Annales Univ. Sci. Budapest., Sect. Comp. 31 (2009) 93-105

IMAGE RETRIEVAL USING GAUSSIAN MIXTURE MODELS

Zs. Robotka and A. Zempléni

(Budapest, Hungary)

Abstract. In this paper we describe the development of an image retrieval system which is able to browse, cluster and classify large digital image databases. This work was motivated by the projects of the Visualisation Centre of the Eötvös Loránd University, where several such datasets are readily available for processing. The system's functions are based on a Gaussian Mixture Model (GMM) representation of the images. Image matching is done by matching the representations with a distance measure based on the approximation of the Kullback-Leibler divergence. The GMMs are estimated with an improved Expectation Maximization (EM) algorithm that avoids convergence to the boundary of the parameter space. Without this improvement the method is inefficient in our case because it converges to singular solutions in most of the cases.

1. Introduction

In our days very large collections of images need to be processed, such as photo collections on the Internet or geospatial image databases. Image retrieval systems may be used for searching and indexing these large digital image databases. Content-based Image Retrieval (CIR) aims at indexing images by automatic description which only depends on their objective visual content (color, shapes, texture, etc.). Our aim was to find a content-based image matching method that we can use in image searching and clustering tasks.

The main issue of image matching is the high dimensionality of the feature space. So the goal is to find a low dimensional representation of images that

is low enough for fast processing but still contains the substantive information about the image. There are two main phases in image representation as you can see on Figure 1. The first phase is to choose the representation space and the second is to define an appropriate distance measure in that space.

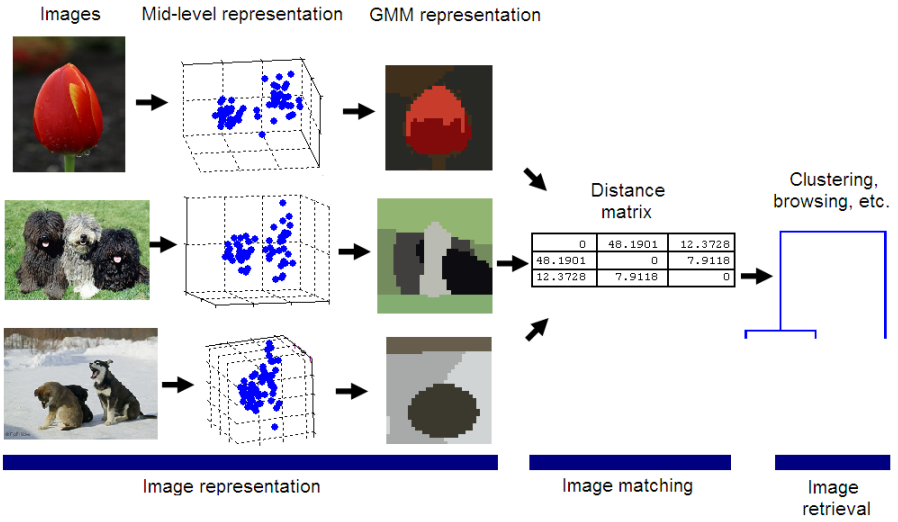

Figure 1. The image retrieval process: GMM representation of the images computed via a mid-level representation. The image matching is based on the matching of the GMMs, the retrieval tasks (browsing, clustering, etc.) are based on the distance matrix.

In our work images are represented by Gaussian Mixture Models (GMMs) after the *blobworld* method introduced by Carson et al. [1]. Each component of the mixture represents one region with similar color and texture (*blob*) of the image. Despite of using some global features this method allows to represent images with the objects found on them. We do not recognize what type of an object it is, but the representation contains the information that - for example - there is a shiny, white, longish object on the left. In the above mentioned paper Carson et al. [1] used the well known Expectation Maximization (EM) algorithm for fitting Gaussian Mixtures for determining a representation of the images. This is a powerful method even in its original form, but it has problems with the determination of the number of components and its convergence properties are not always satisfactory. To handle these problems we used a modified EM algorithm proposed by Figueiredo and Jain [3] and also made some improvements ourselves that is introduced in Section 3.2.

After choosing the image representation the second phase is the definition of

the distance measure in the representation space. The Kullback-Leibler (KL) divergence is a well-known dissimilarity measure of densities, thus it can be used as a distance measure of GMMs. Since there is no closed form expression for the KL-divergence between two GMMs, computing this distance can be done by using Monte-Carlo simulations, but they are very time-consuming. Goldberger et al. [4] introduced a new Matching Based Approximation of the KL distance which we also use.

The rest of the paper is organized as follows. Section 2 reviews the theoretical background of GMMs and the EM algorithm. Section 3 describes image representations with GMMs and the modified EM algorithm. The proposed image matching method is presented in Section 4. In Section 5 we present our results with a software we developed and Section 6 gives some conclusions.

2. Background

2.1. Gaussian Mixture Models

Formally we say that a d dimensional random variable $Y = [Y_1, \ldots, Y_d]^T$ follows a k component mixture distribution if its probability function can be written in the following form:

$$p(y \mid \mathbf{\Theta}) = \sum_{m=1}^{k} \alpha_m p(y \mid \Theta_m),$$

where

- $y = [y_1, \ldots, y_d]^T$ is one particular sample of Y,
- p is a parametric family of d dimensional distributions,
- $\alpha = \alpha_1, \ldots, \alpha_k$ is a discrete distribution,
- Θ_m is the set of the parameters of the mth component,
- $\mathbf{\Theta} = \{\Theta_1, \ldots, \Theta_k, \alpha_1, \ldots, \alpha_k\}$ is the complete set of parameters fully characterizing the mixture.

GMMs are mixtures where components are d dimensional Gaussians.

Having n independent samples of the mixture distibution $\mathcal{Y} = \{y^{(1)}, \ldots, y^{(n)}\}$ the log-likelihood funcion has the following form:

$$\mathbf{l}(\mathbf{\Theta}) = \log p(\mathcal{Y} \mid \mathbf{\Theta}) = \log \prod_{i=1}^{n} p(y^{(i)} \mid \mathbf{\Theta}) = \sum_{i=1}^{n} \log \sum_{m=1}^{k} \alpha_m p(y^{(i)} \mid \Theta_m).$$

So the maximum likelihood (ML) estimation is

$$\hat{\mathbf{\Theta}}_{ML} = \arg\max_{\mathbf{\Theta}} \big\{ \log p(\mathcal{Y} \mid \mathbf{\Theta}) \big\}.$$

2.2. The EM algorithm

Maximization of the log-likelihood $\mathbf{l}(\mathbf{\Theta})$ can be efficiently carried out by the Expectation Maximization (EM) algorithm. EM is a widely used method for estimating the parameter set of models using incomplete data. (Markov chain Monte-Carlo method can also be used for this task [7], but it is computationally demanding.) The missing part of the data is a set of labels $\mathcal{Z} = (z^{(1)}, \ldots, z^{(n)})$ associated with the elements of the samples. These missing variables are k dimensional $z^{(i)} = (z_1^{(i)}, \ldots, z_k^{(i)})$ where $z_m^{(i)}$ indicates whether the ith sample is generated by th mth component, in this case $z_l^{(i)} = 0$ for $l \neq m$ and $z_m^{(i)} = 1$. In the presence of $\mathcal{Y}$ and $\mathcal{Z}$ the (complete-data) log-likelihood function can be written as

$$\log p(\mathcal{Y}, \mathcal{Z} \mid \mathbf{\Theta}) = \sum_{i=1}^{n} \sum_{m=1}^{k} z_m^{(i)} \log \big[\alpha_m p(y^{(i)} \mid \Theta_m) \big].$$

The EM algorithm produces a sequence of estimates $\hat{\mathbf{\Theta}}^{(t)}, k = 1, 2, \ldots$ by alternatively applying the following two steps:

- **E-step:** This step finds the expected value (EV) of the complete-data log-likelihood $\log p(\mathcal{Y}, \mathcal{Z} \mid \mathbf{\Theta})$ with respect to the unknown data $\mathcal{Z}$ given the observed data $\mathcal{Y}$ and the current parameter estimates $\mathbf{\Theta^{(k)}}$. Since the log-likelihood is linear in $\mathcal{Z}$ we only have to compute the conditional expectation of the missing variables: $\mathcal{W} = E\Big[\mathcal{Z} \mid \mathcal{Y}, \hat{\mathbf{\Theta}}^{(t)} \Big]$.
  Plugging this into the log-likelihood, the result - also called the Q-function - is the following:

  $$Q(\mathbf{\Theta}, \hat{\mathbf{\Theta}}^{(t)}) = E\Big[\log p(\mathcal{Y}, \mathcal{Z} \mid \hat{\mathbf{\Theta}}^{(t)}) \mid \mathcal{Y}, \hat{\mathbf{\Theta}}^{(t)} \Big]$$

  or more simply

  $$Q(\mathbf{\Theta}, \hat{\mathbf{\Theta}}^{(t)}) = \log p(\mathcal{Y}, \mathcal{W} \mid \mathbf{\Theta}).$$

- **M-step:** This step of the EM algorithm is to maximize the expectation we computed in the first step. That is, we find

  $$\hat{\mathbf{\Theta}}^{(t+1)} = \arg\max_{\mathbf{\Theta}} Q(\mathbf{\Theta}, \hat{\mathbf{\Theta}}^{(t)}).$$

The convergence properties of the EM algorithm have been discussed in [11]. It is well known that each iteration increases the log-likelihood function. One can find an elegant proof of this theorem in the paper of Chretien et al. [2]. Theoretically, the EM is guaranteed to converge to a (unfortunately) local maxima of the likelihood function at a relatively fast convergence rate. However, in practice, the algorithm frequently fails due to numerical difficulties, because it converges to the boundary of the parameter space. In the case of GMMs this means that a component's diameter is shirnking towards zero or its covariance matrix becomes computationally singular.

2.3. The Kullback-Leibler divergence

The Kullback-Leibler (KL) divergence (also known as information divergence or relative entropy) is a well-known measure of the difference between two distributions. If P and Q are continuous random variables and p and q are their density functions, the KL-divergence is defined to be

$$D(P \mid\mid Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx.$$

Although the KL-divergence is not a metric because it is not symmetric and moreover, it does not satisfy the triangle inequality it still has some good properties for being used as a dissimilarity measure. Firstly, it is non-negative (called Gibbs inequality)

$$D(P \mid\mid Q) \geq 0$$

and it is zero if and only if $P \equiv Q$. Secondly, one of its many motivations from information theory is that KL divergence can be interpreted as the needed extra message-length per datum for sending messages distributed as Q, if the messages are encoded using a code that is optimal for distribution P.

3. Image representation with Gaussian Mixture Models

The representation of the pictures has two phases as one can see on Figure 1. The first is the transformation from pixel representation of the image to a set of low dimensional data points (called mid-level representation). The second is fitting GMMs to these data points with the EM algorithm. In this section we focus on these phases.

3.1. Feature space

Firstly, we represented the images as a set of pixels with attributes. In our work we chose the color features and the position of the pixels, but other features (for example texture) can be adopted, too. Like the earlier works ([1], [4]) we extracted the color features into the (L, a, b) color space, which was shown to be approximately perceptually uniform, thus distances in this space are meaningful [10]. So the images were represented with a set of data points in a five dimensional (three for color and two for position) feature space.

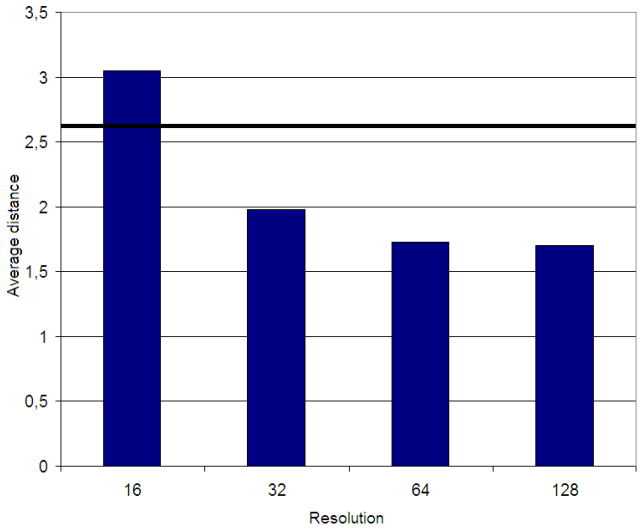

Figure 2. Average distance of GMMs based on different resolutions from the GMMs based on 256×256 resolution.

Another issue is choosing the resolution, respectively choosing the number of data points in the feature space. It is an important question since the running time of the EM algorithm is linear in the data points, so it is quadratic in the sideresolutions if squared images are used. In our experiments we got that the GMMs found with a relatively small resolution (32×32 pixels) are very similar to the GMMs found with a high resolution (256×256 pixels). On Figure 2 one can see the average distance of the GMMs based on different resolutions from the GMMs based on a high resolution (256×256 pixels). The line represents the minimum distance (2.647) between GMMs of different images in our database. These results give that in this case for clustering purposes a grid of 32×32 was fine enough.

3.2. Improved EM algorithm

As we mentioned in Section 2.2 the EM algorithm frequently fails due to numerical difficulties. The problems can be originated in two reasons. The first is the shrinking of the components. This can be easily handled by adding component annihilation to the EM algorithm based on the work of Figueiredo et al. [3]. This method also resolves the model selection problem (choosing the number of components). The second reason is the convergence to singular covariance matrices, which we resolved by some modifications of the algorithm.

EM with component annihilation. Different information criterions have been used for choosing the number of components of a GMM. Without entering into the details, applying the MML (Minimum Message Length) criterion [9] led to the objective function

$$\mathcal{L}(\Theta, \mathcal{Y}) = \frac{N}{2} \sum_{m:\alpha_m>0} \log(n\alpha_m) + \frac{k_{nz}}{2}\left(1 + \log\frac{n}{12}\right) - \log p(\mathcal{Y} \mid \boldsymbol{\Theta}), \tag{3.1}$$

where

- n is the number of data points,
- $k_{nz} = \sum_{m:\alpha_m>0} 1$,
- N is the number of parameters specifying each component.

The Θ with the lowest $\mathcal{L}$ is chosen. If an α_m becomes smaller than $\frac{N}{2n}$, the corresponding component can be annihilated by setting α_m to zero because it decreases the $\mathcal{L}$ function. ($\alpha_m = 0$ means the mth component is not supported by any data points, and the mixture is equivalent to a mixture with $k-1$ components.)

Now EM can be applied starting with a large k and using a modified M step (we annihilate components with low support in every iteration). In this way the problems about component shrinking can be resolved.

M-step with constraints. Using the EM algorithm with component annihilation in our experiments using 32×32 resolution and 8 components more than 91% of the cases led to a singular covariance matrix (Table 1). This problem occurs in many other applications of GMMs such as speaker recognition [6] or signature verification [8], where the problem is usually neglected by using diagonal covariance matrices. We found that this is too strong to assume and it decreases the algorithm capabilities. So we modified the algorithm to handle it. The main clue is putting a constraint to the eigenvalues of the covariance matrix

in the M step. Putting these constraints to the maximization we achieved that we got non-singular solution in all of the cases.

Resolution	Number of components	Frequency of singular solutions
32×32	4	72.7%
	8	91.3%
64×64	4	43.8%
	8	61.2%

Table 1. The frequency of cases where EM led to singular solution.

3.3. Visualization of the GMMs

One can see the visualization of the GMM representation on Figure 3. Firstly the components are represented by a unicolored region (blob). We projected the five dimensional components into two dimensional space (including the positional dimensions: x,y) and assigned each pixel of the original image to the most probable - the component in which it has the largest likelihood - two dimensional Gaussian. The color of the region is computed from the mean of the corresponding data points. Thats why the blobs are not ellipsoids, because the component borders are not isolines but pixels having equal likelihood in two - or more - components. Note that this visualization method hides the fact that the components are in five dimensional space and they are containing more particular information about the image.

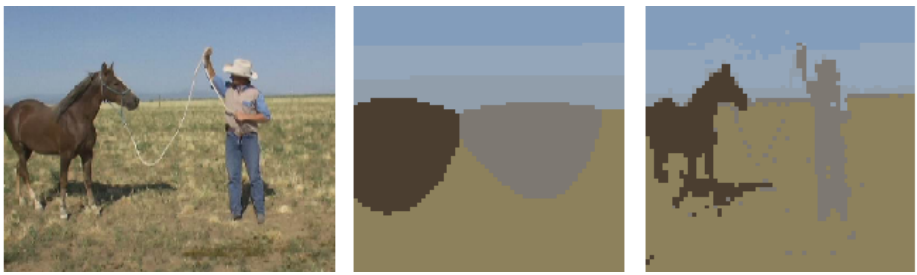

Figure 3. The vizualization of the GMM representation: original picture (left), blob vizualization (center), probabilistic image segmentation (right).

Secondly, using the suggested model each pixel of the original image can be assigned to the most probable five dimensional Gaussian, too, providing for a probabilistic image segmentation (right).

4. Image matching

The GMM representation of the images provides a good low dimensional representation, so the next phase is to define a dissimilarity measure between the mixtures. In this section we present the distance measure we used, based on the KL-divergence introduced in Section 2.3.

4.1. Gaussian Mixture Matching

Although there is no closed form expression for the KL-divergence between two GMMs, there is an analytical way to compute the KL-divergence between each pair of components. The matching-based approximation introduced by Goldberger et.al [4] utilizes a matching of the components of the mixes and aggregates the divergence of the matched components.

Let $f(x) = \sum_{i=1}^{k} \alpha_i f_i(x)$ and $g(x) = \sum_{i=1}^{k} \beta_i g_i(x)$ be two mixtures, where f_i and g_i are continuous densities, and $\alpha = \alpha_1, \ldots, \alpha_k$ and $\beta = \beta_1, \ldots, \beta_k$ are discrete distributions. Without entering into the details a natural approximation of the KL-divergence of the mixes is

$$D(f||g) = \sum_{i=1}^{k} \alpha_i \int f_i \log f - \sum_{i=1}^{k} \alpha_i \int f_i \log g \approx$$

$$\approx \sum_{i=1}^{k} \alpha_i \int f_i \log \alpha_i f_i - \sum_{i=1}^{k} \alpha_i \max_j \int f_i \log \beta_j g_j =$$

$$= \sum_{i=1}^{k} \alpha_i \min_j \left(D(f_i||g_j) + \log \frac{\alpha_i}{\beta_j} \right).$$

One can read more about this approximation in the publication of Goldberger et al. [4].

4.2. Symmetrization

The proposed image matching method provides a dissimilarity measure for the last phase of the image retrieval and clustering tasks as one can see on Figure 1. Since the most clustering algorithms require symmetric distances, we symmetrized the approximated KL-divergence with the resistor average [5]

$$R(f, g) = \frac{1}{\frac{1}{D(f||g)} + \frac{1}{D(g||f)}},$$

but other methods (arithmetical average, geometrical average) can be used, too.

5. Results

Our aim was to develop an image retrieval tool that is able to search, cluster and classify images. We chose GMM representation after some previous works [1], [4], but we noticed that this method does not perform well because of the convergence properties of the EM. Our main result is that we adopted the component annihilation algorithm of Figueiredo et al. [3] and improved it by putting constraints to the M step, so the *blobworld* method became stable and adaptable.

The developed tool is able to manage the GMM building with different parameters, and visualize the GMMs as one can see on Figure 4. The GMM building is very fast, 1000 images can be processed in about 3 minutes. It has functions for finding similar images, clustering (Figure 5 and Figure 6) and classify images - if some of them is labeled.

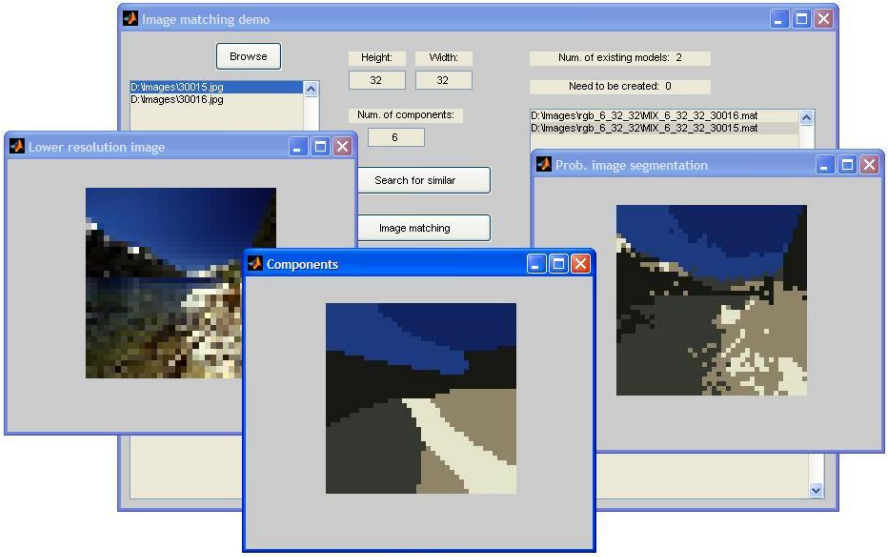

Figure 4. The visualization functions of the software

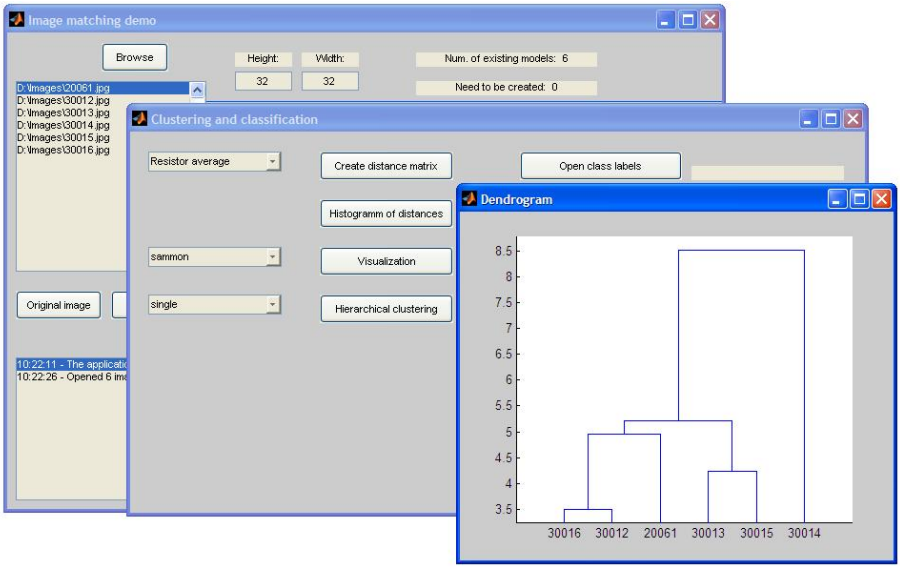

Figure 5. Hierarchical clustering with the software

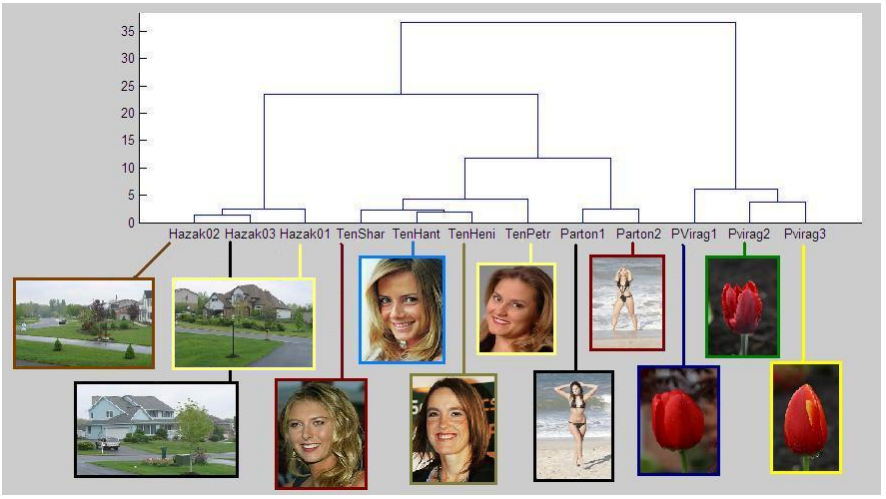

Figure 6. A dendrogram with the corresponding images

6. Conclusion

We have created a software, which is able to search, cluster and classify large data sets of different images. This work was motivated by the projects of the Visualisation Centre of the Eötvös Loránd University, where several such datasets are readily available and the use of our software would allow a more effective testing and exact evaluation of the different transfer technologies.

Future research is needed in order to find the properties of the constrained EM algorithm we suggested in Section 3.2. Further results will be presented at the 7th Annual Conference of ENIBIS, where we show the same image representation method for a data set of 10000 images, combined with genetic clustering algorithms.

7. Acknowledgements

This research was supported by the Hungarian Jedlik Ányos project NKFP2-00031/2005 (Optimization of image transfer in communication networks).

References

[1] **Carson C., Belongie S., Greenspan H. and Malik J.** Blobworld: Image segmentation using expectation-maximization and its application to image querying, *IEEE Trans. Pattern Analysis and Machine Intelligence,* **24** (8) (2002), 1026-1038.

[2] **Chretien S. and Hero A.,** Kullback proximal algorithms for maximum likelihood estimation, *IEEE Trans. on Information Theory,* **46** (5) (2000), 1800-1810.

[3] **Figueiredo M.A.T. and Jain A.K.,** Unsupervised learning of finite mixture models, *IEEE Trans. Pattern Analysis and Machine Intelligence,* **24** (3) (2002), 381-396.

[4] **Goldberger J., Gordon S. and Greenspan H.,** An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mix, *International Conference on Computer Vision (ICCV),* 2003, 487-493.

[5] **Johnson D. and Sinanovic S.,** Symmetrizing the Kullback-Leibler distance, *IEEE Trans. on Information Theory* (submitted) `http://cmc.rice.edu/docs/docs/Joh2001Mar1Symmetrizi.pdf`, 2002.

[6] **Liu L. and He J.,** On the use of orthogonal GMM in speaker recognition, *IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP'99,* 1999, 45-49.

[7] **Richardson S. and Green P.,** On Bayesian analysis of mixtures with unknown number of components, *Journal of the Royal Statistical Society. Series B,* **59** (4) (1997), 731-758.

[8] **Richiardi J. and Drygajlo A.,** Gaussian mixture models for on-line signature verification, *Int. Multimedia Conf., Proc. 2003 ACM SIGMM Workshop on Biometrics Methods and Applications,* 2003, 115-122.

[9] **Oliver J.J., Baxter R.A. and Wallace C.S.,** Unsupervised learning using MML, *Proc. of the Thirteenth Int. Conf. (ICML 96),* Morgan Kaufmann Publishers, 1996, 364-372.

[10] **Wyszecki G. and Stiles W.** *Color science, concepts and methods, quantitative data and formulae,* Wiley, 1982.

[11] **Xu L. and Jordan M.I.,** On convergence properties of the EM algorithm for Gaussian mixtures, *Neural Comput.,* **8** (1996), 129-151.

Zs. Robotka and A. Zempléni
Department of Probability Theory and Statistics
Eötvös Loránd University
Pázmány Péter sétány 1/C.
H-1117 Budapest, Hungary
`zsolt.robotka@gmail.com, zempleni@ludens.elte.hu`