

Fast registration of segmented images by normal sampling

Jan Kybic

Faculty of Electrical Engineering
Czech Technical University in Prague

kybic@fel.cvut.cz

Martin Dolejší

Jiří Borovec

Abstract

It is known that image registration is mostly driven by image edges. We have taken this idea to the extreme. In segmented images, we ignore the interior of the components and focus on their boundaries only. Furthermore, by assuming spatial compactness of the components, the similarity criterion can be approximated by sampling only a small number of points on the normals passing through a sparse set of keypoints. This leads to an order-of-magnitude speed advantage in comparison with classical registration algorithms. Surprisingly, despite the crude approximation, the accuracy is comparable. By virtue of the segmentation and by using a suitable similarity criterion such as mutual information on labels, the method can handle large appearance differences and large variability in the segmentations. The segmentation does not need not be perfectly coherent between images and over-segmentation is acceptable.

We demonstrate the performance of the method on a range of different datasets, including histological slices and Drosophila imaginal discs, using rigid transformations.

1. Introduction

Image registration [31] is one of the basic tasks in biomedical image processing and is part of most biomedical image processing pipelines. In this work we shall address the task of registering the images fast. This remains a challenge as processor speeds are stagnating while the image resolution keeps increasing. Fast registration is important for example for interactive and large throughput applications. Currently used methods typically take tens of seconds for registering medium size images up to several hours for larger 3D images. This is prohibitive for many applications. Hence the motivation to find faster methods.

In computer vision, registration is mostly approached by detecting feature points in both images, matching them, and fitting a motion model [20]. This has a potential of being fast as long as the expected motion model is simple but can only rarely be applied to medical imaging problems where

reliable feature points are hard to find.

Instead, our approach for image registration acceleration is based on three observations: First, it is known that image registration is indeed driven mainly by the edges and the similarity criterion can be well approximated from points of high gradient [26]. Second, when matching images from different physical subjects, even if two regions are supposed to match, because they correspond *e.g.* to the same organ of two different subjects, the internal structure of this organ might be so variable that there are no geometrically corresponding structures in the region interiors of the two subjects. Another example is matching two neighboring histological slices (Fig. 1), each containing physically different cells, so there is no correspondence between the small scale details. The third observation comes from the so-called ‘aperture problem’ which implies that locally, we can only distinguish motion in the direction normal to the boundary between homogeneous regions.

We therefore propose to simplify the input images as much as possible by segmenting them into a set of regions which may correspond *e.g.* to different tissue types (Fig. 1). Furthermore, we ignore the interior of the regions and concentrate on their boundaries. Finally, we sample the segmentation labels in the moving image only on short normal 1D line segments (Fig. 2). As a result, the amount of data the algorithm needs to consider in each iteration is reduced from $10^6 \sim 10^8$ pixels for the classical formulation using all pixels to about $10^3 \sim 10^4$ pixels in the proposed method, with the corresponding potential for acceleration.

As an additional benefit of working with segmentations, we get ‘for free’ the possibility of registering images with very different appearance, *e.g.* coming from different modalities [17].

1.1. Other related work

We are not the first to address the task of registration of segmented images. However, most authors have worked on the special case of alignment of binary images, by moment matching [10] or descriptor matching [13], which is fast but possibly inaccurate. On the other hand, there are also regis-

tration strategies based on level-sets [11] and distance function matching [16], which are accurate but slow.

It is possible to register multiple-class segmented images using the demons algorithm [28], which, similarly to us, samples points on the class boundaries. The idea of normal sampling comes from active contour registration [9]. However, in both cases it is attempted to estimate the necessary movement directly, without explicitly approximating and minimizing the criterion function.

There is a number of methods combining segmentation and registration but also mostly limited to the two-class problem. They propagate a reference segmentation [7], segment objects by thresholding [3, 21], by level sets [2, 30, 29], or GraphCut [22], or alternate the segmentation and registration [17, 24]. In many cases, the fast registration method we describe here could be easily plugged into some of those existing segmentation-registration framework, in order to increase their speed.

Finally, fast registration methods have been successfully developed casting the problem into the discrete optimization setting [12, 14] and using GPUs [27]. Both sources of acceleration could be leveraged in our formulation, too.

2. Method

2.1. Problem definition

We assume we are given two segmented images, $f : \mathbb{R}^d \rightarrow \mathcal{L}^f$ and $g : \mathbb{R}^d \rightarrow \mathcal{L}^g$, and a bounded region of interest $\Omega \subset \mathbb{R}^d$, where the dimension d is usually $2 \sim 3$, and $\mathcal{L}^f, \mathcal{L}^g$ are small sets of class labels (see Fig. 1 for an example). Any of the multitude of existing image segmentation algorithms can be used; we provide some suggestions in Section 2.8.

Let us choose a similarity criterion $J : (\mathbb{R}^d \rightarrow \mathcal{L}^f) \times (\mathbb{R}^d \rightarrow \mathcal{L}^g) \rightarrow \mathbb{R}$ which can be expressed as a sum of pixel contributions

$$J(f, g) = \int_{\mathbf{x} \in \Omega} \varrho(f(\mathbf{x}), g(\mathbf{x})) \, d\mathbf{x} \quad (1)$$

where \mathbf{x} is the pixel coordinate (see Section 2.7 for the discussion of the pixel similarity term ϱ).

The final ingredient is a family of geometrical transformations \mathcal{T} (see Section 2.9), such that $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ for $T \in \mathcal{T}$ is a smooth transformation, depending continuously on a parameter vector $\theta \in \mathbb{R}^{\dim \theta}$. We shall write T_θ when we need to emphasize this dependence.

A segmented image g can be warped with a transformation T , yielding

$$g'(\mathbf{x}) = (g \circ T)(\mathbf{x}) = g(\mathbf{y}) \quad (2)$$

$$\text{with } \mathbf{y} = T(\mathbf{x}) \quad (3)$$

The task of registering the segmented images f and g is then straightforwardly defined as finding the best transformation $T \in \mathcal{T}$, such that the warped image $g' = g \circ T$ of the moving image g is as close as possible to the reference image f in the sense of the similarity criterion. More formally,

$$T^* = \arg \min_{T \in \mathcal{T}} J(T) \quad (4)$$

$$\text{with } J(T) = \int_{\mathbf{x} \in \Omega} \varrho(f(\mathbf{x}), g(T(\mathbf{x}))) \, d\mathbf{x} \quad (5)$$

2.2. Criterion approximation

Numerical evaluation of the criterion (5) can be costly as it needs to visit all pixels within the Ω region, of which there can be easily $10^7 \sim 10^9$ or more in today's high-resolution images.

To reduce the complexity, we will make a number of simplifying assumptions, namely that (i) the optimal transformation T^* is close to a known transformation T_0 ; (ii) the transformation is locally close to affine; and (iii) the segmented regions are sufficiently large. The last assumption is only needed to obtain a good speed-up. The assumption (i) means that we only need to evaluate J for T close to T_0 , while the 'close' transformations can be defined by limiting the maximum displacement to δ

$$\|T(\mathbf{x}) - T_0(\mathbf{x})\| < \delta \quad \text{for all } \mathbf{x} \in \Omega \quad (6)$$

Observe that in (5), it is enough to sum over \mathbf{x} , which are close to class boundaries, as elsewhere the classes do not change and the contribution to the criterion J is therefore constant and does not change the position of the minimum. More formally, let us define the ε -neighborhood $\partial_\varepsilon f$ of class boundaries ∂f in image f

$$\partial_\varepsilon f = \{ \mathbf{x} \in \mathbb{R}^d; \exists \mathbf{y} \in \mathbb{R}^d, \quad (7)$$

$$\| \mathbf{x} - \mathbf{y} \| \leq \varepsilon \wedge f(\mathbf{x}) \neq f(\mathbf{y}) \}$$

$$\partial f = \{ \mathbf{x} \in \mathbb{R}^d; \forall \varepsilon > 0, \exists \mathbf{y} \in \mathbb{R}^d, \quad (8)$$

$$\| \mathbf{x} - \mathbf{y} \| \leq \varepsilon \wedge f(\mathbf{x}) \neq f(\mathbf{y}) \}$$

We can then modify the criterion (5) by integrating only within a distance δ of the class boundaries in g by replacing Ω with

$$\Omega_1 = \Omega \cap (T_0^{-1} \circ \partial_\delta g) = \{ \mathbf{x} \in \Omega; T_0(\mathbf{x}) \in \partial_\delta g \} \quad (9)$$

It can be shown from (8) and (6) that $g(T(\mathbf{x})) = g(T_0(\mathbf{x}))$ for all \mathbf{x} far from the boundaries, $\mathbf{x} \notin \partial_\delta g$. Therefore, for transformations T close to T_0 in the sense of (6), there will be a constant difference between the new criterion J_1 based on (9) and J in (5) and the minimum location is thus unchanged.

Evaluating (9) is often not practical, as it requires inverting T_0 . Hence, we propose to replace J_1 by J_2 obtained by replacing Ω_1 (9) with a region based on the boundaries in f

$$\Omega_2 = \Omega \cap \partial_\gamma f = \{\mathbf{x} \in \Omega; \mathbf{x} \in \partial_\gamma f\} \quad (10)$$

where the size of the neighborhood γ in f is chosen such that $(\partial_\gamma f) \circ T_0 \supseteq \partial_\delta g$. Then $J_2 = J + \text{const}$. A sufficient condition is to choose γ so that $\gamma \geq 2\delta K$, where K is the Lipschitz constant of T_0^{-1} , provided that the class boundaries of f and $g \circ T$ coincide for some T satisfying (6). In practice, exact match of the boundaries is not required and thanks to the smoothness of T , the minima of J_2 and J are sufficiently close even for $\gamma \approx \delta K$. An appropriate value for γ is easy to determine directly as it can be interpreted as the maximum expected displacement in the image f . So if the optimal T is expected to keep scale, $\gamma \approx \delta$ is a good choice, with δ given by (6).

2.3. Normal approximation

We take advantage of the fact that the criterion is only influenced by movement in the direction perpendicular to the class boundaries, as in the direction parallel to the boundaries, the classes are constant. Let us replace the d -dimensional integral (5) over $\Omega_2 = \partial_\gamma f \cap \Omega$ (10) by an outer integral over the boundary $\partial^\Omega f = \partial f \cap \Omega$ and an inner integral in the normal direction

$$J_3(f, g \circ T) = \int_{\mathbf{z} \in \partial^\Omega f} \int_{-\gamma}^{\gamma} \varrho(f(\mathbf{x}), g(T(\mathbf{x}))) dh d\mathbf{z} \quad (11)$$

$$\text{with } \mathbf{x} = \mathbf{z} + \mathbf{n}(\mathbf{z})h$$

where $\mathbf{n}(\mathbf{z})$ is an oriented normal at point \mathbf{z} of the boundary ∂f , with $\|\mathbf{n}\| = 1$. Since the transformation between \mathbf{x} and the new coordinates (\mathbf{z}, h) is unitary, the difference between J_2 and J_3 is only due to the parts of $\partial_\gamma f$ truncated by Ω and by the unsmooth part of the boundary ∂f . In practice the approximation is very good. By definition, $f(\mathbf{x})$ is determined by the side of the boundary where \mathbf{x} falls, so we write

$$f(\mathbf{x}) = f(\mathbf{z} + \mathbf{n}(\mathbf{z})h) = \begin{cases} l^+ & \text{if } h > 0 \\ l^- & \text{if } h < 0 \end{cases} \quad (12)$$

for $l^+, l^- \in \mathcal{L}^f$, assuming that there are no other boundaries within the distance γ .

By assumption (ii) in Section 2.2 we linearize the transformation T_0 around point \mathbf{z}

$$T_0(\mathbf{x}) = T_0(\mathbf{z}) + (\nabla T_0(\mathbf{z}))(\mathbf{x} - \mathbf{z}) = \mathbf{u} + \mathbf{m}h \quad (13)$$

$$\text{with } \mathbf{m} = (\nabla T_0(\mathbf{z}))\mathbf{n}(\mathbf{z}) \text{ and } \mathbf{u} = T_0(\mathbf{z}) \quad (14)$$

and similarly for T . Supposing that T_0 makes the images approximately aligned, we expect $g \circ T_0$ to depend predominantly on the shift h along the normal and not on the movement in the perpendicular direction. Hence we hypothesize

that locally there exists some scalar function $\tilde{g}(\mathbf{u}, h)$ such that

$$g(\mathbf{y}) = \tilde{g}(\mathbf{u}, \langle \mathbf{y} - \mathbf{u}, \tilde{\mathbf{m}} \rangle) \quad (15)$$

where we have used $\tilde{\mathbf{m}} = \mathbf{m}/\|\mathbf{m}\|^2$ for later convenience.

Assuming further that $\nabla T(\mathbf{z}) \approx \nabla T_0(\mathbf{z})$, *i.e.* that our guess T_0 is approximately correct with respect to orientation, we get after some algebraic manipulations

$$g(T(\mathbf{x})) = \tilde{g}(\mathbf{u}, \xi(\mathbf{z}) + h) \quad (16)$$

$$\text{with } \xi(\mathbf{z}) = \langle T(\mathbf{z}) - T_0(\mathbf{z}), \tilde{\mathbf{m}} \rangle \quad (17)$$

Consequently, the integral in (11) can be approximated as

$$J_4(T) = \int_{\mathbf{z} \in \partial^\Omega f} D(\mathbf{z}, \xi(\mathbf{z})) d\mathbf{z} \quad (18)$$

where the inner integral

$$\begin{aligned} D(\mathbf{z}, \xi(\mathbf{z})) &= \int_{-\gamma}^{\gamma} \varrho(f(\mathbf{x}), g(T(\mathbf{x}))) dh \\ &= \int_{-\gamma}^0 \varrho(l^-, \tilde{g}(\mathbf{u}, \xi(\mathbf{z}) + h)) dh \\ &\quad + \int_0^{\gamma} \varrho(l^+, \tilde{g}(\mathbf{u}, \xi(\mathbf{z}) + h)) dh \end{aligned} \quad (19)$$

can be easily precomputed for various normal shifts $\xi(\mathbf{z})$, to accelerate the subsequent optimization.

2.4. Discretization

The last approximation of the criterion $J_4 \approx J + \text{const}$ from (18) can be evaluated numerically as follows. We pick a set of sparse keypoints $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{|P|}\}$ on the class boundaries, $\mathbf{p}_i \in \partial^\Omega f$. For each \mathbf{p}_i , we find the normal direction $\mathbf{n}_i = \mathbf{n}(\mathbf{p}_i)$ and sample the class labels along the normals for $h \in \{-\gamma, -\gamma + 1, \dots, \gamma - 1, \gamma\}$

$$\tilde{g}_i(h) = \tilde{g}(\mathbf{p}_i, h) = g(T_0(\mathbf{p}_i) + \tilde{\mathbf{m}}_i h) \quad (20)$$

with $\mathbf{m}_i = (\nabla T_0(\mathbf{p}_i))(\mathbf{n}_i)$ and $\tilde{\mathbf{m}}_i = \mathbf{m}_i/\|\mathbf{m}_i\|^2$. Nearest neighbor interpolation is used to evaluate g away from integer coordinates. Locations outside of the image or the region of interest return the last class encountered.

The class boundary $\partial^\Omega f$ is partitioned into small and approximately flat pieces S_i such that $\mathbf{p}_i \in S_i \subset \partial^\Omega f$ and $\bigcup_i S_i = \partial^\Omega f$. Then the continuous criterion (18) can be approximated as a sum

$$J_4(T) \approx J_5(T) = \sum_{i=1}^{|P|} |S_i| D_i(\xi_i) \quad (21)$$

where we have written $\xi_i = \xi(\mathbf{p}_i) = \langle T(\mathbf{p}_i) - T_0(\mathbf{p}_i), \tilde{\mathbf{m}}_i \rangle$ and $D_i(\xi) = D(\mathbf{p}_i, \xi)$ to simplify the notation. The keypoint contributions D_i are approximated by replacing the

integral (19) with a sum over h with unit step size, assuming the units are pixels and the scaling is not too extreme

$$D_i(\xi) = \sum_{h=-\gamma}^0 \varrho\left(l^-, \tilde{g}_i(\mathbf{u}, \xi + h)\right) + \sum_{h=0}^{\gamma} \varrho\left(l^+, \tilde{g}_i(\mathbf{u}, \xi + h)\right) \quad (22)$$

The values of $D_i(\xi)$ are precalculated for $-\delta \leq \xi \leq \delta$. In the interest of notational simplicity, we now consider γ and δ to be integers. Note that while the naive implementation has complexity $O(\delta\gamma)$, *i.e.* quadratic if $\gamma \approx \delta$, it is possible to calculate all $2\delta + 1$ values in linear time $O(\delta + \gamma)$ by precalculating the cumulative sums Q^+ , Q^- given by

$$Q^\pm(k) = \sum_{j=-\gamma-\delta}^k \varrho\left(l^\pm, \tilde{g}_i(\mathbf{u}, j)\right) \quad (23)$$

for $-\gamma - \delta \leq k \leq \gamma + \delta$ and combining them

$$D_i(\xi) = Q^-(\xi) - Q^-(\xi - \gamma) + Q^+(\xi + \gamma) - Q^+(\xi) \quad (24)$$

In our implementation, we precalculate Q^\pm for integer k , which allows us to evaluate D_i for integer ξ . For non-integer ξ , the values are linearly interpolated.

The above procedure needs a preprocessing time proportional to $O(|P|(\delta + \gamma))$ per image. Then, evaluating J_5 is very fast, the most costly operations being to evaluate T in $|P|$ points.

2.5. Optimization

Finding the transformation $T^* = T_{\theta^*}$ that minimizes the criterion J_5

$$\theta^* = \arg \min_{\theta} J_5(T_\theta) \quad (25)$$

is solved by standard multidimensional minimization methods such as Powell's BOBYQUA [25], taking advantage of the fast evaluation of $J(T_\theta)$. Derivatives can be calculated by the chain rule

$$\frac{\partial J_5}{\partial \theta_j} = \sum_i |S_i| \frac{\partial D_i}{\partial \xi_i} \tilde{\mathbf{m}}_i^T \frac{\partial T(\mathbf{p}_i)}{\partial \theta_j} \quad (26)$$

to allow higher-order optimization methods such as BFGS [19], which are usually faster. The derivative $\partial D_i(\xi)/\partial \xi$ is calculated by the first-order difference formula from precalculated values of $D_i(\xi)$ on integers.

A barrier function is introduced to keep all displacements ξ_i within the assumed range by replacing D_i in (21) with

$$\tilde{D}_i(\xi) = D_i(\xi) + \beta \max(0, |\xi| - \gamma)^2 \quad (27)$$

with $\beta = \max_{i,\xi} D_i(\xi)$. The idea is that if a displacement larger than the precalculated range is needed, it will be found by an iterative process as described below.

2.6. Iterative improvement and multiresolution

In the absence of a priori information, we set the initial transformation T_0 to identity. If the true transformation is too far from T_0 , the minimum of the J_5 criterion might be a poor approximation for the minimum of the original criterion J we actually want to minimize. Our solution is to iterate the process if we suspect that the approximation is wrong. This is detected if for the optimal T^* found, a fraction larger than κ_1 of the displacements $T^*(\mathbf{p}_i) - T_0(\mathbf{p}_i)$ are either too close to the last sampled point on the normal ($|\xi_i| > \gamma - \kappa_2$), or if it is too far from the normal ($\|T^*(\mathbf{p}_i) - T_0(\mathbf{p}_i)\| > \kappa_3 |\xi_i|$). The constants are set to $\kappa_1 = 0.2$, $\kappa_2 = 0.1\gamma$, $\kappa_3 = 3$ based on preliminary testing. The algorithm is as follows:

1. Given T_0 , sample classes $\tilde{g}_i(\xi)$ along normals (20).
2. Precalculate $D_i(\xi)$ (22)
3. Find T^* minimizing J_5 (26)
4. If the approximation is wrong, $T_0 \leftarrow T^*$ and repeat from the beginning, otherwise return T^* .

To increase speed, robustness and capture range, multiresolution with a dyadic scale is used, with a few particularities due to our formulation. First, when downsampling the segmented images, we use majority voting. Second, the number of keypoints is reduced by taking every second one as long as there is enough left. The maximum displacement γ stays constant when measured in pixels, but as the image size is reduced on coarser scales, the capture range is actually larger on the coarse scale and is progressively reduced.

2.7. Similarity criteria

The similarity criterion is defined by the pixelwise penalty $\varrho(l_f, l_g)$, where $l_f \in \mathcal{L}^f$ and $l_g \in \mathcal{L}^g$ are the classes observed in corresponding locations of the images f and g . If the same classes correspond, a suitable strategy is to penalize the differences

$$\varrho(l_f, l_g) = \llbracket l_f \neq l_g \rrbracket \quad (28)$$

If the relationship between the classes in the two images is not known, for example because unsupervised segmentation has been used, we use negative mutual information [23] on labels (MIL) [17].

$$\varrho(l_f, l_g) = -\log \frac{P_{l_f l_g}}{P_{l_f} P_{l_g}} \quad (29)$$

with $P_{l_f} = \sum_{l_g} P_{l_f l_g}$, $P_{l_g} = \sum_{l_f} P_{l_f l_g}$. The probability P_{l_f, l_g} is calculated from the joint histogram of f and $g \circ T_0$. In the iterative registration procedure (Section 2.6), the values of $\varrho(l_f, l_g)$ for all combinations of l_f, l_g are calculated

and stored before calculating the contributions D_i in Step 2, making the evaluation very fast. Not allowing ϱ to change in each iteration of Step 3 usually has negligible effect on the result, since the changes of ϱ are small.

2.8. Segmentation and geometrical transformation

The method we are presenting works with segmented images. The segmentation algorithm should be ideally tuned to the images at hand, to well identify key structures which are then registered. As a fallback, the following unsupervised strategy often works reasonably well and is used for the examples in Section 3 except one:

1. Find SLIC [1] superpixels.
2. For each superpixel, calculate suitable descriptors such as the mean color, intensity, or texture features.
3. Cluster the descriptor vectors by the k -means algorithm to find $3 \sim 5$ classes.

If needed, the segmentation is regularized spatially using GraphCut [8].

Given a segmentation, we need to find the keypoints \mathbf{p}_i (Section 2.4). If the superpixels are known, for all pairs of neighboring superpixels which were assigned to different classes, we identify all pixels on the boundary between those superpixels. If there are too few, they are discarded. Otherwise, a keypoint is created at the center of gravity of the boundary pixels. A covariance matrix is calculated and the normal is set to the eigenvector corresponding to the smallest eigenvalue (Fig. 2).

If superpixels are not used, we use a greedy, non-maxima-suppression-like approach. First, we identify all class boundary pixels in the image. Second, we randomly pick one of the boundary pixels and make it a keypoint. Third, we go through the neighboring boundary pixels, discarding and marking as already visited all that are closer than the desired distance between keypoints q (typically around $10 \sim 30$ pixels). As soon as a boundary point at a distance q or larger is seen, it becomes a new keypoint and the procedure is iterated. The points closer than q are used to calculate the center and the normal as described above.

2.9. Geometric transformation

In this work, we use a rigid transformation in 2D and 3D, parameterized by 1 or 3 angles, respectively, and a translation, e.g.

$$T(\mathbf{x}) = \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \mathbf{x} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (30)$$

with $\theta = (\phi, t_x, t_y)$ and similarly for 3D. Derivatives with respect to θ needed for the optimization can be calculated analytically. Any linear transformation can be used, for example the affine transformation or B-splines [18].

3. Experiments

We illustrate the registration pipeline on two differently stained (cytokeratin and HER2) histological slices of human prostate of size approximately 2000×2000 pixels (Fig. 1, top). The remaining rows of Fig. 1 show the automatic segmentation of the two images, and the superimposed segmentations and images before and after registration. Fig. 2 illustrates the superpixels, keypoints and normals. In Fig. 1, bottom row, corresponding manually selected landmarks are connected in both images before and after registration. Note, that these landmarks are used only for evaluation, not during the registration process. The registration error is almost zero thanks to the transformation being close to rigid. The mean geometric distance $e = \text{mean}_j \|T(\mathbf{v}_i) - \mathbf{w}_i\|$ between transformed manual landmarks \mathbf{v}_i from the fixed image and the corresponding landmarks in the moving image \mathbf{w}_i is used as an error measure.

Unless said otherwise, in all experiments we set the minimum image size in the multiresolution to 128 px, $\gamma = 20$, $k = 5$, and SLIC superpixel size to 15 px with regularization 20.

3.1. Parameter influence

In Table 1, we evaluate the influence of four different parameters on the speed, accuracy and robustness of method on a database of 153 images histological slices, with 508 image pairs to be registered (see Fig. 3 for examples). We report the mean error e , mean $\mu(e)$, 90% trimmed mean $\mu_{90}(e)$, the robustness (proportion of runs that decrease the error e at least by 50%), and segmentation and registration times. Only one parameter is varied at a time.

If the *minimum size* of the image in the image pyramid exceeds a critical value (512 pixels), the coarsest level displacement magnitude increases and the method progressively starts to fail to recover large transformations, which is translated by the decrease of robustness. For smaller values, the accuracy is stable.

While reducing the *maximum sample displacement* γ reduces the complexity of precomputing the contributions D_i , this reduction is often compensated by the increased number of iterations (Section 2.6), making the overall time and accuracy almost constant. However, if γ is reduced below 10 px, it is no longer possible to recover the deformations reliably even using multiresolution, so the error and robustness deteriorate.

For our data, the optimal *number of classes* k is 5. Increasing k further increases the segmentation time but does not improve the accuracy or robustness. On the other hand, the performance deteriorates a lot when using only two classes.

Setting the *superpixel size* is a trade-off. Bigger superpixels lead to larger registration errors but decrease the segmentation and registration time. Smaller superpixels ap-

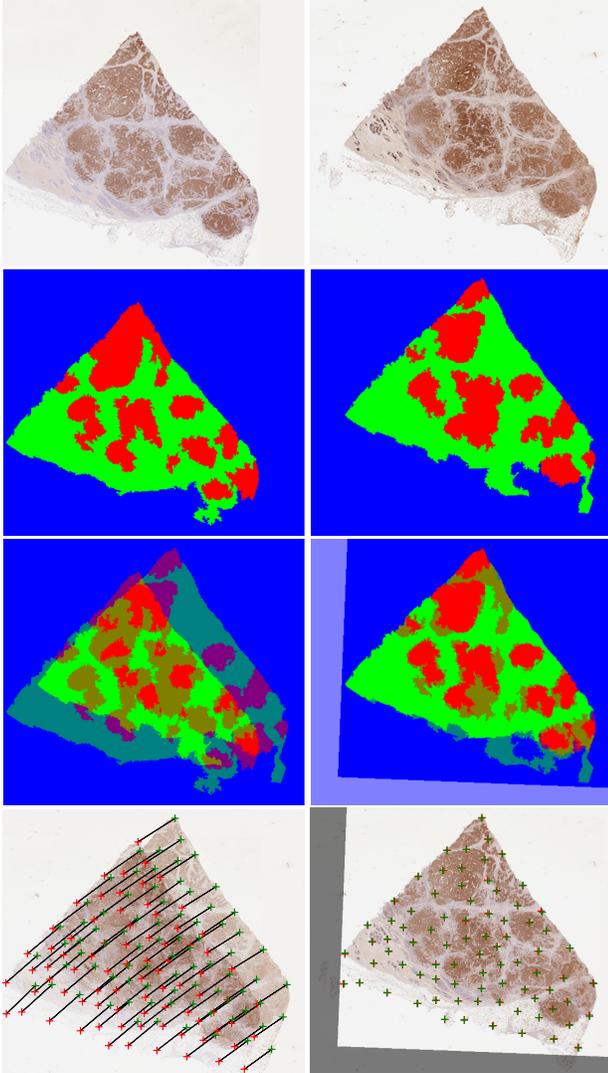


Figure 1. *Top*: Human prostate histological images with different stainings and from different slices to be registered. *2nd row*: Automatic unsupervised segmentation of the images above using SLIC superpixels and k -means based on color. *3rd row*: Overlay of the segmentations before and after registration. *Bottom*: Overlay of the images before and after registration, with corresponding landmarks connected.

proximate the image better and improve the accuracy but the segmentation and registration are slower.

3.2. Comparison with alternative methods

We compare the precision and speed of our algorithm with seven other existing methods using available implementations (see Table 2) — bUnwarpJ [5] and RVSS included in Fiji¹ [4], affine and B-spline registration from Elastix [15], and the simultaneous segmentation and registration [17]. We use the same data as above. Examples of

¹<http://fiji.sc>

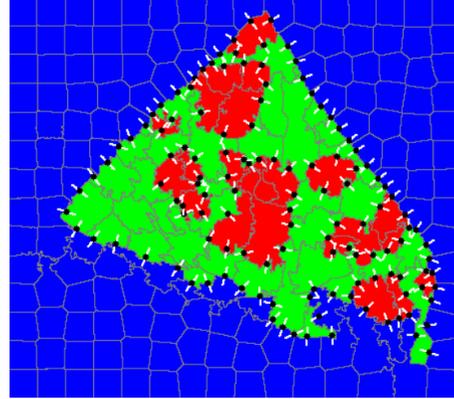


Figure 2. Fixed image segmentation with keypoints (black), edge normals (white) and superpixels (gray). (For better visualization, slightly larger superpixels were chosen.)

Min. image size	64	128	256	512	1024
median(e) [px]	6.74	6.69	6.94	7.38	8.34
$\mu(e)$ [px]	27.39	28.94	31.77	36.06	43.15
$\mu_{90}(e)$ [px]	7.88	7.79	8.48	9.83	13.14
Robustness [%]	88.73	88.91	87.04	85.02	78.22
T_{seg} [s]	17.28	16.77	16.62	16.83	16.59
T_{reg} [s]	3.05	3.18	2.97	2.28	3.04

Max. displac. γ	5	10	25	35	50
median(e) [px]	12.43	7.47	6.71	6.66	6.76
$\mu(e)$ [px]	51.58	34.92	28.98	26.38	29.95
$\mu_{90}(e)$ [px]	20.34	9.46	7.79	7.61	7.69
Robustness [%]	67.38	86.28	88.87	88.94	89.06
T_{seg} [s]	15.63	15.70	15.55	16.14	15.71
T_{reg} [s]	3.41	2.29	2.23	2.82	3.22

Num. classes k	2	3	5	7	10
median(e) [px]	7.69	6.86	6.69	6.87	6.89
$\mu(e)$ [px]	41.55	31.43	28.94	29.22	30.17
$\mu_{90}(e)$ [px]	11.04	8.38	7.79	8.00	8.09
Robustness [%]	81.00	86.49	88.91	89.06	89.90
T_{seg} [s]	11.00	13.95	16.77	19.81	19.11
T_{reg} [s]	2.00	2.22	3.18	3.28	3.25

Superpixel size	10	15	20	40	70
median(e) [px]	6.60	6.69	6.96	7.89	9.63
$\mu(e)$ [px]	26.75	28.94	30.64	32.96	36.44
$\mu_{90}(e)$ [px]	7.59	7.79	8.22	9.39	11.34
Robustness [%]	89.57	88.91	89.26	86.34	83.22
T_{seg} [s]	26.59	16.77	12.31	9.99	8.82
T_{reg} [s]	4.16	3.18	2.06	1.55	0.96

Table 1. The dependence of the method performance on selected parameters. We report the the mean, median and trimmed mean of the geometrical error e , T_{seg} is the duration of the segmentation, and T_{reg} the time spent to register images. Best results are set in bold.

the registration results are in Fig 3. Note that our method performs faster than any other evaluated method, while the precision remains comparable.

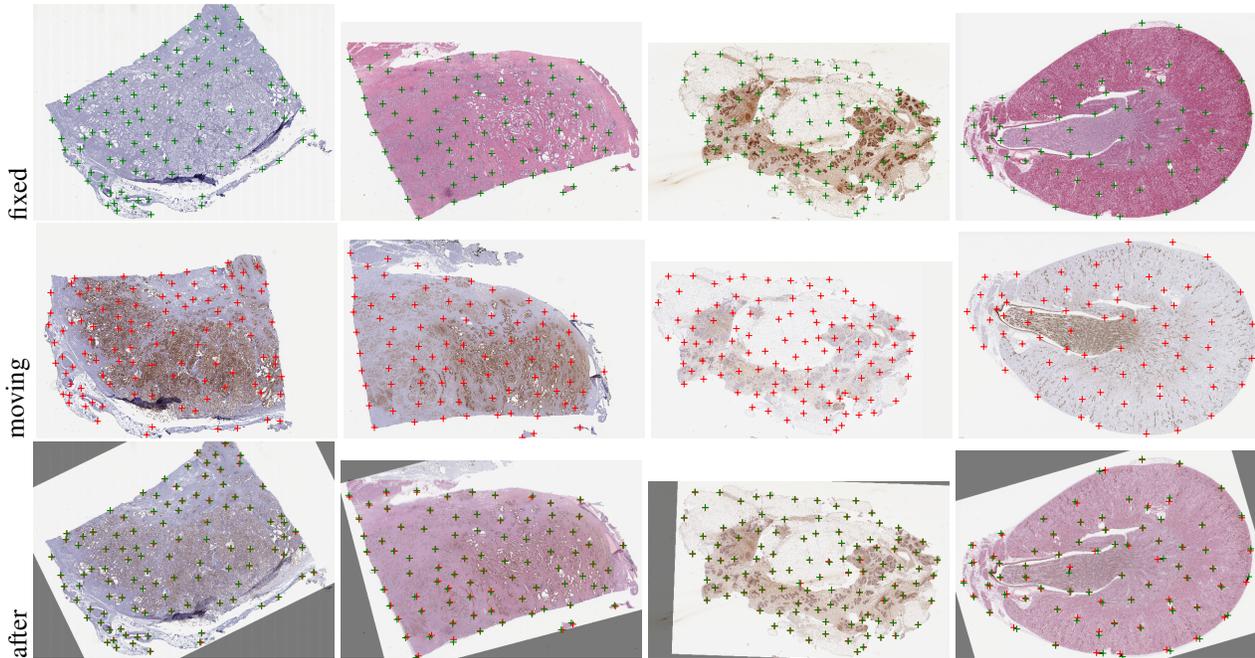


Figure 3. Registration result examples on histological slices evaluated using manually selected landmarks. 1st row: fixed images, 2nd row: moving images, 3rd row: overlay of the images before registration and 4th row: overlay after registration. Landmarks should coincide.

Method	Time	$\mu(e)$	median(e)
ASSAR (Affine) [17]	45.78	37.04	13.45
bUnwarpJ [5]	572.98	51.44	10.19
ASSAR (B-splines) [17]	92.87	44.08	7.62
elastix (Affine) [15]	332.89	45.39	5.23
elastix (B-splines) [15]	555.64	52.17	4.80
RVSS [4]	91.26	83.86	4.89
openCV-SURF [6]	120.02	26.38	4.49
NEW	11.92	27.39	6.74

Table 2. Mean running time in seconds; mean registration error $\mu(e)$, and median registration error in pixels for each method. (Note that the timings are not directly comparable with Table 1, since a different computer had to be used.)

Superpixels	k -means	Precompute	Iteration	Total
6.38	3.94	0.94	0.66	11.92

Table 3. Running time of the algorithm in detail. Note that we run k -means 30 times to increase robustness.

You can see detailed breakup of the elapsed time of our method in Table 3. Note that the registration itself only takes about 15 % of the time.

3.3. Registering Drosophila imaginal discs

Stained images of Drosophila imaginal discs were registered using binary segmentations (Fig. 4) obtained by pixel-wise nearest neighbour classification; two cluster prototypes are selected manually for images pairs, superpixel size is set to 30 px, superpixel regularization to 20. We register

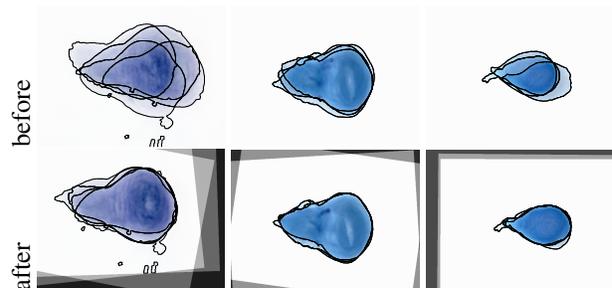


Figure 4. Registration of three sets of 2 ~ 3 images of Drosophila imaginal discs to a reference one. Overlay of the images before and after registration, shown together with the segmentation contours.

image series of the same disc types and the same gene expression to a reference image. The motivation is to identify the spatial expression patterns of the different genes in different disc types by combining several realizations.

4. Conclusions

We have presented a new approach to accelerating image registration by reducing the image information to a small number of 1D samples on lines perpendicular to boundaries. Surprisingly, the registration accuracy is very little affected. While we show that our approach is already faster than the alternatives, we have not yet fully realized the potential for speedup by limiting the amount of data being processed. The main culprit is the repeatability and speed of the seg-

mentation algorithm, which is crucial for the success of the complete pipeline.

Our method requires segmentation but this segmentation does not need to be perfect. In fact, it is enough to simplify the images by reducing it to a small number of classes, as long as all important edges remain. It is perfectly acceptable if the image is oversegmented and a one-to-one correspondence between the two segmentations is not needed.

Our method can be applied to any transformation model but in this work only rigid registration results are shown. Nonlinear registration will be addressed in the future.

Further speedup is possible by using modern powerful optimization methods and hardware acceleration (GPUs). We believe that our work is a step towards bringing image registration into new domains such as interactive image processing pipelines with a human in the loop, as well as fast processing of very large-scale image data collections.

Acknowledgements

This work was supported by the Czech Science Foundation project 14-21421S.

References

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, Nov. 2012. 5
- [2] I. Arganda-Carreras, R. Fernández-González, A. Muñoz Barrutia, and C. Ortiz-De-Solorzano. 3D reconstruction of histological sections: Application to mammary gland tissue. *Microscopy research and technique*, 73(11):1019–29, Oct. 2010. 2
- [3] I. Arganda-Carreras, R. Fernandez-Gonzalez, and C. Ortiz-de Solorzano. Automatic registration of serial mammary gland sections. In *IEEE Engineering in Medicine and Biology Society*, volume 3, pages 1691–4, Jan. 2004. 2
- [4] I. Arganda-Carreras, C. Sorzano, R. Marabini, M. José Carazo, C. Ortiz-de Solorzano, and J. Kybic. Consistent and elastic registration of histological sections using vector-spline regularization. In *Computer Vision Approaches to Medical Image Analysis*, volume 4241, pages 85–95, 2006. 6, 7
- [5] I. Arganda-Carreras, C. O. S. Sorzano, R. Marabini, J. M. Carazo, C. O. de Solorzano, and J. Kybic. Consistent and elastic registration of histological sections. In R. Beichel and M. Sonka, editors, *CVAMIA: Computer Vision Approaches to Medical Image Analysis*, number 3117 in Lecture Notes in Computer Science, pages 85–95, Heidelberg, Germany, May 2006. Springer. http://dx.doi.org/10.1007/11889762_8. 6, 7
- [6] H. Bay, T. Tuytelaars, and L. V. Gool. SURF: Speeded up robust features. *Computer Vision*, 3951:404–17, 2006. 7
- [7] F. Bollenbeck and U. Seiffert. Joint registration and segmentation of histological volume data by diffusion-based label adaptation. In *International Conference on Pattern Recognition*, pages 2440–2443, Washington, DC, USA, 2010. IEEE Computer Society. 2
- [8] Y. Boykov, O. Veksler, and R. Zabih. Efficient approximate energy minimization via graph cuts. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 20(12):1222–1239, November 2001. 5
- [9] T. F. Cootes and C. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *In Proc. SPIE Medical Imaging*, pages 236–248, 2001. 2
- [10] C. Domokos, J. Nemeth, and Z. Kato. Nonlinear shape registration without correspondences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(5):943–958, 2012. 1
- [11] M. Droske and W. Ring. A Mumford-Shah level-set approach for geometric image registration. *SIAM Appl. Math.*, 2007, 2005. 2
- [12] B. Glocker, N. Komodakis, G. Tziritas, N. Navab, and N. Paragios. Dense image registration through MRFs and efficient linear programming. *Medical Image Analysis*, 12(6):731–741, 2008. 2
- [13] H. Gonçalves, J. A. Gonçalves, and L. Corte-Real. Hairis: A method for automatic image registration through histogram-based image segmentation. *IEEE Transactions on Image Processing*, 20(3):776–789, 2011. 1
- [14] M. P. Heinrich, M. Jenkinson, J. M. Brady, and J. A. Schnabel. Non-rigid image registration through efficient discrete optimization. In *MIUA: Medical Image Understanding and Analysis*, pages 187–192, 2011. 2
- [15] S. Klein, M. Staring, and K. Murphy. Elastix: a toolbox for intensity-based medical image registration. *Medical Imaging, IEEE*, 29(1), 2010. 6, 7
- [16] D. Kozinska, O. J. Tretiak, J. Nissanov, and C. Ozturk. Multidimensional alignment using the euclidean distance transform. *CVGIP: Graphical Model and Image Processing*, 59(6):373–387, 1997. 2
- [17] J. Kybic and J. Bovevec. Automatic simultaneous segmentation and fast registration of histological images. In *International Symposium on Biomedical Imaging (ISBI)*, pages 774–777. IEEE, April–May 2014. 1, 2, 4, 6, 7
- [18] J. Kybic and M. Unser. Fast parametric elastic image registration. *IEEE Transactions on Image Processing*, 12(11):1427–1442, November 2003. 5
- [19] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale minimization. *Mathematical Programming*, (45):503–528, 1989. 4
- [20] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1
- [21] B. Ma, Z. Lin, S. Winkelbach, W. Lindenmaier, and K. E. J. Dittmar. Automatic registration of serial sections of mouse lymph node by using Image-Reg. In *Micron (Oxford, England)*, volume 39, pages 387–96, June 2008. 2
- [22] D. Mahapatra and Y. Sun. Joint registration and segmentation of dynamic cardiac perfusion images using MRFs. In *MICCAI: Medical image computing and computer-assisted intervention: Part I*, pages 493–501, Berlin, Heidelberg, 2010. Springer-Verlag. 2

- [23] J. 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, Aug. 2003. 4
- [24] K. Pohl, J. Fisher, W. Grimson, R. Kikinis, and W. Wells. A bayesian model for joint segmentation and registration. *Neuroimage*, 31(1):228–239, 2006. 2
- [25] M. J. D. Powell. The BOBYQA algorithm for bound constrained optimization without derivatives. Technical Report DAMTP 2009/NA06, Cambridge University, Aug. 2009. 4
- [26] M. Sabuncu and P. Ramadge. Gradient based nonuniform subsampling for information-theoretic alignment methods. In *Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1683–6, 2004. 1
- [27] D. P. Shamonin, E. Bron, B. P. F. Lelieveldt, M. Smits, S. Klein, and M. Staring. Fast parallel image registration on CPU and GPU for diagnostic classification of Alzheimer’s disease. *Front. Neuroinform.*, 2014, 2014. 2
- [28] J. Thirion. Image matching as a diffusion process: an analogy with Maxwell’s demons. *Medical image analysis*, 2(3):243–260, 1998. 2
- [29] F. Wang and B. C. Vemuri. Simultaneous registration and segmentation of anatomical structures from brain MRI. In *MICCAI.*, pages 17–25, 2005. 2
- [30] A. Yezzi. A variational framework for integrating segmentation and registration through active contours. *Medical Image Analysis*, 7(2):171–185, 2003. 2
- [31] B. Zitová and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, (21):977–1000, 2003. 1