AUTOMATIC ALIGNMENT OF STACKS OF FILAMENT DATA

V.J. Dercksen, B. Weber, D. Günther, M. Oberlaender, S. Prohaska, H.-C. Hege*

Zuse Institute Berlin (ZIB), Takustraße 7, 14195 Berlin, Germany.
*Max Planck Institute of Neurobiology, Am Klopferspitz 18, 82152, Martinsried, Germany.

ABSTRACT

We present a fast and robust method for the alignment of image stacks containing filamentous structures. Such stacks are usually obtained by physical sectioning a specimen, followed by an optical sectioning of each slice. For reconstruction, the filaments have to be traced and the sub-volumes aligned. Our algorithm takes traced filaments as input and matches their endpoints to find the optimal transform. We show that our method is able to quickly and accurately align sub-volumes containing neuronal processes, acquired using brightfield microscopy. Our method also makes it possible to align traced microtubuli, obtained from electron tomography data, which are extremely difficult to align manually.

***Index Terms*—** Feature-based alignment, serial sections, point matching, neurons, microtubuli.

1. INTRODUCTION

Physical sectioning combined with optical sectioning enables researchers to image large tissue volumes at sub-micrometer resolution. Optical sectioning microscopy, e.g. using confocal laser scanning or transmitted light brightfield microscopy, permits the three-dimensional imaging of the internal structure of tissue by varying the focal plane of the microscope. However, the resolution that can be obtained is limited by the thickness of the tissue, because of light scattering. Thick tissue can be imaged by first cutting the specimen into thin sections, which are mounted and optically sectioned individually to produce a stack of thin, but three-dimensional sub-volumes [1, 2]. When using electron tomography instead of light microscopy, individual cells can be studied at even smaller scale [3].

To reconstruct the entire volume, the sub-volumes representing the physical sections have to be aligned, as the individual sections may have different positions and orientations on the microscope slides. There are basically two different approaches to this problem: image-based and feature-based alignment. Problems with image-based alignment include imaging artifacts like noise, uneven brightness and contrast, as well as the potentially large data sets (multiple gigabytes per section). It is therefore often beneficial to first extract features from the image data and perform the alignment on the features. This is the approach taken in this paper.

We present an automatic method to align a pair of sub-volumes containing filamentous structures. The problem can be described more specifically as follows (see Fig. 1): given two sub-volumes S_0 and S_1 containing a potentially different number of line segments (polylines), we are looking for a transform T which maps the end points of segments in S_1 onto the corresponding end points in S_0 in an optimal way. A prerequisite for computing T is the assignment of correspondences between segments in both slices, for which a solution is presented as well. We assume that enough corresponding segments are available to find a correct match and that the slices have not been severly deformed beyond a rigid transform. We apply our fast and robust method to neurons and microtubuli.

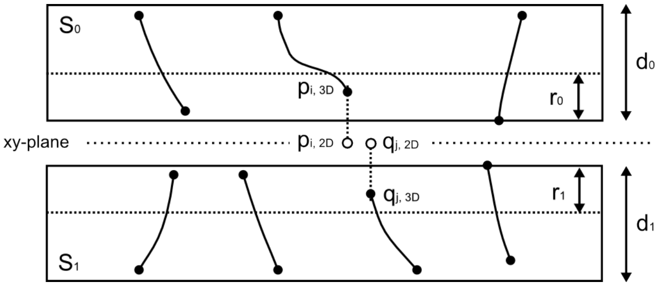

Fig. 1. Problem: alignment of slices containing filaments

2. RELATED WORK

Many different methods exist for alignment based on image data directly and image-based feature descriptors [4]. Such methods however cannot be used in our case as our objects of interest are very thin and difficult to distinguish, especially in noisy images.

In our case, the filaments are obtained by manual or automatic tracing and the features to be aligned are the segment end points. Several methods have been proposed to solve the challenge of simultaneous optimization of 1) point correspondences between two point sets of possibly different size and 2) the transform with respect to a particular objective function.

The Softassign Procrustes Matching Algorithm [5] solves this problem by optimizing a non-linear objective function in which a match matrix, encoding the point correspondences, is introduced into the Procrustes distance function. Chui [6] extends this method for non-rigid point matching. Bin Luo [7] also extends the Procrustes analysis, but casts the problem into an Expectation Maximization framework.

Bajcsy et al. [2] solved the problem in order to align segmented extracellular matrix proteins and blood vessels contained in a stack of Confocal Laser Scanning Microscopy sub-volumes. A rough initial transform is computed by finding a matching of segments with similar cross-section area and similar distances between the centroids. The transform is then optimized taking into account also angular constraints. In [8], the transform for the found matching is further optimized. A polynomial function is fit to the segments and extrapolated onto a plane between the sub-volumes. The optimal transform is then computed by minimizing position deviation of the extrapolated points or by maximizing curve smoothness.

Characterization of molecule similarity, important in drug design, is another problem often posed as 3D-point matching [9, 10]. Kirchner [11] optimizes an objective function involving the fraction of matched points and the *root mean square distance (rmsd)* between matched points after transformation. He shows that there exists a *fully polynomial time approximation scheme* (FPTAS) for this problem, which however runs in polynomial time of high degree.

3. MATCHING AND ALIGNING 2D POINT SETS

Of all methods methods mentioned above, we chose to use the point matching approach of Baum [9] as a basis for the alignment of filament data, as it is fast, robust, flexible and straightforward to implement. In its original form, it computes the correspondence between two 3D point sets of possibly different size and the rigid transform that optimally maps one point set onto the other (in a least squares sense). We extend it to the alignment of 2D point sets obtained from traced filaments, describe how to incorporate an optional uniform scaling in the transform and present some efficiency optimizations.

Given two sections containing multiple polylines, our method computes an optimal transform, i.e. a rotation angle around the z-axis, a 2D translation parallel to the xy-plane and an optional uniform scaling, using the following steps:

1. Find the sets of points $P = \{\mathbf{p}_i\}$ from the first slice S_0 and $Q = \{\mathbf{q}_j\}$ from the other slice S_1, to be matched. We use the line end points in the section boundary regions r_0 and r_1 respectively (see Fig. 1) and obtain 2D point sets by orthogonal projection onto the xy-plane. The region size is chosen as a fraction of the total slice thickness: $r_0 = \beta d_0, \beta \in (0, \frac{1}{2}]$.
2. Find a set of candidate matchings and compute a starting transform for each matching.
3. For each starting transform, optimize the matching and transform with respect to a scoring function. The result is the transform corresponding to the optimal score.

3.1. Generating starting transforms

Given two point sets $P = \{\mathbf{p}_i\}$ and $Q = \{\mathbf{q}_i\}, \mathbf{p}_i, \mathbf{q}_i \in \mathbb{R}^2$, the first step to find the optimal matching and the corresponding transform, is to compute a set of candidate matchings.

A matching M defined on two finite sets P and Q is a bijective function $M : \tilde{P} \rightarrow \tilde{Q}, \tilde{P} \subseteq P, \tilde{Q} \subseteq Q$. M can be written as a set M^* of pairs from $P \times Q$ with $(p, q) \in M^* \Leftrightarrow M(p) = q$. The number of pairs $|M^*|$ will be referred to as the size of M.

Candidate matchings are found by looking at the Euclidean distances between point pairs. The goal is to find subsets of points in P and Q that have the same spatial pattern, i.e. similar mutual distances. In particular, the goal is to find a matching $M : \tilde{P} \rightarrow \tilde{Q}$ such that the distance between each point pair $(\mathbf{p}', \mathbf{p}'')$ in $\tilde{P}$ and the distance of the corresponding point pair $(\mathbf{q}', \mathbf{q}'')$ in $\tilde{Q}$ differ by no more than a given distance threshold d:

$$\big| \|(\mathbf{p}' - \mathbf{p}'')\| - \|(\mathbf{q}' - \mathbf{q}'')\| \big| \leq d. \tag{1}$$

If this condition holds for each point pair in $\tilde{P}$, we say that M is a d-bounded matching. The points in both sets then have a similar positional pattern and M is thus considered a candidate matching.

The parameter d is data-dependent. It should be chosen small for tracings obtained from tissue that was little deformed and larger for more deformed specimens.

Usually, multiple d-bounded matchings exist. In order to find them all, we use a method based on clique detection in the *distance compatibility graph* (DCG). The vertices of this graph consist of all $(\mathbf{p}', \mathbf{q}')$ pairs. Thus, there are $|P| \cdot |Q|$ vertices. Two vertices $(\mathbf{p}', \mathbf{q}')$ and $(\mathbf{p}'', \mathbf{q}'')$ are connected by an edge, if the matching M given by $M^* = \{(\mathbf{p}', \mathbf{q}'), (\mathbf{p}'', \mathbf{q}'')\}$ is d-bounded (see Eq. 1). A clique in a graph is a subset of vertices that induce a complete subgraph, i.e. each vertex in the subset is connected to all other vertices in the subset. A clique is maximal when it is not part of any larger complete sub-graph. In the following, we use clique to refer to a maximal clique. We find all cliques (and thus all candidate matchings) using the Bron-Kerbosch [12] algorithm.

For each candidate matching we compute a transform that minimizes the *root mean square distance* (*rmsd*) of the matched points. The *rmsd* of two point sets P and Q with respect to a matching M and a rigid body transform T is defined as:

$$rmsd(P, Q, M, T) = \sqrt{\frac{\sum_{(\mathbf{p},\mathbf{q}) \in M^*} \|\mathbf{p} - T(\mathbf{q})\|^2}{|M^*|}} \tag{2}$$

This transform is called the *matching transform* of M and can be computed in $\mathcal{O}(|M^*|)$ time, using the algorithm described in [13]. The matching transforms of all candidate matchings are the starting transforms serving as the input for the following optimization step.

3.2. Optimizing matching and rigid transform

In this step, we find the optimal matching M and transform T with respect to a scoring function $score$:

$$score(P, Q, M, T) = \frac{|M^*|}{min(|P|, |Q|)} \cdot e^{-\alpha \cdot rmsd(P,Q,M,T)} \tag{3}$$

The scoring function measures the number of matched points and the root mean square distance of the matching points after transformation. Its range is $[0, 1]$, where 1 corresponds to the best score. The optimal score is achieved when all points have been matched and the transform maps the corresponding points exactly onto each other. The parameter $\alpha \in [0, 1]$ weighs the contradicting goals of a large match (α small) versus small positional error (α large). In an iterative algorithm, we maximize $score$ alternatingly with respect to M and to T, see Algorithm 1. M is optimized by a simple and fast greedy approach [14], see Algorithm 2. The scoring function ensures that no point pairs will be assigned to M that are too far apart. T is optimized by computing the *matching transform* for a given M. The $score$-function is maximized once for each starting transform T_i^{init}, using the respective T_i^{init} as initialization. The final result of the alignment algorithm is the matching and transform corresponding to the highest $score$ across all computed $score$ maxima.

3.3. Uniform scaling for deformed data

For data that suffered from severe deformations, an additional uniform scaling can lead to a better alignment. The scaling factor s for the matching transform is computed as follows:

$$s = \frac{\sum_{i=1}^{|M*|} \mathbf{p}_i' \cdot \mathbf{q}_i'}{\sum_{i=1}^{|M*|} \mathbf{q}_i' \cdot \mathbf{q}_i'}, \quad \mathbf{p}_i' = \mathbf{p}_i - \mathbf{c}_{P'}, \quad \mathbf{q}_i' = R(\mathbf{q}_i - \mathbf{c}_{Q'}), \tag{4}$$

where $\mathbf{c}_{P'}$ and $\mathbf{c}_{Q'}$ are the average point positions of the matched sets P' and Q' respectively. R is the rotation computed as in the rigid case. However, this approach only works if a suitable clique can be found. The points in such a clique are usually spatially concentrated, as the effect of the global scaling on their absolute mutual distances is limited (see e.g. Fig. 2, middle).

```
Algorithm 1: Optimize matching and transform
Input: Point sets $P$ and $Q$, set of starting transforms $\{T_i^{init}\}$
Output: Near-optimal matching $M_{opt}$, transform $T_{opt}$
foreach $T_i^{init}$ do
    $s_{new} \leftarrow 0$
    $T_i \leftarrow T_i^{init}$
    repeat
        $s_i \leftarrow s_{new}$
        $M_i \leftarrow greedyPointMatching(P, Q, T_i)$
        $T_i \leftarrow computeTransform(P, Q, M_i)$
        $s_{new} \leftarrow score(P, Q, M_i, T_i)$
    until $s_{new} \leq s_i$
end
$(M_{opt}, T_{opt}) \leftarrow \arg\max_i score(P, Q; M_i; T_i)$
```

```
Algorithm 2: Greedy Point Matching
Input: Finite point sets $P$ and $Q$, transform $T$
Output: Near-optimal matching $M_{opt}$
$M_0^* \leftarrow \emptyset$
$P' \leftarrow P, Q' \leftarrow Q$
$k \leftarrow 0$
while $\exists p \in P', q \in Q'$ do
    $(p, q) \leftarrow \arg\min_{p \in P', q \in Q'} \|\mathbf{p} - T(\mathbf{q})\|$
    $M_{k+1}^* \leftarrow M_k^* \cup \{(p, q)\}$
    $P' \leftarrow P' \setminus \{p\}$
    $Q' \leftarrow Q' \setminus \{q\}$
    $k \leftarrow k + 1$
end
$M_{opt} \leftarrow \arg\max_k \mathrm{score}(P, Q; M_k^*; T)$
```

3.4. Increasing performance

As the computation of the starting transforms is independent of the point matching optimization, we can generate a set of starting transforms using different point sets and use these for the optimization step. When these point sets are smaller than the original, computing time for clique detection will be reduced. In the case of microtubuli, for example, one could select the endpoints of segments which are close to parallel to the z-axis (and thus have a good chance of having a matching counterpart) for the computation of the starting transforms, resulting in much smaller point sets and a significant speed-up. Of course one can also use the reduced point set for the optimization step, as we do in section 4.2. As long as enough reliable points remain, a correct alignment can be achieved much faster.

A second efficiency improvement is a reduction of the number of starting transforms to compute the optimal matching for, by retaining only transforms corresponding to cliques larger than a certain minimal size c (we set $c = max(2, 0.3 \cdot min(|P|, |Q|))$).

4. RESULTS

4.1. Alignment of neuronal processes

The automatic alignment algorithm was applied to a stack of 30 slices containing segments of a neuron (axons) obtained from bright-field microscopy (see Fig. 2, left). The number of endpoints to match ranged from 3–101.

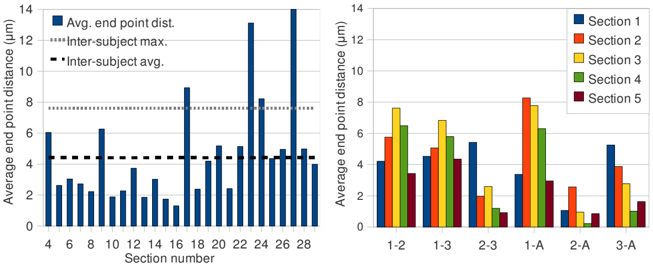

Fig. 3. Left: Average distance between manually and automatically aligned end points ($\overline{\Delta}_{27} = 25.2\mu m$). Right: Average end point distance between the alignments of three subjects (1, 2, 3) and our automatic method (A). The average and maximum value of all subject-subject combinations are indicated by a line in the left figure.

Almost all slices could be aligned, except two at the top and one at the bottom, where there were too few points (3–6) to produce a valid result. A good match, and thus a good alignment, is characterized by a matching fraction that is large enough to rule out randomness and by the absence of mismatches. The results obtained with parameter values $d = 10\mu m$ and $\alpha = 0.25$ fulfilled these criteria, as they resulted in a match size of 32–100% with a minimum of 5 points and no visually apparent mismatches. The computation time ranges from 0.02–8.5s.

The successfully aligned sections (4–29) were compared to their manually aligned counterpart. The manual transform T_M and automatic transform T_A were computed for each slice with respect to the untransformed predecessor section. Then the average distance $\overline{\Delta}$ between the segment end points $\mathbf{q}_i$ that served as input for the alignment was computed: $\overline{\Delta} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \|T_M(\mathbf{q}_i) - T_A(\mathbf{q}_i)\|$. The results are displayed in Fig. 3 (left).

For comparison we assessed the inter-subject differences in manual alignment. A 6-slice data set was aligned automatically and by three individuals. The average point distance $\overline{\Delta}$ was computed as above for all combinations of manual and automatic alignments (see Fig. 3, right). After visual comparison of the outlier sections 23 and 27, the expert who carried out the manual alignment, confirmed that the automatic alignment was at least as good. From these results we conclude that the automatic alignment results in similar quality as the manual alignment.

4.2. Alignment of microtubuli

We also applied our algorithm to the alignment of a stack of 9 slices containing microtubuli, obtained by manual tracing of electron tomography data using the IMOD [15] software. We failed to manually align the lines as there are many segments (∼700 per section) but no prominent, clearly identifiable features.

For the automatic alignment the number of points was reduced using the angle-based selection (see Sec. 3.4) to increase performance. All lines having an angle smaller than 70 degrees with the xy-plane were ignored, reducing the average point set size from 436 to 61. Besides, values $d = 40nm$ and $\alpha = 0.1$ were chosen.

Our method was able to align six out of eight slice pairs within a few seconds (0.10–6.7s). Visual inspection showed that the matchings are generally correct and of sufficient size (18–66% of the points, with a minimum of 10) to obtain a reliable transformation.

However, an additional number of matching point pairs that were not reported by the automatic algorithm could usually be deter-

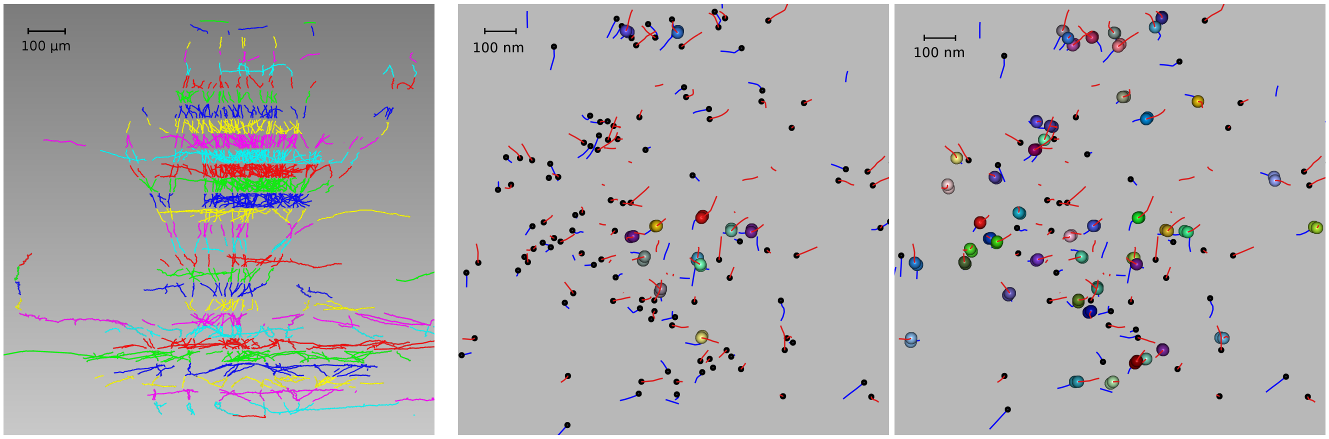

Fig. 2. Left: Aligned stack containing neuron fragments (segments colored by section). Middle: Rigidly aligned slice pair containing microtubuli. The matched points are clustered in the center of the data set (colored point pairs). Further away from the center, there are many unmatched points (black). Many of those could be matched when using an additional scaling (right).

mined by visual inspection (see Fig. 2, middle). Firstly, this can be explained by the defensive choice for α resulting in almost no false pairings, but also in a reduced number of total pairings. Secondly, the results indicate a non-rigid deformation of the sections. To test this, we added a uniform scaling s as an additional degree of freedom (see section 3.3). The resulting scaling factors ranged between 0.5–10%, and particularly for the more severely deformed sections resulted in significantly better alignments (see Fig. 2, right).

Two slice pairs could not be aligned for reasons that are not yet understood. Analysis is difficult as manual alignment is not feasible.

5. CONCLUSION

We presented an automatic method for the alignment of data stacks containing filamentous structures. It uses a point-matching approach to find the optimal rigid transform (with optional uniform scaling) for each section. The application to neuron data showed that it is fast and produces accurate results. Also microtubuli data could be aligned, something that is infeasible to do manually.

6. ACKNOWLEDGEMENTS

We would like to thank A. Borst and B. Sakmann of the Max Planck Institute (MPI) of Neurobiology, Martinsried, Germany, for financial support. We also would like to thank A. Hyman and Q. de Robillard of the MPI of Molecular Cell Biology and Genetics, Dresden, for providing the electron tomograms and the tracings of microtubuli. Finally, we would like to thank D. Baum and M. Koppitz (ZIB).

7. REFERENCES

[1] M. Oberlaender et al., “Transmitted light brightfield mosaic microscopy for three-dimensional tracing of single neuron morphology,” *J. Biomed. Opt.*, vol. 12, no. 6, pp. 1–19, 2007.

[2] P. Bajcsy et al., “Three-dimensional volume reconstruction of extracellular matrix proteins in uveal melanoma from fluorescent confocal laser scanning microscope images,” *J Microsc*, vol. 221, no. 1, pp. 30–45, 2006.

[3] J. Richard McIntosh, Ed., *Cellular Electron Microscopy (Methods in Cell Biology, vol. 79)*, Elsevier, 2007.

[4] B. Zitová and J. Flusser, “Image registration methods: a survey,” *Image and Vision Computing*, vol. 21, no. 11, pp. 977 – 1000, 2003.

[5] A. Rangarajan, H. Chui, and F. L. Bookstein, “The Softassign Procrustes matching algorithm,” in *Proc. IPMI*, 1997, pp. 29–42.

[6] H. Chui and A. Rangarajan, “A new point matching algorithm for non-rigid registration,” *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, 2003.

[7] B. Luo, *Statistical Methods for Point Pattern Matching*, Ph.D. thesis, University of York, 2001.

[8] S.-C. Lee and P. Bajcsy, “Trajectory fusion for three-dimensional volume reconstruction,” *Comput. Vis. Image Underst.*, vol. 110, no. 1, pp. 19–31, 2008.

[9] D. Baum, *A Point-based Algorithm for Multiple 3D Surface Alignment of Drug-sized Molecules*, Ph.D. thesis, Freie Universität Berlin, 2007.

[10] T. Akutsu, “Protein structure alignment using dynamic programming and iterative improvement,” *IEICE Trans. Info. & Syst.*, vol. E79-D, pp. 1629–36, 1996.

[11] S. Kirchner, “An FPTAS for computing the similarity of three-dimensional point sets,” *Int. J. Comput. Geometry Appl.*, vol. 17, no. 2, pp. 161–174, 2007.

[12] C. Bron and J. Kerbosch, “Algorithm 457: Finding all cliques of an undirected graph,” *Commun ACM*, vol. 16, no. 9, pp. 575–577, 1973.

[13] W. Kabsch, “A discussion of the solution for the best rotation to relate two sets of vectors,” *Acta Crystallogr A*, vol. 34, no. 5, pp. 827–828, 1978.

[14] S. Kirchner, “Ein Approximationsalgorithmus zur Berechnung der Ähnlichkeit dreidimensionaler Punkt-mengen,” Diplomarbeit, Humboldt-Universität zu Berlin, 2003.

[15] J. R. Kremer, D. N. Mastronarde, and J. R. McIntosh, “Computer visualization of three-dimensional image data using IMOD,” *J. Struct. Biol.*, vol. 116, pp. 71–76, 1996.