ONLINE LEARNING OF GAUSSIAN MIXTURE MODELS: A TWO-LEVEL APPROACH

Arnaud Declercq, Justus H. Piater
Montefiore Institute, University of Liège, B-4000 Liège, Belgium
Arnaud.Declercq@ULg.ac.be, Justus.Piater@ULg.ac.be

Keywords: Online learning, Gaussian mixture model, Uncertain model.

Abstract: We present a method for incrementally learning mixture models that avoids the necessity to keep all data points around. It contains a single user-settable parameter that controls via a novel statistical criterion the trade-off between the number of mixture components and the accuracy of representing the data. A key idea is that each component of the (non-overfitting) mixture is in turn represented by an underlying mixture that represents the data very precisely (without regards to overfitting); this allows the model to be refined without sacrificing accuracy.

1 INTRODUCTION

Mixture models are used for many purposes in computer vision, e.g. to represent feature distributions or spatial relations. Given a fixed data sample, one can fit a mixture model to it using one of a variety of methods. However, in many applications, it is not possible or convenient to fix a model at the outset; one would rather learn it over time. For example, this would allow the deployment of generic recognition or tracking systems with minimal set-up effort, and training them over time on the task at hand.

However, learning and refining a mixture model incrementally is not an easy task. How is a given model to be updated when new data points arrive? If the data points underlying the current model have been discarded, then there is no general answer to this question. On the other hand, keeping all data around defeats the purpose of learning parametric models incrementally. Thus, a compromise needs to be found. We need to keep around enough information to be able to refine a model without sacrificing model accuracy, but the quantity of this information should grow much more slowly than the number of raw data points.

We address this problem by seeking to represent the data points with (1) sufficient fidelity that we can safely discard them, while at the same time (2) committing to no more predictive precision as the original data support.

These two objectives are mutually exclusive, as the former tends to overfit and the latter to underfit the data. We therefore propose a two-level representation. The first level seeks to summarize the data with high precision, allowing us to discard underlying data without significantly impairing our ability to refine the model. We therefore call it the *precise* model. The second level provides a model that represents no more detail than is supported by the underlying data and then avoids counterproductive bias in future predictions; we call it the *uncertain* model. Each uncertain component is then represented by a precise mixture model that allows it to be split appropriately when it turns out that it oversimplifies the underlying data. In the following development, we use Gaussian mixture models, but most of the principles are applicable to other types of mixture models.

2 LEVEL 1: THE PRECISE MIXTURE MODEL

When a GMM is learned from a data set of n observations, the main difficulty lies in the choice of the mixture complexity (i.e. the number of Gaussian components in the mixture). The most popular offline method is Expectation Maximization (Dempster et al., 1977) for fitting a sequence of GMMs, each with a specified number of components. The optimal model is then selected using a penalty function (Akaike, 1973; Rissanen, 1978; Schwarz, 1978). Online fitting is even more difficult; since the data points have been discarded, they cannot be used to evaluate the fitted models. The problem is then addressed through a split and merge criterion. However, these methods are either too slow for online learning (Hall and Hicks, 2005), assume that data arrives in chunks (Song and Wang, 2005) or does not guarantee the fidelity of the resulting model (Arandjelovic and Cipolla, 2005). Here we propose a new efficient online method that explicitly guarantees the accuracy of the model through a fidelity criterion.

2.1 Update of the Gaussian Mixture Model

Suppose we have already learned a precise GMM from the observations up to time t:

$$p^t(x) = \frac{\sum_{i=1}^{N} \pi_i^t g(x; \mu_i^t, C_i^t)}{\sum_{i=1}^{N} \pi_i^t} \tag{1}$$

where each Gaussian is represented by its weight π_i^t, its mean μ_i^t and its covariance C_i^t. We then receive a new data point represented by its distribution $g^t(x; \mu^t, C^t)$ and its weight π^t. C^t here represents the observation noise. As suggested by Hall and Hicks (Hall and Hicks, 2005), the new resulting GMM is computed in two steps:

1. **Concatenate** – produce a model with $N+1$ components by trivially combining the GMM and the new data into a single model.
2. **Simplify** – if possible, merge some of the Gaussians to reduce the complexity of the GMM.

The GMM resulting from the first step is simply

$$p^t(x) = \frac{\sum_{i=1}^{N} \pi_i^{t-1} g(x; \mu_i^{t-1}, C_i^{t-1}) + \pi^t g^t(x; \mu^t, C^t)}{\sum_{i=1}^{N} \pi_i^{t-1} + \pi^t} \tag{2}$$

The goal of the second step is to reduce the complexity of the model while still giving a precise description of the observations. Hall and Hicks (Hall and Hicks, 2005) propose to group the Gaussians using the Chernoff bound to detect overlapping Gaussians. Different thresholds on this bound are then tested and the most likely result is kept as the simplified GMM. Since this method is too slow for an on-line process, we use a different criterion proposed by Declercq and Piater (Declercq and Piater, 2007) for their *uncertain Gaussian model*. This model provides a quantitative estimate λ of its ability to describe the associated data that takes on a value close to 1 if the data distribution is Gaussian and near zero if it is not. This value, called the *fidelity* in the sequel, is useful to decide if we can merge two given Gaussians without drifting from the real data distribution.

2.2 Estimating the fidelity of a Gaussian model

To estimate the fidelity λ of a Gaussian model, we first need to compute the distance between this model and its corresponding data set. This is done with a method inspired from the Kolmogorov-Smirnoff test,

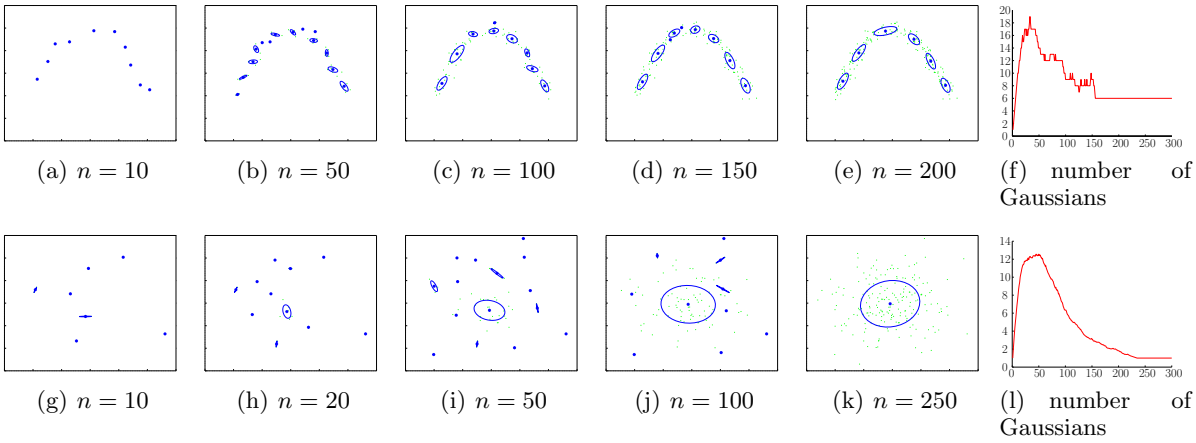
$$D = \frac{1}{|I|} \int_I \left| \hat{F}(x) - F_n(x) \right| \mathrm{d}x, \tag{3}$$

where $F_n(x)$ is the empirical cumulative distribution function of the n observations, $\hat{F}(x)$ is the corresponding cumulative Gaussian distribution, and I is the interval within which the two functions are compared. To simplify matters, the distance D is assumed to have a Gaussian distribution, which leads to the pseudo-probabilistic weighting function

$$\lambda = \mathrm{e}^{\frac{-D^2}{T_D^2}}, \tag{4}$$

where T_D is a user-settable parameter that represents the allowed deviation of observed data from Gaussianity. Whereas the sensitivity of the Kolmogorov-Smirnov test grows without bounds with n, λ provides a bounded quantification of the correspondence between the model and the data. Therefore, this criterion is more appropriate for our case since we need to estimate the correspondence of the data with the model and not their possible convergence to a Gaussian distribution.

Thus, the original data are not required anymore if we keep in memory an approximation of



(a) $n = 10$ (b) $n = 50$ (c) $n = 100$ (d) $n = 150$ (e) $n = 200$ (f) number of Gaussians

(g) $n = 10$ (h) $n = 20$ (i) $n = 50$ (j) $n = 100$ (k) $n = 250$ (l) number of Gaussians

Figure 1: Evolution of the precise mixture model with the number of data points drawn from an arc-shaped distribution.

their cumulative distribution within a given interval. Since the number of dimensions of the data space can be large, we compute the distance D for each dimension separately to keep the computational cost linear in the number of dimensions. The total distance is then simply the sum of these individual distances.

2.3 Simplification of the Gaussian Mixture Model

To decide whether two Gaussians G_i and G_j can be simplified into one, we merge them together and check if the resulting Gaussian has a fidelity λ close to one, say, exceeding a given threshold $\lambda^+_{\min} = 0.95$. The resulting Gaussian is computed using the usual equations supplemented by the combination of the cumulative density functions:

$$\pi = \pi_i + \pi_j, \tag{5}$$

$$\mu = \frac{1}{\pi}[\pi_i\mu_i + \pi_j\mu_j], \tag{6}$$

$$C = \frac{\pi_i}{\pi}\left[C_i + (\mu_i - \mu)^T(\mu_i - \mu)\right] + \frac{\pi_j}{\pi}\left[C_j + (\mu_j - \mu)^T(\mu_j - \mu)\right], \tag{7}$$

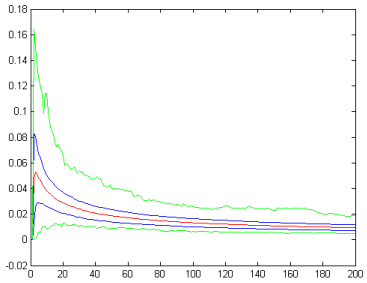
$$F(x) = \frac{1}{\pi}\left[\pi_i F_i(x) + \pi_j F_j(x)\right], \tag{8}$$

At each time, if the current GMM before the concatenation is already the simplest possible precise model of the data, the only Gaussian that can be merged with another is the one representing the new data point. If this Gaussian is successfully merged, the resulting Gaussian is, in its turn, the only available candidate for a simplification. The merging then continues iteratively until the best candidate merge drops below $\lambda^+_{\min}$. This algorithm is very fast since it corresponds, on average, to two nested loops containing only one nearest neighbour search and one merge, respectively. We only try to merge the new Gaussian with its nearest neighbour since this is most likely to provide a precise simplification. While this approach is simplistic, it gives very good results in practise while inducing only a low computational cost.

2.4 Discussion

The first row in figure 1 shows a first example of the evolution of the GMM with data points generated from an arc-shaped distribution. At the beginning, none of the Gaussians can be merged since there is clearly no Gaussian distribution that can summarize more than one observation without a significant loss of information. The complexity of the mixture thus increases by one with each new data point. As the shape of the distribution appears more clearly, the simplification step takes effect, and the number of Gaussians in the mixture decreases until it converges to a trade-off between the mixture complexity and its accuracy. This trade-off is controlled with the parameter T_D defined in equation 4. The larger its value, the farther the model is allowed to deviate from the data and the lower the complexity of the model will be. This dependence will be analyzed in detail in section 4.

Let us now consider the evolution of the model



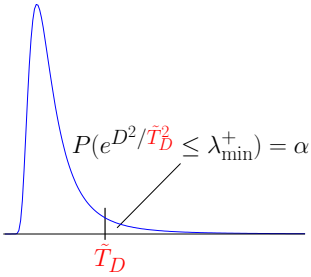


Figure 2: (a) Distribution of D for a number of observations between 1 and 200 estimated over 1000 trials. The red line represents the mean, the blue the standard deviation, and the green lines the extrema of the samples. (b) We then choose T_D such that the risk of incorrectly splitting the Gaussian is bounded by α.

for data generated from a Gaussian distribution with a large covariance. Figure 1(l) shows the mean evolution of the number of Gaussians in the mixture for a series of 50 tests, and figures 1(g)-(k) show the evolution for one of these tests. As one would expect, we first observe an explosion of the complexity of the model before it converges to a single Gaussian. This shows that the effort to faithfully represent the observations leads to gross overfitting of sparse data. Thus, our method is useful to *summarize* past observations but not to *predict* future observations. To address prediction, in the following section we propose a 2-level mixture model containing one level for precise summary of the data and one for a non-overfitted representation of the data.

3 LEVEL 2: THE UNCERTAIN MIXTURE MODEL

Let us consider again the case of a GMM learned from Gaussian-distributed data. What should be the value of the parameter T_D to guarantee that the model will always be a one-Gaussian mixture with a fidelity λ exceeding $\lambda_{\min}^+$? To answer this question we have computed the distribution of the distance D (eqn. 3) for a number of observations between 1 and 200 estimated over 1000 tests. Figure 2(a) shows the results we obtained. As expected, the variance of the distance D is very large when the number of observations is low. We then have to choose T_D such that probability of incorrectly splitting the Gaussian is bounded by a constant α. Since T_D represents a standard deviation (eqn. 4), and since empirical estimates of variance follow a χ^2 distribution, we can limit this probability to, say, $\alpha = 0.005$, by replacing T_D by an adjusted $\tilde{T}_D$ defined as

$$\tilde{T}_D^2 = \frac{N}{\chi^2_{N-1}(\alpha)} {T_D}^2, \tag{9}$$

where $\chi^2_{N-1}(\alpha)$ is the inverse of the cumulative density function of the χ^2 distribution evaluated at probability α. The new fidelity criterion is then defined by

$$\exp\left(\frac{-D^2}{\tilde{T}_D^2}\right) \geq \lambda_{\min}^+. \tag{10}$$

We would now like to express this criterion by a new fidelity criterion $\lambda_{\min}^- < \lambda_{\min}^+$. Substituting eqn. 9 yields

$$\frac{-D^2\chi^2_{N-1}(\alpha)}{NT_D^2} \geq \log \lambda_{\min}^+, \tag{11}$$

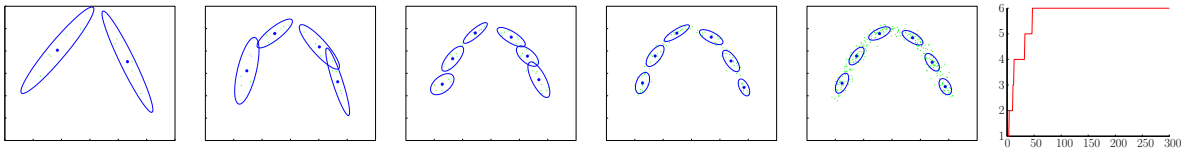
$$exp\left(\frac{-D^2}{T_D^2}\right) \geq \exp\left(\frac{N \log \lambda_{\min}^+}{\chi^2_{N-1}(\alpha)}\right). \tag{12}$$

The complexity of the imprecise GMM is then controlled by the lower threshold on the fidelity

$$\lambda_{\min}^- = \exp\left(\frac{N \log \lambda_{\min}^+}{\chi^2_{N-1}(\alpha)}\right) \tag{13}$$

that can be precomputed in a table since it only depends on $\lambda_{\min}^+$. Thanks to this new threshold, we are able to avoid the overfitting due to an explosion of the GMM complexity.

However, even if we have reduced the complexity of the model, we still face the problem of overfitting through the Gaussian model itself. Indeed, the Gaussian learned from a data set corresponds to the maximum likelihood estimate of these data and not of the complete distribution: Consider, for example, the case of a Gaussian learned from a single observation, it is clear that this Gaussian is not representative of the complete distribution.



(a) $n = 10$ (b) $n = 20$ (c) $n = 50$ (d) $n = 100$ (e) $n = 300$ (f) number of Gaussians

Figure 3: Evolution of the uncertain mixture model with the number of data points drawn from an arc-shaped distribution (compare Fig. 1).

Again, the *uncertain Gaussian model* of Declercq and Piater (Declercq and Piater, 2007), briefy summarized next, provides us with a solution to this problem by accounting for the uncertainty in the relevance due to a lack of observations.

3.1 The Uncertain Gaussian Model

The uncertain Gaussian model represents a distribution with an appropriately weighted sum of informative (Gaussian) and uninformative (uniform) components

$$q(x) = \lambda \exp\left(-\frac{1}{2}(x-\mu)^T \tilde{C}^{-1}(x-\mu)\right) + (1-\lambda) \tag{14}$$

where $\tilde{C}$ is an augmented covariance that bounds the risk of underestimating the true covariance, i.e., $P(\tilde{C} \leq C) = \alpha$, where conventionally $\alpha = 0.05$. Since empirical estimates of variance follow a χ^2 distribution,

$$\tilde{C} = \frac{n}{\chi^2_{n-1}(\alpha)}\hat{C}, \tag{15}$$

where n is the number of observations used to learn the model and $\hat{C}$ is its maximum-likelihood covariance matrix. Thanks to the new threshold λ^-_{min} and the *uncertain Gaussian model*, we are now able to learn a GMM that is kept as general as possible until there is sufficient evidence that the model can be made more specific.

The drawback of this solution is that it is now impossible to recover the data from it. For example, the data in figure 3(a) suggest that the underlying distribution is poorly represented by two Gaussians. Unfortunately, when this fact is detected, it is already too late: The observations are not in memory anymore, leaving you with a poor model that can no longer be refined. This motivates our two-level mixture model where the data are represented by the uncertain mixture model, and where each uncertain Gaussian contains a precise mixture model to describe itself. Thus, when we want to refine an uncertain Gaussian, we can split it according to its underlying mixture components.

3.2 Updating a Two-Level Gaussian Mixture Model

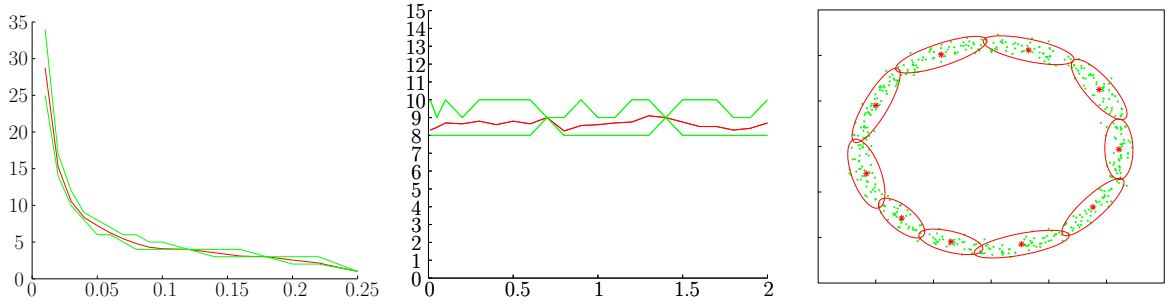
The algorithm used to update the GMM proceeds along the following steps:

1. Merge the new data point with the nearest uncertain Gaussian,
2. **if** the resulting Gaussian has a value of λ below the corresponding λ^-_{min}, replace it with two Gaussians learned from its underlying GMM with EM (Dempster et al., 1977),
3. **else** continue to merge the current uncertain Gaussian with its nearest neighbour until the resulting Gaussian has a value of λ lower than the corresponding λ^-_{min}.

Merging two uncertain Gaussians also involves merging their respective underlying mixture models. This can be done by simply summing the components from both mixtures, and using the simplification step only on the precise Gaussian that contains the new observation. Even if other precise Gaussians could possibly be merge together, we leave that for later when they merge with the current observation. This way, we distribute the computational cost through different time instants.

3.3 Discussion

Figure 3 shows an example of the evolution of the GMM with data points generated from an arc-shaped distribution. This time the complexity of the GMM only increases when there is enough evidence that the observed distribution is too complex for the current model. If we compare figure



(a) T_D (b) angular velocity (c) Example ($T_D = 0.04$)

Figure 4: Dependency of the number of Gaussians in the mixture model on (a) T_D and (b) the observations order (through the angular velocity). The red line represents the means of the 30 tests for each value, and the green lines represent the extrema.

3 with figure 1, we see that the two-level GMM and the precise mixture model converge to the same distribution. The two-level approach then provides a more stable non-overfitted model that can still become more accurate thanks to the precise model level.

4 EXPERIMENTS

4.1 Empirical Analysis of the Behaviour of the 2-Level Model

To analyze the relation between the model complexity and the only parameter T_D, we generated data from a circular distribution for different values of T_D from 0.01 to 0.25. We ran 30 tests per value of T_D and stopped each test after 500 observations. As we can see in figure 4(a), T_D provides us with a simple way to specify the desired trade-off between the model complexity and its accuracy.

Since the learning is incremental, we may wonder if the model will always converge to qualitatively the same result. We therefore performed the same experiment with $T_D = 0.04$ and with angular velocities between 0.01 and 2 rad/frame for the process that generates the observations. As shown in figure 4(b), the model complexity is nearly independent of the order of the observations.

4.2 A Vision Application

Our method provides an under-fitted probability density estimation of the partially observed distribution. It can then be used to predict future observations without exerting strong counterproductive bias. A possible application of our method is then the online learning of an object model with the immediate objective of improving the tracking of this object. This idea was tested by Declercq and Piater in the context of the simultaneous learning and tracking of a visual feature graph models (Declercq and Piater, 2007). The idea is to incrementally learn the relations between a set of tracked features and to use those incompletely learned relations to improve the tracking of the features. While the uncertain model of Declercq and Piater (2007) is limited to rigid relations, our present model is able to describe any relation that can be represented with a Gaussian mixture model. Figure 5 shows an example of the learning of the articulated relation existing between an upper arm and a forearm. The method is first tested with $T_D = 0.04$ and a learning procedure that uses all frames to update the model (row 1). The same procedure is then tested using only one in ten frames (row 2). As the figure shows, the resulting model is not influenced by this difference in the data set (except, of course, in the difference of covariances due to a difference of evidence accumulation). The third row shows the result for a smaller value of T_D which corresponds as expected to a mixture with more Gaussians.

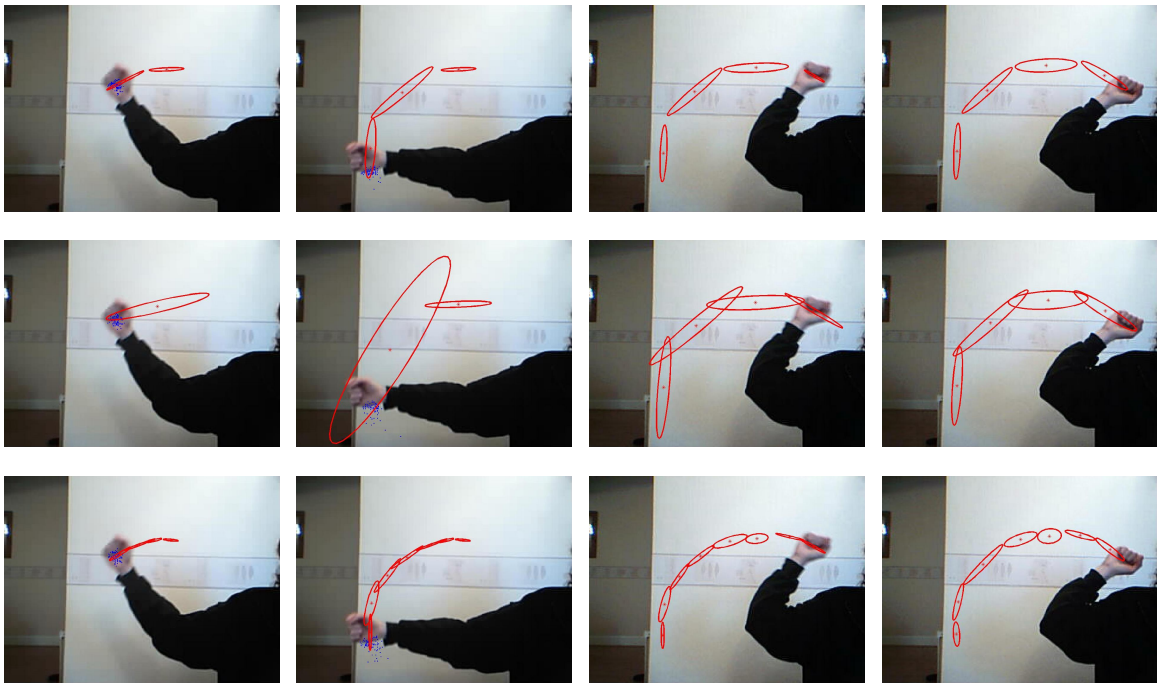


Figure 5: Evolution of a model learned from a webcam. First row: $T_D = 0.04$. Second row: $T_D = 0.04$ but the model is only learned from every tenth frame. Third row: $T_D = 0.02$. In each row, the frames no. 70, 110, 220 and 450 are shown.

5 CONCLUSION

We presented a method for incrementally learning a Gaussian mixture model based on a new criterion for splitting and merging mixture components. This criterion depends on a single user-settable parameter that allows easy tuning of the trade-off between the complexity and the accuracy of the mixture model. Our two-level approach provides a solution to the overfitting problem of small data sets without any compromise on the model accuracy. As more data arrive, the mixture complexity can be increased without any propagation of errors due to a previously underfitted model. As we have demonstrated empirically, this method is nearly independent of the order in which the data are observed.

ACKNOWLEDGEMENTS

This work is supported by a grant from the Belgian National Fund for Research in Industry and Agriculture (FRIA) to A. Declercq and by the EU Cognitive Systems project PACO-PLUS (IST-FP6-IP-027657).

REFERENCES

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *Second International Symposium on Information Theory.*

Arandjelovic, O. and Cipolla, R. (2005). Incremental learning of temporally-coherent gaussian mixture models. *BMVC.*

Declercq, A. and Piater, J. H. (2007). On-line simultaneous learning and tracking of visual feature graphs. *Online Learning for Classification Workshop, CVPR'07.*

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likekihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39:1–38.

Hall, P. and Hicks, Y. A. (2005). A method to add gaussian mixture models. *Tech. Report, University of Bath.*

Rissanen, J. (1978). Modeling by shortest data description. *Automatica, vol. 14, pp. 465-471.*

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, pages 6:461–464.

Song, M. and Wang, H. (2005). Highly efficient incremental estimation of gaussian mixture models for online data stream clustering. *Intelligent Computing: Theory and Application.*