# 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

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 $\mu m$) 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

**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 |
| $\gamma$ | Minimum number of required assignments for $\psi$ |
| $S_A$ | Sensor response for set $A$ |
| $r(S_A)$ | Sensor likelihood ratio |
| $K$ | Total number of iterations |
| $\pi_k$ | Assignment to evaluate at iteration $k$ |
| $\boldsymbol{\pi} = (\pi_1, ..., \pi_K)$ | Sequence of observations to make |
| $\mathcal{C}_A$ | Set of children of $A$ |

where each element $x_i^A$ of $X^A$ maps to at most one element of $X^B$, which is 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 $\gamma$ is the minimum number of matches required to calculate $\psi$. 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 $S_{\pi_k}$. To do this, we consider solving the MAP,

$$\begin{aligned} Y^* = \underset{y \in \mathcal{Y}}{\arg\max}\, P(Y | S_{\pi_1}, \ldots, S_{\pi_K}) &= \underset{y \in \mathcal{Y}}{\arg\max} \left\{ \frac{1}{Z} P(Y) P(S_{\pi_1}, \ldots, S_{\pi_K} | Y) \right\} \\ &= \underset{y \in \mathcal{Y}}{\arg\max} \left\{ \frac{1}{Z} P(Y) \prod_{i=1}^{K} P(S_{\pi_k} | Y) \right\}, \end{aligned} \tag{1}$$

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, $\pi_k$ 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 $\pi_1$ using our chosen sensor $\psi$, compute the posterior distribution of $Y$ given the new information, $S_{\pi_1}$, and select the most promising new set $\pi_2$ 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 $\sigma_n^2$. 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 $\sigma_n^2$,

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

### 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 $\delta$, 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 $\pi_k$, 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 $\pi_k$ has been observed by

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

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 $\epsilon$ 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: $\quad \{\pi_k, \epsilon_k\} = \text{pop}(Q)$; //choose the most likely $\pi_k$
4: $\quad S_{\pi_k} = \psi(\pi_k)$
5: $\quad$ **for** $y \in \mathcal{C}_{\pi_k}$ **do**
6: $\quad\quad Q \leftarrow Push(\pi_k \cup \{Y_{|\pi_k|+1} = y\},\, \epsilon_k r(S_{\pi_k})/|\mathcal{C}_{\pi_k}|)$
7: $\quad$ **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 $\epsilon_k$. 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 $\pi_k$ 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 $\pi^*$ 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 $\psi$, 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

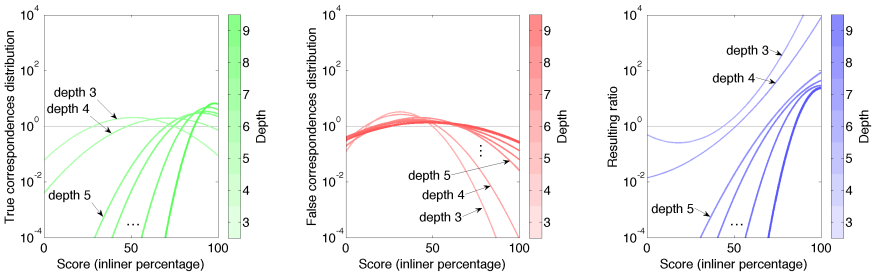



**Fig. 1.** Example for values of distributions taken from **(a)** true correspondences samples, **(b)** false correspondences and **(c)** ratio between the true and false distributions. The sensor used to compute this example was the number of inliers – described in Eq. 5

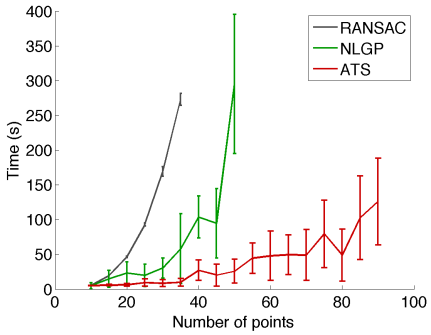



**Fig. 2.** Processing time required by RANSAC and NLGP in comparison to our method as a function of the number of points

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 $\psi$, 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 will mostly encounter correspondences composed of mostly true correspondences, except for the last one. Therefore, we sample many random false correspondences

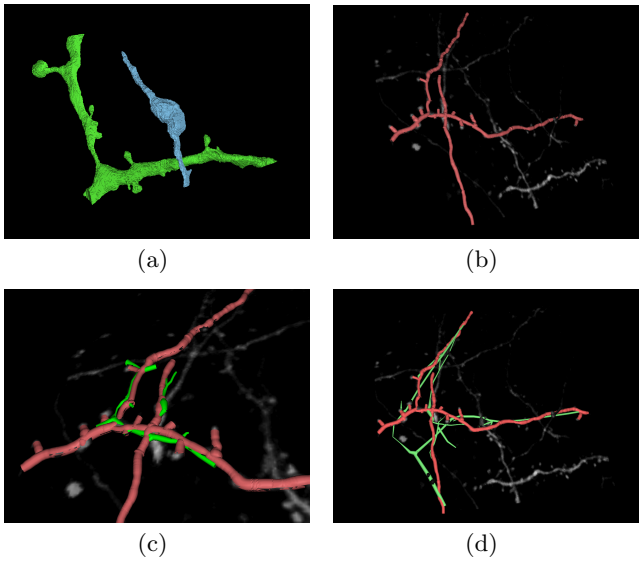



**Fig. 3.** Light and electron microscopy neuronal trees. **(a):** Segmented electron microscopy data. **(b):** Segmented light microscope data. **(c):** Registration of structures using ATS. **(d):** Registration using CPD

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).

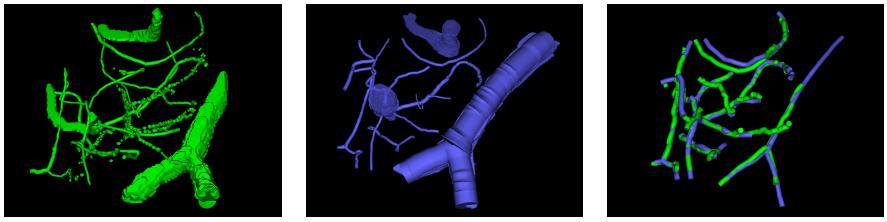
In Fig. 1(c), we can see the likelihood ratio between the distributions for true and false correspondences. This shows that the sensor gets more discriminative as the size of the partial correspondence being tested increases.

## 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

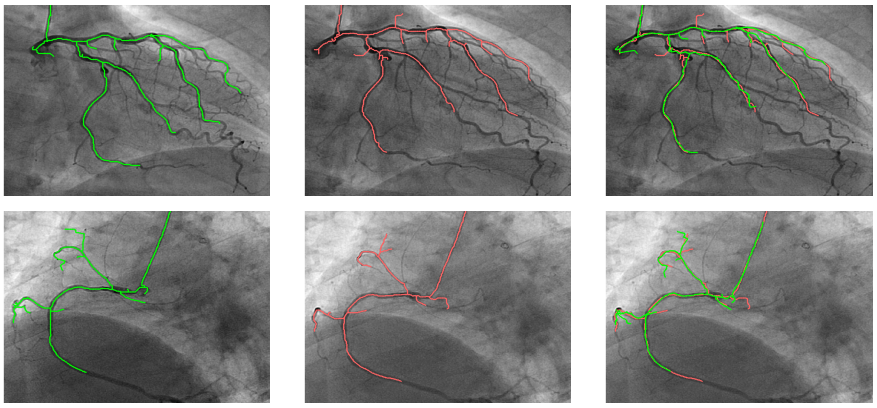


**Fig. 5.** Matching and registration of heart angiograms. **Left and center:** Original data. **Right:** Registration of structures using Active Testing Search.

### 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[4] 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 in Fig. 3, the alignment is not correct and the resulting deformation is not

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

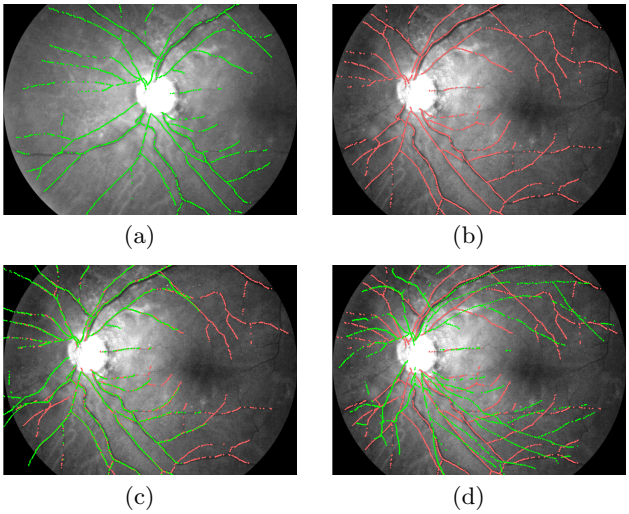



**Fig. 6.** Retinal fundus tree images. **(a) and (b):** Original data. **(c):** Registration of structures using Active Testing Search, **(d):** Registration using CPD

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).

| 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 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)

| 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 |

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

## 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 pre-aligned. 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. IEEE CVPR (2012) 996–1003
2. Geman, D., Jedynak, B.: An active testing model for tracking roads in satellite images. IEEE Trans. on Pattern Analysis and Machine Intelligence **18** (1995) 1–14
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. (1997) 197–212

4. Sznitman, R., Jedynak, B.: Active testing for face detection and localization. IEEE Trans. on Pattern Analysis and Machine Intelligence **32**(10) (2010) 1914–1920
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** (2012) 1
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** (1997) 957–964
7. Myronenko, A., Song, X.: Point set registration: Coherent point drift. IEEE Trans. on Pattern Analysis and Machine Intelligence **32**(12) (2010) 2262–2275
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** (1981) 381–395
9. Torr, P.H.S., Zisserman, A.: MLESAC: A new robust estimator with application to estimating image geometry. Computer Vision and Image Understanding **78** (2000) 138–156
10. Chum, O., Matas, J.: Matching with PROSAC – progressive sample consensus. IEEE CVPR (2005) 220–226
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) (1992) 239–256
12. Pajdla, T., Gool, L.V.: Matching of 3-D curves using semi-differential invariants. In: IEEE ICCV. (1995) 390–395
13. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. International Conference on 3-D Digital Imaging and Modeling (2001) 145–152
14. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. on Pattern Analysis and Machine Intelligence **24** (2001) 509–522
15. Leordeanu, M., Hebert, M.: A Spectral Technique for Correspondence Problems Using Pairwise Constraints. IEEE ICCV **2** (2005) 1482–1489
16. Serradell, E., Moreno-Noguer, F., Kybic, J., Fua, P.: Robust elastic 2D/3D geometric graph matching. SPIE Medical Imaging **8314**(1) (2012) 831408–831408–8
17. Cour, T., Srinivasan, P., Shi, J.: Balanced graph matching. Neural Information Processing Systems (2006) 313–320
18. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Research Logistics **2**(1-2) (1955) 83–97