A Kernel View of Spectral Point Pattern Matching

Hongfang Wang and Edwin R. Hancock

Dept. of Computer Science, University of York
Heslington, York, YO10 5DD, UK
{hongfang,erh}@cs.york.ac.uk

Abstract. This paper investigates spectral approaches to the problem of point pattern matching. Specifically, kernel principle component analysis (kernel PCA) methods are studied and compared with Shapiro and Brady's approach and multidimensional scaling methods on both synthetic data and real world data. We demonstrate that kernel methods can be effectively used for solving the point correspondence matching problem with a performance that is comparable with other iterative-based algorithms in the literature under the existing of outliers and random position jitter. We also provide discussion of the theoretical support from kernel PCA to the earlier approach of Shapiro and Brady.

1 Introduction

The problem of point pattern matching is to find one-to-one correspondences among two given data-sets and serves as an important part in many computer vision tasks. Graph spectral methods have been used extensively for locating correspondences between feature point-sets, e.g. [8, 9]. In [8], Scott and Longuet-Higgins first use a Gaussian weighting function to build an inter-image proximity matrix between feature points in different images being matched and then perform singular value decomposition on the obtained matrix in order to get correspondences from the proximity matrix's singular values and vectors. This method fails when rotation or scaling between the images is too large. To overcome this problem, Shapiro and Brady [9] construct intra-image proximity matrices for the individual point-sets being matched with an aim to capturing relational image structure. The eigenvectors of the individual proximity matrices are used as the columns of a modal matrix. Correspondences are located by comparing the rows of the modal matrices for the point-sets under match. This method can be viewed as projecting the individual point-sets into an eigenspace, and seeking matches by looking for closest point correspondences. Carcassoni and Hancock have attempted to improve the robustness of this method to point-jitter using robust error kernels instead of the Gaussian [2] and have overcome problems due to differences in the structure of the point-sets by using spectral clusters [3]. Multidimensional scaling is also used to solve this problem by performing Procrustes alignment in the eigenspace [6]. However, these two latter approaches involve iterative computing which requires more computation than other approaches.

A. Fred et al. (Eds.): SSPR&SPR 2004, LNCS 3138, pp. 361–369, 2004.

This motivates us to seek point matching algorithms that are both robust and without iteration. Kosibov and Caelli [1] have extended the Shapiro and Brady method of seeking correspondences by searching for matches that maximise the inner product of the truncated and re-normalised eignevactors.

The idea underpinning these spectral methods is to embed point-sets into a common eigenspace, and to find correspondences by performing alignment in this space. The key in this idea is that of finding the appropriate function which captures the essential properties of the given data-set which should also be robust under uncertainties such as outliers, random position jitter, occlusions, etc., and identifying the common eigenspace. Also the captured properties should be common in both data sets. The problem of how to select the best function, is a topic that has recently attracted considerable interest in kernel learning theory. The development of kernel PCA [7] provides us with a theoretically sound way of improving the existing spectral point pattern matching algorithms since it shares many features in common with spectral graph theory.

Our aim in this paper is to investigate the performance of kernel PCA for solving the point correspondence problem and provide a robust one-to-one point pattern matching algorithm which involves no iterations. We focus in detail the Gaussian and polynomial kernels which are invariant to similarity transformations and reflection, and compare their performance in point pattern matching with previous approaches. A common weakness with existing spectral methods is that they are particularly sensitive to structural variations in the point-sets. We demonstrate that the kernel approach is a feasible way for point pattern matching and with an appropriate kernel function, in this work the polynomial kernel, encouraging performance can be obtained and the results are less sensitive to these problems than the previous graph spectral methods.

2 Spectral Point Pattern Matching

The problem of point pattern matching can be described as given two feature point-sets $X_1 = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ and $X_2 = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ extracted from two different images, establish a one-to-one point correspondences between the two data-sets. Ideally, outliers can be removed from the data-sets during matching. In this paper, the feature points in each data-set are in the form of $\mathbf{x}_i = (x_i^{(1)}, x_i^{(2)})$ and $\mathbf{y}_i = (y_i^{(1)}, y_i^{(2)})$, respectively, where i are the indices and superscripts (1) and (2) represent each point's respective abscissa and ordinate. Our aim is to locate correspondences between the two point-sets.

The approaches of graph spectral methods for point pattern matching is to solve the point correspondence problem by first build a graph representation for each data-set where each graph node corresponds to an image feature point, and each edge between nodes corresponds to the relationships of the two feature points. After the graph construction, represent each graph by a matrix and find feature correspondences from the matrices' eigendecompositions. These methods aim to embed the dissimilarity (or similarity) properties of the original data into a common space in which correspondence matching can be performed. As

mentioned in the last section, the two essential ingredients are the dissimilarity function and the embedding procedure. The dissimilarity properties are regarded as weights of the edges and are expressed in the form of a proximity matrix A with its elements A_{ij} represents the dissimilarity relationship between feature points $\mathbf{x}_i$ and $\mathbf{x}_j$.

The objects in image frames are usually subject to transformations such as translation, rotation, scaling, and reflection. Hence, it is desirable for the dissimilarity function to be invariant under these transformations and thus provide a basis for directly comparing images and for finding correspondences between the original feature points and their transformed counterparts. It is known from geometry that the Euclidean distance is invariant to any similarity transformation, so the dissimilarity functions underpinning many existing methods are related to the Euclidean distance between feature points (see for example, [9, 8]). Another example of similarity transformation-invariant property is the directional properties of feature points. This property can also be considered as a good candidate for constructing a suitable similarity function for spectral point matching.

When viewed from the perspective of kernel PCA, applying a dissimilarity or similarity function to the original data set is equivalent to the process of using a kernel function to map the data into a higher, possibly infinite, dimensional space. Moreover, this mapping interpolates the data in the new space according to their transformation invariant properties. From this perspective, we believe that kernel PCA provides us a sound theoretical explanation for spectral pattern matching, and by applying an appropriate kernel function, expected matching results should be obtained.

3 The Kernel Approach

Kernel PCA can be regarded as a generalization of PCA from a linear to a nonlinear transformation space. In the literature it has been shown to provide a better way of recovering the underlying principal components of the given data.

Conventional principal component analysis (PCA) provides an orthogonal transformation of the data from a high dimensional space to a low dimensional one which maximally preserves the variance of the original data. This is done by computing the eigenvalues and eigenvectors of the covariance matrix $\mathbf{C} = \frac{1}{M}\sum_{i=1}^{M}(\mathbf{x_i}-\overline{\mathbf{x}})(\mathbf{x}_i-\overline{\mathbf{x}})^T$, and then use the first N normalized eigenvectors ($N \leq M$, assume the eigenvalues are sorted in descending order) of the covariance matrix as the main projection axes for the training data. Since the method minimizes the residual covariance of the data points projected into the common eigen-subspace, it thus gives an optimum representation of the original data in the projection space.

The main difference between kernel PCA and conventional PCA is that kernel PCA first uses a function $\mathbf{T} : \mathbf{x} \mapsto \mathbf{\Phi}(\mathbf{x})$ to map the data from the low dimensional space into a new feature space F of higher dimension. Conventional PCA is then performed on the transformed data matrix. This gives kernel PCA the property of extracting nonlinear features from the data-set and makes it a powerful tool in many applications.

However, an explicit mapping $\mathbf{T}$ is not always exist. In real practices the mapping is implicitly done by choosing a suitable kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ for data points $\mathbf{x}_i$ and $\mathbf{x}_j$. However, there is a problem when choosing the function $K(\mathbf{x}_i, \mathbf{x}_j)$ since not every function is guaranteed to satisfy the requirements of a feature space. An approach of choosing a qualified kernel function is to use the properties described in the Mercer's theorem [10] which states that any continuous symmetric function $K(\mathbf{x}_i, \mathbf{x}_j)$ that satisfies the positive semidefinite condition $\int_{X\times X} K(\mathbf{x}_i, \mathbf{x}_j) f(\mathbf{x}_i) f(\mathbf{x}_j) d\mathbf{x}_i d\mathbf{x}_j \geq 0$ is ensured to be a kernel for some feature space. This provides a broad way of choosing the kernel mapping functions. In this paper, we study the Gaussian kernel and the polynomial kernels in more detail for reasons described in the last section.

To extract the principal components of the mapped data, first a covariance matrix needs to be constructed for the mapped data. Suppose that the data $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ in space F is centred, then the covariance matrix of the mapped data in this space is:

$$\overline{\mathbf{C}} = \frac{\mathbf{1}}{\mathbf{m-1}} \sum_{\mathbf{i=1}}^{\mathbf{m}} \mathbf{\Phi(x_i)\Phi(x_i)^T}$$

Since the explicit mapping $\mathbf{T}$ is probably unknown, computing the covariance matrix directly is not feasible. Schölkopf, Smola, and Müller showed in [7] that by solving the eigen-equation $m\lambda\alpha = K\lambda$ in which the eigenvalues are $m\lambda$, the p_{th} feature vector, corresponding to the projection of the p_{th} feature point on the eigenspace, takes the form $< v^p, \Phi(\mathbf{x}) >= \frac{1}{\sqrt{\lambda^p}} \sum_{i=1}^{m} \alpha_i^p k(\mathbf{x}_i, \mathbf{x})$, which can be further simplified to ([5])

$$< v^p, \Phi(\mathbf{x}) >= \frac{1}{\sqrt{\lambda^p}} (K\alpha^p)_n = \sqrt{\lambda^p}\alpha_n^p \tag{1}$$

To generalize the method to non-centered data, the kernel function K becomes [7, 5] $K' = (I - ee^T)K(I - ee^T)$ where $e = M^{-1/2}(1, 1, \ldots, 1)^T$.

Based on the interesting transformation invariants, two kernel functions, the Gaussian kernel and the polynomial kernel, are of interest in this work. The polynomial kernel has the form $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + c)^d$, where c and d are constants ($d \neq 0$). It captures the directionality of the data which should be very important for correspondence matching. However, the dot product is not invariant under object's scaling so there is still a magnitude problem we should consider. To solve this problem, one way is to normalize the scaled and truncated eigenvectors. Another method is to scale both of the two eigenvector matrices by the eigenvalue matrix of the model data-set. In this work, the latter method is chosen to eliminate this problem.

The Gaussian kernel has the form $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-d_{ij}^2/\sigma\}$ where d_{ij}^2 is a dissimilarity metric between the points $\mathbf{x}_i$ and $\mathbf{x}_j$, which usually takes the form of a Euclidean distance between these two points as used in this work.

Having the kernel functions described, we now introduce our point matching algorithm using kernel PCA techniques. We expect that a suitable kernel function will capture the object's properties and feature points can be embedded

into a lower dimensional feature space in terms of the extracted properties thus provide a basis for one-to-one correspondence matching. In short, the following procedures are taken for performing point matching by kernel PCA:

- Build a matrix representation A for each image, where the matrix elements are computed by $A_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$, and $\mathbf{x}_i$ and $\mathbf{x}_j$ are two feature points from the same image and also let the matrix A be centred;
- Perform the eigendecomposition of A: $A\lambda = \lambda\alpha$, with λ the eigenvalues and α the eigenvectors for each proximity matrix;
- Use equation (1) to compute the projection of each feature in the eigenspace spanned by α. Only the first two eigenvectors are used in the polynomial kernels, and only the first three are used in the Gaussian kernels;
- The correspondences between two feature point-sets are the pairs which have the smallest Euclidean distance between them.
- The σ values in the Gaussian function are chosen automatically using the heuristic formula $\sigma = \frac{1}{0.09}\sum_{i=1}^{m}(\frac{1}{m}\sum_{j=1}^{m} d_{ij}^2)^2$, where d_{ij} are the Euclidean distances between point pairs $\mathbf{x}_i$ and $\mathbf{x}_j$.

One may find the above algorithm somewhat similar to the approach by Shapiro and Brady [9]. In [9], a proximity matrix A is first built for each image with the matrix elements computed as $A_{ij} = \exp\{-d_{ij}^2/2\sigma^2\}$, where d_{ij}^2 is the Euclidean distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$, and σ is an adjustable parameter. Shapiro and Brady explain this as the mapping of the original two dimensional data to a higher dimensional space and thus capture structural information from the feature points. They then perform eigendecomposition on matrix A to obtain its eigenvalues and eigenvectors and to get a new modal matrix which has the descendant-sorted eigenvectors as its columns for each data set. The rows of the matrix are then considered as the projections of the feature points into the eigenspace. When the data sets are of different size, only the first M leading eigenvectors from each data sets are used where M is the size of the smaller data set. To make the algorithm more robust, Shapiro and Brady also suggest to use the eigenvalues to scale their corresponding eigenvectors and put more emphasis on the more significant eigenvectors. This acts in a more similar way as the kernel PCA. Comparing with the kernel PCA approach described above, one can see that in this way, Shapiro and Brady's method can be regarded as a special case of the kernel PCA approach, which assumes the data in the mapped space has a mean zero and uses the Gaussian as the kernel function. Figure 3 shows the performance of Shapiro and Brady's method and the kernel PCA with a Gaussian kernel.

4 Experimental Results

Experiments are designed to compare the matching performance of the aforementioned spectral point matching algorithms. In addition to the algorithms described above, the multidimensional scaling is also included in this section. MDS is also a method commonly used for data dimension reduction which is

based on eigenvalues and eigenvectors of a dissimilarity matrix [4]. It attempts to preserve the pairwise relationship between data points while mapping the data into a low dimensional space. The experiments here of matching using MDS is performed using the classical MDS in which the Euclidean distance is taken as the dissimilarity measure.

Focuses of the experiments are on the performance of the algorithms when the data are under transformations and contains uncertainties such as outliers and random position jitter. For this purpose, the data in the experiments are designed as in the following subsection.

4.1 The Data

The experiments are taken on both synthesized data and real data sets and have the following designs:

1. Synthetic data: Assume two dimensional affine transformation. Given $X = \{\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}\}$, a synthetic dataset $Y = (X + 5 \times \mathbf{1}\mathbf{1}^T) \times 0.6 \times R$ is generated for testing the algorithms, where $\mathbf{1} = (1, \ldots, 1)^T$, $R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$ is the rotation matrix and $\theta = \frac{10}{180} \times \pi$.
2. Real data: Here we use the hands sequence shown in Figure (1) and the CMU house sequence ([3]) shown in Figure (2).
3. Noisy data: A Gaussian noise is added to the data set to test the robustness of the algorithm. First A 2-D Gaussian random matrix $D \sim N(\mu, \Sigma)$ is generated, and then the data are added to the matrix of the second feature point set X_2 using the equation $X_2 = X_2 + D$
4. Data sets with different size: To simulate structural errors we delete l consecutive points, where $l = 1, \ldots, 5$, from the second data set (the test data). Also feature point sets in the CMU house sequence have different sizes. For frame 01, 02, 03, 04, and 10 displayed in Figure (2), the sizes of each data-set are 30, 32, 32, 30, 30, respectively.

4.2 The Results

To compare the performances of the kernel approaches when deformations are present, experiments are performed on synthetically generated data where a 2D translation, rotation and isoscaling are added. The effect of missing points and random point position jitter in terms of the 2-D Gaussian random matrices with different covariance matrices as described above are also tested. The results are shown in Figure (3). The experimental results of random point jitter are averages of 100 runs for each covariance matrix. In the experiments of different data sizes, at the beginning, both data sets have 30 points. The results of missing points are averaged over all 30 runs. The testing method of missing data are as described above.

The results of the algorithms on real data sets are displayed in Figure (3), and Table (1). The experiments on missing points are performed in the same

Table 1. Matching results (*Total error numbers*)

	Hand data				CMU House			
Frames	08 - 25	09 - 11	09 - 25	11 - 25	01 - 02	01 - 03	01 - 04	01 - 10
KPCA,Gaussian	30	7	22	21	17	21	4	17
KPCA,Polynomial	5	7	6	12	11	11	3	16
MDS	35	5	26	27	17	21	25	29
Shapiro&Brady	30	7	22	21	26	28	3	25

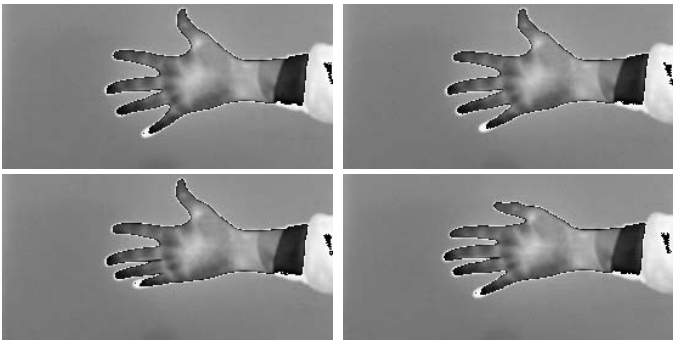

Fig. 1. The hand image data (*From left to right, up to down: frame 08, 09, 11, 25*)

way as for the synthetic data. The results are got from all 44 runs (since in this part, each data set has 44 points at the beginning).

In all the experiments using Shapiro and Brady's method, the eigenvalues are used to enhance their corresponding eigenvectors in order to improve the matching results.

From these experiments, we can see that the kernel PCA approach with a polynomial kernel gives the best results. Experiments on the CMU house data (table 1) also show that the polynomial kernel outperforms all the other algorithms. In all the experiments, the performance of kernel PCA with a Gaussian kernel and the Shapiro and Brady's method ([9]) similar due to their close relationship.

5 Discussion

In this paper we have explored the use of kernel PCA with a polynomial kernel function for finding correspondences between two feature point sets. A relationship with Shapiro and Brady's correspondence method [9] is also discussed. The experimental results reveal that the method offers performance advantages over a number of alternative methods. Besides of the plotted results, the polynomial kernel also shows a more stable performance in experiments of different data size. Even in worst cases it can still maintain a tolerable error rate. The performance of our algorithm is also comparable to the approaches in [2, 3, 6]. The relative

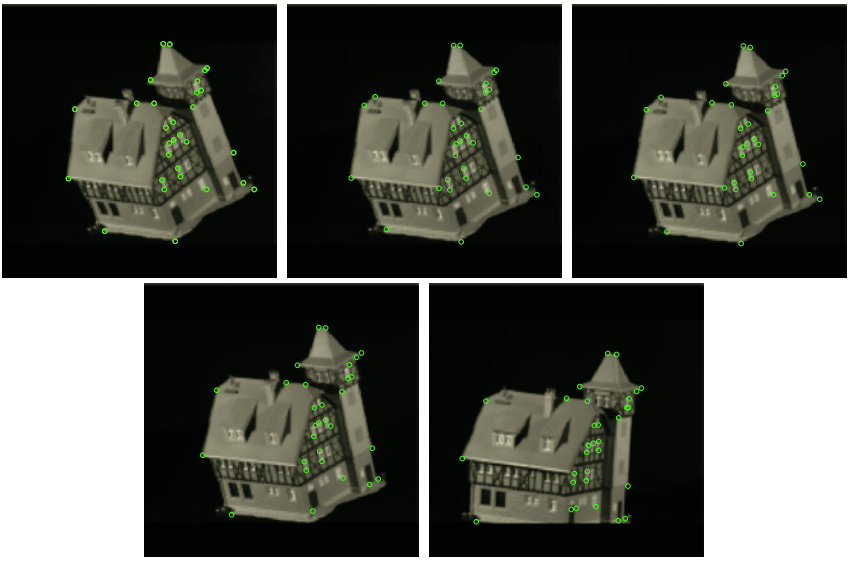

Fig. 2. The CMU house data (*From left to right, up to down: frame 01,02,03,04,10*)

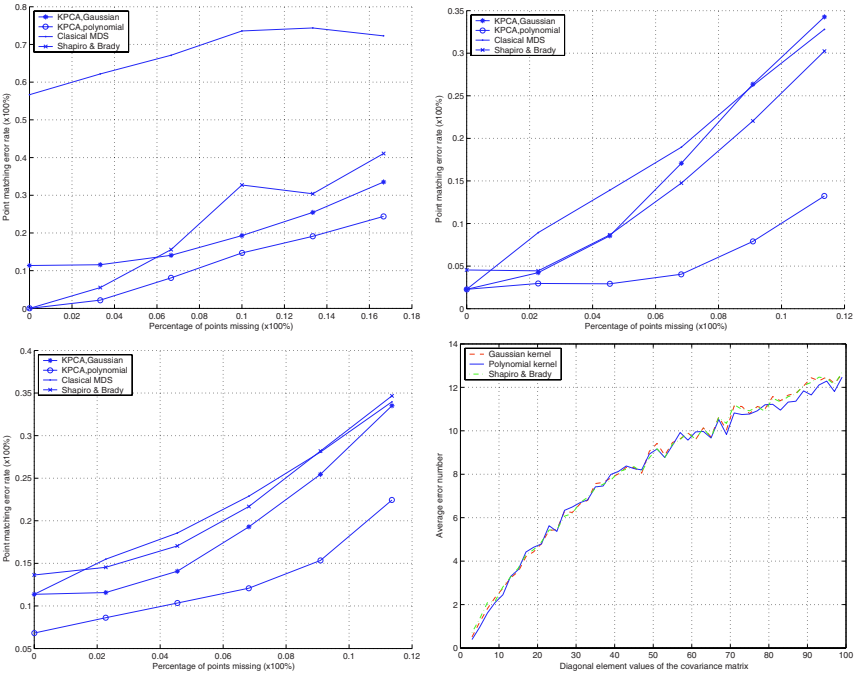

Fig. 3. Matching results (*From left to right, up to down: synthetic data; hand 08 and 09; hand 08 and 11; effects of Gaussian random position jitter*)

weakness of the polynomial kernel is that it requires slightly more computing time that other methods and this might be a problem when very big data sets are being matched and the time requirements is strictly restricted. However this is compensated by its non-iterative property. Comparing with previous iterative-based methods, its computing is more efficient.

In Gaussian kernels, the choose of the parameter σ is not an easy task. In [9], the value is chosen manually. In this paper, we use a heuristic formula based on each data-set's pairwise Euclidean distance matrix to compute the σ value automatically. In our experiments, this formula always chooses an appropriate σ value for different data sets.

Acknowledgements

This work was supported by the UK MOD Corporate Research Programme. The authors thank Dr. Andrew R. Webb for his encouragement and support.

References

1. Caelli,T. and Kosinov, S. "An eigenspace projection clustering method for inexact graph matching". *IEEE Tran. PAMI Vol.26, No.4, 2004*
2. M. Carcassoni and E. R. Hancock. "Spectral correspondence for point pattern matching". *Pattern Recognition. 36(2003) pp.193-204*
3. M. Carcassoni and E. R. Hancock. "Correspondence matching with modal clusters". *IEEE Tran. PAMI Vol.25 No.12, 2003*
4. T. F. Cox and M. A. A. Cox. "Multidimensional Scaling". *Chapman and Hall, London, 1994*
5. J. Ham, D. D. Lee, S. Mika, B. Schölkopf. "A kernel view of the dimensionality reduction of manifolds". *Max Planck Institute for Biological Cybernetics, Technical report TR-110, 2003*
6. B. Luo and E. R. Hancock. "Matching Point-sets using Procrustes alignment and the EM algorithm". *BMVC 1999*
7. B. Schölkopf, A. J. Smola, K. R. Müller. "Nonlinear component analysis as a kernel eigenvvalue problem." *Neural Computation*, 10:1299-1319, 1998.
8. G. L. Scott and H. C. Longuet-Higgins. An Algorithm for Associating the Features of Two Images. *Proc. Royal Soc. London Series B-Biological, vol.244, 1991.*
9. L. S. Shapiro and J. M. Brady. Feature-Based Correspondence - An Eigenvector Approach. *Image and Vision Computing, vol.10, pp.283-288, 1992*
10. V. N. Vapnik. "The nature of statistical learning theory". *Springer Verlag, New York, 1995*