Synthesis of Median Spectral Graph

Miquel Ferrer¹, Francesc Serratosa¹, and Alberto Sanfeliu²

¹ Universitat Rovira I Virgili, Dept. d’Enginyeria Informàtica i Matemàtiques, Spain  
{miquel.ferrer,francesc.serratosa}@urv.net  
² Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial, Spain  
anfeliu@iri.upc.es  

Abstract. In pattern recognition, median computation is an important technique  
for capturing the important information of a given set of patterns but it has the  
main drawback of its exponential complexity. Moreover, the Spectral Graph  
techniques can be used for the fast computation of the approximate graph  
matching error, with a considerably reduced execution complexity. In this pa-  
paper, we merge both methods to define the Median Spectral Graphs. With the use  
of the Spectral Graph theories, we find good approximations of median graph.  
Experiments on randomly generated graphs demonstrate that this method works  
well and it is robust against noise.

1 Introduction

Attributed Graphs (AGs) has been used to solve computer vision problems for deca-  
des and in many applications. Some examples include recognition of graphical sym-  
bols, character recognition, shape analysis, 3D-object recognition and video and im-  
age database indexing. In these applications, AGs represent both unclassified objects  
(unknown input patterns) and prototypes. Moreover, these AGs are typically used in  
the context of nearest-neighbour classification. That is, an unknown input pattern is  
compared with a number of prototypes stored in the database. The unknown input is  
then assigned to the same class as the most similar prototype.

Nevertheless, the main drawback of representing the data and prototypes by AGs is  
the computational complexity of comparing two AGs. The time required by any of the  
optimal algorithms may in the worst case become exponential in the size of the AGs.  
The approximate algorithms, on the other hand, have only polynomial time com-  
plexity, but do not guarantee to find the optimal solution.

Moreover, in some applications, the classes of objects are represented explicitly by  
a set of prototypes which means that a huge amount of model AGs must be matched  
with the input AG and so the conventional error-tolerant graph matching algorithms  
must be applied to each model-input pair sequentially. As a consequence, the total  
computational cost is linearly dependent on the size of the database of model graphs  
and exponential (or polynomial in subgraph methods) with the size of the AGs. For  
applications dealing with large databases, this may be prohibitive.

To alleviate these problems, some attempts have been made to try to reduce the  
computational time of matching the unknown input patterns to the whole set of mod-  
els from the database. Assuming that the AGs that represent a cluster or class are not  
completely dissimilar in the database, only one structural model is defined from the  
AGs that represent the cluster, and thus, only one comparison is needed for each clus-  
ter.
There are two different methodologies to represent the cluster in the literature depending on whether they keep probabilistic information in the structure that represent the cluster of AGs or not. In the probabilistic methods, the models (clusters) are described in the most general case through a joint probability space of random variables ranging over graph vertices and arcs. They are the union of the AGs in the cluster, according to some synthesis process, together with its associated probability distribution [1,2,3]. In the non probabilistic methods, clusters are represented by an AG (which might not be in the cluster) or they are represented by a network of models [4].

Spectral graph theory is concerned with understanding how the structural properties of graphs can be characterised using the eigenvectors of the adjacency matrix of the AGs or the Covariance matrix [5]. Although spectral methods have been used to address the segmentation or correspondence matching problems, there has been less work on using spectral characteristics to perform pattern analysis on AGs. First, an approximate solution to the graph matching problem was presented in [6,7,8] for both undirected and directed AGs based on the eigendecomposition of the adjacency matrix of both graphs. The method was restricted to AGs with only one positive attribute on the nodes and arcs. Recently, AGs with complex numbers as attributes on the nodes or arcs were allowed in the method presented in [9,10], rather than purely real entries.

Given a set of AGs, the median is defined as the AG that has the smallest sum of the distances to all AGs in the set [11]. We can distinguish between set median and generalised median graphs. The difference lies in the space of AGs where the respective median is searched for (formal definitions in section 5).

In this paper, we first define a method to find a sub-optimal labelling between AG vertices based on the correlation between the modal matrices obtained from the adjacency matrices of both AGs (section 4). Moreover, we introduce the novel concepts of set and generalised-median spectral graphs (section 5). While the computation for the set-median spectral graphs is exponential in the size of the input graphs, but polynomially bounded by the number of those graphs, the complexity of computing generalised-median spectral graphs is exponential in both the number of input graphs and their size. For this reason, and with the aim of reducing the exponential complexities, we develop an incremental algorithm in section 6 to synthesise an approximation of the generalised-spectral graph in polynomial complexity respect the number of AGs. Experiments conducted on median spectral graphs in section 7 demonstrate the advantage of this representation and the ability of our synthesis method to find approximate generalised-median spectral graphs.

2 Formal Definitions of Attributed Graphs

An attributed graph $G$ over the domain of the attribute vertices and arcs $(\Delta_v, \Delta_e)$ with an underlying graph structure $H=(\Sigma_v, \Sigma_e)$, where $\Sigma_v = \{v_k | k = 1, ..., n\}$ is a set of vertices (or nodes) and $\Sigma_e = \{e_{ij} | i, j \in \{1, ..., n\}, i \neq j\}$ is a set of arcs, is defined to be a pair $(V, E)$ where $V = (\Sigma_v, \gamma_v)$ is an attributed vertex set and $E = (\Sigma_e, \gamma_e)$ is an attributed arc set. The mappings $\gamma_v : \Sigma_v \rightarrow \Delta_v$ and $\gamma_e : \Sigma_e \rightarrow \Delta_e$ assign attribute values to
vertices and arcs, respectively, where $\Delta_c = \Delta_a \cup \{\Phi\}$ and $\Delta_o = \Delta_a \cup \{\Phi\}$. $\Phi$ ($\Phi \notin \Delta_a$ and $\Phi \notin \Delta_c$) represents a null value of a graph element.

An adjacency matrix $A$ of an attributed graph $G$ of order $n$ is an $n \times n$ matrix whose element with row index $i$ and column index $j$ is:

$$A(i,j)= \begin{cases} y_{e}(e_{ij}) & \text{if} \ e_{ij} \in \Sigma_e \\ \Phi & \text{otherwise} \end{cases}$$

### 3 Spectral Decomposition of Matrices

**Spectral decomposition of matrices:** The spectral decomposition of matrices is obtained as follows. From matrix $A$ to hand, we can calculate the eigenvalues $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$ by solving the equation $|A - \lambda I| = 0$. Moreover, the modal matrix (also called eigenvector matrices) $U = (u_1| u_2| \ldots | u_n)$, composed by the eigenvectors associated to the eigenvalues $\lambda$, is obtained by solving the system of equations $Au = \lambda u$, were $w$ is the eigenmode index and the order of the eigenvectors is decided according to the decreasing magnitude of the eigenvalues, i.e. $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. We emphasise that in the case that the initial matrix is not symmetric, the elements of the modal matrix are complex numbers but in the case that the initial matrix is symmetric, the elements of the modal matrix are real numbers. The original matrix $A$ can be recovered by its eigenvectors and eigenvalues, $A = U \text{diag}(\lambda) U^T$. See [12] for more details.

**Correlation between matrices:** Given a pair of matrices $A = (a_1| a_2| \ldots | a_n)$, and $B = (b_1| b_2| \ldots | b_n)$ of $n \times n$ rows and columns, the correlation $\Gamma$ between them is defined as,

$$\Gamma(A,B) = \max_{\gamma} \sum_{i=1}^{n} a_{\gamma} b_{\gamma}^T$$

### 4 Error-Tolerant Graph Matching

From this section to the rest of the paper we consider that the domain of the vertices and arcs, $\Delta_c$, is the set of the non-negative numbers. The null attribute, $\Phi$, is represented by zero. And also, nodes have no attributes, that is, $\Delta_c = \{\Phi\}$. With this conditions, the adjacency matrix totally characterises the AGs and it is composed by non-negative numbers.

#### 4.1 Distance Between Attributed Graphs Given a Labelling

Let $G^1 = (V^1, E^1)$ and $G^2 = (V^2, E^2)$ be two AGs with $n$ nodes. A global cost $C_f$ can be associated with each structurally correct labelling between vertices of both graphs $f^{1,2} : \Sigma_v \rightarrow \Sigma_v$, and the distance measure between them is defined as the minimum of all such costs [13]:

$$d(G^1, G^2) = \min_{f^{1,2}} \{ C_{f^{1,2}} (G^1, G^2) \}$$


We call the optimal labelling as the labelling that gives the minimum cost, that is, the one used to compute the distance. Finally, the global cost \( C_{f^{i,2}} \) is defined as the difference of the attribute values as follows,

\[
C_{f^{i,2}}(G^1, G^2) = \sum_{i=1}^{n} \sum_{j=1}^{n} \left( y_i^j - y_{w + (f^{i,2}(v_i), v_j)} \right)^2
\]  

(3)

If \( A_{G^1} \) and \( A_{G^2} \) are the adjacency matrices of the AGs \( G^1 \) and \( G^2 \), the global cost \( C_{f^{i,2}} \) defined in (2) can be reformulated as follows using a permutation matrix \( P \) \[6]\:

\[
C_{f^{i,2}} = \left\| PA_{G^1} P^T - A_{G^2} \right\|^2
\]  

(4)

where \( P \) represents the isomorphism \( f^{i,2} \), that is,

\[
P(i, j) = \begin{cases} 
1 & \text{if } f^{i,2}(v_i) = v_j \\
0 & \text{otherwise} 
\end{cases}
\]

(5)

and \( \| \cdot \| \) is the Euclidean norm, \( \| A \| = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|^2} \).

Note that due to \( f^{i,2} \) has to be defined bijective, \( P \) has only one 1 in each row and column. Thus, the problem of finding the optimal labelling is reduced to the problem of finding the permutation matrix \( P \) which minimises \( C_{f^{i,2}} \). We show in the next section how to find the labelling \( f^{i,2} \) using the spectral graph theory.

4.2 Optimal Labelling Between Attributed Graphs

Let \( U_{G^1} = [u^1_1 | u^1_2 | u^1_n] \) and \( U_{G^2} = [u^2_1 | u^2_2 | u^2_n] \) be the modal matrices of the adjacency matrices \( A_{G^1} \) and \( A_{G^2} \). Consider that the eigenvectors \( u_i \) have been enumerated depending on the value of their eigenvalues, that is, \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \).

If we want to find the best labelling \( f^{i,2} \) between \( G^1 \) and \( G^2 \) using the spectral graph theory, we need to project or find a relation between the vertices of the AGs and the spectral decomposition of their adjacency matrices. To do so, we define arbitrarily the bijective functions \( h^1 \) and \( h^2 \) such that \( h^1 : v_i^1 \rightarrow u_i^1 \) and \( h^2 : v_i^2 \rightarrow u_i^2 \) and also the diagonal matrices \( H^1 \) and \( H^2 \) that represent these isomorphisms.

\[
\begin{align*}
G^1 & \xrightarrow{f^{i,2}} G^2 \\
\{ u^1_1 \} & \xrightarrow{f^1} \{ u^2_1 \} \\
\{ u^1_2 \} & \xrightarrow{f^1} \{ u^2_2 \} \\
\{ u^1_n \} & \xrightarrow{f^1} \{ u^2_n \}
\end{align*}
\]

\[
\begin{align*}
G^1 & \xrightarrow{f^{i,2}} G^2 \\
\{ u^1_1 \} & \xrightarrow{f^2} \{ u^2_1 \} \\
\{ u^1_2 \} & \xrightarrow{f^2} \{ u^2_2 \} \\
\{ u^1_n \} & \xrightarrow{f^2} \{ u^2_n \}
\end{align*}
\]

Fig. 1. a) Concatenation of the labelling function between vertices. b) Scheme of the computation of the labelling.

Thus, the function \( f^{i,2} \) is defined as the concatenation \( f^{i,2} = h^1 \circ f^1 \circ h^{-1} \) where \( f^1 \) is a bijective function between the modal matrices \( U_{G^1} \) and \( U_{G^2} \) (figure 1.a). Moreover, the matrix \( P \) can be redefined as,
\[ P = H^1 M H^2 \]

where \( M \) represents the isomorphism \( \gamma^1, \) that is,

\[
M(i,j) = \begin{cases} 
1 & \text{if } \gamma^1(x_i) = u_j \\
0 & \text{otherwise} 
\end{cases}
\]

As demonstrated in [6], the global cost \( C_{\gamma^1} \) is minimised when the correlation \( \Gamma \) between the absolute value of the modal matrices \( U_{G^1} \) and \( U_{G^2} \) is maximised. Then, the permutation matrix \( M \) has to be defined such that this correlation is maximised. We can use an algorithm with exponential cost, i.e., the A*, and get the optimal labelling or choose an algorithm with polynomial cost and get a sub-optimal labelling. For instance, applying the Hungarian method to \( U_{G^1} U_{G^2}^{-T} \) matrix. Figure 1.b shows the basic scheme the method.

### 4.3 Example of Graph Matching by Spectral Graphs

Assume that we want to compute the distance measure between \( G^1 \) and \( G^2 \) and decide the optimal labelling between their vertices. Figure 2 shows the AGs \( G^1 \) and \( G^2 \) and their adjacency matrices. Lines without arrows represent undirected arcs (the attribute in both directions of the edge is the same).

![Fig. 2. The graphs G^1 and G^2 and their adjacency matrices.](image)

The obtained eigenvalues and eigenvectors are

\[
\lambda^{G^1} = (14.794, \ -0.346, \ -1.974, \ -3.656, \ -8.816) \quad \lambda^{G^2} = (15.371, \ 0.036, \ -1.368, \ -4.255, \ -9.783)
\]

\[
U^{G^1} = \begin{pmatrix} 
0.268 & -0.720 & -0.604 & -0.204 & -0.041 \\
0.496 & -0.064 & 0.499 & -0.499 & -0.505 \\
0.487 & 0.597 & -0.408 & -0.452 & 0.344 \\
0.539 & -0.239 & 0.250 & -0.269 & 0.648 \\
0.039 & 0.380 & 0.297 & 0.646 & -0.453 
\end{pmatrix} \quad U^{G^2} = \begin{pmatrix} 
-0.499 & -0.076 & 0.578 & -0.337 & 0.543 \\
-0.473 & -0.199 & -0.374 & 0.683 & 0.359 \\
-0.374 & -0.575 & -0.429 & -0.512 & -0.285 \\
-0.562 & 0.269 & 0.317 & -0.220 & -0.679 \\
-0.264 & 0.742 & -0.489 & -0.328 & 0.177 
\end{pmatrix}
\]

Thus, the \( U_{G^1} U_{G^2}^{-T} \) matrix and the permutation matrix obtained by the Hungarian method is,

\[
\begin{pmatrix} 
0.6308 & 0.6516 & 0.9761 & 0.8916 & 0.6103 \\
0.9838 & 0.9542 & 0.6766 & 0.8353 & 0.9077 \\
0.8174 & 0.8900 & 0.9567 & 0.9959 & 0.8442 \\
0.8847 & 0.9881 & 0.7411 & 0.9045 & 0.8788 \\
0.9603 & 0.8813 & 0.7200 & 0.8405 & 0.9938 
\end{pmatrix} \quad P = \begin{pmatrix} 
0 & 0 & 1 & 0 \ 0 & 1 & 0 & 0 \ 0 & 0 & 0 & 1 \ 0 & 1 & 0 & 0 \ 0 & 0 & 0 & 1 
\end{pmatrix}
\]

The labelling that the matrix \( P \) represents is \( f^2(1)=3, \ f^2(2)=1, \ f^2(3)=4, \ f^2(4)=2 \) and \( f^2(5)=5 \) and the value of the global cost (3) is: 6.
5 Median Spectral Graphs

Let $Y$ be the set of all AGs that can be constructed using the domains of the attribute vertices and arcs $(\Delta_r, \Delta_l)$. Given a set of AGs $S = \{G^1, G^2, ..., G^n\}$, the generalised median graph $\overline{G}$ and the set median graph $\hat{G}$ of $S$ are defined in [11] by

$$\overline{G} = \arg \min_{G \in S} \sum_{i=1}^{n} d(G, G') \quad \text{and} \quad \hat{G} = \arg \min_{G \in S} \sum_{i=1}^{n} d(G, G')$$

(8)

Both the generalised median graph and the set median graph minimise the sum of distances to all input graphs and the only difference lies in the graph space where the median is searched for. The generalised median is the more general concept and, therefore, usually a better representation of the given patterns than the set median. Notice that $\overline{G}$ is usually not a member of $S$.

We extend the median graph concepts to the spectral theory. Let $K$ be the set of all modal matrices. Given a set of modal matrices $L = \{U^1, U^2, ..., U^n\}$, we define the generalised median eigenmode $\overline{U}$ and the set median eigenmode $\hat{U}$ of $L$ by

$$\overline{U} = \arg \max_{U \in K} \sum_{i=1}^{n} \Gamma(U, U') \quad \text{and} \quad \hat{U} = \arg \max_{U \in K} \sum_{i=1}^{n} \Gamma(U, U')$$

(9)

The generalised median eigenmode and the set median eigenmode maximise the sum of the correlations to all modal-matrices in $K$ or $L$. Nevertheless, the computation of both medians is drastically different. While $\hat{U}$ is obtained in polynomial time respect the number of elements, $\overline{U}$ is obtained in exponential time.

6 Synthesis of the Generalised Median Eigenmode

Given a set of AGs $S = \{G^1, G^2, ..., G^n\}$, which are initially supposed to belong to the same class, we do not, in general, have any way of synthesising the generalised median eigenmode $\overline{U}$ that represents the ensemble unless we can first establish a common labelling of their vertices. We would like to choose the common labelling so as to minimise the measures of dissimilarity between the given AGs and so to maximise the correlation between the median eigenmode $\overline{U}$ and the eigenmodes extracted from the adjacency matrices $L = \{U^1, U^2, ..., U^n\}$. This global optimisation problem does not lead to a computationally practical method for choosing the labelling, because there are too many possible orientations to consider, especially when the number and order of the AGs is high.

Two different methods of synthesising a median from a set of elements are used in the literature [2]; the incremental method and the hierarchical method. Therefore, two sub-optimal methods to synthesise $\overline{U}$ could be defined. In the former, $\overline{U}$ is updated by the AGs, which are sequentially introduced. The advantage of this method is that the learning and recognition processes can be interleaved, i.e. the recognition does not need to wait for all the input instances, but is available after each AG has been processed. The main drawback of this incremental approach is that different median eigenmodes can be synthesised from a set of unlabelled AGs depending on the order of presentation of the AGs. To infer some unique $\overline{U}$, a hierarchical method can be defined, which is carried out by successively merging pairs of AGs with minimal distance. The drawback here is that the full ensemble of AGs is needed to generate the median eigenmode. In the experiment results, the median spectral graphs have been obtained by the incremental synthesis.
Algorithm 1 computes the median spectral graph of a set of AGs using an incremental method. It uses three main procedures. The first is the *Eigen-Decomposition* that obtains the eigendecomposition of the adjacency matrix of an AG (section 3). The second is the matching algorithm $M$ that finds a sub-optimal labelling between spectral graphs (section 4.2). An the third procedure, *Median_Graph*, updates the median graph. We have to mention that the median function is not transitive, for this reason, the procedure has to keep the information of an AG that is the addition of all the AGs used to compute the median.

**Algorithm 1:** Incremental-synthesis-of-AGs  
**Inputs:** A sequence of AGs $S=\{G^1, G^2, \ldots, G^z\}$, over a common domain.  
**Output:** The eigendecomposition that represents the median graph of $S$.  
**Begin**  
\[
\{\mathcal{G}, \mathcal{X}\} := \text{Eigen-Decomposition}(\ G^1\ ) \{\mathcal{G} \text{ is composed by only } G^1\}\]  
for $k := 2$ to $z$ do  
\[
\text{let } f^k : G^k \rightarrow \mathcal{G} \text{ be the labelling found by } M\left(U^k, \mathcal{U}\right)
\]
\[
\mathcal{G} := \text{Median_Graph} \left(\mathcal{G}, G^k, f^k\right)
\]
\[
\{\mathcal{U}, \mathcal{X}\} := \text{Eigen-Decomposition} \left(\mathcal{G}\right)
\]
**end-algorithm**

7 Experimental Results

In order to examine the behaviour of the new representation, we performed a number of experiments with randomly generated AGs. We randomly generated 10 initial complete AGs, one for each model. From these AGs, the reference and test sets of 10 AGs each were derived by modifying the attribute value of their arcs applying a gaus-sian noise. Figure 3 left shows the ratio of recognition when the noise is increased (Standard deviation from 1 to 14) with 3 different methods: a) *eig2eig*: Comparing graphs using their eigenvalues (section 4). b) *eig2clus*: Comparing graphs to median graphs using eigenvalues (section 5). Approximate median graphs were obtained by the algorithm sketched in section 6. c): *arg2arg*: Comparing graphs using the edit operation matching algorithm described in [13]. Only the substitution edit operation has been considered since we want to find the best bijective labelling between vertices of both graphs.

![Fig. 3. Recognition ratio and run time in seconds of randomly generated attributed graphs.](image)

It is interesting to emphasize that the ratio of recognition of the spectral methods is only slightly lower than the edit operation methods although the cost of the first ones are polynomial and the second ones are exponential. Considering the spectral methods, we have to stress that in the *eig2clus* method, the run time (figure 3 right) of the rec-
ognition depends only on the number of clusters but not on the number of AGs per cluster. For this reason, in these experiments, the run time of the \textit{eig2clus} method were 10 times lower than the \textit{eig2eig} method. Moreover, when the AGs in a cluster are very different (the noise is high), the \textit{eig2clus} method keeps the structural information of the cluster and for this reason the recognition ratio obtains the best results.

8 Conclusions and Future Work

In this paper, we have merged the median computation with the spectral graph theories to define the Median Spectral Graphs. The aim of this new structure is to obtain the advantage of the structural pattern recognition but with a polynomial computational cost. By decomposing the adjacency matrices of the graphs in their eigenvectors and eigenvalues, we obtain a suboptimal labelling in polynomial cost. Experimental results show that this scheme is useful to keep the structural information of the graphs of each class when they are synthesised in only one median structure. As a future work, we have to define the hierarchical synthesis and test our new methods in a real application.

References