IMAGERANK : SPECTRAL TECHNIQUES FOR STRUCTURAL ANALYSIS OF IMAGE DATABASE

Xiaofei He, Wei-Ying Ma, Hongjiang Zhang*

Microsoft Research Asia, Beijing 100080, China
*Department of Computer Science, University of Chicago

ABSTRACT

Drawing on the correspondence between spectral clustering, spectral dimensionality reduction, and the connections to the Markov Chain theory, we present a novel unified framework for structural analysis of image database using spectral techniques. The framework provides a computationally efficient approach to both clustering and dimensionality reduction, or 2-D visualization. Within this framework, we can also infer the semantic degrees of the images, i.e. ImageRank, which characterize the richness of semantics contained in the images. Some illustrative examples are discussed.

1. INTRODUCTION

Image search has been an active area of research in recent years. Most research activities have been focused on image feature representation and extraction, classification, similarity measures, fast indexing and user relevance feedback mechanisms. However, an equally important problem which has received less attention is how to provide a computationally efficient framework to discover the underlying structure of an image database. Drawing on the correspondence between spectral clustering, spectral dimensionality reduction, and the connections to the Markov Chain theory, we present a novel unified framework for structural analysis of image database using spectral techniques. This framework will help to develop new techniques for ranking, clustering, visualization, and semantic analysis of images in the database.

Query-by-example has been the major search convention for many content-based image retrieval systems [3][5]. However, in many cases, it is difficult for the user to present good examples of what he is looking for without effectively browsing through the database. Hence an efficient browsing tool is essential to introduce the user to the contents of the database.

We present a statistically and geometrically motivated framework. Considering that the local structure is more important than the global structure for image distributions, we first build a nearest-neighbor graph incorporating the neighborhood information of an image database. With this graph, we can compute a two dimensional representation of the database that optimally preserves local neighborhood information in a certain sense, and cluster the images into several semantic categories, using spectral techniques. In the meantime, using the notion of Markov Chain, we can compute a stationary distribution of the Markov Chain that optimally identifies the semantic degrees of the images, i.e. *ImageRank*, which characterize the richness of semantics contained in the images. The transition matrix of the Markov Chain can be directly inferred from the nearest-neighbor graph.

The rest of this paper is organized as follows: Section 2 describes the proposed approach for computing the *ImageRank*. Section 3 describes its connections to spectral clustering and spectral dimensionality reduction. The preliminary experimental results are shown in Section 4. Finally, we give concluding remarks and future work in Section 5.

2. IMAGERANK

As we have discussed in the introduction, the primary motivation of *ImageRank* is to measure the richness of semantics of an image in a database. The formal definition of the *ImageRank* is given below:

Definition of *ImageRank*: *Given a class of images,* $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$*, with respect to a certain topic* τ *, we use ImageRank* $\pi(\mathbf{x}_i)$ *to denote the semantic degree of image* $\mathbf{x}_i$*, i.e. the richness of the semantics contained in the image* $\mathbf{x}_i$ *with respect to the topic* τ*. Without loss of generality, we let* $\sum_i \pi(\mathbf{x}_i) = 1$.

2.1 Constructing the Nearest-Neighbor Graph

Different from previous works for image browsing and navigation which focus on discovering the inherent global structure [6], our approach attaches more importance to the local structure. This is due to our finding that, image distribution in a high-dimensional space is quite disorder, hence only the local structure of images is reliable. The local structure of image distribution is always the focus of the user's attention. When two images are irrelevant, the absolute value of their Euclidean distance gives us little information about their similarity.

The first step of our framework of analysis is to construct a nearest neighbor graph. In this section, we will describe how to construct such a graph, which will be used throughout this paper. Let G denote the nearest-neighbor graph. W is its weight matrix. For any pair of images, the weight is computed according to some pre-defined similarity measure. The similarity measure is determined by the property of the feature space where the images reside on. A general formula for computing weights is as follows:

$$W_{ij} = \begin{cases} sim(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_j \text{ is among } \mathbf{x}_i\text{'s } k \text{ nearest neighbors} \\ 0 & \text{otherwise} \end{cases}$$

The main reason for defining the above formula is to preserve the local structure of the image space, rather than the global structure. Hence it is less insensitive to outliers and noise. If $W_{ij} > 0$, we assign an edge from $\mathbf{x}_i$ to $\mathbf{x}_j$. Note that, if the weight matrix W is symmetric, the graph G is undirected.

2.2 The Algorithm for Computing *ImageRank*

Given m images $\mathbf{x}_1, \ldots, \mathbf{x}_m$ in a feature space, we construct a weighted graph with m nodes, one for each image, and a set of edges connecting similar images. The algorithmic procedure is formally stated below:

1. Constructing the nearest-neighbor graph as described in Section 2.1.
2. Constructing the Markov Chain: We compute a diagonal weight matrix D, whose entries are column (or row, since W is symmetric) sums of W, i.e. $D_{ii} = \sum_j W_{ji}$. The bigger the value D_{ii} (corresponding to the i^{th} image) is, the more "importance" that image is. Here, "importance" means the image is similar to many other images so that its richness of semantics is higher. $P = D^{-1}W$ is the probability transition matrix. That is, P_{ij} is the probability of moving from image $\mathbf{x}_i$ to $\mathbf{x}_j$. P_{ij} also characterizes the semantic relationship from $\mathbf{x}_i$ to $\mathbf{x}_j$. It is easy to show that the row sum of P is 1. Let M_P denote the Markov Chain. P is also called the *normalized Laplacian* throughout this paper, due to its strong connection to the graph *Laplacian* [2].
3. Computing *ImageRank*: Assume the constructed graph G is connected. Compute the following eigenvector problem:

$$P^T \pi = \pi$$

(1) where π is a m-dimensional vector, whose i^{th} entry is just the semantic degree (or, *ImageRank*) of $\mathbf{x}_i$. Note that, π is a left-eigenvector with the eigenvalue 1 of matrix P. If graph G is not connected, we can divide it into several disjoint subgraphs, and proceed with step 3 for each subgraph.

Figure 1 gives an example to illustrate the computation of the *ImageRank* based on the random walk on the nearest neighbor graph. We can imagine that a person is walking on the image graph. At each image node, he jumps to the next image at random, with some probability determined by its similarity to the current image. That is, if image A is more similar with the current image than image B, then the person is more likely to jump to image A than to image B. Consider that the person keeps jumping, and he will arrive at the image $\mathbf{x}_i$ after infinite jumps, with probability π_i. This probability characterizes the richness of the semantics contained in the image $\mathbf{x}_i$. Intuitively, the image with the highest probability (*ImageRank*) that the person "finally" arrives at after infinite jumps can be seen as the most representative images of the image database.

2.3 Analysis of the Algorithm

Recall that given an image dataset we construct a weighted graph G with edges connecting semantically related images. We will frequently use graph-related matrices, such as the weight matrix W, transition matrix P, and their *spectra*, i.e., the set of eigenvalues and associated eigenvectors.

Theorem. If the weight matrix W is symmetric and $\mathbf{u}$ is a left eigenvector of the normalized Laplacian P, associated with a eigenvalue λ, then $D^{-1}\mathbf{u}$ is a right eigenvector of P, associated with the same eigenvalue λ. Moreover, if $\mathbf{v}$ is a right eigenvector with eigenvalue λ, then $D\mathbf{v}$ is a left eigenvector with eigenvalue λ.

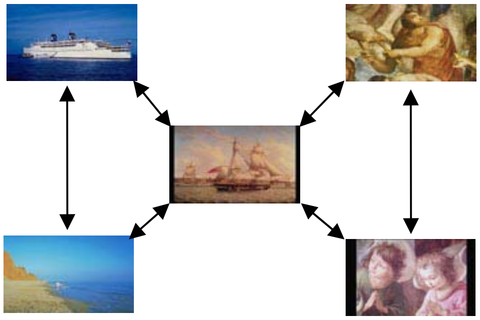

Figure 1. An illustrative example of random walk on an image graph. Note that, in this figure, we just intend to give an intuition to people about our model. For these five images, the center image has the highest *ImageRank*, hence can be viewed as the representative image.

Proof: First, we have $P=D^{-1}W$, where D is a diagonal matrix and W is symmetric. Thus, $P^T = (D^{-1}W)^T = WD^{-1} = DPD^{-1}$. Since $\mathbf{u}$ is a left eigenvector of P, we have

$$\begin{aligned} & P^T \mathbf{u} = \lambda \mathbf{u} \\ \Rightarrow\ & DPD^{-1}\mathbf{u} = \lambda \mathbf{u} \\ \Rightarrow\ & P(D^{-1}\mathbf{u}) = \lambda (D^{-1}\mathbf{u}) \end{aligned}$$

Therefore, $D^{-1}\mathbf{u}$ is a right eigenvector of P, associated with the same eigenvalue λ. If $\mathbf{v}$ is a right eigenvector with eigenvalue λ, we have

$$\begin{aligned} & P\mathbf{v} = \lambda \mathbf{v} \\ \Rightarrow\ & DP(D^{-1}D)\mathbf{v} = \lambda D\mathbf{v} \\ \Rightarrow\ & P^T (D\mathbf{v}) = \lambda (D\mathbf{v}) \end{aligned}$$

Therefore, $D\mathbf{v}$ is a left eigenvector with eigenvalue λ. □

This theorem shows that when W is symmetric, we need only to compute the right (left) eigenvector of the normalized Laplacian, and the left (right) eigenvector can be obtained correspondingly. It will become clear in the following sections that the right eigenvector of the normalized Laplacian can be used for clustering and dimensionality reduction (2-D visualization). In general, W needs not to be symmetric. However, one may take a utilitarian perspective to make W symmetric, when computational complex is a major concern.

3. CONNECTIONS TO SPECTRAL CLUSTERING AND SPECTRAL DIMENSIONAITY REDUCTION

The computation of the *ImageRank* is closely related to spectral clustering [4] and spectral dimensionality reduction [1].

The core of our computational framework is the *Normalized Laplacian P*, which is a matrix inferred from the weight matrix of the nearest-neighbor graph.

3.1 Spectral Graph Partitioning

Spectral partitioning has become one of the most successful heuristics for partitioning graphs and matrices [3]. Here we emphasize graph partition as data clustering using a graph model. Given the attributes (coordinates) of data points in a dataset and the similarity measure between any two points, the matrix containing similarities between all pairs of points forms a weighted adjacency matrix (weight matrix) of a graph. Thus the data clustering problem becomes a graph partition problem.

Throughout this paper, we index the eigenvalues of an $n\times n$ matrix in non-decreasing order: λ_0 represents the smallest eigenvalue, and λ_{n-1} the largest.

Traditional spectral partitioning methods use the *Fiedler* vector, the eigenvector of the second smallest eigenvalue of the Laplacian matrix, to find a small separator of a graph. Here, the *Laplacian*, $L(G)$, of the graph G is defined to be $L(G) = D - W$.

Recently, spectrally based clustering techniques are receiving considerable attentions. Shi and Malik [3] define the normalized cut as

$$Ncut(A,B) = Cut(A,B)\left(\frac{1}{vol(A)} + \frac{1}{vol(B)}\right)$$

$$vol(A) = \sum_{u\in A, v\in V} W(u,v)\,, \quad Cut(A,B) = \sum_{u\in A, v\in B} W(u,v)$$

where A and B are two disjoint subsets of the vertex set V, and $A\cup B = V$. The problem, as formulated by Shi and Malik, is to minimize *Ncut* over all partitions of the vertex set V. In contrast with the polynomial min-cut problem, finding the minimal normalized cut is NP-hard, but with certain approximations it reduces to a generalized eigenvector problem as follows:

$$L\mathbf{y} = \lambda D\mathbf{y}$$

where L is the *Laplacian* matrix, $L = D - W$. It is easy to show that the above eigenvector problem is equivalent to the following one:

$$P\mathbf{y} = (1-\lambda)\mathbf{y}$$

where the matrix P is the normalized Laplacian. The idea of spectral clustering is to find a splitting value s and partition the vertices of G into the set of i such that $\mathbf{y}(i) > s$ and the set such that $\mathbf{y}(i) \le s$.

3.2 Spectral Dimensionality Reduction

The images in a database can be mapped into a 2-dimensional plane by dimensionality reduction techniques, while their spatial distribution is preserved. In this section, we discuss the use of a spectral dimensionality reduction algorithm, called Laplacian eigenmap [1], for image database visualization. The main advantages of it are its locality preserving character and its inherent connections to the manifold structure which may account for the underlying topology of the image data.

The generic problem of dimensionality reduction is the following. Given a set of k points, $\{\mathbf{x}_1, \ldots, \mathbf{x}_k\}\subset R^n$, find a set of points $\{\mathbf{y}_1, \ldots, \mathbf{y}_k\}\subset R^l$ ($l << n$) such that $\mathbf{y}_i$ "represents" $\mathbf{x}_i$. Consider the problem of mapping the weighted graph G to a line so that connected points stay as close as possible. Let $\mathbf{y} = \{y_1, \ldots, y_k\}^T$ be such a map. A reasonable criterion for choosing a "good" map is to minimize the following objective function

$$\sum_{ij} (y_i - y_j)^2 W_{ij}$$

The objective function with the choice of W_{ij} incurs a heavy penalty if neighboring points $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Therefore, minimizing it is an attempt to ensure that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are "close" then y_i and y_j are close as well. The minimization problem reduces to finding

$$\underset{\substack{y \\ \mathbf{y}^T D\mathbf{y}=1}}{\arg\min}\, \mathbf{y}^T L\mathbf{y}$$

where $L = D - W$ is the Laplacian matrix. D is diagonal weight matrix; its entries are column (or row, since W is symmetric) sums of W, $D_{ii} = \sum_j W_{ji}$. The solution is given by minimum eigenvalue solution to the generalized eigenvector problem:

$$L\mathbf{y} = \lambda D\mathbf{y}$$

As can be seen, the eigenvector problem for spectral dimensionality reduction is totally the same as the spectral clustering. Hence its solutions are just the right eigenvectors of the normalized Laplacian P. We leave out the eigenvector $\mathbf{y}_0$ (= **1**) corresponding to eigenvalue 1 and use the next m eigenvectors for embedding in m-dimensional Euclidean space.

$$\mathbf{x}_i \to (\mathbf{y}_1(i), \cdots, \mathbf{y}_m(i))$$

where $\mathbf{y}_j(i)$ denote the i^{th} element of eigenvector $\mathbf{y}_j$.

4. EXPERIMENTAL RESULTS

Let us start with a simple example to demonstrate the effectiveness of our proposed approaches. Nine images from two semantic classes of Corel Image Dataset are selected for this experiment.

The topic of each class is represented by a set of keywords. Each image is manually annotated. There are totally 11 keywords (beach, sunset, man, woman, child, sky, tree, cactus, sea, wave, and boat). Therefore, each image can be represented by an 11-dimensional Boolean vector, each entry of which is associated with a keyword. By simply using the inner product as the similarity measure, we can construct a complete graph with weight matrix W. Note that, the principle eigenvector is a constant vector, i.e. **1**, thus has no discriminant information.

By using sign cut (see section 3) on $\mathbf{y}_2$, we can cluster these 9 images into two classes. The images 1,3,5,8,9 constitute the first class with the topic {beach, man, woman}, while the rest four images form the second class with the topic {sunset, man}. Beside clustering, the eigenvectors $\mathbf{y}_i$ (i = 2,3,…,l) can also be used to map the images into a k dimensional Euclidean space. That is, each image can be represented by a k-dimensional real-valued vector. Here, we map these images into a 2-dimensional plane, as figure 2 shows.

As we described in Section 2, the normalized Laplacian P is also a transition matrix of Markov Chain M_P. By solving equation (1), we get the stationary distribution of M_P, i.e., *ImageRank*, as follows:

$$\pi = (0.116, 0.151, 0.151, 0.035, 0.07, 0.035, 0.14, 0.151, 0.151)$$

These 9 images can be sorted according to their *ImageRank*: (2,3,8,9);(7);(1);(5);(4,6). As can be seen, the higher *ImageRank* the image has, the richer semantics the image contains. Thus, the images 2,3,8,9,7 in the circle can be used to represent the image dataset.

4.1 Structural Analysis of Image Dataset Using Low-Level Feature

In the above example, we use keywords to represent an image. In this example, we use color correlagram automatically extracted from the image content to represent an image. 80 images from 4 semantic classes are used. Each class contains 20 images. For each image, 15 nearest neighbors are found to construct the weighted graph. The similarity measure is defined as follows:

$$W_{ij} = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / t)$$

where t is a suitable constant. When such kind of similarity measure is used, we are intrinsically discovering the manifold structure of the image space, see [1] for details. We can then compute the normalized Laplacian P from the weight matrix W. Using the right eigenvectors of P, we can cluster these images into 2 classes. For each class, we use the same approach to cluster it into 2 subclasses. Finally, we get the results as figure 3 shows.

In each class, three images with the highest ImageRank are selected as the representative images (with blue frame). As can be seen, the selected representative images of each class can reflect the semantics of that class to some extent. We believe that, as the image features used for the structural analysis get more accurate, our clustering performance as well as the obtained ImageRank will be more accurate. Figure 4 shows 2-D visualization result.

5. CONCLUSIONS

In this paper, we presented a unified framework for structural analysis of image database using spectral techniques, drawing on the correspondence between spectral clustering, spectral dimensionality reduction, and the connections to Markov Chain. Within this framework, we proposed a novel approach to compute the semantic degrees of images, i.e. *ImageRank*, which characterizes the richness of semantics contained in the images. The *ImageRank* can be used to determine the most representative images in an image database. The statistical justification of *ImageRank* stems from the Markov China theory. As a result, the *ImageRank* is computed as the stationary distribution of the Markov Chain.

In this paper, the normalized Laplacian P is inferred from the weight matrix of the nearest-neighbor graph, which models the local structure of the image distribution. The normalized Laplacian P is the core of *ImageRank*, spectral clustering, and spectral dimensionality reduction. These spectral techniques together provide us a computationally efficient tool to analyze the structure of an image database for clustering and dimensionality reduction, or 2-D visualization.

REFERENCES

[1] M. Belkin and P. Niyogi "Laplacian eigenmaps and spectral techniques for embedding and clustering". *NIPS*, 2001.

[2] Fan R. K. Chung "Spectral Graph Theory". Regional conference series in mathematics, no 92, American Mathematical Society, Providence, RI, 1997.

[3] Y. Rui, T.S. Huang, and S.-F. Chang, "Image retrieval: current techniques, promising directions and open issues", *Journal of Visual Communication and Image Representation,* Vol. 10, 39-62, March, 1999.

[4] J. Shi and J. Malik, "Normailized cuts and image segmentation". *IEEE trans. On PAMI* 22(8), 888-905, 2000.

[5] W. Y. Ma and B. S. Manjunath, "Netra: A toolbox for navigating large image databases", *ACM Multimedia System,* vol. 7, pp. 184-198, 1999.

[6] Y. Rubner, "Perceptual metrics for image database navigation", Ph.D. dissertation, Stanford University, 1999.

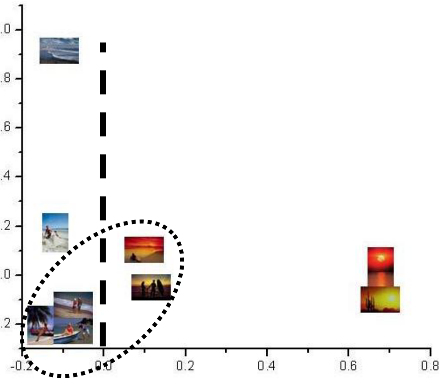

Figure 2. 2-D visualization of the 9 images, which are clustered into two classes divided by the dash vertical line. The 5 images in the circle can be selected as the representative images for each class.

Figure 3. The 80 images are clustered into 4 classes. In each class, three images are selected as the representative images according to their ImageRank.

Figure 4. 2-D visualization of the image database.