We describe now the formalization of the proposed approach. Let $\dot{\mathbf{x}} = (\dot{\mathbf{x}}^1,\ldots,\dot{\mathbf{x}}^O)$ be a vector of O components, where $\dot{\mathbf{x}}^j$ represents the reference model (the appearance) of the object, associated with the component j denoting a posture. For example, if $j = 2$ represents the posture *behind* a face (this component being defined from some information on the orientation of the head), a color based on appearance model $\dot{\mathbf{x}}^2$ will be described by the color of the hair. The joint likelihood is given by:

$$p(\mathbf{y}_t|\mathbf{x}_t,\dot{\mathbf{x}}) = \sum_{j=1}^{O} \varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t)\right) p(\mathbf{y}_t|\mathbf{x}_t,\dot{\mathbf{x}}^j) \tag{1}$$

with $p(\mathbf{y}_t|\mathbf{x}_t,\dot{\mathbf{x}}^j)$ the likelihood of the component j and $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t)\right)$ the weight of the posture j associated with the feature $\gamma(\mathbf{x}_t,\mathbf{y}_t)$, with $\varphi^j : \mathcal{Z} \to [0,1]$ such that $\forall(\mathbf{x}_t,\mathbf{y}_t) \in \mathbb{X}\times\mathbb{Y}, \sum_{i=1}^{O}\varphi^i\left(\gamma(\mathbf{x}_t,\mathbf{y}_t)\right) = 1$, with $\mathcal{Z}$ the feature space, and $\gamma : \mathbb{X}\times\mathbb{Y} \to \mathcal{Z}$ the feature function. Index j represents the j^{th} posture of the decomposition, and $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t)\right)$ the probability of the posture j.

Observations considered for the weights and the likelihoods may be of different kinds. For example, in a $3D$ face tracking application, the posture probabilities, based on the orientation of the head, may be computed using gradient information extracted from the input image, whereas likelihoods conditioned by the appearance models may be defined by a distance between color histograms. Observations are respectively noted $\mathbf{y}_t^{App}$ and $\mathbf{y}_t^{Pos}$. Equation 1 becomes:

$$p(\mathbf{y}_t|\mathbf{x}_t,\dot{\mathbf{x}}) \simeq \sum_{j=1}^{O} \varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right) p(\mathbf{y}_t^{App}|\mathbf{x}_t,\dot{\mathbf{x}}^j) \tag{2}$$

This equation is obtained by considering the conditional independence of the weights with respect to $\mathbf{y}_t^{App}$ given $\mathbf{y}_t^{Pos}$, and the term $p(\mathbf{y}_t^{Pos}|\mathbf{x}_t,\dot{\mathbf{x}}^j,\mathbf{y}_t^{App})$ is simply ignored, since this approximation allows us to make the desired distinction between "posture" and "appearance" information, and does not necessarily require the existence of $\mathbf{y}_t^{Pos}$. An original feature of this decomposition is the conditioning of the weights with respect to $\mathbf{x}_t$ and $\mathbf{y}_t^{Pos}$. This means they are automatically determined by the current observation, and dedicated to the considered particle, which contrasts with existing methods (see Section 3).

Although the application field seems to be vast, we choose to define the weights $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)$ as to be dependent of the state $\mathbf{x}_t$ and the observation $\mathbf{y}_t$, in order to show the modeling potential of the approach. Furthermore, although these weights might be, in many applications, defined by a known closed-form γ, we are interested in the case where they are learned. When the learning database is not labeled, a clustering technique such as k-means algorithm (Dhillon et al., 2004) or an unsupervised classifier (Ben-Hur et al., 2002) can be employed. In this work, we consider a labeled learning database, where the labels are the indices characterizing the postures and are assumed to be known. The methodology consists in using a supervised classifier, here a Support Vector Machine (SVM), in the feature space $\mathcal{Z}$, which allows computing the probability of a couple $(\mathbf{x}_t,\mathbf{y}_t^{Pos})$ to belong to a class j thanks to the feature function γ. The choice of a SVM is motivated by its efficiency, its genericity, and its capability to provide a classification result in the form of probabilities, which will then be integrated in the likelihood model.

5 LEARNING USING SUPPORT VECTOR MACHINES

In the proposed model, the considered classes are the postures, and a SVM classifier is used to automatically determine membership weights to classes. Considering a SVM output interpreted as posterior probabilities (Platt, 2000), we use a strategy of pairwise coupling to compute the probabilities $p_j = \Pr(l = j|z), j = 1,\ldots,O$ (Wu et al., 2004), with O the number of classes, here the number of postures. In the model proposed in Equation 2, the value p_j corresponds to the probability to consider the j^{th} posture of the model, i.e. $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)$. We consider $\binom{O}{2}$ SVM with a learning database $\mathcal{D} = \Big\{\Big(\gamma(\mathbf{x}^{(1)},\mathbf{y}^{Pos,(1)}),l^{(1)}\Big),\ldots,\Big(\gamma(\mathbf{x}^{(M)},\mathbf{y}^{Pos,(M)}),l^{(M)}\Big)\Big\}$, with γ the feature function, and $l^{(M)}$ the label, which denotes the posture index, and so implicitly the appearance. The weights $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)$ are then given by probabilities $\{p_j\}_{j=1}^{O}$: $\forall j \in \{1,\ldots,O\}, \varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right) = \Pr(\mathbf{l} = j|\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})) = p_j$.

We can also use the membership probabilities of samples from the learning database to build appearance models $\{\dot{\mathbf{x}}^j\}_{j=1}^{O}$. They are defined as the weighted sum of the samples where the weights are the membership probabilities to the considered class:

$$\overset{\bullet}{\mathbf{x}}^j = \frac{\sum_{i=1}^M \Pr\left(\mathbf{l}=j|\gamma(\mathbf{x}^{(i)},\mathbf{y}^{Pos,(i)})\right)\eta(\mathbf{x}^{(i)})}{\sum_{i=1}^M \Pr\left(\mathbf{l}=j|\gamma(\mathbf{x}^{(i)},\mathbf{y}^{Pos,(i)})\right)} \tag{3}$$

where $\Pr\left(\mathbf{l}=j|\gamma(\mathbf{x}^{(i)},\mathbf{y}^{Pos,(i)})\right)$ is the probability to consider the posture j according to the data $\gamma(\mathbf{x}^{(i)},\mathbf{y}^{Pos,(i)})$, extracted from a pairwise coupling strategy, and η is a function characterizing an appearance, which will be formalized in Section 6.

6 EXPERIMENTS

We consider an application of mouth tracking, using one, two and three postures. The purpose of these experiments is to show the interest of using several postures into a general particle filter framework, and not to compare results to the ones obtained by mouth tracking dedicated methods. Likelihoods $\{p(\mathbf{y}_t|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)\}_{j=1}^O$ are based on color information, and use the reference histograms $\{\overset{\bullet}{\mathbf{x}}^j\}_{j=1}^O$. Experiments will use different posture weights: color histogram features with a Bhattacharyya distance, or shape features with an Euclidean or a Hausdorff distance.

We propose to use the learning and test sets coming from an annotated database freely available on the Internet[1]. This sequence contains 5000 images showing a human face with changing expressions (Figure 1). The learning database contains 1000 elements. Each of these elements is a mouth shape defined by a set of P control points, from which we extract a color histogram. Let $\mathcal{D} = \left\{\left(\gamma(z^{(1)}),l^{(1)}\right),\ldots,\left(\gamma(z^{(M)}),l^{(M)}\right)\right\}$ be M samples of the learning database, with $\gamma(z^{(i)})$ the feature function used in the SVM extracted from the i^{th} shape $z^{(i)}$.

We propose to track the mouth in a 300 image sequence (not used for the learning database contruction). For comparison, we consider three decompositions of the joint likelihood: a decomposition in one element, *mouth*, which corresponds to a classical case (i.e., no classification); a decomposition in two elements, *closed mouth* and *open mouth* (which includes elements *open mouth* and *smile*); and finally a decomposition in three elements, *closed mouth*, *open mouth*, and *smile*.

The state vector $\mathbf{x}_t = (\mathrm{x}_t,\mathrm{y}_t,\dot{\mathrm{x}}_t,\dot{\mathrm{y}}_t,\theta_t,\mathrm{a}_t)$ contains the $2D$ coordinates $(\mathrm{x}_t,\mathrm{y}_t)$ of the center of the mouth at time t, the velocity $(\dot{\mathrm{x}}_t,\dot{\mathrm{y}}_t)$, the orientation of the mouth θ_t and a set of control points $\mathrm{a}_t = (\mathrm{a}_t^1,\ldots,\mathrm{a}_t^P)$. We consider a constant velocity model. The dynamical model for the parameters $(\dot{\mathrm{x}}_t,\dot{\mathrm{y}}_t,\theta_t,\mathrm{a}_t)$ is the one proposed in (Widynski et al., 2010), which specifically allows us to, in particular, handle non rigid shape transformations.

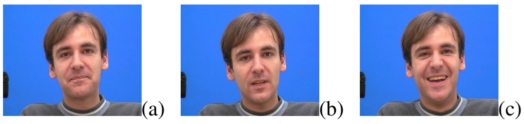

Figure 1: Images taken from the tested sequence, showing samples of three classes of posture for the mouth (a) *closed mouth*, (b) *open mouth* and (c) *smile*.

We consider a likelihood $p(\mathbf{y}_t^{App}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)$ combining color and edge information: $p(\mathbf{y}_t^{App}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j) = p(\mathbf{y}_t^{App,R}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)\,p(\mathbf{y}_t^{App,C}|\mathbf{x}_t)$, where $p(\mathbf{y}_t^{App,C}|\mathbf{x}_t)$ is an edge likelihood based on gradient values computed on the B-Spline interpolation of the control points a_t at position $(\mathrm{x}_t,\mathrm{y}_t)$ and orientation θ_t, and $p(\mathbf{y}_t^{App,R}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)$ a region likelihood, conditioned by the j^{th} reference model. The region likelihood uses a notion of distance between HSV histograms (Pérez et al., 2002). Both likelihoods are also explained in (Widynski et al., 2010). Reference histograms $\{\overset{\bullet}{\mathbf{x}}^j\}_{j=1}^O$ are computed automatically using the SVM (Equation 3), with $\gamma(\mathbf{x}^{(i)},\mathbf{y}^{Pos,(i)}) = \gamma(z^{(i)})$ and $\eta(\mathbf{x}^{(i)}) = h(z^{(i)})$, where $\gamma(z^{(i)})$ is the feature function, and $h(z^{(i)})$ the histogram extracted from the sample $z^{(i)}$ of the learning database. The joint likelihood considering O components for the color likelihood is finally written by:

$$p(\mathbf{y}_t|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}) = \left[\sum_{j=1}^O \varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)\,p(\mathbf{y}_t^{App,R}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)\right] \times p(\mathbf{y}_t^{App,C}|\mathbf{x}_t) \tag{4}$$

It only remains to define the function γ in order to construct the SVM classifier.

We consider Gaussian kernels, in order to obtain a non linear separation of the data, $K(\gamma(z^{(i)}),\gamma(z)) = \exp\left(-\frac{\langle\gamma(z^{(i)}),\gamma(z)\rangle^2}{2\sigma^2}\right)$, with $z^{(i)}$ the i^{th} shape of mouth from the learning database, z a candidate shape of mouth (or while learning, a shape $z^{(i)}$ from the database), σ^2 the fixed variance, and $\langle.\rangle$ the norm that depends on the type of feature used.

As a first criterion characterizing a component probability of the model, we consider **color histograms**. The feature function γ corresponds to a region histogram, hence $\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos}) = h(\mathring{a}_t)$, with $\mathring{a}_t$ the set of points belonging to the region described by $(\mathrm{x}_t,\mathrm{y}_t,\theta_t,\mathrm{a}_t)$. The kernel norm of the SVM is a **Bhat-**

[1] http://personalpages.manchester.ac.uk/staff/timothy.f.cootes/data/talking_face/talking_face.html

tacharyya distance.

As a second experiment, we use a **shape information** to compute component probabilities. In this case, the weight function of the posture does not depend on the observation, since we only consider a set of $2D$ points. Hence γ is written by the set of control points a_t, and $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)=\varphi^j\left(\gamma(\mathbf{x}_t)\right)=\varphi^j\left(\gamma(z)\right)=\varphi^j(\mathrm{a}_t)$. The **Euclidean norm** is used to compute the distance between two shapes.

For the last experiment, we use a **Hausdorff distance** criterion. Like for the Euclidean distance, the weight function does not depend on the observation, and the weights are $\varphi^j\left(\gamma(\mathbf{x}_t,\mathbf{y}_t^{Pos})\right)=\varphi^j(\mathrm{a}_t)$.

Mean tracking errors over 300 images with the three proposed experiments are given in Figure 2, according to the number of particles. Error at time t corresponds to 1 minus the ratio between common points of the estimated shape and the true shape and the largest area of these two objects. The benefice of a multi-appearance model is clear, since, for all criteria, the decreasing error between one and two postures is nearly equals to 25%. Between two and three postures, the difference is less important, since the partitioning is less obvious than with two postures.

Comparisons between criteria are given in Figure 3. We can see that the histogram criterion gives better results than the shape criteria, probably thanks to the quantity of information contained in such a model, which gives a robust weight estimation, even for noisy environments.

Figure 4 illustrates image results with one, two and three classes. The estimated mouth is in blue and corresponds to the Monte-Carlo expected value. The used criterion is the color histogram distance since, as we saw, it provides the best results. The difference between the results obtained with one and two/three postures is clear, as we can see on images 2 to 4. Between two and three postures, estimations on images 2 to 4 also present differences, even if they are less obvious. In particular, the orientation of the mouth estimated in the fourth image with two postures is clearly less better than with three.

Since we consider a multiple model likelihood, the number of decomposition O directly affects the computational time of the proposed approach, since the method requires to compute the likelihood densities $\{p(\mathbf{y}_t^{App}|\mathbf{x}_t,\overset{\bullet}{\mathbf{x}}^j)\}_{j=1}^{O}$. Moreover, the overall computational time also depends on the computation of the weights φ^j. In our experiments, the weights are estimated using the output of SVM, and then only requires to compute distances of the data to the support vectors, which is fast while the number of support vectors and the computational time of the distance stay reasonable.

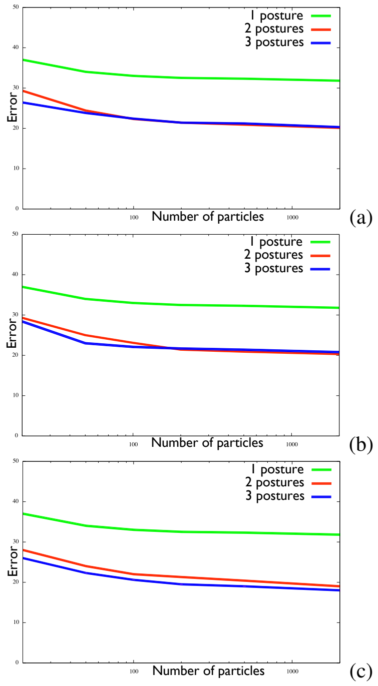

Figure 2: Mouth tracking errors obtained using as weighting criteria an information of (a) shape with an Euclidean distance, (b) shape with a Hausdorff distance, and (c) color with a Bhattacharyya distance.

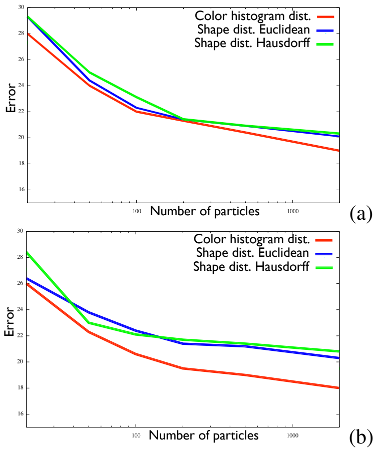

Figure 3: Mouth tracking errors obtained using different criteria to weight the likelihoods, with (a) two postures, and (b) with three postures.

7 CONCLUSIONS

We proposed in this article a multiple model likelihood. The originality of this work is twofold: first, each model is related to an object appearance, an approach that has never been employed. Secondly, the model implementation uses a decomposition of the

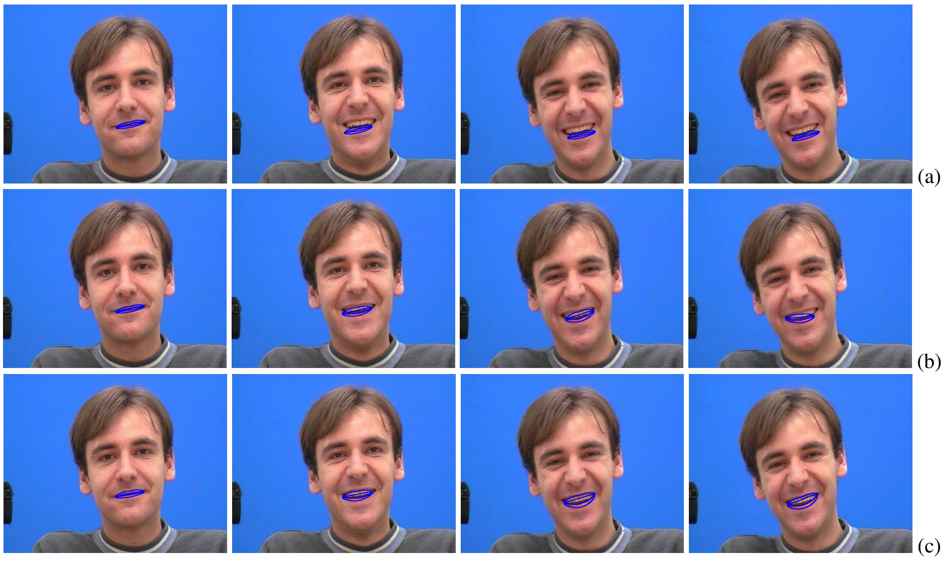

Figure 4: Captions of results obtained with (a) one, (b) two, and (c) three postures, in a test sequence of 300 images. The estimated shape is illustrated in blue.

likelihood as a mixture model, whose weights are determined according to the state and the observation at time t, so weights are unique by particle. This problem, which is a part of our contribution, allows dealing with many applications, improving the tracking robustness while using in an original way several modalities. To compute the weights, we proposed to use a SVM, that separates offline the considered features. We tested our method on a mouth tracking problem. Experiments have shown that using few components, i.e. few postures, improves significantly the results, since while refining the description model it robustifies the likelihood.

REFERENCES

Ben-Hur, A., Horn, D., Siegelmann, H., and Vapnik, V. (2002). Support vector clustering. *The Journal of Machine Learning Research*, 2:125–137.

Brasnett, P., Mihaylova, L., Bull, D., and Canagarajah, N. (2007). Sequential Monte Carlo tracking by fusing multiple cues in video sequences. *Image Vision Computing*, 25(8):1217–1227.

Dhillon, I., Guan, Y., and Kulis, B. (2004). Kernel k-means: spectral clustering and normalized cuts. In *ACM SIGKDD*, pages 551–556.

Doucet, A., De Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo methods in practice*. Springer.

Erdem, E., Dubuisson, S., and Bloch, I. (2010). Particle Filter-Based Visual Tracking by Fusing Multiple Cues with Context-Sensitive Reliabilities. Technical Report 2010D002, Télécom ParisTech.

Hotta, K. (2006). Adaptive Weighting of Local Classifiers by Particle Filter. In *ICPR*, volume 2, pages 610–613.

Maggio, E., Smeraldi, F., and Cavallaro, A. (2005). Combining colour and orientation for adaptive particle filter-based tracking. In *British Machine Vision Conference*, pages 659–668.

Muñoz-Salinas, R., Aguirre, E., García-Silvente, M., and Gonzalez, A. (2008). A multiple object tracking approach that combines colour and depth information using a confidence measure. *Pattern Recognition Letters*, 29(10):1504–1514.

Nummiaro, K., Koller-Meier, E., and Gool, L. V. (2002). Object Tracking with an Adaptive Color-Based Particle Filter. In *Symposium for Pattern Recognition of the DAGM*, pages 353–360.

Pérez, P., Hue, C., Vermaak, J., and Gangnet, M. (2002). Color-Based Probabilistic Tracking. In *ECCV*, pages 661–675.

Platt, J. C. (2000). Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *Advances in Large Margin Classifiers*, pages 61–74.

Widynski, N., Dubuisson, S., and Bloch, I. (2010). Integration of fuzzy spatial information in tracking based on particle filtering. *IEEE Transactions on Systems, Man and Cybernetics SMCB*, To Appear.

Wu, T.-F., Lin, C.-J., and Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005.

Xu, X. and Li, B. (2005). Rao-Blackwellised particle filter for tracking with application in visual surveillance. In *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 17–24.