On the Evaluation of Graph Centrality for Shape Matching

Samuel de Sousa, Nicole M. Artner, and Walter G. Kropatsch

Vienna University of Technology
Pattern Recognition and Image Processing Group
Vienna, Austria
sam,artner,krw@prip.tuwien.ac.at

Abstract. Graph centrality has been extensively applied in Social Network Analysis to model the interaction of actors and the information flow inside a graph. In this paper, we investigate the usage of graph centralities in the Shape Matching task. We create a graph-based representation of a shape and describe this graph by using different centrality measures. We build a Naive Bayes classifier whose input feature vector consists of the measurements obtained by the centralities and evaluate the different performances for each centrality.

Keywords: centrality, shape matching, graph

1 Introduction

Humans have the innate skill of recognizing objects by their appearance, shape, silhouettes, and contours. When this object recognition task is performed by machines, the shape representation is an important factor that needs to be taken into account, which might be considered as a key factor to obtain a good or bad recognition performance. We analyze the shape representation based on graph theory by abstracting pixels of an image as vertices and modeling their spatial relationship with edges.

Several implicit information can be extracted from graphs. For instance, centrality measurements of graphs or networks have been extensively explored in Social Network Analysis (SNA) [19] to understand the flow of information or to identify potential key actors inside the network. However, those measures could be also applied in a different context that may not be related to SNA and still be capable of achieving meaningful results. For instance, graph centralities could be used to represent a graph based on the distribution of centrality values over the vertices.

In this paper, we analyze the impact of using graph centralities in the modeling and description of shapes. To the best of our knowledge, there is no graph-based approach employing centrality measures for 2D shape matching. Hence, we represent a shape using a graph and calculate the following centrality measures: Degree, Betweenness, Closeness, PageRank, and Eigenvector. We divide a dataset of shapes into 8 classes and train classifiers to recognize those shapes by taking into consideration only the centralities of graphs. We add salt noise to the shapes in order to evaluate the impact that the absence of certain nodes and edge has in the recognition task. Our main contribution is a novel way of graph comparison and "matching" based on graph centralities, and the evaluation of their robustness under topological changes of the graph. Our results indicate that the closeness centrality is the most reliable centrality for the matching task under minor changes in the graph.

The remainder of this paper is organized as follows: Section 2 provides a literature review on graph centrality and graph-based shape representation. We introduce the centralities measurements and discuss our overall methodology in Section 3. Our experiments are explained in Section 4 narrowing our feature modeling and examining the results obtained by our classifiers. Finally, we present our conclusions and future work in Section 5.

2 Related Work

The concept of centrality has already been used by Bavelas in 1948 [1] to explain human behavior. It is frequently used in the analysis of different types of networks. Hence, many different measures of centrality have been proposed [4]. Borgatti et al. [4] try to give a graph-theoretic review of centrality measures, where measures are classified according to the features of their calculation. They focus on the three best-known measures of centrality: degree, closeness, and betweenness [9]. One of their findings is that there are four basic dimensions to distinguish between centrality measures: (i) the types of walks considered, (ii) the properties of walks measured, (iii) the type of nodal involvement, and (iv) the type of summarization.

Correa et al. [6] use derivatives of centrality in the visualization of social networks. The derivative of centrality informs how much a given node influences the importance of another node, even if they are not directly connected. They found out that derivatives of centrality are tool for analyzing social networks, they help to simplify the layout of complex networks and to visually measure the centralization degree of a network. Furthermore, they provide information to estimate other metrics like structural balance and uncertainty.

Mukherjee et al. [13] present an application of centrality in human action recognition. They employ centrality to create a compact codebook out of a large vocabulary of poses (bag-of-word approach). Cukierski et al. [7] use centrality to solve an open problem in the ISOMAP algorithm [12], which is a non-linear dimensionality reduction method. The ISOMAP algorithm computes geodesic distances between data points with the help of a neighborhood graph. Unfortunately, these graphs sometimes contain unwanted edges, which connect disparate regions of one or more manifolds and this leads to a distortion of the calculated geodesic distance matrix. This problem is where Cukierski et al. propose an edge-removal method based on graph betweenness centrality, which can robustly identify manifold-shorting edges.

In the literature, shock graphs are frequently employed in graph-based shape matching. Shock graphs are related to medial axis transform [3], but with a higher descriptive power. Each vertex in a shock graph is labeled by the shock type and the edges depend on the shock formation times. Siddiqi et al. [17] match 2D shapes based on directed, acyclic shock graphs. By employing a shock graph grammar the task of matching shock graphs can be reduced to matching shock trees, which can be done in polynomial time.

Sebastian et al. [15] match 2D shape outlines based on the edit distance between their corresponding shock graphs. They propose a novel framework, where they partition the shape space with the help of shock graph topology, discretize the space of deformations based of their shock graph transitions, and find the globally optimal sequence of transitions by employing a graph edit distance algorithm. Torsello et al. [18] present a geometric measure to determine the similarity between shapes calculated from the skeletons. This measure allows to distinguish between perceptually distinct shapes whose skeletons are ambiguous and to distinguish between the main skeletal structure and its ligatures.

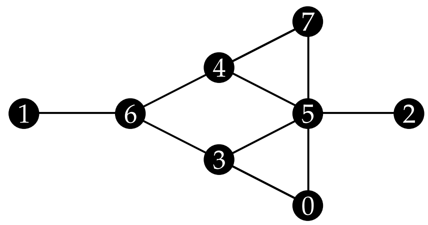
Besides the works on shock graphs and skeletons, there are approaches which match shapes by their contour. Felzenszwalb and Schwartz [8] present a so-called shape-tree, which describes the boundary of shape at multiple levels of resolution. Their representation can be used to determine the similarity between two shapes and for matching a deformable shape model to a cluttered image. In [20], Zhu et al. propose a hierarchical deformable template, which describes an object by a hierarchical graph defined by parent-child relationships. In the top vertex the pose (position, orientation, and scale) of the center of the object is stored and in the child vertices the poses of points on the object boundary are described.

The review of the related work showed that, to the best of our knowledge, there is no graph-based approach employing centrality measures for 2D shape matching. Hence, this paper is the first attempt to use centrality in graph-based shape matching.

3 Graph-based Shape Matching

Our methodology starts by building a graph-based representation of the shape and by calculating several centrality measures of this graph. The resulting centrality-based representation of the graph is further input into a classifier as the feature vector in order to distinguish different shapes by the centrality representation.

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a centrality can be interpreted as a function $f : \mathcal{V} \to \mathbb{R}$, which assigns a real value to each vertex $v \in \mathcal{V}$. In general, centrality measures the importance of a vertex within a graph or the importance of an actor in a social network [19]. We elucidate this concept using the graph displayed in Figure 1a. The measures of centrality computed for this undirected graph are listed in Figure 1b, where $b(v), c(v), d(v), e(v)$, and $r(v)$ stand for degree, closeness, between, eigenvector and rank of a vertex $v \in \mathcal{V}$, respectively. Those measures are explained in the following sections.



(a) Sample graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = 8$ and $|\mathcal{E}| = 10$.

v	d(v)	b(v)	c(v)	e(v)	r(v)
0	2	0.00	0.53	0.58	0.09
1	1	0.00	0.38	0.18	0.06
2	1	0.00	0.43	0.33	0.05
3	3	4.00	0.63	0.72	0.14
4	3	4.00	0.63	0.72	0.14
5	5	9.50	0.70	1.00	0.23
6	3	6.50	0.58	0.55	0.15
7	2	0.00	0.53	0.58	0.09

(b) Centrality values obtained for the graph.

Fig. 1: For each vertex $v \in \mathcal{V}$, we compute the importance of the vertex according to a specific centrality.

3.1 Degree Centrality

The degree of a vertex $d(v)$ is a measure that counts the number of edges incident to v [9]. By evaluating the degree centrality, one can compare the connectivity of vertices, but this measure does not tell how well-positioned a certain vertex is within the graph. In the graph of the example in Figure 1a, the highest degree is 5 and the lowest is 1.

The importance of a vertex with regard to degree centrality depends on the average degree in a graph. For instance, a degree 8 is considered high in a graph whose average degree is 2, but it is low in a graph whose average is 20. Hence, other centrality measures are capable of providing more detailed information about a vertex.

3.2 Betweenness Centrality

The communication between two non-adjacent vertices depends on the path between them. The main idea of betweenness centrality [19] is that vertices that lie on the geodesic path of many other vertices will possess great control over the information flow, due to the fact that they reside “between” others.

The betweenness centrality $b(v)$ of a vertex v is calculated as follows:

$$b(v) = \sum_{s \in V \setminus \{v\}} \sum_{t \in V \setminus \{s,v\}} \frac{\sigma_{st}(v)}{\sigma_{st}}, \tag{1}$$

where σ_{st} is the number of geodesic paths between vertices s and t. The value $\sigma_{st}(v)$ stands for the number of geodesic paths between s and t via v. The computation of the betweenness centrality requires the determination of the geodesic paths by calculating the geodesic distances to all vertices in the graph. The lower complexity for this centrality is $O(\mathcal{V}^2 \log \mathcal{V} + \mathcal{V}\mathcal{E})$ achieved by Johnson et al. [2].

The third column of Figure 1b shows the betweenness centrality calculated for the sample graph. Vertices 5 and 6 have the highest importance considering betweenness centrality. Furthermore, their degree centralities are also high. This shows that there is a certain degree of dependence between the different measures of centrality. A vertex with degree 1 will not have the highest betweenness centrality in a graph (except for a graph with only two vertices connected by an edge). Looking at the two pairs of vertices 0 and 7 and 1 and 2, one can see that even though all of them have betweenness centrality of 0, their degree centralities are different (1 and 2).

3.3 Closeness Centrality

Closeness centrality [9] is a measure that evaluates how close a certain vertex is to all other vertices in the graph. It computes the inverse of the shortest paths from a vertex to all other vertices. Let $d(s, v)$ be the shortest distance between vertices s and v, the closeness centrality $c(v)$ of a vertex v can be computed as follows:

$$c(v) = \sum_{s \in G \setminus \{v\}} \frac{1}{d(s, v)}. \tag{2}$$

In comparison to betweenness centrality, which can result in values equal to 0 (see vertices 0, 1, 2 and 7 in the example in Figure 1a), closeness centrality will only be 0 for vertices which are disconnected from the graph. In the sample graph in Figure 1a, vertex 5 obtained the highest closeness centrality ($c(v)$). However, the $c(v)$ of vertex 5 is not much higher than vertices 3 and 4 (in contrast to $b(v)$). In SNA it is known that if information is spread from the vertex with the highest $c(v)$, it will spread over the whole network in the shortest time possible [4].

3.4 Eigenvector Centrality

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the eigenvector centrality $e(v)$ corresponds to the eigenvector $X = (x_1, x_2, \ldots, x_{|\mathcal{V}|})^T$ associated with the highest eigenvalue λ of the graph adjacency matrix [14].

As proved by the Perron Frobenius theorem [10], a square matrix (such as the adjacency matrix of a graph) with positive values has a unique, largest eigenvalue with strictly, positive eigenvector entries.

3.5 PageRank Centrality

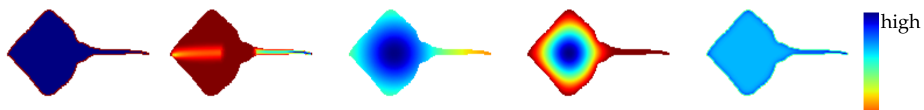
PageRank [5] is an algorithm for measuring the importance of a web page. According to [14], the page rank centrality $r(v)$ is a variant of the eigenvector centrality and it can be determined by the following equation:

$$\mathbf{R} = \frac{1-d}{n}.\mathbf{1} + d\mathbf{LR}, \tag{3}$$

where $\mathbf{R} = (r_1, r_2, \cdots, r_n)^T$ is the page rank vector, where r_i is the page rank of vertex (webpage) i and n is the total number of vertices (webpages), d is a damping factor with d = 0.85, $\mathbf{1}$ is a column vector, and $\mathbf{L}$ is a modified adjacency matrix (for details on the computation see [14]).

3.6 Centrality Shape Descriptor

We propose to describe 2D shapes by histograms of their centrality measures, which are calculated on a 8-neighborhood graph. Each centrality measure describes different properties of a shape. Figure 2 displays the same graph with different representations of centralities.



(a) Degree (b) Betweenness (c) Closeness (d) Eigenvector (e) PageRank

Fig. 2: Visualization of the centrality values of a shape. The centrality values are color coded from low to high values.

The degree centrality (Fig. 2a) shows high values in the center of the shape and low values in the boundaries. Hence, the degree histogram will simply count how many vertices exist of the two groups: (i) vertices inside the shape with 8-connected neighboring vertices, and (ii) vertices on the boundary. The betweenness centrality (Fig. 2b) shows high values for vertices that were frequently traversed by geodesic paths of the graph. Closeness representation (Fig. 2c) contains higher values for vertices that are "closer" to all other vertices and low values for "distant" ones. The eigenvector centrality (Fig. 2d) shows high values in the "center" of the shape and those values decrease towards the boundaries. Finally, PageRank (Fig. 2e) results show high values in the center of the shape and small values on the boundary. However, those values on the boundary are slightly different from the degree results, they are not as uniform as the degree centrality.

4 Experiments

The histograms of the graph centralities are used to create feature vectors for a Naive Bayes classifier. We have used the Kimia's Shape 99 database [16] which contains 11 images of 9 classes. Figure 3 displays one image of each class. Considering that once the graph is built, we do not use any image information such as boundaries, curvature. One way to evaluate the robustness of each centrality when the topology of graph changes, i.e. when the number of vertices and edges of the graph change. To achieve that we add different percentages of random

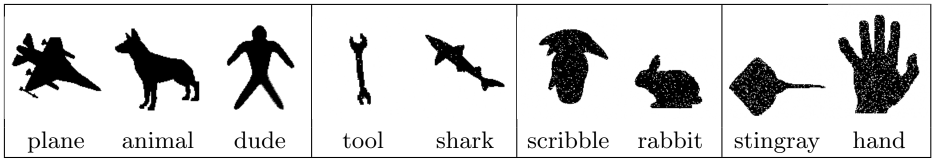
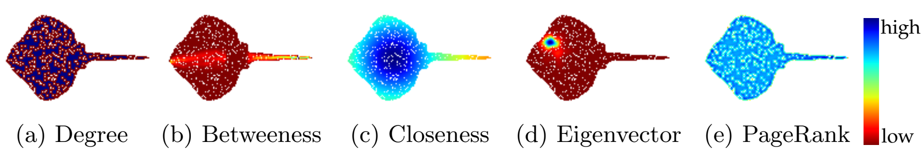


Fig. 3: Sample images of classes used for training.

noise to the images of the database. As we are working with binary images and the foreground (shape) is black, we add random "salt" noise (white pixels) which will cause missing nodes when the graph is constructed. The first three images of Figure 3 do not show any noise. The following two figures (tool and shark) show 1% of noise. Images of scribble and rabbit show 5% of noise and the final two images show 10% of noise.

We build six different classifiers using the centrality values. Five classifiers are computed using each individual histogram of a centrality measure with fixed number of bins (in our experiments bins = 40, value was chosen empirically). One extra classifier (referred from now on as "all") combines all five histograms as a feature vector of the training data.

Our code was developed in python and the centralities were calculated using the igraph library[1]. We train our six classifiers using the original dataset without any noise. We classify the original training set and corrupted versions of it with 1%, 5%, and 10% noise as previously described. The classification of the training data aims to evaluate the correctness of the classifier, It demonstrates, for instance, how well the histograms of each centrality differ from each class and thus allow correct classification of new data. Figure 4 shows the impact of 10% of noise in computation of the centrality values. Closeness centrality (Fig. 4c) shows robust results against noise whereas other centralities suffer a great impact from additive noise.



(a) Degree (b) Betweeness (c) Closeness (d) Eigenvector (e) PageRank

Fig. 4: Visualization of the centrality values of a shape under 10% of noise. The centrality values are color coded from low to high values.

The results of classification are displayed in Table 1. First column displays the classifier evaluated. Second column (no noise) displays the result of the classification on the training data, which shows that the classifier using all centralities obtained the best results (98% of correctness). Most of the individual centralities

[1] http://igraph.sourceforge.net

classifier	No noise	1% noise	5% noise	10% noise
All	**0.98**	0.85	0.65	0.46
Betweenness	0.82	0.67	0.51	0.29
Closeness	0.94	**0.93**	**0.92**	**0.83**
PageRank	0.81	0.32	0.11	0.11
Eigenvector	0.89	0.69	0.27	0.26
Degree	0.81	0.37	0.17	0.14

Table 1: Results of classification using different centrality measures. The closeness centrality achieved the best results under random noise.

were able to perform well (with at least 81% of correctness). However, when 1% of noise is added, the closeness centrality starts to obtain the best performance. The other centrality measures are more sensitive to noise than closeness and their results drop remarkably: the addition of random noise had the highest impact on the results of PageRank, Degree, and Eigenvector. We observe that closeness centrality remains stable in the under the missing nodes in the graph. Even when 10% of noise is added, it still obtains better results than other centralities on the training data.

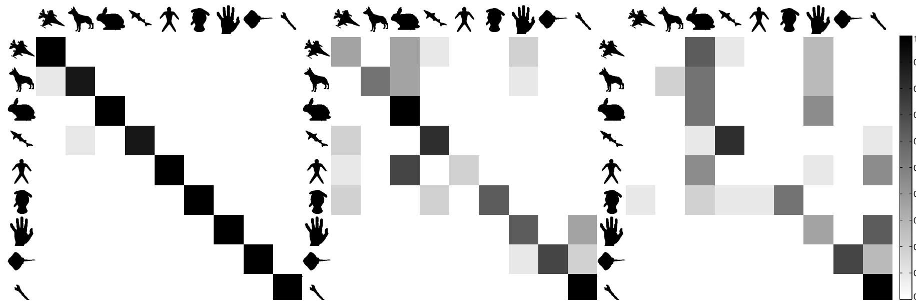
It can be concluded that closeness centrality is the most robust centrality measure for our shape matching and that the simple combination of all centralities did not obtain the best result but it was still considerably better than the other isolated centralities. Furthermore, the performance of the degree centrality changes severely as under "salt" noise due to the fact that the vertices inside the shape have degrees smaller than eight and therefore the whole histogram distribution of degree is significantly different from the training data.

Figure 5 shows the confusion matrices of the classification results for classifiers *all* and *closeness*. For instance, Figure 5d shows that the classifier of closeness centrality misclassified *airplanes*, *dogs*, *sharks*, and *dudes* in the training data. This might have occurred due to the sharp edges of the pictures leading to similar histograms. However, the per-class classification of closeness is still very high after 10% of noise.

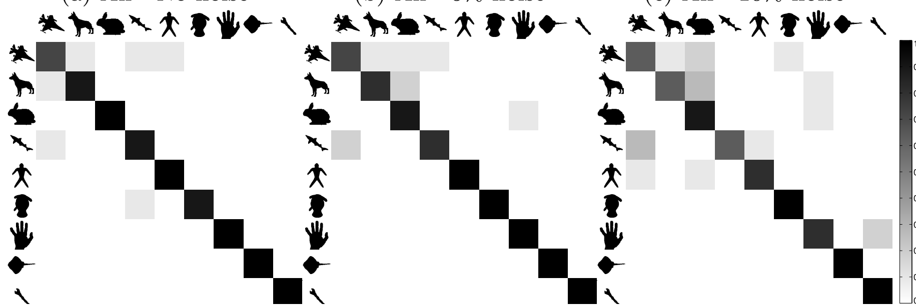
5 Conclusion

In this paper we evaluate the usage of graph centralities as descriptors in 2D shape matching. We create a representation for the shape using the histograms of centralities and we train a Naive Bayes classifier based on that representation. Our results indicate that in the presence of random noise, the closeness centrality obtained the highest classification rates compared to the other centralities.

One clear advantage of such a representation is the rotation invariance achieved by the graph representation. Scale invariance might be achieved if, for instance, histograms are normalized. However, if based solely on the histogram, important details of the shape might not be captured, *i.e.* some shapes might result in the same centrality histogram even though they belong to different



(a) All - No noise (b) All - 5% noise (c) All - 10% noise



(d) Closeness - No noise (e) Closeness - 5% noise (f) Closeness - 10% noise

Fig. 5: Confusion Matrices of classification. First row shows the combination of all histograms. Second row displays the closeness centrality. Each matrix column shows the real class and the row shows the predicted class.

classes. One solution to this problem could consist on adding the relationships between mountains and valleys of the shape in one direction (such as clockwise) and integrating this relationship in the feature modeling.

In future, we plan to investigate other representations for centralities or other combinations of multiple centralities such as using boosting. Also, integrating spatial information into the graph representation might increase the performance of results and help in the disambiguation of similar shapes as performed by [11].

Acknowledgments

Samuel de Sousa acknowledges financial support by the Austrian Agency for International Cooperation in Education & Research (OeAD) within the OeAD Sonderstipendien program, financed by the Vienna PhD School of Informatics.

References

1. A. Bavelas. A mathematical model for group structure. *Human Organization*, 7(3):1630, 1948.

2. P. E. Black. Johnson's algorithm. *Dictionary of Algorithms and Data Structures*, 2004.
3. H. Blum. Biological shape and visual science (part I). *Journal of Theoretical Biology*, 38:205–287, 1973.
4. S. P. Borgatti and M. G. Everett. A graph-theoretic perspective on centrality. *Social Networks*, 28(4):466 – 484, 2006.
5. 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, Apr. 1998.
6. C. Correa and K.-L. Ma. Visualizing social networks. In C. C. Aggarwal, editor, *Social Network Data Analytics*, pages 307–326. Springer US, 2011.
7. W. Cukierski and D. Foran. Using betweenness centrality to identify manifold shortcuts. In *Data Mining Workshops, 2008. ICDMW '08. IEEE International Conference on*, pages 949 –958, dec. 2008.
8. P. Felzenszwalb and J. Schwartz. Hierarchical matching of deformable shapes. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8, june 2007.
9. L. Freeman. Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239, 1979.
10. G. Frobenius. Über Matrizen aus nicht negativen Elementen. *Sitzungsberichte Königlich Preussichen Akademie der Wissenschaft*, pages 456–477, 1912.
11. M. Iglesias-Ham, E. García-Reyes, W. Kropatsch, and N. Artner. Convex deficiencies for human action recognition. *Journal of Intelligent & Robotic Systems*, 64:353–364, 2011.
12. V. d. S. J. B. Tenenbaum and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.
13. S. Mukherjee, S. Biswas, and D. Mukherjee. Recognizing human action at a distance in video by key poses. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(9):1228 –1241, sept. 2011.
14. K. Okamoto, W. Chen, and X.-Y. Li. Ranking of closeness centrality for large-scale social networks. In *Proceedings of the 2nd annual international workshop on Frontiers in Algorithmics*, pages 186–195, Berlin, Heidelberg, 2008. Springer.
15. T. Sebastian, P. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):550 –571, may 2004.
16. D. Sharvit, J. Chan, H. Tek, and B. B. Kimia. Symmetry-based indexing of image databases. *Journal of Visual Communication and Image Representation*, 9(4):366–380, December 1998.
17. K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35:13–32, 1999.
18. A. Torsello and E. R. Hancock. A skeletal measure of 2d shape similarity. *Computer Vision and Image Understanding*, 95(1):1 – 29, 2004.
19. S. Wasserman and K. Faust. *Social Network Analysis. Methods and Applications.* Cambridge University Pres, New York, USA, 1994.
20. L. Zhu, Y. Chen, and A. Yuille. Learning a hierarchical deformable template for rapid deformable object parsing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(6):1029 –1043, june 2010.