# Non-Rigid Graph Registration using Active Testing Search

Eduard Serradell, Miguel Amável Pinheiro, Raphael Sznitman, Jan Kybic, *Senior Member*
Francesc Moreno-Noguer and Pascal Fua, *IEEE Fellow*

**Abstract**—We present a new approach for matching sets of branching curvilinear structures that form graphs embedded in $\mathbb{R}^2$ or $\mathbb{R}^3$ and may be subject to deformations. Unlike earlier methods, ours does not rely on local appearance similarity nor does require a good initial alignment. Furthermore, it can cope with non-linear deformations, topological differences, and partial graphs. To handle arbitrary non-linear deformations, we use Gaussian Processes to represent the geometrical mapping relating the two graphs. In the absence of appearance information, we iteratively establish correspondences between points, update the mapping accordingly, and use it to estimate where to find the most likely correspondences that will be used in the next step. To make the computation tractable for large graphs, the set of new potential matches considered at each iteration is not selected at random as in many RANSAC-based algorithms. Instead, we introduce a so-called *Active Testing Search* strategy that performs a priority search to favor the most likely matches and speed-up the process. We demonstrate the effectiveness of our approach first on synthetic cases and then on angiography data, retinal fundus images, and microscopy image stacks acquired at very different resolutions.

**Index Terms**—Graph matching, Non-rigid registration, Active search.

✦

## 1 INTRODUCTION

Graph-like structures are pervasive in biomedical 2D and 3D images. Examples are blood vessels, pulmonary bronchi, or nerve fibers. They can be acquired at different times and scales, or using different modalities, which may result in vastly diverse image appearances. For example, neuronal structures acquired using a light microscope (LM), such as those in the upper row of Fig. 1, look radically different when imaged using an electron microscope (EM) that, as shown in the bottom row of Fig. 1, has a much higher magnification. Nevertheless, registering them is desirable in order to identify the same region in both images and to combine the specific information each modality provides, in this case large-scale connectivity from the low-resolution data and fine details such as dendritic spines from the high-resolution data.

This kind of drastic appearance changes makes it impractical to use registration techniques that rely on maximizing image similarity [26], [40], in particular when the images are very different and when dealing with thin structures, such as blood vessels or neuronal fibers. The lack of distinguishing features of individual branching points or edges makes the use of feature-based correspondence techniques equally impractical. Since the graph geomet-

rical and topological structure may be the only property shared across modalities, graph matching becomes the only effective registration means. This also includes subgraph matching when the images have been acquired at different resolutions.

Most existing techniques that attempt to do this rely on matching Euclidean or geodesic distances between graph junction points [11], [16], [34], which is very sensitive to the small length changes inherent to the biological structures we consider. This may be valid for pulmonary vessels, which undergo a smooth deformation, or retinal fundus images that show only slight non-linearities produced when the curved surface of the retina is viewed from different viewpoints. Yet, when dealing with images acquired using distinct modalities and at different resolutions, the acquired structures exhibit significant topology changes, for example, due to the failure of one of the methods to display parts of the structure. Similarly, large non-linear deformations may occur because we work with a living specimen and the acquisitions are separated in time or because the deformation is introduced by the sample preparation or handling process. We know of no current method that can simultaneously handle all issues related to this kind of data: non-linear deformation, unknown initial position and lack of distinguishing local features.

We therefore propose a new approach for matching graph structures embedded in either $\mathbb{R}^2$ or $\mathbb{R}^3$, which can deal with these cases while being robust to topological differences between the two graphs and even changes in the distances between vertices. It requires no initial pose estimate, can handle non-linear deformations, and does not rely on local appearance or global distance matrices. Instead, given graphs extracted from the two images or image-stacks

E. Serradell and F. Moreno-Noguer are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, Spain.
M.A. Pinheiro and J. Kybic are with the Center for Machine Perception, Czech Technical University in Prague, Czech Republic.
R. Sznitman and P. Fua are with the Computer Vision Laboratory, EPFL, Lausanne, Switzerland.
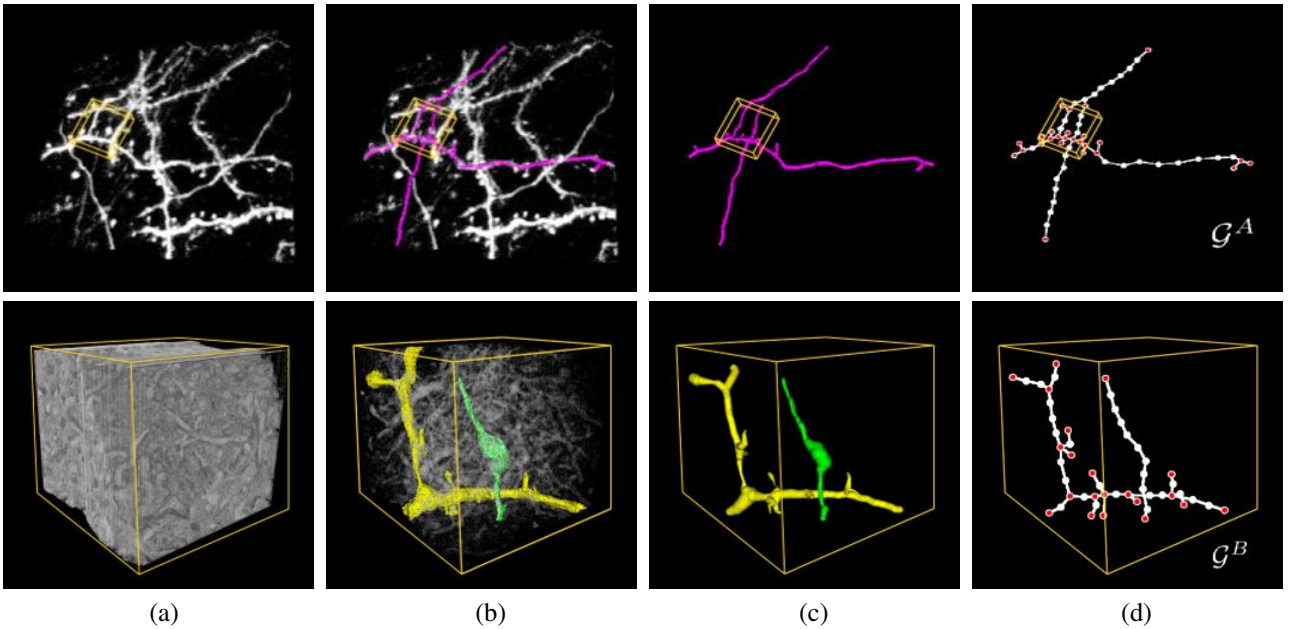
| (a) | (b) | (c) | (d) |

Fig. 1: **Brain tissue at different resolutions. (a)** Image stack acquired using a two-photon light microscope from live brain tissue at a 1 micrometer resolution and a smaller area of the same tissue imaged using an electron microscope, at a 20 nanometer resolution. The orange box in the top image denotes the area from which the EM sample has been extracted. **(b)** Semi-automated delineation of some dendrites overlaid in magenta and manual segmentation of an axon overlaid in green and a dendrite in yellow. **(c)** The segmented structures on a black background. Since the resolution is much higher in the EM data, dendritic spines and synapses are clearly visible. **(d)** Graph representation of the neuronal structures. The red dots, named "graph nodes", are used for a coarse registration of the graphs. The white dots, named "edge points", are used for fine alignment. This figure, as most others in this paper, is best viewed in color.

to be registered, we treat graph nodes as the features to be matched. We model the geometric mapping from one data set to the other as a Gaussian Process whose predictions are progressively refined as more correspondences are added. These predictions are in turn used to explore the set of all possible correspondences starting with the most likely ones, which allows convergence at an acceptable computational cost even though no appearance information is available. To make the computation tractable for large graphs, we introduce an active testing search strategy for speeding up the exploration of the set of possible correspondences by first considering the most likely ones.

We demonstrate the effectiveness of our technique on a variety of registration problems including both synthetic and real data of angiographies, retinal-fundus images acquired at different times and different points of view, ras well as neural image-stacks acquired using different modalities.

This paper is an extended version of [30], that proposed using Gaussian Processes for non-rigid matching and [25], which introduced a preliminary version of the active testing strategy [15]. The present paper combines both previous contributions, provides a more thorough mathematical justification of the active search technique, and includes a more extensive set of experiments and comparisons with other methods.

## 2 RELATED WORK

Area-based registration techniques that maximize image similarity criteria such as correlation or mutual-information [26], [40] are not applicable in our context as they are not designed to deal with truly different appearances and limited capture ranges. We therefore consider only techniques that match graph structures across images, which we have split into four major categories. In most cases described below, only the branching points (nodes) extracted from the input structures are used for matching, while the edges connecting them are ignored.

In the first class, the graph nodes are assumed to be related by a low-dimensional geometric transform, such as a rigid mapping, which can be instantiated from very few correspondences. It is therefore feasible to hypothesize and test random correspondences, as it is done in RANSAC [13]. However, as the number of transformation parameters or graph nodes increases, the space of possible matches becomes too large to explore randomly, and one has to resort to methods like PROSAC [9] or Guided-MLESAC [37] that reduce the search space through priors based on appearance. When appearance information is not available, more sophisticated search strategies have to be used, such as accelerated hypothesis sampling with information derived from the residual sorting [36]. Here, we propose to use an Active Search strategy [15], [25], [35], which iteratively selects the hypotheses that maximize the information gain, allowing to rapidly progress towards the

global minimum.

The second class of approaches typically requires a good initial estimate of the transformation to establish an initial estimate of the correspondences, which are then progressively refined. For rigid transformations, one of the earliest algorithms is the Iterative Closest Point (ICP) [4], later extended to non-rigid transformations using techniques such as Non-Rigid ICP [2], [20], or Coherent Point Drift (CPD) [23]. In any event, a good initial estimate is critical to prevent these methods from falling into incorrect local minima.

The third class of methods relies on having a sufficiently discriminative criterion for evaluating the pairwise compatibility of nodes, such as local appearance descriptors, or the geometric compatibility when considering correspondence pairs [7], [12], [16], [18], [19], [38]. Global nodal matches are then estimated using multidimensional optimization schemes such as graduated assignment [16], spectral techniques [18], [19], [38] or considering the graphs as an absorbing Markov chain [7]. Considering compatibilities as binary tests, the largest consistent set of matches corresponds to the maximum weighted independent set or equivalently, to the maximum weighted clique [12], [31]. Due to its high computational cost, the method is only applicable to small graphs. For specific medical imaging applications, some authors have attempted to register actual biological graphs we may find in structures like the pulmonary vessels [34], or the retina [1], [11]. Yet, while these methods allow for a non-parametric formulation of the problem, they cannot be used when appearance information is not available and the inter-nodal distances vary due to non-linear deformations, which is the case we consider in this paper.

The final category involves methods that simultaneously search for correspondences and estimate the transformation parameters using a Kalman filter-like approach [6], [10], [21], [28], [32], [33]. As soon as a few initial correspondences have been established, the set of remaining potential matches is rapidly reduced, making the search complexity manageable. However, these algorithms, like RANSAC, require an *a priori* parametric model of the transformation whose parameters are computed using the correspondences, and thus, difficult to generalise to arbitrary deformations. Similar limitations are also shared by methods relying on implicit shape models [14], [24]. In the Gaussian Process framework we propose, we also progressively reduce the number of potential correspondences but, in contrast to previous approaches, no a priori parametric deformation model is required. Instead, the deformation is completely defined by the correspondences and can therefore be completely generic. We will demonstrate that this significantly enlarges the applicability domain of our approach and improves the accuracy of the results.

## 3 APPROACH

Let us assume we are given two graphs $\mathcal{G}^A = (\mathbf{X}^A, \mathbf{E}^A)$ and $\mathcal{G}^B = (\mathbf{X}^B, \mathbf{E}^B)$, such as the one of Fig. 1-right,

extracted from two images or image-stacks $A$ and $B$. The $\mathbf{E}$s denote the graphs' edges and the $\mathbf{X}$s their nodes—shown as red dots in the figure—that are points in $\mathbb{R}^D$, where we assume $D \in \{2, 3\}$. The edges, in turn, are represented by dense sets of points—shown as white dots in the figure—that form $\mathbb{R}^2$ or $\mathbb{R}^3$ paths connecting the nodes.

Our goal is to use these two graphs to find a geometrical mapping $m$ from $A$ to $B$ such that $m(\mathbf{x}_i^A)$ is as close as possible to $\mathbf{x}_j^B$ in the least-squares sense assuming that $\mathbf{x}_i^A$ and $\mathbf{x}_j^B$ are corresponding pixels or voxels.

If correspondences between points belonging to the two graphs were given, we could directly use the Gaussian Process Regression (GPR) [27] to estimate a non-linear mapping that would yield a prediction of $m$ and its associated variance [5]. In our case, however, the correspondences are initially unavailable and cannot be established on the basis of local image information because the $A$ and $B$ are too different in appearance. In short, this means that we must rely only on geometrical properties to simultaneously establish the correspondences and estimate the underlying non-linear transform. Since attempting to do this directly for all edge points would be computationally intractable, our algorithm goes through the following two steps:

1) **Coarse alignment:** We begin by only matching graph nodes so that the resulting mapping is a combination of an affine deformation and a smooth non-linear deformation. We initialize the search by randomly picking D correspondences, which roughly fixes relative scale and orientation, and using them to instantiate a Gaussian Process (GP). We then recursively refine it as follows:
Given some matches between $\mathcal{G}^A$ and $\mathcal{G}^B$ nodes, the GP serves to predict where other $\mathcal{G}^A$ nodes should map and restricts the set of potential correspondences. Among these possibilities, we select the most promising one based on geometric or information gain criteria we will define in Section 5, and use it to refine the GP. Repeating this procedure recursively until enough mutually consistent correspondences have been established and backtracking when necessary lets us quickly explore the set of potential correspondences and recover an approximate geometric mapping.

2) **Fine alignment:** Having been learned only from potentially distant graph nodes, the above-mapping is coarse. To refine it, we also establish correspondences between points that form the edges connecting the nodes in such a way that distances along these edges, which we will refer to as *geodesic* distances, are changed as little as possible between the two graphs. Because there are many more such points than nodes, this would be extremely expensive to do from scratch. Therefore, we constrain the correspondence candidates to edges between already matched nodes and rely on a Hungarian algorithm [22] to perform the optimal assignment quickly.

In the remainder of this paper, we first outline the GPR model that we use. We then introduce our procedures for

**Gaussian Process Regression**

| | |
|---|---|
| $\mathcal{G}^A, \mathcal{G}^B$ | Source and target graph |
| $\mathbf{x}_i$ | Point in $\mathbb{R}^D$ |
| $D = \{2, 3\}$ | Dimension of the input points |
| $\mathbf{X}^A = \{\mathbf{x}_1^A, ..., \mathbf{x}_{n_A}^A\}$ | Set of nodes from the source graph |
| $\mathbf{X}^B = \{\mathbf{x}_1^B, ..., \mathbf{x}_{n_B}^B\}$ | Set of nodes from the target graph |
| $\mathbf{\Theta} = \{\theta_0, \theta_1, \theta_2, \theta_3\}$ | Set of GP hyperparameters |
| $\beta$ | Precision of the measurement noise |
| $k(\mathbf{x}_i, \mathbf{x}_j)$ | Kernel function |
| $n_c$ | Number of correspondences |

**Active Test Search Algorithm**

| | |
|---|---|
| $T$ | ATS total number of iterations |
| $\pi_t$ | Partial assignment selected at iteration $t$ |
| $\mathcal{C}_{\pi_t}$ | Set of children of the tree node $\pi_t$ |
| $S_{\pi_t}$ | Quality score of assignment $\pi_t$ |
| $\mathbf{\Omega} = \{\boldsymbol{\omega}_1^u, \boldsymbol{\omega}_0^u\}$ | Scoring noise model parameters |

TABLE 1: Summary of notations used in this paper

coarse and fine alignments. All the notations used in this paper are summarized in Table 1.

# 4 GAUSSIAN PROCESS REGRESSION

Without loss of generality, let us assume that the elements of $\mathbf{X}^A$ and $\mathbf{X}^B$ have been reordered so that the set $\pi = \{\mathbf{x}_l^A \leftrightarrow \mathbf{x}_l^B\}_{1 \leq l \leq n_c}$ denotes correspondences between $D$-dimensional points from $A$ and $B$ respectively. Using the GP approach to non-linear regression and assuming Gaussian i.i.d. noise of precision $\beta$ in all coordinate values, these correspondences can be used to predict that a point $\mathbf{x}^B$ in $B$ corresponding to $\mathbf{x}^A$ in $A$ can be expected to be found at a location with the following mean $m_\pi(\cdot)$ and isotropic variance $\sigma_\pi^2(\cdot)$ :

$$
\begin{aligned}
m_\pi(\mathbf{x}^A) &= \mathbf{k}^T \mathbf{C}_\pi^{-1} \mathbf{X}_\pi^B \ , & (1) \\
\sigma_\pi^2(\mathbf{x}^A) &= k(\mathbf{x}^A, \mathbf{x}^A) + \beta^{-1} - \mathbf{k}^T \mathbf{C}_\pi^{-1} \mathbf{k} \ , & (2)
\end{aligned}
$$

where $k$ is a kernel function, $\beta^{-1}$ is the measurement noise variance, $\mathbf{C}_\pi$ is the $n_c \times n_c$ symmetric matrix with elements $C_{i,j} = k(\mathbf{x}_i^A, \mathbf{x}_j^A) + \beta^{-1}\delta_{i,j}$, $\mathbf{k}$ is the vector $[k(\mathbf{x}_1^A, \mathbf{x}^A), \dots, k(\mathbf{x}_{n_c}^A, \mathbf{x}^A)]^T$, and $\mathbf{X}_\pi^B$ is the $n_c \times D$ matrix $[\mathbf{x}_1^B, \dots, \mathbf{x}_{n_c}^B]^T$.

Among the different types of existing kernel functions [27], we chose the widely used summation of a squared-exponential, a constant term, and a linear term

$$
k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 + \theta_1 \mathbf{x}_i^T \mathbf{x}_j + \theta_2 \exp\left\{-\frac{\theta_3}{2}||\mathbf{x}_i - \mathbf{x}_j||^2\right\}. \quad (3)
$$

Thus, the kernel of (3) models transformations as being rigid with non-linear deformations. Such transformation covers a wide range of situations in medical imaging [11], [26], [33], [34].

Given this expression for $k$, the geometric mapping from

Eq. (1) can now be rewritten as

$$
\begin{aligned}
m_\pi(\mathbf{x}^A) &= \sum_{i=1}^{n_c} \mathbf{a}_i k(\mathbf{x}_i^A, \mathbf{x}^A) \\
&= \sum_{i=1}^{n_c} \mathbf{a}_i(\theta_0 + \theta_1(\mathbf{x}_i^A)^T \mathbf{x}^A) + \\
&\quad \sum_{i=1}^{n_c} \mathbf{a}_i \theta_2 \exp\left\{-\frac{\theta_3}{2}||\mathbf{x}_i^A - \mathbf{x}^A||^2\right\} \ , \quad (4)
\end{aligned}
$$

where $\mathbf{a}_i$ is the $i^{th}$ row of the matrix $\mathbf{C}_\pi^{-1} \mathbf{X}_\pi^B$. The first term of Eq. (4), which contains the $\theta_0$ and $\theta_1$ hyperparameters, is a linear function of the input coordinates while the second one, which involves the $\theta_2$ and $\theta_3$, allows for additional non-linear deformations.

Apart from the mapping $m_\pi(\cdot)$, we also need to evaluate the mapping quality for any particular set of correspondences $\pi$. Let us define a quality score as $S_\pi \in \mathbb{R}$, which is a deterministic function. We use the following two methods to evaluate the quality of a correspondence set:

- **Assigned distance:** We compute the minimum possible total distance between $m_\pi(\mathbf{x}_i^A)$ and points $\mathbf{x}_j^B \in \mathbf{X}^B$ for all $1 \leq i \leq n_A$

$$
S_\pi = \sum_{i=1}^{n_A} \sum_{j=1}^{n_B} H_{i,j} \cdot \text{dist}(m_\pi(\mathbf{x}_i^A), \mathbf{x}_j^B), \quad (5)
$$

where 'dist' is a Euclidean distance and $H$ is the assignment matrix computed by the Hungarian algorithm [22] which, given the distances between all points $m_\pi(\mathbf{x}_i^A)$ and $\mathbf{x}_j^B$, iteratively calculates the optimal assignment of correspondences between all points, by minimizing $S_\pi$.

- **Number of inliers:** We compute the proportion of edge and branching points in $\mathbf{X}^A$ that are mapped near a point in $\mathbf{X}^B$ as

$$
S_\pi = \frac{|I|}{|\mathbf{X}^A|}, \quad (6)
$$

$$
I = \left\{\mathbf{x}_j^B \mid \exists m_\pi(\mathbf{x}_i^A), \ \text{dist}(m_\pi(\mathbf{x}_i^A), \mathbf{x}_j^B) < \beta^{-\frac{1}{2}}\right\}.
$$

Our experiments show that these two measures suffice to recognize good sets of correspondences.

# 5 COARSE ALIGNMENT

Let $\mathbf{X}^A = \{\mathbf{x}_1^A, \dots, \mathbf{x}_{n_A}^A\}$ and $\mathbf{X}^B = \{\mathbf{x}_1^B, \dots, \mathbf{x}_{n_B}^B\}$ be the nodes of our two graphs. Our first goal is to simultaneously retrieve as many correspondences $\pi = \{\mathbf{x}^A \leftrightarrow \mathbf{x}^B\}$ as possible and to determine the underlying non-linear mapping $\mathbf{x}^B = m_\pi(\mathbf{x}^A)$ that best aligns them.

In this section, we present two different approaches for doing this. The first one—Section 5.1—relies on first assigning correspondences to nodes for which there are few to choose from. The second—Section 5.2—uses a more sophisticated strategy that ranks partial solutions and attempts to extend the most promising ones first. We introduced the first strategy in [30] and tested it successfully on relatively small graphs. However, as will be shown in Section 7, its

---

**Algorithm 1** Greedy Alignment $(\mathcal{G}^A, \mathcal{G}^B; \Theta, \beta; T)$

---

1: Initialize Correspondence Set:
   $\pi_0 \leftarrow \{\mathbf{x}_{i_1}^A \leftrightarrow \mathbf{x}_{j_1}^B, \ldots, \mathbf{x}_{i_D}^A \leftrightarrow \mathbf{x}_{j_D}^B\}$
2: **for** $t = 0, \ldots, T$ **do**
3:     $\{m_{\pi_t}, \sigma_{\pi_t}^2\} = \text{ComputeMapping}(\mathbf{X}^A, \mathbf{X}^B, \Theta, \pi_t)$
4:     $S_{\pi_t} = \text{QualityScore}(m_{\pi_t}, \beta)$
5:     **for** $i = 1 \ldots n_A$ **do**
6:        $\mathcal{B}_i = \text{ComputeBoundary}(m_{\pi_t}(\mathbf{x}_i^A), \sigma_{\pi_t}^2(\mathbf{x}_i^A), \mathbf{X}^B)$
7:        $\text{PotCand}_i = \{\mathbf{x}_j^B; \mathbf{x}_j^B \in \mathbf{X}^B \wedge \mathbf{x}_j^B \in \mathcal{B}_i\}$
8:     **end for**
9:     $i^* = \arg\min_i\{|\text{PotCand}_i|\}$ for $|\text{PotCand}_i| \neq 0$
10:    **if** $i^* \neq \emptyset$ **then**
11:       $\mathbf{x}_{j^*}^B = \text{PickRandom}(\text{PotCand}_{i^*})$
12:       $\pi_{t+1} \leftarrow \pi_t \cup \{\mathbf{x}_{i^*}^A \leftrightarrow \mathbf{x}_{j^*}^B\}$
13:    **else**
14:       $\pi_{t+1} \leftarrow \pi_{t-1}$
15:    **end if**
16: **end for**
17: **return** $\pi^* = \arg\max_{\{1,\ldots,T\}} S_{\pi_t}$

---

computational requirements grow quickly with the number of graph nodes. The second strategy, while slightly more complex, scales better.

## 5.1 Greedy Search

Let $m_\pi(\cdot)$ be a GP written using the formulation of Section 4, which we instantiate by first selecting a set of $D$ random correspondences (line 1 in Algorithm 1) – we set the initial size of potential candidates to be the data points dimensionality. This gives us an initial correspondence set $\pi_0$. We then iteratively construct sets of correspondences in $T$ steps as follows.

1) At iteration $t$, we have a set of correspondences $\pi_t$ from which we compute (line 3) the mapping $m_{\pi_t}(\cdot)$ and covariance $\sigma_{\pi_t}^2(.)$ using Eqs. (1) and (2).

2) For each unmatched node $\mathbf{x}_i^A \in \mathbf{X}^A$, we search for potential correspondences $\mathbf{x}_j^B \in \mathbf{X}^B$ in the bounded region $\mathcal{B}_i$ determined by the predicted covariance $\sigma_{\pi_t}^2(\mathbf{x}_i^A)$ (lines 6–7). We use the Mahalanobis distance to define the boundary:

$$\mathcal{M}^2 = \left(m_{\pi_t}(\mathbf{x}_i^A) - \mathbf{x}_j^B\right)^T \left(\sigma_{\pi_t}^2(\mathbf{x}_i^A)\right)^{-1} \left(m_{\pi_t}(\mathbf{x}_i^A) - \mathbf{x}_j^B\right)$$

$$\mathcal{B}_i = \left\{\forall \mathbf{x}_j^B \in \mathbf{X}^B | \mathcal{M}^2(m_{\pi_t}(\mathbf{x}_i^A), \mathbf{x}_j^B) < 2\right\}$$

3) We choose the node $\mathbf{x}_{i^*}^A$ with the smallest number of potential candidates (line 9), and randomly pick one of them to define the match $\mathbf{x}_{i^*}^A \leftrightarrow \mathbf{x}_{j^*}^B$, which we add to the correspondence set $\pi_t$ (lines 11–12). If there is no point from $\mathbf{X}^B$ which satisfies the conditions to be selected, we remove the last added correspondence from $\pi_t$ and continue searching (line 14).

4) We take the quality score $S_{\pi_t}$ to be the number of inliers as defined in Eq. (6).

As described in Algorithm 1, this process uses a depth-first search and is repeated $T$ times, backtracking and selecting sets of correspondences of different sizes, unlike RANSAC which uses a fixed set size at each iteration. We then return the correspondence set $\pi^*$ with the highest score, and its corresponding $m_{\pi^*}$. We also terminate if the inlier consensus $S_{\pi_t}$ becomes large enough. An example of this process is shown in Fig. 2.

The process is controlled by the noise parameter $\beta$ of Eq. (2) and the vector $\Theta = \{\theta_0, \theta_1, \theta_2, \theta_3\}$ containing the kernel hyperparameters of Eq. (3). To avoid having to tune these parameters for each new dataset, we center and scale the $\mathbf{X}^A$ and $\mathbf{X}^B$ coordinates so that their average distance to the origin is one and perform the computation on the scaled datasets. As a result, we were able to use the same $\Theta$ for all experiments described in Section 7.

To speed up the computation, we reject matches that would produce large changes in geodesic distances, which we define as the length of a path connecting the edges between two graph nodes $\mathbf{x}_i$ and $\mathbf{x}_j$. Given already established correspondences $\pi$ between graphs, then for each new potential match, the geodesic distances from the new corresponding points to the already matched nodes in both graphs have to be approximately proportional. We set the tolerance for geodesic distance variations depending on the level of deformations we expect to recover. Proceeding in this way, the algorithm gains robustness against outliers, while avoiding unnecessary checks, thus keeping a low complexity. Note that geodesic distances are invariant to rotations, to the bending of the branches, and to isometric changes.

## 5.2 Active Test Search

We have tested the algorithm described above on graphs containing up to 100 nodes, for which the computation takes more than 1000 seconds in Matlab on an 4-Core 2.3 GHz 64-bit processor. because the computational cost grows exponentially with the number of nodes, it becomes impractical for larger graphs.

We have therefore developed an alternative approach that relies on the Active Testing Search (ATS) [15], [35]. This involves progressively refining an approximate solution by making a budgeted number of observations and computing the posterior distribution over all potential solutions after each test. Hence, the algorithm proceeds iteratively and selects at each step the correspondence set expected to yield the highest information gain based on all previous ones.

In other words, our method does not perform either a depth-first search, such as the one described in Section 5.1, or a traditional breadth-first search, but a priority search. For this purpose, as new correspondences are added to partial solutions, it maintains a sorted list of which ones are most likely to lead to a correct solution. It then attempts to extend these first so that less likely candidates may never be extended at all, saving computation time.

In addition, our ATS approach is adaptive and allows for backtracking without hand-tuned pruning of the search

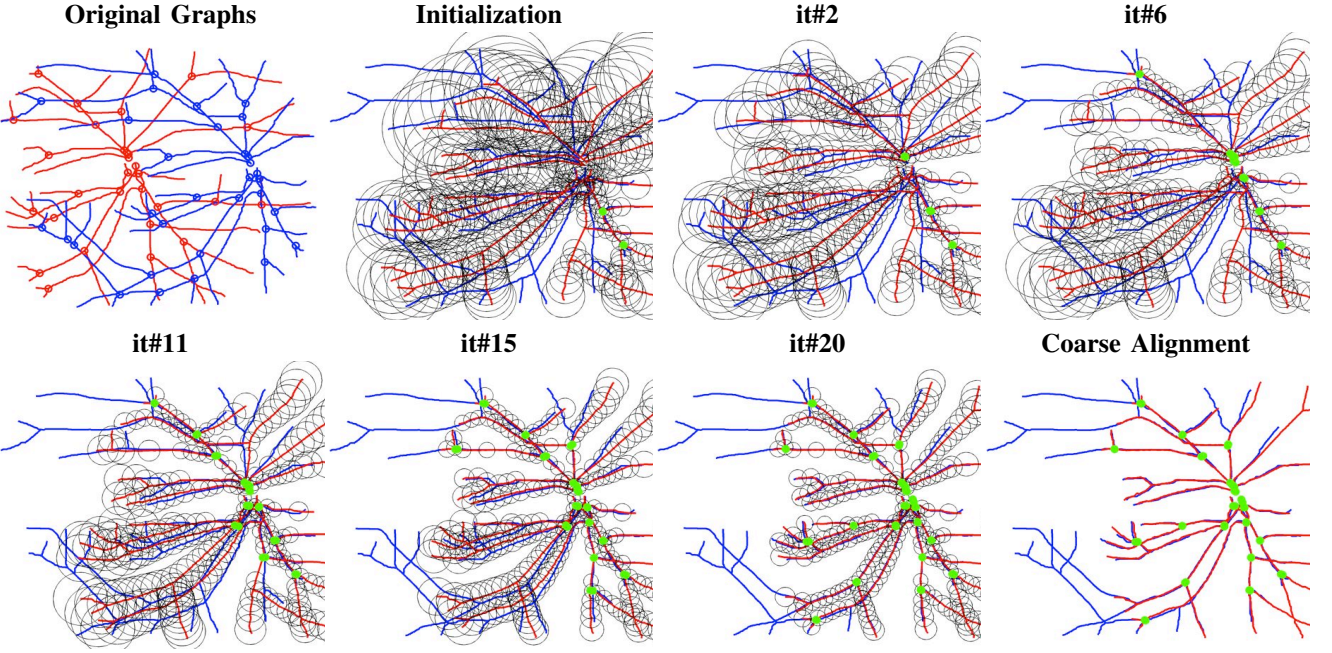| **Original Graphs** | **Initialization** | **it#2** | **it#6** |
|---|---|---|---|
| **it#11** | **it#15** | **it#20** | **Coarse Alignment** |

Fig. 2: **Coarse alignment steps.** The initial graph structures are depicted in the top left-most figure, the model graph in red and the target in blue. Exploration of the search space starts by picking randomly two correspondences, highlighted in green, thus roughly fixing scale and orientation. Then, the next match candidate is chosen among the nodes located inside the bounded regions, which are a function of the GP predicted covariances, shown as black ellipses. Every correspondence added to the hypotheses set helps refining the mapping uncertainty. The final correspondence set defines a coarse alignment of the graphs. Best viewed in color.
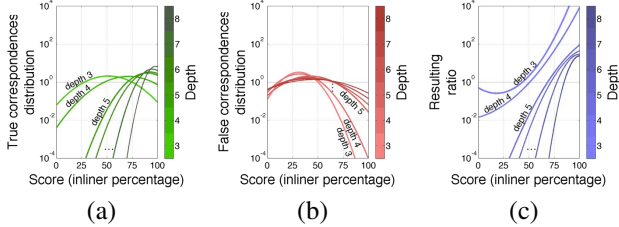
|     |     |     |
|:---:|:---:|:---:|
| (a) | (b) | (c) |

Fig. 3: **Gaussian noise models for percentage of inliers.** (a) Each curve depicts the Gaussian noise model $\mathcal{N}(\cdot; \boldsymbol{\omega}_1^u)$ for a given depth $u$ of the tree. (b) Similarly, each curve depicts the noise models for $\mathcal{N}(\cdot; \boldsymbol{\omega}_0^u)$. (c) Likelihood ratio $r(\cdot)$ between $\mathcal{N}(\cdot; \boldsymbol{\omega}_1^u)$ and $\mathcal{N}(\cdot; \boldsymbol{\omega}_0^u)$ for each value of $u$.

space. It is summarized in Algorithm 2 and we describe it in more details below.

### 5.2.1 ATS—Coarse Alignment

ATS maintains a list of candidate correspondence sets, where we denote the probability that the correspondence set $\pi$ is part of the correct mapping $\pi^*$ as $\epsilon_\pi$. This list of candidates is handled by means of a priority queue $Q$ (line 1 of Algorithm 2), whose elements are $(\pi, \epsilon_\pi)$ pairs sorted in order of decreasing probability.

At first, we form all possible sets of $D$ pairs of correspondences that can be used to initialize a mapping $m_\pi(\cdot)$. We assume that each $\pi$ from this initial set has equal probability $\epsilon$.

At each ATS iteration, $t = 0, ..., T$, we want to select the candidate set $\pi_t$ that is most likely to provide a good

mapping. We therefore select the first element $(\pi_t, \epsilon_{\pi_t})$ of $Q$, which is the one with highest likelihood and then evaluate the quality score $S_{\pi_t}$ (line 5) to verify if it is indeed a good mapping. Given that $S_{\pi_t}$ can be noisy, we consider it to be a random variable with a known noise model, *i.e.* the likelihood model $P(S_{\pi_t} | \pi^*)$ is assumed to be explicitly known and is described in the following section.

To aggregate the information provided by the quality score, we compute the posterior distribution of the correct correspondences given the newly observed score. We further refine our candidates by expanding the candidate set previously evaluated. In particular, from $\pi_t$, we generate a new set of candidate correspondences $\mathcal{C}_{\pi_t} = \{\pi_t \cup \{x_i^A \leftrightarrow x_j^B\} | x_j^B \notin \pi_t\}$, which contains all children of the current node, where $x_i^A \notin \pi_t$ is an additional element of $\mathbf{X}^A$.

As in [15], [35], the posterior for any element $\pi \in \mathcal{C}_{\pi_t}$ can be computed as

$$\epsilon \propto \frac{r(S_t)\epsilon_{\pi_t}}{|\mathcal{C}_{\pi_t}|}, \qquad (7)$$

where $r(S_t)$ is the likelihood ratio (defined explicitly in the following section). The complete derivation of this equation as well as why the normalization constant is not needed are shown in the Appendix. Intuitively, Eq. (7) is simply an application of Bayes rule, where $\epsilon_{\pi_t}$ is a prior, $r(S_t)$ is the data-term and dividing by $|\mathcal{C}_{\pi_t}|$ attributes equal a priori probability to all the expanded candidates of $\pi_t$.

This process is repeated $T$ times or until the likelihood ratio is higher than $\psi$. We then return the assignment $\pi^*$ with the best score.

---

**Algorithm 2** ATS Alignment $(\mathbf{X}^A, \mathbf{X}^B; \boldsymbol{\Theta}, \beta; T, \psi)$

---

1: Initialize Priority Queue:
    $Q \leftarrow \text{Push}\ \left(\{\mathbf{x}_{i_1}^A \leftrightarrow \mathbf{x}_{j_1}^B, \ldots, \mathbf{x}_{i_D}^A \leftrightarrow \mathbf{x}_{j_D}^B\}, \epsilon\right)$
2: **for** $t = 1 \ldots T$ **do**
3:     $\{\pi_t, \epsilon_t\} = \text{Pop}(Q)$
4:     $\{m_{\pi_t}, \sigma_{\pi_t}^2\} = \text{ComputeMapping}(\mathbf{X}^A, \mathbf{X}^B, \boldsymbol{\Theta}, \pi_t)$
5:     $S_{\pi_t} = \text{QualityScore}(m_{\pi_t})$
6:     **if** $r(S_{\pi_t}) > \psi$ **then return** $\pi^* = \pi_t$ **end if**
7:     **for** $\pi_z \in \mathcal{C}_{\pi_t}$ **do**
8:         $Q \leftarrow \text{Push}\ \left(\pi_t, \epsilon_{\pi_t} r(S_{\pi_t})/|\mathcal{C}_{\pi_t}|\ \right)$
9:     **end for**
10: **end for**
11: **return** $\ \pi^* = \arg\max_{\{1,\ldots,T\}} S_{\pi_t}$

---

### 5.2.2 Quality Score Selection and Noise Model

To compute the quality score $S_\pi$ for any set of correspondences $\pi$ we use the previously described assigned distance of Eq. (5) and the number of inliers of Eq. (6). In particular, we compute $S_\pi$ using the assigned distance when the number of correspondences $|\pi|$ is below a certain threshold $\gamma$, which we set to 5 in all experiments. Otherwise we compute $S_\pi$ using the number of inliers. We found that combining two different quality functions provides more informative scores for small and large sets of correspondences. This is similar to the strategies employed in [6].

We consider the quality scores to be random noisy observations and assume the following observation model

$$P(S_\pi = s | \pi^*) = \begin{cases} \mathcal{N}(s; \boldsymbol{\omega}_1^u), & \text{if } \delta(\pi, \pi^*) = 1 \\ \mathcal{N}(s; \boldsymbol{\omega}_0^u), & \text{if } \delta(\pi, \pi^*) = 0 \end{cases}, \quad (8)$$

where $\pi^*$ is the correct set of correspondence assignments, $u$ is the number of correspondences in $\pi$, $\mathcal{N}$ is a Gaussian probability distribution with parameters $\boldsymbol{\omega}_1^u$, $\boldsymbol{\omega}_0^u$ and $\delta(\pi, \pi^*) = 1$ if the correspondences of $\pi$ respect $\pi^*$ and 0 otherwise. From this model, the likelihood ratio can be computed as

$$r(S_\pi) = \frac{\mathcal{N}(S_\pi; \boldsymbol{\omega}_1^u)}{\mathcal{N}(S_\pi; \boldsymbol{\omega}_0^u)}. \quad (9)$$

To learn the parameters of the Gaussian distributions $\mathcal{N}(\cdot; \boldsymbol{\omega}_1^u)$ and $\mathcal{N}(\cdot; \boldsymbol{\omega}_0^u)$, we proceed as follows:

- **True Distribution:** To estimate the parameters for $\mathcal{N}(\cdot; \boldsymbol{\omega}_1^u)$, we synthetically generate $L$ point clouds $\mathbf{X}^A$ such that $n_B > n_A$ and fit a minimum spanning tree to obtain the graph representation. The point cloud $\mathbf{X}^B$ is generated by applying random affine transformations and a smaller amplitude non-linear deformation to $\mathbf{X}^A$. This allows us to know exactly the true correspondence $\pi^*$ and generate a set $\{\{\mathbf{X}^A\}_l, \{\mathbf{X}^B\}_l, \pi_l^*\}_{l=1}^L$. Then, we take subsets of the full set of true correspondences $\pi^*$ and compute $S_\pi$. Once all scores on all $L$ generated sets are computed, we estimate the Gaussian distribution parameters $\{\boldsymbol{\omega}_1^u\}_{u=1}^{n_A} = \{\mu_1^u, \sigma_1^u\}_{u=1}^{n_A}$. An example of the learned probability densities can be seen in Fig. 3(a).

- **False Distribution:** Second, to learn the parameters for $\mathcal{N}(\cdot; \boldsymbol{\omega}_0^u)$, we follow a similar sampling approach. Given the number of possible incorrect correspondences, for partial assignments that include many assignments, *i.e.* when $u$ is large, we construct sets of incorrect correspondences $\pi$ by starting from a subset of $\pi^*$ and adding a few incorrectly matched points. Proceeding in this manner, we take false partial assignments which are close to the true correspondence $\pi^*$ because we expect to only explore the higher depths of the search tree, that is, high values of $u$, when our previous hypotheses are correct. An example of such distribution is depicted in Fig. 3(b).

In practice, we have found the above the process for learning the parameters of the observation models to be effective and robust. If enough training data with known correspondences is available, we could learn more complex models for the real shape of the positive and negative distributions. In addition, even though we use synthetically generated datasets, the same learned parameters are good enough to be used across different experiments in Sec. 7 and indicate that the parameters are fairly robust and valid for different tasks.

## 6 FINE ALIGNMENT

Given two graphs $\mathcal{G}^A$ and $\mathcal{G}^B$, both coarse alignment algorithms described above in Sections 5.1 and 5.2 return a set of corresponding graph nodes $\pi^*$, along with the corresponding mapping $m(\cdot) = m_{\pi^*}$ and the covariance estimator $\sigma^2(\cdot) = \sigma_{\pi^*}^2$.

This set of matches $\pi^*$ relates the graph's nodes and is therefore coarse. Given that the nodes are connected by paths, we can refine the mapping by establishing correspondences between edge points that lie on these paths. We assume the node correspondences to be correct and only establish new ones between points lying on paths linking matching nodes. We proceed iteratively using the following steps:

1) For each pair of paths connected by corresponding graph nodes, we use the Hungarian algorithm [22], guided by the current mapping $m_{\pi_r}(\cdot)$ and covariance estimator $\sigma_{\pi_r}^2(\cdot)$, to establish new matches $\pi_{r+1}$ between the edge points of the two paths. We constrain all the matches to have a consistent geodesic distance with their respective graph nodes.
2) Given these new correspondences $\pi_{r+1}$, reestimate $m_{\pi_{r+1}}(\cdot)$ and $\sigma_{\pi_{r+1}}^2(\cdot)$.
3) Compute the quality of the resulting mapping $S_{\pi_{r+1}}$ using the assigned distance function defined in Eq. (5).
4) If $S_{\pi_{r+1}} > S_{\pi_r}$, iterate. Otherwise, terminate and return $\pi_r$, $m_{\pi_r}(\cdot)$, and $\sigma_{\pi_r}^2(\cdot)$.

This yields a final expanded set of correspondences $\pi_{fine}$, mapping $m_{\pi_{fine}}(\cdot)$, and covariance estimator $\sigma_{\pi_{fine}}^2(\cdot)$. Note that we use the same GP parameters $\boldsymbol{\Theta}$ as before. The whole process is illustrated by Fig. 4 and summarized in Algorithm 3.

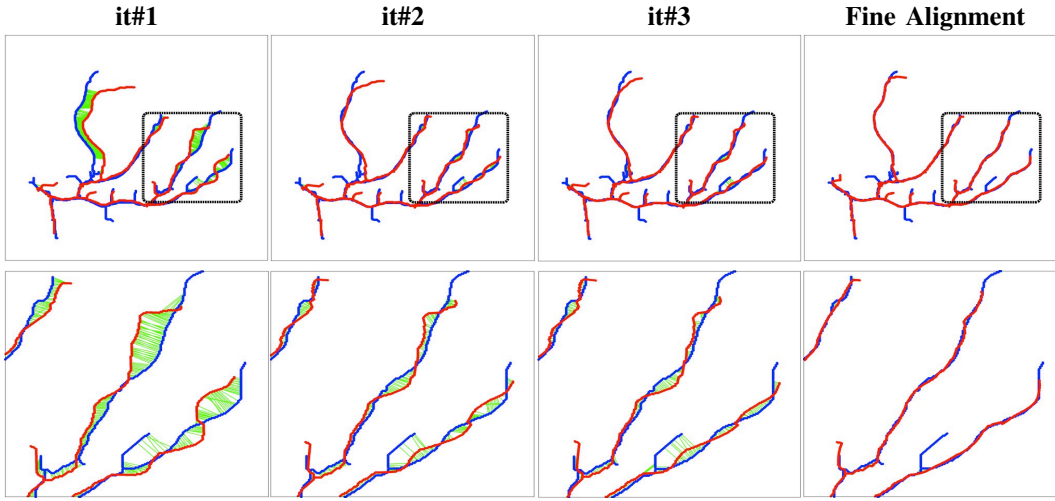| it#1 | it#2 | it#3 | Fine Alignment |
|------|------|------|----------------|



Fig. 4: **Fine alignment steps.** Once a coarse alignment of the two graphs (model in red and target in blue) has been found, the algorithm starts matching points lying on the edges. The assignments (depicted in green) are computed using the Hungarian algorithm and constrained by the graph topology and GP predictions. After a few iterations, the warped structure **(top)** is completely aligned to the target graph. For each successive plot, we zoom to a smaller region **(bottom)** to better show the algorithm at work. Best viewed in color.

---

**Algorithm 3** Fine Alignment $(\mathcal{G}^A, \mathcal{G}^B; \mathbf{\Theta}, \beta; \pi^*)$

1: Initialize Correspondence Set from Coarse Alignment:
   $\pi_r = \pi^*$
   $m_{\pi_r} = m_{\pi_*}, \ \sigma^2_{\pi_r} = \sigma^2_{\pi_*};$
2: **repeat**
3: $\quad \pi_{r+1} \leftarrow \text{OptimalAssignment}(\mathbf{X}^A, \mathbf{X}^B, \mathbf{\Theta}, \pi_r)$
4: $\quad \{m_{\pi_{r+1}}, \sigma^2_{\pi_{r+1}}\} = \text{ComputeMapping}(\mathbf{X}^A, \mathbf{X}^B, \mathbf{\Theta}, \pi_{r+1})$
5: $\quad S_{\pi_{r+1}} = \text{QualityScore}(m_{\pi_{r+1}})$
6: **until** $S_{\pi_{r+1}} \leq S_{\pi_r}$
7: **return** $\pi_{fine} = \pi_r$

---

# 7 EXPERIMENTS

We evaluate our approach on both synthetic and real data against state-of-the-art methods. In the remainder of the paper we will refer to the methods presented in Sections 5.1 and 5.2 as Greedy-RGM (Greedy Search for Robust Graph Matching) and ATS-RGM (Active Testing Search for Robust Graph Matching), respectively. We will initially test all methods on synthetically generated data with increasing levels of noise, non-rigid deformation, missing points and different initial conditions. We will then show the performance of the algorithms on 2D and 3D biomedical images, including retinal images, neuronal structures and heart angiograms. The Gaussian Process parameters $\beta$ and $\theta_i$ should ideally be estimated from training data. However, due to the limited amounts of available data in our applications, we will manually select these parameters once for all experiments. We keep the problem of estimating them from validation data as a possible future work direction.

We will compare our algorithm to several others for non-rigid point matching and shape recovery. As representative examples of *point set registration* we have chosen the original Iterative Closest Point [4] (ICP), the Thin Plate Splines-Robust Point Matching (TPS-RPM) [8], the Coherent Point Drift (CPD) [23] and the recent Gaussian Mixture Model Registration (GMMREG) proposed in [17]. As examples of *graph matching* approaches, we have considered Spectral Matching (SM) [18] and Integer Projected Fixed Point (IPFP) [19], which can be combined, as well as Path Following Algorithm (PATH) [39] [1]. The results of the coarse alignment obtained by ATS-RGM and our previous Greedy-RGM version are virtually the same, therefore only the results for ATS-RGM are shown when comparing accuracy.

We ran all the algorithms on a 2.3 GHz 4-Core 64-bit machine with 8 GB RAM. Most of the aforementioned algorithms are implemented using a combination of Matlab and Mex-C++ functions. Similarly, we implemented the skeleton of our approach in Matlab and used C++ for the most time-consuming parts: the Gaussian Process routines and the Active Testing Search.

## 7.1 Synthetic Experiments

To evaluate our approach against others, we generated pairs of trees composed of a *model tree* computed as the minimum spanning tree of $N = 50$ randomly selected 2D points in a $2 \times 2$ bounding box and *target tree* obtained by deforming it. More specifically, let $\mathbf{x}_i$ be the nodes of the model tree and $\mathbf{x}'_i$ those of the target tree. For all $i$ from 1 to $N$, we write

$$\mathbf{x}'_i = \mathcal{R}(\phi) \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \mathbf{x}_i + \begin{bmatrix} T_x \\ T_y \end{bmatrix} + \mathcal{D}(p_d) + \xi(\sigma_n), \quad (10)$$

where the deformation includes the following components.

- Rotation, scaling, and translation; The target model is rotated by an angle $\phi$, translated by $T_x$ and $T_y$, and scaled by $S_x$ and $S_y$

---

1. All implementations were made available by respective authors, except of ICP, which was made available by a third party and verified to be a correct implementation of the algorithm.
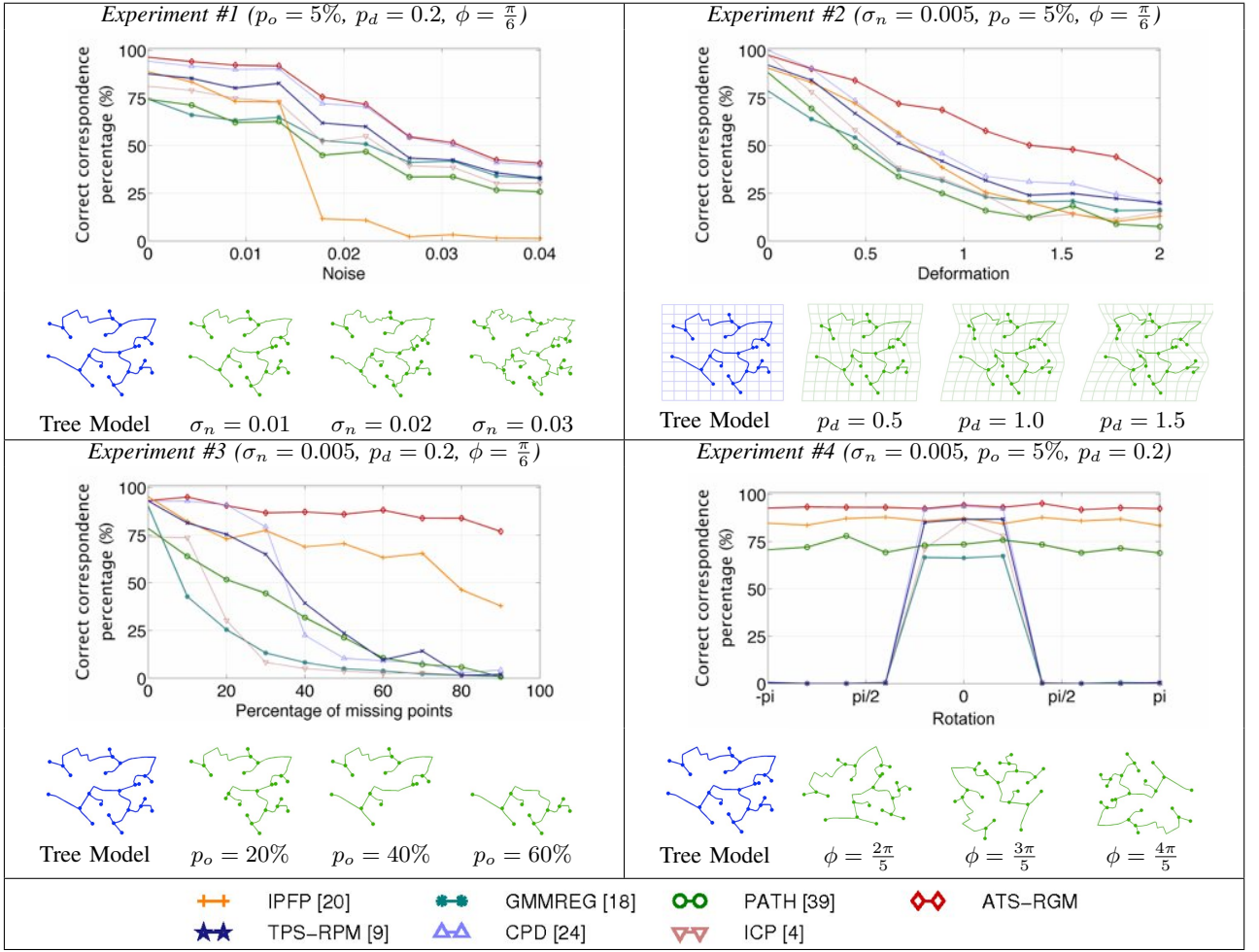
Fig. 5: **Quantitative evaluation on synthetic data**. Performance tests of all competing methods in different configurations of noise (Exp. #1), deformation (Exp. #2), outliers (Exp. #3) and rotation (Exp. #4). Curves represent the median of the correct correspondences percentage achieved by each method. Below each result, we show the *tree model* used in the experiments (in blue) and corrupted illustrations of how each parameter affects the transformed graph (in green).

- Non-linear deformation: We add a non-linear warping $\mathcal{D}(p_d)$ defined as a linear combination of B-splines whose control points are uniformly distributed in the input space. Its magnitude is controlled by $p_d$, which specifies the amount of displacement of the B-spline coefficients.
- Noise: We perturb the node locations of the target graph by a zero mean Gaussian noise $\xi(\sigma_n)$, where $\sigma_n$ is the standard deviation.
- Outliers: We produce outliers by randomly and independently removing a percentage $p_o$ of the total number $N$ of nodes from both the model and target trees. As a result, some branches appear in one tree and not the other.

We use the pairs of trees created in this manner as input to all algorithms. To ensure a fair comparison, we either set the required parameters to the values suggested in the corresponding papers, or manually modify them in case they did not provide good results. The exact values for

these configuration parameters are as follows: [2].

- ICP: Does not require extra parameters.
- TPS-RPM: We set the initial temperature $T_{init}$ to half the maximum Euclidean distance between the nodes of the model tree, and the final temperature to $T_{final} = 0.01 \cdot T_{init}$. The remaining parameters are set to $\lambda_1^{init} = 1$ and $\lambda_2^{init} = 0.01$, as suggested by the authors.
- GMMREG: We set the maximum number of iterations to 10000 with $\alpha = 1$ and $\beta = 2$.
- CPD: We use a non-rigid configuration of the algorithm for all the experiments. We set $\lambda = 3$, $\beta = 3$ and $outliers = 0.2$.
- SM and IPFP: We build the affinity matrix using the description provided in [18]. As we do not have appearance information, we set the label affinity term

2. Note that for ease of reference we are keeping the same parameter notation here as in the original papers. While some of these parameters are also used in our algorithm (*i.e.* : $w$, $T$, $\sigma_d$) their meaning are not necessarily the same.

$M(a, a)$ to zero, making the matching score depend solely on the pairwise geometric information. The pairwise affinity is set to $M(a, b) = 4.5 - \frac{(d_{ij} - d_{i'j'})^2}{2\sigma_d^2}$ if $|d_{ij} - d_{i'j'}| < 3\sigma_d$ and zero otherwise.

- PATH: We build the graphs as suggested by the authors for the application of "alignment of vessel images". We connect each node $x_i$ to all points $x_j$ within a distance $r$, and each edge is assigned a weight $w_{i,j} = \exp(-||x_i - x_j||_2), \forall x_i, x_j \in X, ||x_i - x_j||_2 < r$. This is done for all nodes of both model and target trees. In our case we set $r = 0.3 \cdot d_{max}$, where $d_{max}$ is the maximum Euclidean distance between the points of the model tree.

### 7.1.1 Performance Evaluation

We tested all algorithms for robustness to rotation, deformation, and topology changes by varying the geometric deformation parameters of Eq. (10) as well as the percentage of missing nodes $p_0$. We performed four different independent experiments. For each we generated 50 pairs of model and target trees using the same set of parameters and a fixed small change in scale and translation. In *Experiment #1* we evaluated the influence of noise on the points 2D locations by sweeping the range $\sigma_n \in [0, 0.04]$ and setting $p_o = 5\%$, $p_d = 0.2$ and $\phi = \frac{\pi}{6}$. In *Experiment #2* we tested the behavior of the algorithms against increasing levels of non-linear deformation, varying the deformation parameter within the range $p_d \in [0, 2]$, and setting the rest of parameters to constant values $\sigma_n = 0.005$, $p_o = 5\%$, and $\phi = \frac{\pi}{6}$. Similarly, in *Experiment #3*, we assessed the robustness of the algorithms to the presence of outliers by randomly deleting a percentage $p_o \in [0\%, 90\%]$ of nodes in both trees and setting $\sigma_n = 0.005$, $p_d = 0.2$ and $\phi = \frac{\pi}{6}$. Finally, in *Experiment #4* we tested the invariance of all the methods to initial conditions by changing the orientation of the target within $\phi \in [-\pi, \pi]$ and fixing the rest of the parameters to $\sigma_n = 0.005$, $p_o = 5\%$ and $p_d = 0.2$. To give significance to the magnitudes of the experimental parameters we consider, the images below each performance curve of Fig. 5 show different samples of
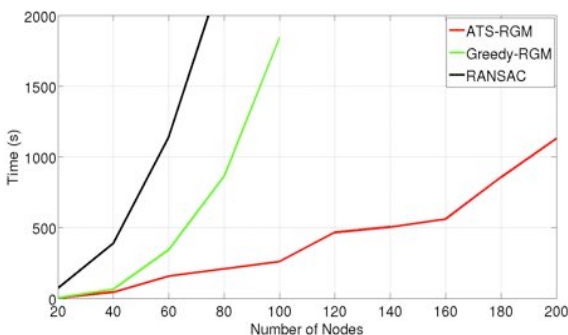


Fig. 6: **Computational Cost.** Processing time required by RANSAC, Greedy-RGM and ATS-RGM as a function of the number of nodes. We computed the median of 20 experiments for each of the methods.

the same model graph and different target graphs generated by varying the levels of noise, deformation, outliers and rotation.

For each experimental parameter setting experiments and each algorithm, we computed the average percentage of correct matches over our 50 tree pairs and plot the results in Fig. 5. Under favorable conditions, that is, relatively small graphs with less than 50 nodes, uncorrupted data and purely affine transformation, all methods exhibit similar performance. However, when we progressively introduce artifacts the differences become clear. For instance, it can be seen that CPD deals poorly when there are missing parts or when the initial rotation is above 70 degrees, as stated in their paper. Similarly, the rigid ICP can only find local solutions, even when dealing with much smaller initial rotation angles. Graph methods (PATH, SM+IPFP) are able to find global solutions as they are invariant to initial conditions by construction. However, as shown in the first row of Fig. 5, they are very sensitive to modifications in the topology of the graph. TPS-RPM and GMMREG underperform our approach for each of the tests. In short, each one of the competing methods can address some of the difficulties, however only ours can handle all of them.

The results of Fig. 5 also indicate the suitability of using our approach in sub-graph matching. The robustness of our approach stems from the randomized search strategy, that allows searching for a global minimum and makes the algorithm insensitive to initial rotations. This is true for both Greedy and Active Testing Search. On the other hand, the non-rigid transformation based on Gaussian Process regression provides robustness to large amounts of noise and deformation.

### 7.1.2 Computational Cost

Finally, we compared the computational cost of the two versions of our algorithm and RANSAC [13]. We generated a new set of experiments in which the new model and target tree pairs are as before except for the fact that we varied the number of nodes from 20 to 200 and performed an affine transformation plus random noise of small amplitude.

As illustrated in Fig. 6, the computation time grows much faster for both Greedy-RGM and RANSAC than for ATS-RGM, which remains manageable. Both versions of our algorithm (ATS-RGM and Greedy-RGM) yield similar performance, only differing in the time consumed to reach the global solution. Note that, since the transformations are almost affine, the absolute run-time value is lower than some of the real experiments we consider in the next section.

Note that the purely local methods, such as CPD and ICP, tend to be much faster and can deal with thousands of points in reasonable amounts of time. Yet, as we have seen in the fourth experiment of Fig. 5, these algorithms require accurate initializations. On the other hand, the graph matching methods, such as SM + IPFP or PATH, treat the problem as an Integer Quadratic Program (IQP) and hence are limited by the construction of the *pairwise score* matrix whose size grows quadratically with the number of nodes.

| (a) | (b) | (c) | (d) | (e) | (f) |
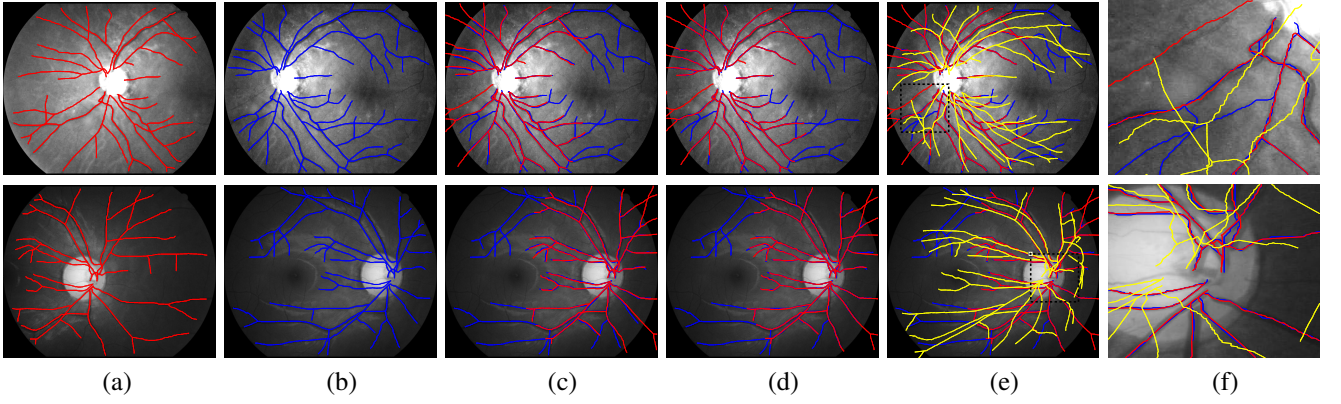
Fig. 7: **Retinal fundus images used in [11].** Each row contains a single experiment. **(a,b)** Two images of the same retina taken from different viewpoints, with the vascular trees overlaid in red and blue. **(c)** The first tree is overlaid in red over the second image after non-linear transformation, which corresponds to the output of the Greedy-RGM coarse alignment. **(d)** Final result of our non-rigid registration: the graph from the first image is overlaid in red over the second image. **(e)** Our result is superposed with the Coherent Point Drift alignment, in yellow. **(f)** Detail of the rectangle in (e). Our algorithm behaves well on this dataset, while CPD fails to recover the correct shape because there are too many non-corresponding branches. Best viewed in color.
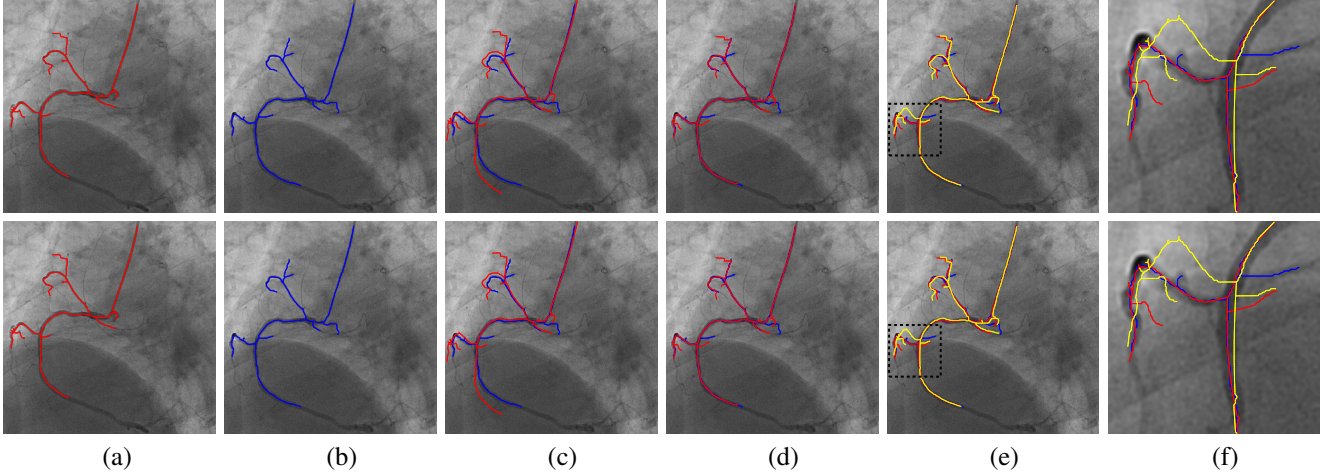


| (a) | (b) | (c) | (d) | (e) | (f) |

Fig. 8: **Angiography images from a beating heart.** **(a)** Two different images with extracted vascular trees overlaid in red. **(b)** Two other images taken later in the heart cycle with extracted vascular trees overlaid in blue. **(c)** The original red trees are shown after the non-linear coarse alignment of the tree nodes, obtained using our Greedy-RGM. **(d)** The resulting warped trees are overlaid in red after non-linear registration. Note that the trees —in particular in the first example— have distinctly different topologies, which affects our algorithm very little. **(e)** Comparison with the result obtained using non-linear Coherent Point Drift, in yellow. **(f)** A zoom of a region of interest. Using the graph intrinsic geometry grants us robustness against vessel bendings and outliers, achieving a better registration of the two shapes. Best viewed in color.

## 7.2 Real Data Experiments

We next present several examples of results obtained by ATS-RGM and Greedy-RGM on real biomedical datasets. The graphs were extracted semi-automatically using a plugin [3] for the Fiji platform [29].

To evaluate the accuracy of the different methods in the absence of ground truth, we assigned each node in the deformed graphs —overlaid in red for our method and yellow for CPD on Figs. 7 and 8 (d,e,f)— to its assumed match in the target graph overlaid in blue. To this end, we use the Hungarian algorithm to find it by taking the Euclidean distance as the cost to minimize, while rejecting

outlier branches by setting a distance threshold defined *ad hoc* for each of the datasets. We use this error measure since there is no true correspondence between the sampled points along the edges of the graphs. This error measure gives an idea of the quality of the alignment.

In Table 2, we show these errors and the corresponding computation times. For ATS-RGM and Greedy-RGM, we distinguish the times required for coarse and fine alignment. We have not provided the error for IPFP and PATH because these methods only give correspondence hypotheses, and are unable to define a valid transformation without an outlier rejection step.
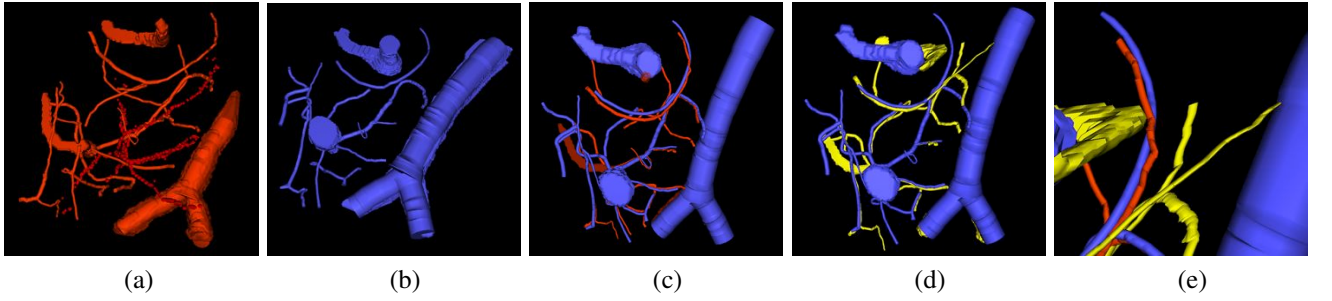
| (a) | (b) | (c) | (d) | (e) |

Fig. 9: **Blood vessels in brain tissue. (a)** Segmented two photon microscopy data. **(b)** Segmented bright-field optical microscopy data. **(c)** Registration of structures using Active Testing Search, in red. **(d)** Alignment using CPD, in yellow. **(e)** A view in detail at the results of both ATS-RGM and CPD. Best viewed in color.
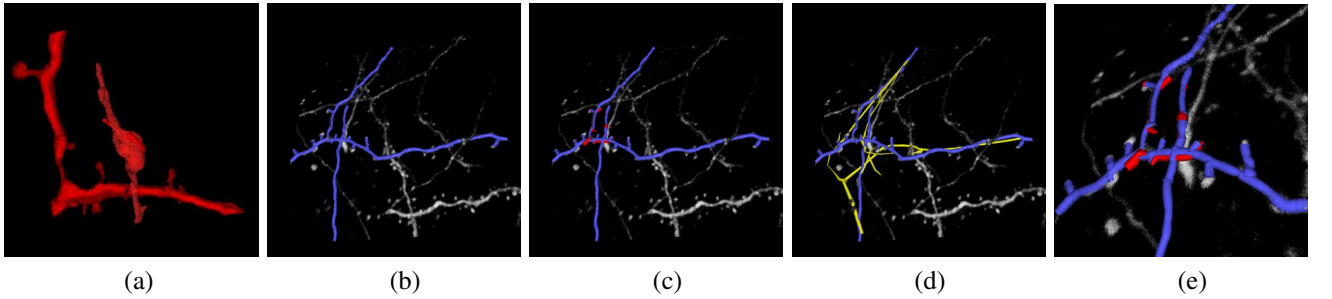


| (a) | (b) | (c) | (d) | (e) |

Fig. 10: **Light and electron microscopy neuronal trees. (a)** Graph structure extracted from the electron microscopy image stack, in red. **(b)** Segmented light microscope neurons in blue. **(c)** After the non-linear registration process using ATS-RGM, the EM segmented neuron is deformed and aligned over the LM extracted neuron. **(d)** Registration using CPD, in yellow, which falls into a local minimum. **(e)** A zoom over the region where the EM stack has been extracted. The two neurons have been completely aligned. Best viewed in color.

| Dataset | | ATS-RGM | Greedy-RGM | CPD [23] | ICP [4] | TPS-RPM [8] | GMMREG [17] |
|---|---|---|---|---|---|---|---|
| **Retina I** | Error (pix) | **2.68** | **2.68** | 20.67* | 20.04* | 19.32* | 21.30* |
| (Fig. 7 Top) | Time (s) | 1293.3 + 406.4 | 5998.9 + 336.8 | 580.2 | 24.3 | 4236.8 | 139.7 |
| **Retina II** | Error (pix) | 2.94 | **2.51** | 20.45* | 20.46* | 17.79* | 20.84* |
| (Fig. 7 Bot.) | Time (s) | 280.0 + 301.4 | 16353.1 + 261.2 | 468.5 | 67.9 | 5791.1 | 147.1 |
| **Angio I** | Error (pix) | 1.16 | **1.05** | 2.95 | 9.77* | 2.92 | 4.21 |
| (Fig. 8 Top) | Time (s) | 307.8 + 129.4 | 1240.9 + 162.8 | 144.3 | 8.1 | 726.7 | 31.9 |
| **Angio II** | Error (pix) | **1.57** | 1.81 | 3.42 | 4.84* | 3.21 | 6.56 |
| (Fig. 8 Bot.) | Time (s) | 167.9 + 77.2 | 112.0 + 95.4 | 68.8 | 5.0 | 327.0 | 21.1 |
| **Brain vessels** | Error (vox) | 4.38 | 4.89 | **4.19** | 7.23 | 6.67 | 12.71 |
| (Fig. 9) | Time (s) | 593.7 + 55.5 | 15029.1 + 19.9 | 37.1 | 30.9 | 334.8 | 31.2 |
| **Neuronal** | Error ($\mu m$) | **0.05** | 0.07 | 0.25* | 0.27* | 0.20* | 0.46* |
| (Fig. 10) | Time (s) | 42.4 + 15.8 | 116.1 + 18.2 | 22.2 | 28.2 | 28.5 | 22.4 |

TABLE 2: **Error:** Geometric error on real datasets for the proposed approach and other state of the art methods. Failed experiments (producing incorrect alignment) are marked with an *, see Fig. 7(d) or Fig. 10(d) for examples. **Elapsed Time:** Processing time for each method, in seconds. For ATS-RGM and Greedy-RGM, we distinguish the times required for coarse and fine alignment.

In Fig. 7 we show registration results for retinal fundus vascular graphs that are deformed from one image to the next because the camera is looking from different viewpoints. This produces distortions of the curved retinal surface's projection, that are well modeled by an affine transform. Thus, there is very little non-linearity in the deformation and these results are similar to those of [11], even though the trees only partially overlap. However, as the amount of spurious branches is quite large, CPD fails to recover the correct shape. In contrast, our approach can naturally handle such artifacts.

In the 2D X-ray angiography images of Fig. 8 the non-linearities of the transformation are much more apparent. As shown in the zoomed-in area, our algorithm nevertheless does a good job of recovering this more complex deformation and aligning the trees. Again, we assessed the performance of the CPD on these images and observed that it could not retrieve a correct solution unless a relatively accurate initialization was provided. Even when we supplied with our coarse transformation estimate, CPD could only deal with small non-linearities.

Next, we register two 3D datasets. A blood vessels

network in brain tissue is shown in Fig. 9. One of the 3D image stacks is acquired using a two-photon microscope and the other using bright-field microscopy after excision and fixation. As the resulting segmentations are partially aligned, the experiment's difficulty consists in identifying the non-linearities of the deformation. Our algorithm clearly outperforms other state-of-art methods and provides results similar to the CPD. In the zoomed image of Fig. 9(e) it is possible to appreciate it. Although CPD works well when the initial conditions are favorable, it completely misaligns one of the branches while our result respects the topology.

Finally, we register the 3D neuronal stacks extracted from the brain tissue of Fig. 1 using light (LM) and electron (EM) microscopy, where the EM block is a small section of the LM volume and has been non-linearly deformed due to the extracting process. The intended application is to automatically localize the EM volume in the corresponding part of the LM volume. Even though the two images look extremely different, our algorithm returns a valid deformation as shown in Fig. 10. No other method was able to recover the correct alignment.

## 8 CONCLUSION

We have shown that our algorithm can match graphs with neither appearance information nor initial pose estimate, while allowing for partial matches and non-linear deformations. This is made possible by using Gaussian Processes to model the geometric mapping from one graph to another and using this mapping to progressively constrain the search for correspondences between graph nodes. Because this mapping is non-parametric, we can effectively handle situations containing high levels of non-linear deformations with many nodes and without assuming any particular transformation model beforehand.

For relatively small graphs a simple depth-first approach to establishing the correspondences yields good results. As the graphs become bigger, the set of all possible correspondences becomes too large and we therefore proposed a more sophisticated approach that ranks partial solutions and attempts to extend the most promising ones first. This allows us to efficiently and correctly align biological structures that are non-linearly transformed and extracted with different techniques, without the need of prealigning them. We have also shown in quantitative experiments that our method consistently outperforms state-of-the-art.

Yet, in cases where the number of points to match exceeds a few thousands, the initial search space is excessively large and it remains challenging to find promising regions of the search space efficiently. For this reason, we plan to investigate how appearance data can also be included in our framework to provide more effective strategies of exploring the search space. Similarly, in cases where the model parameters used are very different from the true structure, our efficient search method can be ineffective. As such, we will also explore how to estimate the parameters of our model in a exploration-exploitation scenario.

## REFERENCES

[1] W. Aguilar, Y. Frauel, F. Escolano, M. Elena Martinez-Perez, A. Espinosa-Romero, and M. Angel Lozano. A Robust Graph Transformation Matching for Non-Rigid Registration. *Image and Vision Computing*, 27:897–910, 2009.

[2] B. Amberg, S. Romdhani, and T. Vetter. Optimal Step Nonrigid ICP Algorithms for Surface Registration. In *Conference on Computer Vision and Pattern Recognition*, 2007.

[3] F. Benmansour, E. Turetken, and P. Fua. Tubular Geodesics: Fiji plugins, 2012. http://cvlab.epfl.ch/software/delin/index.php.

[4] P. Besl and N. Mckay. A Method for Registration of 3D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

[5] C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[6] M. Chli and A.J. Davison. Active Matching. In *European Conference on Computer Vision*, pages 72–85, 2008.

[7] M. Cho, J. Lee, and K.M. Lee. Reweighted Random Walks for Graph Matching. In *European Conference on Computer Vision*, pages 492–505, 2010.

[8] H. Chui and A. Rangarajan. A New Point Matching Algorithm for Non-Rigid Registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.

[9] O. Chum and J. Matas. Matching with Prosac - Progressive Sample Consensus. In *Conference on Computer Vision and Pattern Recognition*, pages 220–226, June 2005.

[10] A.J. Davison. Active search for real-time vision. In *International Conference on Computer Vision*, pages 66–73, 2005.

[11] K. Deng, J. Tian, J. Zheng, X. Zhang, X. Dai, and M. Xu. Retinal Fundus Image Registration via Vascular Structure Graph Matching. *International Journal of Biomedical Imaging*, 2010.

[12] O. Enqvist, K. Josephson, and F. Kahl. Optimal Correspondences from Pairwise Constraints. In *International Conference on Computer Vision*, 2009.

[13] M.A Fischler and R.C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications ACM*, 24(6):381–395, 1981.

[14] K. Fujiwara, K. Nishino, J. Takamatsu, B. Zheng, and K. Ikeuchi. Locally Rigid Globally Non-Rigid Surface Registration. In *International Conference on Computer Vision*, 2011.

[15] D. Geman and B. Jedynak. An Active Testing Model for Tracking Roads in Satellite Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):1–14, January 1996.

[16] S. Gold and A. Rangarajan. A Graduated Assignment Algorithm for Graph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:377–388, 1996.

[17] B. Jian and B. C. Vemuri. Robust Point Set Registration Using Gaussian Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633–1645, 2011.

[18] M. Leordeanu and M. Hebert. A Spectral Technique for Correspondence Problems Using Pairwise Constraints. In *International Conference on Computer Vision*, pages 1482–1489, October 2005.

[19] M. Leordeanu, M. Hebert, and R. Sukthankar. An Integer Projected Fixed Point Method for Graph Matching and Map Inference. In *Advances in Neural Information Processing Systems*, pages 1114–1122, 2009.

[20] H. Li, R.W. Sumner, and M. Pauly. Global Correspondence Optimization for Non-Rigid Registration of Depth Scans. In *Symposium on Geometry Processing*, pages 1421–1430, 2008.

[21] F. Moreno-noguer, V. Lepetit, and P. Fua. Pose Priors for Simultaneously Solving Alignment and Correspondence. In *European Conference on Computer Vision*, October 2008.

[22] J. Munkres. Algorithms for the Assignment and Transportation Problems. *Journal of the Society of Industrial and Applied Mathematics*, 5(1):32–38, March 1957.

[23] A. Myronenko and X.B. Song. Point-Set Registration: Coherent Point Drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, 2010.

[24] N. Paragios, M. Rousson, and V. Ramesh. Non-Rigid Registration Using Distance Functions. *Computer Vision and Image Understanding*, pages 142–165, 2003.

[25] M.A. Pinheiro, R. Sznitman, E. Serradell, J. Kybic, F. Moreno-Noguer, and P. Fua. Active Testing Search for Point Cloud Matching. In *Information Processing in Medical Imaging,*, 2013.

[26] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual Information Based Registration of Medical Images: A Survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, 2003.

[27] C. E. Rasmussen and C. K. Williams. *Gaussian Process for Machine Learning*. MIT Press, 2006.

[28] J. Sanchez-Riera, J. Ostlund, P. Fua, and F. Moreno-Noguer. Simultaneous Pose, Correspondence and Non-Rigid Shape. In *Conference on Computer Vision and Pattern Recognition*, June 2010.

[29] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, and B. Schmid. Fiji: an open-source platform for biological-image analysis. 9(7):676–682, 2012. Code available at http://pacific.mpi-cbg.de.

[30] E. Serradell, P. Glowacki, J. Kybic, F. Moreno-Noguer, and P. Fua. Robust Non-Rigid Registration of 2D and 3D Graphs. In *Conference on Computer Vision and Pattern Recognition*, pages 996–1003, June 2012.

[31] E. Serradell, F. Moreno-Noguer, J. Kybic, and P. Fua. Robust Elastic 2D/3D Geometric Graph Matching. In *SPIE Medical Imaging*, 2012.

[32] E. Serradell, M. özuysal, V. Lepetit, P. Fua, and F. Moreno-Noguer. Combining Geometric and Appearance Priors for Robust Homography Estimation. In *European Conference on Computer Vision*, pages 58–72, September 2010.

[33] E. Serradell, A. Romero, R. Leta, C. Gatta, and F. Moreno-Noguer. Simultaneous Correspondence and Non-Rigid 3D Reconstruction of the Coronary Tree from Single X-Ray Images. In *International Conference on Computer Vision*, November 2011.

[34] D. Smeets, P. Bruyninckx, J. Keustermans, D. Vandermeulen, and P. Suetens. Robust Matching of 3D Lung Vessel Trees. In *MICCAI Workshop on Pulmonary Image Analysis*, 2010.

[35] R. Sznitman and B. Jedynak. Active Testing for Face Detection and Localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(10):1914–1920, June 2010.

[36] J. Yu T.-J. Chin and D. Suter. Accelerated Hypothesis Generation for Multi-Structure Data via Preference Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):625–638, 2012.

[37] B. Tardoff and D. W. Murray. Guided Sampling and Consensus for Motion Estimation. In *European Conference on Computer Vision*, pages 82–96, 2002.

[38] L. Torresani, V. Kolmogorov, and C. Rother. Feature Correspondence via Graph Matching: Models and Global Optimization. In *European Conference on Computer Vision*, pages 596–609, 2008.

[39] M. Zaslavskiy, F. Bach, and J-P. Vert. A Path Following Algorithm for the Graph Matching Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242, 2009.

[40] B. Zitova and J. Flusser. Image Registration Methods: A Survey. *Image and Vision Computing*, 11(11):977–1000, 2003.

**Eduard Serradell** received the Telecommunications degree from the Universitat Politècnica de Catalunya in 2005. After that, he worked at the French Space Agency (Toulouse, France) until 2006. Since 2009, he is a PhD student at the Institut de Robòtica i Informàtica Industrial. His current research focuses on robust matching of deformable shapes.



**Miguel Amável Pinheiro** received an MSc degree in electrical and computers engineering from the Faculty of Engineering of the University of Porto (FEUP) in 2010. He is now pursuing a PhD degree at the Center of Machine Perception, which is a part of the Department of Cybernetics of the Faculty of Electrical Engineering of the Czech Technical University (CTU) in Prague. His research interests include point cloud and graph matching, image registration and medical imaging.



**Raphael Sznitman** received his BSc in cognitive systems from the University of British Columbia in 2007. Following this, he received his MSc and PhD in computer science from the Johns Hopkins University in 2011. Currently, he is a postdoctoral fellow at the École Polytechnique Fédérale de Lausanne (Switzerland) where he works in the computer vision laboratory. His research interests are primarily in computational vision, probabilistic methods and statistical learning, applied to applications in biomedical imaging.



**Jan Kybic** received a master degree from the Czech Technical University in Prague, Czech Republic and a Ph.D. from EPFL, Switzerland, in 1998 and 2001, respectively. He held a post-doc position at INRIA, France, in 2002–2003. Since 2003 he is at Czech Technical University in Prague, becoming an Associate Professor in 2010 and currently serving as a Department Head. He was a visiting researcher at EPFL for his sabbatical in 2010. Jan Kybic has authored and co-authored 21 articles in peer-reviewed international scientific journals, one book, two book chapters, and over 60 conference publications. He has supervised 10 PhD students, 5 of them have already successfully graduated. He is an Associate Editor for IEEE Transactions on Medical Imaging. His current research interests include algorithms for biomedical signal and image processing, especially image registration, segmentation and classification, numerical methods, interpolation, inverse problems, and algorithm theory.



**Francesc Moreno-Noguer** received the MSc degrees in industrial engineering and electronics from the Universitat Politècnica de Catalunya (UPC) and the Universitat de Barcelona, in 2001 and 2002, respectively. He obtained his PhD degree from UPC in 2005, and his work received the UPC's Doctoral Dissertation Extraordinary Award. From 2006 to 2008 he was a postdoctoral fellow at the computer vision departments of Columbia University and the École Polytechnique Fédérale de Lausanne. In 2009 he joined the Institut de Robòtica i Informàtica Industrial in Barcelona, as an associate researcher of the Spanish Scientific Research Council. His research interests are focused on retrieving rigid and non-rigid shape, motion and camera pose from single images and video sequences



**Pascal Fua** received an engineering degree from Ecole Polytechnique, Paris, in 1984 and the Ph.D. degree in Computer Science from the University of Orsay in 1989. He joined EPFL (Swiss Federal Institute of Technology) in 1996 where he is now a Professor in the School of Computer and Communication Science. Before that, he worked at SRI International and at INRIA Sophia-Antipolis as a Computer Scientist. His research interests include shape modeling and motion recovery from images, analysis of microscopy images, and Augmented Reality. He has (co)authored over 200 publications in refereed journals and conferences. He is an IEEE Fellow and has been an associate editor of IEEE journal Transactions for Pattern Analysis and Machine Intelligence. He often serves as program committee member, area chair, and program chair of major vision conferences.