

Reconstructing Evolving Tree Structures in Time Lapse Sequences

Przemysław Głowacki¹ * Miguel Amável Pinheiro² † Engin Türetken¹ Raphael Sznitman¹
Daniel Lebrecht³ Jan Kybic² Anthony Holtmaat³ Pascal Fua¹

¹Computer Vision Laboratory (EPFL), CH-1015 Lausanne, Switzerland
{przemyslaw.glowacki, engin.turetken, raphael.sznitman, pascal.fua}@epfl.ch

²CMP, Dept. of Cybernetics, Faculty of Elec. Eng., Czech Technical University in Prague
amavemig@cmp.felk.cvut.cz, kybic@fel.cvut.cz

³Department of Basic Neurosciences, University of Geneva, 1211 Geneva, Switzerland
{daniel.lebrecht, anthony.holtmaat}@unige.ch

Abstract

We propose an approach to reconstructing tree structures that evolve over time in 2D images and 3D image stacks such as neuronal axons or plant branches. Instead of reconstructing structures in each image independently, we do so for all images simultaneously to take advantage of temporal-consistency constraints.

We show that this problem can be formulated as a Quadratic Mixed Integer Program and solved efficiently. The outcome of our approach is a framework that provides substantial improvements in reconstructions over traditional single time-instance formulations. Furthermore, an added benefit of our approach is the ability to automatically detect places where significant changes have occurred over time, which is challenging when considering large amounts of data.

1. Introduction

Reliably reconstructing networks of curvilinear structures from images remains an open computer vision problem. So far, it has mostly been addressed in terms of modeling structures that have been captured at a specific moment in time. However, these networks, be they made of axons and dendrites seen *in vivo* in optical microscopy image stacks [10], blood vessels in retinal-scans [16], plant roots in time-lapse imagery, or roads in aerial images taken at long intervals, evolve over time. Modeling this evolution is of great value in many scientific domains to help understand underlying processes and analyzing the effects of bi-

*This work was supported in part by the Swiss National Science Foundation.

†This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS12/190/OHK3/3T/13, the Czech Science Foundation project P202/11/0111 and the Fundação para a Ciência e Tecnologia grant SFRH/BD/77134/2011.

ological [15] or geographic environmental conditions [20].

In this paper, we propose an approach to reconstructing evolving tree structures simultaneously in all images. In this way, we can enforce temporal consistency over stable parts of the structure and reliably detect changes elsewhere. This is in contrast to recovering the relevant structures in each image individually and only then comparing them, which we will show to be far less effective.

To this end, we first process individual images to find pixels or voxels that are very likely to be on the centerlines of linear structures. Finding tree structures in individual images could then be achieved by solving a Quadratic Mixed Integer Program (QMIP) to minimize an appropriate objective function [24]. Instead, we find centerline points that correspond to identical features across time instances by means of a Gaussian Process Regression (GPR) model [21] and connect these temporal correspondences by *temporal edges*. Combining both types of edges yields a *spatio-temporal* graph that lets us incorporate into our objective function terms that enforce temporal consistency. Conveniently, this optimization problem remains a QMIP that can be solved efficiently.

Our contribution is therefore a novel approach to modeling trees over several images simultaneously while enforcing temporal consistency. Not only is this more reliable than doing so over individual images but has the added benefit of making it easy to spot the regions that have significantly changed, which is tedious and hard to do for human operators. We demonstrate the power of our approach on a time-lapse sequence of a growing bean plant and on sequences of *in vivo* two-photon micrographs of neuronal networks.

2. Related Work

For most automatic reconstruction techniques of tree-like structures, the process begins by estimating a local

measure of *tubularity*, *i.e.* the likelihood that a point sits along the centerline of a tubular structure [5, 12, 17]. Matched filters [1, 28], Hessian and Oriented Flux functionals [7, 13, 14, 22], and classification scores derived from steerable filter responses [8, 11] have all been used for this purpose.

These tubular measures are then used within a search or optimization framework to reconstruct tree structures image by image. In this context, the search techniques come in one of two forms, either local or global. Local methods reconstruct the tree structure piece by piece in a greedy fashion, making them extremely efficient [1, 3, 26] but less robust to image noise and prone to errors when there are large gaps separating filaments. Conversely, global methods optimize the entire tree in one shot making them more computationally demanding but also more robust. This typically involves connecting high tubularity points to form a weighted graph and then finding a tree within that graph by optimizing an objective function. This last step can be done using Minimum Spanning Trees (MST) [6, 26, 28], Shortest Path Trees (SPT) [19], k-Minimum Spanning Trees (k-MST) [25], and Quadratic Mixed Integer Programming (QMIP) [24].

Yet, by and large, existing strategies reconstruct structures one instance at a time. As we will show in our experiments, using temporal information to enforce time-consistency can significantly improve performance. This is well known in a number of applications [4, 15, 18] but has not yet to be exploited for the purpose of tree structure reconstruction.

3. Approach

For many tree structures that evolve over time, significant changes from one frame to the next tend to be fairly localized, while the general topology and geometry remain relatively stable up to minor local deformations. Consider, for example, a real-world tree whose branches are growing over time. In images taken at sufficiently long time intervals, there may be significant changes at the tips of existing branches while the rest remains largely unchanged. The same principle applies in the case of the neuronal network of Fig. 1 captured *in vivo* at intervals of a week. Most of the structure is preserved over time, except for a few branches that have either grown to form new connections, retracted or moved to new positions. To exploit the overall consistency while allowing some degree of change, we propose the following approach.

Given N , D -dimensional images $\mathcal{I} = \{I^n\}_{n=1}^N$ taken in sequence and showing an evolving tree structure, our goal is to reconstruct a tree in each individual image such that they collectively form a temporally consistent sequence. By this, we mean that branches do not appear or disappear randomly. As a starting point, we find corresponding points

across images and use them as nodes of a graph whose edges can either connect to nodes within the same image or to other images. As in [24], the final set of trees can then be reconstructed by solving a QMIP problem.

We now briefly outline how to reconstruct trees using the QMIP formulation in single images and then introduce our own framework.

3.1. Reconstruction in a Single Image

For an image I , the procedure of [24] starts by computing a tubularity at all image locations $\mathbf{x}_i \in I$. It then selects regularly spaced local-maxima of tubularity and connects them to their neighbors by high-probability paths to which are assigned image-based quality scores. This produces a *spatial graph*, $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, whose nodes $\mathcal{X} = \{\mathbf{x}_i\}$ are the selected local maxima and whose *spatial edges* $\mathcal{E} = \{e_{ij} = (\mathbf{x}_i, \mathbf{x}_j)\}$ represent connections between nodes \mathbf{x}_i and \mathbf{x}_j . Then an image-based probability p_{ijk} is associated to each *edge pair* connected by a common node, *e.g.* edges e_{ij} and e_{jk} . This probability corresponds to the likelihood that the edge pair is indeed part of a larger curvilinear structure.

Given this *curvilinear graph*, the final tree or set of trees can then be reconstructed by selecting an appropriate set of spatial edges that minimize a function of the p_{ijk} [24]. That is, the produced solution is a set of *directed* edges that stem from a root node and which together form a tree.

3.2. Reconstruction in all Images Simultaneously

Repeating the above procedure for each image I^n would yield N distinct trees that would be difficult to compare to other trees, as it is unlikely for their nodes to be at the same locations in different images. To avoid this problem and to enforce temporal consistency constraints, we modify the framework in two key ways:

First, we find temporally consistent nodes \mathbf{x}_i^n in all images by looking for local-maxima of tubularity in one image and then finding corresponding high-tubularity points in the others. This lets us create *temporal edges* $e_{ij}^{n_1, n_2}$ between node $\mathbf{x}_i^{n_1}$ found in I^{n_1} and its matched node $\mathbf{x}_j^{n_2}$ in I^{n_2} .

Second, we build a *spatio-temporal* graph whose edges are both the spatial edges as in [24] and temporal edges that connect nodes from one individual image to another. In such a graph, minimizing an objective function that only considers the spatial edges, as described in the previous subsection, would yield the same result as before. However, we can use the temporal edges to add terms favoring edges persistent between time instances, thus enforcing time consistency. Minimizing this extended objective function can still be expressed as a QMIP. Our approach therefore goes through the following steps:

1. Find graph nodes in individual images as tubularity

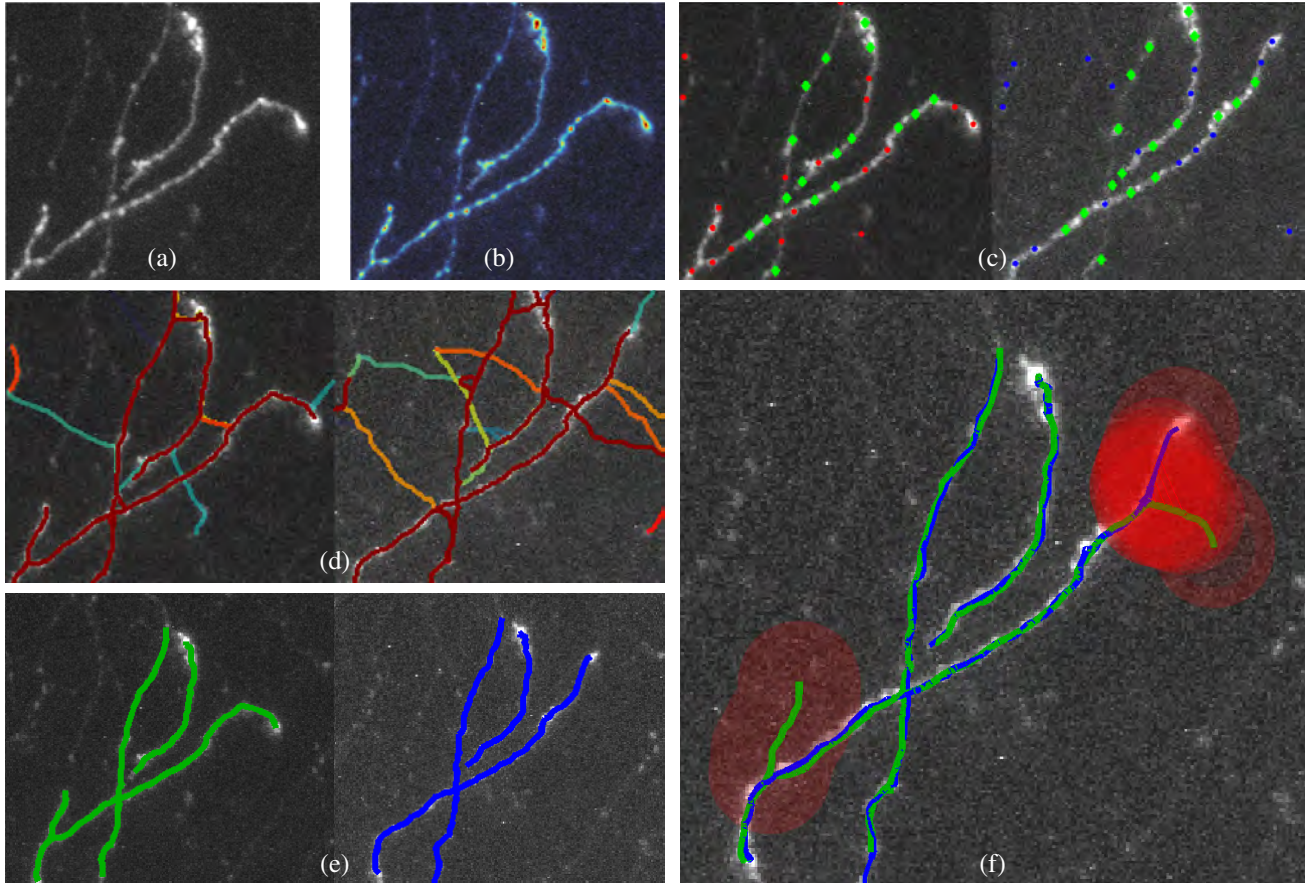


Figure 1. Key steps of the algorithm, best viewed in color. (a) Maximum intensity projection of one of three *in vivo* image-stacks of a neural network taken at one week intervals. (b) Corresponding tubularity image. (c) Maxima of tubularity selected as graph nodes in two different stacks. Those shown in green have been determined to correspond to the same location in both, while those in red or blue appear in only one. (d) Connecting neighboring nodes by high-tubularity paths produces a spatial graph in each image. High-quality paths are shown as red while low quality ones appear as blue. (e) Connecting the corresponding vertices across images turns the spatial graphs into a single spatio-temporal one and solving the corresponding QMIP problem yields two temporally consistent trees. (f) The red tree from the first image can be deformed and superposed on the blue tree in the second one, making the changes highlighted in red easy to detect.

maxima and corresponding nodes, if any, in other images, as in Fig. 1(b-c).

2. Build a spatio-temporal graph such as the one depicted in Fig. 1(d) by linking nodes both within images when they are close enough and across images when they match.
3. Solve an extended QMIP problem to find a set of temporally-consistent trees, such as those of Fig. 1(e).
4. Align these trees spatially to identify places where substantial changes have occurred, as can be seen in Fig. 1(f).

In the following two sections, we first describe the construction of our spatio-temporal graphs in more detail. We then define our QMIP problem and the corresponding objective function.

4. Building Spatio-Temporal Graphs

The first step in building our spatio-temporal graph is to find corresponding nodes across images, such as those shown in Fig. 1(c). As discussed above, we assume that there may be some non-linear deformation from one image to the next but that it is smooth.

Finding an Initial Set of Correspondences We first use the Optimally Oriented Flux [14] filter to compute a tubularity measure in each image independently.

Then, for $m = 1, \dots, M$ iterations, we find the point \mathbf{x}_m^n that maximizes tubularity across all images, where n refers to the image in which it was found. Then for each of the remaining images $I^{\bar{n}} \in \mathcal{I} \setminus I^n$, we compute the Normalized Cross Correlation (NCC) score of a square or cubic patch centered on a point \mathbf{x}_m^n and a neighbourhood of locations around $\mathbf{x}_m^{\bar{n}}$. Within each evaluated neighbourhood,

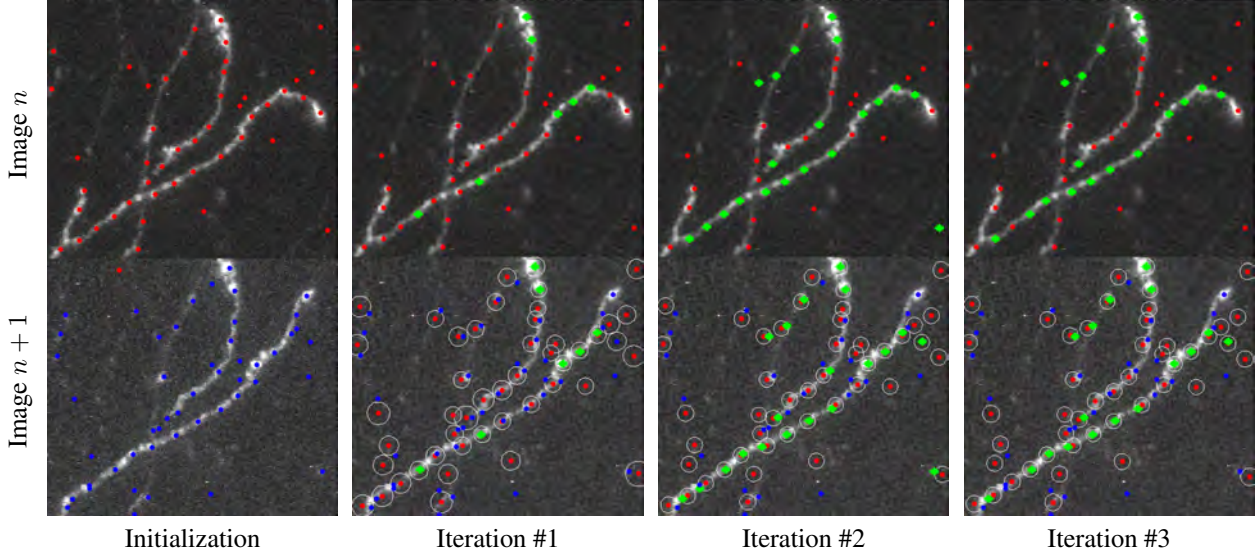


Figure 2. Iterating until a stable correspondence set has been found. (Initialization) A set of corresponding points with possible inconsistencies in the transformation model is found in each image using high-tubularity locations and NCC. (Iteration #1) A set of corresponding points (shown in green) with the highest tubularity likelihoods has been selected, which are then used to instantiate a GPR that maps the remaining red points in image n to the red locations in image $n + 1$. The blue points in image $n + 1$ that are close enough to these red locations and correlate well with the original red points in image n are taken to form new correspondences. (Iterations #2 and #3) They are added to the set of correspondences, shown in green. The process is then repeated.

we associate the location $\mathbf{x}_m^{\bar{n}}$ with the maximum computed NCC score provided it is above a given minimum threshold. From this set, we keep all the consecutive pairs of points $\{\mathbf{x}_m^{n'} \leftrightarrow \mathbf{x}_m^{n'+1}\}_{1 \leq n' \leq N-1}$ as correspondences, as illustrated by the green points of Fig. 1(c). Once computed, the tubularity is set to zero in both the neighborhood of \mathbf{x}_m^n and that of the found corresponding points. The procedure is then iterated until the tubularity of the selected point \mathbf{x}_m^n is below a certain value.

Enforcing Geometric Consistency The procedure described above relies solely on the NCC scores computed locally and does not guarantee that the displacements of neighboring points are spatially consistent with each other. To enforce this and remove potential mismatches, we use a Gaussian Processes Regression (GPR) [21] to remove correspondences that are not consistent with a non-linear but locally smooth deformation model.

Hence, to find a geometrically consistent set of correspondences \mathcal{S}_n between images I^n and I^{n+1} , we first select from our correspondences a set $\mathcal{S}_n^0 = \{\mathbf{x}_l^n \leftrightarrow \mathbf{x}_l^{n+1}\}_{1 \leq l \leq L}$ of the L points with the highest average local tubularity. In the example of Fig. 2 (Iteration #1), the selected \mathbf{x}_l^n points are shown in green. We treat \mathcal{S}_n^0 as being a reliable set and use the GPR to estimate the mean and covariance of the location of a point \mathbf{x}^n in I^{n+1} . This can be computed as

$$\begin{aligned} m_{\mathcal{S}_n^0}(\mathbf{x}^n) &= \mathbf{k}' \Gamma_{\mathcal{S}_n^0}^{-1} \mathbf{X}_{\mathcal{S}_n^0}^{n+1}, \\ \sigma_{\mathcal{S}_n^0}^2(\mathbf{x}^n) &= k(\mathbf{x}^n, \mathbf{x}^n) + \beta^{-1} - \mathbf{k}' \Gamma_{\mathcal{S}_n^0}^{-1} \mathbf{k}, \end{aligned} \quad (1)$$

where k is a kernel function that implicitly defines a mapping composed of an affine and a non-linear transformation as in [23, 27], β^{-1} is a measurement noise variance, $\Gamma_{\mathcal{S}_n^0}$ is the $L \times L$ symmetric matrix with elements $\Gamma_{i,j} = k(\mathbf{x}_i^n, \mathbf{x}_j^n) + \beta^{-1} \delta_{i,j}$, \mathbf{k} is the vector $[k(\mathbf{x}_1^n, \mathbf{x}^n), \dots, k(\mathbf{x}_L^n, \mathbf{x}^n)]^T$ and $\mathbf{X}_{\mathcal{S}_n^0}^{n+1}$ is the $L \times D$ matrix $[\mathbf{x}_1^{n+1}, \dots, \mathbf{x}_L^{n+1}]^T$.

We then add all correspondences that are consistent with this GPR to \mathcal{S}_n^0 , which is determined when the Mahalanobis distance between corresponding points \mathbf{x}^{n+1} and $m_{\mathcal{S}_n^0}(\mathbf{x}^n)$ is sufficiently small. This gives us an augmented set of correspondences \mathcal{S}_n^1 , such as the one depicted by Fig. 2 (Iteration #2). We then repeat the process using \mathcal{S}_n^1 to compute the regression of Eq. 1 and iterate until the set stabilizes, typically after 4 to 5 iterations, as shown in Fig. 2 (Iteration #3).

This is performed for each consecutive image pair, which yields sets of points in each image $\mathcal{X}^n = \{\mathbf{x}_i^n\}$ and sets of geometrically consistent correspondences \mathcal{S}_n across consecutive images.

Building the Graph We treat points in all the \mathcal{X}^n as nodes of our graph and create two kinds of edges. As in the single-image case of Section 3.1, the *spatial edges* $\mathcal{E}_s^n = \{\mathbf{e}_{ij}^n = (\mathbf{x}_i^n, \mathbf{x}_j^n)\}$ correspond to edges connecting points within I^n and consecutive pairs of such edges are assigned an image-based probability of being part of the final curvilinear structure. To these, we add *temporal edges* $\mathcal{E}_t^n = \{\mathbf{e}_{ij}^{n,n+1} = (\mathbf{x}_i^n, \mathbf{x}_j^{n+1}) \mid (\mathbf{x}_i^n \leftrightarrow \mathbf{x}_j^{n+1}) \in \mathcal{S}_n\}$ that

connect nodes in I^n and I^{n+1} that belong to the set \mathcal{S}_n of geometrically consistent correspondences.

5. Finding Temporally Consistent Trees

Given a spatio-temporal graph $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, where $\mathcal{X} = \{\bigcup_{n=1}^N \mathcal{X}^n\}$ and $\mathcal{E} = \mathcal{E}_s \cup \mathcal{E}_t = \{\bigcup_{n=1}^N \mathcal{E}_s^n\} \cup \{\bigcup_{n=1}^{N-1} \mathcal{E}_t^n\}$ such as the one discussed in the previous section, our goal now is to find a subgraph forming a set of trees that evolve consistently over time. For every image in the sequence, the locations of the tree roots are provided by an operator and are added to the set of graph nodes. An additional imaginary root \mathbf{x}_r is created and connected to all these root nodes in all time instances. This way, reconstructing the trees in all images can be achieved by finding the most likely arborescence rooted in \mathbf{x}_r .

5.1. Objective Function

Reconstructing the trees of interest means making a decision as to whether each edge of the graph \mathcal{G} should be part of the solution or not. To this end, we take Bayesian point of view as in [24]. Let $Y_{ij} \in \{0, 1\}$ be a binary random variable denoting the presence or absence of the edge \mathbf{e}_{ij} in the final solution and Y be the set of all Y_{ij} variables. Our goal is to infer the most likely tree Y .

To obtain the most likely Y while enforcing temporal consistency between reconstructions across time, we introduce a constant q that denotes the edge persistence probability. That is, for a given pair of edges $(\mathbf{e}_{ij}^n, \mathbf{e}_{kl}^{n+1})$, we assume that the probability of both edges being part, or not, of the final solution is equal to q . Conversely, the probability that one of the edges is part of the solution while the other is not is equal to $1 - q$. And let us therefore denote $\bar{\mathcal{E}}_t = \{(\mathbf{e}_{ij}^n, \mathbf{e}_{kl}^{n+1}) | \mathbf{e}_{ij}^n, \mathbf{e}_{kl}^{n+1} \in \mathcal{E}_s \wedge \mathbf{e}_{ik}^{n,n+1}, \mathbf{e}_{jl}^{n,n+1} \in \mathcal{E}_t\}$ be the set of all pairs of spacial edges in consecutive time frames whose endpoints are connected with temporal edges.

With this, describing the posterior distribution of Y given the spatial edges \mathcal{E}_s and the temporal edges \mathcal{E}_t can then be expressed as

$$P(Y = y | \mathcal{I}, \mathcal{X}, \mathcal{E}_s, \mathcal{E}_t) \propto P(\mathcal{I}, \mathcal{X}, \mathcal{E}_s | Y = y) P(Y = y | \mathcal{E}_t),$$

assuming that the image data and the spatial edges are conditionally independent of the temporal edges given Y . Following similar steps as in [24], computing the optimal tree involves solving the *maximum a posteriori* problem,

$$\begin{aligned} y^* &= \arg \min_{y \in \mathcal{Y}} P(\mathcal{I}, \mathcal{X}, \mathcal{E}_s | Y = y) P(Y = y | \mathcal{E}_t), \quad (2) \\ &= \arg \min_{y \in \mathcal{Y}} \sum_{\mathbf{e}_{ij}^n, \mathbf{e}_{jk}^n \in \mathcal{E}_s} w_{ijk} y_{ij} y_{jk} \\ &\quad + \sum_{(\mathbf{e}_{ij}^n, \mathbf{e}_{kl}^{n+1}) \in \bar{\mathcal{E}}_t} w_p (2y_{ij} y_{kl} - y_{ij} - y_{kl}), \quad (3) \end{aligned}$$

where $w_{ijk} = -\log \frac{p_{ijk}}{1-p_{ijk}}$, $w_p = -\log \frac{q}{1-q}$, p_{ijk} is the probability that the edge pair $(\mathbf{e}_{ij}^n, \mathbf{e}_{jk}^n)$ is a part of a tubular structure and \mathcal{Y} is the set of all feasible trees with root \mathbf{x}_r . The complete derivation of Eq. (3) can be found in the appendix.

Note that the temporal constant $0.5 \leq q < 1$ allows flexibility in the amount of time consistency desired across time instances, *i.e.* higher values enforce more consistent results. In the special case where $q = 0.5$ the persistence weight w_p is equal to 0 and the problem is reduced to that of [24].

5.2. Finding the Optimal Tree

To find a tree that minimizes the objective function defined above, we solve the quadratic mixed integer program (QMIP) as described in [24] using a max-flow min-cut formulation of the minimum arborescence problem using [9]. Note that in this formulation, the input graph must have directed edges in order to compute the flow of a given solution. Hence, as in [24], we treat each possible spatial edge pair $(\mathbf{e}_{ij}^n, \mathbf{e}_{jk}^n)$ with associated weight p_{ijk} , as a directed path and also give the opposite directed edge pair $(\mathbf{e}_{kj}^n, \mathbf{e}_{ji}^n)$ the weight p_{ijk} . As a result, the solution is a directed tree with root node \mathbf{x}_r , connected to a sub-tree in each image I^n , as depicted in Fig. 1(e).

5.3. Fine Alignment and Change Detection

Having both the set of temporally corresponding vertices \mathcal{S}_n and the reconstructed trees for each image, we can now construct a large set of correspondences between trees in I^n and I^{n+1} . To do this, we use the method of [23], which uses a GPR to assign correspondences between separate tree instances, in a similar way as described in Section 4, but initializing the set of correspondences with the previously found assignments \mathcal{S}_n . In Fig. 1(f) we show sets of locally concentrated outliers, whose projected locations differ greatly from the tree location.

6. Experiments and Results

We evaluate our method on 3D 2-photon images of axons in the brain of a mouse, and on 2D time-lapse images of a growing runner bean. We use the DIADEM metric [2] to quantify our results.

Throughout the experiments, we suppress all tubularity values below 30% of the highest observed value, and set the initial number of values in \mathcal{S}_n to be $L = 10$. In this section, we use $q = 0.75$ as our edge persistence probability and have observed that the results are very similar for q in the range 0.65 to 0.8.

6.1. Change Detection in Plant Growing Time Lapse

We first tested our algorithm on a simple time lapse sequence of a growing runner bean. Monitoring the growth

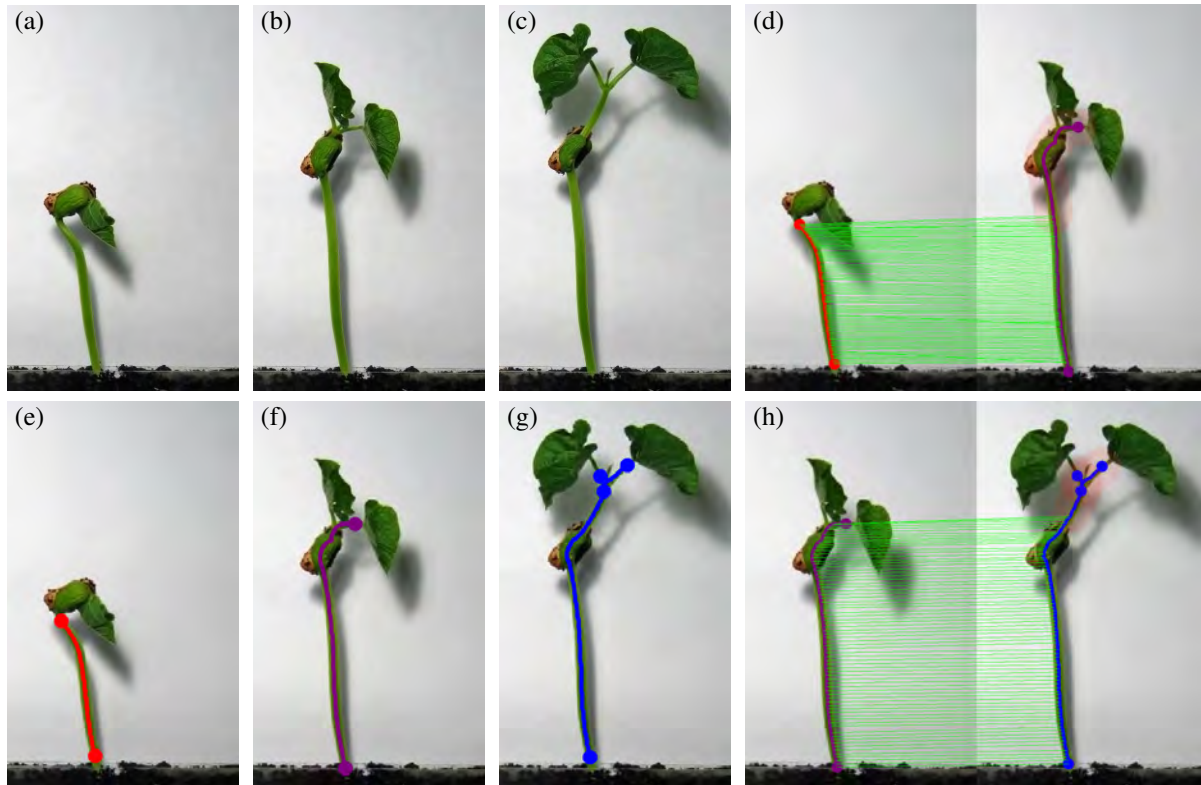


Figure 3. Results for the reconstruction and automatic change detection for the growing runner bean images. (a), (b) and (c) are the original images and (e), (f) and (g) present the reconstructed trees. (d) and (h) depict the result of the automatic change detection from (e) to (f) and from (f) to (g), respectively.

of a plant has many uses. These include testing different environmental conditions, getting to understand the influence of specific pesticides or other agricultural products, or evaluating models of plant development and growth [20].

Here, we trained the path classifier using 20000 positive samples and 20000 negative samples, extracted from six images from the sequence. These training images were selected at random and we manually traced the tree in each one to produce positive samples.

Fig. 3 depicts the results. The branch structure is correctly reconstructed and the important topological changes are automatically found. In Fig. 3(d) in particular, one can see that there is nonlinear deformation between the structures over time. Initially the plant is partially bent and then straightens. Nonetheless, since the GPR allows for nonlinearity, the correct correspondence between the tree structures are found and the tree reconstructions and registration are achieved accurately.

6.2. Automatic Change Detection in Brain Circuits

Long-term memory is thought to be stored in the configuration of the synaptic wiring diagram of brain circuits. The synaptic connections between neurons are found on tree-like dendrites and axons through which they receive in-

put and provide output respectively. The complex nature of dendrites and axons allows neurons to gather and distribute information from and to a plethora of other neurons that reside in spatially segregated areas. The rewiring of synaptic circuits could be accomplished by structural changes in the branches of those input and output trees, which would thereby reprogram the circuits function. This may be important for learning and memory formation. We collaborate with neuroscientists who aim at mapping structural circuit changes in the mouse brain during the learning processes.

To this end they acquire large-scale 2-photon laser scanning microscopy images of a sparse set of fluorescently labeled neurons in the neocortex. Images are taken through a permanently implanted cranial window, which lets them track specific structures over months during which the mouse learns new tasks or undergoes new experiences.

We used four large image stacks, labeled 1 to 4, of the same area of the brain at four different times. To train the path classifier, we selected a region from stacks 2 and 4, asked an expert to manually annotate them, and sampled 20000 positive and 20000 negative paths. One of the two training stacks is depicted by Fig. 4. Three sequences of smaller volumes were then selected from image stacks 1, 2 and 3 for testing. A single test sequence consists of three

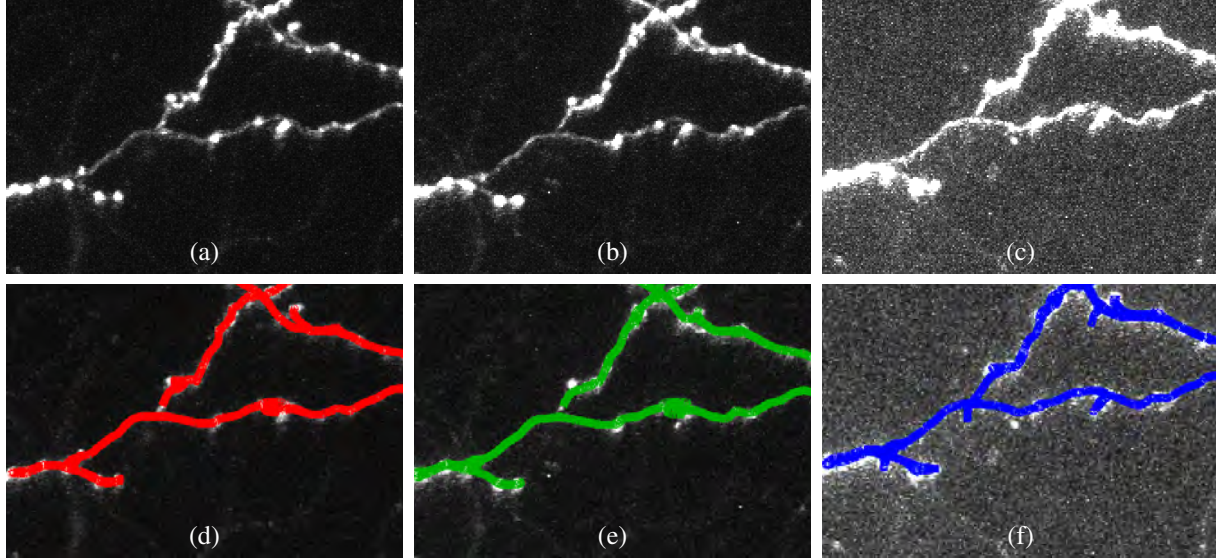


Figure 4. Results for the reconstruction using the images featuring brain circuits of dataset DS2. Automated reconstruction on DS2. (a,b,c) Maximum intensity projection of the images. (d,e,f) Reconstructions with DIADEM scores of 0.8471, 0.6422 and 0.5248, respectively. Note that the DIADEM score penalizes heavily even the relatively small errors in (f).

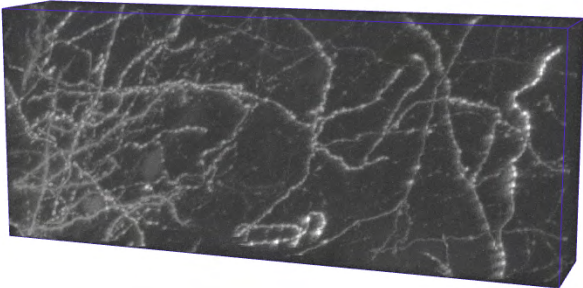


Figure 5. One of the volumes used for training.

	Single [24]	Pair	Triplet
DS1	Image #1	0.0944	0.9473
	Image #2	0.1828	0.8720
	Image #3	0.2985	0.9413
DS2	Image #1	0.2312	0.8471
	Image #2	0.1712	0.5475
	Image #3	0.0165	0.6236
DS3	Image #1	0.3369	0.5507
	Image #2	0.3177	0.6819
	Image #3	0.2423	0.6905

Table 1. Tree reconstruction DIADEM score [2] on our three datasets. These scores were obtained using either single images without temporal consistency or image pairs and triplets and enforcing time consistency.

volumes representing roughly the same brain area, each one taken from a different stack. We will refer to them as DS1, DS2, and DS3.

For each volume in a dataset, we evaluated the recon-

struction performance of our approach when using either zero, one, or two additional time instances. When no additional time instance is used, we simply pick regularly spaced high-tubularity points for the vertices of the graph and our approach reduces to that of [24]. Figs. 1 and 5 depict our results when using all three images simultaneously on DS1 and DS2, respectively.

In Table 1, we show the resulting DIADEM scores, which can range from 0.0 to 1.0 with 1.0 being best. That is in each entry of the table, we show the reconstruction score obtained for each image when using a specific number of additional time instances to reconstruct the neural structures. Note that our approach consistently produces more reliable reconstructions than those obtained using a single instance.

To further quantify the impact of the different components of our approach, we reran our algorithm using multiple image instances but setting $q = 0.5$, which is the only time we changed the value of q . Recall that this implies that the temporal inconsistencies are not penalized in the QMIP. Hence, in this configuration, any difference in performance between using one, two, or three images can be attributed to the strategy used to select vertices across multiple images, as discussed in Section 4. When using the three images of the DS1 dataset, this yields DIADEM scores of 0.9485, 0.8734, and 0.5528 for images #1, #2 and #3 respectively. This is lower than what we get when enforcing the time-consistency constrains but considerably higher than the scores obtained for single images with traditional sampling. In other words, using our approach to generate stable sets of correspondences, regardless of whether we

also enforce temporal consistency during the optimization, already has a very significant impact.

7. Conclusion

We have proposed a novel framework for extracting and reconstructing trees from networks of curvilinear structures across multiple time instances. The heart of our approach lies in finding local and stable structures that are consistent over time, and which can be used to disambiguate cases where individual time-instance reconstructions would fail. These additional time constraints are combined with more spatial constraints as inputs to a Quadratic Mixed Integer Program, and allow all time instance trees to be reconstructed at once.

We showed experimentally that our approach successfully takes advantage of temporal information to produce more reliable and accurate reconstructions of tree structures. In addition, we showed that our approach has the added benefit of automatically detecting regions of significant change in tree structures.

References

- [1] K. Al-Kofahi, S. Lasek, D. Szarowski, C. Pace, G. Nagy, J. Turner, and B. Roysam. Rapid Automated Three-Dimensional Tracing of Neurons from Confocal Image Stacks. *TITB*, 6(2):171–187, 2002. 1, 2
- [2] G. Ascoli, K. Svoboda, and Y. Liu. Digital Reconstruction of Axonal and Dendritic Morphology DIADEM Challenge, 2010. 5, 7
- [3] E. Bas and D. Erdogmus. Principal Curves as Skeletons of Tubular Objects - Locally Characterizing the Structures of Axons. *Neuroinformatics*, 9(2-3):181–191, 2011. 2
- [4] H. BenShitrit, J. Berclaz, F. Fleuret, and P. Fua. Multi-Commodity Network Flow for Tracking Multiple People. *PAMI*, 2013. 2
- [5] D. Donohue and G. Ascoli. Automated Reconstruction of Neuronal Morphology: An Overview. *Brain Research Reviews*, 67:94–102, 2011. 1
- [6] M. Fischler, J. Tenenbaum, and H. Wolf. Detection of Roads and Linear Structures in Low-Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique. *CVIP*, 15(3):201–223, March 1981. 2
- [7] A. Frangi, W. Niessen, K. Vincken, and M. Viergever. Multiscale Vessel Enhancement Filtering. *Lecture Notes in Computer Science*, 1496:130–137, 1998. 1
- [8] G. Gonzalez, F. Aguet, F. Fleuret, M. Unser, and P. Fua. Steerable Features for Statistical 3D Dendrite Detection. In *MICCAI*, pages 625–32, September 2009. 1
- [9] Gurobi Optimizer, 2012. <http://www.gurobi.com/>. 5
- [10] A. Holtmaat, J. Randall, and M. Canea. Optical Imaging of Structural and Functional Synaptic Plasticity in Vivo. *European Journal of Pharmacology*, 2013. 1
- [11] M. Jacob and M. Unser. Design of Steerable Filters for Feature Detection Using Canny-Like Criteria. *PAMI*, 26(8):1007–1019, August 2004. 1
- [12] C. Kirbas and F. Quek. Vessel Extraction Techniques and Algorithms: A Survey. In *Symposium on Bioinformatics and BioEngineering*, pages 238–245, 2003. 1
- [13] M. Law and A. Chung. Three Dimensional Curvilinear Structure Detection Using Optimally Oriented Flux. In *ECCV*, 2008. 1
- [14] M. Law and A. Chung. An Oriented Flux Symmetry Based Active Contour Model for Three Dimensional Vessel Segmentation. In *ECCV*, pages 720–734, 2010. 1, 3
- [15] Q. Li, Z. Deng, Y. Zhang, X. Zhou, U. V. Nagerl, and S. T. C. Wong. A Global Spatial Similarity Optimization Scheme to Track Large Numbers of Dendritic Spines in Time-Lapse Confocal Microscopy. *TMI*, 30(3):632–641, 2011. 1, 2
- [16] Z. Li, S. Liu, R. Weinreb, J. Lindsey, M. Yu, L. Liu, C. Ye, Q. Cui, W. Yung, C. Pang, D. Lam, and C. Leung. Tracking Dendritic Shrinkage of Retinal Ganglion Cells After Acute Elevation of Intraocular Pressure. *Invest Ophthalmol Vision Science*, 52(10):7205–12, 2011. 1
- [17] E. Meijering. Neuron Tracing in Perspective. *Cytometry Part A*, 77(7):693–704, 2010. 1
- [18] I. Mikic, S. Krucinski, and J. Thomas. Segmentation and tracking in echocardiographic sequences: Active contours guided by optical flow estimates. *TMI*, 17:274–284, 1998. 2
- [19] H. Peng, F. Long, and G. Myers. Automatic 3D Neuron Tracing Using All-Path Pruning. *Bioinformatics*, 27(13):239–247, 2011. 2
- [20] P. Prusinkiewicz. Modeling of spatial structure and development of plants: a review. *Scientia Horticulturae*, 74(1–2):113–149, 1998. 1, 5
- [21] C. E. Rasmussen and C. K. Williams. *Gaussian Process for Machine Learning*. MIT Press, 2006. 1, 4
- [22] Y. Sato, S. Nakajima, H. Atsumi, T. Koller, G. Gerig, S. Yoshida, and R. Kikinis. 3D Multi-Scale Line Filter for Segmentation and Visualization of Curvilinear Structures in Medical Images. *MIA*, 2:143–168, June 1998. 1
- [23] E. Serradell, P. Glowacki, J. Kybic, F. Moreno, and P. Fua. Robust Non-Rigid Registration of 2D and 3D Graphs. In *CVPR*, June 2012. 4, 5
- [24] E. Turetken, F. Benmansour, and P. Fua. Automated Reconstruction of Tree Structures Using Path Classifiers and Mixed Integer Programming. In *CVPR*, June 2012. 1, 2, 5, 7
- [25] E. Turetken, G. Gonzalez, C. Blum, and P. Fua. Automated Reconstruction of Dendritic and Axonal Trees by Global Optimization with Geometric Priors. *Neuroinformatics*, 9(2-3):279–302, 2011. 2
- [26] Y. Wang, A. Narayanaswamy, and B. Roysam. Novel 4D Open-Curve Active Contour and Curve Completion Approach for Automated Tree Structure Extraction. In *CVPR*, pages 1105–1112, 2011. 2
- [27] X. Yu, J. Tian, and J. Liu. Transformation Model Estimation for Point Matching via Gaussian Processes. In *World Congress of Engineering*, 2007. 4
- [28] T. Zhao, J. Xie, F. Amat, N. Clack, P. Ahammad, H. Peng, F. Long, and E. Myers. Automated Reconstruction of Neuronal Morphology Based on Local Geometrical and Global Structural Models. *Neuroinformatics*, 9:247–261, May 2011. 1, 2