

LAPLACIAN MATRIX IN ALGEBRAIC GRAPH THEORY

A.Rameshkumar^{*}, R.Palanikumar^{} and S.Deepa^{***}**

^{*}Head, Department of Mathematics, Srimad Andavan Arts and Science College, T.V.Kovil, Trichy – 620005

^{**}Asst. Professor, Department of Mathematics, Srimad Andavan Arts and Science College, T.V.Kovil, Trichy – 620005

^{***}Asst.Professor, Sri Saratha Arts and Science College for women, Perambalur.

*Email Id: rameshmaths@gmail.com, palanirkumar@yahoo.co.in, sdeepamathematics@yahoo.com

ABSTRACT:

In this paper we turn to the spectral decomposition of the Laplacian matrix. We show how the elements of the spectral matrix for the Laplacian can be used to construct symmetric polynomials that are permutation invariants. The coefficients of these polynomials can be used as graph features which can be encoded in a vectorial manner. We extend this representation to graphs in which there are unary attributes on the nodes and binary attributes on the edges by using the spectral decomposition of a Hermitian property matrix that can be viewed as a complex analogue of the Laplacian.

Key words: *Laplacian Matrix, Spectral Matrix, Hermitian Property Matrix, Permutation matrix, symmetric polynomials*

MSC 2011: 05CXX

1. INTRODUCTION

Graph structures have proved computationally cumbersome for pattern analysis. The reason for this is that before graphs can be converted to pattern vectors, correspondences must be established between the nodes of structures which are potentially of different size. To embed the graphs in a pattern space, we explore whether the vectors of invariants can be embedded in a low dimensional space using a number of alternative strategies including **principal components analysis** (PCA), **multidimensional scaling** (MDS) and **locality preserving projection** (LPP). Experimentally, we demonstrate the embeddings result in well defined graph clusters.

The analysis of relational patterns, or graphs, has proven to be considerably more elusive than the analysis of vectorial patterns. Relational patterns arise naturally in the representation of data in which there is a structural arrangement, and are encountered in computer vision, genomics and network analysis. One of the challenges that arise in these domains is that of knowledge discovery from large graph datasets. The tools that are required in this endeavour are robust algorithms that can be used to organise, query and navigate large sets of graphs. In particular, the graphs need to be embedded in a pattern space so that similar structures are close together and dissimilar ones are far apart.

Moreover, if the graphs can be embedded on a manifold in a pattern space, then the modes of shape variation can be explored by traversing the manifold in a systematic way. The process of constructing low dimensional spaces or manifolds is a routine procedure with pattern-vectors. A variety of well established techniques such as principal components analysis ^[1], multidimensional scaling ^[7] and independent components analysis, together with more recently developed ones such as locally linear embedding, iso-map and the Laplacian eigenmap ^[2] exist for solving the problem.

In addition to providing generic data analysis tools, these methods have been extensively exploited in computer vision to construct shape-spaces for 2D or 3D objects, and in particular faces, represented using coordinate and intensity data. Collectively these methods are sometimes referred to as manifold learning theory. However, there are few analogous methods which can be used to construct low dimensional pattern spaces or manifolds for sets of graphs.

2. NOTATIONS

PCA – Principal Component Analysis

MDS – Multi Dimensional Scaling

LPP – Locality Preserving Projection

LDA – Linear Discriminant Analysis

P_U – Permutation Matrix

L – Laplacian Matrix

H – Hermitian Matrix

B_K – symmetric Polynomials

U's – Orthogonal Matrices

Δ 's – Diagonal Matrices

\dagger – dagger operator

3. SPECTRAL GRAPH REPRESENTATIONS

Consider the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ with node-set $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$, edgeset

$\mathcal{E} = \{e_1, e_2, \dots, e_m\} \subset \mathcal{V} \times \mathcal{V}$ and weight function $\mathcal{W} : \mathcal{E} \rightarrow (0, 1]$. The adjacency matrix A for the graph \mathcal{G} is the $n \times n$ symmetric matrix with elements

$$A_{ab} = \begin{cases} 1, & \text{if } (v_a, v_b) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases}$$

In other words, the matrix represents the edge structure of the graph. Clearly if the graph is undirected, the matrix A is symmetric. If the graph edges are weighted, the adjacency matrix is defined to be

$$A_{ab} = \begin{cases} w(v_a, v_b), & \text{if } (v_a, v_b) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases}$$

The Laplacian of the graph is given by $L = D - A$, where D is the diagonal node degree matrix whose elements $D_{aa} = \sum_{b=1}^n A_{ab}$ are the number of edges which exit the individual nodes. The Laplacian is more suitable for spectral analysis than the adjacency matrix since it is positive semidefinite.

In general the task of comparing two such graphs \mathcal{G}_1 and \mathcal{G}_2 involves finding a correspondence mapping between the nodes of the two graphs, $f : v_1 \cup \phi \leftrightarrow v_2 \cup \phi$. The recovery of the correspondence map f can be posed as that of minimising an error criterion.

The minimum value of the criterion can be taken as a measure of the similarity of the two graphs. The additional node ‘ ϕ ’ represents a null match or dummy node. Extraneous or additional nodes are matched to the dummy node. A number of search and optimisation methods have been developed to solve this problem [5, 9]. We may also consider the correspondence mapping problem as one of finding the permutation of nodes in the second graph which places them in the same order as that of the first graph.

This permutation can be used to map the Laplacian matrix of the second graph onto that of the first. If the graphs are isomorphic then this permutation matrix satisfies the condition $L_1 = PL_2P^T$. When the graphs are not exactly isomorphic, then this condition no longer holds. However, the Frobenius distance $\|L_1 - PL_2P^T\|$ between the matrices can be used to gauge the degree of similarity between the two graphs. Spectral techniques have been used to solve this problem. For instance, working with adjacency matrices the permutation matrix P_U that minimises the Frobenius norm $J(P_U) = \|P_U A_1 P_U^T - A_2\|$. The method performs the singular value decompositions $A_1 = U_1 \Delta_1 U_1^T$ and $A_2 = U_2 \Delta_2 U_2^T$, where the U 's are orthogonal matrices and the Δ 's are diagonal matrices [1]. Once these factorizations have been performed, the required permutation matrix is approximated by $U_2 U_1^T$.

In some applications, especially structural chemistry, eigenvalues have also been used to compare the structural similarity of different graphs. However, although the eigenvalue spectrum is a permutation invariant, it represents only a fraction of the information residing in the eigen system of the adjacency matrix.

Since the matrix L is positive semidefinite, it has eigenvalues which are all either positive or zero. The spectral matrix is found by performing the eigenvector expansion for the Laplacian matrix L , i.e.

$$L = \sum_{i=1}^n \lambda_i e_i e_i^T$$

where λ_i and e_i are the n eigenvectors and eigenvalues of the symmetric matrix L . The spectral matrix then has the scaled eigenvectors as columns and is given by

$$\Phi = \sqrt{\lambda_1} e_1, \sqrt{\lambda_2} e_2 \dots \sqrt{\lambda_n} e_n \dots \dots \dots (1)$$

The matrix Φ is a complete representation of the graph in the sense that we can use it to reconstruct the original Laplacian matrix using the relation $L = \Phi \Phi^T$.

The matrix Φ is a unique representation of L iff all n eigenvalues are distinct or zero. This follows directly from the fact that there are n distinct eigenvectors when the eigenvalues are also all distinct. When an eigenvalue is repeated, then there exists a subspace spanned by the eigenvectors of the degenerate eigenvalues in which all vectors are also eigenvectors of L . In this situation, if the repeated eigenvalue is non-zero there is continuum of spectral matrices representing the same graph. However, this is rare for moderately large graphs. Those graphs for which the eigenvalues are distinct or zero are referred to as simple.

4. NODE PERMUTATIONS AND INVARIANTS

The topology of a graph is invariant under permutations of the node labels. However, the adjacency matrix, and hence the Laplacian matrix, is modified by the node order since the rows and columns are indexed by the node order. If we relabel the nodes, the Laplacian matrix undergoes a permutation of both rows and columns. Let the matrix P be the permutation matrix representing the change in node order. The permuted matrix is $L' = PLP^T$. There

is a family of Laplacian matrices which are can be transformed into one another using a permutation matrix. The spectral matrix is also modified by permutations, but the permutation only reorders the rows of the matrix Φ To show this, let L be the Laplacian matrix of a graph G and let $L' = PLP^T$ be the Laplacian matrix obtained by relabelling the nodes using the permutation P . Further, let e be a normalized eigenvector of L with associated eigenvalue λ , and let $e' = Pe$. With these ingredients, we have that

$$L'e' = PLP^TPe = PLe = \lambda e'$$

Hence, e' is an eigenvector of L' with associated eigenvalue λ . As a result, we can write the spectral matrix Φ' of the permuted Laplacian matrix $L' = \Phi'\Phi'^T$ as $\Phi' = P\Phi$. Direct comparison of the spectral matrices for different graphs is hence not possible because of the unknown permutation.

The eigenvalues of the adjacency matrix have been used as a compact spectral representation for comparing graphs because they are not changed by the application of a permutation matrix. The eigenvalues can be recovered from the spectral matrix using the identity

$$\lambda_j = \sum_{i=1}^n \Phi_{ij}^2$$

The expression $\sum_{i=1}^n \Phi_{ij}^2$ is in fact a symmetric polynomial in the components of eigenvector e_i . A symmetric polynomial is invariant under permutation of the variable indices^[3]. In this case, the polynomial is invariant under permutation of the row index i .

In fact the eigenvalue is just one example of a family of symmetric polynomials which can be defined on the components of the spectral matrix. However, there is a special set of these polynomials, referred to as the elementary symmetric polynomials (S) that form a basis set for symmetric polynomials. In other words, any symmetric polynomial can itself be expressed as a polynomial function of the elementary symmetric polynomials belonging to the set S .

We therefore turn our attention to the set of elementary symmetric polynomials^[4]. For a set of variables $\{v_1, v_2 \dots v_n\}$ they can be defined as

$$\begin{aligned}
 S_1(v_1, v_2, \dots, v_n) &= \sum_{i=1}^n v_i \\
 S_2(v_1, v_2, \dots, v_n) &= \sum_{i=1}^n v_i \sum_{j=i+1}^n v_j \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 S_r(v_1, v_2, \dots, v_n) &= \sum_{i_1 < i_2 < \dots < i_r} v_{i_1} v_{i_2} \dots v_{i_r} \\
 &\vdots \\
 &\vdots \\
 &\vdots
 \end{aligned}$$

$$S_n(v_1, v_2, \dots, v_n) = \prod_{i=1}^n v_i$$

The power symmetric polynomial functions

$$P_1(v_1, v_2, \dots, v_n) = \sum_{i=1}^n v_i$$

$$P_2(v_1, v_2, \dots, v_n) = \sum_{i=1}^n v_i^2$$

.

.

.

$$P_r(v_1, v_2, \dots, v_n) = \sum_{i=1}^n v_i^r$$

.

.

.

$$P_n(v_1, v_2, \dots, v_n) = \sum_{i=1}^n v_i^n$$

also form a basis set over the set of symmetric polynomials. As a consequence, any function which is invariant to permutation of the variable indices and that can be expanded as a Taylor series, can be expressed in terms of one of these sets of polynomials. The two sets of polynomials are related to one another by the Newton-Girard formula:

$$S_r = \frac{(-1)^{r+1}}{r} \sum_{k=1}^r (-1)^{k+1} P_k S_{r-k} \dots\dots\dots(2)$$

where we have used the shorthand S_r for $S_r(v_1, \dots, v_n)$ and P_r for $P_r(v_1, \dots, v_n)$. As a consequence, the elementary symmetric polynomials can be efficiently computed using the power symmetric polynomials.

In this paper, we intend to use the polynomials to construct invariants from the elements of the spectral matrix. The polynomials can provide spectral “features” which are invariant under node permutations of the nodes in a graph and utilise the full spectral matrix. These features are constructed as follows; each column of the spectral matrix Φ forms the input to the set of spectral polynomials. For example the column $\{\Phi_{1,i}, \Phi_{2,i}, \dots, \Phi_{n,i}\}$ produces the polynomials $S_1(\Phi_{1,i}, \Phi_{2,i}, \dots, \Phi_{n,i}), S_2(\Phi_{1,i}, \Phi_{2,i}, \dots, \Phi_{n,i}), \dots, S_n(\Phi_{1,i}, \Phi_{2,i}, \dots, \Phi_{n,i})$. The values of each of these polynomials are invariant to the node order of the Laplacian. We can construct a set of n^2 spectral features using the n columns of the spectral matrix in combination with the n symmetric polynomials.

Each set of n features for each spectral mode contains all the information about that mode up to a permutation of components. This means that it is possible to reconstruct the original components of the mode given the values of the features only. This is achieved using the relationship between the roots of a polynomial in x and the elementary symmetric polynomials. The polynomial

$$\prod_i (x - v_i) = 0 \dots\dots\dots(3)$$

has roots v_1, v_2, \dots, v_n . Multiplying out Equation (3) gives

$$x^n - S_1x^{n-1} + S_2x^{n-2} + \dots + (-1)^n S_n = 0 \tag{4}$$

where we have again used the shorthand S_r for $S_r(v_1, \dots, v_n)$. By substituting the feature values into Equation (4) and finding the roots, we can recover the values of the original components. The root order is undetermined, so as expected the values are recovered up to a permutation.

5. UNARY AND BINARY ATTRIBUTES

Attributed graphs are an important class of representations and need to be accommodated in graph matching. We have suggested the use of an augmented matrix to capture additional measurement information. While it is possible to make use of a such an approach in conjunction with symmetric polynomials, another approach from the spectral theory of Hermitian matrices suggests itself.

A Hermitian matrix \mathbf{H} is a square matrix with complex elements that remains unchanged under the joint operation of transposition and complex conjugation of the elements, i.e. $\mathbf{H}^\dagger = \mathbf{H}$. Hermitian matrices can be viewed as the complex number counterpart of the symmetric matrix for real numbers. Each off-diagonal element is a complex number which has two components, and can therefore represent a 2-component measurement vector on a graph edge. The on-diagonal elements are necessarily real quantities, so the node measurements are restricted to be scalar quantities.

There are some constraints on how the measurements may be represented in order to produce a positive semidefinite Hermitian matrix. Let $\{x_1, x_2, \dots, x_n\}$ be a set of measurements for the node-set \mathcal{V} . Further, let $\{y_{1,2}, y_{1,3}, \dots, y_{n,n}\}$ be the set of measurements associated with the edges of the graph, in addition to the graph weights^[6]. Each edge then has a pair of observations $(W_{a,b}, y_{a,b})$ associated with it. There are a number of ways in which the complex number $H_{a,b}$ could represent this information, for example with the real part as W and the imaginary part as y . However, we would like our complex property matrix to reflect the Laplacian. As a result the off-diagonal elements of \mathbf{H} are chosen to be

$$H_{a,b} = -W_{a,b}e^{iy_{ab}} \tag{5}$$

In other words, the connection weights are encoded by the magnitude of the complex number $H_{a,b}$ and the additional measurement by its phase. By using this encoding, the magnitude of the numbers is the same as in the original Laplacian matrix. Clearly this encoding is most suitable when the measurements are angles. If the measurements are easily bounded, they can be mapped onto an angular interval and phase wrapping can be avoided. If the measurements are not easily bounded then this encoding is not suitable.

The measurements must satisfy the conditions $-\pi \leq y_{a,b} < \pi$ and $y_{a,b} = -y_{b,a}$ to produce a Hermitian matrix. To ensure a positive definite matrix, we require $H_{aa} > -\sum_{b \neq a} |H_{ab}|$. This condition is satisfied if

$$H_{aa} = x_a + \sum_{b \neq a} W_{a,b} \text{ and } x_a \geq 0 \tag{6}$$

When defined in this way the property matrix is a complex analogue of the weighted Laplacian for the graph. For a Hermitian matrix there is an orthogonal complete basis set of eigenvectors and eigenvalues obeying the eigenvalue equation $\mathbf{H}\mathbf{e} = \lambda\mathbf{e}$. In the Hermitian case, the eigenvalues λ are real and the components of the eigenvectors \mathbf{e} are complex. There is a potential ambiguity in the eigenvectors, in that any multiple of an

eigenvector is also a solution, i.e. $\mathbf{H}\alpha\mathbf{e} = \lambda\alpha\mathbf{e}$. In the real case, we choose α such that \mathbf{e} is of unit length. In the complex case, α itself may be complex, and needs to determine by two constraints. We set the vector length to $|\mathbf{e}_i| = 1$ and in addition we impose the condition $\arg \sum_{i=1}^n e_i = 0$, which specifies both real and imaginary parts.

This representation can be extended further by using the four-component complex numbers known as quaternions. As with real and complex numbers, there is an appropriate eigen decomposition which allows the spectral matrix to be found. In this case, a edge weight and three additional binary measurements may be encoded on an edge. It is not possible to encode more than one unary measurement using this approach. However, for the experiments in this paper, we have concentrated on the complex representation. When the eigenvectors are constructed in this way the spectral matrix is found by performing the eigenvector expansion

$$H = \sum_{i=1}^n \lambda_i e_i e_i^\dagger$$

where λ_i and e_i are the n eigenvectors and eigenvalues of the Hermitian matrix H ^[8]. We construct the complex spectral matrix for the graph \mathcal{G} using the eigenvectors as columns, i.e.

$$\Phi = (\sqrt{\lambda_1} e_1, \sqrt{\lambda_2} e_2, \dots, \sqrt{\lambda_n} e_n) \dots\dots\dots(7)$$

We can again reconstruct the original Hermitian property matrix using the relation $\mathbf{H} = \Phi\Phi^\dagger$.

Since the components of the eigenvectors are complex numbers and therefore each symmetric polynomial is complex. Hence, the symmetric polynomials must be evaluated with complex arithmetic and also evaluate to complex numbers. Each S_r therefore has both real and imaginary components. The real and imaginary components of the symmetric polynomials are interleaved and stacked to form a feature-vector \mathbf{B} for the graph. This feature-vector is real.

6. GRAPH EMBEDDING METHODS

We explore three different methods for embedding the graph feature vectors in a pattern space. Two of these are classical methods. Principal components analysis (PCA) finds the projection that accounts for the variance or scatter of the data. Multidimensional scaling (MDS), on the other hand, preserves the relative distances between objects. The remaining method is a newly reported one that offers a compromise between preserving variance and the relational arrangement of the data, and is referred to as locality preserving projection [10].

In this paper we are concerned with the set of graphs $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k, \dots, \mathcal{G}_N$. The k^{th} graph is denoted by $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{E}_k)$ and the associated vector of symmetric polynomials is denoted by \mathbf{B}_k .

6.1 Principal component analysis

We commence by constructing the matrix $X = [\mathbf{B}_1 - \bar{\mathbf{B}} | \mathbf{B}_2 - \bar{\mathbf{B}} | \dots | \mathbf{B}_k - \bar{\mathbf{B}} | \dots | \mathbf{B}_N - \bar{\mathbf{B}}]$, with the graph feature vectors as columns. Here $\bar{\mathbf{B}}$ is the mean feature vector for the dataset. Next, we compute the covariance matrix for the elements of the feature vectors by taking the matrix product $C = XX^T$. We extract the principal components directions by performing the eigendecomposition $C = \sum_{i=1}^N l_i u_i u_i^T$ on the covariance matrix C , where the l_i are the eigenvalues and the u_i are the eigenvectors. We use the first s leading eigenvectors to represent the graphs extracted

from the images. The coordinate system of the eigenspace is spanned by the s orthogonal vectors $U = (u_1, u_2, \dots, u_s)$. The individual graphs represented by the long vectors $B_k, k= 1, 2, \dots, N$ can be projected onto this eigenspace using the formula $y_k = U^T(B_k - \bar{B})$. Hence each graph G_k is represented by an s -component vector y_k in the eigenspace.

Linear discriminant analysis (LDA) is an extension of PCA to the multiclass problem. We commence by constructing separate data matrices X_1, X_2, \dots, X_{N_c} for each of the N_c classes. These may be used to compute the corresponding class covariance matrices $C_i = X_i X_i^T$. The average class covariance matrix $\tilde{C} = \frac{1}{N_c} \sum_{i=1}^{N_c} C_i$ is

found. This matrix is used as a sphering transform. We commence by computing the eigendecomposition

$$\tilde{C} = \sum_{i=1}^N l_i u_i u_i^T = U \Lambda U^T$$

where U is the matrix with the eigenvectors of \tilde{C} as columns and $\Lambda = \text{diag}(l_1, l_2, \dots, l_n)$ is the diagonal eigenvalue matrix. The sphered representation of the data is $X' = \Lambda^{-1/2} U^T X$. Standard PCA is then applied to the resulting data matrix X' . The purpose of this technique is to find a linear projection which describes the class differences rather than the overall variance of the data.

6.2 Multidimensional scaling

Multidimensional scaling (MDS) is a procedure which allows data specified in terms of a matrix of pairwise distances to be embedded in a Euclidean space. Here we intend to use the method to embed the graphs extracted from different viewpoints in a lowdimensional space. To commence we require pairwise distances between graphs. We do this by computing the norms between the spectral pattern vectors for the graphs. For the graphs indexed i_1 and i_2 , the distance is $d_{i_1, i_2} = (B_{i_1} - B_{i_2})^T (B_{i_1} - B_{i_2})$. The pairwise distances d_{i_1, i_2} are used as the elements of an $N \times N$ dissimilarity matrix R , whose elements are defined as follows

$$R_{i_1, i_2} = \begin{cases} d_{i_1, i_2} & \text{if } i_1 \neq i_2 \\ 0 & \text{if } i_1 = i_2 \end{cases} \dots\dots\dots(8)$$

In this paper, we use the classical multidimensional scaling method [7] to embed the graphs in a Euclidean space using the matrix of pairwise dissimilarities R . The first step of MDS is to calculate a matrix T whose element with row r and column c is given by $T_{rc} = -\frac{1}{2} d_{rc}^2 - \hat{d}_r^2 - \hat{d}_c^2 + \hat{d}_{..}^2$ where $\hat{d}_r = \frac{1}{N} \sum_{c=1}^N d_{rc}$ is the average dissimilarity value over the r^{th} row, \hat{d}_c is the dissimilarity average value over the c^{th} column and $\hat{d}_{..} = \frac{1}{N^2} \sum_{r=1}^N \sum_{c=1}^N d_{r,c}$ is the average dissimilarity value over all rows and columns of the dissimilarity matrix R .

6.3 Locality preserving projection

The LPP is a linear dimensionality reduction method which attempts to project high dimensional data to a low dimensional manifold, while preserving the neighbourhood structure of the data set. The method is relatively insensitive to outliers and noise. This is an important feature in our graph clustering task since the outliers are usually introduced by imperfect segmentation processes. The linearity of the method makes it computationally efficient.

The graph feature vectors are used as the columns of a data-matrix $X = (|B_1| |B_2| \dots |B_N|)$.

The relational structure of the data is represented by a proximity weight matrix W with elements $W_{i_1, i_2} = \exp[-k d_{i_1, i_2}]$, where k is a constant. If Q is the diagonal degree matrix with the row weights

$Q_{k,k} = \sum_{j=1}^N W_{k,j}$ as elements, then the relational structure of the data is represented using the Laplacian matrix $J =$

$Q - W$. The idea behind LPP is to analyse the structure of the weighted covariance matrix XWX^T . The optimal projection of the data is found by solving the generalised eigenvector problem

$$XJX^T u = \lambda XQX^T u. \dots\dots\dots(9)$$

We project the data onto the space spanned by the eigenvectors corresponding to the s smallest eigenvalues. Let $U = (u_1, u_2, \dots, u_s)$ be the matrix with the corresponding eigenvectors as columns, then projection of the k th feature vector is $y_k = U^T B_k$.

7. SHOCK GRAPHS

7.1 Tree Attributes

Before processing, the shapes are firstly normalised in area and aligned along the axis of largest moment. After performing these transforms, the distances and angles associated with the different shapes can be directly compared, and can therefore be used as attributes.

To abstract the shape skeleton using a tree, we place a node at each junction point in the skeleton. The edges of the tree represent the existence of a connecting skeletal branch between pairs of junctions. The nodes of the tree are characterised using the radius $r(a)$ of the smallest bitangent circle from the junction to the boundary. Hence, for the node a , $x_a = r(a)$. The edges are characterised by two measurements. For the edge (a, b) the first of these, $y_{a,b}$ is the angle between the nodes a and b , i.e. $y_{a,b} = \theta(a, b)$. Since most skeleton branches are relatively straight, this is an approximation to the angle of the corresponding skeletal branch.

Furthermore, since $-\pi \leq \theta(a, b) < \pi$ and $\theta(a, b) = -\theta(b, a)$, the measurement is already suitable for use in the Hermitian property matrix.

In order to compute edge weights, we note that the importance of a skeletal branch may be determined by the rate of change of boundary length l with skeleton length s , which we denote by dl/ds . This quantity is related to the rate of change of the bitangent circle radius along the skeleton, i.e. dr/ds , by the formula

$$\frac{dl}{ds} = \sqrt{1 - \left(\frac{dr}{ds}\right)^2}$$

The edge weight $W_{a,b}$ is given by the average value of dl/ds along the relevant skeletal branch.

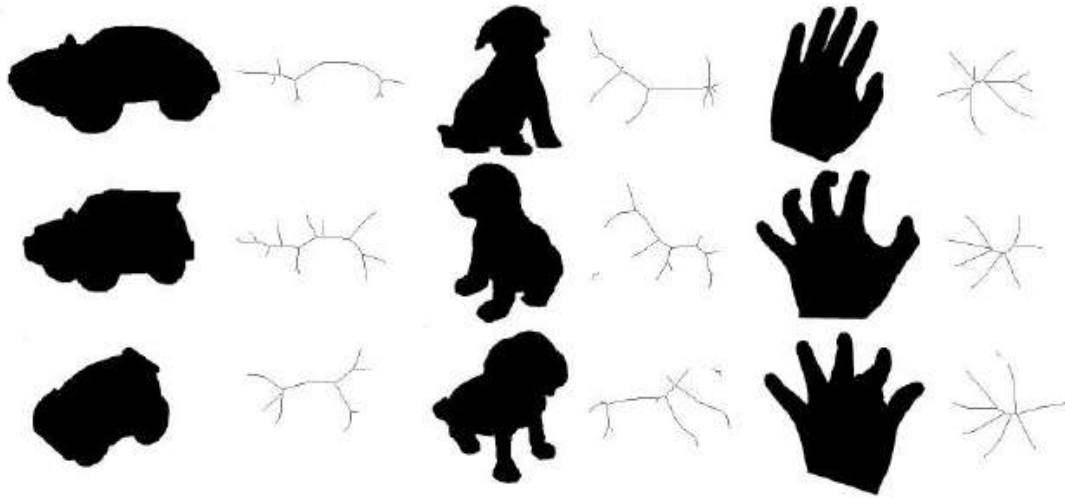


Figure A: Examples from the shape database with their associated skeletons.

We extract the skeleton from each binary shape and attribute the resulting tree in the manner outlined in Figure A shows some examples of the types of shape present in the database along with their skeletons. We commence by showing some results for the three shapes shown in Figure A. The objects studied are a hand, some puppies and some cars. The dog and car shapes consist of a number of different objects and different views of each object. The hand category contains different hand configurations. We apply the three embedding strategies outline to the vectors of permutation invariants extracted from the Hermitian variant of the Laplacian.

7.2 Experiments with shock trees

Our experiments are performed using a database of 42 binary shapes. Each binary shape is extracted from a 2D view of a 3D object. There are 3 classes in the database, and for each object there are a number of views acquired from different viewing directions and a number of different examples of the class.



Figure – B: MDS (left), PCA (middle) and LPP (right) applied to the shock graphs.

We commence in the **left hand panel of Figure – B** by showing the result of applying the MDS procedure to the three shape categories. The hand shapes form a compact cluster in the MDS space. There are other local clusters consisting of three or four members of the remaining two classes. This reflects the fact that while the hand shapes

have very similar shock graphs, the remaining two categories have rather variable shock graphs because of the different objects.

The middle panel of Figure – B shows the result of using PCA. Here the distributions of shapes are much less compact. While a distinct cluster of hand shapes still occurs, they are generally more dispersed over the feature space. There are some distinct clusters of the car shape, but the distributions overlap more in the PCA projection when compared to the MDS space.

The right-hand panel of Figure – B shows the result of the LPP procedure on the dataset. The results are similar to the PCA method. One of the motivations for the work presented here was the potential ambiguities that are encountered when using the spectral features of trees. To demonstrate the effect of using attributed trees rather than simply weighting the edges, we have compared the LDA projections using both types of data.

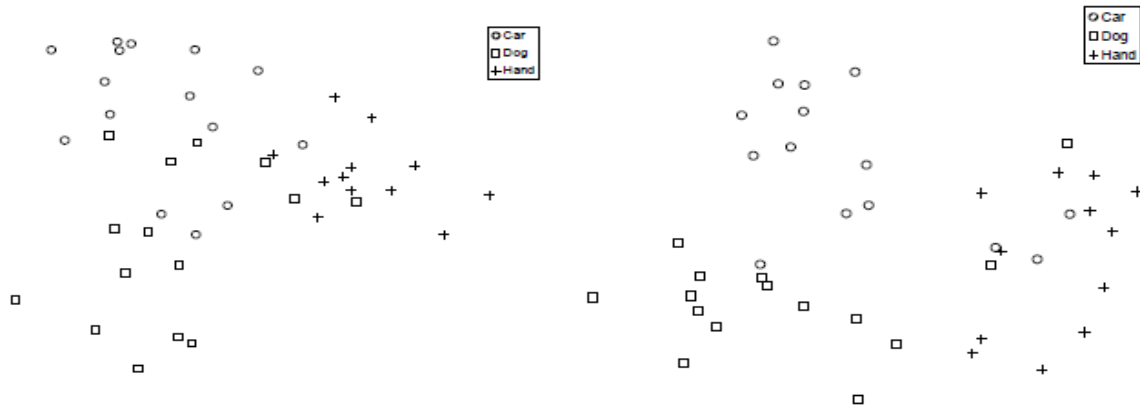


Figure – C : A comparison of attributed trees with weighted trees.

Left : trees with edge weights based on boundary lengths.

Right : Attributed trees with additional edge angle information.

In Figure – C the right-hand plot shows the result obtained using the symmetric polynomials from the eigenvectors of the Laplacian matrix $L=D-W$, associated with the edge weight matrix. The left-hand plot shows the result of using the Hermitian property matrix. The Hermitian property matrix for the attributed trees produces a better class separation than the Laplacian matrix for the weighted trees.

The separation can be measured by the Fisher discriminant between the classes, which is the squared distance between class centres divided by the variance along the line joining the centres.

Table – I

Classes	Separations	
	Hermitian property matrix	Weighted matrix
Car/dog	1.12	0.77
Car/hand	0.97	1.02
Dog/hand	0.92	0.88

8. CONCLUSIONS

We have shown how graphs can be converted into pattern vectors by utilizing the spectral decomposition of the Laplacian matrix and basis sets of symmetric polynomials. These feature vectors are complete, unique and continuous. However, and most important of all, they are permutation invariants. We investigate how to embed the vectors in a pattern space, suitable for clustering the graphs. Here we explore a number of alternatives including PCA, MDS and LPP. In an experimental study we show that the feature vectors derived from the symmetric polynomials of the Laplacian spectral decomposition yield good clusters when MDS or LPP are used. There are clearly a number of ways in which the work presented in this paper can be developed. For instance, since the representation based on the symmetric polynomials is complete, they may form the means by which a generative model of variations in graph structure can be developed. This model could be learned in the space spanned by the permutation invariants, and the mean graph and its modes of variation reconstructed by inverting the system of equations associated with the symmetric polynomials.

REFERENCES

- 1) A.D. Bagdanov and M. Worring, "First Order Gaussian Graphs for Efficient Structure Classification", *Pattern Recognition*, 36, pp. 1311-1324, 2003.
- 2) M.Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation", *Neural Computation*, 15(6), pp. 1373-1396, 2003.
- 3) N. Biggs, "Algebraic Graph Theory", CUP.
- 4) P. Betti and R. Morris, "Almost all trees share a complete set of inmanantal polynomials", *Journal of Graph Theory*, 17, pp. 467-476, 1993.
- 5) W. J. Christmas, J. Kittler and M. Petrou, "Structural Matching in ComputerVision using Probabilistic Relaxation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17, pp. 749-764, 1995.
- 6) F.R.K. Chung, "Spectral Graph Theory", American Mathmatical Society Ed.,CBMS series 92, 1997.
- 7) T. Cox and M. Cox, "Multidimensional Scaling", Chapman-Hall, 1994.
- 8) D. Cvetkovic, P. Rowlinson and S. Simic, "Eigenspaces of graphs", Cambridge University Press, 1997
- 9) S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18, pp. 377-388, 1996.
- 10) X.He and P. Niyogi, "Locality preserving projections", *Advances in Neural InformationProcessing Systems* 16, MIT Press, 2003.
- 11) D. Heckerman, D. Geiger and D.M. Chickering, "Learning Bayesian Networks: The combination of knowledge and statistical data", *Machine Learning*, 20, pp. 197-243, 1995.
- 12) G. Nirmala and M. Vijya "Strong Fuzzy Graphs on Composition, Tensor And Normal Products" *Aryabhata. J. Maths. & Info.* Vol. 4 (2) 2012, pp 343-350.
- 13) K. Thilakam & A. Sumathi "Wiener Index of Chain Graphs" *Aryabhata. J. Maths. & Info.* Vol. 5 (2) 2013, pp 347-352.