

Unsupervised Image Clustering using the Information Bottleneck Method

Jacob Goldberger¹, Hayit Greenspan² and Shiri Gordon²

¹ CUTE Systems Ltd. Tel-Aviv, Israel, jacob@cute.co.il

² Faculty of Engineering, Tel Aviv University, Tel Aviv 69978, Israel,
hayit@eng.tau.ac.il

Abstract. A new method for unsupervised image category clustering is presented, based on a continuous version of a recently introduced information theoretic principle, the information bottleneck (IB). The clustering method is based on hierarchical grouping: Utilizing a Gaussian mixture model, each image in a given archive is first represented as a set of coherent regions in a selected feature space. Images are next grouped such that the mutual information between the clusters and the image content is maximally preserved. The appropriate number of clusters can be determined directly from the IB principle. Experimental results demonstrate the performance of the proposed clustering method on a real image database.

Keywords: Image Categories; Unsupervised Clustering; Image Grouping; Gaussian mixture modeling; Kullback-Leibler distance; Information bottleneck.

1 Introduction

Image clustering and categorization is a means for high-level description of image content. The goal is to find a mapping of the archive images into classes (clusters) such that the set of classes provide essentially the same prediction, or information, about the image archive as the entire image set collection. The generated classes provide a concise summarization and visualization of the image content. Image archive clustering is important for efficient handling (search and retrieval) in large image databases [6, 2]. In the retrieval process, the query image is initially compared with all the cluster centers. The subset of clusters that has the largest similarity to the query image is chosen. The query image is next compared with all the images within the selected subset of clusters. Search efficiency is improved due to the fact that the query image is not compared exhaustively to all the images in the database.

In most clustering methods (e.g. K-means), a distance measure between two data points or between a data point and a class centroid is given a priori as part of the problem setup. The clustering task is to find a small number of classes with low intra-class variability. However in many clustering problems, e.g. image clustering, the objects we want to classify have a complicated high dimensional

structure and choosing the right distance measure is not a straight-forward task. A choice of a specific distance measure can influence the clustering results.

The clustering framework presented in this work is based on hierarchical grouping: image pixels are first grouped into coherent regions in feature space; these are modelled via Gaussian mixture models (GMMs). Next, utilizing the information bottleneck (IB) method [7], the image models are grouped, bottom-up, into coherent clusters. Characteristics of the proposed method include: 1) Image *models* are clustered rather than raw image pixels. The clustering is thus done in a continuous domain. 2) The IB method provides a simultaneous construction of both the clusters and the distance measure between them. 3) A natural termination of the bottom-up clustering process can be determined as part of the IB principle. This provides an automated means for finding the relevant number of clusters per archive.

2 Grouping pixels into GMMs

In the first layer of the grouping process we shift from image pixels to a mid-level representation of an image in which the image is represented as a set of coherent regions in feature space. In this work we model each image as a mixture of Gaussians in the color ($L * a * b$) feature space. The representation model is a general one, and can incorporate any desired feature space (such as texture, shape, etc) or combination thereof. In order to include spatial information, the (x, y) position of the pixel is appended to the feature vector. Following the feature extraction stage, each pixel is represented with a five-dimensional feature vector, and the image as a whole is represented by a collection of feature vectors in the five-dimensional space.

Pixels are grouped into homogeneous regions, by grouping the feature vectors in the selected five-dimensional feature space. The underlying assumption is that the image colors and their spatial distribution in the image plane are generated by a mixture of Gaussians. The distribution of a d -dimensional random variable is a mixture of k Gaussians if its density function is:

$$f(y) = \sum_{j=1}^k \alpha_j \frac{1}{\sqrt{(2\pi)^d |\Sigma_j|}} \exp\left\{-\frac{1}{2}(y - \mu_j)^T \Sigma_j^{-1} (y - \mu_j)\right\} \quad (1)$$

Learning a Gaussian mixture model is in essence an unsupervised clustering task. The Expectation-Maximization (EM) algorithm is used [4], to determine the maximum likelihood parameters of a mixture of k Gaussians in the feature space (similar to [1]). The first step in applying the EM algorithm to the problem at hand is to initialize the mixture model parameters. The K-means algorithm is utilized to extract the data-driven initialization. The updating process is repeated until the log-likelihood is increased by less than a predefined threshold from one iteration to the next. In this work we choose to converge based on the log-likelihood measure and we use a 1% threshold. Using EM, the parameters representing the Gaussian mixture are found. K -Means and EM are calculated

for $k \geq 1$, with k corresponding to the model size. The Minimum Description Length (MDL) principle [3] serves to select among values of k . In our experiments, k ranges from 3 to 6.



Fig. 1. Input image (left) Image modeling via Gaussian mixture (right).

Figure 1 ¹ shows two examples of learning a GMM model for an input image. In this visualization each localized Gaussian mixture is shown as a set of ellipsoids. Each ellipsoid represents the support, mean color and spatial layout, of a particular Gaussian in the image plane.

3 The Agglomerative Information Bottleneck

The second layer of the image grouping process is based on information theoretic principle, the information bottleneck method (IB) [7]. Using the IB method, the extracted image models are grouped, bottom-up, into coherent clusters ². The IB principle states that among all the possible clusterings of the object set into a fixed number of clusters, the desired clustering is the one that minimizes the loss of mutual information between the objects and the features extracted from them. The IB method can be motivated from Shannon’s rate distortion theory [3] which provides lower bounds on the number of classes we can divide a source given a distortion constraint. Given a random variable, X , and a distortion measure, $d(x_1, x_2)$, defined on the alphabet of X , we want to classify-quantize the symbols of X such that the average quantization error is less than a given number D . It is clear that we can reduce the average quantization error by enlarging the number of clusters. Shannon’s rate distortion theorem states that the minimum log number of clusters needed to keep the average quantization error below D is given by the following rate-distortion function:

$$R(D) = \min_{p(\hat{x}|x) | Ed(x, \hat{x}) \leq D} I(X; \hat{X}) \quad (2)$$

where the average distortion $Ed(x, \hat{x})$ is $\sum_{x, \hat{x}} p(x)p(\hat{x}|x)d(x, \hat{x})$ and $I(X; \hat{X})$ is the mutual information between X and \hat{X} given by:

$$I(X; \hat{X}) = \sum_{x, \hat{x}} p(x)p(\hat{x}|x) \log \frac{p(\hat{x}|x)}{p(\hat{x})}$$

¹ A color version of the paper may be found in <http://www.eng.tau.ac.il/~hayit>

² Recent work using related information-theoretic concepts for within-image clustering, or segmentation, can be found in [5].

The random variable \hat{X} can be viewed as a soft-probabilistic classification of X .

Unlike classical rate distortion theory, the IB method avoids the arbitrary choice of a distance or a distortion measure. Instead, clustering of the object space (denoted by X) is done by preserving the relevant information about another “feature” space (denoted by Y). In our case the objects are images and the feature space consists of local information we extract for each pixel (e.g. color, texture). We assume, as part of the IB approach, that $\hat{X} \rightarrow X \rightarrow Y$ is a markov chain, i.e. given X , the clustering \hat{X} is independent of the feature space, Y . Consider the following distortion function:

$$d(x, \hat{x}) = D(p(y|X=x) || p(y|\hat{X}=\hat{x})) \quad (3)$$

where $D(f||g) = E_f \log \frac{f}{g}$ is the Kullback-Leibler divergence [3]. Note that $p(y|\hat{x}) = \sum_x p(x|\hat{x})p(y|x)$ is a function of $p(\hat{x}|x)$. Hence, $d(x, \hat{x})$ is not predetermined. Instead it depends on the clustering. Therefore, as we search for the best clustering we also search for the most suitable distance measure.

The loss in the mutual information between X and Y caused by the (probabilistic) clustering \hat{X} is in fact the average of this distortion measure:

$$\begin{aligned} I(X;Y) - I(\hat{X};Y) &= \sum_{x,\hat{x},y} p(x, \hat{x}, y) \log \frac{p(x|y)}{p(x)} - \sum_{x,\hat{x},y} p(x, \hat{x}, y) \log \frac{p(y|\hat{x})}{p(y)} = \\ &= \sum_{x,\hat{x},y} p(x, \hat{x}, y) \log \frac{p(y|x)}{p(y|\hat{x})} = \sum_{x,\hat{x}} p(x, \hat{x}) \sum_y p(y|x) \log \frac{p(y|x)}{p(y|\hat{x})} = E[D(p(y|x)||p(y|\hat{x}))] \end{aligned} \quad (4)$$

Substituting distortion measure (3) in the rate distortion function we obtain:

$$R(D) = \min_{p(\hat{x}|x) | I(X;Y) - I(\hat{X};Y) \leq D} I(X; \hat{X}) \quad (5)$$

which is exactly the minimization criterion proposed by IB principle, namely, finding a clustering that causes minimum reduction of the mutual information between the objects and the features.

The minimization problem posed by the IB principle can be approximated by a greedy algorithm based on a bottom-up merging procedure [7]. The algorithm starts with the trivial clustering where each cluster consists of a single point. In every greedy step we merge the two classes such that the loss in the mutual information caused by merging them is the smallest. This ensures minimization of the overall information loss. Let c_1 and c_2 be two image clusters. The information loss due to the merging of c_1 and c_2 is:

$$d(c_1, c_2) = I(C_{before}, Y) - I(C_{after}, Y) = E[D(p(y|C_{before}) || p(y|C_{after}))] \geq 0$$

where $I(C_{before}, Y)$ and $I(C_{after}, Y)$ are the mutual information between the classes and the feature space before and after c_1 and c_2 are merged into a single class. Standard information theory manipulation reveals:

$$d(c_1, c_2) = \sum_{y,i=1,2} p(c_i, y) \log \frac{p(c_i, y)}{p(c_i)p(y)} - \sum_y p(c_1 \cup c_2, y) \log \frac{p(c_1 \cup c_2, y)}{p(c_1 \cup c_2)p(y)} =$$

$$= \sum_{y,i=1,2} p(c_i, y) \log \frac{p(y|c_i)}{p(y|c_1 \cup c_2)} = \sum_{i=1,2} p(c_i) D(p(y|c_i) || p(y|c_1 \cup c_2)) \quad (6)$$

4 Using the IB Method for Clustering of Image GMMs

In this section we present the image clustering algorithm proposed, as defined from the IB principle. To apply the IB principle we have first to define a joint distribution on the images and the features. In the following we denote by X the set of images we want to classify. We assume uniform prior probability $p(x)$ of observing an image. Denote by Y the random variable associated with the feature vector extracted from a single pixel. The Gaussian mixture model we use to describe the feature distribution within an image x (section 2) is exactly the conditional density function $p(y|x)$. Thus we have a joint image-feature distribution $p(x, y)$. Note that since $p(y|x)$ is a GMM distribution, the density function per cluster c , $p(y|c) = \frac{1}{|c|} \sum_{x \in c} p(y|x)$, is a mixture of GMMs and therefore it is also a GMM. Let f_1, f_2 be the GMMs associated with image clusters c_1, c_2 respectively. The GMM of the merged cluster $c_1 \cup c_2$, denoted by f , is:

$$f(y) = \frac{|c_1|}{|c_1 \cup c_2|} f_1(y) + \frac{|c_2|}{|c_1 \cup c_2|} f_2(y)$$

According to expression (6), the distance between the two image clusters c_1 and c_2 is:

$$d(c_1, c_2) = \frac{|c_1|}{N} D(f_1(y) || f(y)) + \frac{|c_2|}{N} D(f_2(y) || f(y)) \quad (7)$$

where N is the size of the image database. Hence, to compute the distance between two image clusters c_1 and c_2 we need to compute the KL distance between two GMM distributions.

Since the KL distance between two GMMs can not be analytically computed, we can numerically approximate it through Monte-Carlo procedures. Denote the feature set extracted from the images, that belongs to cluster c_1 , by $y_1 \dots y_n$. The KL distance $D(f_1 || f)$ can be approximated as follows:

$$D(f_1 || f) \cong \frac{1}{n} \sum_{t=1}^n \log \frac{f_1(y_t)}{f(y_t)} \quad (8)$$

Another possible approximation is to use synthetic samples produced from the Gaussian mixture distribution f_1 instead of the image data. This enables us to compute the KL distance without referring to the images from which the models were built. Image categorization experiments show no significant difference between these two proposed approximations of the KL distance. The expression $D(f_2 || f)$ can be approximated in a similar manner. The agglomerative IB algorithm for image clustering is the following:

1. Start with the trivial clustering where each image is a cluster.

2. In each step merge clusters c_1 and c_2 such that information loss $d(c_1, c_2)$ (equation 7) is minimal.
3. Continue the merging process until the information loss $d(c_1, c_2)$ is more than a predefined threshold, indicating that we attempt to merge two non-similar clusters.

5 Results

An image databases consists of 100 natural scenery images was randomly extracted from the COREL database. A Gaussian mixture model was built for each image as described in section 2. Next the bottom-up clustering method, described in section 4, was applied to the image models. We started with 100 clusters where each image is a cluster. After 99 steps all the images were merged into a single cluster.

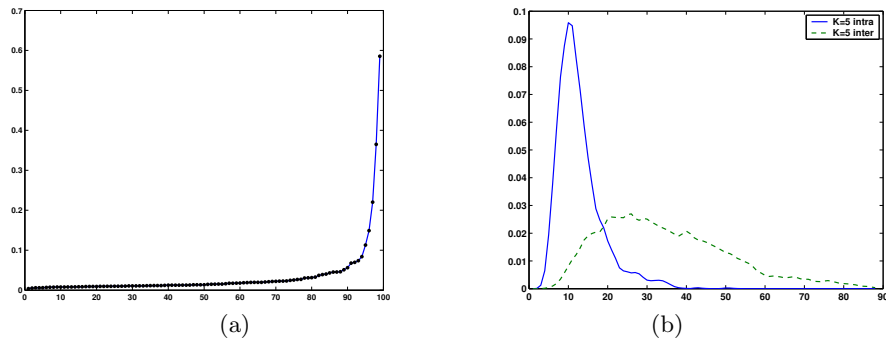


Fig. 2. (a) Loss of mutual information during the clustering process. (b) Statistical analysis: intra-class distance (solid) and inter-class distance (dashed).

The loss of mutual information during each merging step of the clustering process is presented in Figure 2(a). The x -axis indicates the steps of the algorithm. The y -axis shows the amount of mutual information loss (in bits) caused by merging the two clusters selected at the corresponding step. There is a gradual increase in information loss until a break point is reached indicating a significant loss of information. This point helps us to determine the “right” number of clusters that exist in the database. From that point on every merge causes a significant degradation of information and therefore leads to a worse clustering scenario. As can be seen from Figure 2(a) the break point is found in the transition from five clusters to four clusters, indicating that an appropriate number of clusters in this case is five. Figure 3 presents the grouping of the images database into the five clusters. The GMM model generated for each cluster is shown on the right. A clear distinction between the groups is evident in the Gaussian mixture characteristics, in blob color features and their spatial layouts. Progressing an

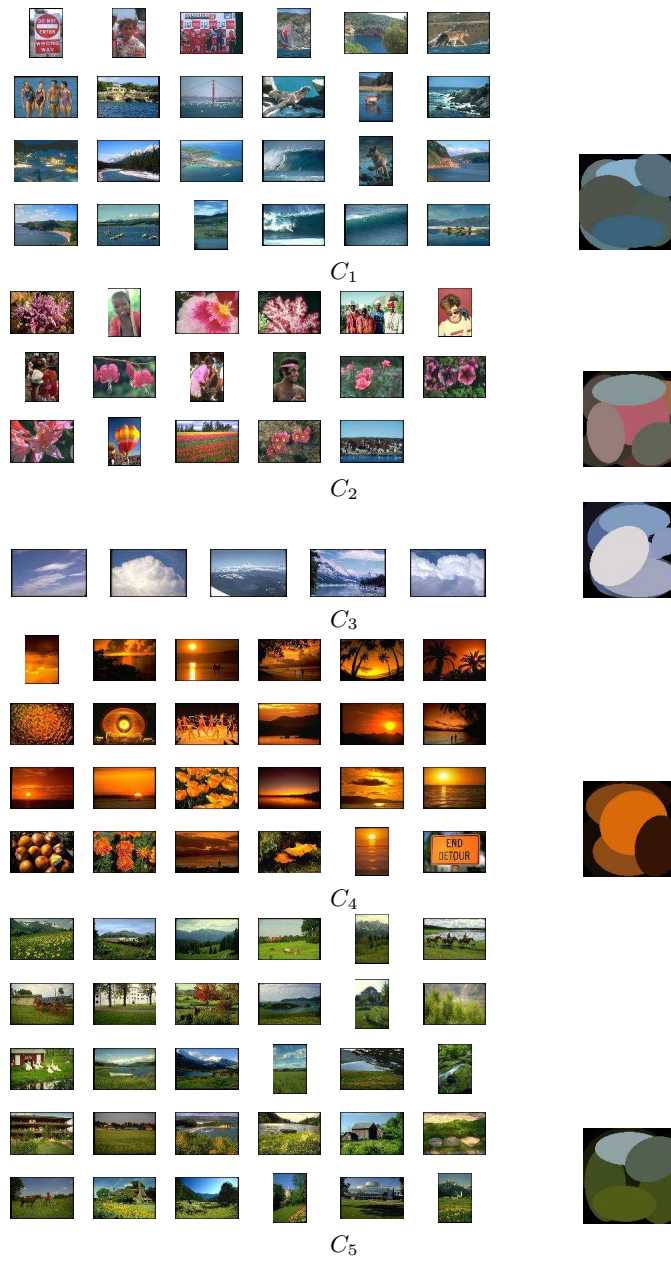


Fig. 3. Results: Clustering the database into 5 groups. A GMM model generated for each cluster is shown on the right.

additional step of the algorithm, towards four clusters, results in the merging of classes c_2 and c_3 . We note that the two classes appear rather different. The visual inhomogeneity is consistent with the significant loss of information, as indicated via the information loss function.

To evaluate the quality of grouping presented in Figure 3, a histogram of the “intra-class” distances (symmetric KL-distances between all image pairs within the same cluster) is shown in Figure 2(b), and compared with the histogram of “inter-class” distances (distances between image pairs across clusters). The x -axis is the value of the KL distance and the y -axis is the frequency of occurrence of the respective distance in each of the two distance sets. We note that the “intra-class” distances are narrowly spread at the lower end of the axis (close to zero), as compared to the wide-spread and larger distance values of the “inter-class” set, showing that the image variability within clusters is much less than the variability across clusters.

In conclusion, we have presented an unsupervised clustering scheme that is based on information theoretic principles and provides image sets for a concise summarization and visualization of the image content within a given image archive. Several limitations that need to be addressed are the limited feature space (can be extended to include texture and shape) and the use of GMM for image representation that describes the image as a set of convex regions. Future work entails utilizing the extracted clusters for efficient search and retrieval and evaluating the performance by comparing with alternative clustering techniques.

References

1. S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color and texture-based image segmentation using em and its application to content based image retrieval. In *Proc. of the Int. Conference on Computer Vision*, pages 675–82, 1998.
2. J. Chen, C.A. Bouman, and J.C. Dalton. Hierarchical browsing and search of large image databases. *IEEE transactions on Image Processing*, 9(3):442–455, March 2000.
3. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, 1991.
4. A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Statistical Soc. B*, 39(1):1–38, 1977.
5. L. Hermes, T. Zoller, and J. M. Buhmann. Parametric distributional clustering for image segmentation. In *Proceedings of ECCV02*, volume III, pages 577–591, 2002.
6. S. Krishnamachari and M. Abdel-Mottaleb. Hierarchical clustering algorithm for fast image retrieval. In *IS&T/SPIE Conference on Storage and Retrieval for Image and Video databases VII*, pages 427–435, San-Jose, CA, Jan 1999.
7. N. Slonim and N. Tishby. Agglomerative information bottleneck. In *In Proc. of Neural Information Processing Systems*, pages 617–623, 1999.