Non-Rigid Graph Registration using Active Testing Search

Journal:	*Transactions on Pattern Analysis and Machine Intelligence*
Manuscript ID:	TPAMI-2013-08-0618
Manuscript Type:	Regular
Keywords:	I.2.10 Vision and Scene Understanding < I.2 Artificial Intelligence < I Computing Methodologies, I.2.1.b Computer vision < I.2.1 Applications and Expert Knowledge-Intensive Systems < I.2 Artificial Intelligence < I Computing Methodologies, I.2.8.e Graph and tree search strategies < I.2.8 Problem Solving, Control Methods, and Search < I.2 Artificial Intelligence < I Computing Methodologies, I.4.3.d Registration < I.4.3 Enhancement < I.4 Image Processing and Computer Vision < I Computing Methodologies

SCHOLARONE™
Manuscripts

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.

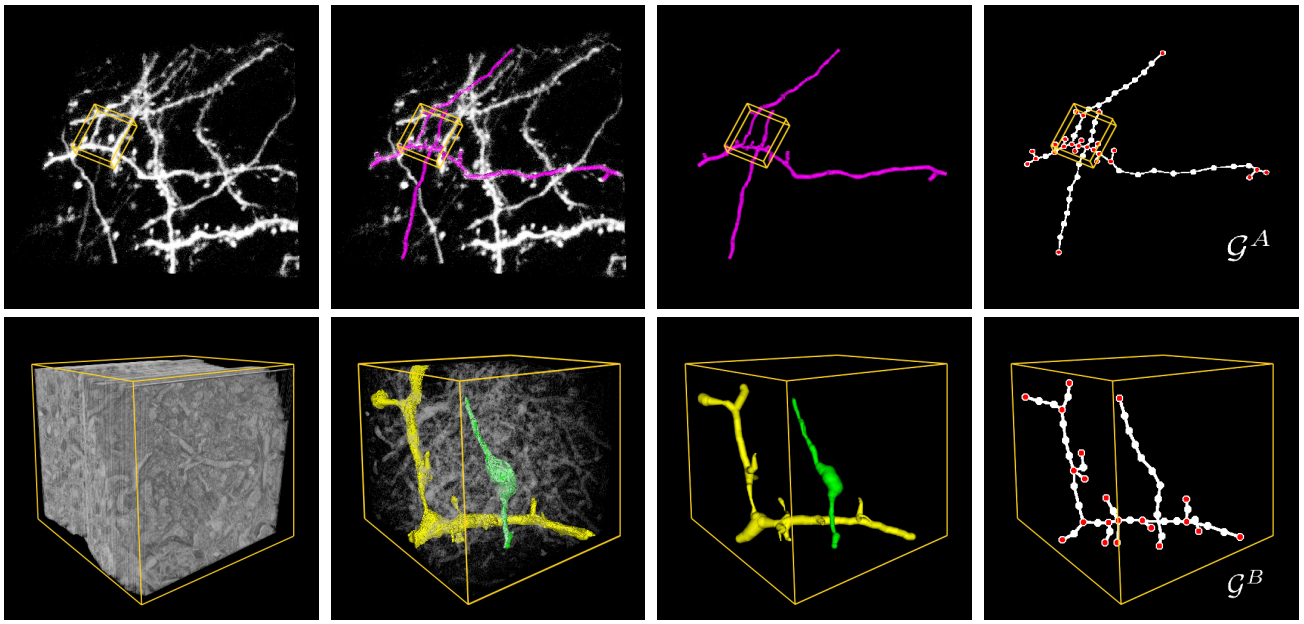
This kind of drastic appearance changes makes it impractical to use registration techniques that rely on maximizing image similarity [27], [40], in particular when the images are very different and when dealing with thin structures, such as blood vessels or neuronal fibers. The lack of distinguishing features of individual branching points or edges makes the use of feature-based correspondence techniques equally impractical. Since the graph geometrical 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 [12], [17], [35], 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 esti-

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

This work has been partially funded by the EU Micro Nano and Chist-Era ViSen projects, by the Spanish Ministry of Economy and Competitiveness under project PAU+ DPI2011-27510, and by the Czech Science Foundation project P202/11/0111.



(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.

mate, 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 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, as well as neural image-stacks acquired using different modalities.

This paper is an extended version of [31], where using Gaussian Processes was proposed, and [26], which introduced a preliminary version of the active testing strategy. 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 [27], [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 graphs 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 [14]. 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 [10] 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 [6]. Here, we propose to use an Active Search strategy [16], [26], [36], 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], [21], or Coherent Point Drift (CPD) [24]. 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 [8], [13], [17], [19], [20], [38]. Global nodal matches are then estimated using multidimensional optimization schemes such as graduated assignment [17], spectral techniques [19], [20], [38] or considering the graphs as an absorbing Markov chain [8]. 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 [13], [32]. 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 [35], or the retina [1], [12]. 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 those methods that simultaneously search for correspondences and estimate the transformation parameters using a Kalman filter-like approach [7], [11], [22], [29], [33], [34]. 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, cannot generalize to arbitrary deformations. Similar limitations are also shared by methods relying on implicit shape models [15], [25]. In the Gaussian Process framework we propose, we also progressively reduce the number of potential correspondences but, in contrast to previous approaches, no 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) [28] 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 [23] 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
β	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
π_t	Partial assignment selected at iteration t
$\mathcal{C}_{\pi_t}$	Set of children of the tree node π_t
S_{π_t}	Quality score of assignment π_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 by 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 $\pi = \left\{\mathbf{x}_l^A \leftrightarrow \mathbf{x}_l^B\right\}_{1 \leq l \leq n_c}$ 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 β 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 $\sigma_\pi^2(\cdot)$:

$$m_\pi(\mathbf{x}^A) = \mathbf{k}^T \mathbf{C}_\pi^{-1} \mathbf{X}_\pi^B \ , \quad (1)$$

$$\sigma_\pi^2(\mathbf{x}^A) = k(\mathbf{x}^A, \mathbf{x}^A) + \beta^{-1} - \mathbf{k}^T \mathbf{C}_\pi^{-1} \mathbf{k} \ , \quad (2)$$

where k is a kernel function, β^{-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 $[k(\mathbf{x}_1^A, \mathbf{x}^A), \ldots, k(\mathbf{x}_{n_c}^A, \mathbf{x}^A)]^T$, and $\mathbf{X}_\pi^B$ is the $n_c \times D$ matrix $[\mathbf{x}_1^B, \ldots, \mathbf{x}_{n_c}^B]^T$.

Among the different types of existing kernel functions [28], we chose the widely used summation of a squared-exponential, a constant term, and a linear term

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

We found this kernel to be the most appropriate for our purposes, because it implicitly defines a mapping function composed of an affine plus a non-linear transformation. This approximates well most of the warps appearing in biomedical imaging.

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 θ_0 and θ_1 hyperparameters, is a linear function of the input coordinates while the second one, which involves the θ_2 and θ_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 π. 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 $1 \leq i \leq n_A$

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

  where 'dist' is a Euclidean distance and H is the assignment matrix computed by the Hungarian algorithm [23].
- **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

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

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

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

5 Coarse Alignment

Let $\mathbf{X}^A = \left\{\mathbf{x}_1^A, \ldots, \mathbf{x}_{n_A}^A\right\}$ and $\mathbf{X}^B = \left\{\mathbf{x}_1^B, \ldots, \mathbf{x}_{n_B}^B\right\}$ 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 [31] and tested it successfully on relatively small graphs. However, as will be shown in Section 7, its computational requirements grow quickly with the number of graph nodes. The second strategy, while slightly more complex, scales better.

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

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

5.1 Greedy Search

Let $m_\pi(\cdot)$ be a GP written using the formulation of Section 4, which we instantiate by first randomly selecting only D correspondences (line 1 in Algorithm 1). This gives us an initial correspondence set π_0, which we use to get a rough estimation of the global scale and rotation. We then iteratively construct sets of correspondences in T steps as follows.

1) At iteration t, we have a set of correspondences π_t from which we compute (line 3) the mapping $m_{\pi_t}(\cdot)$ and covariance $\sigma^2_{\pi_t}(.)$ using Eqs. (1) and (2).
2) For each unmatched node $\mathbf{x}^A_i \in \mathbf{X}^A$, we search for potential correspondences $\mathbf{x}^B_j \in \mathbf{X}^B$ in the bounded region $\mathcal{B}_i$ determined by the predicted covariance $\sigma^2_{\pi_t}(\mathbf{x}^A_i)$ (lines 6–7). We use the Mahalanobis distance to define the boundary:

$$\mathcal{M}^2 = \left(m_{\pi_t}(\mathbf{x}^A_i) - \mathbf{x}^B_j\right)^T \left(\sigma^2_{\pi_t}(\mathbf{x}^A_i)\right)^{-1} \left(m_{\pi_t}(\mathbf{x}^A_i) - \mathbf{x}^B_j\right)$$

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

3) We choose the node $\mathbf{x}^A_{i^*}$ with the smallest number of potential candidates (line 9), and randomly pick one of them to define the match $\mathbf{x}^A_{i^*} \leftrightarrow \mathbf{x}^B_{j^*}$, which we add to the correspondence set π_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 π_t and continue searching (line 14).
4) We take the quality score S_{π_t} to be the number of inliers as defined in Eq. (6).

As described in Algorithm 1, this process is repeated T times, backtracking and selecting different correspondences using a depth-first search. We then return the correspondence set π^* with the highest score, and its corresponding m_{π^*}. We also terminate if the inlier consenus S_{π_t} becomes large enough. This process is depicted in Fig. 2.

The process is controlled by the noise parameter β of Eq. (2) and the vector $\mathbf{\Theta} = \{\theta_0, \theta_1, \theta_2, \theta_3\}$ 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 $\mathbf{\Theta}$ 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 π 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) [16], [36]. 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 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 π is part of the correct mapping π^* as ϵ_π. This list of candidates is handled by means of a priority queue Q (line

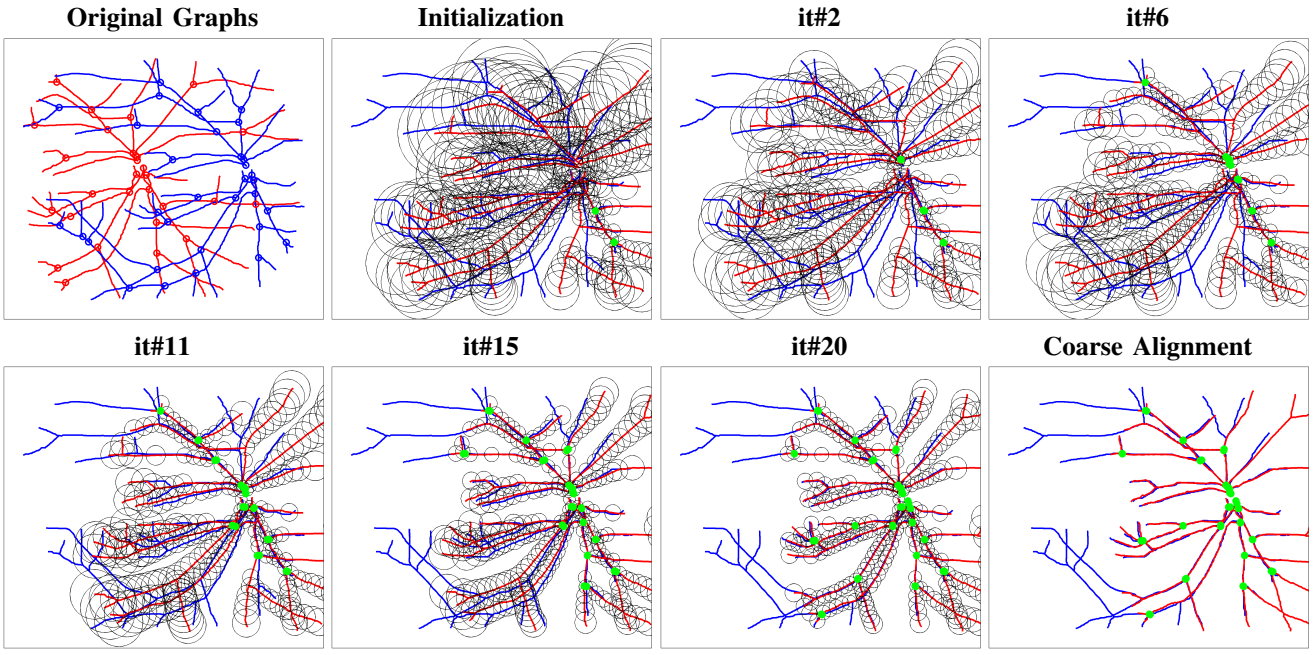


Fig. 2: **Coarse alignment steps.** The initial graph structures are depicted in the top left-most figure, the model graph in red and the target in blue. Exploration of the search space starts by picking randomly two correspondences, highlighted in green, thus roughly fixing scale and orientation. Then, 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.

1 of Algorithm 2), whose elements are (π, ϵ_π) 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 π from this initial set has equal probability ϵ.

At each ATS iteration, $t = 0, ..., T$, we want to select the candidate set π_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_{π_t} (line 5) to verify if it is indeed a good mapping. Given that S_{π_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. Simultaneously, we further refine our candidates by expanding the candidate set previously evaluated. In particular, from π_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 additional element of $\mathbf{X}^A$ considered at the depth below π_t.

As in [16], [36], the posterior for any element $\pi \in \mathcal{C}_{\pi_t}$ can be computed as

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

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 ϵ_{π_t} is a prior, $r(S_t)$ is the data-term and dividing by $|\mathcal{C}_{\pi_t}|$ attributes equal probability to all the expanded candidates of π_t.

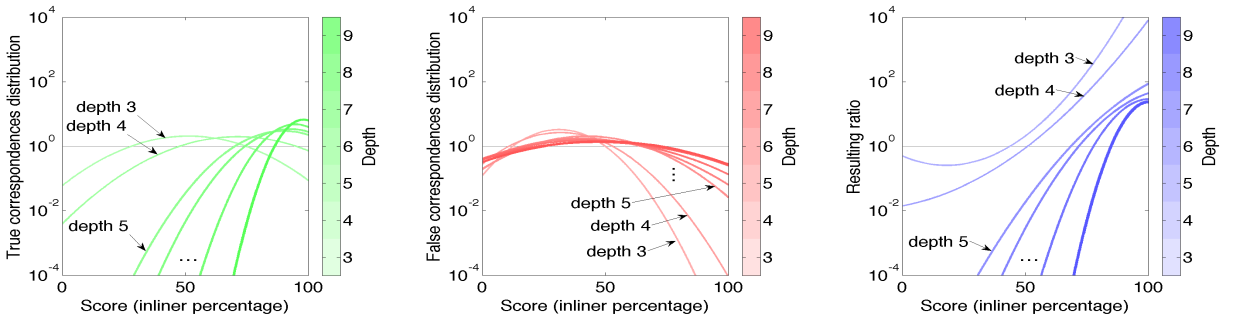
Once this process is repeated T times or until the likelihood ratio is higher than ψ, we return the assignment π^* with the best score.

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

1: Initialize Priority Queue:
   $Q \leftarrow$ 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: $\{\pi_t, \epsilon_t\}$ = Pop(Q)
4: $\{m_{\pi_t}, \sigma^2_{\pi_t}\}$ = ComputeMapping($\mathbf{X}^A, \mathbf{X}^B, \mathbf{\Theta}, \pi_t$)
5: S_{π_t} = QualityScore(m_{π_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: $Q \leftarrow$ Push $(\pi_t, \epsilon_{\pi_t} r(S_{\pi_t})/|\mathcal{C}_{\pi_t}|)$
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_π for any set of correspondences π we use the previously described assigned distance of Eq. (5) and the number of inliers of Eq. (6). In particular,



(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.

we compute S_π using the assigned distance when the number of correspondences $|\pi|$ is below a certain threshold γ, which we set to 5 in all experiments. Otherwise we compute S_π using the number of inliers. We found that using two different quality functions provides more informative scores for small and large sets of correspondences. This is similar to the strategies employed in [7].

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 π^* is the correct set of correspondence assignments, u is the number of correspondences in π, $\mathcal{N}$ is a Gaussian probability distribution with parameters $\boldsymbol{\omega}_1^u$, $\boldsymbol{\omega}_0^u$ and $\delta(\pi,\pi^*) = 1$ if the correspondences of π respect π^* 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 π^*. – generating 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 π^* and compute S_π. 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 π by starting from a subset of π^* and adding a few incorrectly matched points. Proceeding in this manner, we take false partial assignments which are close to the true correspondence π^* 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 π^*, along with the corresponding mapping $m(\cdot) = m_{\pi^*}$ and the covariance estimator $\sigma^2(\cdot) = \sigma^2_{\pi^*}$.

This set of matches π^* 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 [23], guided by the current mapping $m_{\pi_r}(\cdot)$ and covariance estimator $\sigma^2_{\pi_r}(\cdot)$, to establish new matches π_{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.

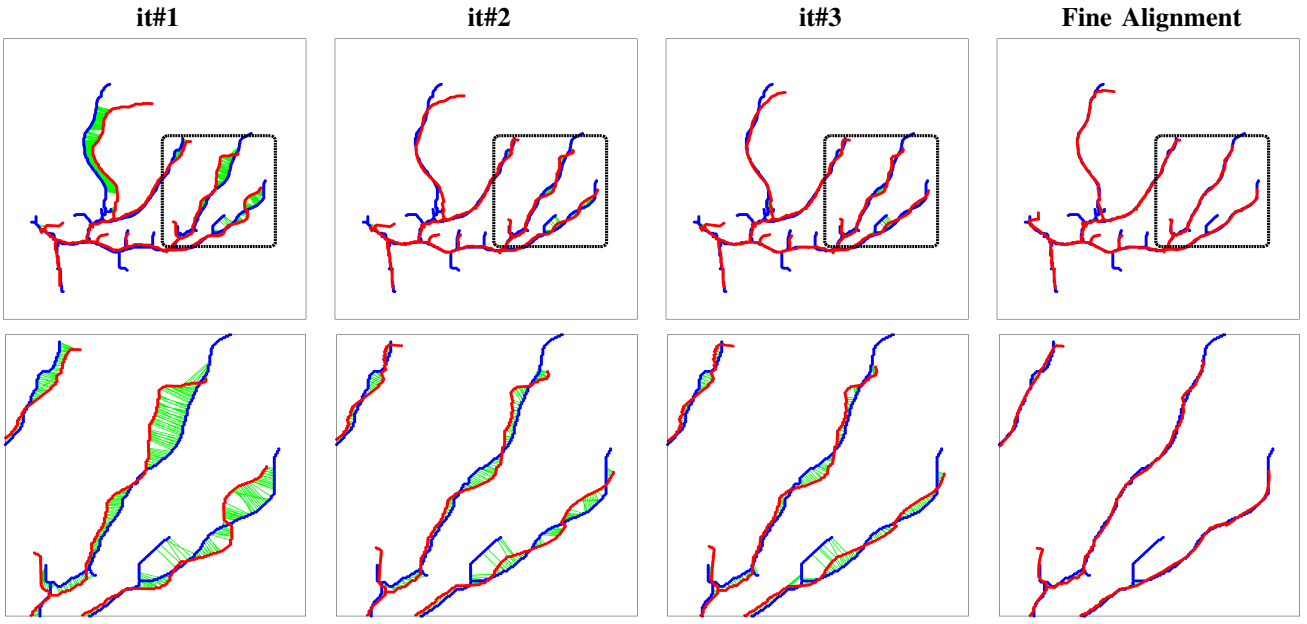


Fig. 4: **Fine alignment steps.** Once a coarse alignment of the two graphs (model in red and target in blue) has been found, the algorithm starts matching points lying on the edges. The assignments (depicted in green) are computed using the Hungarian algorithm and constrained by the graph topology and 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.

2) Given these new correspondences π_{r+1}, reestimate $m_{\pi_{r+1}}(\cdot)$ and $\sigma^2_{\pi_{r+1}}(\cdot)$.
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 π_r, $m_{\pi_r}(\cdot)$, and $\sigma^2_{\pi_r}(\cdot)$.

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

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: $\pi_{r+1} \leftarrow$ OptimalAssignment$(\mathbf{X}^A, \mathbf{X}^B, \mathbf{\Theta}, \pi_r)$
4: $\{m_{\pi_{r+1}}, \sigma^2_{\pi_{r+1}}\}$=ComputeMapping$(\mathbf{X}^A, \mathbf{X}^B, \mathbf{\Theta}, \pi_{r+1})$
5: $S_{\pi_{r+1}}$ = 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.

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) [9], the Coherent Point Drift (CPD) [24] and the recent Gaussian Mixture Model Registration (GMMREG) proposed in [18]. As examples of *graph matching* approaches, we have considered Spectral Matching (SM) [19] and Integer Projected Fixed Point (IPFP) [20], which can be combined, as well as Path Following Algorithm (PATH) [39]. 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.

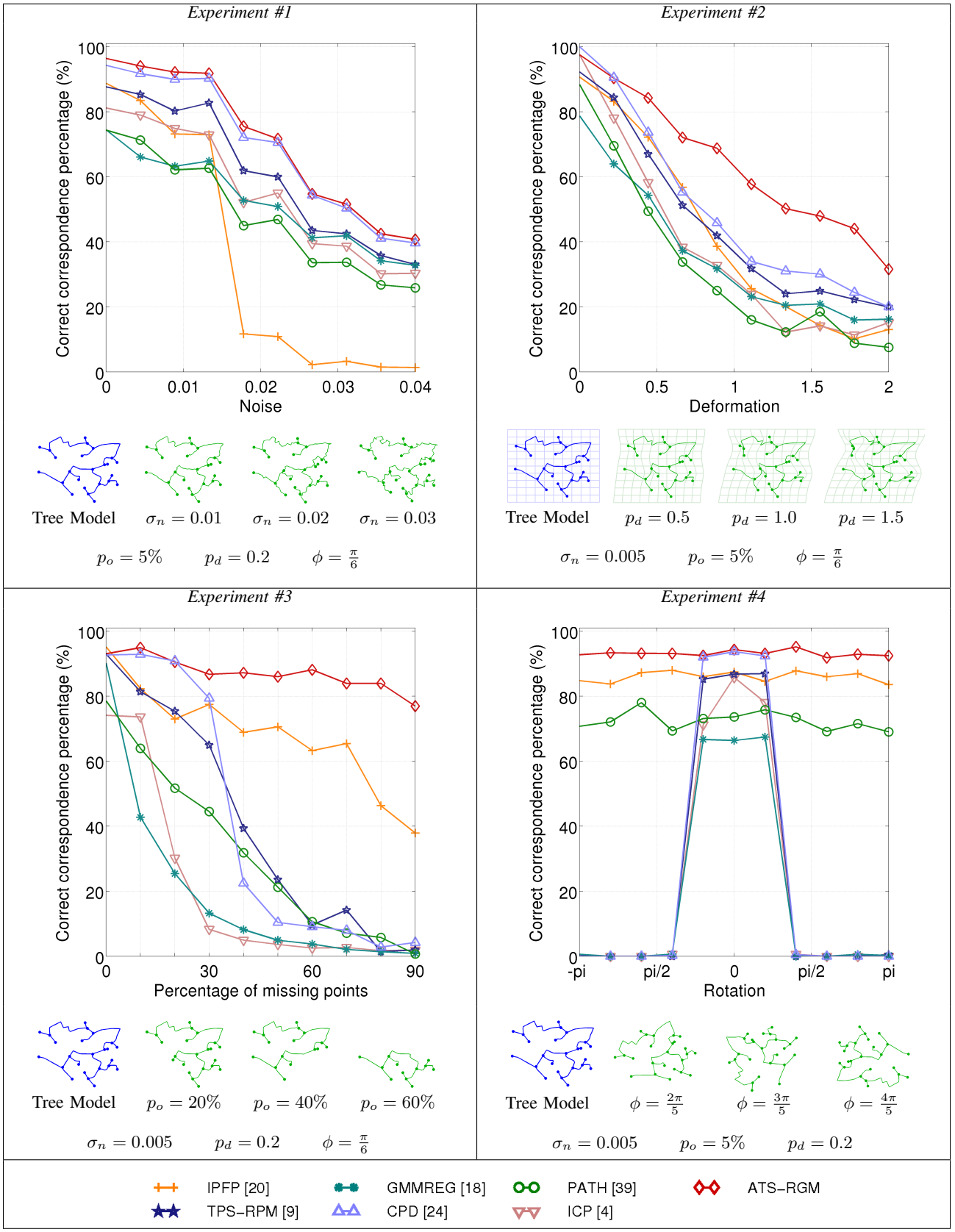


Fig. 5: **Quantitative evaluation on synthetic data**. Performance tests of all competing methods in different configurations of noise, deformation, outliers and rotation. The curves represent the median of the correct correspondences percentage achieved by each method. Below each result, we show the *tree model* used in all the experiments (in blue) and some corrupted samples to illustrate how each parameter affects the transformed graph (in green).

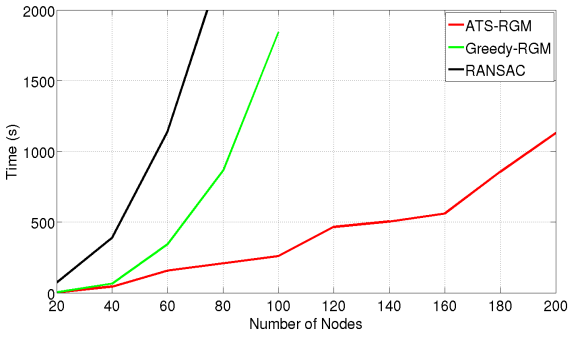


Fig. 6: **Computational Cost.** Processing time required by RANSAC, Greedy-RGM and ATS-RGM as a function of the number of nodes. We computed the median of 20 experiments for each of the methods.

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 2×2 bouding box and *target tree* obtained by deforming and randomizing 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

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

where the deformation includes the following components.

- Rotation, scaling, and translation; The target model is rotated by an angle ϕ, translated by T_x and T_y, and scaled by S_x and S_y
- 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 σ_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 set the required parameters to the values suggested in the corresponding papers, which we list below[1].

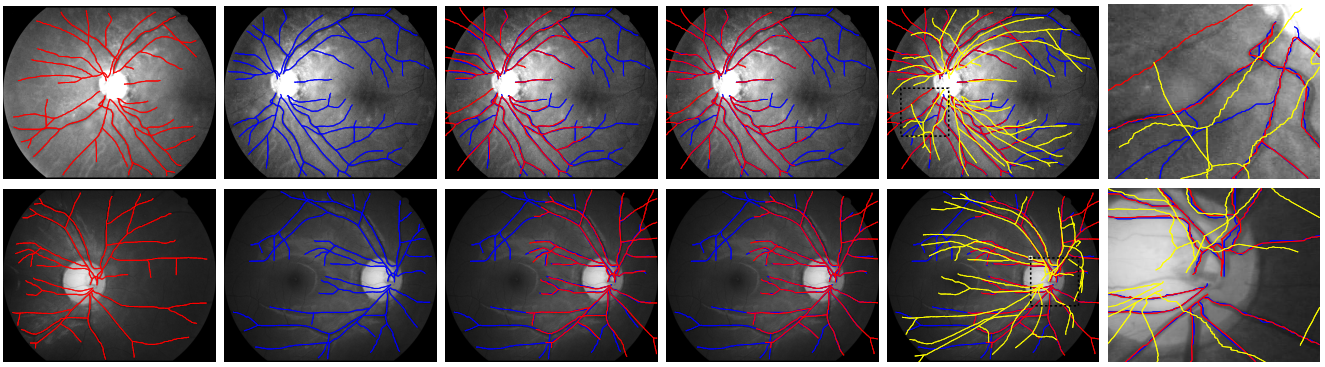
- ICP: Does not require extra parameters.
- TPS-RPM: We set the initial temperature T_{init} to half the maximum Euclidean distance between the nodes of the model tree, and the final temperature to $T_{final} = 0.01 \cdot T_{init}$. The remaining parameters are set to $\lambda_1^{init} = 1$ and $\lambda_2^{init} = 0.01$, as suggested by the authors.
- GMMREG: We set the maximum number of iterations to 10000, and α and β to 1 and 2, respectively.
- 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 [19]. As we do not have appearance information, we set the label affinity term $M(a, a)$ 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 –which turned to outlier tree branches– 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 the same model graph and different target graphs generated by varying the levels of noise, deformation, outliers and rotation.

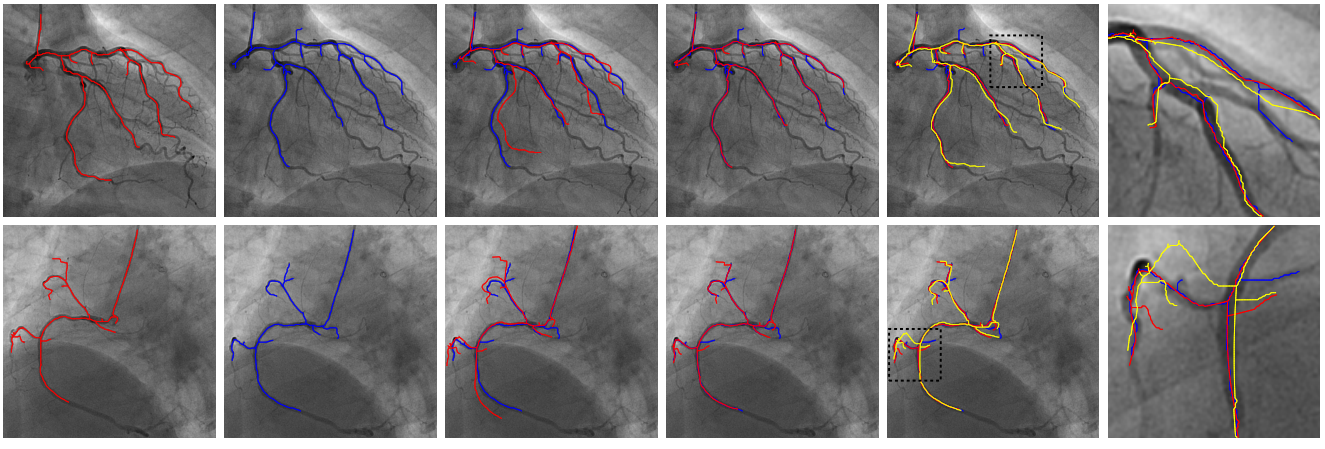
For each experimental parameter setting experiments and each algorithm, we computed the average percentage of

1. 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, σ_d) their meaning do not necessarily need to be the same.



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

Fig. 7: **Retinal fundus images used in [12].** 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.

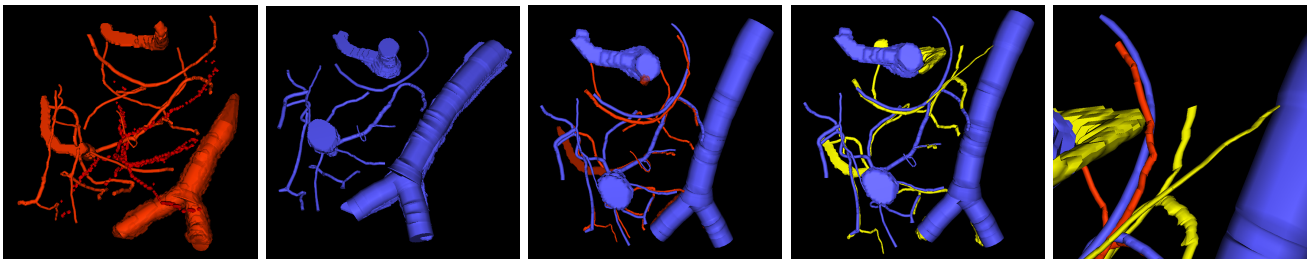


(a) (b) (c) (d) (e) (f)

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.

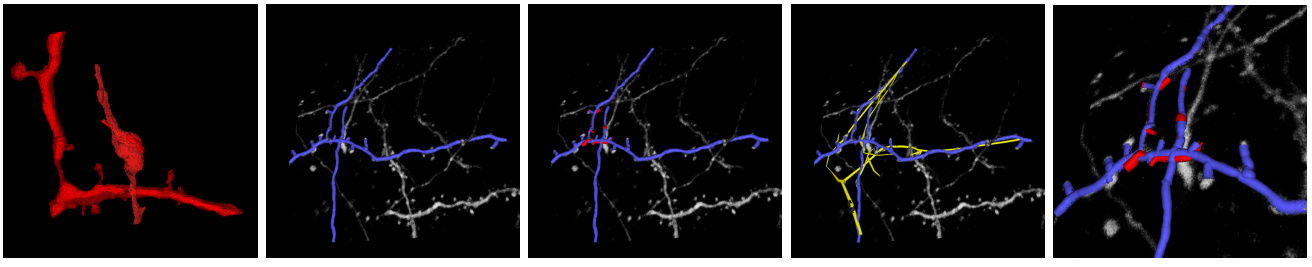
correct matches over our 50 tree pairs and plot the results in Fig. 5. Under favorable conditions, that is, relatively small graphs with less than 50 nodes, uncorrupted data and purely affine transformation, all methods exhibit similar performance. However, when we progressively introduce artifacts the 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,



(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.

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 [14]. 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.

7.2 Real 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 [30].

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. Observe for example, the assignments (in green) obtained in the two images of Fig. 11, for the Greedy-RGM and CPD methods.

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.

In Fig. 7 we show registration results for retinal fundus vascular graphs that are deformed from one image to

Dataset		ATS-RGM	Greedy-RGM	CPD [24]	ICP [4]	TPS-RPM [9]	GMMREG [18]
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 (μm)	**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

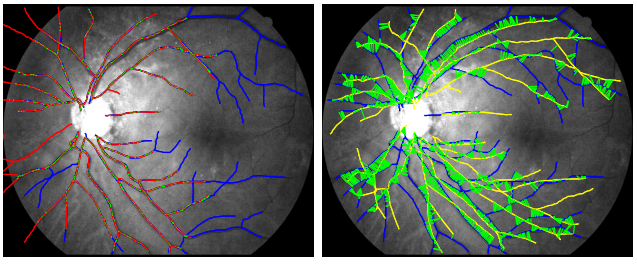
TABLE 2: **Error:** Geometric error on real datasets for the proposed approach and other state of the art methods. Failed experiments (producing incorrect alignment) are marked with an *, see Fig. 7(d) or Fig. 10(d) for examples. **Elapsed Time:** Processing time for each method, in seconds. For ATS-RGM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

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 [12], 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.



(a) (b)

Fig. 11: **Accuracy Measure.** Two alignment solutions computed from the retina dataset by Greedy-RGM **(a)** and CPD **(b)**. The assignments (in green) between the points of the two graphs after registration are used to compute the residual distance error. Best viewed in color.

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 area 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 to correctly align biological structures that are non-linearly transformed and extracted with different techniques, without the need of prealigning them, in a manageable amount of time. We have also shown in quantitative experiments that our method consistently outperforms state-of-the-art.

As future work, we will explore how we can include graph constraints to further reduce the computational cost of the search, allowing for a registration of bigger non-linear structures.

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

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

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

[9] 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.

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

[11] A.J. Davison. Active Search for Real-Time Vision. In *International Conference on Computer Vision*, pages 66–73, 2005.

[12] 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.

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

[14] 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.

[15] 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.

[16] 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.

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

[18] 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.

[19] 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.

[20] 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.

[21] 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.

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

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

[24] 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.

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

[26] 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.

[27] 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.

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

[29] 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.

[30] 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. *Nature Methods*, 9(7):676–682, 2012. Code available at http://pacific.mpi-cbg.de.

[31] 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*, June 2012.

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

[33] 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.

[34] 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.

[35] 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.

[36] 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.

[37] B. Tordoff and D.W. Murray. Guided Sampling and Consensus for Motion Estimation. In *European Conference on Computer Vision*, pages 82–98, 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.

Raphael Sznitman
EPFL IC CVLAB
Station 14
CH-1015 Lausanne
Switzerland

August 20, 2013

Dear Editor

It is our pleasure to submit our article named "Non-Rigid Graph Registration using Active Testing Search" for review. This article is a combined and extended version of two earlier conference papers (CVPR and IPMI), which we submit as supplementary material. The most significant differences between this submission and the earlier papers are listed below.

- We have substantially improved the Active Testing Strategy with respect to the one introduced in the IPMI paper. We also refined its description and now provide a much more thourough mathematical justification for it.
- We compare to more competing techniques on real-data and performed a much more extensive set of synthetic experiments to showcase the strenghs of our proposed approach and analyze its behavior in the presence of different kinds of perturbations.
- We have unified the notations between the two conference papers and have attempted to provide a consistent whole.

Please note that our manuscript includes several color images and we believe that color printing would be of great value for the clarity of these figures, especially when several graphs are overlaid on top of each other or on top of a black and white image.

Sincerely,

On behalf of the authors,
Raphael Sznitman

Active Testing Search for Point Cloud Matching

Miguel Amável Pinheiro[1], Raphael Sznitman[2], Eduard Serradell[3], Jan Kybic[1], Francesc Moreno-Noguer[3], and Pascal Fua[2]

[1] Center for Machine Perception, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic
amavemig@cmp.felk.cvut.cz
http://cmp.felk.cvut.cz/~amavemig
[2] Computer Vision Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[3] Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain

Abstract. We present a general approach for solving the point-cloud matching problem for the case of mildly nonlinear transformations. Our method quickly finds a coarse approximation of the solution by exploring a reduced set of partial matches using an approach to which we refer to as *Active Testing Search* (ATS). We apply the method to registration of graph structures by branching point matching. It is based solely on the geometric position of the points, no additional information is used nor the knowledge of an initial alignment. In the second stage, we use dynamic programming to refine the solution. We tested our algorithm on angiography, retinal fundus, and neuronal data gathered using electron and light microscopy. We show that our method solves cases not solved by most approaches, and is faster than the remaining ones.

Keywords: point cloud matching, graph matching, image registration, active search, dendrites.

1 Overview

In this manuscript we consider the problem of point-cloud to point-cloud (PTP) matching. The problem consists of finding correspondences between two populations of points, related by a geometrical transformation. The transformation is assumed to be non-linear but not far from affine. The correspondences can be partial. We do not require an initial alignment nor any additional information except the point coordinates. However, if such information is available (e.g. local appearance or connectivity), it can be incorporated to reduce the search problem.

The main difficulty of the PTP problem is the large set of possible matches. The major challenge lies in the ability to formulate a search procedure that is tractable and still provides an acceptable solution. This is particularly true when the transformation between the two populations is non-rigid.

We consider this problem in the context of medical image registration. Three important challenges lie in such registration tasks. First, the transformation

J.C. Gee et al. (Eds.): IPMI 2013, LNCS 7917, pp. 572–583, 2013.
© Springer-Verlag Berlin Heidelberg 2013

between curvilinear structures is generally non-rigid, which induces complex solutions that are difficult to compute. Second, appearance based measures of similarity (e.g. key point descriptors) cannot be used in some cases due to the fact that registration may be between different modalities (e.g. Electron Microscopy (EM) and Light Microscopy (LM)) [1]. Finally, registration may be at different physical scales (e.g. nm and μm) and hence consists of registering one domain to a substructure of another much larger structure.

In our approach, *Active Testing Search* (ATS) we take a Bayesian point of view and consider the correspondences to be random. We use a *sensor*: a black box function, which scores the quality any set of partial or complete point correspondences. The probability of the correctness of the match given a sensor output is given by a *sensor model*, which we learn from data. We make observations sequentially and integrate information received from the sensor by computing the posterior probability of the correspondence correctness. We explore the space of possible potential correspondences by performing a priority search based on the information gain, adding one point match per step, similar to the *Twenty Questions* game with noisy outputs [2–5].

2 Related Work

The difficulty of registering medical images lies on the nonlinearity between structures and also high number of outliers, such as in the case of EM and LM images. These structures can be interpreted as point clouds or as graphs. In the first case, some authors have proposed transformation minimizations between the sets [6, 7], which however fails when the sets are not roughly aligned or when the number of outliers is too high. Another approach to this problem is to try to find the correct correspondence between the points [8–10].

Another popular approach is ICP (Iterative Closest Point) [11], which iteratively calculates the closest distance between points, assigns correspondences and calculates the rigid transformation between the sets until convergence. The method and its variants [12, 13] also require the initial position of the sets to be relatively close.

Local similarities such as geometric compatibilities and feature descriptors could also help establish correspondences between points [14, 15]. However, in the presence of shearing and nonrigid transformations, the approach proves to be sensitive.

Using graph information can provide further constraints in the problem, such as local connectivity and geodesic distance preservation [16, 1]. However, most of these approaches are either not robust enough to solve harder cases [17] or are not scalable [1].

3 Notation

Consider two sets of points $X^A = \{x_1^A, ..., x_N^A\}$ and $X^B = \{x_1^B, ..., x_M^B\}$ of size N and M respectively, with $x_i^A \in \mathbb{R}^{D_A}$ and $x_j^B \in \mathbb{R}^{D_B}$. We want to find a matching where each element x_i^A of X^A maps to at most one element of X^B, which is

Table 1. Summary of Notation

$X^A = \{x_1^A, ..., x_N^A\}$	Source point cloud
$X^B = \{x_1^B, ..., x_M^B\}$	Target point cloud
$Y = (Y_1, ..., Y_N)$	Correspondences for X^A
$Y^* = (Y_1^*, ..., Y_N^*)$	True correspondences
$\mathcal{Y}$	Space of feasible correspondences
$A = \{Y_1 = y_1, ..., Y_d = y_d\}$	Partial assignment
$\mathcal{A}_d$	Set of partial assignments of length d
$\psi(A)$	Sensor function
$(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0)$	Sensor noise model parameters
γ	Minimum number of required assignments for ψ
S_A	Sensor response for set A
$r(S_A)$	Sensor likelihood ratio
K	Total number of iterations
π_k	Assignment to evaluate at iteration k
$\boldsymbol{\pi} = (\pi_1, ..., \pi_K)$	Sequence of observations to make
$\mathcal{C}_A$	Set of children of A

represented by an index $Y_i \in \{-1, 1, \ldots, M\}$ to X^B, with a virtual element of index -1 meaning no match (an outlier). We consider $Y = (Y_1, ..., Y_N) \in \mathcal{Y}$, where $\mathcal{Y}$ is the space of all possible solutions, to be a discrete random vector, with probability $P(Y) = P(Y_1, ..., Y_N)$. Note that the ordering of Y is important. Our objective is to find Y^*, the true correspondence between X^A and X^B.

A *partial assignment* is a vector $A = (Y_1 = y_1, ..., Y_d = y_d)$, where we require the correspondences to be determined in order. We denote $\mathcal{A}_d$ the set of all possible partial correspondences of d elements.

The sets $\mathcal{A}_d$ can be organized hierarchically into a tree, where children are formed from parents by adding one additional match. The children of A are $\mathcal{C}_A = \{A \cup \{Y_{|A|+1} = y\}\}$, with $y \in \{-1, 1, \ldots, M\}, y \notin A$.

A sensor is a task specific function $\psi : \mathcal{A} \to \mathbb{R}$ such that $S_A = \psi(A)$ evaluates a partial correspondence A for $|A| \geq \gamma$, where γ is the minimum number of matches required to calculate ψ. Let $\boldsymbol{\pi} = (\pi_1, ..., \pi_K)$ be the sequence of subsets observed throughout the algorithm, where $\pi_k \in \mathcal{A}$, is the k^{th} set of partial assignments to observe.

4 Objective

Our objective is to estimate Y^* from some observation S_{π_k}. To do this, we consider solving the MAP,

$$Y^* = \arg\max_{y \in \mathcal{Y}} P(Y | S_{\pi_1}, \ldots, S_{\pi_K}) = \arg\max_{y \in \mathcal{Y}} \left\{ \frac{1}{Z} P(Y) P(S_{\pi_1}, \ldots, S_{\pi_K} | Y) \right\}$$
$$= \arg\max_{y \in \mathcal{Y}} \left\{ \frac{1}{Z} P(Y) \prod_{i=1}^{K} P(S_{\pi_k} | Y) \right\}, \quad (1)$$

where Z is a constant factor.

Clearly, considering all possible correspondences in $\boldsymbol{\pi}$ is intractable. RANSAC [8] and MLESAC [9] can been viewed as solving Eq. 1 when $\boldsymbol{\pi}$ contains only randomly chosen partial assignments of fixed size (i.e. $\forall k, |\pi_k| = \text{const}$, depending on the number of degrees of freedom of the transformation).

In our approach, we differ from RANSAC and MLESAC in two important ways. First, π_k are selected sequentially and on the fly, based on the previous values observed from $\pi_1, \ldots, \pi_{k-1}$. This makes our selection process adaptive and fully data-driven. Second, to allow maximum flexibility with respect to the types of possible correspondences (i.e. non-rigid transformations), we let $|\pi_k|$ vary; it will typically increase as the transformation is refined and which is vital for estimating correspondences for non-rigid transformations..

5 Active Testing Search

Our method attempts to approximately solve the MAP of Eq. 1. To do this, we begin with a prior on Y, observe π_1 using our chosen sensor ψ, compute the posterior distribution of Y given the new information, S_{π_1}, and select the most promising new set π_2 to evaluate based on the posterior distribution. This process repeats K times and the best correspondence set, defined as the set with the highest number of inliers, is retained.

5.1 Sensor and Sensor Model

As described previously, our sensor is a function $\psi : \mathcal{A} \rightarrow \mathbb{R}$, with a random response $\psi(A) = S_A$. We assume the following model

$$P(S_A = s_A|Y) = \begin{cases} \xi(S_A = s_A; \boldsymbol{\theta}_1^d), & \text{if } A \subset Y^* \\ \xi(S_A = s_A; \boldsymbol{\theta}_0^d), & \text{if } A \not\subset Y^* \end{cases} \tag{2}$$

where $d = |A|$, $\xi(S_A = s_A; \boldsymbol{\theta}_1^d)$ and $\xi(S_A = s_A; \boldsymbol{\theta}_0^d)$ are respectively the *positive* and *negative* distributions and $\boldsymbol{\theta}_1^d$ and $\boldsymbol{\theta}_0^d$ its parameters. We also define likelihood ratio

$$r(s_A) = \frac{\xi(S_A = s_A\ ; \boldsymbol{\theta}_1^d)}{\xi(S_A = s_A\ ; \boldsymbol{\theta}_0^d)}. \tag{3}$$

The sensor score implicitly characterizes the expected geometrical transformations and depends directly on the number of assignments d in A. For simplicity, we will assume $\xi(\cdot; \boldsymbol{\theta}_1^d)$ and $\xi(\cdot; \boldsymbol{\theta}_0^d)$ to be Gaussian and we will describe in Sec. 7 how the parameters of these distributions can be obtained from training data. Using the Gaussian Processes non-linear regression (GPR) described in [1], we can estimate the position of a match of a point x_i^A in X^A, which we denote $\bar{x}_i^A$. The GPR models the geometrical transformation as affine with a small random nonlinear component, which is spatially correlated and its amplitude is controlled by a parameter σ_n^2. Note that the prediction is based on a partial assignment A.

We have used GPR to generate the following two sensors:

Assigned Distance. We use the predictions from GPR to define the total cost of assigning the points $\{\bar{x}_i^A\}$ to X^B

$$S_A = \sum_{i=1}^{N}\sum_{j=1}^{M} H_{i,j} \cdot \mathrm{dist}(\bar{x}_i^A, x_j^B), \tag{4}$$

where $\mathrm{dist}(\bar{x}_i^A, x_j^B)$ is the Euclidean distance between $\bar{x}_i^A$ and x_j^B and H is the optimal assignment matrix computed by the Hungarian algorithm [18] so that S_A is minimal. We make use of an assignment so that we penalize situations where $\bar{x}_i^A$ is positioned solely around a subset of small size of X_B.

Number of inliers. We also calculate the relative number of points consistent with the GPR. This is calculated as the ratio over $|X^A|$ of the number of points in X^B which have some point $\{\bar{x}_i^A\}$ closer than σ_n^2,

$$S_A = \frac{|I|}{|X^A|}, \quad I = \left\{ x_j^B \in X^B \mid \exists \bar{x}_i^A, \mathrm{dist}(x_j^B, \bar{x}_i^A) < \sigma_n^2 \right\}. \tag{5}$$

5.2 Hierarchical Search

In many datasets, we can select a smaller number of important points B^A from all points X^A to be matched, $B^A \ll X^A$. For example, in a dataset created by segmenting a dendritic tree, the branching points are structurally more important than points on the edges connecting the branching points.

Our strategy then is to use the sensor $S_A = \psi(A)$ from (4) only on the 'important' points B^A, for 'small' partial matches A where $|A| < \delta$. For partial matches bigger than δ, we switch to the sensor (5) evaluated on the full set of points X^A. This allows for a fast search at low depths of the search tree, which constitutes most of the evaluated proposals π_k, and a more discriminative selection at higher depths.

5.3 Computing Posterior Probability Distributions

In this setting, aggregating observations can be achieved by using a Bayesian formulation. We can compute the posterior distribution when π_k has been observed by

$$P(Y|S_{\pi_1}, \ldots, S_{\pi_k}) = \frac{1}{Z}\Big[r(S_{\pi_k}) \mathbb{1}_{\pi_k \subset Y} + \mathbb{1}_{\pi_k \not\subset Y} \Big] P(Y|S_{\pi_1}, \ldots, S_{\pi_{k-1}}), \tag{6}$$

where

$$Z = r(S_{\pi_k}) P(\pi_k) + 1 - P(\pi_k) \tag{7}$$

and $r(S_{\pi_k})$ is defined in Eq. 3. There are two important aspects of (6). First, it is recursive, allowing the posterior $P(Y \mid S_{\pi_1} \ldots S_{\pi_k})$ to be computed from the previous posterior. This allows online integration of new information. Second, the normalization factor Z is independent on Y and can therefore be ignored when comparing the likelihood of different hypotheses Y.

5.4 Implementation and Algorithm

The search method is given in Algorithm 1. The probabilities $P(Y|S_{\pi_1}, \ldots, S_{\pi_k})$ are stored in a priority queue Q (line 1). Initially, this queue will hold all the elements of the subspace $\mathcal{A}_\gamma$ with the same likelihood $\epsilon = 1/|\mathcal{A}_\gamma|$ of being contained in the true set of correspondences (i.e. uniform prior on Y). The priority queue is ordered by the likelihood ϵ that a partial assignment is correct.

Algorithm 1. Active Testing Search $(X^A, X^B; K, \psi, \boldsymbol{\theta}_1, \boldsymbol{\theta}_0, \gamma)$

1: Initialize Priority Queue: $Q \leftarrow Push(A, 1/|\mathcal{A}_\gamma|)$, $\forall A \in \mathcal{A}_\gamma$
2: **for** $k = 1 \ldots K$ **do**
3: $\{\pi_k, \epsilon_k\} = \text{pop}(Q)$; //choose the most likely π_k
4: $S_{\pi_k} = \psi(\pi_k)$
5: **for** $y \in \mathcal{C}_{\pi_k}$ **do**
6: $Q \leftarrow Push(\pi_k \cup \{Y_{|\pi_k|+1} = y\}, \epsilon_k r(S_{\pi_k})/|\mathcal{C}_{\pi_k}|)$
7: **end for**
8: **end for**
9: **return** $\pi^* = \arg\max_{\{\pi_1, \ldots, \pi_K\}} S_{\pi_k}$

For each iteration k, we select the partial assignment with the biggest likelihood ϵ_k. We use the sensor and compute the noisy score $S_{\pi_k} = \psi(\pi_k)$. At this point we must compute the posterior distribution given this new observation. To do this, we first generate children $\mathcal{C}_{\pi_k}$ of π_k and insert them into the queue using (6) (line 6). The queue maintains an unnormalized posterior distribution to avoid unnecessary computational costs. This process is repeated K times, at which point we return the assignment π^* which scored the highest. Our method does not perform a breadth-first, or depth-first search as in traditional search strategies. Rather, it is an adaptive strategy which allows constant backtracking and avoids hand-tune pruning of the search space.

6 Fine Alignment

Depending on the choice of K, Algorithm 1 will find only a subset of all inliers. A fine alignment can be added as a post-processing stage, to identify remaining inliers and if possibly slightly modifies the transformation. An algorithm such as the coherent point drift [7] is very well suited for this task. We use the approach described in [1], which locally finds assignment of the yet unassigned points by the Hungarian algorithm [18], using the already assigned points as constraints. The GPR transformation model is updated and the process is iterated until convergence.

7 Learning the Distributions

Given a specific sensor ψ, as described in Sec. 5.1, we need to learn the sensor model parameters. To reduce the number of degrees of freedom, we assume that

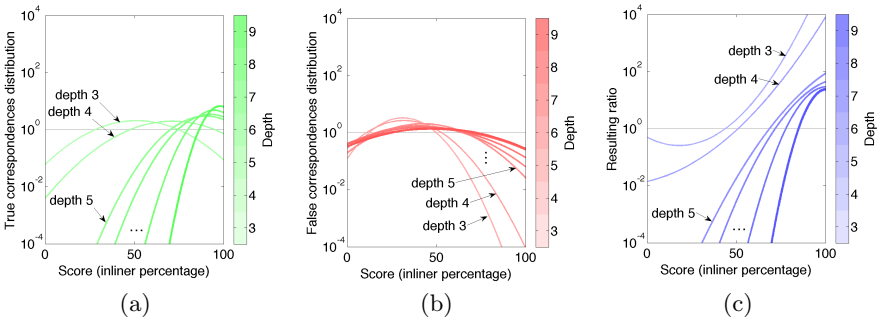


Fig. 1. Example for values of distributions taken from **(a)** true correspondences samples, **(b)** false correspondences and **(c)** ratio between the true and false distributions. The sensor used to compute this example was the number of inliers – described in Eq. 5.

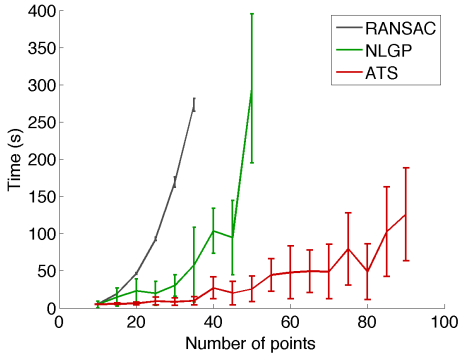


Fig. 2. Processing time required by RANSAC and NLGP in comparison to our method as a function of the number of points

it does not depend on the number of points M of X^B, at least when M is of the same order of magnitude as N.

In general, for a given sensor ψ, an outlier ratio R_O and a point set size N, we learn the parameters $\boldsymbol{\theta}_1^d$ and $\boldsymbol{\theta}_0^d$ as follows: we generate L point clouds X^A and L random affine transformations, together with a nonlinear deformation to each point from which we compute X^B and for which we know the correspondence Y^* – generating a set $\{\{X^A\}_l, \{X^B\}_l, Y_l^*\}_{l=1}^L$. Then, for $\gamma \leq d \leq N$, we sample assignments $A \in \mathcal{A}_d$ such that $A \in Y$ and compute S_A. Once all $N - \gamma$ scores on all L generated sets are computed, we estimate the Gaussian distribution parameters $\{\boldsymbol{\theta}^d\}_{d=\gamma}^N = \{\mu_d, \sigma_d\}_{d=\gamma}^N$. The learned probability densities can be seen in Fig. 1(a).

For the distribution of false correspondences we follow a similar sampling approach. However, especially for larger correspondences deeper in the tree, we

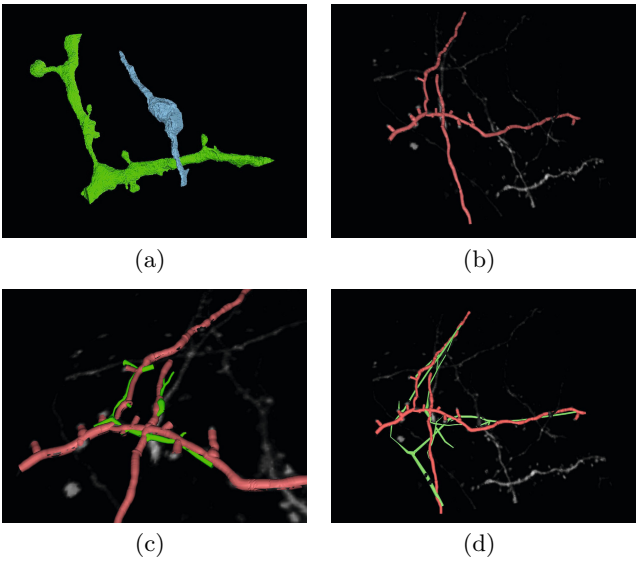


Fig. 3. Light and electron microscopy neuronal trees. **(a):** Segmented electron microscopy data. **(b):** Segmented light microscope data. **(c):** Registration of structures using ATS. **(d):** Registration using CPD.

will mostly encounter correspondences composed of mostly true correspondences, except for the last one. Therefore, we sample many random false correspondences at lower depths and false correspondences close to the true ones at higher depths. An example of such distribution can be seen in Fig. 1(b).

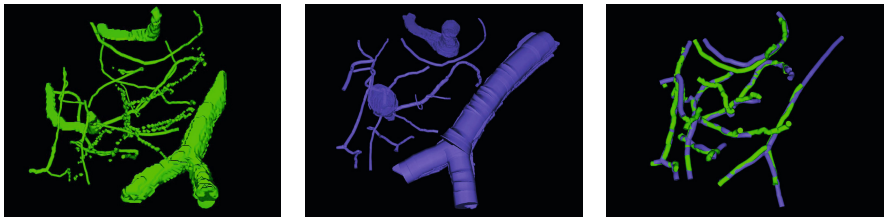
In Fig. 1(c), we can see the likelihood ratio between the distributions for true and false correspondences. This shows that the sensor gets more discriminative as the size of the partial correspondence being tested increases.

8 Experiments

We present a number of experiments to illustrate the performance of our method (Active Testing Search – ATS) against state of the art approaches in point matching, with or without additional structure information. We have tested: Non-Linear GP (NLGP) [1], Coherent Point Drift (CPD) [7], Iterative Closest Point [11] and RANSAC [8]. For RANSAC, we test affine transformations from the random branching points, applying the result on all the nodes.

8.1 Experiments on Synthetic Data

We generated random two dimensional point data sets with random affine transformation. For the NLGP that requires connection information, a Minimum Spanning Tree was found. Observe that the processing time (Fig. 2) increases much faster with both NLGP and RANSAC, while for ATS it stays manageable.



(a) (b) (c)

Fig. 4. 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.

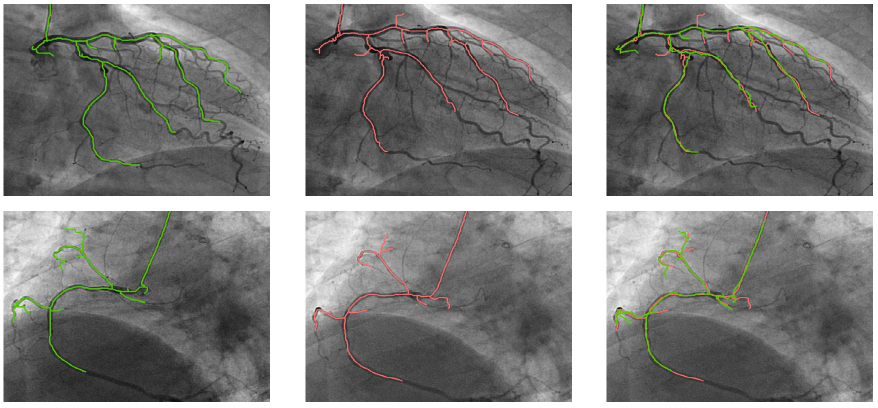


Fig. 5. Matching and registration of heart angiograms. **Left and center:** Original data. **Right:** Registration of structures using Active Testing Search.

8.2 Experiments on Real Data

A variety of datasets from medical imaging was collected. The graphs were extracted through a semi-automatic approach using the Fiji[1] platform and its plugins. In Tables 2 and 3, we can see the error obtained and times elapsed, respectively, for every method tested. The computed error is the mean distance of each point to its nearest neighbor in the original data to which the graph was matched. For ATS and NLGP, we show the coarse and fine alignment algorithm times separately.

Fig. 3 shows 3D neuronal structures from electron (EM) and light (LM) microscopy, where electron microscopy data is a nonlinearly deformed subset of the data light microscopy. The intended application is to automatically localize the EM volume in the LM volume. Only ATS and NLGP are able to correctly align the structures. CPD obtains a numerically small error, but as seen

[1] `http://pacific.mpi-cbg.de`

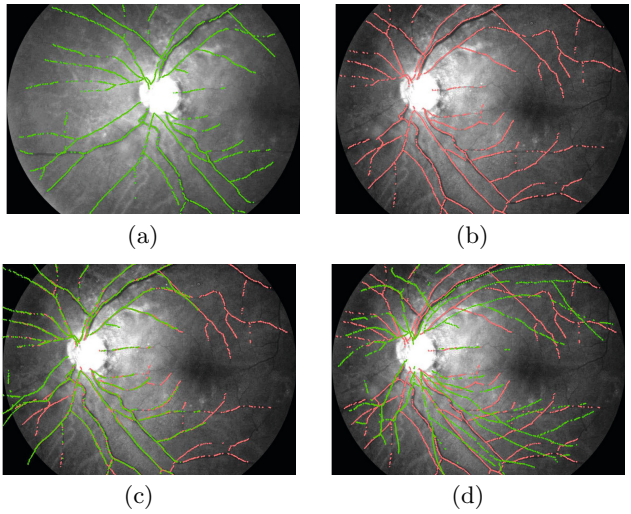


Fig. 6. Retinal fundus tree images. **(a) and (b):** Original data. **(c):** Registration of structures using Active Testing Search, **(d):** Registration using CPD.

in Fig. 3, the alignment is not correct and the resulting deformation is not realistic. The great advantage of ATS over NLGP is the much shorter elapsed time (Table 3).

In Fig. 4, a blood vessels network in brain tissue is imaged. One of the 3D image stacks is acquired using a two photon microscope and the other using bright-filed microscopy after excision and fixation.

In Fig. 5 and Fig. 6, we show 2D datasets of heart angiograms and retinal fundus images. The angiograms present a nonlinear transformation which is correctly recovered by the approach. The retinal fundus images present a high number of outliers in both images. Nonetheless, the algorithm correctly identifies the alignment. For the data from retinal fundus, the remaining methods do not recover correctly the alignment, although CPD and ICP present a small error (Table 2).

Table 2. Real data error for ATS and other state of the art methods. * Not correctly aligned (see Fig. 3(d) or Fig. 6(d) for example).

Error (pixels)					
Dataset (Fig.)	**ATS**	**NLGP**	**CPD**	**ICP**	**RANSAC**
Neuronal (3)	0.161	0.181	0.563*	2.995*	0.449*
Brain tissue (4)	0.159	0.171	0.164	0.851*	0.606*
Angio. (5 Top)	1.361	1.178	1.232	1.430	12.487*
Angio. (5 Bottom)	2.072	2.074	2.195	2.122	35.384*
Retina (6)	5.587	5.613	5.503*	6.524*	10.762*

Table 3. Processing time for each method and each dataset, in seconds

Elapsed time (seconds)					
Dataset (Fig.)	**ATS (coarse+fine)**	**NLGP (coarse+fine)**	**CPD**	**ICP**	**RANSAC**
Neuronal (3)	42.4 + 15.8	116.1 + 18.2	22.2	28.2	606.9
Brain tissue (4)	593.7 + 55.5	15029.1 + 19.9	37.1	30.9	570.7
Angio. (5 Top)	307.8 + 129.4	1240.9 + 162.8	144.3	8.1	1608.1
Angio. (5 Bottom)	167.9 + 77.2	112.0 + 95.4	68.8	5.0	346.5
Retina (6)	1293.3 + 406.4	5998.9 + 336.8	580.2	24.3	8901.5

9 Conclusion

We presented a general approach for the exploration of a tree of possible correspondences between two sets of points, using partial assignments and a Bayesian model. We have shown how we can include graph constraints to reduce the number of points, allowing for a faster search. We have also shown that our method is able to correctly align biological structures that are nonlinearly transformed and extracted with different techniques. These structures need not to be prealigned. Our method finds the correct alignment for all considered datasets and is faster than NLGP and RANSAC. It allows a considerably faster exploration of correspondences over the method which correctly finds a solution for harder datasets.

Acknowledgments The authors would like to acknowledge the Fundação para a Ciência e Tecnologia (FCT) for the Ph.D. grant SFRH/BD/77134/2011. This work was also supported by the Czech Science Foundation under the project P202/11/0111, by the Grant Agency of the Czech Technical University in Prague under the grant SGS12/190/OHK3/3T/13, by the EU ERC project Micro-Nano, and also by the Spanish Ministry of Economy and Competitiveness under projects PAU+ DPI2011-27510 and MIPRCV Consolider Ingenio 2010 CSD2007-00018.

References

1. Serradell, E., Glowacki, P., Kybic, J., Moreno-Noguer, F., Fua, P.: Robust non-rigid registration of 2D and 3D graphs. In: IEEE CVPR, pp. 996–1003 (2012)
2. Geman, D., Jedynak, B.: An active testing model for tracking roads in satellite images. IEEE Trans. on Pattern Analysis and Machine Intelligence 18, 1–14 (1995)
3. Yuille, A., Coughlan, J.: Twenty questions, focus of attention, and A*: A theoretical comparison of optimization strategies. In: International Workshop on Energy Minimization Methods in CVPR, pp. 197–212 (1997)
4. Sznitman, R., Jedynak, B.: Active testing for face detection and localization. IEEE Trans. on Pattern Analysis and Machine Intelligence 32(10), 1914–1920 (2010)
5. Sznitman, R., Richa, R., Taylor, R.H., Jedynak, B., Hager, G.D.: Unified detection and tracking of instruments during retinal microsurgery. IEEE Trans. on Pattern Analysis and Machine Intelligence 99, 1 (2012)

6. Gold, S., Rangarajan, A., Lu, C.P., Mjolsness, E.: New algorithms for 2D and 3D point matching: Pose estimation and correspondence. Pattern Recognition 31, 957–964 (1997)
7. Myronenko, A., Song, X.: Point set registration: Coherent point drift. IEEE Trans. on Pattern Analysis and Machine Intelligence 32(12), 2262–2275 (2010)
8. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24, 381–395 (1981)
9. Torr, P.H.S., Zisserman, A.: MLESAC: A new robust estimator with application to estimating image geometry. Computer Vision and Image Understanding 78, 138–156 (2000)
10. Chum, O., Matas, J.: Matching with PROSAC – progressive sample consensus. In: IEEE CVPR, pp. 220–226 (2005)
11. Besl, P.J., McKay, N.D.: A method for registration of 3-D shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence 14(2), 239–256 (1992)
12. Pajdla, T., Van Gool, L.: Matching of 3-D curves using semi-differential invariants. In: IEEE ICCV, pp. 390–395 (1995)
13. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: International Conference on 3-D Digital Imaging and Modeling, pp. 145–152 (2001)
14. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. on Pattern Analysis and Machine Intelligence 24, 509–522 (2001)
15. Leordeanu, M., Hebert, M.: A Spectral Technique for Correspondence Problems Using Pairwise Constraints. In: IEEE ICCV, vol. 2, pp. 1482–1489 (2005)
16. Serradell, E., Moreno-Noguer, F., Kybic, J., Fua, P.: Robust elastic 2D/3D geometric graph matching. SPIE Medical Imaging 8314(1), 831408-1–831408-8 (2012)
17. Cour, T., Srinivasan, P., Shi, J.: Balanced graph matching. In: Neural Information Processing Systems, pp. 313–320 (2006)
18. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Research Logistics 2(1-2), 83–97 (1955)

Robust Non-Rigid Registration of 2D and 3D Graphs

Eduard Serradell[1], Przemyslaw Glowacki[2], Jan Kybic[3], Francesc Moreno-Noguer[1], Pascal Fua[2]

[1]Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain
[2]Computer Vision Laboratory (EPFL), Lausanne, Switzerland
[3]Faculty of Electrical Engineering (CTU), Prague, Czech Republic

Abstract

We present a new approach to matching graphs embedded in $\mathbb{R}^2$ or $\mathbb{R}^3$. Unlike earlier methods, our approach does not rely on the similarity of local appearance features, does not require an initial alignment, can handle partial matches, and can cope with non-linear deformations and topological differences.

To handle arbitrary non-linear deformations, we represent them as Gaussian Processes. In the absence of appearance information, we iteratively establish correspondences between graph nodes, update the structure accordingly, and use the current mapping estimate to find the most likely correspondences that will be used in the next iteration. This makes the computation tractable.

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.

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 on the left of Fig. 1 look radically different when imaged using an electron microscope (EM), as shown on the right of Fig. 1. Nevertheless, registering them is desirable 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. Such drastic appearance changes make it impractical to use registration techniques that rely on maximizing image similarity [26, 18], in particular when the images are very different and when dealing with thin structures, such as blood ves-

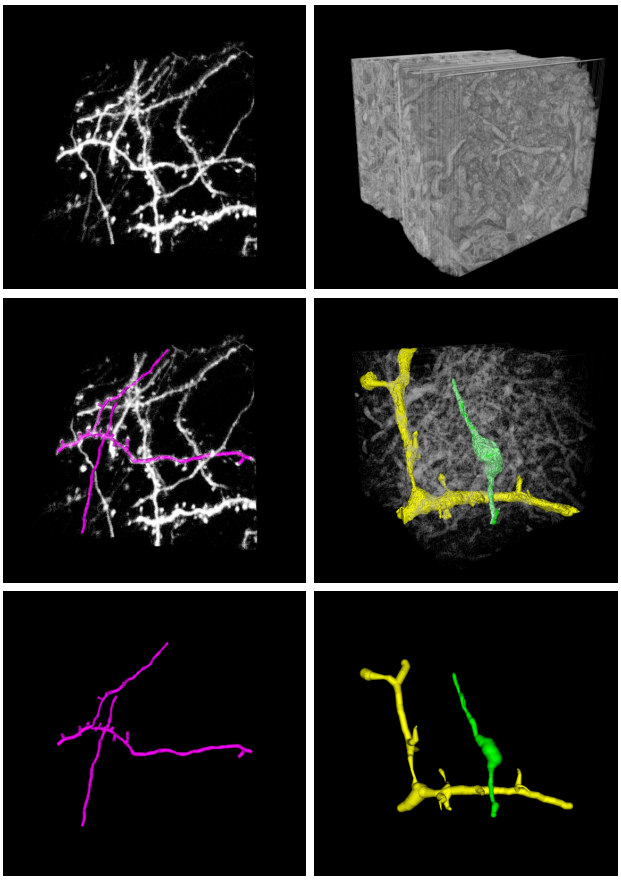


Figure 1. **Brain tissue at different resolutions. Top:** *Left)* Image stack acquired using a light microscope from live brain tissue at the micron resolution. *Right)* A smaller area of the same tissue imaged using an electron microscope after excision and fixation, at a 20 nanometer resolution or 5 times that of the two-photon microscope. **Center:** *Left)* Semi-automated delineation of some dendrites overlaid in magenta. *Right)* Manual segmentation of an axon overlaid in green and a dendrite in yellow. **Down:** The same structures on a black background. Because the resolution is so much higher in the EM data, dendritic spines and synapses are clearly visible. This figure, as most others in this paper, is best viewed in color.

sels 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 geometrical 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 distance between graph junction points [9, 24, 5], which is very sensitive to the small length changes inherent to the biological structures we consider. This may be valid for pulmonary vessels, which just undergo smooth deformations, or retinal fundus images that show slight non-linearities –due to the curved surface of the retina– when viewed from different perspectives. Yet, when dealing with images acquired using distinct modalities and at different resolutions, the structures to match exhibit significant topology changes and large non-linear deformations. Furthermore, we know of no current method that can simultaneously handle 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 handle 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 in the two images or image-stacks 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.

We demonstrate the effectiveness of our technique at registering angiography and retinal-fundus images acquired at different times and different points of view, as well as neural image-stacks acquired using different modalities.

2. Related Work

Area-based registration techniques that maximize image similarity criteria such as correlation or mutual-information [26, 18] 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 can be divided into four main classes. In most cases the branching points (nodes) are extracted and used for matching, while the edges connecting them are often ignored.

In the first class the graphs are assumed to be related by a low-dimensional geometric transform, such as a rigid one, which can be instantiated from very few –only 3 in some cases– correspondences. It is therefore feasible to hypothesize and test random correspondences, as it is done in RANSAC [7] or its variants [4]. However, RANSAC-like approaches quickly become impractical as the number of transformation parameters or graph nodes increases.

The second class of approaches 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 such algorithm is the Iterative Closest Point (ICP) method [2], later extended to non-rigid transformations using techniques such as Non-Rigid ICP [1, 13], or Coherent Point Drift (CPD) [16]. A good initial estimate is critical as the algorithm will typically fall into a bad local minimum without it.

The third class of methods relies on having a sufficiently discriminative criterion for pairwise compatibility between nodes, such as local appearance descriptors or geometric compatibility between correspondence pairs [9, 11, 25, 6, 12, 3]. Global correspondences between nodes are then estimated using multidimensional optimization schemes such as graduated assignment [9], spectral techniques [11, 25, 12] or considering the graphs as an absorbing Markov chain [3]. Considering compatibilities as binary tests, the largest consistent set of matches corresponds to the maximum weighted independent set or equivalently the maximum weighted clique [6]. Due to its high computational cost, the method is only applicable to small graphs. Within medical imaging, some authors have exploited the graph matching formulation to register slightly deformed images or volumes such as pulmonary vessels [24], or warping retinal fundus images [5]. While these methods allow a non-parametric formulation of the problem, they cannot be used when appearance information is unavailable and distances vary due to non-linear deformations, which is the case we consider in this paper.

The final class of methods involves simultaneously searching for correspondences and estimating the transformation parameters using a Kalman filter approach [14, 22, 20, 23, 21]. As soon as a few initial correspondences have been established, the set of potential correspondences is rapidly reduced, making the search complexity manageable. However, these algorithms, like RANSAC, require an *a priori* parametric model whose parameters are computed using the correspondences, and thus, cannot generalize to arbitrary deformations. Similar limitations are also shared by methods relying on implicit shape models [17, 8]. In the Gaussian Process framework we propose, we also progres-

sively reduce the number of potential correspondences but, in contrast to these previous approaches, no 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 enlarges significantly the area of applicability and yields better results.

3. Approach

In this work, we assume that we are provided with two graphs $\mathcal{G}^A = (\mathbf{X}^A, \mathbf{E}^A)$ and $\mathcal{G}^B = (\mathbf{X}^B, \mathbf{E}^B)$ extracted from two images or image-stacks A and B, where the $\mathbf{E}$s represent the graphs' edges and the $\mathbf{X}$s their nodes that can be either 2D or 3D vertices. Our goal is to use them to find a mapping m from A to B such that $m(\mathbf{x}^A)$ is as close as possible to $\mathbf{x}^B$ in the least-squares sense assuming that $\mathbf{x}^A$ and $\mathbf{x}^B$ are corresponding pixels or voxels. If correspondences between points belonging to the two graphs were given, we could directly use the Gaussian Process (GP) formalism to compute the mapping as a non-linear regressor [19] that yields a mean prediction m and its associated variance. 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 only rely on geometrical properties to simultaneously establish the correspondences and estimate the underlying non-linear transform. To this end, our algorithm goes through the following steps:

1. **Coarse alignment:** We begin by matching graph nodes so that distances along the edges connecting corresponding nodes, which we will refer to as *geodesic* distances, are changed as little as possible.

   We initialize the process by randomly picking two correspondences, which roughly fixes relative scale and orientation, and use them to instantiate a GP. We then refine it using the recursive procedure described below. If this procedure fails, we pick another random pair until it succeeds.

   Given some correspondences 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, especially given the fact that geodesic distances must be preserved. Among these possibilities, we select the most likely one, use it to refine the GP, and iterate. 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 a rough mapping.

2. **Fine alignment:** The mapping discussed above has been learned only from the graph nodes (branching points), and is therefore coarse. To refine it, we also establish correspondences between points lying on edges connecting graph nodes. Because there are many more of those than they were nodes, this would be combinatorially explosive if we did it from scratch as done previously. Instead, we constrain the correspondences to be between samples belonging to edges of already connected corresponding nodes and rely on a Hungarian algorithm [15] to perform the optimal assignment quickly.

In the remainder of this section, we first outline the GP formalism that we use. We then discuss our procedures for coarse and fine alignments.

3.1. Non-Linear Regression

Consider the case where we are given N correspondences $(\mathbf{x}_i^A, \mathbf{x}_i^B)_{1\leq i\leq N}$ between 2D or 3D points from A and B respectively. Using the GP approach to non-linear regression and assuming Gaussian i.i.d. noise of precision β^{-1} in the coordinate values, these correspondences can be used to predict that the point $\mathbf{x}^B$ in B corresponding to $\mathbf{x}^A$ in A can be expected to be found at a location whose mean m_N and variance σ_N can be computed as

$$m_N(\mathbf{x}^A) = \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{X}_N^B \ , \quad (1)$$

$$\sigma_N^2(\mathbf{x}^A) = k(\mathbf{x}^A, \mathbf{x}^A) + \beta^{-1} - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k} \ , \quad (2)$$

where k is a kernel function, β is proportional to the expected noise-level in the data, $\mathbf{C}_N$ is the $N \times N$ 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 $[k(\mathbf{x}_1^A, \mathbf{x}^A), \ldots, k(\mathbf{x}_N^A, \mathbf{x}^A)]^T$, and $\mathbf{X}_N^B$ is the vector $[\mathbf{x}_1^B, \ldots, \mathbf{x}_N^B]^T$.

Among the different types of kernel functions [19] we chose the widely used summation of a squared-exponential, a constant term, and a linear one

$$k(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\theta}_0 + \boldsymbol{\theta}_1 \mathbf{x}_i^T \mathbf{x}_j + \boldsymbol{\theta}_2 \exp\left\{-\frac{\boldsymbol{\theta}_3}{2}||\mathbf{x}_i - \mathbf{x}_j||^2\right\}. \quad (3)$$

We found this kernel to be the most appropriate for our purposes, because it implicitly defines a mapping function composed of an affine plus a non-linear transformation. This accounts for most of the warps appearing in biomedical imaging.

Given this expression for k, the mean prediction of Eq. 1 can now be rewritten as

$$\begin{aligned} m_N(\mathbf{x}^A) &= \sum_{i=1}^{N} a_i k(\mathbf{x}_i^A, \mathbf{x}^A) \\ &= \sum_{i=1}^{N} a_i (\boldsymbol{\theta}_0 + \boldsymbol{\theta}_1 (\mathbf{x}_i^A)^T \mathbf{x}^A) + \\ &\quad \sum_{i=1}^{N} a_i \boldsymbol{\theta}_2 exp\left\{-\frac{\boldsymbol{\theta}_3}{2}||\mathbf{x}_i^A - \mathbf{x}^A||^2\right\} \ , \quad (4) \end{aligned}$$

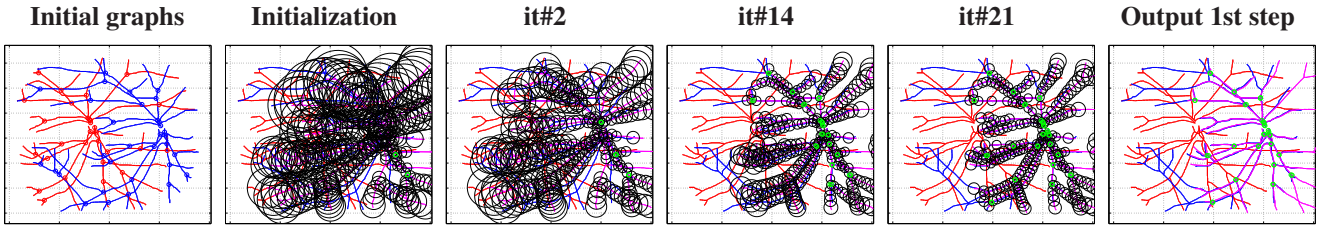


Figure 2. **Coarse alignment steps.** The initial graph structures are depicted in the left-most figure, the model graph in red and the target in blue. Exploration of the search space starts by picking randomly two correspondences, highlighted in green, thus roughly fixing scale and orientation. Then, 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, overlaid in magenta. Best viewed in color.

where a_i is the i^{th} element of the vector $\mathbf{C}_N^{-1}\mathbf{X}_N^B$. The first term of Eq. 4, containing the $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ hyperparameters, is a linear function of the input variables while the second one, involving the $\boldsymbol{\theta}_2$ and $\boldsymbol{\theta}_3$ hyperparameters and an exponential, allows for additional non-linear deformations.

3.2. Coarse Alignment

Let $\mathbf{X}^A = \{\mathbf{x}_1^A, \ldots, \mathbf{x}_{n_A}^A\}$ and $\mathbf{X}^B = \{\mathbf{x}_1^B, \ldots, \mathbf{x}_{n_B}^B\}$ be the nodes of our two graphs. As discussed at the beginning of this section, our first goal is to simultaneously retrieve as many correspondences $\{\mathbf{x}^A \leftrightarrow \mathbf{x}^B\}$ as possible and the underlying non-linear mapping $\mathbf{x}^B = m(\mathbf{x}^A)$ that best aligns them.

We take m to be a GP written using the formalism of Section 3.1, which we instantiate by first randomly selecting only two matches. This gives us an initial correspondence set H_0, enough to roughly fix the global scale and rotation, and then we recursively add new correspondences as follows.

1. Given k correspondences, we compute the mapping $m_k(.)$ and covariance estimator $\sigma_k^2(.)$ of 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 defined by the predicted covariance $\sigma_k^2(\mathbf{x}_i^A)$.
3. We choose the node $\mathbf{x}_i^A$ with the smallest number of potential correspondences, 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 H_0.

This iterative process is repeated until a large enough H_0 is found, as is done in RANSAC. If that does not happen, the algorithm backtracks and selects different correspondences using a depth-first search strategy. This is depicted in Fig. 2.

The process is controlled by the vector $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3\}$ containing the kernel hyperparameters of Eq. 3 and the noise parameter β of Eq. 2. 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 distances to the origin is one and perform the computation on the scaled versions. As a result, we were able to use the same $\boldsymbol{\Theta}$ and β for all experiments described in Section 4.

To speed up the computation, we reject correspondences that would produce overly large changes in geodesic distances, which we define as the length γ_{ij} of a path connecting the edges between two graph nodes $\mathbf{x}_i$ and $\mathbf{x}_j$. Given N already established correspondences between graphs, for each new potential match, the geodesic distances connecting the new corresponding points to the nodes in both graphs have to be proportional. We set the tolerance to geodesic distance variations depending on the level of deformations we expect to recover. Proceeding in this way, the algorithm gains robustness against outliers, while it avoids 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.

3.3. Fine Alignment

Having performed the rough alignment as described above, we now have an initial set of N correspondences H_0 between graph nodes, as well as the corresponding GP mapping $m_N(.)$ and covariance estimator $\sigma_N^2(.)$. Since the graph nodes are connected by paths, we can refine the mapping by also establishing correspondences between points that lie on these paths.Since allowing any point to potentially be matched to any other would be prohibitively expensive, we assume the correspondences between nodes to be correct and only establish new ones between points lying on paths linking matching vertices. We again do this iteratively using the following two steps:

1. For each pair of paths connecting pairs of corresponding vertices, we use the Hungarian algorithm [15] to establish matches between the points forming the path. To this end, we use the Mahalanobis distance between potential assignments computed using the current mapping $m(.)$ and covariance estimator $\sigma^2(.)$. All the matches need to have consistent geodesic distances to the nodes in their respective graphs.
2. Given these new correspondences, we reestimate $m(.)$ and $\sigma^2(.)$ and iterate.

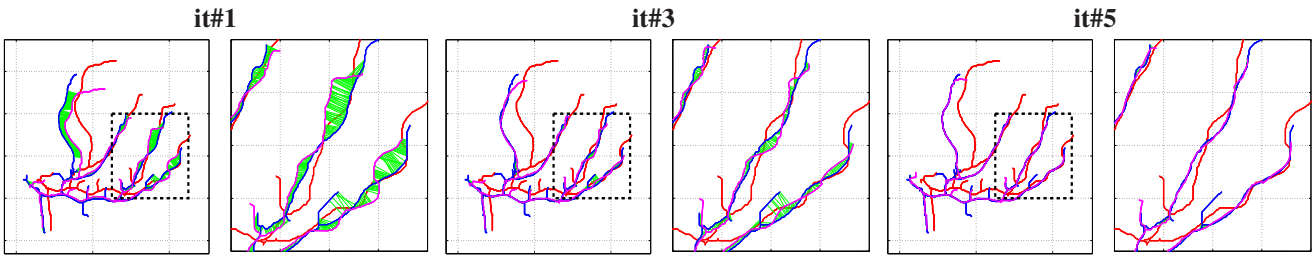


Figure 3. **Fine alignment steps.** Once a coarse alignment of the two graphs (model in red and target in blue) has been found, the algorithm starts to match points lying on the edges. The assignments (depicted in green) are computed using the Hungarian algorithm and constrained by the graph topology and GP predictions. After a few iterations, the warped structure (magenta) is completely aligned to the target graph. For each successive plot, we zoom to a smaller region to better show the algorithm at work. Best viewed in color.

Eventually, the distance error between both graphs can not be further minimized and the process ends. This yields a final expanded set of correspondences H_R and corresponding $m_R(.)$ and covariance estimator $\sigma^2_R(.)$. Note that we use the same GP parameters $\mathbf{\Theta}$ and β as before. The whole process is illustrated by Fig. 3.

4. Experiments

We now present the results on both synthetic and real data, by first extensively evaluating our algorithm in controlled experiments with known ground truth and then showing the results in registering real 2D and 3D biomedical images. For the synthetic experiments we compare our approach (denoted *Non-Linear GP*) to the Coherent Point Drift (*CPD*) [16], which is a representative example of the state-of-the-art in non-rigid point matching and shape recovery. We also compare it against [21], which uses a Kalman filter based approach to learn an initial affine transform and refines the output with a local non-linear warping. We refer this approach as *Affine Kalman*.

4.1. Synthetic Data

We applied all algorithms to synthetic 3D structures with increasing levels of deformation and amount of outliers. We used Vascusynth[10] to synthesize several 3D trees of 25 nodes each within a volume of $100\times100\times100$ voxels, such as the one shown in Fig. 4. We simulated the difficulties encountered with real data, by adding increasing levels of noise σ_n to the tree node locations, bending the branches and introducing spurious nodes and corresponding branches, which we will refer to as outliers. The magnitude of the bending was controlled through a parameter p_b, which establishes the length ratio between the original edge and the deformed one. The percentage of outliers p_o refers to the number of introduced non-corresponding nodes with respect to the original number in the rigid tree. We then registered the original reference shape and the synthetic deformed graphs using each of the algorithms.

Two different types of experiments were performed. We first evaluated the amount of deformation each algorithm was able to recover by sweeping the variances of the accumulative joint noise within the range $\sigma_n \in [0-1]$ voxels, and fixing $p_b = 25\%$ and $p_o = 30\%$. To give significance to these levels of deformation and reconstruction errors, Fig. 4-right depicts different deformations of the reference model corresponding to specific values of σ_n and p_b. In a second experiment we computed the robustness to outliers by synthetically introducing random nodes –which turned to outlier tree branches– within the range $p_o \in [0-100]\%$, and setting $\sigma_n = 0.3$ voxels and $p_b = 25\%$.

For each set of experimental parameters, we performed 10 trials and compared our approach to the Affine Kalman, and to the CPD both in its affine and non-linear versions. For a fair comparison, since the CPD allows for control of the amount of non-linearity by tuning some internal parameters, we tried several configurations and retained the solution yielding the best results. The graphs on Fig. 4 depict the mean 3D reconstruction error, expressed in voxels, both for increasing deformation levels, and increasing number of outliers. Observe that our algorithm consistently outperforms CPD in all experiments. This demonstrates the advantage of using the geodesic distance compatibility between points, which is inherently used in our approach.

We also clearly outperform [21], mainly for large levels of deformation. While our non-linear algorithm is able to warp the graph while searching for matching nodes, the affine search of [21] only yields reasonable results for low levels of deformation. In addition, observe that the magnitudes of error obtained by our approach are in fact very good approximations.

Finally, we compared against [12], which is representative of the graph matching algorithms. Note that these kind of approaches only tackle the problem of assigning the correspondences, but are not specifically designed for recovering the underlying transformation. We therefore focused only on the retrieved matches. We observed that [12] is only effective when all inlier correspondences show a similar pattern, but has more difficulties under non-linear deformations, as shown in Fig. 5.

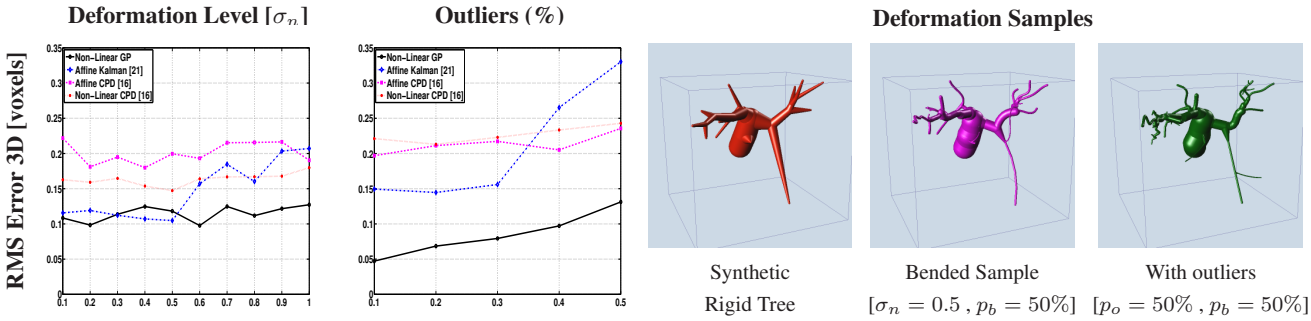


Figure 4. **Synthetic experiments. Left:** RMS error for 3D reconstruction for each experiment. The plot compares the median of the result for each algorithm. **Right:** Synthetic samples generated using different configurations of the control parameters.

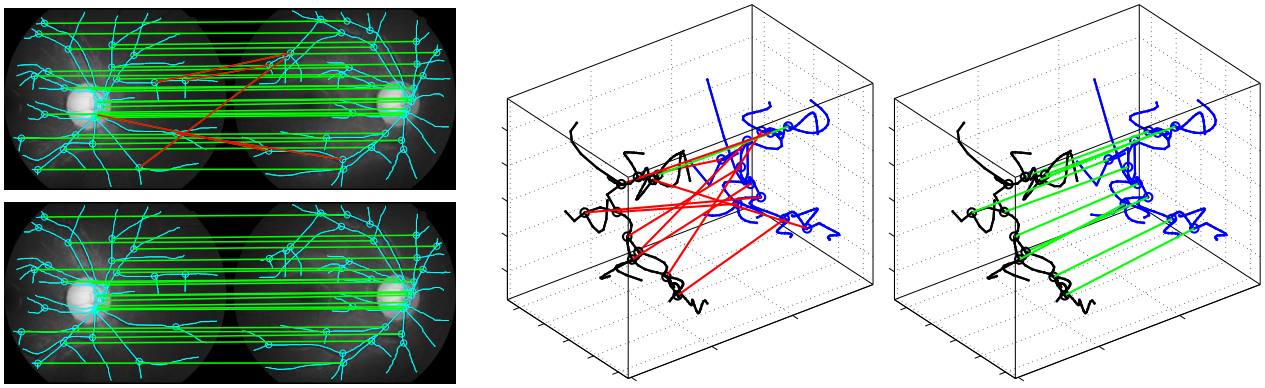


Figure 5. **Search for correspondences.** Graph matching methods are a common technique to find correspondences. **Left:** In the retinal fundus images were the deformation is quasi-affine, both the graph matching algorithm [12] (**top**) and our method (**down**) were able to recover the correct matching. **Right:** However, when the structures to register present a too large deformation, the graph matching method (**left**) missed most of the correct assignments, while our method is still successful (**right**). Best viewed in color: the correct correspondences are painted in green and the outlier ones in red.

4.2. Real Data

We next present some real examples of the results obtained by our algorithm on several biomedical datasets. The graphs were extracted semi-automatically using the Fiji [1] platform and its plugins.

In Fig. 6 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 points of view. This results in apparent distortions of the curved retinal surface's projection, which are well modeled by an affine transform. Because there is very little non-linearity in the deformation, these results are similar to those of [5], 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 Non-Linear GP can naturally handle such artifacts.

In the 2D X-ray angiography images of Fig. 7 the deformations are much more non-linear. As shown on the zoomed 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. And even when we supplied our affine estimate, CPD only succeeded when dealing with small non-linearities, but not otherwise, as shown in Fig. 7.

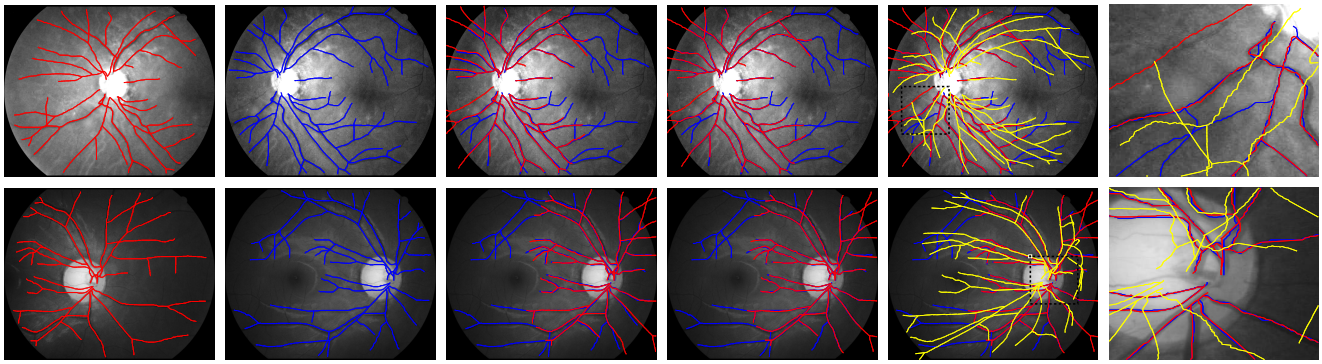
Image (Fig.#)	Image Size	Error NLGP	Error CPD
Angio. (7 a)	512x512 pix	1.1969 pix	3.1633 pix
Angio. (7 b)	512x512 pix	1.8080 pix	3.4183 pix
Ret. (6 a)	1548x1260 pix	2.6757 pix	20.6653 pix
Ret. (6 b)	1548x1260 pix	2.5109 pix	20.4496 pix
Neuronal (8)	4.4x5.7x6.0 μm	0.0702 μm	0.2628 μm

Table 1. Comparison of our Non-Linear GP (NLGP) and the Coherent Point Drift (CPD) for real data.

Finally, we register the 3D neuronal stacks extracted from the brain tissue of Fig. 1 using two different modalities. Even though the two images look extremely different, our algorithm returns a valid deformation as shown in Fig. 8. Neither CPD nor Affine Kalman were able to recover the correct alignment.

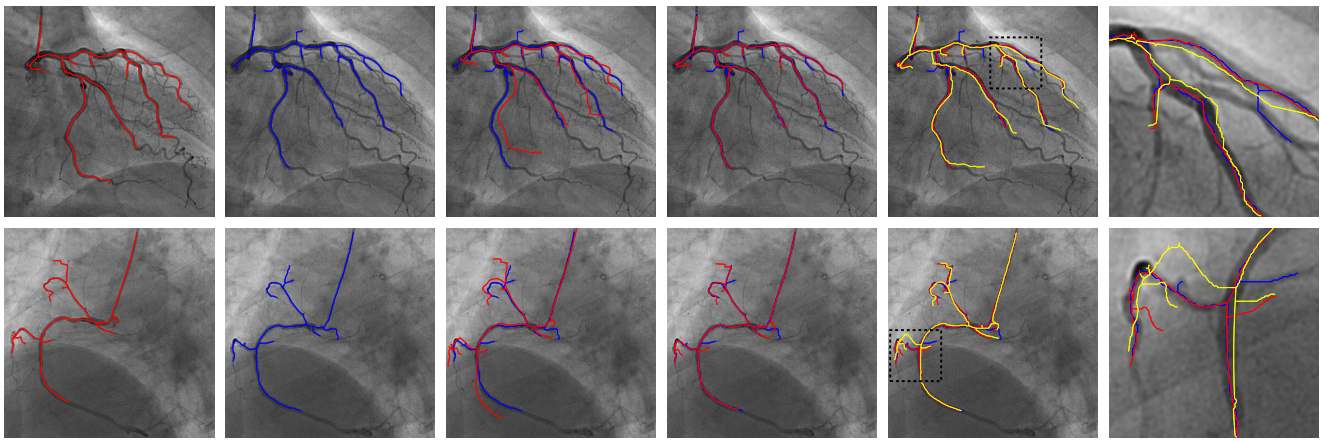
In order to quantitatively compare the CPD accuracy against ours in the absence of ground truth, we computed an optimal assignment for each node in the deformed graphs

[1] http://pacific.mpi-cbg.de



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

Figure 6. **Retinal fundus images used in [5]. (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 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,f)** Our result is superposed with the Coherent Point Drift alignment. In this dataset, our algorithm behave well, but CPD fails to recover the correct shape because of there are too many non-correponding branches. Best viewed in color.



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

Figure 7. **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. **(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 robustnees against vessel bendings and outliers, achieving a better registration of the two shapes. Best viewed in color.

– overlaid in red for our method and yellow for CPD on Figs. 6 and 7 (d,e,f) – to its nearest neighbor in the corresponding graph overlaid in blue. We report the results of this approximate error in table 1.

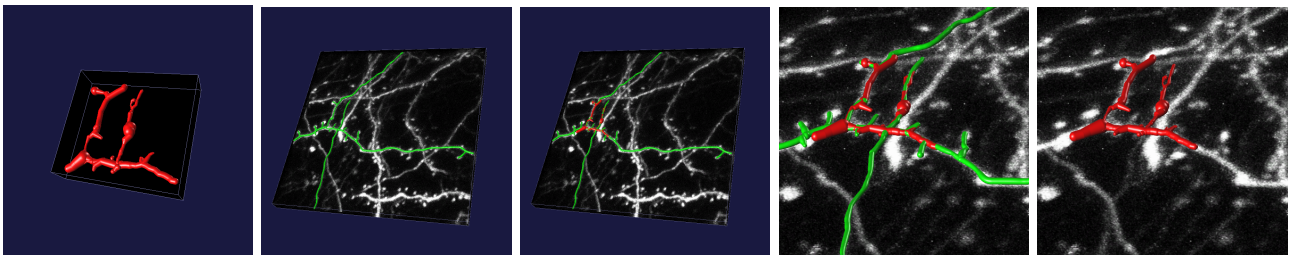
5. Conclusion

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 mapping from one graph to another and using this mapping to progressively constrain the search area for correspondences between graph nodes.

We have demonstrated our algorithm on graphs containing up to 100 nodes, for which the computation takes approximately 500 seconds in MATLAB on a 8 Core 1.6GHz 64-bit Processor. A C implementation would produce a significant speedup. A further one will result from refining the strategy we currently use to explore the set of possible correspondences, which is what we are currently working on.

Acknowledgments

This work has been partially funded by the ERC projects MicroNano and GARNICS FP7-247947, the Spanish Ministry of Economy and Competitiveness under projects PAU+ DPI2011-27510 and MIPRCV Consolider Ingenio 2010 CSD2007-00018, and the Czech Science Foundation under project P202/11/0111. We thank the project ‘La Marató’ de TV3 082131 for providing the heart dataset.



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

Figure 8. **Neuronal image stacks: multimodal registration.** **(a)** Graph structure extracted from the EM image stack, in red. **(b)** The segmented graph structure in green, overlaid over the LM image stack. **(c)** After the non-linear registration process, the EM segmented neuron is deformed and aligned over the LM extracted neuron. **(d)** A zoom over the region where the EM stack has been extracted. The two neurons have been completely aligned. **(e)** The EM neuron, embedded in the LM image stack. Best viewed in color.

References

[1] B. Amberg, S. Romdhani, and T. Vetter. Optimal Step Nonrigid ICP Algorithms for Surface Registration. In *CVPR*, 2007.

[2] 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.

[3] M. Cho, J. Lee, and K. Lee. Reweighted Random Walks for Graph Matching. In *ECCV*, pages 492–505, 2010.

[4] O. Chum and J. Matas. Matching with PROSAC - Progressive Sample Consensus. In *CVPR*, pages 220–226, June 2005.

[5] 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.

[6] O. Enqvist, K. Josephson, and F. Kahl. Optimal Correspondences from Pairwise Constraints. In *ICCV*, 2009.

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

[8] K. Fujiwara, K. Nishino, J. Takamatsu, B. Zheng, and K. Ikeuchi. Locally Rigid Globally Non-Rigid Surface Registration. In *ICCV*, 2011.

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

[10] P. Jassi and G. Hamarneh. VascuSynth: Vascular Tree Synthesis Software, 2011. *http://hdl.handle.net/10380/3260*.

[11] M. Leordeanu and M. Hebert. A Spectral Technique for Correspondence Problems Using Pairwise Constraints. In *ICCV*, pages 1482–1489, October 2005.

[12] 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.

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

[14] F. Moreno-Noguer, V. Lepetit, and P. Fua. Pose Priors for Simultaneously Solving Alignment and Correspondence. In *ECCV*, October 2008.

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

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

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

[18] 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.

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

[20] J. Sanchez-Riera, J. Ostlund, P. Fua, and F. Moreno-Noguer. Simultaneous Pose, Correspondence and Non-Rigid Shape. In *CVPR*, June 2010.

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

[22] E. Serradell, M. Özuysal, V. Lepetit, P. Fua, and F. Moreno-Noguer. Combining Geometric and Appearance Priors for Robust Homography Estimation. In *ECCV*, pages 58–72, September 2010.

[23] 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 *ICCV*, November 2011.

[24] 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.

[25] L. Torresani, V. Kolmogorov, and C. Rother. Feature Correspondence via Graph Matching: Models and Global Optimization. In *ECCV*, pages 596–609, 2008.

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