REPRESENTATION AND GENERATION OF PLANS USING GRAPH SPECTRA

099

Sean Hanna
The Bartlett School of Graduate Studies, UCL

Sean Hanna
The Bartlett School of Graduate Studies, University College London
1-19 Torrington Place, London, WC1E 7HB
s.hanna@cs.ucl.ac.uk

Abstract

Numerical comparison of spaces with one another is often achieved with set scalar measures such as global and local integration, connectivity, etc., which capture a particular quality of the space but therefore lose much of the detail of its overall structure. More detailed methods such as graph edit distance are difficult to calculate, particularly for large plans. This paper proposes the use of the graph spectrum, or the ordered eigenvalues of a graph adjacency matrix, as a means to characterise the space as a whole. The result is a vector of high dimensionality that can be easily measured against others for detailed comparison.

Several graph types are investigated, including boundary and axial representations, as are several methods for deriving the spectral vector. The effectiveness of these is evaluated using a genetic algorithm optimisation to generate plans to match a given spectrum, and evolution is seen to produce plans similar to the initial targets, even in very large search spaces. Results indicate that boundary graphs alone can capture the gross topological qualities of a space, but axial graphs are needed to indicate local relationships. Methods of scaling the spectra are investigated in relation to both global local changes to plan arrangement. For all graph types, the spectra were seen to capture local patterns of spatial arrangement even as global size is varied.

Introduction

Various graph types, including adjacency, axial and boundary graphs (Hillier & Hanson 1984; Turner 2005), have been used to effectively represent spaces, the comparison of which is then normally achieved by set scalar measures such as integration, control (Hillier & Hanson 1984) or clustering coefficient (Watts & Strogatz 1998) derived from these. These measures capture locally that particular quality of the space, but are not sufficient to identify the graph as a whole uniquely. For detailed statistical analyses, database search, and applications that may refer to the overall structure of the space a richer quantification of the entire space is desirable. This paper introduces the use of graph spectral analysis to automatically represent spaces by graphs derived from the plan. The resulting representation is

sufficiently rich to be used not only for detailed statistical comparison, but also in an optimisation algorithm to generate plans with similar spatial configuration. Both axial line and boundary graphs are evaluated as methods of representation, to determine which one is the more appropriate basis for measurement.

Several approaches to similarity measurement have been based on small graphs of adjacency or connectivity of spaces in plan. Dalton and Kirsan (2005) use the edit distance between two graphs to measure the similarity between buildings, and have shown that these correlate with cultural similarities and differences. Jupp and Gero (2003) suggest an analysis based on similarity and complexity measures of semantic graphs. With very large graphs as generated by axial lines (Turner 2005) or visibility graph analysis (Turner et al. 2001), calculation of similarity becomes more difficult. Graph spectra have been used in image analysis and pattern recognition to effectively index, classify and retrieve complex, high dimensional data (Luo et al. 2003; Robles-Kelly & Hancock 2003) and are used here in a similar manner.

Graphs may be initially derived from a given plan in several ways, each one encoding different features of spatial structure. This paper examines which graph types are necessary to capture the arrangement of the plan both in terms of topology and specific shape at both the local and global level. A definition of the graph types is given in the next section, followed by a description of the spectral feature vector. Section 4 examines the ways in which different spectra are affected by global and local variations in a set of test plans. To strictly and fully represent spaces, it is necessary that a procedure exists whereby any given plan can be mapped to exactly one spectrum, and also that the resulting spectrum can be mapped to exactly one plan. This second criterion is known not to hold true for graphs of small size, but it is thought that almost all graphs have unique spectra, increasingly so as the number of nodes increases to the level of detail in real plans (Van Dam and Haemers 2002), and in Section 5 a genetic algorithm (GA) is used to generate plans to match a given arrangement, thereby evaluating the effectiveness of the representation. The objective function is set to minimise the Euclidian distance between the generated population and a set target as represented by the graph spectra. Evolution is seen to produce plans similar to the initial targets, even in very large search spaces.

Selection of Plan Features

The plans used in this work are of office interiors, and consist of units of paired desks and chairs arranged orthogonally, but this overall structure is generic enough to just as well represent units of built/open space at an urban scale or other spaces. The situation of a desk, however, highlights the fact that there are two types of interaction involved at each unit: the connection of the chair to an open space of the floor, and the possible connection of the desk itself to other desks through which documents, messages and conversation might pass in the course of normal office activities. Three map types used by Hillier and Hanson (1984), the axial, convex and interface maps, result in graph representations of space, but this drawing of space alone does not capture the relationship between desks that face one another, so an additional graph of desk adjacencies is introduced.

Three map types have been used, two for open space and one for desk connections. Each captures different features of the plan, and their use alone and together will be compared. Common to all three is that the basic unit if spatial division is the desk or chair grid point rather than the more generic notion of the convex space. This follows roughly from the terminal nodes of buildings used in the interface map

099-02

(Hillier and Hanson 1984), which are essential in expressing connection in systems consisting of two types of space, and is also a standard which is economical to implement, particularly for the iterative optimisation in section 5. On this basis three graphs are constructed: a visibility graph between seats, and two versions of a modified boundary graph: one for open spaces and another for desk groups.

Axial / Visibility Graph Features

An algorithm was implemented to generate axial (visibility) maps from the cellular desk structure. This differs from the axial line map defined by plan vertices, and is more like Depthmap's (Turner 2001) grid visibility analysis, but the use of the set modules with chair points as the only possible nodes greatly reduces the computation time necessary to generate a graph. Graph nodes in this representation are not the lines themselves, but the chair points in the plan.

099-03

The algorithm simply draws links between from each chair node to every other chair node that can be connected by an unobstructed, direct sight line through empty space. Three examples of the resulting axial graphs are displayed in figure 1 (a).

Figure 1:

Graph types displayed on three different plans

a) Axial graphs

b) Boundary graphs

c) Desk graphs
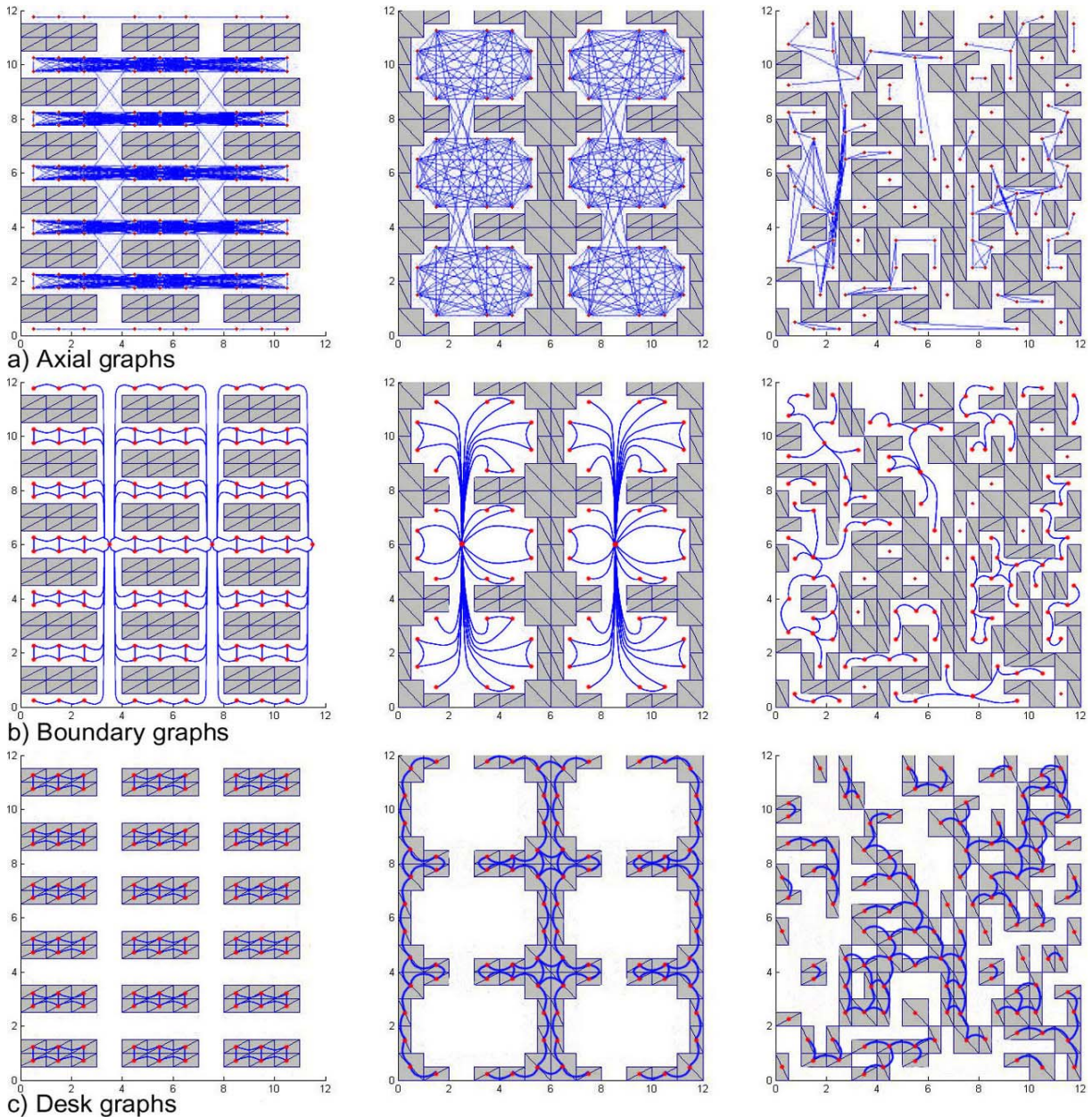
Boundary Graph Features

The boundary graph treats as a unity all space within a given boundary, regardless of its shape, and therefore captures all immediate or continuous spatial connections, regardless of sight. The basic node is still taken as the individual chair point, but empty space is also considered and links are drawn only between nodes that are directly face-wise adjacent. A chair point is connected to any other immediately behind or to the side or to an open space onto which it backs. All adjacent void spaces are then grouped together into one single node, regardless of size, representing all continuous space within the boundaries formed by the desks. This is roughly equivalent to the interface map (Hillier and Hanson 1984). Boundary graphs are shown in figure 1 (b).

Desk Graph Features

Desk graphs capture the relationships between adjacent desks. They are generated by the same method as the boundary graphs, except the graph nodes are the desks rather than chairs, and these are connected to all other adjacent desks. The resulting graph will generally not be unified, but segmented into discrete sub-graphs each corresponding to a connected group of desks. The desk graphs for three plans are in figure 1 (c).

Defining the Spectrum for Plan Representation

The spectrum of a graph, or ordered set of eigenvalues of its connectivity matrix, is useful in that it can be used to represent the graph as a single feature vector. This spectrum is useful as a representation of the graph because it is invariant under all permutations of the original matrix, and therefore identical for all isomorphic graphs. (Zhu and Wilson 2005) While it is possible that two non-isomorphic graphs can share the same spectrum, it has been suggested that this occurs less frequently as the graph size increases (Zhu and Wilson 2005) and therefore almost all graphs, particularly of the sizes yielded by plans, may be uniquely determined by their spectrum (Van Dam and Haemers 2002).

For any graph with a set of nodes V and a set of edges E, the most straightforward way of representing the graph in matrix form is to use the adjacency matrix A, a $|V| \times |V|$ matrix defined by:

$$A(i,j) = \begin{cases} 1 \text{ if } (i,j) \in E \\ \quad or, \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

Alternatively, the Laplacian is often used to represent the graph. Spectra of the Laplacian or its derivatives have been shown to be superior to the straight adjacency matrix for graph representation and classification (Zhu and Wilson 2005). In particular, they have been shown to have a higher correlation to graph edit distance (the number of edges deleted or added to change one graph to another) and result in fewer cospectral graphs (graphs which have identical spectra). Where the elements of a diagonal matrix $D = diag(\ deg(V_1),\ deg(V_2),\ \dots\ deg(V_{|V|}))$ indicate the degree of each of the nodes, the Laplacian is constructed from this and the adjacency matrix,

$$L = D - A \tag{2}$$

The spectrum of the graph is found by taking the eigendecomposition of the matrix representation. The eigenvalues λ and eigenvectors φ for A are given by solving for,

$$A = \Phi \Lambda \Phi^T \tag{3}$$

or

$$L = \Phi \Lambda \Phi^T, \tag{4}$$

where the matrix $\Phi = (\varphi^1 \mid \varphi^2 \mid \ldots \mid \varphi^{|V|})$ contains the eigenvectors as columns and the matrix $\Lambda = \mathrm{diag}(\lambda^1, \lambda^2, \ldots, \lambda^{|V|})$ contains the eigenvalues as diagonal elements. The spectrum is defined as the set of ordered eigenvalues

$$\{ \lambda^1, \lambda^2, \ldots, \lambda^{|V|} \} \tag{5}$$

Assembling the Feature Vector

Several approaches may be taken to assembling the spectral feature vector from the above set. It is essential that this be ordered consistently for all graphs, and for ease of comparison that it be a constant length.

While graph nodes and eigenvalues of the adjacency matrix have no intrinsic order, the spectrum has to be sorted such that any isomorphic graphs will have the same order of eigenvalues. In many analyses (Luo et al. 2003), values (and corresponding vectors) are sorted by absolute magnitude, such that $|\lambda^1| > |\lambda^2| > \ldots > |\lambda^{|V|}|$ and to ensure a constant length the spectrum is a vector composed of the first n values:

$$S = (\lambda^1, \lambda^2, \ldots, \lambda^n)^T \tag{6}$$

This can be problematic when the set of eigenvalues contains several values that are of the same magnitude, either positive or negative, and the resulting sort yields a different order for identical graphs. Sorting by actual value, including the sign such that $\lambda^1 > \lambda^2 > \ldots > \lambda^{|V|}$, avoids this problem and is the method used here.

Local / Global Features and Scale Changes

In representing the plan as a whole, both global attributes such as overall plan size, and local attributes such as desk configuration, are relevant. This section examines the plan spectra to determine the effect of changes to the plan at both scales on the feature vector.

Capturing Local and Global Features in the Spectrum

The spectra of two plan types were compared to judge the effects of global vs. local changes to each. In the plots below (figure 2), the same *local* arrangement of desks is repeated a different number of times for both the straight rows and the outward facing clusters, but the *global* configuration is changed by increasing the plan area and total number of desks. The spectra of the boundary graph (left) and axial graph (right) are shown, with axes scaled to the same overall length for ease of comparison. In both cases the overall distribution of values in the spectrum and their magnitudes appear identical, but are spread over a wider number of values when the plans increase in size. Spectra have a total number of values equal to the number of nodes in the graph, thus doubling the graph increases this.

Local changes to a simple pattern of desks affect both the values in the spectrum and their distribution. In the plans in figure 3, extra desks are added across the horizontal rows in the desk clusters, which increases the values of the spectrum and its overall length. Although the precise relationship is not immediately obvious, values appear to be roughly proportional to spatial integration – in particular the largest magnitude eigenvalues, which increase as the number of desks adjoining the open space grows.
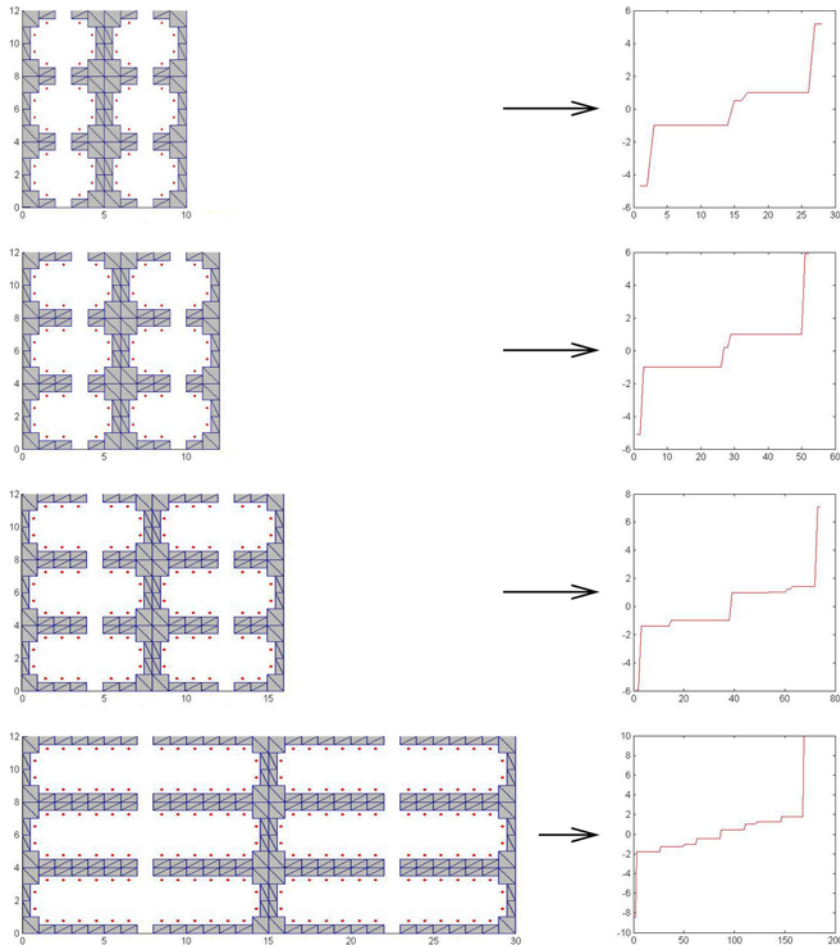
Figure 2:

Global changes to the plan increase the length of the spectrum, but not the distribution of values
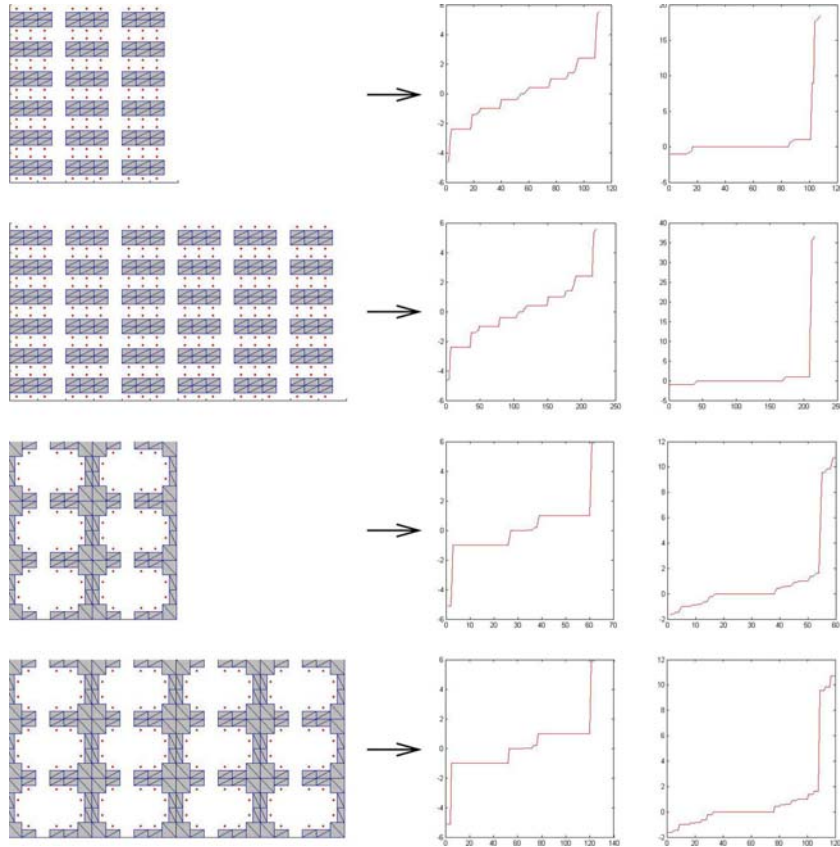
099-06

Figure 3:

Adding nodes locally to the plan increases the magnitude of the spectral range, and the number of distinct values

The number of positive or negative 'steps', or distinct values in the spectra of the boundary graphs shown, is always equal to the number

of nodes of distinct connectivity (degree) values in the graph, and the number of units within each 'step' is equal to the number of nodes of that type. In the bottom plan, for example, the first step is of two values of very high magnitude, corresponding to the two main open spaces. This is followed by four smaller steps corresponding to the four levels of integration along the rows of desks. These steps decrease appropriately in the smaller plans above.

Choices of Graph Spectra

The full graph spectrum has |V| values, or the number of nodes in the corresponding graph, but a feature vector of constant dimensionality is required even when comparing different sized graphs. To achieve this, the feature vector length n must be predetermined, and then either truncated by discarding some eigenvalues as in (6) or interpolated over n new values. In the second case a feature vector of dimensionality n is derived by performing a cubic spline interpolation of the |V| values in (5) to a new vector

$$S = (\; l^1, l^2, \dots, l^n)T \qquad\qquad (7)$$

Luo et al. (2003) suggest truncation as in (6), however the similarity of spectra as changes are made to global configuration (figure 2) would suggest that interpolation is also appropriate. The two methods were examined with the spectra of both the adjacency matrix and the Laplacian.

Zhu and Wilson (2005) measure the effectiveness of spectral representations by comparing sequential distances between spectra with known edit distances when creating the graphs themselves. A similar method is adopted here in a comparison of the spectra of plans that differ by a standard amount. Both global and local changes are considered. A new plan set, *SeqPlan*, was used, consisting of four groups of 100 plans, each of which varies monotonically in ten steps of two parameters. One of these parameters is local (number of desks to a group), and the other is global (overall plan size). Each of these four plan sets has a different general configuration of desks.

The distances in parameter space are taken as the number of additions to an initial plan: with the linear addition of between one and nine extra desks along a group, or the increase in plan size in increments of two units from 12 x 12 to 30 x 30. Spectral feature spaces that best describe the incremental progression should be those in which the Euclidian distances between spectra have the same linear progression – i.e. the ideal output in the plots in figure 4 will be linear.

Each plot in figure 4 displays these distances for each of the four plan types in the set for a single plan progression (solid line) and for the mean over ten of a similar type (dotted line). The mean relative deviation was measured for each. Progressions are shown for both the global and local parameters. The spectra are either interpolated or truncated to 100 dimensions, and spectra from both adjacency matrices (3) and Laplacians (4) are used.

All four spectral methods display roughly linear approximations to the increasing distances in parameter space. The choice between the use of the adjacency matrix or the Laplacian for the spectrum does not appear to make a great deal of difference, although the Laplacian slightly outperforms the adjacency in three out of four cases.

When global changes are made (plan sizes), the interpolated spectra perform reasonably well as suggested by the tests in section 4.1 (figure 2), however local changes (desk groups) are represented far better by truncation. Truncation yields a lower deviation than interpolation in each case.

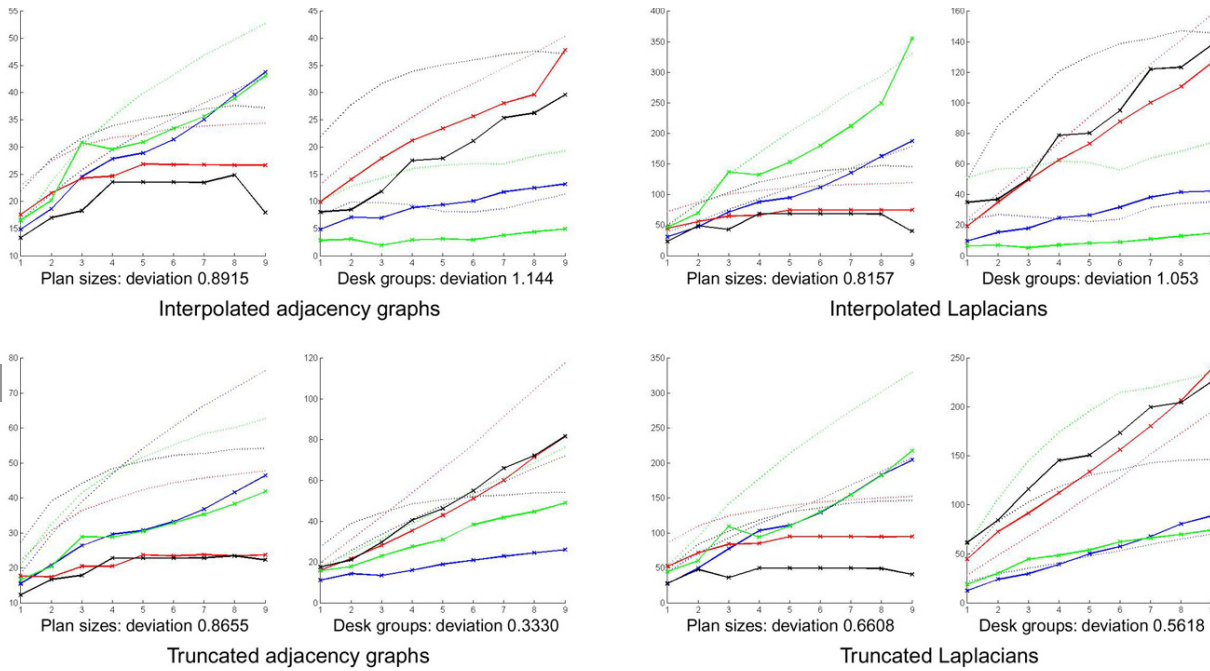Interpolated adjacency graphs

Interpolated Laplacians

Truncated adjacency graphs

Truncated Laplacians

099-08

Figure 4:

Incremental plan changes are plotted as distances in four spectral feature spaces

Optimisation toward Plan Spectra

To investigate the use of the GA in plan generation, optimisation was initially performed toward a preset goal of a given plan. This plan can be considered a *prototype*, as it is a single, real example that the optimisation algorithm is set to match. Evaluation of success in this case would thus be a simple comparison of how similar the result was to the initial goal plan, and so the objective function is a distance measurement in the *n*-dimensional space of the spectra. More precisely, the fitness is taken to be inversely proportional to the distance between a given plan's graph and that of the given prototype:

$$ f(i) = \sqrt{\sum_{i=1:n}(S_i(j) - S_{goal}(j))^2} \tag{8} $$

where S is the spectrum as given in (6) or (7).

The genome representation chosen was the simplest known to avoid (as much as possible) building in bias and as a general test of the method: each possible desk position on the planning grid is equated with a single base-6 allele indicating the orientation or absence of a desk. As a result the search space involved is vast (6^{n^2} for an *n* × *n* plan or roughly 10^{112} for even a small plan of n=12), and seemingly simple changes to plans as rotation of sections, mirroring, etc., can not be expressed except by individual changes to desks. It is highly unlikely that a matching plan will be found using this representation in a reasonable length of time, but the analysis of the results can indicate what plan features are being captured by the graphs. Boundary graphs only are investigated in the following section, and the results then compared with the use of other graphs.

Plan Matching with Boundary Graphs

The algorithm (a GA with a population size of 35) was tested with goals of two generic plan types: an arrangement of simple rows, and a linked set of outward facing clusters. The diagrams below show first the prototype plan (left), then the result of the fitness evaluation over time counted in generations, and the resulting final output when the optimisation was terminated.

Test plan 1

099-09

Test plan 2
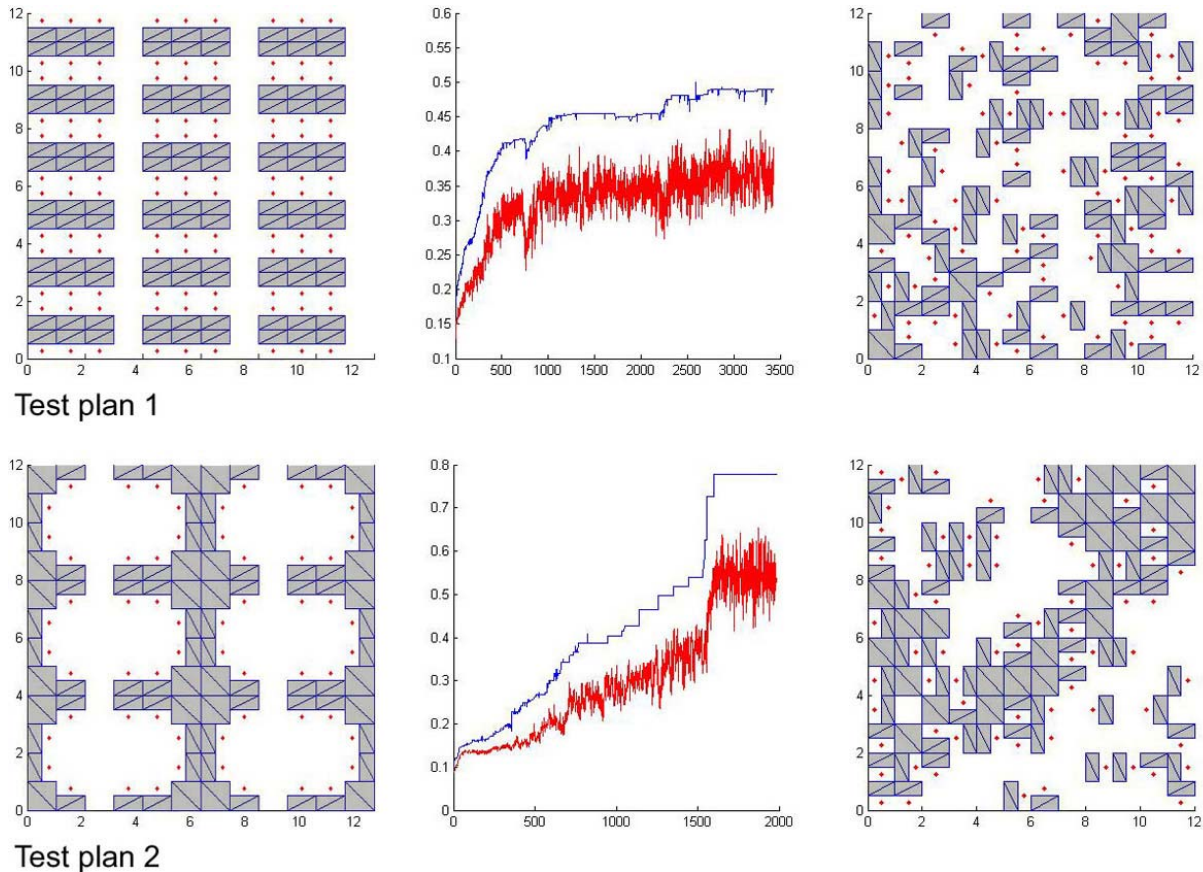
The results initially appear unpromising, except for the replication of the gross topology of the second plan: a series of two separated spaces divided by a central row of desks. A closer inspection reveals that the finer details of desk adjacency have also been duplicated, however, inasmuch as they are captured by the boundary graph alone. The graph represents all connected open spaces as a single node, and each chair as a separate node, with connections between face-wise adjacent neighbouring squares. The graph in the second test plan consists of two unconnected sub-graphs, each of which has a single node (of white space) connected to 12 pairs of adjacent desks and 6 single desks. The resulting optimised solution has a similar structure, except the number of pairs is only 10 and the number of connected single desks is 5 or 7. The structure of the first test plan is less clear, but the general arrangement of three main open spaces joined by groups of desks (in many cases they are three, but not all) is captured. In both cases the number of desks or ratio of empty space is also approximately correct.

Correlation between Topology and Fitness Level

A sharp increase in fitness is evident in the evolution of the plan to match the second test. Fitness jumps from about 0.5 to 0.8 in just a few generations around generation 1600. Because of the simplicity of the genome representation used, it appears likely that this was due to a sudden arrival at a large feature match like the overall topological division into two sub-graphs. This is examined more closely by comparing the spectra and fitnesses of very similar plans that do or do not display this topological division. The initial prototype plan is shown below (top), with its spectrum to the right. Below this are the plan produced by the genetic algorithm, and finally the same plan with several desks removed to connect the two separated spatial regions.
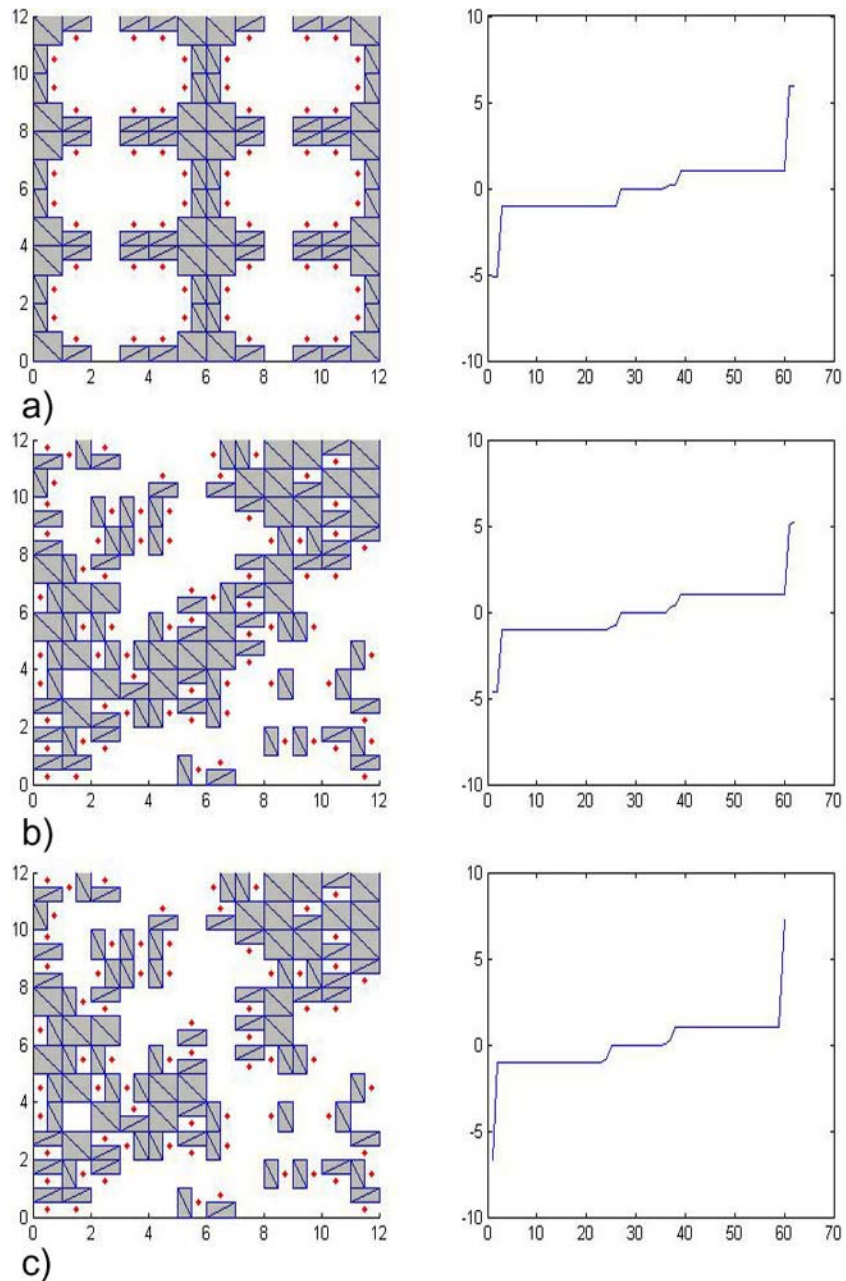
Figure 5:

Results of a GA search for a boundary graph spectrum taken from straight rows (above) and convex desk groups (below)

Figure 6:

Spectra of Topologically Similar (a & b), and Distinct Plans (c)

099-10

a)

b)

c)

The spectra appear very similar to one another in overall shape and magnitude, but in fact the numerical difference between the two lower plans is significant. The distance of the optimised plan (centre) from the goal is 1.2953, resulting in a fitness of 0.7780, whereas the distance of the linked plan below from the goal is 7.8061, resulting in a fitness of only 0.1281. This difference is largely due to the change in the eigenvalues of second greatest magnitude (plotted first and last on the horizontal axis, above), which correspond to the division of the graph into two distinct sections.

The overall form of the plan in each case looks unlike the initial goal, in part because of the vast search space and relative simplicity of the genome to efficiently represent patterns. But the plan similarities and the degree to which the spectrum of the evolved graph resembles its goal indicate this is due to the inherent limit of what the boundary graph alone can represent. As a method of fitness measurement, the distance between spectra appears to be an appropriate indication of overall topology, and the optimisation appears successful in capturing

the essential topological features represented by the boundary graph chosen.

Using Multiple Graphs and Spectra for GA Search

The use of the other graph types will be used to clarify the results obtained in section 5.1 above. Graph representations based on unobstructed lines of sight, either as axial lines or grid visibility graphs, are the most prevalent in space syntax analyses and the type represented in Depthmap. While the boundary graphs used above capture only the topology of the space, axial graphs record what can be seen from a given point, and therefore the shape of the space: the typical minimal line axial map produced by Depthmap, for example, represents each convex space by a retained axial line. The following tests compare the result of plan matching by GA with the use of the spectra of axial graphs as described in section 2.1, with those combining both axial and boundary graphs.

Figure 7:

Comparative results of a GA search for two plans using various graph types: (a) axial graphs, (b) axial and boundary graphs, (c) self-connected axial graphs, and (d, e) self-connected axial and boundary graphs

099-11

a) axial graphs only

b) axial and boundary graphs

c) self-connected axial graphs only

d) self-connected axial and boundary graphs

e) continued evolution with self-connected axial and boundary graphs
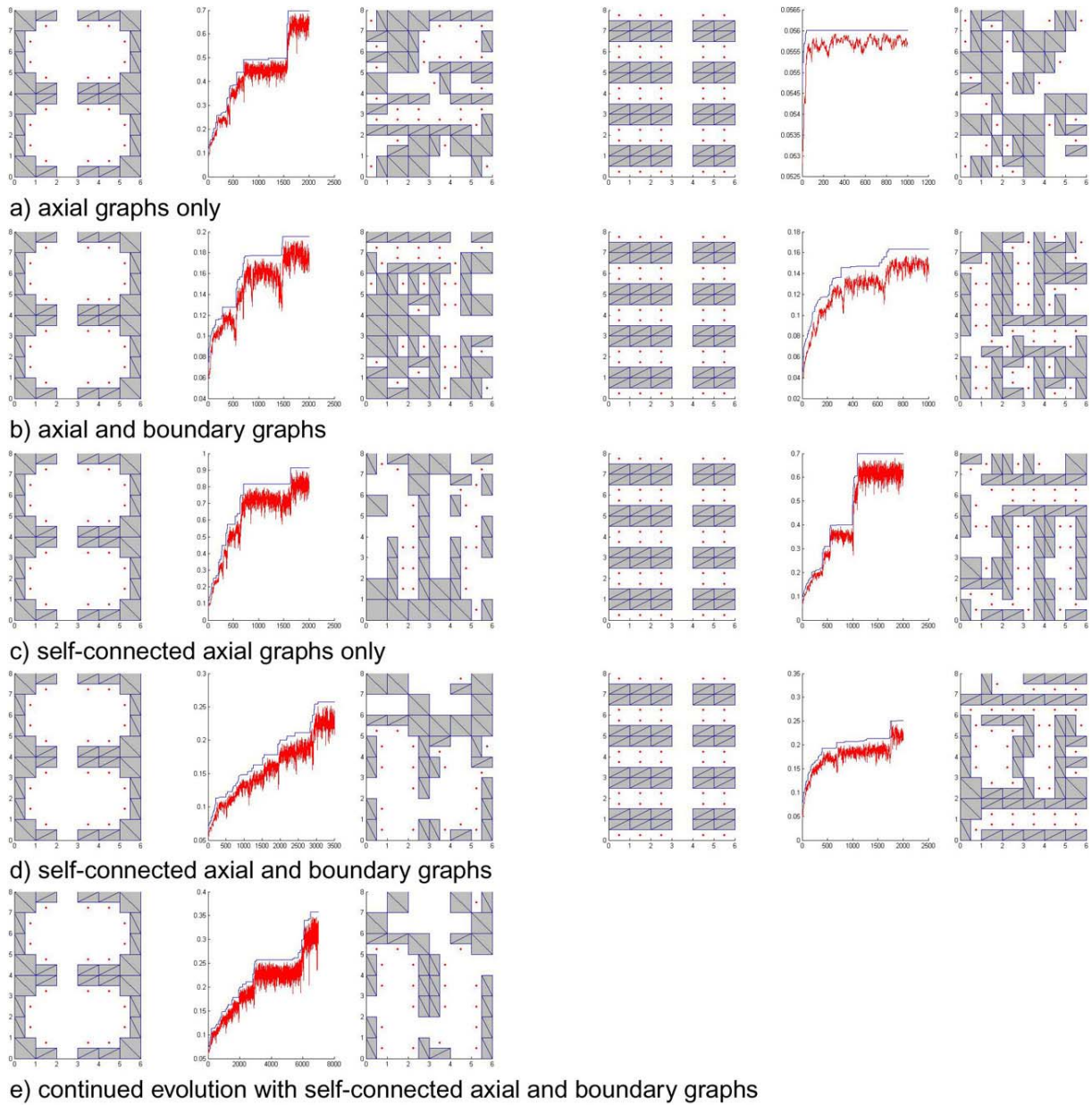
A further refinement to the representation is also tested. Axial graphs do not typically have self connected nodes, that is *diag(A)* = 0 for the adjacency matrix, which effectively removes nodes which have no other connections from the graph entirely. Isolated desks as appear in

the results of section 5.1 are therefore invisible, and ignored in the similarity measurement that determines fitness. The results are compared for both versions of the axial graph: without self-connected nodes (figure 7 a, b) and with self-connected nodes (figure 7 c, d, e).

The largest improvement in terms of the characterisation of the open spaces (and removal of unconnected nodes) came from connecting each node to itself, thus allowing unconnected nodes to appear on the resulting graphs and thereby providing an automatic penalty to the fitness.

In none of the examples is the original plan duplicated exactly, but as mentioned in section 5 it seems unlikely that the genome representation would do so even with a graph that captures all the features of the plan, simply because of the vastness of the search space. In most cases the fitness measurement rises initially but appears eventually to plateau at an incorrect local optimum, a result of the genome only expressing individual desks and not larger scale patterns. The rise in fitness appears smoother however when two graph types are used than with one.

The use of both graph types together also produces better evolved plans. It appears from the GA output that the axial method is far better at characterising the space than the boundary graphs alone, yielding clearer reconstructions of the initial plans. The plan in figure 7 (d, right) very closely resembles the convex group arrangement of its goal. If evolution is continued for several thousands of generations the plan improves to quite closely approximate the target plan (figure 7 e). The addition of both the boundary and desk graphs to the axial appears to yield marginally better results, although with a further increase in computation time. This similarity and correspondence of measured fitness to perceived plan similarity indicates that the essential features of the plan are indeed captured by the combination of plan spectra.

Conclusion

The spectra of various graphs has been shown in this paper to be an effective representation of spaces, which can be used to measure similarity of both global and local spatial structure. For all graph types, the spectra were seen to capture local patterns of spatial arrangement even as global size is varied, and thus may be used in comparing plans of differing overall scale. They constitute a reliable metric of plans, in that similar plans have spectral vectors that fall close together in a high dimensional space, while very different plans fall farther apart (section 4.2). In GA search, even with a large search space and an intentionally restricted genome, the resulting plans resembled the goal sufficiently well to suggest that the spectrum encodes almost all of the spatial structure of the plan.

Several graph types were examined, and results indicate that boundary graphs alone can capture the gross topological qualities of a space, but axial graphs are needed to indicate local relationships concerned with the actual shape of spaces and lines of sight. Using both the boundary and axial graphs together better represents the overall geometry of the space, and allows for a finer and more effective measurement of similarity. The result in a GA search appears to be that the fitness increase is smoother and the final plan more closely resembles the goal, as it is less likely to become trapped in a local optimum.

As a high dimensional vector the spectrum represents a more detailed description of the overall structure of the space than any single graph measurement taken in isolation, and can therefore be used for more detailed statistical comparison. Unlike string edit distance (Dalton and Kirsan 2005) and similar measurements, it is straightforward to

calculate even for large graphs and map to a universal space for comparison. Recent related work (Hanna 2006) has shown that axial graphs of buildings cluster well in PCA mapping into groups that correlate highly with building type. The use of a distance measurement has been used here to guide an optimisation algorithm to reproduce individual plans, but the technique may also be employed in comparative analyses between large sets of plans, and in any application for which a method of comparison is required but an appropriate single measure of a spatial structure is unknown.

099-13

References

Dalton, R.C., Kirsan, C., 2005, "Small Graph Matching and Building Genotypes", *Environment and Planning B: Planning and Design*, (In press).

Hanna, S., 2006, "Representing Style by Feature Space Archetypes: Description and Emulation of Spatial Styles in an Architectural Context, J.S. Gero (Ed.), *Design Computing and Cognition '06*, Springer.

Hillier, B. Hanson, J., 1984, *The Social Logic of Space*, Cambridge University Press.

Jupp, J.,Gero J. S., 2003, "Towards Computational Analysis Of Style In Architectural Design", S. Argamon (Ed.), *IJCAI03 Workshop on Computational Approaches to Style Analysis and Synthesis*, IJCAI, Acapulco, pp 1-10.

Luo, B., Wilson, R.C., Hancock, E.R., 2003, "Spectral Embedding of Graphs", *Pattern Recognition*, vol. 36, pp. 2213-2233.

Robles-Kelly, A., Hancock, E.R., 2003, "Edit Distance From Graph Spectra", *Proceedings*, 9th IEEE International Conference on Computer Vision (ICCV 2003).

Turner A., 2001, "Depthmap: A Program to Perform Visibility Graph Analysis", *Proceedings*, 3rd International Symposium on Space Syntax, pp. 31.1–31.9.

Turner A., 2005, "An Algorithmic Definition of the Axial Map", *Environment and Planning B: Planning and Design*, 32(3) 425-444.

Turner, A., Doxa, M., O'Sullivan, D., Penn, A., 2001, "From Isovists to Visibility Graphs: A Methodology for the Analysis of Architectural Space", *Environment and Planning B: Planning and Design*, 28(1) 103-121.

Van Dam E.R., Haemers, W.H., 2002, "Spectral Characterizations of Some Distance-regular Graphs", Journal of Algebraic Combinatorics, 15, pp. 189-202.

Watts D.J., Strogatz S.H., 1998, "Collective Dynamics of 'Small-world' Networks", *Nature*, vol. 393, pp. 440-442.

Zhu P., Wilson R. C., 2005, "A Study of Graph Spectra for Comparing Graphs", *British Machine Vision Conference 2005*, BMVA, pp 679–688.

099-14