Global Journal of
Information Technology

Volume 06, Issue 1, (2016) 11-17

http://sproc.org/ojs/index.php/gjit

Kernel principal component analysis for multimedia retrieval

Guang-Ho Cha *, Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX 77030, USA.

Suggested Citation:
Cha, G.H. (2016). Kernel principal component analysis for multimedia retrieval. *Global Journal of Information Technology*. *6*(1), 11-17.

Received 25 January, 2016; revised 18 February, 2016; accepted 10 March, 2016.
Selection and peer review under responsibility of Prof. Dr. Adem Karahoca, Bahcesehir University, Turkey

Abstract

Principal component analysis (PCA) is an important tool in many areas including data reduction and interpretation, information retrieval, image processing, and so on. Kernel PCA has recently been proposed as a nonlinear extension of the popular PCA. The basic idea is to first map the input space into a feature space via a nonlinear map and then compute the principal components in that feature space. This paper illustrates the potential of kernel PCA for dimensionality reduction and feature extraction in multimedia retrieval. By the use of Gaussian kernels, the principal components were computed in the feature space of an image data set and they are used as new dimensions to approximate image features. Extensive experimental results show that kernel PCA performs better than linear PCA with respect to the retrieval quality as well as the retrieval precision in content-based image retrievals.

Keywords: Principal component analysis, kernel principal component analysis, multimedia retrieval, dimensionality reduction, image retrieval

*ADDRESS FOR CORRESPONDENCE: **Guang-Ho Cha,** Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX 77030, USA. *E-mail address*: ghcha@snut.ac.kr

1. Introduction

An explosion in the amount of digital image data has brought about the need for image retrieval systems. This demand has made image retrieval a very active re-search area in recent years. Content-based image retrieval (CBIR) supports image searches based on visual features such as color, texture, and shape. In a CBIR system, these features are extracted and stored as feature vectors. During the retrieval process, the feature vector of the query image is computed and matched against those in the database. The returned images should be similar to the query image. This similarity (or nearest neighbor) indexing/retrieval problem can be solved efficiently when the feature vectors have low or medium dimensionalities (e.g., less than 8) by the use of existing indexing methods such as the R*-tree [1] and the HG-tree [2]. So far, however, there has been no efficient solution to this problem when the feature vectors have high dimensionalities, say over 100. In fact, for a high dimensionality, in theory or in practice, the performance of existing indexing methods degenerates to being worse than that of the brute-force sequential scan that compares the query object to each data object [3]. So the issue is to overcome the curse of dimensionality – a phenomenon that the retrieval performance degrades drastically as the dimensionality increases. Motivated by the dimensionality curse, approaches to reduce the dimensionality of image feature vectors have been attempted by the use of some dimensionality reduction techniques such as principal component analysis [4, 7].

Principal component analysis (PCA) [5] has been widely used for re-expressing multidimensional data. It allows researchers to reorient the data so that the first few dimensions account for as much of the available information as possible. If there is substantial redundancy present in the data set, then it may be possible to account for most of the information in the original data set with a relatively small number of dimensions. In other words, PCA finds out for the original data set the new structure given by the linear combination of the original variables. However, one cannot assert that linear PCA will always detect all structure in a given data set. By the use of suit-able nonlinear features, one can extract more information. In this paper, we investigate the potential of a nonlinear form of PCA for dimensionality reduction and feature extraction in content-based image retrieval.

In the next section, we review the standard linear PCA and introduce the kernel PCA that is a nonlinear extension of PCA. In Section 3, we provide the process of the dimensionality reduction and feature extraction using the kernel PCA. In Section 4, we discuss the measures to assess the performance of the kernel PCA. Section 5 presents experimental results to compare the performances of the kernel PCA and the linear PCA. In Section 6, we conclude with a discussion of the significance of the work.

2. Kernel PCA

PCA is an orthogonal basis transformation. The new basis is found by diagonalizing the covariance matrix C of a centered data set $\{x_i \in R^N \mid i = 1, \ldots, m\}$, defined by

$$C = \frac{1}{m}\sum_{j=1}^{m} x_j x_j^t, \quad \sum_{j=1}^{m} x_j = 0$$

The coordinates in the Eigenvector basis are called principal components. The principal components are given by the linear combination of the original variables. The size of an Eigenvalue λ corresponding to an Eigenvector v of C equals the amount of variance in the direction of v. Furthermore, the directions of the first n Eigenvectors corresponding to the biggest n Eigenvalues cover as much variance as possible by n orthogonal directions. In many applications, they contain the most interesting information, for example, in data compression, where we project onto the directions with biggest variance to retain as much information as possible, or in data de-noising, where we deliberately drop directions with small variance.

Clearly, one cannot assert that linear PCA will always detect all structure in a given data set. Moreover, it can be very sensitive to “wild” data (“outliers”). By the use of suitable nonlinear features, one can extract more information. Kernel PCA is very well suited to extract interesting nonlinear structures in the data [13].

The purpose of our work is therefore to consider the potential of kernel PCA for dimensionality reduction and feature extraction in CBIR. Kernel PCA first maps data into some feature space F via a (usually nonlinear) function Φ and then performs linear PCA on the mapped data. As the feature space F might be very high dimensional (e.g., when mapping into the space of all possible d-th order monomials of input space), kernel PCA employs Mercer kernels instead of carrying out the mapping Φ explicitly. A Mercer kernel is a function $k(x, y)$ which for all data sets $\{\boldsymbol{x}_i\}$ gives rise to a positive matrix $K_{ij} = k(x_i, y_j)$ [11]. Using function k instead of a dot product in input space corresponds to mapping the data with some Φ to a feature space F, i.e., $k(x, y) = (\Phi(x) \cdot \Phi(y))$. Fig. 1 shows the basic idea of the kernel PCA. After mapping the original data onto some high-dimensional feature space F via Φ (right in Fig.1(b)), we perform linear PCA, just as a PCA in input space (Fig.1(a)). Since F is nonlinearly related to input space (via Φ), the contour lines of constant projections onto the principal Eigenvector (drawn as an arrow) become nonlinear in input space.

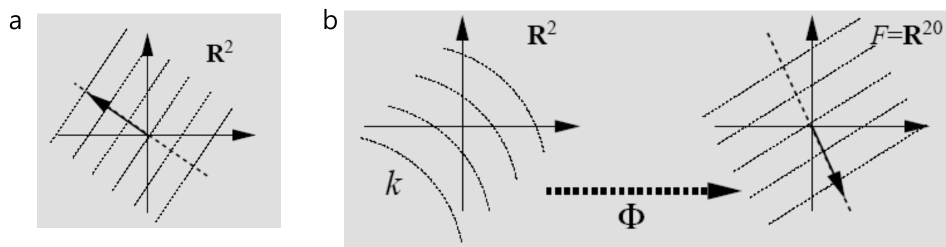

Fig. 1. The basic idea of kernel PCA: (a) linear PCA; (b) kernel PCA

To perform PCA in feature space F, we need to find Eigenvalues $\lambda > 0$ and Eigenvectors $v \in F - \{0\}$ satisfying $\lambda v = C' v$ with the covariance matrix C' in F, defined as

$$C' = \frac{1}{m}\sum_{j=1}^{m}\Phi(x_j)\Phi(x_j)^t, \quad \sum_{j=1}^{m}\Phi(x_j) = 0$$

Substituting C' into the Eigenvector equation, we note that all solutions $\boldsymbol{v}$ must lie in the span of Φ-images of the sample data. This implies that we can consider the equivalent equation

$$\lambda(\Phi(x_j)\cdot v) = (\Phi(x_j)\cdot C'v) \quad \text{for all } j = 1, \ldots, m \tag{1}$$

and that there exist coefficients $\alpha_1, \ldots, \alpha_m$ such that

$$v = \sum_{j=1}^{m}\alpha_j \Phi(x_j)\cdot \tag{2}$$

Substituting C' and (2) into (1), and defining $m \times m$ Gram matrix $K_{ij} = k(\Phi(x_i), \Phi(y_j)) = k(x_i, y_j)$, we arrive at a problem which is cast in terms of dot product. Solve

$$m\lambda\alpha = K\alpha$$

for nonzero Eigenvalues λ and Eigenvectors $\alpha = (\alpha_1, \ldots, \alpha_m)^t$ subject to normalization condition $\lambda^k(\alpha^k \cdot \alpha^k) = 1$. To extract nonlinear principal components for the Φ-image of a test point x, we compute the projection onto the k-th component by

$$v^k \cdot \Phi(x) = \sum_{j=1}^{m} \alpha_j^k k(x, x_j) = \sum_{j=1}^{m} \alpha_j^k \Phi(x)\Phi(x_j) \cdot$$

3. Feature Extraction and Dimensionality Reduction

Fig. 2 shows the architecture of kernel PCA for image feature extraction, which involves three layers with entirely different roles. The input layer is made up of source nodes that connect the kernel PCA to its environments. The input x and the expansion patterns x_i are nonlinearly mapped via Φ into a feature space F where dot products are computed. Through the use of the kernel k, these two layers are in practice computed in one step. The results are linearly combined using weight α_i^l, found by solving an Eigenvalue problem, resulting in the l-th nonlinear principal component corresponding to Φ. Therefore the first p principal components (assuming that the Eigenvectors are sorted in descending order of their Eigenvalue size) constitute the p-dimensional feature vector for an image. By selecting the proper kernels k, various mappings Φ can be indirectly induced. In our work, we use the Gaussian kernel $k(x, x') = \exp(-(||x–x'||^2/2\sigma^2))$ because it is widely used in content-based image retrieval and pattern recognition [9, 14].

Unlike linear PCA, kernel PCA allows the extraction of a number of principal components

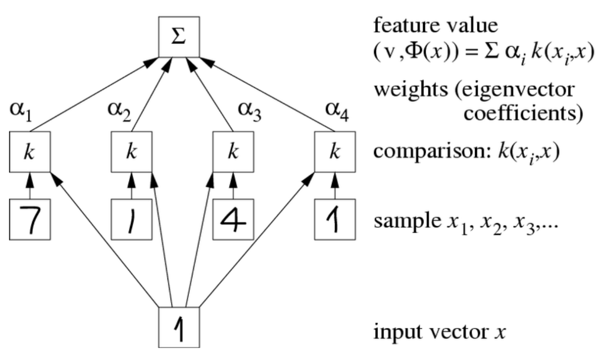

Fig. 2. Image feature extraction architecture with kernel PCA

which can exceed the input dimensionality because it diagonalizes the $m \times m$ Gram matrix K, $K_{ij} = k(x_i, y_j)$, instead of the covariance matrix C. However, images are not randomly distributed in high-dimensional feature space. As an instance, our image database consists of 13,724 256-color images of U.S. stamps and photos. Stamps often come in series (e.g., states, birds, flowers) with common colors and related designs, and the U.S. Post Office has often used similar colors for many long-running stamps. As a result, this real image data set shows highly clustered distribution. In other words, the dimensionality of the feature space may be much smaller than that of the input space, and it is appropriate to reduce the dimensionality of the original image data. In our work, we retain a sufficient number of principal components so that we can account for at least 80% of the variance in each original variable.

4. Feature Extraction and Dimensionality Reduction

In this section, we consider the measures to assess the performance of kernel PCA. In traditional (document) information retrieval, performance is often measured by using precision and recall [10, [12]. Recall measures the ability of the system to retrieve useful items, while precision measures its ability to reject useless items. For a given query, let T the total number of relevant items available, R_r the number of relevant items retrieved, and T_r the total number of retrieved items. Then precision is defined as R_r / T_r, and recall as R_r / T.

Precision and recall can also be applied to image retrieval. In IBM QBIC that performs similarity retrieval as opposed to exact match, *normalized* precision and recall have been suggested [6]. These reflect the positions in which the set of relevant items appear in the retrieval sequence (ordered by some similarity measure). If there are T relevant images in the database, then for an ideal retrieval, all T relevant items occur in the first T retrievals (in any order). Faloutsos *et al.* [6] define this as IAVRR, the ideal AVRR (average rank of all relevant, retrieved images). It is the maximum when all relevant images are retrieved on the top: IAVRR = $(0 + 1 + \ldots + (T-1)) / T$ (where the first position is the 0-th). The ratio of AVRR to IAVRR gives a measure of the effectiveness of the retrieval. In an ideal case of retrieval, this ratio would be 1.

For example, if the relevant images for the query image A are defined as:

A46, A18, A101, A52, A35, A102

so that $T = 6$, and a CBIR system returns, in order:

A102, A109, A50, **A18**, A74, **A46**, **A52**, A57, A17, **A35**, A63, A16, A58, **A101**

then relevant items appear at 0, 3, 5, 6, 9 and 13. The AVRR for this is therefore (0 + 3 + 5 + 6 + 9 + 13)/6 = 6. The IAVRR would be (0 + 1 + 2 + 3 + 4 + 5)/6 =2.5. Thus AVRR / IAVRR is 2.4.

If the order of retrieval matters, Kendall's tau can be used to provide measures of association between two sets of data [8]. Kendall's tau can be viewed as a coefficient of disorder. For example, consider the following two rankings, where both have selected the same four images, but have placed them in a different order:

1 2 3 4
2 1 4 3

Tau is calculated as

(no. of pairs in order – no. of pairs out of order) / (total no. of possible pairs)

For this example, 2 in the bottom row is followed by 1, 4, and 3, 2-1 is out of order, scoring -1, and 2-4, 2-3 are in order, scoring +1 each. Similarly, 1 is followed by 4 and 3. Both are in order, scoring +1 each. Finally, 4 is followed by 3, scoring -1. The number of in-order pairs is four, and out-of-order pairs is two, therefore the total is +2, divided by the maximum number of in-order pairs, $N(N-1)/2$, which here is 6, since $N = 4$. The value of tau is therefore 2/6, or 0.3333. This gives a measure of the "disarray" or difference in ranking, between the two. It ranges from -1, which represents complete disagreement, through 0, to +1, complete agreement.

For each experiment in our work, we report the average of the above measures over 100 k nearest neighbor (k-NN) queries. For k-NN queries, precision and recall are the same because $T = T_r$, i.e., the total number of relevant items and the total number of retrieved items are the same. There we compute only the precision measure as a representative. The ratio of AVRR to IAVRR gives a measure that how much the results are close to the top. Kendall's tau provides a measure of the order for the k-NN search results.

5. Experimental Results

To demonstrate the effectiveness of kernel PCA, we performed an extensive experimental evaluation for kernel PCA and compared it to linear PCA. For our experiments we used 13,724 256-color images of U.S. stamps and photos. To obtain feature vectors for experiments, we used four MPEG-7 visual features: (1) color structure descriptor (256 dimensions), (2) homogeneous texture descriptor (30 dimensions), (3) edge histogram descriptor (80 dimensions), and (4) region-based shape descriptor (35 dimensions). These descriptors are general descriptors that can be used in CBIR. The color structure descriptor represents an image by both the color distribution of the image (similar to a color histogram) and the local spatial structure of the color. The homogeneous texture descriptor characterizes the region texture using the mean energy and the energy deviation from a set of frequency channels. The edge histogram descriptor represents local-edge distribution in the image. The region-based shape descriptor takes into account all pixels constituting the shape, that is, both the boundary and interior pixels.

We applied kernel PCA and linear PCA to those four data sets consisting of MPEG-7 visual descriptors, respectively, in order to reduce their dimensionality. We posed k nearest neighbor queries to 3 kinds of data sets, i.e., (1) the original data set whose dimensionality is not reduced, (2) the data set whose dimensionality is reduced by kernel PCA, and (3) the data set whose dimensionality is reduced by linear PCA. In all experiments, the numbers of nearest neighbors to find were 20, 40, 60, 80 and 100 and we averaged their results. 100 random *k*-NN queries were processed in each experiment and the results were averaged.

Table 1. Color structure experiments

original	Linear PCA			Kernel PCA		
dim = 256	Precision	AVRR/IAVRR	Tau	Precision	AVRR/IAVRR	Tau
95% (*dim* = 164)	0.62	3.45	0.42	0.73	3.05	0.70
90% (*dim* = 113)	0.59	3.65	0.37	0.67	3.17	0.62
85% (*dim* = 77)	0.53	3.70	0.38	0.64	3.25	0.63

Table 2. Homogeneous texture experiments

original	Linear PCA			Kernel PCA		
dim = 30	Precision	AVRR/IAVRR	Tau	Precision	AVRR/IAVRR	Tau
95% (*dim* = 14)	0.60	3.50	0.41	0.71	3.15	0.69
90% (*dim* = 10)	0.56	3.75	0.37	0.66	3.30	0.64
85% (*dim* = 9)	0.51	3.80	0.36	0.62	3.45	0.61

Table 3. Edge histogram experiments

original *dim* = 80	Linear PCA			Kernel PCA		
	Precision	AVRR/IAVRR	Tau	Precision	AVRR/IAVRR	Tau
95% (*dim* = 30)	0.59	3.48	0.45	0.69	3.27	0.61
90% (*dim* = 25)	0.54	3.45	0.42	0.64	3.27	0.60
85% (*dim* = 22)	0.50	3.68	0.39	0.61	3.35	0.58

Tables 1 – 4 show that the experimental results for four MPEG-7 visual features. The first column of each table represents the dimensionality of the original data and the dimensionalities of the transformed data after the dimensionality reduction. In addition, the percentages of the variance in each original variable we retain are also provided in the first column of each table. The performance of kernel PCA is better than that of linear PCA with respect to all three performance parameters, i.e., precision, the ratio of AVRR to IAVRR, and Kendall's tau. In terms of precision and AVRR/IAVRR, kernel PCA is 10% – 20% better than linear PCA. With respect to Kendall's tau, kernel PCA is better than linear PCA more than 50%. These experimental results indicate that kernel PCA can be successfully employed as a generalized nonlinear extension of linear PCA.

6. Conclusion

In this paper, we described the potential of kernel PCA for dimensionality reduction and feature extraction in content-based image retrieval. Through the use of Gaussian kernel, a kernel PCA was able to work effectively within the feature space of image data sets, thereby producing a good performance. Compared with linear PCA, kernel PCA showed better performance with respect to the retrieval quality as well as the retrieval precision in content-based image retrieval. Therefore, we can conclude that kernel PCA can be successfully employed as a generalized nonlinear extension of linear PCA.

Acknowledgements

This study was financially supported Seoul National University of Science and Technology.

References

[1] Beckmann, N., Kriegel, H.-P., Schneider, R., & Seeger, B. (1990). The R*-tree: An efficient and robust access method for points and rectangles. *Proc. of ACM SIGMOD Conf.* 322-331.

[2] Cha, G.-H., & Chung, C.-W. Chung (1998). A New Indexing Scheme for Content-Based Image Retrieval, *Multimedia Tools and Applications*, *6*, 263-288.

[3] Cha, G.-H., Zhu, X., Petkovic, D., &Chung, C.-W. (2002). An Efficient Indexing Method for Nearest Neighbor Searches in High-Dimensional Image Databases. *IEEE Transactions on Multimedia*. 4, 76-87.

[4] Chakrabarti, K., & Mehrotra, S. (2000). Local Dimensionality Reduction: A New Approach to Indexing High Dimensional Spaces, *Proc. of the Int'l Conf. on VLDB*., 89-100.

[5] Diamantaras, K.I., & Kung, S.Y. (1996). *Principal Component Neural Networks*, Wiley, New York.

[6] Faloutsos, C. et al. (1994). Efficient and Effective Querying by Image Content. *Journal of Intelligent Information Systems*, *3,* 231-262.

[7] Kanth, K.V.R., Agrawal, D., & Singh, A. (1998). Dimensionality Reduction for Similarity Searching in Dynamic Databases, *Proc. of the ACM SIGMOD Conf.*, 166-176. 8.

[8] Payne, J., Hepplewhite, L., & Stonham, T. J. (2000). Texture, Human Perception and Information Retrieval Measures, *Proc. of ACM SIGIR 2000 Workshop*.

[9] Peng, J., & Heisterkamp, D. R. (2003). Kernel indexing for Relevance Feedback Image Retrieval, *Proc. of IEEE Int'l Conf. on Image Processing*, 733-736. 10.

[10] Rijsbergen, C.J.V. (1979). *Information Retrieval*. Butterworths, London.

[11] Saitoh, S. (1988), *Theory of Reproducing Kernels and its Applications*. Longman Scientific & Technical, Harlow, England.

[12] Salton, G., & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*, McGraw-Hill, New York.

[13] Schölkopf, B., Smola, A., & Müller, K. (1998). Nonlinear Component Analysis as a Kernel Eigenvalue Problem, *Neural Computation*, *10*, 1299-1319.

[14] Schölkopf, B. et al. (1997). Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers, *IEEE Transactions on Signal Processing*, *45,* 2758-2765.