



ELSEVIER

Contents lists available at ScienceDirect

## Pattern Recognition

journal homepage: [www.elsevier.com/locate/pr](http://www.elsevier.com/locate/pr)

# Graph-based point drift: Graph centrality on the registration of point-sets

Samuel de Sousa\*, Walter G. Kropatsch

Vienna University of Technology, PRIP, Favoritenstraße 9/186-3, Vienna 1040, Austria

## ARTICLE INFO

### Article history:

Received 3 September 2013

Received in revised form

15 June 2014

Accepted 16 June 2014

### Keywords:

Point set registration

Graph centrality

Coherent point drift (CPD)

## ABSTRACT

The problem of point-set registration often arises in Pattern Recognition whenever one needs to match information available in images, such as feature locations, landmarks, or points representing a surface of an object. It is a challenging task and a widely explored topic in stereo vision, image alignment, medical imaging, and other fields. Many of those problems have been addressed using graph theory by taking advantage of the structural information available in graphs. In this paper, graph centralities are explored in the point-set registration problem for the first time. We propose a variant of the Coherent Point Drift (CPD) by integrating the degree, betweenness, closeness, eigenvector, and pagerank centralities. The centrality values bring topological information used during the computation of correspondence between points. We analyse the performance on several datasets and our results indicate that the registration can converge faster when the centrality is combined with the spatial information in the traditional probabilistic framework. Our novel contribution introduces the social network centralities as a good source of prior information for the registration problem and it demonstrates how one can take advantage of such information.

© 2014 Elsevier Ltd. All rights reserved.

## 1. Introduction

Point-set registration is a widely explored topic in Pattern Recognition. It is an essential piece in feature matching [1], stereo vision [2], tracking [3], medical imaging [4] and many other applications. Intuitively, the problem consists of retrieving the transformation and computing the correspondence between two or more point-sets on their own coordinate systems.

Registration approaches are classified into either rigid or non-rigid methods. Rigid registration only considers rigid transformations whilst non-rigid registration allows more involved tasks once it can account for anisotropic scaling, skews, and complex deformations (e.g. articulations, morphing). There are several surveys reviewing algorithms for both rigid and non-rigid registration techniques [5–10]. Despite the extensive number of approaches available in the literature, the two popular techniques are worth mentioning: the Iterative Closest Point (ICP) [11] and the Coherent Point Drift (CPD) [12]. They have been investigated and extended by several authors [13] and in this paper, we focus on the CPD algorithm.

One way to formulate the registration problem is via graph theory [14–20]. Graph-based approaches provide structural information in which edges describe relationships between entities (e.g. components of an object). When there is no need to recover the spatial transformation or such information is not available, the correspondence problem is reduced to the Graph Matching (GM) problem [21].

Many fields of science take advantage of graph representations. In particular, the field of Social Network Analysis (SNA) has employed the concept of centrality for many years [22–26]. The notion of centrality tries to capture the measure of importance within a network (graph). Although well-known, such a concept has not been broadly explored outside SNA. By using centralities, one can state that a certain node is more relevant than another or that it is possible to rank the nodes according to their importance. In this paper, we bring the so-called graph centralities into the registration of point-sets and use this topological information on the pursuit of the correct registration between points. We propose to integrate the centrality measures during the computation of correspondence. Thus, not only the spatial information is taken into account but also the significance of nodes according to each centrality measure.

The flavor of centrality has already been applied in computer vision before, especially when approaches emerged from the spectral graph theory [14,27–30]. Nonetheless, to the best of our

\* Corresponding author.

E-mail addresses: [sam@prip.tuwien.ac.at](mailto:sam@prip.tuwien.ac.at) (S. de Sousa), [krw@prip.tuwien.ac.at](mailto:krw@prip.tuwien.ac.at) (W.G. Kropatsch).

<http://dx.doi.org/10.1016/j.patcog.2014.06.011>

0031-3203/© 2014 Elsevier Ltd. All rights reserved.

knowledge, our approach is the first one to explicitly adopt centralities during the registration of point-sets in computer vision. The techniques which use similar concepts often explore eigenvectors of some proximity matrix encoding the location [27,28] or the orientation of points. Some centralities (e.g. eigenvector and pagerank) work directly on the eigenvectors of the adjacency matrix.

Our results indicate that when a graph centrality is combined with the spatial information of points, one can obtain a faster decay of the alignment error and a good registration can be obtained in fewer iterations when compared with the original algorithm. We have evaluated 495 datasets and discuss the results accordingly. On our best results, we were able to converge with three times less iterations than CPD. Thus, graph centralities can be used as a good source of prior information in computer vision. Also, exploiting topological information can bring us more benefits instead of solely relying on the spatial one. Thus, one could extend those ideas to other vision tasks as well. The contributions of this paper are the following:

- It introduces the centrality concept originated in the Social Sciences into the point-set registration problem.
- It shows how one can integrate topological information by using such measures instead of trusting only spatial information of unstructured point-sets. We call this variant as Graph-based Point Drift (GPD).
- We bring up the importance of edges in a graph. Many researchers neglect the role of the edges and we discuss how fundamental this aspect is for the improvement of the results.
- It evaluates the performance of five centrality variants along with the original algorithm. GPD can converge in less iterations by using the centralities.

The remainder of this paper is organized as follows: [Section 2](#) provides a review on point-set registration algorithms focusing on those whose methodologies are closely related to ours. The role of the edges is discussed in [Section 3](#). [Section 4](#) introduces the concept of graph centrality and overviews the earlier usages of centrality ideas in computer vision. Our main methodology is disclosed in [Section 5](#). We address the graph creation out of unstructured point-sets<sup>1</sup> and explain the introduction of centralities into the CPD algorithm. Our experiments are discussed in [Section 6](#) when we provide the results obtained for the registration of 2D and 3D point-sets under both rigid and non-rigid transformations. Finally, our conclusions and future work are presented in [Section 7](#).

## 2. Point-set registration approaches

Given two point-sets,  $\mathbf{X}$  and  $\mathbf{Y}$ , our task is to retrieve the transformation that best maps  $\mathbf{Y}$  onto  $\mathbf{X}$  and to estimate the correspondence between points in both sets. More specifically, there are several types of transformations that can be assumed: rigid, similarity, affine, and non-rigid. Our approach is based on the work of Myronenko and Song [12] and is also closely related to the spectral method proposed by Carcassoni and Hancock [30]. In this section, we review some approaches akin to our work.

<sup>1</sup> Points containing only spatial coordinates without any topological linkage to each other.

### 2.1. Expectation maximization methods

Many researchers decouple the registration problem into the estimation of transformation and correspondence. It is often referred to an alternate optimization scheme, i.e. the transformation between point-sets is fixed while the correspondence between points is estimated. Later, the correspondence is fixed and the transformation is obtained [12,15–17,30–32].

Cross and Hancock [15] developed a framework based on the Expectation Maximization (EM) algorithm as a dual step optimization. Previous works [33,34] estimated the affine and Euclidean parameters of point-sets and could be considered as one of the first ones to introduce structural constraints into correspondence. Authors employed the Delaunay [35] triangulation to build the graph and to constrain the correspondences matches based on the edges of the graph. They define a structure called the *superclique* as the sets of nodes connected to a central node. This structure is later used in the computation of the probability associated with the match. It is required to have a dictionary of possible mappings and this is clearly the bottleneck of their approach. The matching algorithm estimates the geometric transformations and correspondences by jointly maximizing the likelihood over the space of matches and transformations using the EM algorithm.

### 2.2. Spectral-based methods

One of the first usages of spectral methods in graph matching was conducted by Umeyama [14] who performed an eigendecomposition of the graph adjacency matrix in order to obtain the permutation matrix  $\mathbf{P}$ . Such a matrix represents the correspondence between graphs. The solution created by Scott and Higgins [27] and later refined by Shapiro and Brady [28] was also based on a proximity matrix between points ([Section 4.4](#)). Carcassoni and Hancock [30] defined a point-proximity matrix  $\mathbf{H}$  which can be constructed in many different ways. For instance, it could be based on a Gaussian weighting function similar to [27,28], or on a sigmoid or an Euclidean function. They use the eigenvectors of  $\mathbf{H}$  in order to compute the probability of assignment between points along with the EM algorithm. In their experiments, they outperformed Shapiro and Brady [28].

Jain and Zhang [36] dealt with the shape correspondence problem by embedding the shapes into the spectral domain. They encode the spatial coordinates of each point-set using a Gaussian kernel with geodesic distances to create two affinity matrices. Once they find the spectral embedding of those matrices, they perform an iterative alignment and obtain the correspondence by finding the best matching according to the  $\ell_2$  distance. In their experiments, they outperformed the previous approaches [14,28,30].

Other approaches using spectral theory include the work of Mateus et al. [32] that tackled the shape matching problem by exploiting the Laplacian matrix ([Section 4.4](#)).

### 2.3. Other methods

More Recently, Zhou and De la Torre [19,20] exploited the problem of point correspondence via graph matching. They first proposed the Factorized Graph Matching (FGM) [19] and later the Deformable Graph Matching (DGM) [20]. The latter creates an affinity matrix  $\mathbf{K}$  encoding the similarity of both nodes and edges as well as the pairwise geometry of points. They factorize the matrix  $\mathbf{K}$  into matrices that preserve local structure of each graph. This factorization decouples the structures of the graph by replacing  $\mathbf{K}$  with six smaller matrices. They alternate the optimization of the correspondence (using the path-following algorithm) and

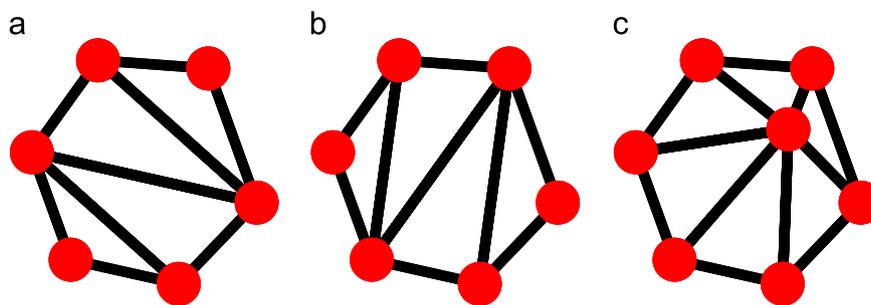


Fig. 1. Unstable configurations (a, b) and the impact of noise in the Delaunay triangulation (c).

the geometric transformation (with an optimization scheme similar to ICP).

Torresani et al. [37] also propose a dual decomposition scheme for the problem of correspondence. The scenario is now the correspondence of sparse features extracted from images and their solution minimizes an energy function based on several terms: feature appearance, compatibility of correspondence, and spatial coherence.

### 3. The role of the edges

The centrality concept tries to measure the importance of a node within a network. The importance is quantified via the existing relationships in the graph and the meaning of who is central and who is not is solely established by the edges in the graph. Thus, the edges play an essential role on the computation of centrality. Considering that we only have points embedded in a  $d$ -dimensional space, it is necessary to build the so-called data graph, i.e. a graph created based on the spatial information of points.

The data graph can be obtained in a variety of ways. The most frequent choice is the Delaunay triangulation which is based on the condition that no other point should lie inside the circumcircle of any triangle. It is the dual of the Voronoi diagram partitioning the embedding space into regions closest to the point set. We adopt the Delaunay triangulation in order to understand its impact on the results. However, we are aware that it is not an optimal strategy for our task and we are working on developing more suitable alternatives to it. Some of the problems we face when using the Delaunay triangulation are the following:

- The possibility of unstable point configurations. The same set of points might generate several different graphs (Fig. 1a and b).
- If we consider the noisy scenario, this strategy might assign many edges to a node which is actually noise. For instance, in Fig. 1c, the addition of one noisy point caused an abrupt change of topology compared with the graphs in Fig. 1a and b.

Points in computer vision often arise from measurements of an object and it would be more meaningful to use a data graph technique which could take that into account. As aforementioned, there are point configurations where the Delaunay triangulation is ambiguous: if more than three points are located on the circumcircle of a triangle, then any triangulation of the circle can be chosen as part of the Delaunay triangulation. This weak configuration gives also rise to instabilities: any perturbation of the points on the circumcircle can create another graph.

Several studies have investigated the stability of Delaunay triangulation, such as in the work of Boissonnat et al. [38]. There are several alternatives for constructing graphs such as Reeb

graphs [39], Gabriel Graph, and Minimum Spanning Tree. Outliers could be understood as those points that do not belong to the surface of an object but are present in the point-set. The location of an outlier with respect to the real data might be crucial in such a way that many edges can be assigned to it and therefore they will possess high centrality values (Fig. 1c), which is clearly not the desired behaviour. One natural solution for the outlier problem in the data graph is to fit some models to the point-set, e.g. conic fitting or a regression in order to abstract primitive structures instead of using only the Euclidean distance for pairing up nodes.

Lian and Zhang [40] propose to represent shapes using a fan-shaped structure by solving the Traveling Salesman Problem [41]. Their approach is restricted to simple polygons without holes and it results in a set of connected triangles. Later, they proposed an alternative for graph creation based on the Minimum Spanning Tree (MST) and Star Graph to be able to apply Dynamic Programming (DP) in order to find the best embedding of the point-sets [42]. Other approaches [43] focus on aesthetic aspects of point-sets. In the conclusions section, we will discuss some research directions that we are currently developing in order to address some of the previously mentioned problems.

### 4. Graph centrality

The idea of centrality was born in the Social Sciences when Bavelas [44,45] analysed group processes in human communication. It was later when Freeman [22] examined the studies of different researchers that the concept of centrality was in fact clarified. Freeman illustrates some distinct conceptual properties using a star graph<sup>2</sup> in order to explain the notion behind centrality. For instance, the central node of such graph not only possesses the highest *degree* among the others, but it also lies on the geodesic path *between* any two nodes<sup>3</sup>. Given the fact that it has the minimum distance to any node, it is considered to be *closest* to them. Those ideas compose the core of the degree, betweenness, and closeness centralities.

Concepts initially applied in social networks are being brought into other fields that are not necessarily human-related such as virtual network [46], power-grid [47], image saliency [48], and resistor network [49,50]. Nevertheless, we did not find many works in computer vision taking advantage of centralities. In this section, we review five of the most important centrality measurements: Degree, Betweenness, Closeness, Eigenvector, and

<sup>2</sup> A star graph  $S_r$  is a tree of depth one where the central node is the root of the tree.

<sup>3</sup> The geodesic of any pair of nodes  $s_i, s_k \in S_r$  is the shortest path between them. In a star graph, the shortest path between two non-adjacent nodes includes the central node.

Pagerank. For the remainder of this paper, we shall introduce the following notations:

- $\mathbf{X}, \mathbf{Y}$  are point-sets whose dimensions are  $N$  and  $M$ .
- $\mathcal{X}, \mathcal{Y}$  are nodes of the data graphs created of  $\mathbf{X}$  and  $\mathbf{Y}$  which preserve the point coordinates and contain edges describing relationship between nodes.
- $\mathbf{x}_n$  is the  $n$ th point  $\in \mathbf{X}$  or the  $n$ th node  $\in \mathcal{X}$ , for  $1 \leq n \leq N$ .
- $\mathbf{y}_m$  is the  $m$ th point  $\in \mathbf{Y}$  or the  $m$ th node  $\in \mathcal{Y}$ , for  $1 \leq m \leq M$ .
- $\mathbf{A}$  is the adjacency matrix of  $\mathcal{X}$ .
- $v(\mathbf{x}_n)$  is a function  $v: \mathcal{X} \rightarrow \mathbb{R}$  which calculates the centrality value of the given node  $\mathbf{x}_n$ .
- $b(\mathbf{x}_n), c(\mathbf{x}_n), d(\mathbf{x}_n), e(\mathbf{x}_n)$ , and  $r(\mathbf{x}_n)$  are different centrality functions that stand for the betweenness, closeness, degree, eigenvector, and pagerank centralities of the node  $\mathbf{x}_n$  respectively. In our equations, we use  $v(\mathbf{x}_n)$  as a generalization of those centrality functions.

#### 4.1. Node degree – $d(\mathbf{x}_n)$

Node degree is one of the most well known centrality measurements not only due to its simplicity but also due to its widespread usage in many graph theoretical problems. The degree of a point  $d(\mathbf{x}_n)$  measures how many nodes are connected to  $\mathbf{x}_n$ , in other words, how many edges are incident to this point. This concept is further broadened into *in-degree* and *out-degree* for a directed graph related to the edges that arrive in  $\mathbf{x}_n$  and edges that exit from it.

The degree  $\mathbf{x}_n$  can be computed as the sum of elements in the row (or column)  $n$  of  $\mathbf{A}$ :

$$d(\mathbf{x}_n) = \sum_{k=1}^N \mathbf{A}_{n,k} = \sum_{k=1}^N \mathbf{A}_{k,n}. \quad (1)$$

Nevertheless, the importance of a node with respect to the degree centrality is not really clear as it depends on the average degree of the graph. For instance, a degree 8 might be considered high in a graph whose average degree is 2, but it is low in a graph whose average degree is 30. Hence, to overcome this, one could simply normalize  $d(\mathbf{x}_n)$ :

$$d(\mathbf{x}_n) = \frac{d(\mathbf{x}_n)}{\max \{d(\mathbf{x}_k) : 1 \leq k \leq N\}}, \quad (2)$$

where the maximum degree value of the graph would now be equal to 1. Alternatively, Freeman [22] suggests that this normalization should occur with respect to the highest difference between centrality values within the graph.

In computer vision, the node degree has been explored in several manners such as in the abstraction of rectangular lattice defining a 4-neighborhood until more specialized usages such as in segmentation. However, the work of Pal et al. [48] treated the degree of a node as centrality, i.e. measuring the importance of the nodes in a graph. In their work, they aimed at extracting visual saliency of images. They were motivated by studies reporting that objects whose locations differ significantly from their surrounding are considered as salient by means of drawing attention. Thus, they decided to design networks that model such saliency (called Visual Saliency Networks) by encoding salient regions as central nodes using the degree centrality.

#### 4.2. Betweenness centrality – $b(\mathbf{x}_n)$

The communication between two non-adjacent nodes depends totally on the path between them, i.e. the nodes that lie between them [51]. The main idea of betweenness centrality defined by Freeman [52] 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  $b(\mathbf{x}_n)$  of a node  $\mathbf{x}_n$  is calculated as follows:

$$b(\mathbf{x}_n) = \sum_{s \neq n \neq t \in V} \frac{\gamma_n(\mathbf{x}_s, \mathbf{x}_t)}{\gamma(\mathbf{x}_s, \mathbf{x}_t)}, \quad (3)$$

where  $\gamma(\mathbf{x}_s, \mathbf{x}_t)$  is the number of geodesic paths between nodes  $\mathbf{x}_s$  and  $\mathbf{x}_t$  and  $\gamma_n(\mathbf{x}_s, \mathbf{x}_t)$  is the number of those geodesics passing through  $\mathbf{x}_n$ . The computation of the betweenness centrality requires the calculation of the geodesic paths to all vertices in the graph. It is important to mention that there is another measure of betweenness centrality based on random walks defined by Newman [53]. Nevertheless, in this paper, we will only refer to Freeman's betweenness.

Li et al. [54] used betweenness centrality for scene image categorization. They build a social network of images of one class (e.g. kitchen) and calculate the betweenness of a certain image of this class with respect to the network. In this way, this is not an image-to-image measure, but an image-to-class measure estimating the overall connectivity to the class to which this image belongs to. Their results were superior to the methods in the state-of-the-art on three datasets.

Mantrach et al. [55] proposed the so-called Sum-over-Path (SoP) covariance measure which is a similarity measure between nodes in a graph that considers two nodes as highly correlated if they co-occur frequently on the same (shortest) paths between nodes in the graph. Thus, they define the SoP betweenness measure whose concept estimates the expected number of times a node occurs on a path and they compare their betweenness measure with both Freeman's and Newman's. A SoP formulation for string edit distance has been proposed by García-Díez et al. [56] and the Sum-over-Forests (SoF) index was later proposed by Senelle et al. [57] with the same inspiration that large forests would occur with lower probability and short forests would occur with high probability in a graph.

#### 4.3. Closeness centrality – $c(\mathbf{x}_n)$

The closeness centrality is a measure that evaluates how close a certain node is to all the other nodes in the graph. It was initially proposed in [45,58–60]. As defined by Sabidussi [60], the closeness centrality  $c(\mathbf{x}_n)$  of a node  $\mathbf{x}_n$  can be computed as follows:

$$c(\mathbf{x}_n) = \sum_{\mathbf{x}_k \in \mathcal{X} \setminus \{\mathbf{x}_n\}} \frac{1}{d(\mathbf{x}_n, \mathbf{x}_k)}, \quad (4)$$

where  $d(\mathbf{x}_n, \mathbf{x}_k)$  is the shortest distance between vertices  $\mathbf{x}_n$  and  $\mathbf{x}_k$ . Roughly speaking, when computing the distance between two nodes, we are able to define how far those nodes are. Therefore, by inverting it, we intuitively obtain the measure of how close they are.

When analyzing shapes, de Sousa et al. [61] described a shape by the histogram of the centrality values using an 8-connected neighborhood graph. A desired behaviour would expect that similar shapes should have similar histograms with respect to their centrality values. They have evaluated the robustness of each individual centrality by randomly removing nodes of the graph and calculating the new histogram after this topological change in the graph. The results indicated that the closeness centrality was the most robust centrality against that type of noise for the shape matching task.

#### 4.4. Eigenvector centrality – $e(\mathbf{x}_n)$

The eigenvector centrality of graph  $\mathcal{X}$  corresponds to the eigenvector  $\mathbf{w} = (w_1, w_2, \dots, w_N)^T$  associated with the highest

eigenvalue  $\lambda$  of the adjacency matrix  $\mathbf{A}$  [24]. The study of eigenvalues and eigenvectors of the adjacency matrix and the Laplacian matrix of a graph has received a vast contribution from many authors [62–64] in mathematics and spectral graph theory. In computer vision, it has important results in image segmentation such as in the famous work of graph-cuts by Shi and Malik [65] and other segmentation methods exploring the Laplacian matrix [66].

The idea of *eigenvectors* of some matrix describing relationship between points was already exploited in point-set registration. In fact, one of the earlier works employing eigenvector analysis (although not in a graph) was conducted by Scott and Higgins [27] and later refined by Shapiro and Brady [28]. Both approaches apply a Singular Value Decomposition (SVD) of a matrix  $\mathbf{H}_{i,j} = \exp(-r_{ij}^2/2\sigma^2)$  which uses the squared distance ( $r_{ij}^2$ ) between nodes  $i$  and  $j$  to define the proximity of nodes. Shapiro and Brady [28] handled some limitations of the original work such as coping with large rotations and translations in the image plane. The eigenvector analysis employed by them occurred over the matrix  $\mathbf{H}$  (proximity) while the so-called eigenvector centrality occurs directly on the adjacency matrix of the graph. However, for point-set registration, it is not uncommon to define a proximity matrix between points. The recent work of Zhou and De la Torre [19,20] (described in Section 2.3) factorizes a proximity matrix for the graph matching problem.

Leordeanu and Hebert [29] propose a solution to the registration problem by exploring the principal Eigenvector of an association matrix  $\mathbf{M}$ . The interpretation of the eigenvector by the authors relates to the confidence of a certain assignment. For each correspondence element  $a = (i, i') \in \mathbf{M}$ , the eigenvector  $\mathbf{x}^*(a)$  stands for the confidence of  $a$ .

#### 4.5. Pagerank centrality – $r(\mathbf{x}_n)$

PageRank [67] is an algorithm for measuring the importance of a web page. According to [24], the Pagerank centrality  $r(\mathbf{x}_n)$  is a variant of the eigenvector centrality and it can be determined by the following equation:

$$\mathbf{w} = \frac{1-d}{N} \cdot \mathbf{1} + d\mathbf{A}\mathbf{w}, \quad (5)$$

where  $\mathbf{w} = (r(\mathbf{x}_1), r(\mathbf{x}_2), \dots, r(\mathbf{x}_n))^T$  is the PageRank vector, while  $r(\mathbf{x}_n)$  stands for the PageRank of node  $\mathbf{x}_n$  and  $N$  is the total number of nodes,  $d$  is a damping factor with  $d=0.85$ ,  $\mathbf{1}$  is a column vector, and  $\mathbf{A}$  is a modified adjacency matrix (for details on the computation see [24]).

In pattern recognition, Gomo [68] proposed an algorithm inspired on PageRank for image denoising which explores the topological structure of the similarity between image pixels.

#### 4.6. Remarks

Many other centrality measures have been proposed in the past years and are not covered in this paper. For instance, Mukherjee et al. [69] used the eccentricity<sup>4</sup> of graph as a measure of importance for modeling human action recognition. This concept could be considered as a centrality and it was also used by Kropatsch et al. [71] in the analysis and matching of binary shapes.

Another meaningful property of a graph is the Laplacian matrix which is commonly defined as  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ , where  $\mathbf{D}$  is a diagonal matrix with the degree of the nodes and  $\mathbf{A}$  is the adjacency matrix. This Laplacian matrix has important results in spectral graph theory and it is closely related to the commute time [72]. In fact,

it can be computed using the pseudoinverse of the Laplacian matrix of the graph. The commute time is a distance measure between nodes in a graph using the idea of average number of steps a random walker would take to reach a node  $j$  starting from a node  $i$  and to return back to the origin. It has a relation to the resistance distance proposed by Klein and Randi [49] in their study of resistive electrical network. This concept has been explored in computer vision by Qiu and Hancock [66,73] on the problems of image segmentation and clustering.

## 5. Coherent point drift (CPD) framework

The coherent point drift (CPD) addresses the point-set registration problem with an iterative approach that aligns one point-set towards the other until convergence. It considers the points in one set (e.g.  $\mathcal{Y}$ ) as Gaussian Mixture Model (GMM) centroids in which the correspondence and the transformation are estimated via Expectation Maximization. For instance, points in  $\mathcal{Y}$  are considered as GMM centroids with the following probability density function:

$$p(\mathbf{x}) = \sum_{m=1}^{M+1} P(m)p(\mathbf{x}|m), \quad (6)$$

where  $P(m) = 1/M$  is a uniform probability for the GMM centroids and the authors also introduce  $p(x|M+1) = 1/N$  to account for outliers. Expanding Eq. (6) with a linear combination using parameter  $0 \leq w \leq 1$ , they end up in the form

$$p(\mathbf{x}) = w\frac{1}{N} + (1-w) \sum_{m=1}^M P(m)p(\mathbf{x}|m). \quad (7)$$

During the computation of probabilities  $p(\mathbf{x}|m)$  and  $P(m|\mathbf{x}_n)$ , CPD considers the spatial coordinates associated with the points. In this work, we introduce the centrality values along with the spatial clues. Finally, the original CPD probability function is defined as

$$P_c(m|\mathbf{x}_n) = \frac{\exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_m, \theta)}{\sigma} \right\|^2\right)}{\sum_{k=1}^M \exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_k, \theta)}{\sigma} \right\|^2\right) + c} \quad (8)$$

where  $c = (2\pi\sigma^2)^{D/2}(w/(1-w))M/N$  and  $\theta$  and  $\sigma$  are the parameters estimated by EM. We use the subscript  $c$  in  $P_c(m|\mathbf{x}_n)$  to differentiate from our proposed probability estimation in the next section.

### 5.1. Centrality-based probability estimation

Given a point-set  $\mathbf{X}$  and its associated graph  $\mathcal{X}$ , we shall refer to  $\mathbf{x}$  as a point belonging to both  $\mathbf{X}$  and  $\mathcal{X}$ , hence, assuming a bijection  $\mathcal{X} \leftrightarrow \mathbf{X}$ . A straightforward way to bring the aforementioned centralities into the estimation of correspondences would consist of computing the probability  $P(m|\mathbf{x}_n)$  associated with the centrality values of nodes. So, the probability of  $\mathbf{y}_m$  being matched to  $\mathbf{x}_n$  only according to the centrality could be defined as

$$P(m|\mathbf{x}_n) = \frac{h_m^c \exp\left(-\frac{\|v(\mathbf{y}_m) - v(\mathbf{x}_n)\|^2}{2\phi^2}\right)}{\sum_{k=1}^M h_k^c \exp\left(-\frac{\|v(\mathbf{y}_k) - v(\mathbf{x}_n)\|^2}{2\phi^2}\right)}, \quad (9)$$

in which  $h_m^c$  stands for the histogram bin associated with the centrality  $v(\mathbf{y}_m)$  of node  $\mathbf{y}_m$  indicating how many nodes were found in that graph whose centrality is the same as  $\mathbf{y}_m$ . Intuitively, the nodes  $\mathbf{x} \in \mathcal{X}$  whose centrality values are closer to the centrality of  $\mathbf{y}_m$  would have a higher probability of being matched. In fact, our Eq. (9) is very similar to the one proposed by Carcassoni and Hancock [30, Eq. 16]. However, they employ the eigenvectors of a proximity matrix  $\mathbf{H}$  with different weighting functions. If we apply the eigenvector centrality as  $v(\mathbf{x}_n)$  for our Eq. (9), we are

<sup>4</sup> The eccentricity is the maximum geodesic between a node to all the other nodes in a graph [70].

computing the difference between the elements of the eigenvector of the adjacency matrix, not of a proximity matrix created with a specific weighting function.

Fig. 2a shows the centrality values obtained when calculating the closeness centrality  $c(\mathbf{x}_n)$  of  $\mathcal{X}$ . Blue nodes represent low centrality values while pink nodes represent more central ones. In order to illustrate the idea, we estimate the probability of a node being matched within the graph ( $\mathcal{X} = \mathcal{Y}$ ). Fig. 2b indicates the node  $\mathbf{x}_n$  which we would like to match according to Eq. (9). Fig. 2c shows the probability of each node  $\mathbf{y}_m \in \mathcal{Y}$  being matched to  $\mathbf{x}_n$ . We can see that many nodes are assigned the colour blue, meaning that they have lower chance of being matched to  $\mathbf{x}_n$  due to the fact that their centrality value differs significantly. The intuition of Eq. (9) is to combine a Gaussian distribution with the one estimated by the histogram to enhance the probability of nodes whose values are closer to the one we would like to match. Fig. 3a shows the histogram of the centrality distribution in Fig. 2a while the final probability is displayed in Fig. 3b. The colour scheme displays the importance of the node according to the closeness centrality. The unimodal Gaussian is centered at the centrality value of  $\mathbf{x}_n$  and  $\phi$  is the variance of the centralities in  $\mathcal{X}$  which controls how strong the influence of the Gaussian is.

It is clear that Eq. (9) by itself would not be able to successfully register both sets without integrating spatial information. There are many nodes with similar centralities which turns the matching process into a difficult task if we rely only on the centrality

information. However, having such information helps us prune nodes that do not represent good matches as in a role of a prior. Finally, we propose to combine both centrality information along with spatial coordinates. We define the spatial contribution as  $S(\mathbf{x}, m) = \|\mathbf{x} - \mathbf{y}_m\|^2 / 2\sigma^2$  and the centrality contribution as  $C(\mathbf{x}, m) = \|v(\mathbf{x}) - v(\mathbf{y}_m)\|^2 / 2\phi^2$ , where  $\sigma^2$  is the variance associated with the locations of the points and  $\phi^2$  the variance associated with the centrality values. We combine both terms in Eq. (10) and in this approach one could add a free parameter ( $\kappa$ ) to control the importance of each term via linear combination. In our current configuration, however, both terms contribute equally to the results. We discuss more about the impact of a free parameter  $\kappa$  in the experiments:

$$\psi(\mathbf{x}, m) = -(S(\mathbf{x}, m) + C(\mathbf{x}, m)), \quad (10)$$

The probability  $p(\mathbf{x}|m)$  takes the form of

$$p(\mathbf{x}|m) = \frac{h_m^c \exp(\psi(\mathbf{x}, m))}{(2\pi\sigma^2)^{D/2}}, \quad (11)$$

After some operations, Eq. (12) summarizes the probability function which considers both the spatial information of points and how central they are in the graph:

$$P_g(m|\mathbf{x}_n) = \frac{h_m^c \exp(\psi(\mathbf{x}_n, m))}{\sum_{k=1}^M h_k^c \exp(\psi(\mathbf{x}_n, k)) + c} \quad (12)$$

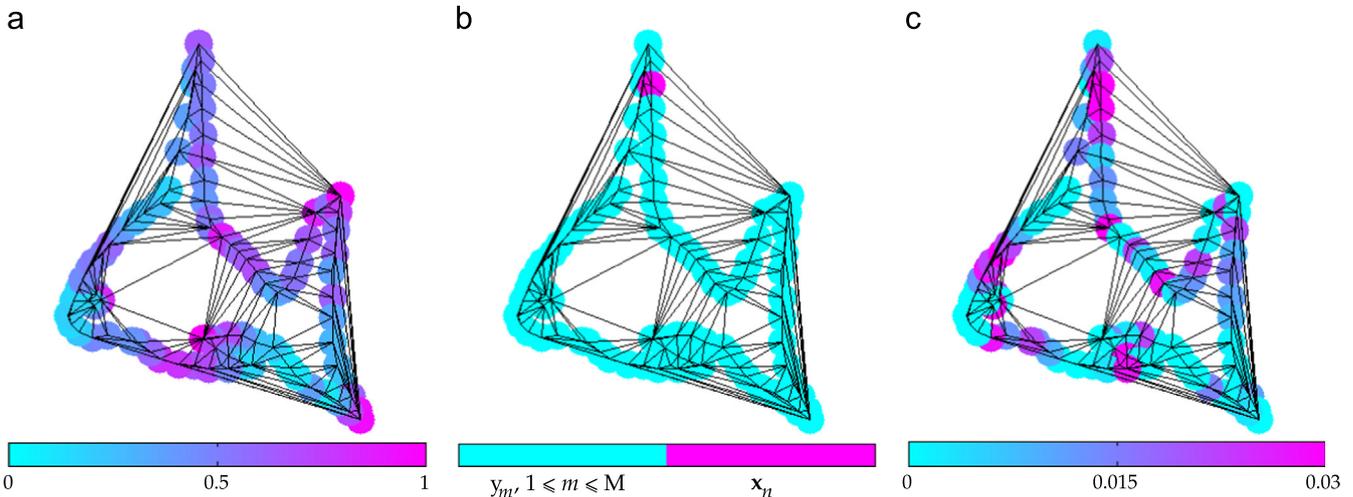


Fig. 2. (a) Graph  $\mathcal{X}$  built using the Delaunay triangulation colour-coded using the closeness centrality. The pinkish the node is, the more central it is. (b) Node  $\mathbf{x}_n$  chosen for calculating the probability of being matched with respect to the centrality of the others. (c) Probability  $P_g(m|\mathbf{x}_n)$  of  $\mathbf{x}_n$  being matched as in Eq. (9). Finally,  $\sum_{n=1}^N P_g(m|\mathbf{x}_n) = 1$ . (For interpretation of the references to colour in this figure caption, the reader is referred to the web version of this paper.)

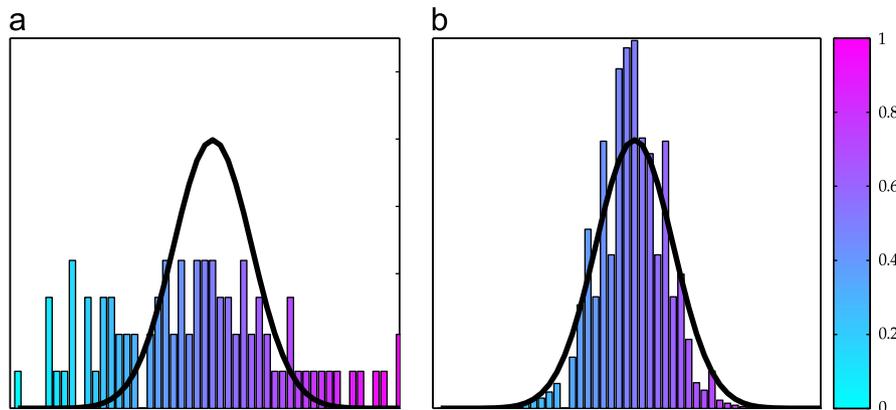


Fig. 3. (a) The histogram of the closeness centrality of Fig. 2a overlapped with a Gaussian distribution centered at  $\mathbf{x}_n$ . (b) The probability  $P(m|\mathbf{x}_n)$  as computed in Eq. (9).

where  $c = (2\pi\sigma^2)^{D/2}(w/(1-w))M/N$  (as in Eq. (8)) and each centrality  $v(\mathbf{x}_n)$  can be applied in order to compute the probabilities. We evaluate the performance of them in the experiments. We use the subscript  $g$  in  $P_g(m|\mathbf{x}_n)$  to differentiate with the probability  $P_c(m|\mathbf{x}_n)$  originally estimated by CPD.

Our Eq. (12) states that the centrality value introduced will either reward or penalize the connection between  $\mathbf{y}_m$  and  $\mathbf{x}_n$  based on the centrality.

## 5.2. Relation between GPD and CPD

In order to better understand the relation between our centrality-based probability estimation and the original CPD computation, we have the following proposition:

**Proposition 1.** Let  $\mathcal{X}^*$ ,  $\bar{\mathcal{X}}$  be graphs associated with a point-set  $\mathbf{X}$  and  $\mathcal{Y}^*$ ,  $\bar{\mathcal{Y}}$  be graphs associated with a point-set  $\mathbf{Y}$ .  $\mathcal{X}^*$  and  $\mathcal{Y}^*$  are complete graphs ( $K_t$ ) with  $t$  nodes. Similarly,  $\bar{\mathcal{X}}$ ,  $\bar{\mathcal{Y}}$  are disconnected graphs ( $\bar{K}_t$ ) with  $t$  nodes and 0 edges. Then, the probabilities computed in Eq. (12) are the same for all graphs:

$$P_g(m|\mathbf{x}_n) = P_g(\bar{m}|\bar{\mathbf{x}}_n) = P_c(m|\mathbf{x}_n) \quad (13)$$

where  $\mathbf{y}_m \in \mathcal{Y}^*$ ,  $\mathbf{x}_n \in \mathcal{X}^*$ ,  $\bar{\mathbf{y}}_m \in \bar{\mathcal{Y}}$ ,  $\bar{\mathbf{x}}_n \in \bar{\mathcal{X}}$ , and finally  $\mathbf{y}_m \in \mathbf{Y}$ ,  $\mathbf{x}_n \in \mathbf{X}$ .

Proposition 1 states that when using a  $K_t$  or  $\bar{K}_t$ , the GPD probability of  $m$  given  $\mathbf{x}_n$  are the same for the complete graph, for the disconnected graph, and for CPD where no graph approach is used. In order to prove Proposition 1, we need to show that for a complete graph  $K_t$  of  $t$  nodes, all nodes have the same centrality value, and therefore we state Proposition 2.

**Proposition 2.** Let  $K_t$  be a complete graph with  $t$  nodes and  $t(t-1)/2$  edges. Then,  $v(\mathbf{x}_k) = v(\mathbf{x}_i) \forall \{\mathbf{x}_k, \mathbf{x}_i\} \in K_t$ .

We will only provide the proof for the eigenvector centrality inspired by Brouwer and Haemers [74]. Therefore, Proposition 1 holds true for any centrality measurement that preserves the same value for nodes when using a  $K_t$  graph.

**Proof of Proposition 2** (based on [74]). Let  $\mathbf{J}$  be a  $\mathbf{1}$  matrix of order  $t$ , the rank of  $\mathbf{J}$  is 1 with spectrum  $t^1, 0^{t-1}$ . The adjacency matrix  $\mathbf{A}$  of  $K_t$  is  $\mathbf{A} = \mathbf{J} - \mathbf{I}$ , where  $\mathbf{I}$  is the identity matrix. Considering that the spectrum of  $\mathbf{I}$  is  $1^n$ , the spectrum of  $K_t$  is  $(n-1)^1$  and  $(-1)^{n-1}$ . In order to fulfill  $\mathbf{A}\mathbf{w} = \lambda\mathbf{w}$  for eigenvalue  $\lambda = n-1$ , the eigenvector  $\mathbf{w}$  associated with it needs to be a  $\mathbf{1}$ -vector. Therefore, for a  $K_t$  graph, the centrality  $e(\mathbf{x}_p) = 1$ , for  $1 \leq p \leq t$ .  $\square$

Considering all centrality types that respect Proposition 2, we can now prove Proposition 1.

**Proof of Proposition 1.** Due to the fact that  $\mathcal{X}^*$  and  $\mathcal{Y}^*$  are complete graphs, then

$$v(\mathbf{x}_k) = v(\mathbf{x}_i), \quad \text{for all } \{\mathbf{x}_k, \mathbf{x}_i\} \in \mathcal{X}^* \quad (14)$$

and

$$h_i^c = t, \quad \text{for any } i \in \mathcal{X}^* \quad (15)$$

without loss of generality, the same holds for  $\mathcal{Y}^*$ . Hence, by plugging those previous values into Eq. (12),

$$P_g(m|\mathbf{x}_n) = \frac{t \exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_m, \theta)}{\sigma} \right\|^2\right) \exp(0)}{\sum_{k=1}^M t \exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_k, \theta)}{\sigma} \right\|^2\right) \exp(0) + c} \quad (16)$$

and we arrive that

$$P_g(m|\mathbf{x}_n) = \frac{\exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_m, \theta)}{\sigma} \right\|^2\right)}{\sum_{k=1}^M \exp\left(-\frac{1}{2} \left\| \frac{\mathbf{x}_n - \mathcal{T}(\mathbf{y}_k, \theta)}{\sigma} \right\|^2\right) + c} \quad (17)$$

showing that Eq. (17) is equivalent to Eq. (8).

Graphs  $\bar{\mathcal{X}}$  and  $\bar{\mathcal{Y}}$  are disconnected graphs. Thus, the content inside the exponential function would automatically be zero<sup>5</sup> and the same results would be obtained.  $\square$

Proposition 1 indicates that neither a completely disconnected graph  $\bar{K}_t$  (just a point-set) nor a complete graph  $K_t$  improves CPD. Thus, any improvement in the actual algorithm only occurs when the graph is neither empty nor complete but when the edges play the important role of resembling the actual structure of the object being matched. We brought the importance of edges in Section 3.

## 5.3. Optimization

CPD reparametrizes the GMM centroids using parameters  $\theta$  and  $\sigma^2$  which are estimated by minimizing the negative log-likelihood function:

$$E(\theta, \sigma^2) = - \sum_{n=1}^M \log \sum_{m=1}^M P(m)p(\mathbf{x}|m) \quad (18)$$

and they solve the transformation  $\mathcal{T}$  for the rigid, affine, and non-rigid cases separately. For details on the final estimation of those parameters, refer to [12].

## 6. Experiments

In order to understand the impact of a graph centrality on the registration of point-sets, we need to perform different types of experiments. We will split the experiments in two parts. On the first part, we try to observe the behaviour of centralities on three distinct scenarios: (i) similarity transformation with no noise, (ii) affine transformation with missing points, and (iii) non-rigid transformation with the addition of noise and missing points. On the second part, we evaluate 495 point-sets with different characteristics: the sampling function, the distributions applied for drawing points, etc. The aim of the second part is to isolate certain properties of the point-sets which could help one to understand the behaviour of the individual centrality. Therefore, to some extent, one could predict which centrality is advantageous giving some knowledge about the point-set.

*Similarity transformation:* Fig. 5 shows the similarity registration of the fish<sup>6</sup> dataset with 91 points. The first row displays the alignment at the 3rd iteration, second row at the 6th, and the third row at the 9th iteration. Closeness and eigenvector variants obtained good results at the 6th iteration, which are better than CPD at the 9th iteration. Closeness converges at the 9th iteration while CPD still needs more iterations to finish the registration. We consider the alignment error as the  $\ell_2$  norm between the original point set  $\mathbf{X}$  and the point set  $\mathbf{Y}$  after the recovered transformation:  $\|\mathbf{X} - \mathcal{T}(\mathbf{Y})\|$ . Fig. 4a shows the alignment error with respect to the number of iterations. The faster the curve decay, the faster the registration. The curve should approach zero in the ideal case. Observing the curves, the GPD variants using closeness and eigenvector centralities obtained the quickest decay, while the degree, betweenness, and pagerank decayed slowly compared with them, although they are still faster than CPD for this example. This indicates that topological information might help the algorithm converge faster. We should investigate this behaviour in a bigger amount of data for the second part of the experiment.

<sup>5</sup> Assuming that the centrality of disconnected nodes is either zero or a constant but not infinity, e.g. Freeman [22] considers the distance between disconnected nodes as infinity, hence, we could consider that the closeness  $c(\mathbf{x}_n) = \sum 1/\infty \rightarrow 0$ . Freeman [52] also defines the betweenness of disconnected nodes as 0.

<sup>6</sup> <http://sites.google.com/site/myronenko/research/cpd>

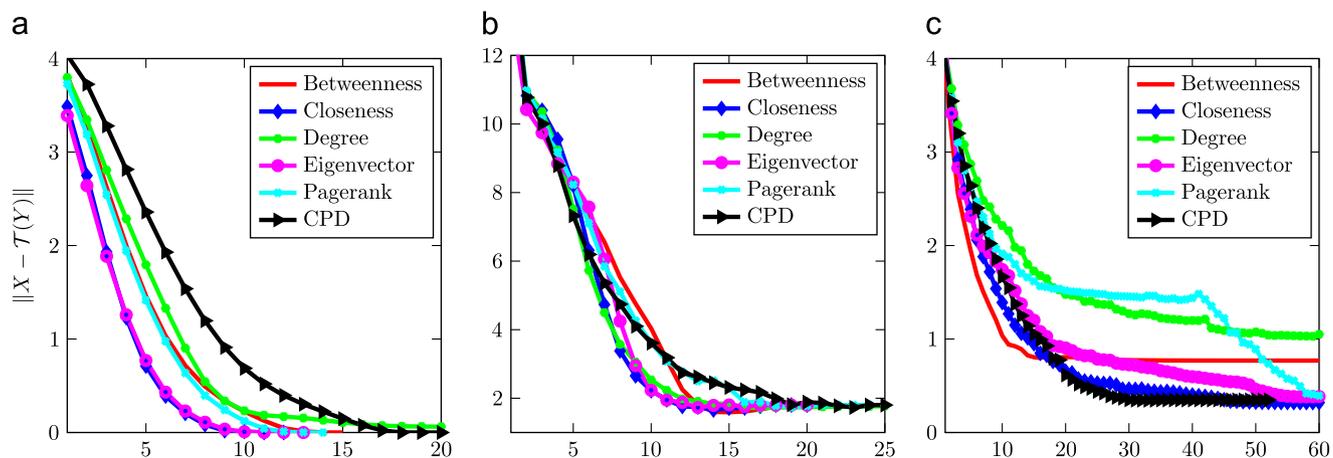


Fig. 4. Landmark errors of the (a) 2D fish dataset under similarity transformation, the (b) 3D face dataset with missing points under affine transformation and the (c) fish dataset with missing points and added noise.

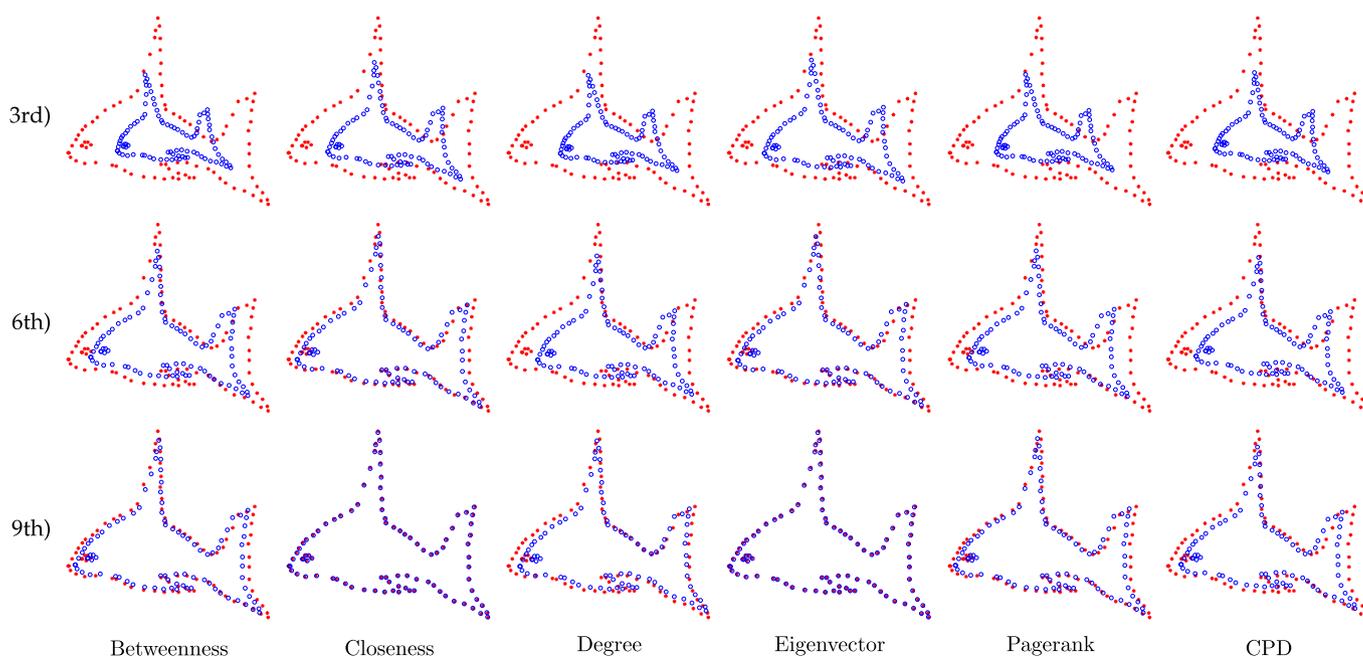


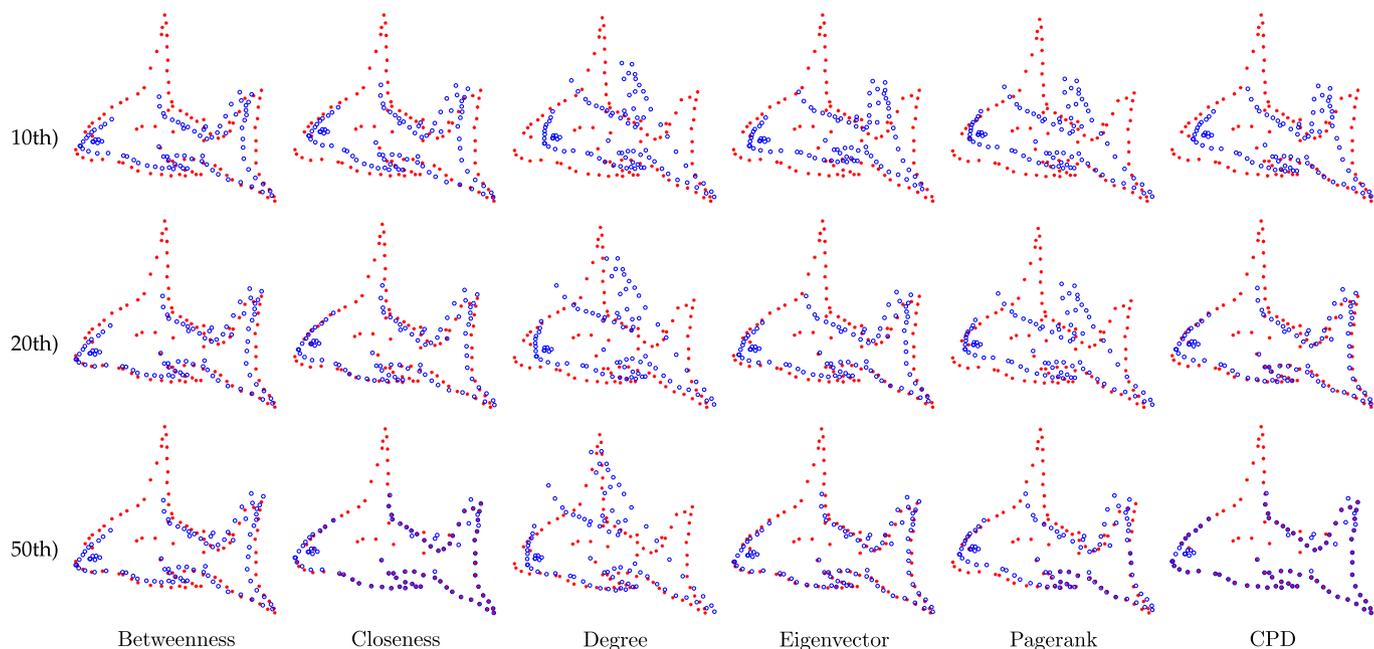
Fig. 5. Registration of the fish point-set under similarity transformation. The first row shows results at the 3rd iteration, second row at 6th iteration, and third row at 9th iteration. We can observe that closeness and eigenvector centralities converged faster than CPD.

*Affine transformation:* The second scenario of the first part considered a 3D face dataset with 392 points under affine transformation with different missing points (for each set). It is also possible to notice, in Fig. 4b, that the alignment error of some centralities decayed faster than CPD. At iteration 10, many centralities had lower error than CPD, while CPD reaches this error around iteration 20, showing that for this example, centralities are still able to decrease the error faster. This is a more challenging example than the fish one not only due to the higher number of dimensions but also due to the missing points.

*Non-rigid transformation:* Finally, the last scenario (Fig. 6) consists of the fish dataset under non-rigid registration. In this scenario, we also add outliers and remove real points from both datasets. This type of transformation is particularly challenging for all algorithms and the decay is presented in Fig. 4c. Among all centrality algorithms, the closeness centrality was the one with the best performance in terms of error decay and accuracy of the alignment. The degree centrality failed to converge and incorrectly

matched the dorsal fin of one fish to the caudal fin of the other. Such deformation would not be possible in the rigid case, but it is allowed in the non-rigid case. Nevertheless, the cause for failure of the degree centrality is due to the fact that the prior given provides wrong information. Centralities are based on graph and the way graph is built plays an essential role for the success of the registration. As already mentioned (Fig. 1c), when the Delaunay triangulation assigns many edges to a noisy point, this node becomes more central and, therefore, jeopardizes the registration if it is, in fact, noise.

*Discussion:* Those three distinct examples provide some intuition for the behaviour of centralities. The success and quality of the registration is associated with several aspects: (i) the type of centrality used, (ii) the data graph construction, and (iii) the peculiarity of each point-set. In this paper we only apply the Delaunay triangulation. Hence, we should expect that sets with noisy and missing points will not have significant better performance, or even worse, as the prior might be wrong due to the



**Fig. 6.** Registration of two 2D point sets under non-rigid transformation with outliers and missing points. First row shows results at the 10th iteration, second row at 20th iteration, and third row at 50th iteration. It is still possible to observe that closeness centrality has similar results as CPD, while the degree completely fails to converge, betweenness and page rank obtained high error.

triangulation, it can cause failure in the convergence. However, there are still some questions which can be raised and are essential to understand the remaining results of the registration:

1. What is the data graph construction technique which can best cope with noise in point-sets?
2. Is the noise in the triangulation process the only reason for the poor performance of the centrality in the third example?
3. If no noise is present, is the centrality always able to converge faster than CPD for any point-set?
4. What are the properties of point-sets which can be successfully addressed by the centralities?

The first question could be considered as an open research question. Although there are many options for data graph construction<sup>7</sup>, there is still no clear answer to such question due to the fact that the concept of noise in a point-set is context dependent. We referred to noise in the fish example because we knew that true points were sampled from an object's surface and they should lie on the silhouette. If no information about the point-set is given and the algorithm is asked to create a graph which is robust to noise, it will be a challenging task to decide what noise actually is in order to be able to reduce its impact. We will be moving towards that question on our future work: characterizing noise in point-sets and minimizing its impact on the data graph.

In order to answer the remaining questions, we perform extensive experiments in the noise free scenario once we already know that noise will not improve the results of the centralities. We fixed two aspects of the process: the data graph construction (Delaunay) and the noise scenario. Therefore, we can vary the class of point-sets and evaluate which properties cause impact on the results. We provide five different classes of point-sets that share similar properties. For each class, there are 99 distinct point-sets and we aim to evaluate whether the registration behaves similarly for each class and algorithms used, considering that those other

two parameters are fixed (noise and data graph). Fig. 7 shows samples of those five classes: (i) Clusters, (ii) Random, (iii) Gaussian sampling, (iv) Contour, and (v) Grid sampling. The images of the Kimia-99 dataset [75] were used to generate the contour and the different sampling images. The databases, our results, and the video demonstrating the registration process are available online.<sup>8</sup>

Table 1 shows the number of iterations each algorithm took to converge on those 495 point-sets. It contains the average and standard deviation per class and algorithm. The results are calculated only on the registrations that succeeded to converge. The convergence of each algorithm is displayed in Table 2. On the first class, which are clusters, all centralities obtained a faster decay than CPD. In fact, the closeness centrality was on average 2.66 times faster than CPD. Among the 99 sets of this class, the best result was approximately 3 times faster than CPD. We measure the speed of registration by the number of iterations the algorithm took to converge. When analyzing the standard deviation for both degree and eigenvector centralities for this class, it is noticeable that they are really high compared with the other algorithms. Also, degree and eigenvector were the only ones not able to converge on all databases of this class.

For the random class, all centralities obtained faster convergence than CPD, but the degree was this time the only algorithm which did not converge on all sets. Eigenvector and closeness centralities were the fastest ones, more than 2 times faster than CPD on average. The eigenvector algorithm was quite stable for this sort of data once its standard deviation is quite low compared with the previous class. The next class consists of an object contour. We extracted the contour of images from the Kimia dataset in order to create the contour-based point-set. Closeness centrality was again the fastest algorithm to converge. The degree and the eigenvector were unable to totally converge and this time the average convergence time for the degree is higher than CPD.

Finally, we sample the object using a grid-based strategy. Notice that inside the object, all nodes will have the same node degree, while they will differ on the boundaries. This was the only

<sup>7</sup> Possibilities include the Gabriel Graph, Travel Salesman Problem formulation, Euclidean Minimum Spanning Tree.

<sup>8</sup> <http://prip.tuwien.ac.at/gpd>

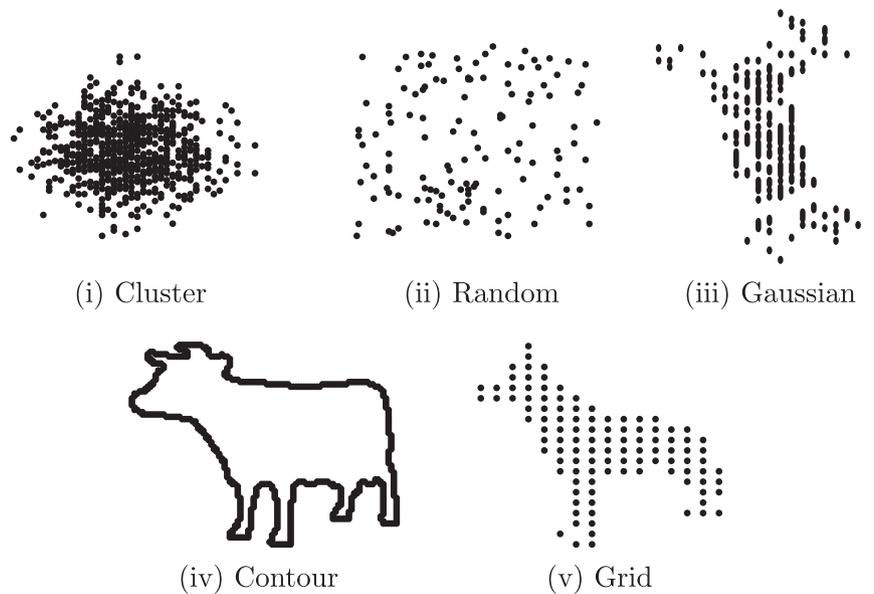


Fig. 7. Sample point-sets of 495 database used. Each image represents one sample of each class.

**Table 1**  
Convergence speed for registering the 495 datasets. Each cell shows the average and standard deviation for the number of iterations until the algorithm converged. The fastest algorithm in average is highlighted in bold.

Database	<b>B</b> ( $\mu \pm \sigma$ )	<b>C</b> ( $\mu \pm \sigma$ )	<b>D</b> ( $\mu \pm \sigma$ )	<b>E</b> ( $\mu \pm \sigma$ )	<b>P</b> ( $\mu \pm \sigma$ )	<b>CPD</b> ( $\mu \pm \sigma$ )
Cluster	42.72 $\pm$ 4.36	<b>22.72 <math>\pm</math> 1.26</b>	53.67 $\pm$ 19.89	41.95 $\pm$ 17.15	49.84 $\pm$ 7.16	60.56 $\pm$ 3.91
Random	19.94 $\pm$ 1.94	15.20 $\pm$ 1.21	30.46 $\pm$ 15.97	<b>14.56 <math>\pm</math> 0.96</b>	21.71 $\pm$ 1.89	37.25 $\pm$ 3.90
Gaussian	36.72 $\pm$ 9.33	<b>22.08 <math>\pm</math> 3.37</b>	50.45 $\pm$ 20.47	37.20 $\pm$ 18.22	45.98 $\pm$ 13.42	51.54 $\pm$ 13.82
Contour	23.82 $\pm$ 3.74	<b>17.68 <math>\pm</math> 3.66</b>	28.49 $\pm$ 5.03	20.49 $\pm$ 8.99	22.78 $\pm$ 3.99	26.39 $\pm$ 4.27
Grid	32.60 $\pm$ 18.39	26.31 $\pm$ 17.28	31.23 $\pm$ 21.08	<b>24.71 <math>\pm</math> 13.41</b>	47.12 $\pm$ 29.72	34.29 $\pm$ 13.05

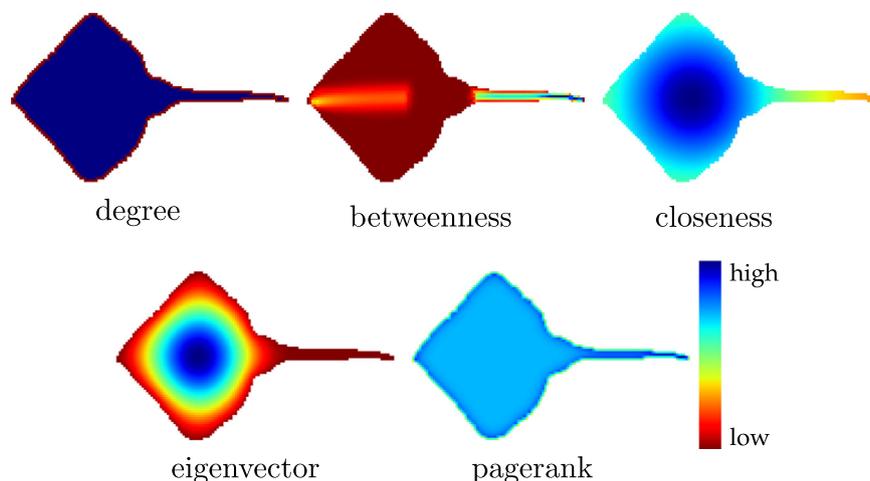
**Table 2**  
Convergence on 495 point-sets under similarity transformation. Centralities **D** and **E** did not converge on all classes. On grid class, no centrality converged 100%.

Dataset	<b>B</b> (%)	<b>C</b> (%)	<b>D</b> (%)	<b>E</b> (%)	<b>P</b> (%)	<b>CPD</b> (%)
Clusters	100	100	81	86	100	100
Random	100	100	55	100	100	100
Gaussian	100	100	75	86	100	100
Contour	100	100	90	95	100	100
Grid	95	69	35	77	68	100

case in which no centrality was able to converge 100%. If one, for instance, creates the data graph using an 8-connected neighborhood, all points inside the object are likely to be matched, based on only the degree. Therefore, when the point-set is grid sampled, it is better to rely on CPD which considers only the spatial information once the prior becomes non-informative. It is important to mention that one could add a free parameter  $\kappa$  (e.g.  $\kappa S + (1 - \kappa)C$ ) into Eq. (10) and balance the importance of each term  $S$  and  $C$  in the equation (spatial vs. topological information). Thus, it would be possible to obtain either better or equal results to CPD by cross validating  $\kappa$ . In our experiments, we decided not to add such a free parameter to avoid the results being dependent on how well  $\kappa$  was tuned. For instance, in the last grid experiment, one could set  $\kappa=1$  and turn off the contribution of the centralities ( $C$ ). In fact, by letting both terms equally contribute, we can better understand when the centrality term causes a positive or a negative impact and whether such a behaviour can be reproduced in other sets under the same conditions. So, by isolating certain properties of the point-sets, we now have a good indication of when the centralities can improve the registration process.

We plot the distribution of centrality values in Fig. 8 in order to understand why the centralities had poor convergence performance on the last experiment. Due to the grid sampling, many points have the same centrality value (depicted by the same colour). For instance, all points inside the object in the degree image have the same centrality value (e.g. 8 for an 8-connected neighborhood). When sampling using the degree, the distribution will have a high peak in 8 and smaller peaks for the points in the boundary. Eigenvector and closeness centralities have most discriminative distributions under grid sampling, although there are many nodes with similar centrality values, which are concentric inside the shape. Such property would explain why the eigenvector centrality and closeness were the winners in the last experiment, although all centralities suffered from the same convergence problem. Therefore, when performing registration of grid sampled points, centralities can still produce good results but it is clear that the impact on the registration should be reduced via  $\kappa$ .

**Computational complexity:** The computational cost of the overall method can be decomposed into several parts. There is the cost associated with the data graph construction, considering that the input is a point-set. In this case, the complexity is bounded by the technique applied, such as Delaunay triangulation and Minimum Spanning Tree. If the input is already a graph, this step is not required. The bottleneck of our approach is the centrality computation, which varies from the simple degree calculation ( $O(|V| + |E|)$ ) to the most complex centrality estimation. For instance, considering that it is required to know the shortest path between all points in the graph, the solution can be achieved by the Floyd-Warshall algorithm in which complexity is  $O(|V|^3)$  or with Johnson's algorithm on a sparse graph whose complexity is  $O(|V|^2 \log |V| + |V||E|)$ .



**Fig. 8.** Centrality distribution of an 8-connected graph. Points are colour-coded with the centrality value. (For interpretation of the references to colour in this figure caption, the reader is referred to the web version of this paper.)

## 7. Conclusions and future work

Point-set registration is a challenging problem in Computer Vision. In this paper, we introduced the graph centralities from the Social Network field into the registration problem. We added the topological contribution given by the centralities into the Coherent Point Drift algorithm. We performed extensive experiments and highlighted the important aspects for the successful centrality-based registration. The first aspect is the data graph creation which plays a major role. In this work, we only used the Delaunay triangulation but in the future we will be developing a centrality oriented technique for connecting unstructured points. This technique should cope with noise by reducing its impact. We fixed the noise and data graph parameters and varied properties of point-sets to understand the centrality behaviour. In the future, we will vary the data graph approach and the amount of noise in order to delineate which technique is able to move us one step further: faster convergence under the presence of noise.

Our main contribution and conclusions consist of bringing the centralities into the registration problem and showing how well they can perform and in which conditions they improve the registration results. Among all centralities, we observed that the closeness was the one which produced the best results. We also would like to study further the aggregation of centralities in order to combine them. For instance, we noticed through our experiments that eigenvector centrality has high discriminability in the grid example and it was also the fastest one in the random scenario. By combining them, one centrality could contribute in a different manner and cope with the weakness of the other. It is noticeable that the degree obtained poor results on all experiments and we would like to study each strength and weakness of individual centrality measures for vision tasks and to determine how can they be combined to improve the results.

### Conflict of interest

None declared.

### Acknowledgements

The authors would like to thank Séverine Habert and Marina Oikawa for the meaningful discussions and suggestions. The first author was financially supported by the Austrian Agency for

International Cooperation in Education & Research within the ÖAD Sonderstipendien program, financed by the Vienna PhD School of Informatics.

### References

- [1] W. Lian, L. Zhang, D. Zhang, Rotation-invariant nonrigid point set matching in cluttered scenes, *IEEE Trans. Image Process.* 21 (2012) 2786–2797.
- [2] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *Int. J. Comput. Vis.* 47 (2002) 7–42.
- [3] H. Jiang, S.X. Yu, D.R. Martin, Linear scale and rotation invariant matching, *IEEE Trans. Pattern Anal. Mach. Intell.* (2011).
- [4] S. Habert, P. Khurd, C. Chef'd'hotel, Registration of multiple temporally related point sets using a novel variant of the coherent point drift algorithm: application to coronary tree matching, *Proc. SPIE 8669* (2013) 86690M-1–86690M-11.
- [5] M.A. Audette, F.P. Ferrie, T.M. Peters, An algorithmic overview of surface registration techniques for medical imaging, *Med. Image Anal.* 4 (2000) 201–217.
- [6] M. Rodrigues, R. Fisher, Y. Liu, Special issue on registration and fusion of range images, *Comput. Vis. Image Understand.* 87 (2002) 1–7.
- [7] H. Pottmann, Q.-X. Huang, Y.-L. Yang, S.-M. Hu, Geometry and convergence analysis of algorithms for registration of 3D shapes, *Int. J. Comput. Vis.* 67 (2006) 277–296.
- [8] J. Salvi, C. Matabosch, D. Fofi, J. Forest, A review of recent range image registration methods with accuracy evaluation, *Image Vis. Comput.* 25 (2007) 578–596.
- [9] O. van Kaick, H. Zhang, G. Hamarneh, D. Cohen-Or, A survey on shape correspondence, *Comput. Graph. Forum* 30 (2011) 1681–1707.
- [10] G. Tam, Z.-Q. Cheng, Y.-K. Lai, F. Langbein, Y. Liu, D. Marshall, R. Martin, X.-F. Sun, P. Rosin, Registration of 3d point clouds and meshes: a survey from rigid to nonrigid, *IEEE Trans. Vis. Comput. Graph.* 19 (2013) 1199–1217.
- [11] P. Besl, N.D. McKay, A method for registration of 3-d shapes, *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (1992) 239–256.
- [12] A. Myronenko, X. Song, Point set registration: coherent point drift, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (2010) 2262–2275.
- [13] S. Rusinkiewicz, M. Levoy, Efficient variants of the icp algorithm, in: *The Third International Conference on 3-D Digital Imaging and Modeling*, 2001, pp. 145–152.
- [14] S. Umeyama, An eigendecomposition approach to weighted graph matching problems, *IEEE Trans. Pattern Anal. Mach. Intell.* 10 (1988) 695–703.
- [15] A.D.J. Cross, E. Hancock, Graph matching with a dual-step em algorithm, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (1998) 1236–1253.
- [16] B. Luo, E.R. Hancock, A unified framework for alignment and correspondence, *Comput. Vis. Image Understand.* 92 (2003) 26–55.
- [17] B. Luo, E.R. Hancock, Iterative procrustes alignment with the em algorithm, *Image Vis. Comput.* (2002).
- [18] T. Caetano, T. Caelli, D. Schuurmans, D.A.C. Barone, Graphical models and point pattern matching, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (2006) 1646–1663.
- [19] F. Zhou, F. De la Torre, Factorized graph matching, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [20] F. Zhou, F. De la Torre, Deformable graph matching, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [21] D. Conte, P. Foggia, C. Sansone, M. Vento, Thirty years of graph matching in pattern recognition, *Int. J. Pattern Recognit. Artif. Intell.* (2004).

- [22] L.C. Freeman, Centrality in social networks conceptual clarification, *Soc. Netw.* (1978) 215.
- [23] A. Bavelas, A mathematical model for group structures, *Hum. Organ.* 7 (1948).
- [24] K. Okamoto, W. Chen, X.-Y. Li, Ranking of closeness centrality for large-scale social networks, in: Proceedings of the Second Annual International Workshop on Frontiers in Algorithmics, Springer, Berlin, Heidelberg, 2008, pp. 186–195.
- [25] S. de Sousa, M. Balieiro, J. dos R. Costa, C. de Souza, Multiple social networks analysis of floss projects using sargas, in: HICSS '09. The 42nd Hawaii International Conference on System Sciences, 2009, pp. 1–10.
- [26] M.A. Balieiro, S. de Sousa, C.R.B. de Souza, Facilitating social network studies of FLOSS using the OSSNetwork environment, in: Proceedings of International Conference on Open Source Systems, OSS, IFIP, Springer, US, 2008, pp. 343–350.
- [27] G. Scott, L.H. Higgins, An algorithm for associating the features of two patterns, *Proc. R. Soc. Lond. B244* (1991), 21–26.
- [28] L.S. Shapiro, J.M. Brady, Feature-based correspondence: an eigenvector approach, *Image Vis. Comput.* 10 (1992) 283–288.
- [29] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: Proceedings of the 10th IEEE International Conference on Computer Vision - Volume 2, ICCV '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 1482–1489.
- [30] M. Carcassoni, E.R. Hancock, Spectral correspondence for point pattern matching, *Pattern Recognit.* 36 (2003) 193–204.
- [31] Y. Sahillioglu, Y. Yemez, Minimum-distortion isometric shape correspondence using em algorithm, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (2012) 2203–2215.
- [32] D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, E. Boyer, Articulated shape matching using laplacian eigenfunctions and unsupervised point registration, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [33] J. Utans, Mixture Models and the EM Algorithm for object recognition within compositional hierarchies, Technical Report, ICSI Berkeley, 1993.
- [34] S. Gold, A. Rangarajan, E. Mjølness, Learning with preknowledge: clustering with point and graph matching distance measures, *Neural Comput.* 8 (1996) 787–804.
- [35] B.N. Delaunay, Sur la sphère vide, *Bull. Acad. Sci. USSR* (1934) 793–800.
- [36] V. Jain, H. Zhang, Robust 3d shape correspondence in the spectral domain, in: IEEE International Conference on Shape Modeling and Applications, 2006, pp. 19–19.
- [37] L. Torresani, V. Kolmogorov, C. Rother, A dual decomposition approach to feature correspondence, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (2013) 259–271.
- [38] J.-D. Boissonnat, R. Dyer, A. Ghosh, Constructing stable delaunay triangulations, Research Report RR-8275, INRIA, 2013.
- [39] X. Ge, I. Safa, M. Belkin, Y. Wang, Data skeletonization via reeb graphs, in: NIPS, 2011, pp. 837–845.
- [40] W. Lian, D. Zhang, Rotation-invariant nonrigid shape matching in cluttered scenes, in: European Conference in Computer Vision, 2010.
- [41] C.H. Papadimitriou, The Euclidean travelling salesman problem is np-complete, *Theor. Comput. Sci.* 4 (1977) 237–244.
- [42] W. Lian, D. Zhang, Rotation-invariant nonrigid point set matching in cluttered scenes, *IEEE Trans. Image Process.* 21 (2012) 2786–2797.
- [43] S. Ohrhallinger, S.P. Mudur, An efficient algorithm for determining an aesthetic shape connecting unorganized 2d points, *Comput. Graph. Forum* 32 (2013) 72–88.
- [44] A. Bavelas, A mathematical model for small group structures, *Hum. Organ.* 7 (1948) 16–30.
- [45] A. Bavelas, Communication patterns in task oriented groups, *J. Acoust. Soc. Am.* 22 (1950) 271–282.
- [46] Z. Wang, Y. Han, T. Lin, H. Tang, S. Ci, Virtual network embedding by exploiting topological information, in: IEEE Global Communications Conference (GLOBECOM), 2012, pp. 2603–2608.
- [47] A.B.M. Nasiruzzaman, H. Pota, M. Mahmud, Application of centrality measures of complex network framework in power grid, in: IECON 2011–37th Annual Conference on IEEE Industrial Electronics Society, 2011, pp. 4660–4665.
- [48] R. Pal, A. Mukherjee, P. Mitra, J. Mukherjee, Modelling visual saliency using degree centrality, *IET Comput. Vis.* 4 (2010) 218–229.
- [49] D. Klein, M. Randic, Resistance distance, *J. Math. Chem.* 12 (1993) 81–95.
- [50] E. Bozzo, M. Franceschet, Resistance distance, closeness, and betweenness, *Soc. Netw.* 35 (2013) 460–469.
- [51] S. Wasserman, K. Faust, *Social Network Analysis. Methods and Applications*, Cambridge University Press, New York, USA, 1994.
- [52] L.C. Freeman, A set of measures of centrality based on betweenness, *Sociometry* 40 (1977) 35–41.
- [53] M. Newman, A measure of betweenness centrality based on random walks, *Soc. Netw.* 27 (2005) 39–54.
- [54] Q. Li, Z. Qin, L. Chai, H. Zhang, J. Guo, B. Bhanu, Representative reference-set and betweenness centrality for scene image categorization, in: 20th IEEE International Conference on Image Processing (ICIP), 2013.
- [55] A. Mantrach, L. Yen, J. Callut, K. Françoise, M. Shimbo, M. Saerens, The sum-over-paths covariance kernel: a novel covariance measure between nodes of a directed graph, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (2010) 1112–1126.
- [56] S. García-Díez, F. Fouss, M. Shimbo, M. Saerens, A sum-over-paths extension of edit distances accounting for all sequence alignments, *Pattern Recognit.* 44 (2011) 1172–1182.
- [57] M. Senelle, S. García-Díez, A. Mantrach, M. Shimbo, M. Saerens, F. Fouss, The sum-over-forests density index: identifying dense regions in a graph, *Clin. Orthop. Relat. Res.* (2013), abs/1301.0725.
- [58] E. Harary, Status and contrastatus, *Sociometry* 22 (1959) 23–43.
- [59] M. Beauchamp, An improved index of centrality, *Behav. Sci.* (1965).
- [60] G. Sabidussi, The centrality index of a graph, *Psychometrika* (1966).
- [61] S. de Sousa, N.M. Artner, W.G. Kropatsch, On the evaluation of graph centrality for shape matching, in: GbRPR, Lecture Notes in Computer Science, vol. 7877, Springer, Springer Berlin Heidelberg, 2013, pp. 204–213.
- [62] J. Cheeger, A lower bound for the smallest eigenvalue of the Laplacian, in: *Problems in Analysis*, 1970.
- [63] B. Mohar, Laplace eigenvalues of graphs—a survey, *Discr. Math.* 109 (1992) 171–183.
- [64] W. Donath, A. Hoffman, Lower bounds for the partitioning of graphs, *IBM Res. Dev.* (1973).
- [65] J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (2000) 888–905.
- [66] H. Qiu, E.R. Hancock, Image segmentation using commute times, in: British Machine Vision Conference, 2005, pp. 929–938.
- [67] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, *Comput. Netw. ISDN Syst.* 30 (1998) 107–117.
- [68] P. Gomo, Pagerank image denoising, in: A. Campilho, M. Kamel (Eds.), *Image Analysis and Recognition. Lecture Notes in Computer Science*, vol. 6111, Springer, Berlin, Heidelberg, 2010, pp. 1–10.
- [69] S. Mukherjee, S. Biswas, D. Mukherjee, Recognizing human action at a distance in video by key poses, *IEEE Trans. Circuits Syst. Video Technol.* 21 (2011) 1228–1241.
- [70] F. Harary, R. Norman, *Graph Theory as a Mathematical Model in Social Science*, University of Michigan Press, Ann Arbor, 1953.
- [71] W.G. Kropatsch, A. Ion, Y. Haxhimusa, T. Flanitzner, The eccentricity transform (of a digital shape), in: *Discrete Geometry for Computer Imagery, Lecture Notes in Computer Science*, vol. 4245, Springer, Berlin, Heidelberg, 2006, pp. 437–448.
- [72] M. Saerens, F. Fouss, L. Yen, P. Dupont, The principal components analysis of a graph, and its relationships to spectral clustering, in: Proceedings of the 15th European Conference on Machine Learning, Springer-Verlag, Springer Berlin Heidelberg, 2004, pp. 371–383.
- [73] H. Qiu, E.R. Hancock, Clustering and embedding using commute times, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (2007) 1873–1890.
- [74] A. Brouwer, W. Haemers, *Spectra of Graphs*, Springer, New York, 2012.
- [75] D. Sharvit, J. Chan, H. Tek, B.B. Kimia, Symmetry-based indexing of image databases, *J. Vis. Commun. Image Represent.* 9 (1998) 366–380.

**Samuel de Sousa** received his B.Sc. degree in Computer Science, in 2009, from the Federal University of Pará (UFPA), Brazil, his M.Sc., in 2012, from the Federal University of Minas Gerais (UFMG), Brazil. He is currently a Ph.D. Candidate at the Vienna University of Technology, Austria. His research interests include 3D Computer Vision, Pattern Recognition, and Machine Learning.

**Walter G. Kropatsch** founded the PRIP group at Vienna University of Technology, in 1990. He has received his Ph.D. in computer science from the Technical University of Graz, in 1982, and his habilitation (Venia Docendi) from the University of Innsbruck, in 1991. He has done sustained research in the fields of irregular pyramids, graphs, and their application in various computer vision areas.