Spectral Median Graphs Applied to Graphical Symbol Recognition

Miquel Ferrer¹, Ernest Valveny¹, and Francesc Serratosà²

¹ Computer Vision Center, Dep. Ciències de la Computació
Universitat Autònoma de Barcelona, Bellaterra, Spain
`{mferrer, ernest}@cvc.uab.es`

² Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili, Tarragona, Spain
`francesc.serratosà@urv.cat`

Abstract. Generalized median graph is a general concept useful to capture the essential information of a set of graphs. In addition, spectral techniques can be used to obtain approximate solutions of graph matching problems in a reasonable time. In this work we use the novel concept of spectral median graph which takes advantage of both the median concept and the spectral techniques, to synthesize the representative of a set of graphical symbols. The results show that this concept represents appropriately the most important intra-class features, while rejecting small distortions and, for extension, it can be used to infer a prototype of a set of symbols.

1 Introduction

Graphs, specially labelled or attributed relational graphs, are general and powerful data structures for object representation in structural pattern recognition and computer vision applications. When objects are represented by graphs, *graph matching* is used to compare such objects. Algorithms for graph matching include graph and subgraph isomorphism [1]. However, due to errors and noise in the input data, many times it is not possible to find a perfect match between two elements and then, algorithms for approximate or error-tolerant matching must be considered. These algorithms compute a similarity measure between two given graphs. Numerous applications for exact and error-tolerant graph matching such as character recognition, schematic drawing analysis, 2D shape analysis, 3D object interpretation and machine learning have been reported in the literature [2].

In some of these applications it may be necessary to obtain the prototype of a set of objects. Given a set of noisy samples of a certain object, error-tolerant graph matching can be useful to infer a representative model that captures the essential information of the class while rejecting small distortions due to noise. In this context the concept of median graph [3] can be very useful and it has already been applied to the synthesis of a prototype of a set of graphical symbols[4].

¹ This work was sponsored research Fellowship number 401-027 (UAB) / Cicyt TIC2003-09291 (Ministerio Ciencia y Tecnología).

It is well-known that one of the drawbacks of graph matching is its computational complexity. However, in the last years, spectral graph theory have been applied to graph matching as an alternative way to obtain approximate solutions in a reasonable time [5]. We have used spectral graph theory to develop an efficient algorithm to compute the generalized median graph [6]. In this paper, we propose an application of the spectral median graph to compute the representative prototype of a set of graphical symbols. Graph matching has been widely used for recognition of graphical symbols and the definition of symbol prototypes is one of the open issues [7]. Thus, we will define a graph representation of symbols that is suitable for applying spectral techniques and we will show how the spectral median graph yields a good prototype of the set of symbols, in terms of recognition rates and theoretic definition. Results are compared with those obtained using the Graduated Assignment Algorithm [8].

The rest of the paper is organized as follows. In section 2, we present some theoretical concepts required to understand the rest of this work. Section 3 presents the concept of spectral median graph. In section 4 we introduce the representation of graphical symbols used to perform the experiments. Section 5 presents the experiments and the results obtained. We terminate with some conclusions and possible future research lines.

2 Definitions and Notation

In this work, we consider undirected labeled graphs. Given L_V and L_E , the sets of vertex and edge labels respectively, formally speaking a labeled *graph* G is a 4-tuple $G = (V, E, \mu, \beta)$ where V is the set of nodes, $E \subseteq (V \times V)$ is the finite set of edges, $\mu : V \longrightarrow L_V$ is a function which labels each vertex $v \in V$ and $\beta : E \longrightarrow L_E$ is a function which labels each edge $e \in E$.

2.1 Median Graph

Median graph [3] is an useful concept in object prototyping. Given a set of graphs, the median is defined as the graph that has the smallest sum of distances to all graphs in the set. We can distinguish between set and generalized median graph. Formally speaking, median graph can be defined as follows:

Definition 1. *Let Z be the set of graphs that can be constructed using labels from L_V and L_E . Given $S = \{G_1, G_2, \dots, G_n\} \in Z$, the generalized median graph $\bar{g}$ and set median graph $\hat{g}$ of S are defined as follows:*

$$\bar{g} = \arg \left(\min_{G \in Z} \sum_{G_i \in S} d(G, G_i) \right) \quad (1)$$

$$\hat{g} = \arg \left(\min_{G \in S} \sum_{G_i \in S} d(G, G_i) \right) \quad (2)$$

For the median graph computation of a similarity or distance measure between two given graphs, $d(G_1, G_2)$ is needed. Error-tolerant graph matching [9] can be used to find the best correspondence between two graphs, providing such a measure of similarity.

The only difference between both definitions is the space where the median is computed. The generalized median graph is computed among the set of all possible graphs. Thus, it is a more general concept and usually gives a better representation than set median graph, which is computed among the set of sample graphs. Notice that $\bar{g}$ is usually not a member of S , and in general more than one generalized and set median graph can be found for a given set.

Conceptually, while the computation of set median graph is exponential in the size of input graphs and polynomially bounded in the number of those graphs (we must compute $\frac{1}{2}n(n-1)$ pairwise graph distances), the computation of generalized median graph is exponential in both the size of input graphs and the number of those graphs. As a consequence, we are committed to use suboptimal methods to compute approximate solutions in a reasonable time.

An application of median graph used in the synthesis of graphical symbols has been presented in [4]. In this work a genetic algorithm has been employed to compute the generalized median graph. The results obtained by such algorithm have been compared with those obtained by an A^* -search optimal algorithm.

3 Spectral Median Graph

Spectral graph theory is concerned with understanding how the structural properties of graphs can be characterized using the eigenvalues and eigenvectors of the adjacency matrix of a graph. A pioneering work concerning inexact graph matching using spectral techniques is [5]. Using this work, we have extended the median graph concepts to the spectral graph theory to define set and generalized median eigenmode [6] using the modal matrices (matrices containing the eigenvectors obtained from an adjacency matrix):

Definition 2. Let K be the set of all modal matrices of order n . Given a set of modal matrices $L = \{U_1, U_2, \dots, U_m\}$, the generalized median eigenmode $\bar{U}$ and the set median eigenmode $\hat{U}$ of L are defined by:

$$\bar{U} = \arg \left(\max_{U \in K} \sum_{U_i \in L} \Gamma(U, U_i) \right) \quad (3)$$

$$\hat{U} = \arg \left(\max_{U \in L} \sum_{U_i \in L} \Gamma(U, U_i) \right) \quad (4)$$

where $\Gamma(A, B)$ is the correlation between two matrices A and B and is computed with the method described in [5].

The generalized median eigenmode and the set median eigenmode maximize 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 with respect to the number of elements, $\bar{U}$ is obtained in exponential time (see [6] for more details).

Fig. 1. Synthesis of median spectral graph

Synthesis of the Generalized Median Eigenmode: An incremental algorithm has been used in [6] to compute the generalized median eigenmode. First, we maximize the correlation between the modal matrices of two graphs in the set using the procedure explained in [5], giving an intermediate median graph. Then, the modal matrix of this intermediate median graph is used to maximize the correlation to the next graph in the set, and the process is repeated iteratively until the last graph in the set is processed, giving the final median graph. Figure 1 shows an example of such algorithm.

4 Synthesis of Graphical Symbols

In order to test the spectral median graph we have applied this method to compute the prototype of a given set of graphical symbols. We have chosen a subset of the symbols used in the *Sixth IAPR International Workshop on Graphics Recognition - GREC 2005* [10]. This subset contains 80 different symbols (classes), extracted from architectural, electric and other technical fields. Some representative symbols of such subset are shown in figure 2. Notice that all of them are composed of a set of straight lines. Each segment terminates either with a terminal point or a junction point (confluence point between two or more segments). For convenience, from now to the end of this work, we will refer to these kinds of points as TP and JP respectively.

Fig. 2. Six symbols corresponding to *GREC 2005* database

In order to prove the robustness of the prototypes against noise, 7 different levels of distortion have been introduced. Distortion is generated moving each TP or JP randomly within a circle of radius r , given as a parameter for each level, centered at original coordinates of the point. If a JP is randomly moved, all the segments connected to it are also moved. With such distortion, gaps in line segments, missing line segments and wrong line segments are not allowed. But the number of nodes of each symbol is not changed. Figure 3 shows an example of such distortions. For each class and for each distortion, we have created 100 images. Thus for each class we have 700 elements (100 for each distortion). Therefore, we have 5600 (80×700) images to perform the experiments.

Fig. 3. Original model (a) and distorted models from level 1, (b) to level 7, (h)

In order to compute the prototypes a graph-based representation of the symbols must be defined. This representation must take into account some restrictions imposed by the two approximate methods that have been used to compute the prototypes, namely the spectral median graph and the Graduated Assignment algorithm. Spectral techniques can only work with graphs with the same number of nodes. On the other hand, the adjacency matrices used in the graduated assignment algorithm must have 0-element in the main diagonal. In addition, in both cases the attributes of nodes or edges must be real positive numbers.

We have defined two different representations, namely *node-based* representation and *edge-based* representation. In both of them a symbol is represented as an undirected labeled graph, where the TPs and JPs are represented as nodes. Edges correspond to the segments connecting those points. The information associated to nodes or edges are their coordinates (x, y) . As labels can only be real numbers we have created two adjacency matrices for each symbol, one of them

containing x-coordinates and the other containing y-coordinates. In the *edge-based* representation, information associated to nodes is always 0 while edge labels are the coordinates (x, y) of the mid point of the segment. In the *node-based* representation, labels of nodes are the coordinates (x, y) of the point while labels of edges are always 1. In both cases we store a 0 when no edge exists between two nodes. The distance between two symbols will be the mean between the x and y distances. Figure 4 shows the two representations of a symbol.

Fig. 4. Two graph-based representations of a graphical symbol

5 Experiments and Results

In this paper, two methods are compared, namely spectral median graph and the graduated assignment algorithm for graph matching. While the spectral graph method has been tested using the two graph-based representations cited above, the graduated assignment has only been tested using the edge-based representation. We will refer to such combinations as S-edge, S-node (for spectral applied to both edge and node-based representations) and GA-edge (for graduated assignment). In addition, we propose three measures in order to test the accuracy and the robustness of such methods to compute the prototype of a set of a given models. We call these measures *Intra-class Median Accuracy*, *Recognition Rate* and *Median Evolution*. The two first measures are quantitative measures while the third one is more qualitative and is useful to perform a visual evaluation of the methods. In order to make the results more general, we have calculated for each class and for each level of distortion, the corresponding median using 1,5,10,30,50,70 and 100 images in the class. In the next lines, these three experiments will be explained and the results obtained for each one will be presented.

Intra-class Median Accuracy: This measure computes the similarity between the computed median graph and the true median graph. The sum of distances (SOD) of the median to all the other elements in the class is computed and compared to the SOD of all the elements in the class. According to its definition, the median graph would always have the minimum SOD. So, if we rank the median and all the other elements according to SOD, the lower is the position of the median, the better is the representation for the median. The results obtained

Fig. 5. *Experiment 1: Intra-Class Accuracy*

for all the classes are shown in figure 5. Curves show the mean accumulate position of the median over all distortion levels for a certain number of images used to compute the median.

When the median is computed with only one model, the position of the SOD of the median with respect to the rest of the elements in the class is distributed randomly. As a consequence, the accumulative frequency tends to be linear from 0 to 100. For the other cases, we can see that S-edge and GA-edge methods have a very similar behavior. In both cases, the SOD of the median is the minimum around the 20-25% of times and around 40-45% is among the fifth best positions. Results are much better for the S-node method. The accumulative frequency at the fifth position is between 80-90% depending on the number of images. Moreover, the evolution is much more abrupt than for S-edge and GA-edge. From these results we can draw three main conclusions: firstly, the *node-based* representation is more appropriate to compute the median; secondly, there is no significant differences between spectral methods and graduated assignment with respect to the correctness of the computed median; and thirdly, there is no significant improvement in the median accuracy when using more than 5 images

per class. It seems that using between 5 and 10 images per class is enough to compute the median.

Recognition Rate: In this case, one median was computed for each class. Then, all the models in the database were matched against the computed medians and classified according to the median with minimum distance. It must be noticed, that, as spectral graph matching requires the two graphs to have the same number of nodes, only 20 classes, those with the same number of nodes in their elements, have been used. Table 1 shows the mean recognition rates. On the left, recognition rates are the mean for each level of distortion while on the right, recognition rates are the mean (over all classes and distortion levels) for the median computed with 1,5,10,30,50,70 and 100 images/class.

Results show that S-edge and GA-edge methods have similar recognition rates, being S-edge slightly better than GA-edge while S-node has the best results in all cases. As expected, recognition rate decreases as distortion level increases, being this tendency more clear in the S-node results. No significant improvement of recognition rates can be seen as a function of the #images, beginning with 5 images. Thus, the behavior pattern of the methods is very similar to that detected with the *intra-class median accuracy* measure.

Median Evolution: This experiment shows the evolution of the median as a function of the distortion level and the #images used to compute the median. Since experiments 1 and 2 show that the best combination is the S-node method, this third experiment has only been performed with such method. Figure 6 shows the results obtained for a particular class (the ideal model for this class is shown in figure 2). In this figure, rows correspond to the distortion level while columns correspond to the number of images used to compute the median. We can see that the largest is the number of symbols used to compute the median graph, the bigger is the similarity between it and the ideal model. Obviously, the number of images to obtain a close representation of ideal symbol by means of median graph depends on the distortion level. For instance, we obtained a similar results in median computation using 5 symbols for a distortion level of 10 and 100 symbols for a distortion level of 7. It must be noticed that even with high degrees of distortion, using enough number of images, the median always tends to the ideal shape.

Table 1. *Experiment 2:* Recognition Rate

Recognition Rate vs. Distortion				Recognition Rate vs. #symbols			
Distortion	S-edge	GA-edge	S-node	# symbols	S-edge	GA-edge	S-node
1	96.10	93.54	99.78	1	92.02	93.22	94.50
2	93.19	92.43	98.28	5	93.35	92.62	96.62
3	93.46	90.75	97.24	10	93.17	91.87	96.45
4	92.94	92.81	96.27	30	93.35	92.71	97.03
5	92.55	92.19	96.10	50	94.68	93.52	97.12
6	93.54	93.97	94.84	70	94.27	92.14	97.23
7	92.05	93.12	93.78	100	93.01	92.72	97.35

Fig. 6. *Experiment 3: Median Evolution*

Table 2. Computation time (minutes): S-edge, GA-edge and S-node algorithms

Algorithm	# Symbols/median						
	1	5	10	30	50	70	100
S-edge	15.6	16.2	16.5	18.1	20	20.9	23
GA-edge	14.7	27.7	45.1	109.6	174.3	241.23	337.05
S-node	9.87	10.15	10.25	11.56	13.37	14.71	16.26

Computation Time: Table 2 shows the computation time required for S-edge, GA-edge and S-node to perform experiment 3 on a Intel Centrino 1.73 GHz with 512 MB of RAM. In the case where the median is computed with one symbol all the methods have similar computation times. In the other cases, GA-edge computation time is always greater than those obtained by S-edge and S-node, being this difference bigger as the number of symbols used increases. The difference in computation times in S-edge and S-node is due to the representation, but in both cases is practically constant with respect to the number of symbols.

6 Conclusions

Median graph is a useful concept to compute the prototype of a given set of symbols. In this work we adopted a spectral approach to compute the generalized median graph of a set of graphical symbols. We compared the results obtained by our method with those obtained using the graduated assignment algorithm.

We have shown the practical usefulness of the proposed approximate method to represent the concept of median graph. In particular, the experimental results

have demonstrated the ability of the solutions to represent the prototype of graphical symbols keeping the most important information in the class and rejecting small distortions due to some form of noise. Although S-edge and GA-edge have similar intra-class accuracy and recognition rate results, computation times are drastically different being significantly lower in the S-edge. We have defined two graph representations of graphical symbols, obtaining better results with the node-based representation. These results suggest that a deep study of the influence of the representation and the structure of the adjacency matrix should be done in order to characterize as well as possible the behavior of spectral techniques. In this sense it would be useful to extend this study to other matrices such as the Laplacian matrix, and compare the results with those obtained with the adjacency matrix.

References

1. Ullman, J.R.: An algorithm for subgraph isomorphism. *Journal of ACM* **23**(1) (1976) 31–42
2. Bunke, H., Messmer, B.T.: Recent advances in graph matching. *IJPRAI* **11**(1) (1997) 169–203
3. Jiang, X., Münger, A., Bunke, H.: On median graphs: Properties, algorithms, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(10) (2001) 1144–1151
4. Jiang, X., Münger, A., Bunke, H.: Synthesis of representative graphical symbols by computing generalized median graph. In Chhabra, A.K., Dori, D., eds.: *GREC*. Volume 1941 of *Lecture Notes in Computer Science.*, Springer (1999) 183–192
5. Umeyama, S.: An eigendecomposition approach to weighted graph matching problems. *IEEE Transactions on Pattern Recognition and Image Analysis* **10**(5) (1988) 695–703
6. Ferrer, M., Serratos, F., Sanfeliu, A.: Synthesis of median spectral graph. In Marques, J.S., de la Blanca, N.P., Pina, P., eds.: *IbPRIA (2)*. Volume 3523 of *Lecture Notes in Computer Science.*, Springer (2005) 139–146
7. Lladós, J., Valveny, E., Sánchez, G., Martí, E.: Symbol Recognition: Current Advances and Perspectives. In Blostein, D., Kwon, Y.B., eds.: *Graphics Recognition – Algorithms and Applications*. Volume 2390. (2002) 104–127
8. Gold, S., Rangarajan, A.: A graduated assignment algorithm for graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **18**(4) (1996) 377–388
9. Sanfeliu, A., Fu, K.: A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man and Cybernetics* **13**(3) (1983) 353–362
10. Dosch, P., Valveny, E.: Report on the second symbol recognition contest (to appear in *IncS* series). (2006)