

Commute Times, Discrete Green's Functions and Graph Matching

Huaijun Qiu and Edwin R. Hancock

Department of Computer Science,
University of York, York, YO10 5DD, UK

Abstract. This paper describes a graph-spectral method for simplifying the structure of a graph. Our starting point is the lazy random walk on the graph, which is determined by the heat-kernel of the graph and can be computed from the spectrum of the graph Laplacian. We characterise the random walk using the commute time between nodes, and show how this quantity may be computed from the Laplacian spectrum using the discrete Green's function. Our idea is to augment the graph with an auxiliary node which acts as a heat source. We use the pattern of commute times from this node to decompose the graph into a sequence of layers. These layers can be located using the Green's function. We exploit this decomposition to develop a layer-by-layer graph-matching strategy. The matching method uses the commute time from the auxiliary node as a node-attribute.

1 Introduction

Spectral graph theory [2] is concerned with characterising the structural properties of graphs using information conveyed by the eigenvalues and eigenvectors of the Laplacian matrix (the degree matrix minus the adjacency matrix). One of the most important tasks that arises in the analysis of graphs is that of how information flows with time across the edges connecting nodes. This process can be characterised using the heat equation [5]. The solution of the heat equation, or heat kernel, can be found by exponentiating the Laplacian eigensystem over time. The heat kernel contains a considerable amount of information concerning the distribution of paths on the graph. For instance, it can be used to compute the lazy random walk on the nodes of the graph. It may also be used to determine commute times under the random walk between pairs of nodes. An alternative, but closely related, characterisation of the graph is the discrete Green's function which captures the distribution of sources in the heat flow process. Not surprisingly, there is a direct link between commute times and the Green's function [3].

Random walks [12] have found widespread use in information retrieval and structural pattern analysis. For instance, the random walk is the basis of the Page-Rank algorithm which is used by the Googlebot search engine [1]. In computer vision random walks have been used for image segmentation [8] and clustering [11]. More recently both Gori, Maggini and Sarti [4], and, Robles-Kelly and Hancock [10] have used random walks to sort the nodes of graphs for the

purposes of graph-matching. However, most of these methods use a simple approximate characterisation of the random walk based either on the leading eigenvector of the transition probability matrix, or equivalently the Fiedler vector of the Laplacian matrix [6]. However, a single eigenvector can not be used to determine more detailed information concerning the random walk such as the distribution of commute times.

The aim in this paper is to draw on more detailed information contained within the Laplacian spectrum, and to use the commute time as means of characterising graphs. Our overall aim is to develop a means of graph matching. Rather than using the string-like characterisations that result from the approximate random walks used by Gori, Maggini and Sarti, and, Robles-Kelly and Hancock we aim to develop one based on the concentric layers that result by repeatedly peeling away the boundary of the graph. The reason for this is that the pattern of concentric layers is less likely to be disturbed by structural noise than the random walk, which can be diverted. To address this problem using the apparatus of the heat equation, we augment the graph with an auxiliary node. This node is connected to each of the boundary nodes by an edge, and acts as a heat source. Concentric layers are characterised using the commute time from the auxiliary node. We show how to compute the commute times using the Green's function for the graph, and this may be effected using the Laplacian spectrum. We match graphs by separately matching the concentric layers.

2 Heat Kernel and Path-Weighted Matrix

Let the weighted graph Γ be the quadruple (V, E, Ω, ω) , where V is the set of nodes, E is the set of arcs, $\Omega = \{W_u, \forall u \in V\}$ is a set of weights associated with the nodes and $\omega = \{w_{u,v}, \forall (u, v) \in E\}$ is a set of weights associated with the edges. Further let $T = \text{diag}(d_v; v \in V(\Gamma))$ be the diagonal weighted degree matrix with $T_u = \sum_{v=1}^n w_{u,v}$. The un-normalised weighted Laplacian matrix is given by $L = T - A$ and the normalized weighted Laplacian matrix is defined to be $\mathcal{L} =$

$$T^{-1/2} L T^{-1/2}, \text{ and has elements } \mathcal{L}_{uv}(\Gamma) = \begin{cases} 1 & \text{if } u = v \\ -\frac{w_{u,v}}{\sqrt{d_u d_v}} & \text{if } u \neq v \text{ and } (u, v) \in E. \\ 0 & \text{otherwise} \end{cases}$$

The spectral decomposition of the normalised Laplacian is $\mathcal{L} = \Phi \Lambda \Phi^T$, where $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{|V|})$ is the diagonal matrix with the ordered eigenvalues as elements satisfying: $0 = \lambda_1 \leq \lambda_2 \dots \leq \lambda_{|V|}$ and $\Phi = (\phi_1 | \phi_2 | \dots | \phi_{|V|})$ is the matrix with the ordered eigenvectors as columns.

In the paper we are interested in the heat equation associated with the graph Laplacian, i.e. $\frac{\partial \mathcal{H}_t}{\partial t} = -\mathcal{L} \mathcal{H}_t$ where \mathcal{H}_t is the heat kernel and t is time. The solution of the heat-equation is found by exponentiating the Laplacian eigenspectrum i.e. $\mathcal{H}_t = \exp[-t\mathcal{L}] = \Phi \exp[-t\Lambda] \Phi^T$. The heat kernel is a $|V| \times |V|$ matrix, and for the nodes u and v of the graph Γ the element of the matrix is $\mathcal{H}_t(u, v) = \sum_{i=1}^{|V|} \exp[-\lambda_i t] \phi_i(u) \phi_i(v)$.

Let us consider the matrix $P = I - \mathcal{L}$, where I is the identity matrix, then the heat kernel can be rewritten as $\mathcal{H}_t = e^{-t(I-P)}$. We can perform a McLaurin

expansion on the heat-kernel to re-express it as a polynomial in t . The result of this expansion is

$$\mathcal{H}_t = e^{-t(I-P)} = e^{-t} \left(I + tP + \frac{(tP)^2}{2!} + \frac{(tP)^3}{3!} + \dots \right) = e^{-t} \sum_{r=1}^{\infty} P^r \frac{t^r}{r!} \quad (1)$$

Using the spectral decomposition of the normalised Laplacian, we have $P^r = (I - \mathcal{L})^r = \Phi(I - \Lambda)^r \Phi^T$ and as a result

$$P^r(u, v) = \sum_{i=1}^{|V|} (1 - \lambda_i)^r \phi_i(u) \phi_i(v) = \sum_{\pi_r} \prod_i \frac{w(u_i, u_{i+1})}{\sqrt{d_{u_i} d_{u_{i+1}}}} \quad (2)$$

Hence, P^r can be interpreted as the sum of weights of all walks of length r joining nodes u and v . A walk π_r is a sequence of vertices u_0, \dots, u_r such that $u_i = u_{i+1}$ or $(u_i, u_{i+1}) \in E$.

3 Graph Derivation and the Multilayer Representation

We commence by constructing an augmented graph from the original graph by adding an auxiliary external node. We refer to this new graph as the *affixation graph*. It is constructed by connecting the additional node to each of the nodes on the boundary (or perimeter) of the original graph. Our aim in constructing this affixation graph is to simulate heat flow from the external node, which acts like an external heat source. We assign the label τ to the auxiliary node, and the *affixation graph* $\mathcal{A}(V', E')$ can be defined by $V' = V \cup \{\tau\}$ and $E' = E \cup \{(\tau, u), \forall u \in \text{Boundary}(\Gamma)\}$.

By analysing the heat-flow from the auxiliary node on the affixation graph, we can generate a multilayer representation of the original graph. The idea is to characterise the structure of the graph using the pattern of heat-flow from the source node. To embark on this study, let us first consider the relationship between the heat kernel and the lazy random walk.

Theorem 1. *The heat kernel is the continuous time limit of the lazy random walk.*

Proof. Consider a lazy random walk $R = (1 - \alpha)I + \frac{W}{T}\alpha$ which migrates between different nodes with probability α and remains static at a node with probability $1 - \alpha$, where W is the weighted adjacency matrix and T is the degree matrix.

Let $\alpha = \alpha_0 \Delta t$ where $\Delta t = \frac{1}{N}$. Consider the distribution $R(V_N | V_0)$, which is the probability of the random walk joining node 0 and N , in the limit $\Delta t \rightarrow 0$

$$\lim_{N \rightarrow \infty} R^N = \lim_{N \rightarrow \infty} \left(I + \left(\frac{W}{T} - I \right) \alpha_0 \frac{1}{N} \right)^N = e^{(\frac{W}{T} - I)\alpha_0} \quad (3)$$

while

$$\frac{W}{T} - I = T^{-1}A - I = T^{-1}(T - L) - I = -T^{-1}L \quad (4)$$

Now consider the discrete Laplace operator Δ with the properties:
 $\mathcal{L} = T^{1/2} \Delta T^{-1/2} = T^{-1/2} L T^{-1/2}$ which implies:

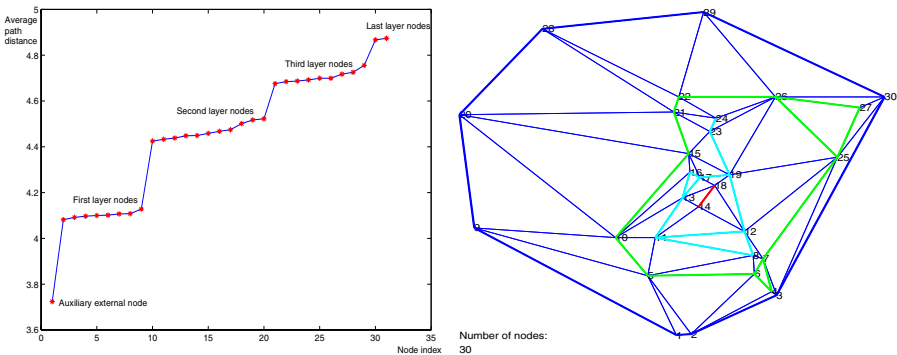
$$\Delta = T^{-1} L \tag{5}$$

Substituting Equation (4) and (5) into Equation (3), we get
 $\lim_{N \rightarrow \infty} R^N = e^{-\Delta \alpha_0}$ which is just the expression for the heat kernel. □

Corollary 1. The path-weighted matrix $P^r(u, v)$ defined in Equation 2 is the sum of the probabilities of all the lazy random walks of length r connecting node u and v .

Since P^r is the probability of the lazy random walk with a certain path length, we can make an estimate of the heat flow on the graph by taking the average value of P^r according to the path length r : $\mathcal{D}(u, v) = \frac{\sum_r r P^r(u, v)}{\sum_r P^r(u, v)}$. We take the external node τ to be the heat source and consider all the random walks starting from this point. The average path distance $\mathcal{D}(\tau, v)$ for all v in $V(\Gamma)$ follows a staircase distribution, which we can use to classify nodes into different layers.

Figure 1(a) illustrates this staircase property. The nodes with the same average distance correspond to the same layer of the graph. The corresponding multilayer graph representation is shown in Figure 1(b), where the nodes connected by edges of the same color belong to the same layer. Our matching process is based on the layers extracted in this way. To do this, we match the nodes in each layer in one graph to the nodes of the corresponding layer in a second graph. To do this we need a score-function to distinguish the different nodes in the same layer. Unfortunately, the average path distance can not be used for this purpose, since it is too coarsely quantised and can not be used to differentiate between the nodes in the same layer of a graph. We seek a



(a) Staircase distribution of the average path distance.

(b) An example of a multilayer graph.

Fig. 1. The staircase distribution and a multilayer graph

score function which is related to the heat kernel, and hence the heat-flow from the external source node, but offers us more distinct values for each individual node.

4 Score Function

Our score function is derived from the properties of the random walk. To commence, we note that the *hitting time* $Q(u, v)$ of a random walk on a graph is defined as the expected number of steps before node v is visited, commencing from node u . The *commute time* $O(u, v)$, on the other hand, is the expected time for the random walk to travel from node u to reach node v and then return. As a result $O(u, v) = Q(u, v) + Q(v, u)$. Our score function is the commute time between the external source node and the nodes of the original graph.

As we will demonstrate later, the commute time has some useful properties that render it suitable to our needs. First, we consider how the commute time can be computed.

The hitting time $Q(u, v)$ is given by [3]

$$Q(u, v) = \frac{vol}{d_v}G(v, v) - \frac{vol}{d_u}G(u, v)$$

where $vol = \sum_{v \in V(\Gamma)} d_v$ is the volume of the graph and function G is the Green's function on the graph.

The Green's function is the left inverse operator of the Laplace operator Δ , defined by $G\Delta(u, v) = I(u, v) - \frac{d_u}{vol}$. A physical interpretation of the Green's function is the temperature at a node in the graph due to a unit heat source applied to the external node. It is related with the heat kernel \mathcal{H}_t in the following manner

$$G(u, v) = \int_0^\infty d_u^{1/2} (\mathcal{H}_t(u, v) - \phi_1(u)\phi_1(v)) d_v^{-1/2} dt$$

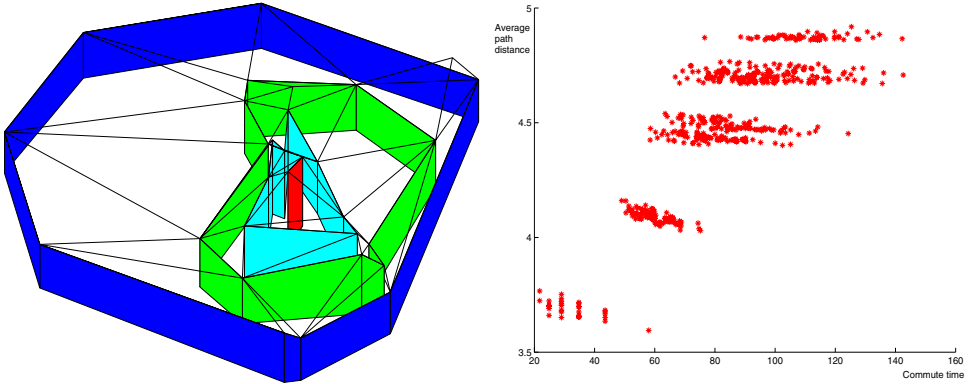
Here ϕ_1 is the eigenvector associated with eigenvalue 0 and its k -th entry is $\sqrt{d_k/vol}$. Furthermore, G can also be computed from the Laplacian spectrum using the formula $G(u, v) = \sum_{i=2}^{|V|} \frac{1}{\lambda_i} d_u^{1/2} \phi_i(u)\phi_i(v)d_v^{-1/2}$. So, the commute time is finally given by

$$O(u, v) = Q(u, v) + Q(v, u) = \frac{vol}{d_u}G(u, u) + \frac{vol}{d_v}G(v, v) - \frac{vol}{d_u}G(u, v) - \frac{vol}{d_v}G(v, u) \tag{6}$$

The score function S_u for node u is defined as $S_u = O(\tau, u)$ which is the commute time between node u and the external source node τ .

It is interesting to note that as consequence of (6) the commute time is a metric on the graph. The reason for this is that if we take the elements of G as inner products defined in a Euclidean space, O will become the norm satisfying: $\|x_i - x_j\|^2 = \langle x_i - x_j, x_i - x_j \rangle = \langle x_i, x_i \rangle + \langle x_j, x_j \rangle - \langle x_i, x_j \rangle - \langle x_j, x_i \rangle$.

Figure 2(a) shows a visualisation of the score functions for the Delaunay graph 1(b). Here the score is plotted on the z-axis. The scores for the nodes on the same layer are distinct enough to separate them. In Figure 2(b) we show a scatter plot of commute times $O(u, v)$ versus the average path length distance $\mathcal{D}(u, v)$. From this plot it is clear that the commute time varies more smoothly and has a larger range.



(a) 3D visualisation of the scores on the nodes. (b) Scatter plot of the commute time and the average path distance.

Fig. 2. 3D score visualisation and the scatter plot

5 Matching Process

Since we have divided the graph into several separate layers, our graph matching step can proceed on a layer-by-layer basis. To perform the matching process we peel layers of nodes from the boundary inwards. Each layer is a cycle graph where each node is connected to its two adjacent nodes only. In the case when a node has got only one neighbour in the layer, the edge between them is duplicated to form a cycle. We match the nodes in the corresponding layers of different graphs by performing a cyclic permutation of the nodes. The cyclic permutation permits possible null-insertions to accommodate missing or extraneous nodes. The cyclic permutation minimises the sum-of-differences in commute times between nodes in the graphs being matched. If C_l denotes the set of nodes in the k th layer of the graph, then the permutation \mathcal{P} minimises the cost function

$$\mathcal{E}(\mathcal{P}) = \sum_k \sum_{l \in C_k^M} \sum_{m \in C_k^D} (S_l - S_{\mathcal{P}(m)})^2$$

6 Experiments

The data used in our study is furnished by a sequence of views of a model-house taken from different camera viewing directions. In order to convert the images into abstract graphs for matching, we extract point features using a corner detector. Our graphs are the Delaunay triangulations of the corner-features. Examples of the images are shown in Figure 4.

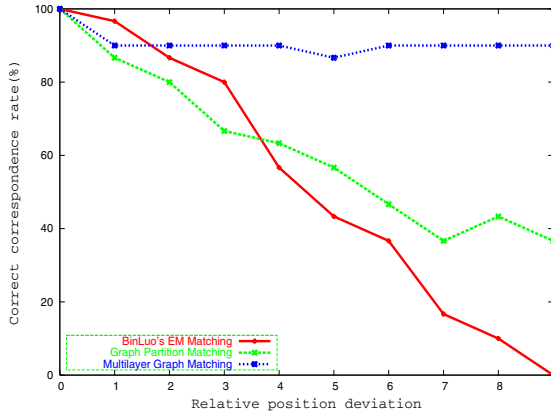


Fig. 3. Comparison of results

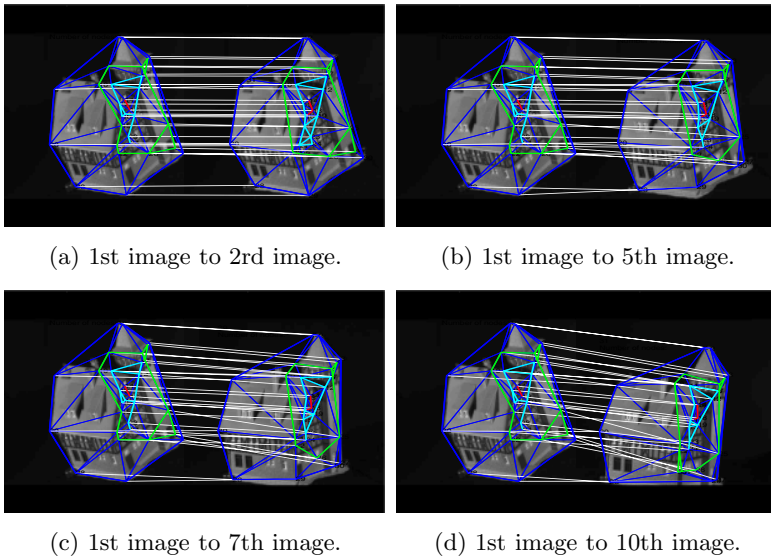


Fig. 4. Matched samples

We have matched the first image to each of the subsequent images in the sequence by using the multilayer matching method outlined earlier in this paper. The results are compared with those obtained using the method of Luo and Hancock [7] and the partition matching method of Qiu and Hancock [9] in Table 1. This table contains the number of detected corners to be matched, the number of correct correspondences, the number of missed corners and the number of miss-matched corners. Figure 3 shows us the correct correspondence rate as a function of the difference in view number for the three methods based on the data in the Table 1.

Table 1. Correspondence allocation results and comparison with the other methods

Method	House index	0	1	2	3	4	5	6	7	8	9
		Corners	30	32	32	30	30	32	30	30	30
EM[7]	Correct	-	29	26	24	17	13	11	5	3	0
	False	-	0	2	3	8	11	12	15	19	24
	Missed	-	1	2	3	5	6	7	10	8	6
Partition matching[9]	Correct	-	26	24	20	19	17	14	11	13	11
	False	-	3	5	8	11	12	16	15	17	19
	Missed	-	1	1	2	0	1	0	4	0	0
Multilayer matching	Correct	-	27	27	27	27	26	27	27	27	27
	False	-	3	3	2	2	3	2	2	2	2
	Missed	-	0	0	1	1	1	1	1	1	1

From the results, it is clear that our new method out performs both Luo and Hancock's EM method and, Qiu and Hancock's partition matching method for large differences in viewing angles. Figure 4 shows the results for some examples image pairs. There are clearly significant structural differences in the graphs including rotation, scaling and perspective distortion. Even in the worst case, our method has a correct correspondence rate of 86.67%.

7 Conclusion

In this paper we have described how the commute time to an auxiliary node can be used for the purposes of graph matching. We make two uses of this attribute. First, we use it to define cycle graphs, concentric with the boundary of the graph. Second, we use it as a node attribute which is used to establish correspondences between the cycles under cyclic-permutation. We demonstrate how to compute the commute time using the discrete Green's function of the graph, and explain how this is related to the Laplacian spectrum. Our future plans involve using the commute times to embed the nodes of the graph in a low dimensional space, and to use the characteristics of the embedded node points for the purposes of graph-clustering.

References

1. S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.
2. F.R.K. Chung. *Spectral Graph Theory*. CBMS series 92. American Mathematical Society Ed., 1997.
3. F.R.K. Chung and S.-T. Yau. Discrete green’s functions. In *J. Combin. Theory Ser.*, pages 191–214, 2000.
4. M. Gori, M. Maggini, and L. Sarti. Graph matching using random walks. In *ICPR04*, pages III: 394–397, 2004.
5. R. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. *19th Intl. Conf. on Machine Learning (ICML) [ICM02]*., 2002.
6. Lszl Lovsz. Random walks on graphs: A survey.
7. B. Luo and E. R. Hancock. Structural graph matching using the em algorithm and singular value decomposition. *IEEE PAMI*, 23(10):1120–1136, 2001.
8. M. Meila and J. Shi. A random walks view of spectral segmentation, 2001.
9. H. Qiu and E.R. Hancock. Spectral simplification of graphs. ECCV, 2004.
10. A. Robles-Kelly and E. R. Hancock. String edit distance, random walks and graph matching. In *Int. Journal of PRAI*, 18(3):315–327, 2004.
11. M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. In *LN-AI*, 2004.
12. V. Sood, S. Redner, and D. ben Avraham. First-passage properties of the erdos-screnyi random graph. *J. Phys. A: Math. Gen.*, (38):109–123, 2005.