

SIMPLIFYING GAUSSIAN MIXTURE MODELS VIA ENTROPIC QUANTIZATION

Frank Nielsen, Vincent Garcia, and Richard Nock

École Polytechnique, LIX
91128 Palaiseau Cedex, FRANCE
Sony Computer Science Laboratories, FRL, JAPAN
{nielsen,garciav}@lix.polytechnique.fr, mock@martinique.univ-ag.org

ABSTRACT

Mixture models are a crucial statistical modeling tool at the heart of many challenging applications in computer vision, machine learning, and text classification among others. In this paper, we describe a novel and efficient algorithm for simplifying Gaussian mixture models using a generalization of the celebrated k -means quantization algorithm tailored to relative entropy in statistical distribution spaces. Our algorithm extends easily to *arbitrary* mixture of exponential families. The proposed method is shown to compare favourably well with the state-of-the-art unscented transform clustering algorithm both in terms of time and quality performances.

1. INTRODUCTION AND PRIOR WORK

A mixture model provides a powerful framework to estimate a probability density function of a random variable using a mixture distribution. For instance, the Gaussian mixture models (GMMs for short) – also known as mixture of Gaussians (MoGs) – have been widely used in many different area domains such as image processing, finance, etc.

The density of a mixture model f evaluated at point $x \in \mathbb{R}^d$ is given by

$$f(x) = \sum_{i=1}^n \alpha_i f_i(x) \quad (1)$$

where $\alpha_i \geq 0$ denotes the weight of each component of the mixture, with $\sum_{i=1}^n \alpha_i = 1$. If f is a Gaussian mixture model, each function f_i is a multivariate Gaussian function

$$f_i(x) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp\left(-\frac{(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)}{2}\right) \quad (2)$$

parametrized by its mean $\mu_i \in \mathbb{R}^d$ and its covariance symmetric positive-definite matrix $\Sigma_i \succ 0$. It is common to estimate model parameters from independent and identically-distributed observations using expectation-maximization (EM) local optimization algorithm [3].

A typical operation on mixture models is the estimation of statistical measures such as Shannon entropy or the Kullback-Leibler divergence. With large number of components in the mixture model (e.g. arising from a kernel-based Parzen density estimation [11]), the estimation of these measures can be a bottleneck in terms of computation time. The computational requirements can be strongly decreased by reducing the number of components in the mixture model. Given a mixture model f containing n components (see Eq. (1)), the problem of mixture model simplification con-

sists in computing a simpler mixture model g

$$g(x) = \sum_{j=1}^m \alpha'_j g_j(x) \quad (3)$$

with m components ($1 \leq m < n$) such as g is the “best” approximation of f with respect to a similarity measure. Another way to obtain a compact representation of f is to re-learn the mixture model directly from the source dataset. However, depending on the application, this may not be applicable [5]. Indeed, the estimation of a mixture model is computationally expensive with large datasets, or sometimes, the initial dataset is not anymore available. Reducing the initial mixture model is then the only way.

Some methods of mixture model simplification have been proposed in the last decade. Zhang and Kwok [12] have proposed to simplify a GMM by first grouping similar components together and then performing local fitting through function approximation. By using the squared loss to measure the distance between mixture models, their algorithm naturally combines the two different tasks of component clustering and model simplification. Goldberger *et al.* [5] have proposed a fast GMM simplification algorithm named UTAC (Unscented Transform Approximation Clustering) based on the Unscented Transform (UT) method [4, 8]. The UTAC algorithm proceeds by maximizing the UTA (Unscented Transform Approximation of the negative cross-entropy) criterion computed between the two GMMs, f and g . The authors show that the UTA criterion can be maximized with a standard EM-like algorithm. Davis and Dhillon [2] have proposed a hard clustering algorithm based on the decomposition of the relative entropy as the sum of a Burg matrix divergence with a Mahalanobis distance parametrized by the covariance matrices. Goldberger and Roweis [6] have proposed a GMM simplification algorithm based on the k -means hard clustering. A common drawback of these methods is that they only consider the problem of GMM simplification. However, other kind of mixture models have been successfully used in different applications such as multinomial mixture models in text classification [10]. Proposing a simplification algorithm working not only on GMMs but on a generic wider class of mixture models, called exponential families, is necessary.

In this paper, we describe a novel and efficient algorithm for simplifying Gaussian mixture models using a generalization of the celebrated k -means quantization algorithm tailored to relative entropy. Our algorithm extends easily to *arbitrary* mixture of exponential families. The proposed

method is shown to compare favourably well with the state-of-the-art UTAC algorithm both in terms of time and quality performances.

2. ENTROPIC QUANTIZATION OF GMMS

2.1 Relative entropy and Bregman divergence

The fundamental measure between statistical distributions is the *relative entropy*, also called the Kullback-Leibler divergence (denoted by KLD). For f_i and f_j two distributions, the KLD is an oriented distance (asymmetric) and is defined as

$$\text{KLD}(f_i||f_j) = \int f_i(x) \log \frac{f_i(x)}{f_j(x)} dx. \quad (4)$$

This fastidious integral computation yields for multivariate normals

$$\begin{aligned} \text{KLD}(f_i||f_j) &= \frac{1}{2} \log \left(\frac{\det \Sigma_j}{\det \Sigma_i} \right) + \frac{1}{2} \text{tr} \left(\Sigma_j^{-1} \Sigma_i \right) \\ &\quad + \frac{1}{2} (\mu_j - \mu_i)^T \Sigma_j^{-1} (\mu_j - \mu_i) - \frac{d}{2} \end{aligned} \quad (5)$$

where $\text{tr}(\Sigma)$ is the matrix trace operator, *i.e.* the sum of its diagonal elements.

It turns out that we can *bypass* the integral computation using the canonical form of *exponential families* [1] $\exp \{ \langle \tilde{\Theta}, t(x) \rangle - F(\tilde{\Theta}) + C(x) \}$ where $\tilde{\Theta}$ are the *natural parameters* associated with the *sufficient statistics* $t(x)$. The *log normalizer* $F(\tilde{\Theta})$ is a strictly convex and differentiable function that specifies uniquely the exponential family, and the function $C(x)$ is the carrier measure (*e.g.*, Lebesgue or counting measures). The relative entropy between two distribution members of the *same* exponential family is equal to the Bregman divergence defined for the log normalizer F on the natural parameter space:

$$\text{KLD}(f_i||f_j) = D_F(\tilde{\Theta}_j||\tilde{\Theta}_i) \quad (6)$$

where

$$D_F(\tilde{\Theta}_j||\tilde{\Theta}_i) = F(\tilde{\Theta}_j) - F(\tilde{\Theta}_i) - \langle \tilde{\Theta}_j - \tilde{\Theta}_i, \nabla F(\tilde{\Theta}_i) \rangle. \quad (7)$$

The $\langle \cdot, \cdot \rangle$ denotes the inner product $\langle p, q \rangle = p^T q$ and ∇F is the gradient operator. For multivariate normals, we consider *mixed-type* vector/matrix parameters. The sufficient statistics is *stacked* onto a two-part d -dimensional vector/matrix entity $\tilde{x} = (x, -\frac{1}{2}xx^T)$ associated with the natural parameter $\tilde{\Theta} = (\theta, \Theta) = (\Sigma^{-1}\mu, \frac{1}{2}\Sigma^{-1})$. The log normalizer specifying the exponential family is [9]

$$F(\tilde{\Theta}) = \frac{1}{4} \text{tr}(\Theta^{-1}\theta\theta^T) - \frac{1}{2} \log \det \Theta + \frac{d}{2} \log 2\pi.$$

The inner product $\langle \tilde{\Theta}_p, \tilde{\Theta}_q \rangle$ becomes a *composite* inner product obtained as the sum of two inner products of vectors and matrices: $\langle \tilde{\Theta}_p, \tilde{\Theta}_q \rangle = \langle \Theta_p, \Theta_q \rangle + \langle \theta_p, \theta_q \rangle$. For matrices, the inner product $\langle \Theta_p, \Theta_q \rangle$ is defined by the trace of the matrix product $\Theta_p \Theta_q^T$: $\langle \Theta_p, \Theta_q \rangle = \text{tr}(\Theta_p \Theta_q^T)$. The gradient ∇F is given in mixed vector-matrix type by

$$\nabla F(\tilde{\Theta}) = \left(-\frac{1}{2}\Theta^{-1} - \frac{1}{4}(\Theta^{-1}\theta)(\Theta^{-1}\theta)^T, \frac{1}{2}\Theta^{-1}\theta \right).$$

2.2 Bregman k -means

Banerjee *et al.* [1] extended Lloyd's k -means algorithm to the class of Bregman divergences, generalizing also the former Linde-Buzo-Gray and information-theoretic clusterings. They proved that the simple Lloyd's iterative algorithm minimizes *monotonically* the Bregman (right-sided) loss function:

$$\text{LossFunction}_F(\{x_1, \dots, x_n\}; k) = \min_{c_1, \dots, c_k} \sum_k \sum_i D_F(x_i || c_k).$$

where x_i are the source point sets and c_k the respective cluster centroids. Thus a right-sided Bregman k -means is a left-sided differential entropic (*i.e.* KLD) clustering, and vice-versa. In this paper, we propose a GMM simplification algorithm based on Bregman k -means. The k -means algorithm is the repetition until convergence of two steps: first calculate membership in clusters (repartition step), and second recompute the centroids. The algorithms 1 and 2 respectively present the right-sided and the left-sided Bregman k -means clustering algorithms (noted BKMC).

Figure 1 shows the left-sided (blue) and right-sided (red) entropic centroids of a set of five bivariate normals.

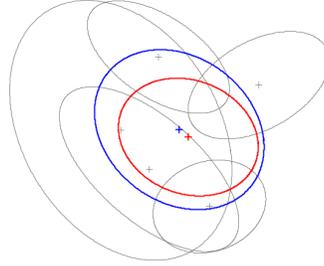


Figure 1: Screenshot of the left-sided (blue) and right-sided (red) entropic centroid of bivariate normals. Each normal is represented by a point in dimension 5 and rasterized on the canvas as a centered ellipse.

Algorithm 1 BKMC right-sided(f, m)

- 1: Initialize the GMM g .
- 2: **repeat**
- 3: Compute the cluster C . The Gaussian f_i belongs to cluster C_j if and only if

$$D_F(\tilde{\Theta}_i || \tilde{\Theta}'_j) < D_F(\tilde{\Theta}_i || \tilde{\Theta}'_l), \forall l \in [1, m] \setminus \{j\} \quad (8)$$

- 4: Compute the centroids. The natural parameters of the j -th centroid (*i.e.* Gaussian g_j) are given by:

$$\alpha'_j = \sum_i \alpha_i, \quad \theta'_j = \frac{\sum_i \alpha_i \theta_i}{\sum_i \alpha_i}, \quad \Theta'_j = \frac{\sum_i \alpha_i \Theta_i}{\sum_i \alpha_i} \quad (9)$$

The sum \sum_i is performed on $i \in [1, m] / f_i \in C_j$.

- 5: **until** the cluster does not change between two iterations.
-

2.3 Symmetric Bregman k -means

For some applications (*e.g.* content-based image retrieval), the use of a symmetric similarity measure is required. The

Algorithm 2 BKMC left-sided(f, m)

- 1: Initialize the GMM g .
- 2: **repeat**
- 3: Compute the cluster C . The Gaussian f_i belongs to cluster C_j if and only if

$$D_F(\tilde{\Theta}'_j || \tilde{\Theta}_i) < D_F(\tilde{\Theta}'_l || \tilde{\Theta}_i), \forall l \in [1, m] \setminus \{j\}$$

- 4: Compute the centroids. The natural parameters of the j -th centroid (*i.e.* Gaussian g_j) are given by:

$$\alpha'_j = \sum_i \alpha_i, \quad \tilde{\Theta}'_j = \nabla F^{-1} \left(\sum_i \frac{\alpha_i}{\alpha'_j} \nabla F(\tilde{\Theta}_i) \right) \quad (10)$$

where

$$\nabla F^{-1}(\tilde{\Theta}) = \begin{pmatrix} -(\Theta + \theta\theta^T)^{-1} \theta \\ -\frac{1}{2}(\Theta + \theta\theta^T)^{-1} \end{pmatrix} \quad (11)$$

- 5: **until** the cluster does not change between two iterations.
-

BKMC algorithm can be adapted to satisfy this condition by using the symmetrized Bregman divergence. Previously, the repartition step used the right-sided or the left-sided Bregman divergence to gather the Gaussians of f into a cluster. Instead, we propose to use the symmetric Bregman divergence noted SD_F . Given two Gaussians $\tilde{\Theta}_p$ and $\tilde{\Theta}_q$ (natural parameters), SD_F is the mean of the right-sided and left-sided Bregman divergence:

$$SD_F(\tilde{\Theta}_p, \tilde{\Theta}_q) = \frac{D_F(\tilde{\Theta}_q || \tilde{\Theta}_p) + D_F(\tilde{\Theta}_p || \tilde{\Theta}_q)}{2} \quad (12)$$

The centroid step consists first in computing the right-sided and left-sided centroids (respectively noted c_r and c_l) as explained in algorithms 1 and 2. The symmetric centroid c belongs to the geodesic link between c_r and c_l . A point on this link is given by

$$c_\lambda = \nabla F^{-1}(\lambda \nabla F(c_r) + (1 - \lambda) \nabla F(c_l)) \quad (13)$$

where $\lambda \in [0, 1]$. The symmetric centroid $c = c_\lambda$ verifies

$$SD_F(c_\lambda, c_r) = SD_F(c_\lambda, c_l). \quad (14)$$

A standard dichotomy search on λ allows to quickly find the symmetric centroid c for a given precision.

3. EXPERIMENTS

3.1 Bregman k -means clustering

In this section, we apply the BKMC algorithm for simplifying Gaussian mixture models. We compare the influence of the Bregman divergence type (right-sided, left-sided, or symmetrized) on the quality of the simplified GMM g . This quality is usually evaluated through the standard right-sided KLD between f and g . The KLD is estimated with a classical Monte-Carlo algorithm [7] since it does not admit any closed-form solution.

For this experiment, the initial GMM f is composed of 32

Gaussians and is computed from the image Baboon (see figure 4): first we perform a standard k -means algorithm to gather RGB pixels in 32 classes, second we compute f with a standard Expectation-Maximization algorithm (EM). The dimension of the Gaussians is 3 (components RGB: red, green, blue).

The figure 2 shows the evolution of the KLD as a function of m (number of the Gaussians in the simplified GMM) for the different Bregman divergence types (right-sided, left-sided, or symmetric) used in the BKMC algorithms. First, the KLD decreases with m as expected whatever the Bregman divergence type used. Indeed, the quality of the approximation of the initial GMM f increases with the number of Gaussians in the simplified model g . Second, the left-sided Bregman divergence gives the best results and the right-sided the worst. The measure used to evaluate the quality of the simplification is the right-sided KLD. The left-sided Bregman clustering on natural parameters amounts to compute a right-sided KLD clustering on corresponding probability measures [9], and vice-versa. Obtaining the best approximation with the left-sided BKMC is then the expected behaviour. The symmetric centroid, being computed from right and left centroids, provides better results than right-sided divergence but worse than left-sided divergence. It turns out that the differential entropic clustering method of Davis and Dhillon [2] is a right-sided Bregman clustering presented as a left-sided KLD entropic clustering. In the paper remainder, we will use the left-sided BKMC.

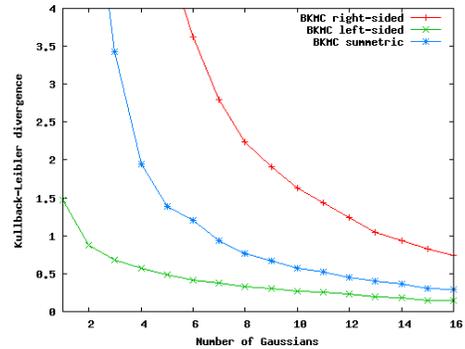


Figure 2: Evolution of the (right-sided) KLD as a function of m for algorithms right-sided, left-sided, and symmetric BKMC. The left-sided BKMC provides the best approximation of the initial GMM.

3.2 BKMC versus UTAC

The figure 3 shows the evolution of the KLD as a function of m (number of classes for the simplified GMM) for algorithms UTAC and BKMC (left-sided). Both algorithms are written in Java. The initial GMM is computed as in section 3.1. With the two methods, the KLD decreases with m . BKMC provides the best results and is faster than UTAC algorithm: for $m = 16$, the clustering process is performed in 20 milliseconds for BKMC and 900 milliseconds for UTAC on a Dell Precision M6400 laptop (Intel Core 2 duo @ 2.53GHz, 4Go DDR2 memory, Windows Vista 64 bits, Java 1.6). Indeed, BKMC is based on a k -means algorithm which generally quickly converges. UTAC uses a EM method known to converge slowly (*i.e.* within a threshold after large number

of iterations). We automatically stop the UTAC process after 30 iterations if the process has not converged.

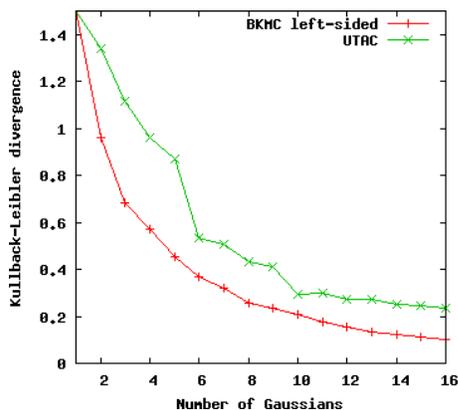


Figure 3: Evolution of the Kullback-Leibler divergence as a function of m for algorithms BKMC (proposed method) and UTAC.

3.3 Clustering-based image segmentation

In this section, we apply the GMM simplification methods in the context of clustering-based image segmentation problem. Given a color image, a pixel x can be considered as a point in \mathbb{R}^3 . Given a GMM f of n Gaussians, the segmentation is performed by affecting each pixel x to the most probable class C_i :

$$f_i(x) > f_j(x) \forall j \in [1, n] \setminus \{i\}$$

The segmentation is then illustrated by assigning the value of the class representative μ_i to the pixel x .

For this experiment (see figure 4), we first consider an image (first column) and we compute an initial GMM f of 32 components as described in section 3.1. From this GMM, we compute the image segmentation (second column). f is then simplified into a 16 components GMM with algorithms UTAC and BKMC providing two different image segmentations (respectively third and fourth columns of the figure 4). The images used for the experiment are Baboon, Lena, Colormap, and Shantytown. The figure 4 also shows the value of the KLD (right-sided) between f and g .

With all images tested, the algorithm BKMC provides the best results (in terms of KLD value). The visual segmentation seems to be better with BKMC (closer to the initial segmentation). However, it is difficult to objectively judge the quality of the simplified GMM. Only a similarity measure such as Kullback-Leibler allows to compare both approaches.

4. CONCLUDING REMARKS

We have described a novel algorithm (BKMC) for simplifying Gaussian models based on a powerful generalization of Lloyd's celebrated k -means algorithm to entropic Bregman divergences [1]. Our algorithm extends easily to arbitrary mixture of exponential families. Interestingly, BKMC bypasses the problem of solving costly eigenvalue problems to find out the sigma points required by the state-of-the-art unscented transform clustering algorithm [5] (UTAC). We thus obtain faster simplification processing times as the dimension increase. Experiments corroborate that BKMC yields

better results in shorter computational time.

Quantizing GMM by k -means amounts to locally minimize the loss function defined as the Bregman information of all Gaussians minus the Bregman information of the codebook. Our method is thus related to rate distortion and information bottleneck theory as explained in [1]. As a future work, we are considering learning automatically the most appropriate number of components m in a simplified model. A Java applet illustrating the BKMC quantization algorithm is available on-line at:

<http://www.sonycs1.co.jp/person/nielsen/KMj/>

5. ACKNOWLEDGEMENTS

We thank Professor Jacob Goldberger [5] for email correspondences concerning the UTAC algorithm. This research was financially supported by ANR-07-BLAN-0328-01 GAIA (Computational Information Geometry and Applications) and DIGITEO GAS 2008-16D (Geometric Algorithms & Statistics).

REFERENCES

- [1] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6:234–245, 2005.
- [2] J. V. Davis and I. Dhillon. Differential entropic clustering of multivariate gaussians. In *Neural Information Processing Systems*, 2006.
- [3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society B*, 39:1–38, 1977.
- [4] J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures. In *IEEE International Conference on Computer Vision*, 2003.
- [5] J. Goldberger, H. Greenspan, and J. Dreyfuss. Simplifying mixture models using the unscented transform. *IEEE Transactions Pattern Analysis Machine Intelligence*, 30:1496–1502, 2008.
- [6] J. Goldberger and S. Roweis. Hierarchical clustering of a mixture model. In *Neural Information Processing Systems*, 2004.
- [7] J. R. Hershey and P. A. Olsen. Approximating the Kullback Leibler divergence between gaussian mixture models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2007.
- [8] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92:401–422, 2004.
- [9] F. Nielsen, J.-D. Boissonnat, and R. Nock. On Bregman Voronoi diagrams. In *SIAM Symposium on Discrete Algorithms*, 2007.
- [10] J. Novoviov and A. Malk. Application of multinomial mixture model to text classification. In *Pattern Recognition and Image Analysis*, 2003.
- [11] E. Parzen. On the estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.
- [12] K. Zhang and J. T. Kwok. Simplifying mixture models through function approximation. In *Neural Information Processing Systems*, 2006.

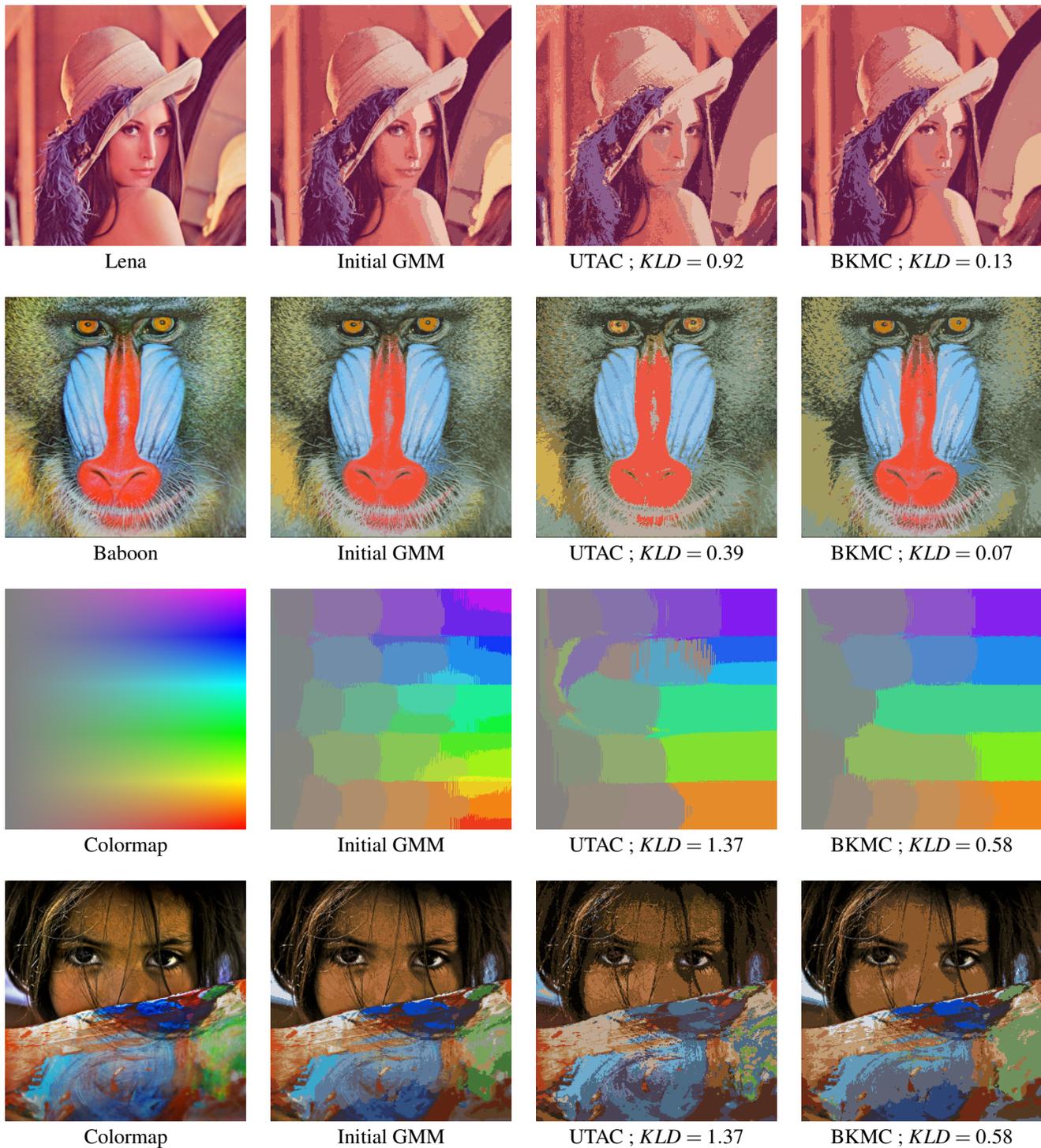


Figure 4: Application of GMM simplification algorithms for clustering-based image segmentation. The first and second columns show respectively the input image and the segmentation computed from the initial GMM f composed of 32 Gaussians. The third and fourth columns show the segmentations respectively computed after the simplification of f with the algorithms UTAC and BKMC. The BKMC algorithm provides the best results according to the KLD value. The images tested are (from left to right) Baboon, Lena, Colormap, and Shantytown.