

# A Study of Graph Spectra for Comparing Graphs

Ping Zhu and Richard C. Wilson  
Computer Science Department  
University of York, UK

## Abstract

The spectrum of a graph has been widely used in graph theory to characterise the properties of a graph and extract information from its structure. It has been less popular as a representation for pattern matching for two reasons. Firstly, more than one graph may share the same spectrum. It is well known, for example, that very few trees can be uniquely specified by their spectrum. Secondly, the spectrum may change dramatically with a small change structure. In this paper we investigate the extent to which these factors affect graph spectra in practice, and whether they can be mitigated by choosing a particular matrix representation of the graph.

There are a wide variety of graph matrix representations from which the spectrum can be extracted. In this paper we analyse the adjacency matrix, combinatorial Laplacian, normalised Laplacian and unsigned Laplacian. We also study the use of the spectrum derived from the heat kernel matrix and path length distribution matrix. We investigate the cospectrality of these matrices over large graph sets and show that the Euclidean distance between spectra tracks the edit distance over a wide range of edit costs, and we analyse the stability of this relationship. We then use the spectra to match and classify the graphs and demonstrate the effect of the graph matrix formulation on error rates.

## 1 Introduction

The spectrum of a graph has been widely used in graph theory to characterise the properties of a graph and extract information from its structure. They have been much less widely employed as a graph representation for matching and comparison of graphs. There are two main reasons for this, firstly, more than one graph may share the same spectrum. Secondly, the spectrum may change dramatically with a small change structure. While these factors count against the spectrum, they may or may not be a factor in practical graph matching problems.

Graph structures have been used to represent structural and relational arrangements of entities in many vision problems. The key problem in utilising graph representations lies in measuring their structural similarity. Many authors have employed the concept of graph edit distance. In recent work [14, 15, 7], we have shown how spectral features can be found which can characterise a graph and which can be used for graph comparison. This approach is based on spectral graph theory, which is a branch of mathematics that is

concerned with characterising the structural properties of graphs using the eigenvectors of the adjacency matrix or the closely related Laplacian matrix (the degree matrix minus the adjacency matrix) [2]. One of the well known successes of spectral graph theory in computer vision is the use eigenvector methods for grouping via pairwise clustering. Examples include Shi and Malik’s [11] iterative normalised cut method which uses the Fiedler (i.e. second) eigenvector for image segmentation and Sarkar and Boyer’s use of the leading eigenvector of the weighted adjacency matrix [9]. Graph spectral methods have also been used to correspondence analysis. Kosinov and Caelli[5] have used properties of the spectral decomposition to represent graphs and Shokoufandeh et al[12] has used eigenvalues of shock graphs to index shapes. We have previously shown[14, 15] how permutation invariant polynomials can be used to derive features which describe graphs and make full use of the available spectral information.

The spectrum of a graph is generally considered to be too weak to be a useful tool for representing the graph, main due to the result of Schwenk[10] who showed that for trees at least, a sufficiently large tree nearly always has a partner with the same spectrum. Trees therefore cannot be uniquely defined by the spectrum. However, it is not known to what extent this is a problem in practice. Computational simulations by Haemers et al[13] have shown that the fraction of cospectral graphs reaches 21% at 10 vertices (for the adjacency matrix) and is less for 11 vertices, which is the limit of their simulations. While far from conclusive, their results suggest that it may be possible that nearly all graphs do have a unique spectrum.

A number of alternative matrix representations have been proposed in the literature. These include the adjacency matrix, Laplacian and normalised Laplacian. More recently, variations of the heat kernel on the graph have also been used. The spectrum of all of these representations may be used to characterise the graph, and each may reveal different graph properties. Some of these representations may be more stable to perturbations in the graph. In this paper we analyse these matrices and quantify the effect the matrix representation has on the stability and representational power of the eigenvalues of the graph. In section 2, we review the standard graph representations. In section 3, we investigate the cospectrality properties of these matrices. Section 4 describes how we measure the stability and representative power of the eigenvalues. Finally, section 5 details the experiments aimed at measuring the utility of these representations.

## 2 Standard Graph Representations

In this section, we review the properties of some standard graph representations and their relationships with each other. The graphs under consideration here are undirected graphs. Whilst we do not consider weighted graphs here, these ideas are straightforwardly extended to such graphs. We denote a graph by  $G = (V, E)$  where  $V$  is the set of nodes and  $E \subseteq V \times V$  is the set of edges. The degree of a vertex  $u$  is the number of edges leaving the vertex  $u$  and is denoted  $d_u$ .

## 2.1 Adjacency matrix

The most basic matrix representation of a graph is using the adjacency matrix  $A$  for the graph. This matrix is given by

$$A(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Clearly if the graph is undirected, the matrix  $A$  is symmetric. As a consequence, the eigenvalues of  $A$  are real. These eigenvalues may be positive, negative or zero and the sum of the eigenvalues is zero. The eigenvalues may be ordered by their magnitude and collected into a vector which describes the graph spectrum.

## 2.2 Combinatorial Laplacian matrix

In some applications, it is useful to have a positive semidefinite matrix representation of the graph. This may be achieved by using the Laplacian. We first construct the diagonal degree matrix  $D$ , whose diagonal elements are given by the node degrees  $D(u, u) = d_u$ . From the degree matrix and the adjacency matrix we then can construct the standard Laplacian matrix

$$L = D - A \quad (2)$$

i.e. the degree matrix minus the adjacency matrix. The Laplacian has at least one zero eigenvalue, and the number of such eigenvalues is equal to the number of disjoint parts in the graph. The *signless* Laplacian has all entries greater than zero and is defined to be

$$|L| = D + A \quad (3)$$

## 2.3 Normalized Laplacian matrix

The normalized Laplacian matrix is defined to be the matrix

$$\hat{L} = \begin{cases} 1 & \text{if } u = v \\ -\frac{1}{\sqrt{d_u d_v}} & \text{if } u \text{ and } v \text{ are adjacent} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

We can also write it as  $\hat{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$ . As with the Laplacian of the graph, this matrix is positive semidefinite and so has positive or zero eigenvalues. The normalisation factor means that the largest eigenvalue is less than or equal to 2, with equality only when  $G$  is bipartite. Again, the matrix has at least one zero eigenvalue. Hence all the eigenvalues are in the range  $0 \leq \lambda \leq 2$ .

## 2.4 Heat Kernel

The heat kernel is based on the diffusion of heat across the graph. It is a representation which has attracted recent interest in the literature. We are interested in the heat equation associated with the Laplacian, i.e.  $\frac{\partial h_t}{\partial t} = -L h_t$  where  $h_t$  is the heat kernel and  $t$  is time. The solution is found by exponentiating the Laplacian eigenspectrum, i.e.

$h_t = \Phi \exp[-t\Lambda] \Phi^T$ . The heat kernel is a  $|V| \times |V|$  matrix, and for the nodes  $u$  and  $v$  of the graph  $G$  the resulting component is

$$h_t(u, v) = \sum_{i=1}^{|V|} \exp[-\lambda_i t] \phi_i(u) \phi_i(v) \quad (5)$$

When  $t$  tends to zero, then  $h_t \simeq I - Lt$ , i.e. the kernel depends on the local connectivity structure or topology of the graph. If, on the other hand,  $t$  is large, then  $h_t \simeq \exp[-t\lambda_m] \phi_m \phi_m^T$ , where  $\lambda_m$  is the smallest non-zero eigenvalue and  $\phi_m$  is the associated eigenvector, i.e. the Fiedler vector. Hence, the large time behavior is governed by the global structure of the graph. By controlling  $t$ , we can obtain representations of varying degrees of locality.

## 2.5 Path Length Distribution

It is interesting to note that the heat kernel is also related to the path length distribution on the graph. If  $D_k(u, v)$  is the number of paths of length  $k$  between nodes  $u$  and  $v$  then

$$h_t(u, v) = \exp[-t] \sum_{k=1}^{|V|^2} D_k(u, v) \frac{t^k}{k!} \quad (6)$$

The path length distribution is itself related to the eigenspectrum of the Laplacian. By equating the derivatives of the spectral and the path-length forms of the heat kernel it is straightforward to show that

$$D_k(u, v) = \sum_{i=1}^{|V|} (1 - \lambda_i)^k \phi_i(u) \phi_i(v) \quad (7)$$

Hence,  $D_k(u, v)$  can be interpreted as the sum of weights of all walks of length  $k$  joining nodes  $u$  and  $v$ .

## 2.6 Spectral decomposition of representation matrix

The spectrum of the graph is obtained from one of the representations given above using the eigendecomposition. Let  $X$  be the matrix representation in question. Then the eigendecomposition is  $X = \Phi \Lambda \Phi^T$  where  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{|V|})$  is the diagonal matrix with the ordered eigenvalues as elements and  $\Phi = (\phi_1 | \phi_2 | \dots | \phi_{|V|})$  is the matrix with the ordered eigenvectors as columns. The spectrum is the set of eigenvalues

$$\{\lambda_1, \lambda_2, \dots, \lambda_{|V|}\}$$

The spectrum is particularly useful as a graph representation because it is invariant under the similarity transform  $PLP^T$ , where  $P$  is a permutation matrix. In other words, two isomorphic graphs will have the same spectrum. As noted earlier, the converse is not true, two nonisomorphic graphs may share the same spectrum.

### 3 Cospectrality of graphs

Two graphs are said to be *cospectral* if they have the same eigenvalues with respect to the matrix representation being used. Haemers and Spence[4] have investigated the cospectrality of graphs up to size 11, extending a previous survey by Godsil and McKay[3]. They show that the adjacency matrix appears to be the worst representation in terms of producing a large number of cospectral graphs. The Laplacian is superior in this regard and the signless Laplacian even better. The signless Laplacian produces just 3.8% cospectral graphs with 11 vertices. Furthermore, there appears to be a peak in the fraction of cospectral graphs which then reduces. These results are shown in Figure 1, bottom right. Trees are known to be a particular problem with regard to cospectrality; Schwenk[10] showed that for large enough trees, they are nearly all cospectral to another tree.

Here we complement the investigation of Haemers and Spence by looking at the cospectrality of trees up to size 21. These trees were generated using the method described by Li and Ruskey[6]. It is clear from the matrix definitions above that the eigenvalues of  $L$ ,  $H$  and  $D_k$  are related and so cospectrality in one implies cospectrality in another. Since trees are bipartite, the spectrum of  $|L|$  is also related to  $L$ . We therefore confine our attention to  $A$ ,  $L$  and  $\hat{L}$ .

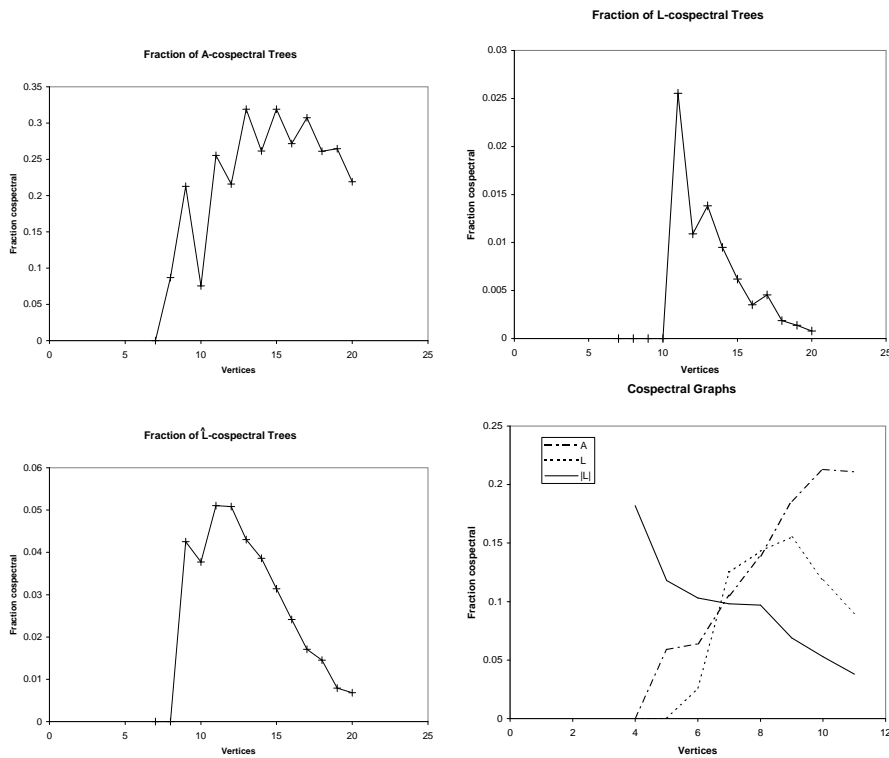


Figure 1: Fractions of trees which are cospectral with respect to the matrices  $A$ ,  $L$  and  $\hat{L}$ , and fractions of cospectral graphs[4]

Size	Number	$A$	$L$	$\hat{L}$	A & L (number)
8	23	0.087	0	0	0
9	47	0.213	0	0.0426	0
10	106	0.075	0	0.0377	0
11	235	0.255	0.0255	0.0511	2
12	551	0.216	0.0109	0.0508	2
13	1301	0.319	0.0138	0.0430	2
14	3159	0.261	0.0095	0.0386	10
15	7741	0.319	0.0062	0.0314	2
16	19320	0.272	0.0035	0.0241	14
17	48629	0.307	0.0045	0.0171	40
18	123867	0.261	0.0019	0.0145	38
19	317955	0.265	0.0014	0.0079	64
20	823065	0.219	0.0008	0.0068	148
21	2144505	0.213	0.0005	0.0036	134

Table 1: Fractions of trees which are cospectral with respect to the matrices  $A$ ,  $L$  and  $\hat{L}$

The results are summarised in Figure 1 and Table 1. The fractions here refer to the number of trees which do not have a unique spectrum. The Laplacian is clearly superior in this regard, having a very small fraction of cospectral graphs at all sizes. Both the Laplacian and its normalised counterpart show a decreasing trend, suggesting that for larger trees the fraction which are cospectral in these matrices could be negligible. The trend for the adjacency matrix is less clear, but the fraction appears to decrease after 15 vertices. Our results clearly show that the combinatorial Laplacian is by far the best representation in terms of the fraction of trees uniquely represented by the spectrum. This result is in line with that of Haemers and Spence[4] for all graphs which suggested that the signless Laplacian was the best.

The final column in Table 1 shows the number of pairs of trees which are cospectral in  $A$  and  $L$  at the same time. Interestingly, cospectral pairs for  $A$  and  $L$  seem to be uncorrelated with each other, and so combining the two spectra leads to very few cospectral graphs. For example, at 21 vertices, there are only 134 cospectral examples from more than 2 million trees.

## 4 Measuring the stability and representational power of eigenvalues

One aim in this paper is to assess the usefulness of the eigenvalues for representing the differences between graphs. In addition, we aim to determine which matrix representation is most appropriate for this task.

### 4.1 Graph distance

The fundamental structure of a pattern space can be determined purely from the distances between patterns in the space. There are a number of ways to measure the distance between two graphs, but the most appropriate in this case is the edit distance[8, 1]. The

edit distance is defined by a sequence of operations, including edge and vertex deletion and insertion, which transform one graph into another. Each of these operations has an associated cost, and the total cost of a sequence of edits is the sum of the individual costs. The sequence of minimal cost which transforms one graph into another is the edit distance between the graphs. In the examples here, we have assigned a cost of 1 to edge insertions and deletions. Clearly, if the spectrum is to be a good representation in this sense, then the requirement is that the distance between spectra should be related to the edit distance between the graphs.

## 4.2 Classification

Classifying a large number of different kinds of graphs is also a common and important task. Any representation which fails to do this well is not a particularly good or practical one. Therefore, as well as determining the distance between graphs, it is also important to be able to classify them using the representation. If the spectrum is a good representation, then we should be able to identify the class of a graph even under noisy conditions. In our second set of experiments, we therefore investigate the classification of graphs when the graphs to be classified are perturbed by edge deletion operations.

## 5 Experiments

In this section, we provide some experimental evaluation of the six graph representation methods given in the previous sections. There are two aspects of this study; first, we show that the more similar the two graphs are, the smaller the Euclidean distance of the eigenvalues will become. We use both Delaunay graphs and random graphs to demonstrate this. We also compute the relative deviation of the Euclidean distance to assess the accuracy of this relationship. Second, we compute the error rate for classification using random graph matching.

In the first experiment we compute the Euclidean distance between the vector of eigenvalues of the Delaunay graph with thirty vertices and its altered graph, modified by edge deletion from one to thirty edges, using six graph representation methods mentioned before. The edge to be deleted is chosen at random. For each level of editing, we perform 100 trials in order to obtain an average and deviation in the distance. The  $t$  in heat kernel equation is set to 3.5 and the length of path is path length distribution is 2. We can obtain the mean Euclidean distance and the standard deviation at each edge deletion of these matrix representations. The results are shown in Figure 2

The second experiment is much the same as the first one. The only difference is that this time we use random graphs. In this experiment, we generate random graph with thirty vertices and seventy edges. The other parameters are identical to the previous experiment.

These plots show that all these representations give a spectrum which follows the edit distance closely, although the adjacency and Laplacian matrices seem marginally less linear. In Tables 2 and 3 we give the relative deviation of the samples for 5, 10, 20 and 30 edit operations. The relative deviation is the standard deviation of the samples divided by the mean. This value gives an indication of how reliably the spectrum predicts the edit distance. In this regard, the heat kernel matrix is clearly superior to the other methods.

We now construct a classification experiment using 50 graph classes. Each class is represented by a single graph. We create graphs to be classified by performing random

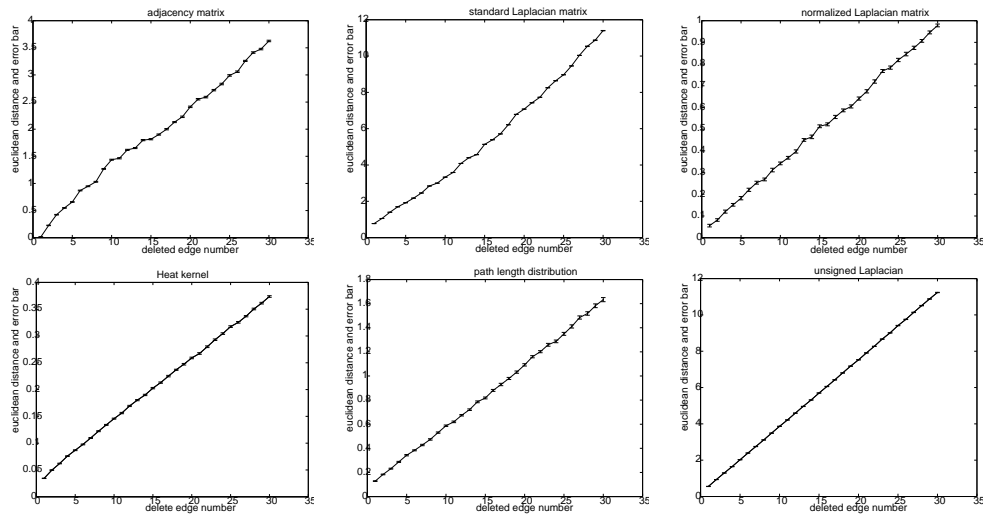


Figure 2: Euclidean distance of Delaunay graphs

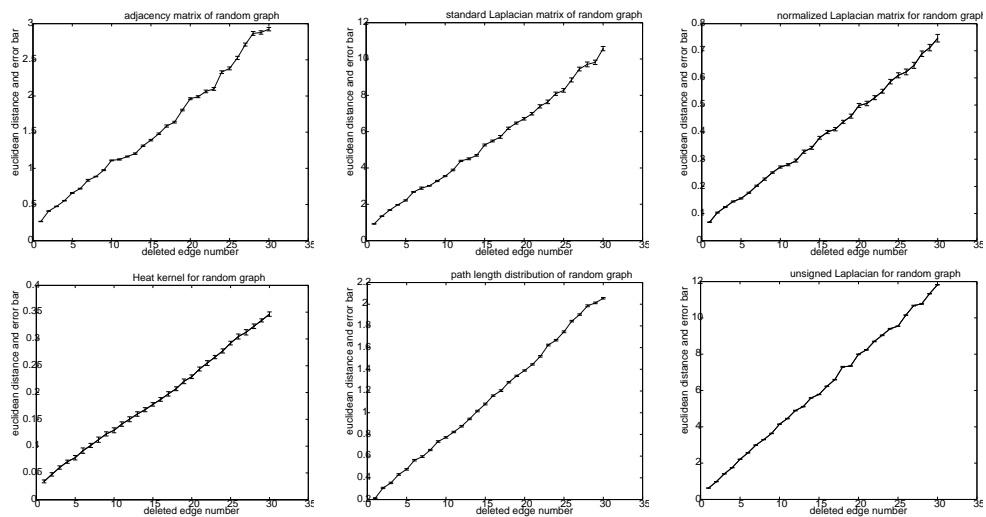


Figure 3: Euclidean distance of random graphs

edit operations on the class graphs. The graphs are classified using a simple 1-NN classifier and the Euclidean distance between the spectra; the aim here is to investigate the efficacy of the representation rather than the classifier. Figure 4 shows the classification error rates over a range of numbers of edit operations. Here the heat kernel matrix is the best method followed by the path length distribution. The adjacency matrix is a poor representation whereas the combinatorial and normalized Laplacian have the same performance.



Matrix	5 edge deletion	10 edge deletion	20 edge deletion	30 edge deletion
$A$	0.0918	0.0827	0.0716	0.0530
$L$	0.0802	0.0727	0.0619	0.0498
$\hat{L}$	0.0753	0.0676	0.0571	0.0414
$ L $	0.0523	0.0449	0.0268	0.0121
$H$	0.0358	0.0287	0.0193	0.0105
$D_2$	0.0420	0.0313	0.0252	0.0127

Table 2: Relative deviation of Delaunay graphs

Matrix	5 edge deletion	10 edge deletion	20 edge deletion	30 edge deletion
$A$	0.1164	0.1023	0.0805	0.0657
$L$	0.1042	0.0930	0.0771	0.0592
$\hat{L}$	0.0947	0.0830	0.0651	0.0558
$ L $	0.0647	0.0586	0.0401	0.0253
$H$	0.0582	0.0494	0.0299	0.0175
$D_2$	0.0607	0.0523	0.0385	0.0225

Table 3: Relative deviation of random graphs

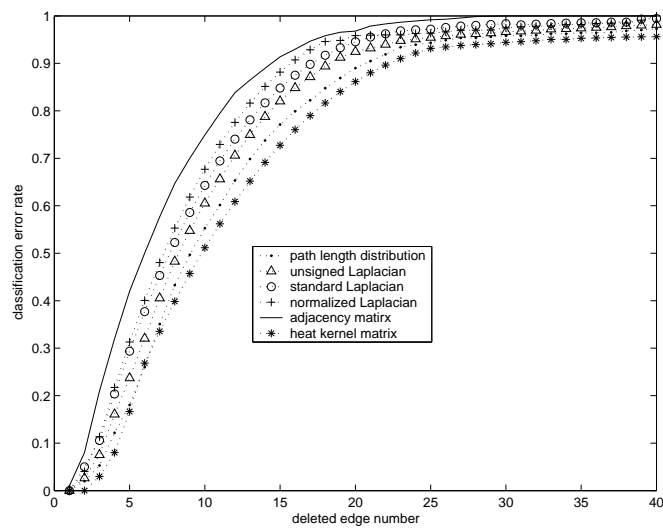


Figure 4: Error rate of six methods of matrix for random graphs

## 6 Conclusions

Our results show that use of the Laplacian matrix or its derivatives can drastically reduce the problem of cospectrality between trees. If the trend we have seen continues, then virtually all trees will have a unique Laplacian spectrum. In terms of a representation for graph matching, the heat kernel matrix outperforms the alternatives both in terms of

tracking edit distance and classification. Again, the adjacency matrix is inferior.

## References

- [1] H. Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18:689–694, 1997.
- [2] F. R. K. Chung. *Spectral Graph Theory*. AMS, 1997.
- [3] C. D. Godsil and B. D. McKay. Constructing cospectral graphs. *Aequationes Mathematicae*, 25:257–268, 1982.
- [4] W. H. Haemers and E. Spence. Enumeration of cospectral graphs. *European Journal of Combinatorics*, 25(2):199–211, 2004.
- [5] S. Kosinov and T. Caelli. Inexact multisubgraph matching using graph eigenspace and clustering models. *Structural, Syntactic and Statistical Pattern Recognition, LNCS*, 2396:133–142, 2002.
- [6] G. Li and F. Ruskey. The advantages of forward thinking in generating rooted and free trees. In *10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 939–940, 1999.
- [7] B. Luo, R. C. Wilson, and E. R. Hancock. Graph manifolds from spectral polynomials. In *17th International Conference on Pattern Recognition*, volume III, pages 402–405, 2004.
- [8] A. Sanfeliu and K. S. Fu. A distance measure between attributed relational graphs for pattern-recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13(3):353–362, 1983.
- [9] S. Sarkar and K. L. Boyer. Preceptual organization in computer vision. *IEEE Trans. Systems, Man and Cybernetics*, 23:382–399, 1993.
- [10] A. J. Schwenk. Almost all trees are cospectral. In F. Harary, editor, *New directions in the theory of graphs*, pages 275–307. Academic Press, New York, 1973.
- [11] J. Shi and J. Malik. Normalized cuts and image segmentation. *CVPR*, pages 731–737, 1997.
- [12] A. Shokoufandeh, S. Dickinson, K. Siddiqi, and S. Zucker. Indexing using a spectral coding of topological structure. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, number 491–497, 1999.
- [13] E. R. van Dam and W. H. Haemers. Which graphs are determined by their spectrum? *Linear Algebra and its Applications*, 356:241–272, 2003.
- [14] R. C. Wilson and E. R. Hancock. Pattern spaces from graph polynomials. In *12th International Conference on Image Analysis and Processing*, pages 480–485, 2003.
- [15] R. C. Wilson and E. R. Hancock. Contour segments from spline interpolation. In *Syntactic and Structural Pattern Recognition Workshop*, volume LNCS 3138, pages 57–65. Springer, 2004.