

Robust elastic 2D/3D geometric graph matching

Eduard Serradell^a, Jan Kybic^b, Francesc Moreno-Noguer^a, Pascal Fua^c

^aInstitut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain

^bFaculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic

^cEcole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

ABSTRACT

We present an algorithm for geometric matching of graphs embedded in 2D or 3D space. It is applicable for registering any graph-like structures appearing in biomedical images, such as blood vessels, pulmonary bronchi, nerve fibers, or dendritic arbors. Our approach does not rely on the similarity of local appearance features, so it is suitable for multimodal registration with a large difference in appearance. Unlike earlier methods, the algorithm uses edge shape, does not require an initial pose estimate, can handle partial matches, and can cope with nonlinear deformations and topological differences.

The matching consists of two steps. First, we find an affine transform that roughly aligns the graphs by exploring the set of all consistent correspondences between the nodes. This can be done at an acceptably low computational expense by using parameter uncertainties for pruning, backtracking as needed. Parameter uncertainties are updated in a Kalman-like scheme with each match.

In the second step we allow for a nonlinear part of the deformation, modeled as a Gaussian Process. Short sequences of edges are grouped into superedges, which are then matched between graphs. This allows for topological differences. A maximum consistent set of superedge matches is found using a dedicated branch-and-bound solver, which is over 100 times faster than a standard linear programming approach. Geometrical and topological consistency of candidate matches is determined in a fast hierarchical manner.

We demonstrate the effectiveness of our technique at registering angiography and retinal fundus images, as well as neural image stacks.

Keywords: graph matching, image registration, fibers, vessels, dendrites

1. INTRODUCTION

Graph-like structures are pervasive in biomedical 2D and 3D images. They may be blood vessels, pulmonary bronchi, nerve fibers, or dendritic arbors, which can be acquired at different times and scales, or using different modalities. This may result in vastly diverse image appearances. For example, blood vessels acquired using two-photon microscopy such as those on the left of Fig. 1 look radically different when imaged using an electron microscope, as shown in the center and right of Fig. 1.

Such drastic appearance changes make it impractical to use registration techniques that rely either on maximizing image similarity or finding image-based correspondences.¹ Since the graph structure may be the only thing that is shared across modalities, matching graphs, or subgraphs when the image have been acquired at different resolutions, therefore becomes the only effective registration means. Unfortunately, most existing techniques that attempt to do this rely on matching Euclidean or geodesic distance between graph junction points.²⁻⁴ This assumes that they are preserved and cannot handle true non-rigid motion or graphs with different topology.

In this paper, we therefore propose a new approach to matching graph structures embedded in either \mathbb{R}^2 or \mathbb{R}^3 . We require no initial pose estimate, can handle nonlinear deformations, and do not rely on local appearance or global distance matrices. Instead, given graphs in the two images or image-stacks to be registered, we treat graph nodes and edges as the features to be matched. We first find an affine transform that roughly aligns the images. To this end, we show that we can start with completely unknown deformation parameters and progressively refine them as more and more correspondences are hypothesized. This, in turn, reduces the number of potential correspondences for unmatched nodes. Thus, even though the total number of potential correspondences is *a priori* huge, we can nevertheless explore very efficiently the complete

E-mail: eserradell@iri.upc.edu, kybic@fel.cvut.cz

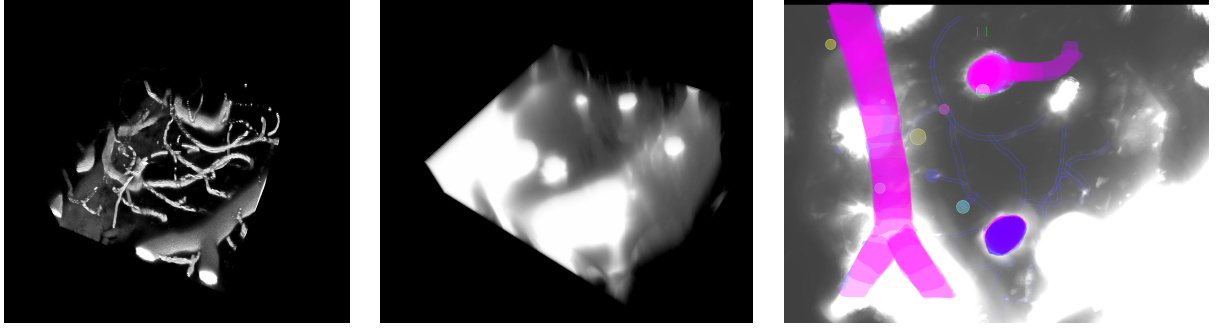


Figure 1. Blood vessels network in brain tissue. **Left:** Projection of the 3D image stack acquired using a two photon microscope. **Center:** The same tissue imaged using a bright-field optical microscope after excision and fixation. The quality is much worse and the smaller blood vessels are barely visible. **Right:** Maximum intensity projection overlaid with a manual segmentation of the blood vessels.

set until we find a satisfactory subset. This yields an estimate of the desired affine transform. By using a branch-and-bound strategy, we can then add to it an elastic deformation component that allows us match additional nodes and achieve better accuracy.

We demonstrate the effectiveness of our technique by registering angiography and retinal-fundus images acquired at different times, as well as neural image-stacks acquired using different modalities. The first application is important to quantify the deformations and the second to bring the clues found in different images into a common frame of reference, so as to build a unified model.

2. RELATED WORK

Many registration techniques rely on finding a rigid or non-rigid transformation that maximizes image-similarity, often expressed in terms of correlation or mutual-information.¹ However, when dealing with images such as those of Fig. 1, the dissimilarity is such, that these methods do not perform well. A possible approach is to explicitly match the graph structures. Current methods can be categorized into three classes:

The first class assumes the graphs to be related by a low-dimensional geometric transformation, such as a rigid one. In this case, using either RANSAC⁵ or an improved version of it that takes appearance into account⁶ is an attractive option. Unfortunately RANSAC-style approaches do not scale-up well to transformations with higher number of parameters.

The second class requires a good initial estimate of the transformation and progressively refines it. For rigid transformations, one of the earliest algorithm is the Iterative Closest Point (ICP) method,⁷ later extended to non-rigid transformations.^{8,9} A related techniques is the Coherent Point Drift (CPD).¹⁰ However, these approaches only perform well when the initial estimate is good enough.

The third class relies on local descriptors to determine pairwise similarity between nodes, based on appearance, Euclidean or geodesic distances to other nodes,^{3,4} or neighborhood similarity.² Correspondences are then estimated using spectral techniques,¹¹ linear or quadratic programming,⁴ or multidimensional optimization techniques such as softassign.² However, this is impractical when the local appearance of nodes is very different in the two graphs or distances may vary due to nonlinear deformations or topology changes.

By contrast, the technique we propose here can recover a high-dimensional elastic deformation, does not require an initial estimate of this deformation, and performs well even if the local appearances of the graph nodes are very different.

3. APPROACH OVERVIEW

Let us assume we are given two graphs $A = (V_A, E_A)$ and $B = (V_B, E_B)$ that have been obtained from two images or image-stacks to be registered. The E represent edges and the V nodes, which can be either 2D or 3D vertices. These graphs may have been subject to preprocessing, in order to eliminate very short branches, merge close nodes, attaching nodes to close branches, suppressing non-branching nodes, smoothing and reinterpolating the edges.

In a first processing stage described in Section 4, we rely on the fact that an affine transformation usually approximates well the nonlinear deformation that we finally want to recover. We take the initial mapping to be the identity with a large uncertainty on the parameters. We then pick a random pair of nodes as our first correspondence. This lets us invoke a Kalman-like scheme¹² to reduce the uncertainties and to narrow down the set of potential correspondences for the next node. Repeating this procedure recursively until enough mutually consistent correspondences have been established and backtracking when necessary lets us explore the set of potential correspondences quickly. This yields both an acceptable set of correspondences and an estimate of the affine deformation relating the two images or image stacks. This algorithm is described in more details in Section 4.

Because the deformation is usually not truly affine, in a second processing stage described in Section 5, we introduce an elastic deformation component that allows us to match additional nodes and edges. It uses a branch-and-bound strategy to find the largest possible non-conflicting set of correspondences by maximizing the total length of matched edges, thus allowing for controlled topological differences.

4. ROBUST AFFINE MATCHING

Let \mathbf{x}_i and \mathbf{y}_i be the location of corresponding nodes in A and B, respectively. Our goal is to retrieve as many such correspondences as possible and the affine mapping $\mathbf{y} = T_\theta \mathbf{x}$ that best aligns them.

Drawing our inspiration from,^{12,13} we propose an iterative approach to simultaneously establishing correspondences and retrieving the mapping parameters. We represent T_θ by a rotation matrix, an upper triangular matrix of shear and scaling parameters, and a translation vector.

$$\mathbf{y} = T_\theta \mathbf{x} = R(\omega_x, \omega_y, \omega_z) \begin{bmatrix} 1 & s_{xy} & s_{xz} \\ 0 & 1 & s_{yz} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} \mathbf{x} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (1)$$

where the matrix T_θ has been decomposed into components representing rotation, scaling, shear and translation. To avoid the gimbal lock problem related to Euler angles, \mathbf{R} is expressed as an exponential map, which is not singular for small amounts of rotation. Therefore, the transform is represented by a parameter vector $\theta = \{\omega_x, \omega_y, \omega_z, s_x, s_y, s_z, s_{xy}, s_{xz}, s_{yz}, t_x, t_y, t_z\}$ which is a 12-dimensional vector in 3D. In 2D, θ is a 7-dimensional vector.

After establishing k matches $\{\mathbf{x}_k, \mathbf{y}_k\}$ between graphs A and B and assuming that our transformation approximately preserves geodesic distance ratios, we take the geodesic compatibility ratio \mathcal{V} of a new potential correspondence $\mathbf{x}_l \leftrightarrow \mathbf{y}_m$ to be

$$\mathcal{V}(\mathbf{x}_l, \mathbf{y}_m) = \text{var} \left\{ \frac{g_{1l}^A}{g_{1m}^B}, \dots, \frac{g_{kl}^A}{g_{km}^B} \right\} . \quad (2)$$

where g_{ij} is the geodesic distance. It is zero if the potential correspondences perfectly preserve the distance ratios and grows as this constraint becomes more severely violated.

We consider T_θ to be random, with normally distributed parameters θ . Initially, we set the mean of θ such that T_θ is identity with a large diagonal covariance matrix Σ_θ . We initialize the set of matches M_0 by randomly selecting a pair of nodes from A and B. We then recursively add new correspondences and update θ as follows.

1. We project all nodes $\hat{\mathbf{y}}_i = T_\theta \mathbf{x}_i$ using the current transformation.
2. For each $\hat{\mathbf{y}}_i$, we search for potential matches $\mathbf{y}_j \in B$ in a bounded elliptical region defined by the variance of $\hat{\mathbf{y}}_i$. We further reduce the number of potential matches by considering only points $\mathbf{y}_j \in B$ whose geodesic compatibility ratio $\mathcal{V}(\mathbf{x}_i, \mathbf{y}_j)$ is below a threshold.
3. We then choose the node \mathbf{x}_i with the smallest number of potential matches, and randomly pick one of them to define the match $\mathbf{x}_i \leftrightarrow \mathbf{y}_j$. We update the set $M_0 \leftarrow M_0 \cup \{\mathbf{x}_i \leftrightarrow \mathbf{y}_j\}$.
4. Given the potential match $\mathbf{x}_i \leftrightarrow \mathbf{y}_j$ we update the vector of parameters and covariance matrix using the Kalman-like update equations

$$\begin{aligned} \theta^+ &= \theta + K(\mathbf{y}_j - T_\theta \mathbf{x}_i) \\ \Sigma_{\theta^+} &= (I - KJ)\Sigma_\theta \end{aligned}$$

where \mathbf{J} is the Jacobian of T_θ , \mathbf{K} is the Kalman gain and \mathbf{I} the identity matrix.

Hypothesizing new matches and updating the parameters is repeated recursively until a large enough inlier consensus set is found, as in RANSAC. If that does not happen, the algorithm backtracks and selects different correspondences using a depth-first search strategy. Thus, it explores quickly and exhaustively the whole set of potential correspondences until it finds the consensus set. Note that we do not use geodesic distances as features to be matched as in.^{3,4} Instead, we use geodesic distance ratios, which vary little under the affine transformations we seek, as weak priors for pruning, allowing small deviations from linearity.

5. ELASTIC MATCHING

We now attempt to find a nonlinear transformation that is consistent with the initial matches M_0 and close to T_θ computed above, with additional nodes and edges being matched. To this end, we seek a maximally consistent set of matches, where the matches are between short paths consisting of at most L consecutive edges (typically $L = 3$) that we will refer to as *superedges*. By matching superedges of different lengths, we effectively allow some nodes to be skipped, i.e. the graphs do not have to be isomorphic.

5.1 Gaussian process deformation model

Corresponding points \mathbf{x}_i in graph A and \mathbf{y}_i in graph B are now assumed to be related through $\mathbf{y}_i = T_\theta(\mathbf{x}_i) + \boldsymbol{\xi}(\mathbf{x}_i)$ where the nonlinear part of the deformation, $\boldsymbol{\xi}$, is modeled as a Gaussian process (GP),¹⁴ with spatial correlation

$$\mathbb{E}[\boldsymbol{\xi}_i(\mathbf{x})\boldsymbol{\xi}_j(\mathbf{x}')] = \begin{cases} 0 & \text{if } i \neq j \\ \sigma_\xi^2 \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma_\lambda^2}\right) & \text{if } i = j \wedge \mathbf{x} \neq \mathbf{x}' \\ \sigma_\xi^2 + \rho_i^2 & \text{if } i = j \wedge \mathbf{x} = \mathbf{x}' \end{cases} \quad (3)$$

Correspondences between superedges are converted to point pair correspondences by sampling the paths equidistantly, for leaf edges only taking the length of the shorter edge. A set of matches is accepted if the Mahalanobis norm of the residuals $\boldsymbol{\xi}$ is smaller than a threshold determined from the desired acceptance rate. Finally, we determine the deformation function everywhere by krigging.¹⁴

To evaluate the acceptability of a set of matches including superedges, we represent the paths by equidistantly spaced points. This way, we obtain a set of point matches and the Mahalanobis norm test is used. For matches involving leaf edges, the longer superedge is cut to the same length as the shorter one.

5.2 Maximum weighted independent set formulation

We construct a set of possible node matches $M \subseteq V_A \times V_B$, which are geometrically consistent, and optionally also geodesically consistent, with the initial matches M_0 . Then, we generate a set Q of allowable superedge pairs, called *superpairs*. A pair (p, q) is allowable if: (i) both endnode matches are among the set of possible matches M , (ii) the ratio between the lengths of p and q is at most κ , and (iii) the match (p, q) is geometrically consistent with M_0 . Fortunately, the number of superpairs $|Q|$ does not grow very fast with the number of edges. For example, the two trees shown in Figure 4 (bottom) have 24 and 31 nodes, which leads to 186 and 244 superedges but only 436 superpairs. Note also that reversing all edges in a superpair produces another valid superpair.

Finally, we determine pairwise compatibilities between superpairs. Two superpairs $k, l \in Q$ are compatible (denoted $(k, l) \in C$) if (i) the constituent superedges have no overlap, (ii) the endpoint assignments are not conflicting, (iii) the superpairs are geometrically consistent and optionally also geodesically consistent. On the average, we found about 25% of pairs of superpairs to be compatible. Finding C is easily parallelisable.

The graph matching problem becomes one of finding the best compatible set of superpairs

$$Q^* = \arg \max_{Q' \subseteq Q} \sum_{k \in Q' \subseteq Q} \varphi_k \quad \text{such that} \quad \forall k, l \in Q'; (k, l) \in C, \quad (4)$$

where the contribution φ_k of each superpair $k = (p, q)$ is the length of p or the shorter of p, q in case of leaf edges. This length is further multiplied by θ^n , with $\theta = 0.98$ and n the number of constituent edges of p to slightly penalize skipped nodes.

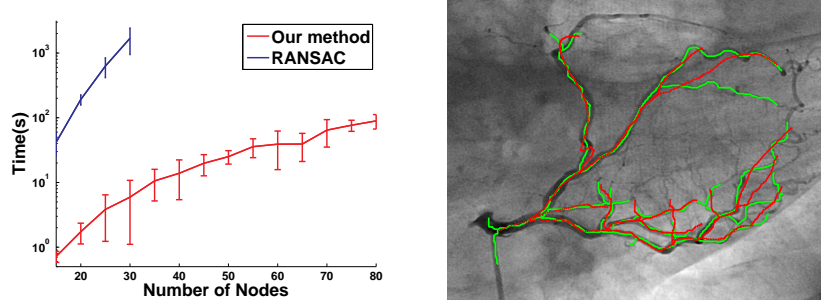


Figure 2. Comparing against other approaches. **Left:** Time required by RANSAC and our algorithm as a function of the number of nodes. **Right:** Results of the CPD algorithm¹⁰ for the angiography dataset of Fig. 4 given the initial affine estimate computed using the method of Section 4.

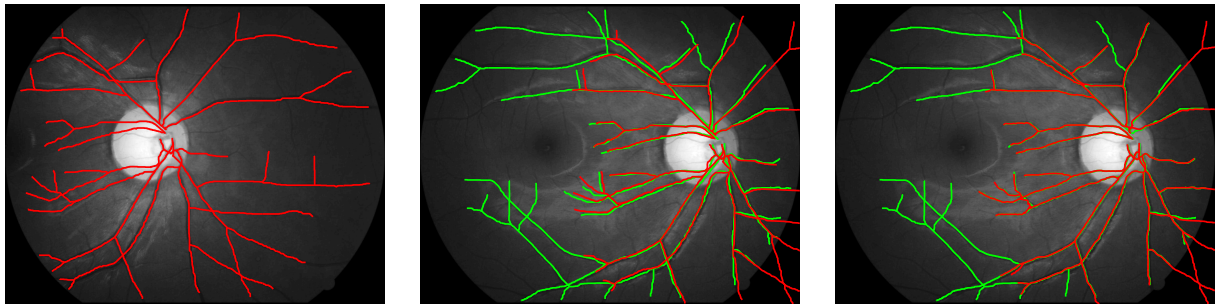


Figure 3. Retinal fundus images used in.⁴ **Left,Center:** Two images of the same retina taken from different viewpoints, with the vascular trees overlaid in red and green. The first tree is overlaid in red over the second image after affine transformation. **Right:** The first tree is overlaid in red over the second image after affine plus elastic deformation.

5.3 Branch and bounds algorithm

The superpairs are considered in the decreasing order of φ_k . In each search state, the membership of the first $K - 1$ superpairs in Q' is fixed, and we branch into two new states based on the membership of superpair K . A lower bound F^- of F is calculated by a greedy algorithm. The upper bound F^+ based on the fact that the number of matched superpairs cannot exceed the number of edges is used for pruning. The states are kept in a priority queue ordered by F^+ , possibly with a limited length.

The computational time varies greatly depending on the size and the density of the graph, which is related to the measurement accuracy and the amount of non-linear deformation. For the examples in Section 6 it took between 15 s and 30 min on 2×4 core Intel Xeon E5620.

6. EXPERIMENTS

In this section, we present our results first on synthetic and then on real data.

6.1 Synthetic Data

To compare our method's computational complexity against that of a standard method such as RANSAC,⁵ we generated random sets of 2D points of increasing cardinality, computed a Minimum Spanning Tree to create the A graphs of Section 3, applied a random affine transform to create the corresponding B graphs, corrupted the points' locations with noise, and added 50% additional points not belonging to the trees to act as outliers.

We then ran both RANSAC and the affine-estimation algorithm of Section 4 on the resulting datasets and repeated the experiment 100 times. The graph of Fig. 2 shows the required processing time as a function of the number of nodes on a log-scale. Above 30 nodes, RANSAC becomes impractically slow whereas our approach remains feasible much longer.

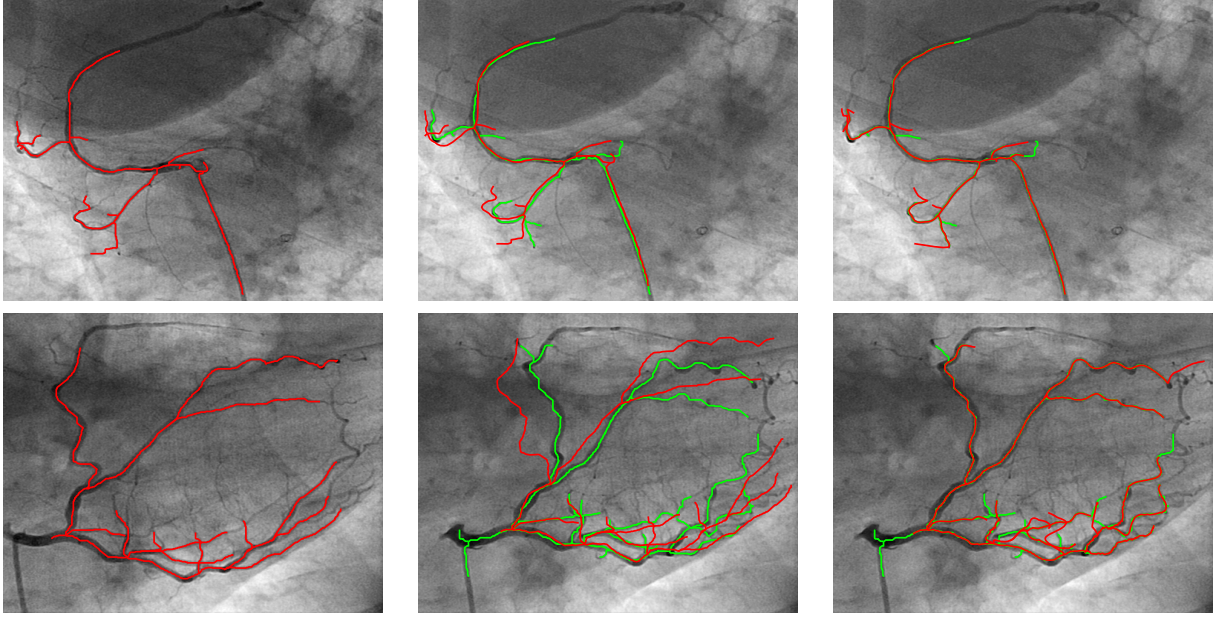


Figure 4. Angiography images from a beating heart. **Left column:** Two different images with extracted vascular trees overlaid in red. **Center column:** Two other images taken later in the cycle with extracted vascular trees overlaid in green. The red trees are also shown after affine transformation. **Right column:** The red trees are overlaid after affine plus elastic deformation. Note that the trees in the second have distinctly different topologies, which affects our algorithm very little.

6.2 Real Data

Next, we present some examples of the results obtained by our algorithm on several different real datasets. The graphs were extracted semi-automatically using the Fiji * platform and its plugins.

In Fig. 3 we show registration results for retinal fundus vascular trees that are deformed from one image to the next because the camera is placed in different points. This results in apparent distortions of the curved retinal surface’s projection, which are well modeled by an affine transform. Because there is very little nonlinearity in the deformation, these results are similar to those of the method of Deng et al.,⁴ even though the trees overlap only partially.

In the 2D X-ray angiography images of Fig. 4 the deformations are much more nonlinear. As shown on the right, our algorithm nevertheless does a good job of recovering this more complex deformation and aligning the trees. For comparison purposes, we tried the CPD¹⁰ method on these images. If we start it without a correct estimate of the nonlinear part of the transform, it fails completely. If we supply it with our affine estimate, CPD works as long as the nonlinear deformation is small but not otherwise, as shown on the right of Fig. 2. To compare CPD accuracy against ours in the absence of ground truth, we computed the minimum distance assignment for each node in the deformed graphs, overlaid in red on the right side of Figs. 2 and 4 (bottom), to the nearest neighbor in the corresponding graph overlaid in green. The average distance for our method is 1.78 pixels against 4.36 for CPD, thus indicating better accuracy on images of size 512×512 . The corresponding numbers for the less deformed angiograms at the top of Fig. 4 are 1.38 and 1.58 and for the retinal scans of Fig. 3 are 2.2124 and 2.9822 for images of size 1260×1548 .

Finally, we register the 3D vascular trees extracted from the brain tissue of Fig. 1 using two different modalities. Even though the two images look extremely different, our algorithm returns a valid deformation as shown in Fig. 5.

7. CONCLUSIONS

We have shown that our algorithm can match graphs without either appearance information or initial pose estimate, while allowing for partial matches and nonlinear deformations. The key is to use the graph structure itself as a feature. We

*<http://pacific.mpi-cbg.de>

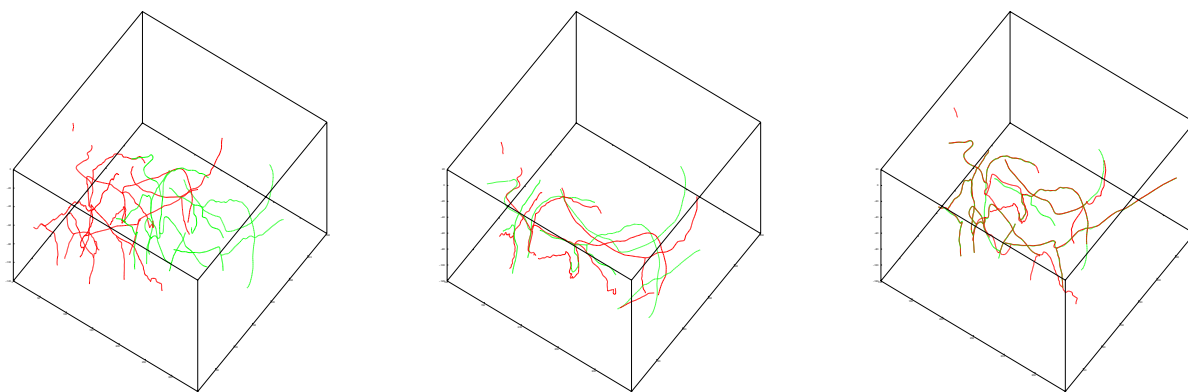


Figure 5. Results on the microscopy images of Fig. 1. **Left:** The trees extracted from the two image stacks before registration. **Center:** After affine alignment. **Right:** After affine and elastic alignment.

use geometrical and geodesical constraints as weak priors for robust sparse matching. The constraints are then relaxed and the combinatorial matching problem is converted to a maximum independent set form. The algorithm handles graphs containing up to $100 \sim 150$ nodes, even when reliable appearance information is not available and nonlinear deformations are present. In the future, algorithmic improvements should enable us to handle even larger graphs.

Acknowledgements

Jan Kybic was supported by the Czech Science Foundation project P202/11/0111. This project was also funded in part by the ERC project MicroNano, the Spanish Ministry of Science and Innovation CICYT project PAU+ DPI2011-27510, the MIPRCV Consolider Ingenio 2010 project CSD2007-00018 and by the EU project GARNICS FP7-247947.

REFERENCES

- [1] Zitova, B. and Flusser, J., “Image Registration Methods: a Survey,” *Image Vision Computing* **11**(11), 977–1000 (2003).
- [2] Gold, S., Rangarajan, A., Lu, C., and Mjolsness, E., “New Algorithms for 2d and 3D Point Matching: Point Estimation and Correspondence,” *Pattern Recognition* **31**(8), 1019–1031 (1998).
- [3] Smeets, D., Bruyninckx, P., Keustermans, J., Vandermeulen, D., and Suetens, P., “Robust Matching of 3D Lung Vessel Trees,” in [*MICCAI Workshop on Pulmonary Image Analysis*], (2010).
- [4] Deng, K., Tian, J., Zheng, J., Zhang, X., Dai, X., and Xu, M., “Retinal Fundus Image Registration Via Vascular Structure Graph Matching,” *International Journal of Biomedical Imaging* (2010).
- [5] Fischler, M. and Bolles, R., “Random Sample Consensus: a Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography,” *Communications ACM* **24**(6), 381–395 (1981).
- [6] Chum, O. and Matas, J., “Matching With Prosac - Progressive Sample Consensus,” in [*Conference on Computer Vision and Pattern Recognition*], 220–226 (June 2005).
- [7] Besl, P. and Mckay, N., “A Method for Registration of 3D Shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**, 239–256 (February 1992).
- [8] Hahnel, D., Thrun, S., and Burgard, W., “An Extension of the ICP Algorithm for Modeling Nonrigid Objects With Mobile Robots,” in [*International Joint Conference on Artificial Intelligence*], 915–920 (2003).
- [9] Li, H., Sumner, R., and Pauly, M., “Global Correspondence Optimization for Non-Rigid Registration of Depth Scans,” in [*Proceedings of the Symposium on Geometry Processing*], 1421–1430 (2008).
- [10] Myronenko, A. and Song, X., “Point-Set Registration: Coherent Point Drift,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(12), 2262–2275 (2010).
- [11] Leordeanu, M. and Hebert, M., “A Spectral Technique for Correspondence Problems Using Pairwise Constraints,” in [*International Conference on Computer Vision*], 1482–1489 (October 2005).

- [12] Moreno-Noguer, F., Lepetit, V., and Fua, P., “Pose Priors for Simultaneously Solving Alignment and Correspondence,” in [*European Conference on Computer Vision*], (October 2008).
- [13] Serradell, E., Özuysal, M., Lepetit, V., Fua, P., and Moreno-Noguer, F., “Combining Geometric and Appearance Priors for Robust Homography Estimation,” in [*European Conference on Computer Vision*], 58–72 (September 2010).
- [14] Rasmussen, C. E. and Williams, C. K., [*Gaussian Process for Machine Learning*], MIT Press (2006).