Point Pattern Matching Via Spectral Geometry[*]

Antonio Robles-Kelly[1] and Edwin R. Hancock[2]

[1] National ICT Australia, RSISE Bldg 115, ANU, ACT 0200, Australia
`Antonio.Robles-Kelly@anu.edu.au`
[2] Dept. of Comp. Science, The University of York, York YO10 5DD, UK
`erh@cs.york.ac.uk`

Abstract. In this paper, we describe the use of Riemannian geometry, and in particular the relationship between the Laplace-Beltrami operator and the graph Laplacian, for the purposes of embedding a graph onto a Riemannian manifold. Using the properties of Jacobi fields, we show how to compute an edge-weight matrix in which the elements reflect the sectional curvatures associated with the geodesic paths between nodes on the manifold. We use the resulting edge-weight matrix to embed the nodes of the graph onto a Riemannian manifold of constant sectional curvature. With the set of embedding coordinates at hand, the graph matching problem is cast as that of aligning pairs of manifolds subject to a geometric transformation. We illustrate the utility of the method on image matching using the COIL database.

1 Introduction

The problem of embedding relational structures onto manifolds is an important one in computer science. Furthermore, in the pattern analysis community, there has recently been renewed interest in the use of embedding methods motivated by graph theory. One of the best known of these is ISOMAP [14]. Related algorithms include locally linear embedding which is a variant of PCA that restricts the complexity of the input data using a nearest neighbor graph [11] and the Laplacian eigenmap that constructs an adjacency weight matrix for the data-points and projects the data onto the principal eigenvectors of the associated Laplacian matrix [1]. Lafferty and Lebanon [8] have proposed a number of kernels for statistical learning which are based upon the heat equation on a Riemannian manifold.

Embedding methods can also be used to transform the graph-matching problem into one of point-pattern alignment. The problem is to find matches between pairs of point sets when there is noise, geometric distortion and structural corruption. There is a considerable literature on the problem and many contrasting approaches, including relaxation [4] and optimisation [6], have been attempted. However, the main challenge in graph matching is how to deal with differences in node and edge structure. One of the most elegant recent approaches to the graph matching problem has been to use graph-spectral methods [5], and exploit information conveyed by the eigenvalues and eigenvectors of the adjacency matrix. For instance, Umeyama [16] has developed a method

---

for finding the permutation matrix which best matches pairs of weighted graphs of the same size, by using a singular value decomposition of the adjacency matrices. Scott and Longuet-Higgins [12], on the other hand, align point-sets by performing singular value decomposition on a point association weight matrix. Shapiro and Brady [13] have reported a correspondence method which relies on measuring the similarity of the eigenvectors of a Gaussian point-proximity matrix. Kosinov and Caelli [2] have improved this method by allowing for scaling in the eigenspace.

Our aim in this paper is to seek an embedding of the nodes of a graph which allows matching to be effected using simple point-pattern matching methods. In particular, we aim to draw on the field of mathematics known as spectral geometry, which aims to characterise the properties of operators on Riemannian manifolds using the eigenvalues and eigenvectors of the Laplacian matrix [3]. This approach has a number of advantages. Firstly, our definition of the edge weight is linked directly to the geometry of the underlying manifold. Secondly, the relationship between the Laplace-Beltrami operator and the graph Laplacian provides a clear link between Riemannian geometry and graph-spectral theory [5]. Furthermore, by making use of the Laplace-Beltrami operator to relate the apparatus of graph-spectral theory to Riemannian geometry, the results presented here allow a better understanding of these methods. Finally, the recovery of the embedding coordinates and the geometric transformation via linear algebra yields an analytical solution which is devoid of free parameters.

2   Riemannian Geometry

In this section, we provide the theoretical basis for our graph embedding method. Our aim is to embed the graph nodes as points on a Riemannian manifold. We do this by viewing pairs of adjacent nodes in a graph as points connected by a geodesic on a manifold. In this way, we can make use of Riemannian invariants to recover the embedding of the point pattern on the manifold. With this characterisation at hand, we show how the properties of the Laplace-Beltrami operator can be used to recover a matrix of embedding coordinates. We do this by establishing a link between the Laplace-Beltrami operator and the graph Laplacian. This treatment allows us to relate the graph Laplacian to a Gram matrix of scalar products, whose entries are, in turn, related to the squared distances between pairs of points on the Riemannian manifold.

2.1   Riemannian Manifolds

In this section, we aim to provide a means of characterising the edges of a graph using a geodesic on a Riemannian manifold. The weight of the edge is the cost or energy associated with the geodesic. To commence, let $G = (V, E, W)$ denote a weighted graph with index-set V, edge-set $E = \{(u,v)|(u,v) \in V \times V, u \neq v\}$ and the edge-weight function is $W : E \to [0,1]$. If the nodes in the graph are viewed as points on the manifold, then the weight $W_{u,v}$ associated with the edge connecting the pair of nodes u and v can be computed using the the energy $\mathcal{E}_{p_u,p_v}$ over the geodesic connecting the pair of points p_u and p_v on the manifold. To do this, we employ concepts from differential geometry [3, 10]. In this way, we establish a relationship with the curvature tensor, which, in turn, allows us to characterise the sectional curvature of the manifold. The

reasons for using the curvature tensor are twofold. Firstly, the curvature tensor is natural, i.e. it is invariant under isometries (that is bijective mappings that preserve distance). Secondly, the curvature tensor can be defined intrinsically through coordinate changes. Hence, the curvature tensor is one of the main invariants in Riemannian geometry.

To commence our development, we require some formalism. Let the vector fields Y, X and Z be the extensions over a neighbourhood of the point $p \in M$ of the vectors $\eta, \xi, \zeta \in M_p$. The curvature tensor, which is quadrilinear in nature [3], is denoted by $R(\xi, \eta)$. To obtain a bilinear form, i.e. the sectional curvature, from the curvature tensor we use two linearly independent vectors $\eta, \xi \in M_p$ and write

$$\mathcal{K}(\xi, \eta) = \frac{\langle R(\xi, \eta)\xi, \eta \rangle}{\mid \xi \mid^2 \mid \eta \mid^2 - \langle \xi, \eta \rangle} \tag{1}$$

Further, consider the parametric geodesic curve $\gamma : t \in [\alpha, \beta] \mapsto M$. We define the Jacobi field along γ as the differentiable vector field $Y \in M_p$, orthogonal to γ, satisfying Jacobi's equation $\nabla_t^2 Y + R(\gamma', Y)\gamma' = 0$, where ∇ is said to be a Levi-Civita connection [3].

With these ingredients, we model the edges in the graph as geodesics in a manifold by substituting the shorthands for the derivative of the parametric geodesic curve $\gamma : t \in [\alpha, \beta]$ with respect to the time parameter t, i.e. γ', and the Jacobi field Y into the expression for the sectional curvature introduced in Equation 1. We get

$$\mathcal{K}(\gamma', Y) = \frac{\langle R(\gamma', Y)\gamma', Y \rangle}{\mid \gamma' \mid^2 \mid Y \mid^2 - \langle \gamma', Y \rangle} \tag{2}$$

To simplify the expression for the sectional curvature further, we make use of the fact that, since Y is a Jacobi field, it must satisfy the condition $\nabla_t^2 Y = -R(\gamma', Y)\gamma'$. Hence, we can write

$$\mathcal{K}(\gamma', Y) = \frac{\langle R(\gamma', Y)\gamma', Y \rangle}{\mid \gamma' \mid^2 \mid Y \mid^2} = \frac{\langle -\nabla_t^2 Y, Y \rangle}{\langle Y, Y \rangle} \tag{3}$$

where we have used the fact that Y is orthogonal to γ', substituted $\mid Y \mid^2$ with $\langle Y, Y \rangle$ and set $\mid \gamma' \mid = 1$. As a result, it follows that $\nabla_t^2 Y = -\mathcal{K}(\gamma', Y)Y$. Hence, the Laplacian operator $\nabla_t^2 Y$ is determined by the sectional curvature of the manifold.

This suggests a way of formulating the energy over the geodesic $\gamma \in M$ connecting the pair of points corresponding to the nodes indexed u and v. Consider the geodesic γ subject to the Jacobi field Y. The energy over the geodesic γ can be expressed making use of the equations above as

$$\mathcal{E}_{p_u, p_v} = \int_\gamma \mid \gamma' + \nabla_t^2 Y \mid^2 dt = \int_\gamma \mid \gamma' - \mathcal{K}(\gamma', Y)Y \mid^2 dt \tag{4}$$

We can provide a physical intepretation of the above result. It can be viewed as the energy associated with the geodesic from the point indexed u to the point indexed v, which is the sum of the kinetic energy and the potential energy contributed by the Jacobi field over γ. Hence, the edge-weight is small if a pair of points are close to one another or the curvature along the geodesic between them is small.

In practice, we will confine our attention to the problem of embedding the nodes on a constant sectional curvature surface. For such a surface, the sectional curvature is constant i.e. $\mathcal{K}(\gamma', Y) \equiv \kappa$. Under this restriction the Jacobi field equation becomes $\nabla_t^2 Y = -\kappa Y$. With the boundary conditions $Y(0) = 0$ and $\mid \nabla_t Y(0) \mid = 1$, the solution is

$$Y(t) = \begin{cases} \frac{\sin(\sqrt{\kappa}t)}{\sqrt{\kappa}}\eta & \text{if } \kappa > 0 \\ t\eta & \text{if } \kappa = 0 \\ -\frac{\sinh(\sqrt{-\kappa}t)}{\sqrt{-\kappa}}\eta & \text{if } \kappa < 0 \end{cases} \tag{5}$$

where the vector η is in the tangent space of M at p_u and is orthogonal to γ' at the point indexed u, i.e. $\eta \in M_{p_u}$ and $\langle \eta, \gamma' \mid_{p_u} \rangle = 0$.

With these ingredients, and by rescaling the parameter t so that $\mid \gamma' \mid = a(u,v)$, we can express the weight of the edge connecting the nodes indexed u and v as follows

$$W(u,v) = \begin{cases} \int_0^1 \left(a(u,v)^2 + \kappa \left(\sin(\sqrt{\kappa}a(u,v)t) \right)^2 \right) dt & \text{if } \kappa > 0 \\ \int_0^1 a(u,v)^2 dt & \text{if } \kappa = 0 \\ \int_0^1 \left(a(u,v)^2 - \kappa \left(\sinh(\sqrt{-\kappa}a(u,v)t) \right)^2 \right) dt & \text{if } \kappa < 0 \end{cases} \tag{6}$$

where $a(u,v)$ is the Euclidean distance between each pair of points in the manifold, i.e. $a(u,v) = \parallel p_u - p_v \parallel$.

2.2   Recovery of the Embedding Coordinates

To construct a set of embedding coordinates for the nodes of the graph, we use multidimensional scaling with double centering [15]. We depart from a matrix of embedding coordinates $\mathbf{J}$ obtained from the centred Laplacian using the factorisation $\mathbf{H} = \mathbf{JJ}^T$. The double centering procedure introduces a linear dependency over the columns of the matrix. The double-centered graph Laplacian $\mathbf{H}$ is, in fact, a Gram matrix and, thus, we can recover the node-coordinates making use of a matrix decomposition approach. We construct the centering matrix as follows

$$\mathbf{H} = -\frac{1}{2}\mathbf{B}\mathcal{L}\mathbf{B}^T \tag{7}$$

where $\mathbf{B} = \mathbf{I} - \frac{1}{|V|}\mathbf{ee}^T$ is the centering matrix, $\mathbf{I}$ is the identity matrix, $\mathbf{e}$ is the all-ones vector and $\mathcal{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - W)\mathbf{D}^{-\frac{1}{2}}$ is the normalised graph Laplacian. In the expression above, $\mathbf{D}$ is a diagonal matrix such that $\mathbf{D} = diag(deg(1), deg(2), \ldots, deg(|V|))$ and $deg(i)$ is the degree of the node indexed i.

It is also worth noting that the double centering operation on the graph Laplacian also has the effect of translating the coordinate system for the embedding to the origin. This allows us to pose the problem of matching as an alignment one that involves only rotation.

To perform this factorisation of the matrix $\mathbf{H}$, we make use of Young-Householder theorem [17]. Let $\Lambda = diag(\lambda_1, \lambda_2, .., \lambda_{|V|})$ be the diagonal matrix with the ordered eigenvalues of $\mathbf{H}$ as elements and $\Phi = (\phi_1|, |\phi_2|, \ldots, |\phi_{|V|})$ be the matrix with the

corresponding ordered eigenvectors as columns. Here the ordered eigenvalues and corresponding eigenvectors of the matrix $\mathbf{H}$ satisfy the condition $\mid \lambda_1 \mid \geq \mid \lambda_2 \mid \geq \cdots \geq \mid \lambda_{|V|} \mid > 0$. As a result, we can write $\mathbf{H} = \Phi \Lambda \Phi^T = \mathbf{J}\mathbf{J}^T$, where $\mathbf{J} = \sqrt{\Lambda}\Phi$. The matrix which has the embedding coordinates of the nodes as columns is $\mathbf{D} = \mathbf{J}^T$. Hence, $\mathbf{H} = \mathbf{J}\mathbf{J}^T = \mathbf{D}^T\mathbf{D}$ is a Gram matrix, i.e. its elements are scalar products of the embedding coordinates. Consequently, the embedding of the points is an isometry.

3   Graph Matching by Point Set Alignment

In this section, we show how the graph matching process can be posed as one of manifold alignment. This can be effected by finding the geometric transformation which minimises a quadratic error measure, i.e. least squares distance, between pairs of embedded points. To commence, we require some formalism. Suppose that $\mathbf{H}_D = \Phi_D \Lambda_D \Phi_D^T$ is the centred Laplacian matrix for the set of $\mid V^{\mathcal{D}} \mid$ "data" points whose embedded co-ordinates are given by the matrix $\mathbf{D} = \sqrt{\Lambda_D}\Phi_D^T$. Similarly, $\mathbf{H}_M = \Phi_M \Lambda_M \Phi_M^T$ is the centred Laplacian matrix for the set $\mid V^{\mathcal{M}} \mid$ of "model" points whose embedded co-ordinates are given by the matrix $\mathbf{M} = \sqrt{\Lambda_M}\Phi_M^T$. In practice, the sets of points to be matched may not be of the same size. To accommodate this feature of the data, we assume that the model point set is the larger of the two, i.e. $|V^{\mathcal{M}}| \geq |V^{\mathcal{D}}|$. As a result of the Young-Householder factorisation theorem used in the previous section, the embeddings of the data and model point patterns onto the manifolds $M^{\mathcal{D}} \in \Re^{|V^{\mathcal{D}}|}$ and $M^{\mathcal{M}} \in \Re^{|V^{\mathcal{D}}|}$, respectively, will be assumed to have a dimensionality which is equal to the number of points in the corresponding point-set. Hence, in order to be consistent with our geometric characterisation of the point pattern matching problem, we consider the manifold $M^{\mathcal{D}} \in \Re^{|V^{\mathcal{D}}|}$ to be a covering map, or projection, of the manifold $M^{\mathcal{M}} \in \Re^{|V^{\mathcal{M}}|}$. Here, in order to avoid ambiguities, we are interested in coverings of multiplicity one and, therefore, as an alternative to the matrix $\mathbf{D}$, we work with the matrix of coordinates $\tilde{\mathbf{D}} = [\mathcal{D} \mid \mathbf{n}_{|V^{\mathcal{D}}|+1} \mid \mathbf{n}_{|V^{\mathcal{D}}|+2} \mid \ldots \mid \mathbf{n}_{|V^{\mathcal{M}}|}]$, where $\mathbf{n}_i$ is a vector of length $\mid V^{\mathcal{D}} \mid$ whose entries are null.

With these ingredients, the problem of finding a transformation which can be used to map the data points onto their counterparts in the model point-set can be viewed as that of finding the rotation matrix $\mathbf{R}$ and the point-correspondence matrix $\tilde{\mathbf{P}} = [\mathbf{P} \mid \mathbf{O}]$, where $\mathbf{P}$ is a permutation matrix of order $\mid V^{\mathcal{D}} \mid$ and $\mathbf{O}$ is a null matrix of size $\mid V^{\mathcal{D}} \mid \times \mid V^{\mathcal{M}} - V^{\mathcal{D}} \mid$, which minimise the quadratic error function

$$\epsilon = \parallel \mathbf{M} - \tilde{\mathbf{P}}\mathbf{R}\tilde{\mathbf{D}} \parallel^2 \tag{8}$$

To solve the problem, we divide it in to two parts. First, we find the rotation matrix $\mathbf{R}$ by assuming the point-correspondence matrix $\tilde{\mathbf{P}}$ is known. Second, with the optimum rotation matrix at hand, we recover the point-correspondence matrix $\tilde{\mathbf{P}}$.

To recover the rotation matrix $\mathbf{R}$, we make use of the fact that both matrices, $\mathbf{R}$ and $\tilde{\mathbf{P}}$, are orthogonal and write

$$\epsilon = \text{Tr}[\mathbf{M}\mathbf{M}^T] + \text{Tr}[(\tilde{\mathbf{P}}\mathbf{R}\tilde{\mathbf{D}})(\tilde{\mathbf{P}}\mathbf{R}\tilde{\mathbf{D}})^T] - 2\text{Tr}[\mathbf{M}(\mathbf{R}\tilde{\mathbf{D}})^T\tilde{\mathbf{P}}] \tag{9}$$

From the equation above, it is clear that maximising $\text{Tr}[\mathbf{M}(\mathbf{R}\tilde{\mathbf{D}})^T\tilde{\mathbf{P}}]$ is equivalent to minimising ϵ. Further, assuming that the optimum correspondence matrix $\tilde{\mathbf{P}}$ is known,

we can view the matrix $\tilde{\mathbf{P}}$ as an augmented permutation matrix, and, hence maximising $\mathrm{Tr}[\mathbf{M}\tilde{\mathbf{D}}^T\mathbf{R}]$ is the same as maximising $\mathrm{Tr}[\mathbf{M}(\mathbf{R}\tilde{\mathbf{D}})^T\tilde{\mathbf{P}}]$. This observation is important, because it implies that the rotation matrix $\mathbf{R}$ is the solution to a Procrustean transformation over the embedding coordinates for the set of data-points. Recall that a Procrustes transformation is of the form $\mathbf{Q} = \mathbf{R}\tilde{\mathbf{D}}$ which minimises $\| \mathbf{M} - \mathbf{Q} \|^2$. It is known that minimising $\| \mathbf{M} - \mathbf{Q} \|^2$ is equivalent to maximising $\mathrm{Tr}[\tilde{\mathbf{D}}\mathbf{M}^T\mathbf{R}]$. This is effected by using Kristof's inequality, which states that, if $\mathbf{S}$ is a diagonal matrix with non-negative entries and $\mathbf{T}$ is orthogonal, we have $\mathrm{Tr}[\mathbf{TS}] \geq \mathrm{Tr}[\mathbf{S}]$.

Let the singular value decomposition (SVD) of $\tilde{\mathbf{D}}\mathbf{M}^T$ be $\mathbf{USV}^T$. Using the invariance of the trace function over cyclic permutation, and drawing on Kristof's inequality, we can write $\mathrm{Tr}[\tilde{\mathbf{D}}\mathbf{M}^T\mathbf{R}] = \mathrm{Tr}[\mathbf{USV}^T\mathbf{R}] = \mathrm{Tr}[\mathbf{V}^T\mathbf{RUS}] \geq \mathrm{Tr}[\mathbf{S}]$. It can be shown that $\mathbf{V}^T\mathbf{RU}$ is orthogonal since $\mathbf{R}$ is orthogonal. Furthermore, the maximum of $\mathrm{Tr}[\mathbf{M}(\mathbf{R}\tilde{\mathbf{D}})^T\tilde{\mathbf{P}}]$ is achieved when $\mathbf{V}^T\mathbf{RU} = \mathbf{I}$. As a result, the optimal rotation matrix $\mathbf{R}$ is given by $\mathbf{R} = \mathbf{VU}^T$.

With the rotation matrix at hand, the correspondence matrix $\tilde{\mathbf{P}}$ can be recovered by noting that the product $\mathbf{M}(\mathbf{R}\tilde{\mathbf{D}})^T = \mathbf{MQ}^T$ is the matrix of pairwise inner products between the embedding coordinates for the data and model point-sets. Since the rotation of $\tilde{\mathbf{D}}$ over $\mathbf{R}$ is optimum, the normalised inner product between pairs of matching points is, in the ideal case, equal to unity, i.e. the angle between normalised coordinate vectors is zero. To take advantage of this, we construct the matrix of normalised pairwise inner products and then use it to recover $\tilde{\mathbf{P}}$. Hence, consider the matrix $\mathbf{Z}$ of order $\mid V^{\mathcal{D}} \mid \times \mid V^{\mathcal{M}} \mid$ whose element indexed i, j is given by the normalised inner product of the respective embedding coordinate vectors, after being aligned by rotation, for the data-point indexed i and j^{th} model-point. The elements of the matrix $\mathbf{Z}$ are hence given by

$$\mathbf{Z}(i,j) = \frac{\sum_{k=1}^{|V^{\mathcal{M}}|} \mathbf{Q}(i,k)\mathbf{M}(j,k)}{\sqrt{\sum_{k=1}^{|V^{\mathcal{M}}|} \mathbf{Q}(i,k)^2}\sqrt{\sum_{k=1}^{|V^{\mathcal{M}}|} \mathbf{M}(j,k)^2}} \tag{10}$$

Since the correspondence matrix $\tilde{\mathbf{P}}$ can be viewed as a matrix which slots over the matrix $\mathbf{Z}$ of normalised pairwise inner products and selects its largest values, we can recover $\tilde{\mathbf{P}}$ from $\mathbf{Z}$ in the following way. We commence by clearing $\tilde{\mathbf{P}}$ and, recursively, do

1.- $\tilde{\mathbf{P}}(i,j) = 1$, where $\{i,j \mid \mathbf{Z}(i,j) = \max_{\mathbf{Z}(i,j)\neq 0}(\mathbf{Z})\}$.
2.- $\mathbf{Z}(i,k) = 0 \ \forall \ k \in \mid V^{\mathcal{M}} \mid$ and $\mathbf{Z}(l,j) = 0 \ \forall \ l \in \mid V^{\mathcal{D}} \mid$.

until $\mathbf{Z} \equiv 0$. The data-point indexed i is then a match to the jth model-point *if and only if* $\tilde{\mathbf{P}}(i,j) = 1$. It is important to note that $\mathbf{Z}$ is the equivalent to the correlation, in a scalar-product sense, between the rows of $\mathbf{M}$ and the columns of $\mathbf{Q}^T$. It can be shown that the matrix $\tilde{\mathbf{P}}$ maximises the trace of $\tilde{\mathbf{P}}^T\mathbf{MQ}^T$ and, hence, minimises the quadratic error function ϵ.

This geometric treatment of the node-correspondence problem and its relationship to the correlation, as captured by the entries of $\mathbf{Z}$, between the rows of $\mathbf{M}$ and the columns of $\mathbf{Q}^T$ lends itself naturally to further refinement via statistical approaches such as EM algorithm [9] or relaxation labelling [4].

4   Experiments

The experimental evaluation of our method is divided into two parts. In Section 4.1, we illustrate the effect of the embedding on a sample point-set. In Section 4.2, we experiment with real world data provided by the COIL data-base.

4.1   Point-Set Deformation

In this section, we illustrate the utility of our method for the purposes of embedding a set of data points in a Riemannian manifold of constant sectional curvature. For this purpose, we have used a set of 25 points sampled regularly from a two-dimensional lattice.
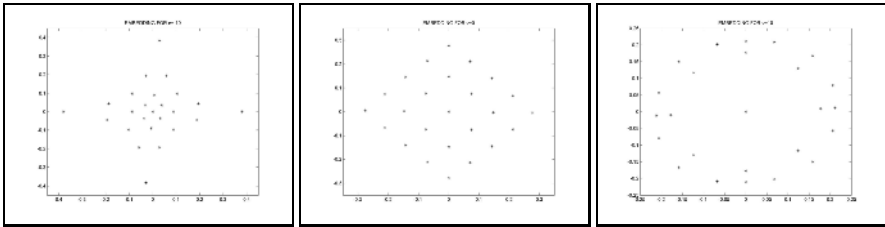

Fig. 1. From left-to-right: embedding results with $\kappa = -10$, 0 and 10 for a point-lattice

In Figure 1 we show, the results obtained by our algorithm for increasing values of κ. From the Figure 1, it is clear that the sectional curvature κ has an important effect in the recovery of the embedding coordinates. For $\kappa = 0$, the embedding is just a rotated version of the original distribution of the points in the plane. When κ is non-zero, then different patterns of behaviour emerge. In the case of negative sectional curvature (i.e. hyperbolic geometry), the embedding "collapses" the distribution of points towards the origin. For positive sectional curvature (i.e. elliptic geometry) the effect is to push the points away from the origin, and the point distribution forms an annulus. This behaviour is consistent with the fact that, for hyperbolic surfaces ($\kappa < 0$) parallel lines diverge. For spherical manifolds ($\kappa > 0$), parallel lines intersect.

4.2   Feature Point Matching

In this section, we aim at assessing the quality of the matching results delivered by our algorithm. As an experimental vehicle, we use the Columbia University COIL-20
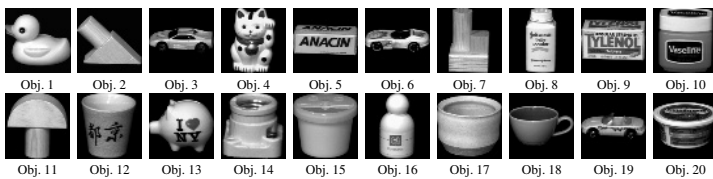

Fig. 2. Sample views for the objects in the Columbia University COIL database

Table 1. Normalised average ratio ε as a function of the sectional curvature κ

Object Index	Normalised average ratio ε of incorrect to correct correspondences			Object Index	Normalised average ratio ε of incorrect to correct correspondences		
	$\kappa = -15$	$\kappa = 0$	$\kappa = 15$		$\kappa = -15$	$\kappa = 0$	$\kappa = 15$
1	0.063	0.068	0.071	11	0.064	0.065	0.068
2	0.066	0.071	0.078	12	0.061	0.066	0.069
3	0.064	0.068	0.075	13	0.059	0.067	0.073
4	0.063	0.067	0.071	14	0.062	0.066	0.071
5	0.062	0.066	0.068	15	0.063	0.068	0.072
6	0.064	0.068	0.069	16	0.061	0.067	0.073
7	0.062	0.067	0.071	17	0.062	0.067	0.074
8	0.063	0.068	0.072	18	0.061	0.064	0.072
9	0.065	0.067	0.069	19	0.063	0.069	0.071
10	0.061	0.068	0.071	20	0.063	0.066	0.069

database. The COIL-20 database contains 72 views for 20 objects acquired by rotating the object under study about a vertical axis. In Figure 2, we show sample views for each of the objects in the database. For each of the views, our point patterns are comprised of feature points detected using the Harris corner detector [7].

   To evaluate the results of matching pairs of views in the database, we have adopted the following procedure. For each object, we have used the first 15 views, 4 of these are "model" views and the remaining 12 are "data" views. We have then matched, by setting κ to -15, 0 and 15, the feature points for the selected "model" views with those corresponding to the two previous and two subsequent views in the database, i.e. we have matched the feature points for the ith view with those corresponding to the views indexed $i-2$, $i-1$, $i+1$ and $i+2$. To provide more qualitative results, we have ground-truthed the correspondences between the "model" and "data" views and computed the normalised average ratio ε of incorrect to correct correspondences $\mu(k,j)$ between the "model" view indexed k and the corresponding "data" view indexed j. The quantity ε is then given by

$$\varepsilon = \frac{1}{4 \mid \Pi \mid} \sum_{k \in \Pi} \sum_{j=k-2, j \neq k}^{j=k+2} \frac{\mu(k,j)}{\rho(k,j)} \tag{11}$$

where $\Pi = \{3, 6, 9, 12\}$ is the set of indices for the "model" views and $\rho(k,j)$ is the maximum number of correspondences between the "model" and the "data" view. In Table 1, we show the values of ε as a function of object index for the three values of κ used in our experiments. Note that, in the table, we have used the object indexes in Figure 2.

   From the quantitative results shown in Table 1, we conclude that the value of the sectional curvature κ has an important effect in the results delivered by the method. The method performs consistently better for negative values of κ. This is the case in which the alignment, is performed between manifolds that are hyperbolic in nature.

5   Conclusions

In this paper, we have shown how the nodes of a graph can be embedded on a constant sectional curvature manifold. The procedure can be viewed as a transformation to the edge-weights of the graph, which modifies the edge-weights using the sectional curvature. When the sectional curvature is positive, then the effect is to emphasise local or short-distance relationships. When the sectional curvature is negative on the other hand, then the effect is to emphasise long-distance relationships. Using the embedded coordinates corresponding to the nodes of the graph, we show how the problem of graph-matching can be transformed into one of Procrustean point-set alignment.

References

[1] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Neural Information Processing Systems*, number 14, pages 634–640, 2002.

[2] T.M. Caelli and S. Kosinov. An eigenspace projection clustering method for inexact graph matching. *PAMI*, 26(4):515–519, April 2004.

[3] I. Chavel. *Riemannian Geometry: A Modern Introduction*. Cambridge University Press, 1995.

[4] W. J. Christmas, J. Kittler, and M. Petrou. Structural matching in computer vision using probabilistic relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):749–764, 1995.

[5] Fan R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[6] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *PAMI*, 18(4):377–388, April 1996.

[7] C. J. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conference*, pages 147–151, 1988.

[8] J. Lafferty and G. Lebanon. Diffusion kernels on statistical manifolds. *Journal of Machine Learning Research*, 6:129–163, 2005.

[9] Bin Luo and E. Hancock. Iterative procrustes alignment with the em algorithm. *Image and Vision Computing*, 20:377–396, 2002.

[10] B. O'Neill. *Elementary Differential Geometry*. Academic Press, 1997.

[11] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[12] G. Scott and H. Longuet-Higgins. An algorithm for associating the features of two images. In *Proceedings of the Royal Society of London*, number 244 in B, pages 21–26, 1991.

[13] L. Shapiro and J. M. Brady. Feature-based correspondance - an eigenvector approach. *Image and Vision Computing*, 10:283–288, 1992.

[14] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[15] W. S. Torgerson. Multidimensional scaling I: Theory and method. *Psychometrika*, 17:401–419, 1952.

[16] S. Umeyama. An eigen decomposition approach to weighted graph matching problems. *PAMI*, 10(5):695–703, September 1988.

[17] G. Young and A. S. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22, 1938.