An Analytic Distance Metric for Gaussian Mixture Models with Application in Image Retrieval

G. Sfikas, C. Constantinopoulos*, A. Likas, and N.P. Galatsanos

Department of Computer Science,
University of Ioannina
Ioannina, Greece GR 45110
{sfikas, ccostas, arly, galatsanos}@cs.uoi.gr

Abstract. In this paper we propose a new distance metric for probability density functions (PDF). The main advantage of this metric is that unlike the popular Kullback-Liebler (KL) divergence it can be computed in closed form when the PDFs are modeled as Gaussian Mixtures (GM). The application in mind for this metric is histogram based image retrieval. We experimentally show that in an image retrieval scenario the proposed metric provides as good results as the KL divergence at a fraction of the computational cost. This metric is also compared to a Bhattacharyya-based distance metric that can be computed in closed form for GMs and is found to produce better results.

1 Introduction

The increasing supply of cheap storage space in the past few years has led to multimedia databases with ever-increasing size. In this paper we consider the case of content-based image retrieval (CBIR) [3]. That means that the query is made using a sample image, and we would like the CBIR system to give us the images that resemble the most our sample-query. A common approach to CBIR is through the computation of image *feature histograms* that are subsequently modeled using probability density function (PDF) models. Then, the PDF corresponding to each image in the database is compared with that of the query image, and the images closest to the query are returned to the user as the query result. The final step suggests that we must use some distance metric to compare PDFs. There is no universally accepted such distance metric; two commonly used metrics for measuring PDF distances is the *Kullback-Liebler* divergence and the *Bhattacharyya* distance [4]. In this paper, we explore a new distance metric that leads to an analytical formula in the case where the probability density functions correspond to Gaussian Mixtures.

It is obvious that the distance metric we choose to employ is of major importance for the performance of the CBIR system. It is evident that the query results are explicitly affected by the metric used. Also, a computationally demanding metric can slow

* This research was funded by the program "Heraklitos" of the Operational Program for Education and Initial Vocational Training of the Hellenic Ministry of Education under the 3rd Community Support Framework and the European Social Fund.

W. Duch et al. (Eds.): ICANN 2005, LNCS 3697, pp. 835 – 840, 2005.

down considerably the whole retrieval process, since the sample image must be compared with every image in the database.

2 GMM Modeling and PDF Distance Metrics

At first, we need as we noted to construct a *feature histogram* for each image in the database, as shown in [2] for color features. There are a number of reasons, though, that feature histograms are not the best choice in the context of image retrieval and it is preferable to model the feature data using parametric probability density function models, like for example Gaussian mixture models (GMM).

Consider the case where we choose color as the appropriate feature and construct color histograms. It is well-known that color histograms are sensitive to noise interference like lighting intensity changes or quantization errors ("binning problem"). Also, the number of bins in a histogram grows exponentially with the number of feature components ("curse of dimensionality"). These problems, which apply in feature histograms in general, can be solved by modeling the histogram using a probability density function model.

A good way to model probability density functions (PDF) is assuming that the target distribution is a Finite Mixture Model [1]. A commonly used type of mixture model is the Gaussian Mixture Model (GMM). This model represents a PDF as

$$p(x) = \sum_{j=1}^{K} \pi_j N(x : \mu_j, \Sigma_j) \tag{1}$$

where K stands for the number of Gaussian kernels mixed, π_j are the mixing weights and μ_j, Σ_j are the mean vector and the covariance matrix of Gaussian kernel *j*. GMMs can be trained easily with an algorithm such as EM (Expectation – Maximization) [1].

So we come to the point where the sample image used for the query and the images in the database have their feature histogram and Gaussian Mixture Model been generated. The final step is to compare the GMM of the sample image with the GMMs of the stored images in order to decide which images are the closest to the sample. Therefore, we need a way to calculate a distance metric between PDFs.

A common way to measure the distance between two PDFs *p(x)* and *p'(x)*, is the Kullback-Liebler divergence [4]:

$$KL(p \,\|\, p') = \int p(x) \ln \frac{p(x)}{p'(x)} dx \cdot$$

Notice that $KL(p \,\|\, p')$ is not necessarily equal to $KL(p' \,\|\, p)$. Thus, it is more reasonable to use a symmetric version of the Kullback-Liebler divergence:

$$SKL(p, p') = \left| \frac{1}{2} \int p(x) \ln \frac{p(x)}{p'(x)} dx + \frac{1}{2} \int p'(x) \ln \frac{p'(x)}{p(x)} dx \right| \tag{2}$$

where SKL stands for Symmetric Kullback-Liebler. The absolute value is taken in order for the metric to have distance properties. Since the SKL metric cannot be computed in closed form, we have to resort to a Monte-Carlo approximation based on the

formula $\int f(x)p(x)dx \to (N)^{-1}\sum_{i=1}^{N} f(x_i)$ as $N \to \infty$,where the samples x_i are assumed to be drawn from $p(x)$. Thus, SKL can be computed as:

$$SKL_{MK}(p,p') = \left| \frac{1}{2N}\sum_{x\sim p} \ln p(x) - \frac{1}{2N}\sum_{x\sim p} \ln p'(x) + \frac{1}{2N}\sum_{x\sim p'} \ln p(x) - \frac{1}{2N}\sum_{x\sim p'} \ln p'(x) \right|$$

where *N* is the number of data samples generated from the *p(x)* and *p'(x)*. Note that the above formula can be very computationally demanding, since it consists of sums over *4xN* elements – *N* must be large if we want to get an accurate result. Also, when the dimensionality of the *x* vectors is high, things get worse, since *N* must be even larger.

3 The PDF Distance Metric

We can take advantage of the fact that the PDFs we need to compare are Gaussian Mixtures, not *any* distributions. A GMM can be described only by the mean and covariance of its Gaussian kernels, plus the mixing weights. This suggests that we might construct a distance metric using the values *μ, Σ, π* for each one of the two distributions compared, thus creating a fast to compute metric.

The metric we considered in its general form is the following [5]:

$$C2(p,p') = -\log\left[\frac{2\int p(x)p'(x)dx}{\int p^2(x) + p'^2(x)dx}\right] \tag{3}$$

This metric is zero when $p(x)$ and $p'(x)$ are equal and is symmetric and positive. In the case where the PDFs compared are GM, eq. (3) yields

$$C2(p,p') = -\log\left[\frac{2\sum_{i,j}\pi_i\pi_j' \sqrt{\frac{|V_{ij}|}{e^{k_{ij}}|\Sigma_i||\Sigma_j'|}}}{\sum_{i,j}\left\{\pi_i\pi_j\sqrt{\frac{|V_{ij}|}{e^{k_{ij}}|\Sigma_i||\Sigma_j|}}\right\} + \sum_{i,j}\left\{\pi'_i\pi'_j\sqrt{\frac{|V_{ij}|}{e^{k_{ij}}|\Sigma'_i||\Sigma'_j|}}\right\}}\right] \tag{4}$$

where

$$V_{ij} = \left(\Sigma_i^{-1} + \Sigma'^{-1}_j\right)^{-1},$$

$$k_{ij} = \mu_i^T\Sigma_i^{-1}(\mu_i - \mu'_j) + \mu'^T_j\Sigma'^{-1}_j(\mu'_j - \mu_i),$$

π, π' the mixing weights, i and j are indexes on the gaussian kernels, and, finally, μ, Σ and μ', Σ' are mean and covariance matrices for the kernels of the Gaussian mixtures $p(x)$ and $p'(x)$ respectively.

4 Numerical Experiments

To test the effectiveness of the above distance metric we consider an image database consisting of pictures that can be classified in 5 categories, according to their theme. These are: 1) Pictures of cherry trees ("Cherries"), 2) Pictures of bushes and trees in general ("Arborgreens"), 3) Pictures of a seaside village in a rainy day ("Cannon-beach"), 4) Pictures in a university campus (outdoor) in Fall ("Campus in Fall") and 5) Shots of a rugby game ("Football"). Forty 700x500 images per class were considered.

We have generated a Gaussian mixture model for each of the images, using color (RGB space) as the feature vector. The number of the Gaussian components for every GMM was empirically chosen to be five and the Gaussian mixture models were trained using the EM algorithm. In the case of an actual image retrieval query, we would need to compare the GMM of the sample image with every other model in the database. Instead, in this experiment we compare the models of every image with one another, once for each of three distance metrics, which are 1) Symmetric Kullback-Liebler (with 4096 samples per image), 2) a Bhattacharyya based distance for GMMs and 3) the proposed C2 distance. The times required to compute all distances among the five sets are 154,39 sec., 674,28 sec. and 33161,62 sec for Bhattacharyya-based

Table 1. Average distance among classes for three distance metrics

	Cherr	**Arbor**	**Football**	**Cann**	**Campus**
Cherries	1	1,12	1,12	1,43	1,67
Arbor	2,84	1	1,87	2,57	2,94
Football	4,96	3,26	1	6,98	3,87
Cann	1,88	1,32	2,07	1	2,35
Campus	2,94	2,03	1,54	3,15	1

(a) Average SKL distances

	Cherr	**Arbor**	**Football**	**Cann**	**Campus**
Cherr	1	1,55	1,08	1,28	1,91
Arbor	1,89	1,05	1	2,78	1,92
Football	1,66	1,25	1	2,34	1,76
Cann	1	1,77	1,19	1,02	1,87
Campus	1,65	1,36	1	2,08	1,36

(b) Average Bhattacharyya-based distances

	Cherr	**Arbor**	**Football**	**Cann**	**Campus**
Cherr	1	1,64	1,69	1,45	1,5
Arbor	1,75	1	1,91	2,15	1,42
Football	2,28	2,43	1	2,67	1,82
Cann	1,12	1,56	1,52	1	1,5
Campus	1,57	1,39	1,4	2,02	1

(c) Average C2 based distances

distance, C2, and Symmetric Kullback-Liebler metrics respectively. The computations were performed in Matlab on a Pentium 2.4 GHz PC.

Note that the Bhattacharyya-based distance that was used is

$$BhGMM(p, p') = \sum_{i=1}^{N}\sum_{j=1}^{M}\pi_i\pi'_j B(p_i, p'_j),$$

where p, p' are Gaussian mixture models consisting of N and M kernels respectively, p_i, p'_j denote the kernel parameters and π_i, π'_j are the mixing weights. B denotes the Bhattacharyya distance between two Gaussian kernels, defined as [4]:

$$B(p, p') = \frac{1}{8}(\mu-\mu')^T\left(\frac{\Sigma+\Sigma'}{2}\right)^{-1}(\mu-\mu') + \frac{1}{2}\ln\left[\frac{\left|\frac{\Sigma+\Sigma'}{2}\right|}{\sqrt{|\Sigma||\Sigma'|}}\right]$$

where μ, Σ and μ', Σ' stand for the means and covariance matrices of Gaussian kernels p, p' respectively.

In Table 1 we provide for each metric the resulting distances among image classes normalized so that the minimum distance value over each line is 1. These distances are the means over each of the image categories. For example, by distance of group 'Cherry' to group 'Campus in fall', we mean the average distance of every image in 'Cherry' to every image in 'Campus in fall'. An issue to check out in this Table is the distance of an image group with itself (i.e the diagonal elements); if it is comparatively small, then the metric works well. In other words, the more ones in the diagonal the better the metric is. Notice that while C2 is about four times slower than the Bhattacharyya-based distance, it provides better results.

Table 2. Average between-class distances between original and sub-sampled images

	Cherr	**Arbor**	**Foot**	**Cann**	**Camp**
S-Cher	3,6e16	2,8e19	1	1,21	5,7e19
S-Arbo	2,14	1,21	1	2,87	2,14
S-Foot	1,86	1,38	1	2,45	1,94
S-Cann	1	2,12	1,15	1	2,86
S-Camp	1,8	1,56	1	2,12	1,7

(a) Average Bhattacharyya-based distances

	Cherr	**Arbor**	**Foot**	**Cann**	**Camp**
S-Cher	1	1,56	1,66	1,39	1,52
S-Arbo	1,5	1	1,62	1,68	1,27
S-Foot	2,17	2,41	1	2,22	1,82
S-Cann	1,23	1,74	1,67	1	1,69
S-Camp	1,47	1,39	1,31	1,67	1

(b) Average C2 distances

The Symmetric Kullback – Liebler (SKL) distance provides good results, however it is very slow to compute even when only 4096 (about 1/85 of the total) pixels per image are used.

To test the robustness of the metrics, we have conducted a second set of experiments. That is, we produced a sub-sampled copy of each of the original images, which has only half the width and height of the original. Then, based on the RGB values of the sub-images the GM models have been computed. Then, the distances of the GM models of the sub-sampled images were compared to those of the full images.

We have conducted the above test for the Bhattacharyya and C2 metrics, computing average distances as in the 'non-subsampled' scenario. This time, we compare each original image category with each sub-sampled image category. The distances computed are shown in Table 2. (Note that the S- prefix is used for the sub-sampled images).

5 Conclusions – Future Work

We have experimented with a new distance metric for PDFs that seems to work well for image retrieval when the images are modeled using GMMs. The metric is fast to compute, since it has a closed form when a GM model is used for the PDF, it also provides as good separation between different classes of images, similar to that produced by symmetric KL divergence which was computed using Monte-Carlo. Furthermore, in an initial test it also seems to be robust. We also compared this metric with a Bhattacharyya-based metric which, although it is fast to compute, it does not provide as good results in terms of class separation. In the future we plan test this metric with more features (edge, texture) and with a larger image database. Also we plan to test the accuracy of the SKL metric as the number of samples used in the Monte-Carlo approximation is reduced.

References

1. G. McLachlan, D. Peel: "*Finite Mixture Models*", Wiley 2000
2. J.Han, K.Ma: "Fuzzy Color Histogram and its use in Color Image Retrieval", *IEEE Trans. on Image Processing*, vol. 11, No. 8, August 2003.
3. A. Del Bimbo: "*Visual information Retrieval*", Morgan Kaufmann publishers, San Francisco, 1999.
4. K. Fukunaga: "Introduction to Statistical Pattern Recognition", Academic Press 1990.
5. Surajit Ray, "Distance-Based Model Selection with Application to the Analysis of Gene Expression Data", Ms Thesis, Dept. of Statistics, Pennsylvania State University, 2003.