

Fast graph-cut based optimization for practical dense deformable registration of volume images

Simon Ekström^a, Filip Malmberg^{a,b,*}, Håkan Ahlström^{a,c}, Joel Kullberg^{a,c}, Robin Strand^{a,b}

^a Department of Surgical Sciences, Uppsala University, Sweden

^b Department of Information Technology, Uppsala University, Sweden

^c Antares Medical, Mölndal, Sweden

ARTICLE INFO

Article history:

Received 21 May 2019

Received in revised form 30 March 2020

Accepted 29 May 2020

Keywords:

Image registration

Optimization

ABSTRACT

Deformable image registration is a fundamental problem in medical image analysis, with applications such as longitudinal studies, population modeling, and atlas-based image segmentation. Registration is often phrased as an optimization problem, i.e., finding a deformation field that is optimal according to a given objective function. Discrete, combinatorial, optimization techniques have successfully been employed to solve the resulting optimization problem. Specifically, optimization based on α -expansion with minimal graph cuts has been proposed as a powerful tool for image registration. The high computational cost of the graph-cut based optimization approach, however, limits the utility of this approach for registration of large volume images. Here, we propose to accelerate graph-cut based deformable registration by dividing the image into overlapping sub-regions and restricting the α -expansion moves to a single sub-region at a time. We demonstrate empirically that this approach can achieve a large reduction in computation time – from days to minutes – with only a small penalty in terms of solution quality. The reduction in computation time provided by the proposed method makes graph-cut based deformable registration viable for large volume images. Graph-cut based image registration has previously been shown to produce excellent results, but the high computational cost has hindered the adoption of the method for registration of large medical volume images. Our proposed method lifts this restriction, requiring only a small fraction of the computational cost to produce results of comparable quality.

© 2020 Published by Elsevier Ltd.

1. Introduction

Deformable image registration is a fundamental task in medical image analysis. Registration can be defined as the task of finding a spatial transformation that aligns two or more images with each other. We consider registration of two images – one referred to as the *source* image and the other referred to as the *target* image. In this case, the deformable registration problem consists of finding a deformation field that indicates, for each voxel in the target image, the location of the corresponding voxel in the source image. Applications of registration include fusion of multi-modal data, change detection and quantification, longitudinal studies, population modeling and studies of normal anatomical variability, and atlas-based image segmentation (Sotiras et al., 2013; Maintz and Viergever,

1998). Another application highly dependent on efficient image registration is the Imiomics analysis concept (Strand et al., 2017) which aims to analyze large datasets of whole-body MRI.

This work aims to present an efficient image registration method to enable high-quality registration of large volume images (e.g., whole-body MRI) at a lower computational cost. This opens up for analysis of very large datasets such as the UK Biobank.¹

Image registration methods can broadly be divided into two categories; *parametric* and *non-parametric* (Sotiras et al., 2013). In parametric methods, the space of feasible transformations is reduced by introducing a parametrization (model) of the transformation. For example, the transformation may be required to be rigid or affine. Non-linear deformations may be parametrized using, e.g., B-splines, where a sparse grid of control points is used to specify a smooth deformation field. Here, we consider non-parametric, or *dense* registration methods, where each point in the source image

* Corresponding author at: Department of Surgical Sciences, Uppsala University, Sweden.

E-mail address: filip.malmberg@it.uu.se (F. Malmberg).

¹ <https://www.ukbiobank.ac.uk/>.

may be displaced arbitrarily, e.g., a separate displacement vector is stored for each voxel in the image.

Dense registration is often phrased as an optimization problem, i.e., finding a deformation field that is optimal according to a given objective function which typically consists of two terms: a *data* term measuring the degree to which the deformed source image resembles the target image, and a *regularization* term enforcing the smoothness of the deformation field itself.

By discretizing the set of allowed displacement vectors for a voxel, we may formulate dense deformable registration as a discrete labeling problem that can be solved using a variety of powerful inference methods. Here, we focus on the class of optimization methods referred to as *move-making* algorithms. These algorithms start from an initial solution and iteratively seek to find better solutions. At each step, the algorithms search for a better solution in a limited subset of the combinatorial search space, referred to as the *move-space*, and selects the one that leads to the largest reduction of the objective function. The algorithms terminate when no such improving solution can be found. Provided that the move-space includes the current solution, these algorithms will converge in a finite number of steps. The resulting solution is then guaranteed to be locally optimal with respect to the move-space, in the sense that no better solution exists within the move-space of the current solution.

An important characteristic of a move-making algorithm is the size of its move-space. A large move space reduces the risk of getting stuck in poor local minima and can lead to faster convergence of the algorithm. A prominent example of a move-making algorithm is the *iterated conditional modes* (ICM) method, which iteratively attempts to improve the current solution by changing the label of a single element at a time while keeping the others fixed. The move-space for ICM is thus small, and the algorithm is prone to get stuck in poor local minima. Another move-making algorithm is the α -*expansion* method by Boykov et al. (2001). This algorithm selects a label α and considers moves that allow all elements to either keep their current label or change label to α . Under certain conditions, the best such move can be determined in low-order polynomial time by solving a max flow/min cut problem on a suitably constructed graph. Compared to ICM, the α -expansion algorithm has a much larger move-space since it allows the labels of multiple elements to be changed simultaneously. In practice, this leads to much stronger local optima (Szeliski et al., 2006). The drawback of the α -expansion method is its high computational complexity. Iteratively solving max flow/min cut problems on large graphs is very computationally expensive, and the computational cost grows nonlinearly with respect to the image size. The high computational cost limits the practical utility of this approach for registration of large volume images (Szmul et al., 2016). Here, we propose an approach for reducing the computation time of graph-cut based registration, while still producing high-quality solutions.

We may view the ICM and α -expansion algorithms as two extremes in terms of their move-spaces; ICM only allows updates to a single variable at a time, while α -expansion allows simultaneous updates to all variables. While the ability to simultaneously update the labels of many voxels yields superior results, we hypothesize here that interactions between voxels that are very far away from each other are unlikely to affect the result significantly. Based on this hypothesis, we introduce a novel move-making strategy that explores the middle-ground between ICM and α -expansion: we divide the image into small overlapping sub-regions, and define the move-space as the set of solutions that can be obtained by an α -expansion within a single sub-region at a time. The optimal move within a sub-region can still be determined by solving a minimal graph cut problem, with special care taken to correctly handle the borders of each subregion. Selecting the size of the sub-regions involves a trade-off between computation time on one

hand and solution quality on the other. We demonstrate empirically that this trade-off is benign, in the sense that a large reduction in computation time can be achieved with only a small penalty in terms of solution quality. The reduction in computation time makes graph cut based deformable registration viable for large volume images. Additionally, we perform an empirical comparison between the proposed method to the state-of-the-art method proposed by Glocker et al. (2011). All experiments are performed on registration of whole-body fat-water magnetic resonance (MR) images.

2. Related work

As shown by Boykov et al. (2001), minimal graph cuts are a powerful tool for solving discrete optimization problems arising in image analysis and computer vision. The use of minimal graph cuts for deformable image registration was, to our knowledge, first proposed by Tang and Chung (2007). The use of graph cuts in this context has been shown to produce excellent results, but the high computational cost associated with this method has limited its use in medical imaging. In recent years, the topic has since then been revisited by several authors. Heinrich et al. (2013), formulate a similar optimization problem, but then prune the graph to a tree on which the optimal labeling (to an approximation of the original problem) can be found by efficient belief propagation. Szmul et al. (2016, 2017) proposed to accelerate graph-cut based image registration by partitioning the image into supervoxels to yield a sparse graph-based image representation. This reduces computation time, but the result is restricted by the assumption that the displacement field is constant within each supervoxel. In contrast, our proposed method is free to assign distinct displacement vectors to all individual voxels. Glocker et al. (2008, 2011) consider optimization problems similar to that presented here but use a linear programming solver to find optimal solutions.

3. Preliminaries

3.1. Notation

We define a scalar volume image \mathbf{I} as a pair (V, I) consisting of a set V of voxels (represented by points in \mathbb{R}^3) and a mapping $I : V \rightarrow \mathbb{R}$ where the value $I(v)$ of I at $v \in V$ represents the image intensity at the voxel v . The voxels in V are assumed to be arranged on a regular grid so that each voxel can be identified by a coordinate in \mathbb{Z}^3 . We let \mathcal{N} denote the set of all adjacent voxels, i.e., $(v, w) \in \mathcal{N}$ if v and w are adjacent. Throughout, we will use the standard 6-neighborhood to define adjacency.

3.2. Deformation model

We denote the source image by $\mathbf{S} = (V_S, S)$ and the target image by $\mathbf{T} = (V_T, T)$. The two images are related by a transformation $W : \mathbb{R}^3 \rightarrow \mathbb{R}^3$. This transformation is assumed to map the target image to the source image, i.e., it is a *backward transform* (Sotiras et al., 2013). The main motivation for using a backward transform is that it simplifies the interpolation problem that needs to be solved to calculate the deformed source image. With a backward transform, this calculation is straightforward; for every voxel in the target image we simply interpolate the value at the corresponding location in the source image. The transformation at every voxel position \mathbf{x} is given as the addition of an identity transform with a displacement field \mathbf{u} , i.e.,

$$W(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x}). \quad (1)$$

We store \mathbf{u} for each voxel in the target image and use tri-linear interpolation to determine the value of the deformation field at non-grid locations. Note that by Eq. (1), it is trivial to transform between \mathbf{u} and W .

4. Deformable registration by discrete optimization

In this section, we formulate the deformable registration problem as a discrete optimization problem.

4.1. Matching criterion

Matching criteria used for deformable image registration typically consist of two terms: a *data* term measuring the degree to which the deformed source image resembles the target image and a *regularization* term enforcing the smoothness of the deformation field itself. Many different data and regularization terms have been proposed in the literature, see, e.g., Sotiras et al. (2013) for an overview. In this section, we present the specific matching criterion used here for all experiments. We emphasize that our proposed efficient optimization scheme could also be used with other matching criteria, so long as they satisfy the criteria given in Section 5.

The images that are to be registered are assumed to be quantitative and of the same modality. Thus, the data term D is simply taken to be the sum of squared differences (SSD) between the target image and the deformed source image:

$$D(\mathbf{u}) = \sum_{v \in V_T} |T(v) - S(v + \mathbf{u}(v))|^2. \quad (2)$$

For multi-channel images, the data term is taken to be the average SSD across all image channels.

For the regularization term R we consider a *diffusion* regularizer (Glocker et al., 2008), penalizing large first order derivatives in the deformation field:

$$R(\mathbf{u}) = \sum_{(v,w) \in \mathcal{N}} \|\mathbf{u}(v) - \mathbf{u}(w)\|^2. \quad (3)$$

The total matching criterion f is thus defined as

$$f(\mathbf{u}) = (1 - \alpha)D + \alpha R, \quad (4)$$

where α is a user-defined parameter controlling the balance between the data and regularization terms. We seek a transformation W' that minimizes $f(W)$.

4.2. Optimization via minimal graph cuts

By discretizing the space of possible deformation vectors, the problem of finding a deformation field minimizing the matching criterion f becomes a discrete labeling problem. In this section, we review how such labeling problems can be solved via minimal graph cuts, employing variations on the α -expansion method proposed by Boykov et al. (2001). The optimization procedure outlined below is computationally expensive, but forms the basis for the more efficient method proposed in Section 5.

Let \mathbf{u} be a deformation field, and let $\delta \in \mathbb{R}^3$ be a vector. We say that a deformation field \mathbf{u}' is δ -adjacent to \mathbf{u} if \mathbf{u}' can be obtained from \mathbf{u} by adding the vector δ to the deformation vector at some of the voxels $v \in V_T$. Formally, \mathbf{u}' is δ -adjacent to \mathbf{u} if it can be written as

$$\mathbf{u}'(v) = \mathbf{u}(v) + L(v)\delta, \quad (5)$$

where L is a binary labeling function that assigns to each voxel in V_T either 0 or 1, i.e., $L : V_T \rightarrow \{0, 1\}$. Given a set Δ of 3D vectors, we say that \mathbf{u}' is Δ -adjacent to \mathbf{u} if it is δ -adjacent to \mathbf{u} for some $\delta \in \Delta$. We say that a deformation field \mathbf{u} is *locally optimal* with respect to Δ if

there is no Δ -adjacent deformation field \mathbf{u}' such that $f(\mathbf{u}') < f(\mathbf{u})$. In practice, we may relax this condition somewhat, and consider \mathbf{u} to be locally optimal if there is no Δ -adjacent deformation field \mathbf{u}' such that $f(\mathbf{u}') + \Upsilon < f(\mathbf{u})$ for some small, user defined tolerance value Υ . A strategy for finding a deformation field that is locally optimal with respect to Δ can be defined as follows:

1. Start with an initial deformation field \mathbf{u} .
2. While there exists a deformation field \mathbf{u}' that is Δ -adjacent to \mathbf{u} such that $f(\mathbf{u}') < f(\mathbf{u}) + \Upsilon$, set $\mathbf{u} \leftarrow \mathbf{u}'$.

This procedure is an instance of a move-making optimization algorithm, where the move-space is given by all Δ -adjacent solutions. Thus, the algorithm is guaranteed to terminate and return a locally optimal deformation field. The crucial issue in the above procedure is to determine, at every iteration of step 2, whether there exists a Δ -adjacent deformation field to the current solution for which the matching criterion is improved. For any given deformation field, the set of Δ -adjacent deformation fields are extremely large – it has $|\Delta|2^{|V|}$ elements – and so an exhaustive search of this set is not feasible. Instead, the Δ -adjacent solution with the lowest matching criterion value is found by solving a minimal graph cut problem on a suitably constructed graph (Kolmogorov and Zabini, 2004).

For the specific matching criterion considered here we observe, for a fixed \mathbf{u} and $\delta \in \Delta$, that the value of the matching criterion for a δ -adjacent deformation field \mathbf{u}' can be expressed as a function of the labeling function L :

$$f(\mathbf{u}') = \sum_{v \in V_T} \phi_v(L(v)) + \sum_{(v,w) \in \mathcal{N}} \phi_{v,w}(L(v), L(w)). \quad (6)$$

In the above equation, the *unary* terms ϕ_v represent the data and constraint term of the matching criterion, while the *binary* terms $\phi_{v,w}$ represent the smoothness term. Formally, the unary terms are given by

$$\phi_v(0) = (1 - \alpha)|T(v) - S(v + \mathbf{u}(v))|^2, \quad (7)$$

$$\phi_v(1) = (1 - \alpha)|T(v) - S(v + \mathbf{u}(v) + \delta)|^2. \quad (8)$$

Similarly, the binary terms are given by

$$\phi_{v,w}(0, 0) = \alpha\|\mathbf{u}(v) - \mathbf{u}(w)\|^2, \quad (9)$$

$$\phi_{v,w}(1, 1) = \alpha\|(\mathbf{u}(v) + \delta) - (\mathbf{u}(w) + \delta)\|^2, \quad (10)$$

$$\phi_{v,w}(1, 0) = \alpha\|(\mathbf{u}(v) + \delta) - \mathbf{u}(w)\|^2, \text{ and } \quad (11)$$

$$\phi_{v,w}(0, 1) = \alpha\|\mathbf{u}(v) - (\mathbf{u}(w) + \delta)\|^2. \quad (12)$$

We are interested in finding a labeling L for which the matching criterion $f(\mathbf{u}')$ is minimal. As established by Kolmogorov and Zabini (2004) a globally optimal solution to binary labeling problems of the form given in Eq. (6) can be found by solving a maximum flow/minimum cut problem, provided that all binary terms are *submodular*. In Appendix A, we give a proof that the binary terms given by Eqs. (9)–(12) are submodular for any \mathbf{u} and δ .

This result enables an efficient approach for determining if a given deformation field \mathbf{u} is locally optimal with respect to Δ : For every $\delta \in \Delta$, we solve a max-flow problem to find, among all deformation fields that are δ -adjacent to \mathbf{u} , one for which f is minimal. This results in $|\Delta|$ deformation fields. If the value of the matching criterion for any of these fields is smaller than $f(\mathbf{u})$, we accept the deformation field as the new current solution. If not, then \mathbf{u} is locally optimal with respect to Δ .

To implement this approach in practice, we need to select a set of vectors Δ . The number of vectors in Δ and their capture range has a significant effect on the registration process. Generally, a larger set of vectors allows the algorithm to “explore” a larger region around

the current solution, potentially leading to stronger optima. On the other hand, we wish to keep the set of vectors small for efficient optimization. Here, we have opted for a small set of displacement vectors Δ , representing small local displacements of the current deformation field along the main axis directions. Formally, we take Δ to be the set of vectors δ of the form

$$\delta = \pm \epsilon \mathbf{e}_i, \quad (13)$$

where $\epsilon \in \mathbb{R}$ is a user specified step length and \mathbf{e}_i is one of the standard basis vectors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$. Typically, we set ϵ to a fraction of the voxel spacing, to allow for sub-voxel precision in the registration.

To find a good solution with such a small set Δ , the algorithm must be initialized with a deformation field that is close to the desired one. To alleviate this requirement and increase the “capture range” of the registration, we follow the common practice of combining the move-making optimization with a multi-resolution strategy. Starting with the original image, we obtain a lower resolution image by applying a Gaussian filter at the current resolution and then discarding every second voxel of the filtered image along each dimension. This procedure is repeated $k - 1$ times to obtain k resolution levels, where k is specified by the user. Registration then starts at the coarsest resolution, and applies the optimization method described above to obtain a locally optimal transformation at the current resolution. The resulting deformation field is up-sampled using tri-linear interpolation to obtain a starting guess for the next finer resolution. The process is repeated until a locally optimal solution has been calculated at the finest resolution.

5. Proposed efficient optimization method

The cost of solving a single max flow/min cut problem grows non-linearly with the number of nodes in the graph. Thus, iteratively finding optimal adjacent solutions for the entire volume is computationally expensive. To reduce computational cost, we propose to divide the volume into smaller sub-regions and restrict the move-space so that only voxels within the same sub-region are allowed to simultaneously update their displacement vectors within a single move. This is motivated by the hypothesis that interactions between voxels that are very far away from each other are unlikely to affect the result significantly.

Let $V' \subset V_T$ be a subset of the voxels in the target image. Given a deformation field \mathbf{u} and a vector δ , we say that a deformation field \mathbf{u}' is (V', δ) -adjacent to \mathbf{u} if \mathbf{u}' can be obtained from \mathbf{u} by adding the vector δ to the deformation vector at some of the voxels $v \in V'$. Formally, \mathbf{u}' is (V', δ) -adjacent to \mathbf{u} if it can be written on the form given in Eq. (5), with the additional condition that $L(v) = 0$ for all voxels $v \notin V'$.

Given a set Δ of 3D vectors and set β of subsets of V_T , we say that \mathbf{u}' is (β, Δ) -adjacent to \mathbf{u} if it is (V', δ) -adjacent to \mathbf{u} for some $\delta \in \Delta$ and $V' \in \beta$. We say that a deformation field \mathbf{u} is *locally optimal* with respect to (β, Δ) if there is no (β, Δ) -adjacent deformation field \mathbf{u}' such that $f(\mathbf{u}') < f(\mathbf{u}) + \Upsilon$.

The move-making strategy for finding a deformation field that is locally optimal with respect to (β, Δ) can be defined as follows:

1. Start with an initial deformation field \mathbf{u} .
2. While there exists a deformation field \mathbf{u}' that is (β, Δ) -adjacent to \mathbf{u} such that $f(\mathbf{u}') + \Upsilon < f(\mathbf{u})$, set $\mathbf{u} \leftarrow \mathbf{u}'$.

As before, we use minimal graph cuts to efficiently search the set of (β, Δ) -adjacent deformation fields for a solution with better matching criterion value. Let \mathcal{N}' be the set of adjacent voxel pairs in \mathcal{N} for which at least one of the voxels in the pair is in V' . For a fixed \mathbf{u} , voxel subset $V' \in \beta$ and vector δ , the value of the matching criterion

for a (V', δ) -adjacent deformation field \mathbf{u}' can again be expressed as a function of the labeling function L :

$$\begin{aligned} E(L) = & \sum_{v \in V_T \setminus V'} \phi_v(0) + \sum_{(v,w) \in \mathcal{N} \setminus \mathcal{N}'} \phi_{v,w}(0, 0) \\ & + \sum_{v \in V'} \phi_v(L(v)) + \sum_{(v,w) \in \mathcal{N}'} \phi_{v,w}(L(v), L(w)). \end{aligned} \quad (14)$$

The first two terms in the above equation are constant with respect to L and thus do not affect the optimization of L . Let \mathcal{N}'' be the set of adjacent voxel pairs in \mathcal{N} for which *both* voxels in the pair are in V' . We can then rewrite the remaining terms in Eq. (14) as

$$\begin{aligned} & \sum_{v \in V'} \phi_v(L(v)) + \sum_{(v,w) \in \mathcal{N}''} \phi_{v,w}(L(v), L(w)) \\ & + \sum_{(v,w) \in \mathcal{N}' \setminus \mathcal{N}''} \phi_{v,w}(L(v), 0) \end{aligned} \quad (15)$$

In the last term of the above equation, we make the assumption that w is the voxel in the pair (v, w) that is not in V' . This last term is a sum over binary terms, but since one of the labels, $L(w)$, is fixed to 0 we can treat each $\phi_{v,w}(L(v), 0)$ as a unary term for the remaining label $L(v)$. Thus a labeling L that globally minimizes Eq. (15) can be found by computing a minimal graph cut. Pseudo-code for the full proposed optimization algorithm is listed in Algorithm 1.

Algorithm 1. Optimization algorithm

Input: An initial deformation field \mathbf{u} . A set of vectors Δ , a set of voxel subsets β , and a tolerance value Υ .

Output: A locally optimal deformation field \mathbf{u}

$\mathbf{u}(v) = 0$ for all $v \in V$

done = false **while** done = false **do**

 done = true

for all $V' \in \beta$ **do**

for all $\delta \in \Delta$ **do**

 Solve a min-cut/max-flow problem to find, among all deformation fields that are (V', δ) -adjacent to \mathbf{u} , a deformation field \mathbf{u}' for which $f(\mathbf{u}')$ is minimal.

if $f(\mathbf{u}') + \Upsilon < f(\mathbf{u})$ **then**

 Set $\mathbf{u} = \mathbf{u}'$ Set done = false

end if

end for

end for

end while

To select the set of voxel subsets β in practice, we consider a grid of rectangular blocks covering the target image. Each block in B , has a size of $n \times n \times n$ voxels. Taking $\beta = B$ can potentially lead to artifacts at the boundaries between the blocks. To avoid this, we also consider a second set of blocks, B_{shift} , with blocks of the same size as in B but translated by $n/2$ along each dimension, and take $\beta = B \cup B_{\text{shift}}$. The size n of the blocks offers a trade-off between computational cost and solution quality. If n is too small, the optimization algorithm will tend to get stuck in poor local optima. Selecting a too large n , on the other hand, may result in very long computation times but not much gain in solution quality. In Section 6, we explore this trade-off empirically.

5.1. Computational optimizations

The move-making optimization procedure described above divides the voxels of the target volume into a set of overlapping rectangular blocks β . The algorithm then repeatedly iterates over all blocks, and tries to improve the current solution by modifying the displacement vectors within a single block at a time by solving a graph cut problem. Reducing the size of the graph cut problems to be solved during optimization greatly reduces the computation time. The division into subregions also facilitates some additional computational optimizations, described in this section, that further accelerate the algorithm.

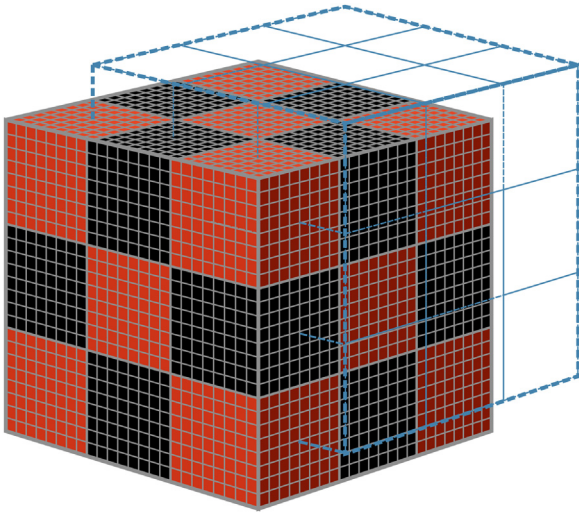


Fig. 1. Illustration of red-black ordering of the blocks B . In this illustration, each of the $3 \times 3 \times 3$ blocks contains $n \times n \times n = 8 \times 8 \times 8$ voxels. The dotted blue lines show the second set of blocks B_{shift} , translated relative to B by half the block size in each dimension. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To solve each graph cut problem, we use the fast graph cut algorithm proposed by Boykov and Kolmogorov (2004). As can be observed in Eqs. (14) and (15), the optimization of a single block V' does not depend on the current deformation vectors at all target image voxels. In fact, the optimization of a given block $V' \in \beta$ only depends on the deformation vector at a voxel v if

1. v is in V' , or
2. v is adjacent to a voxel in V' .

We denote by $\Gamma(V')$ the set of voxels in V_T satisfying one of the above criteria. If, for two blocks V' and V'' , it holds that $\Gamma(V') \cap \Gamma(V'') \neq \emptyset$, then we say that V' and V'' are *dependent*.

The first computational optimization is to parallelize the iteration over the voxel subsets in β . Recall that we define β as the union of two grids of rectangular blocks B and B_{shift} , translated relative to each other by half the block size in each dimension. We arrange the loop over β in Algorithm 1 so that we first iterate over all blocks in B , and then over all blocks in B_{shift} . We observe that when iterating over a single set of blocks $B' \in \{B, B_{\text{shift}}\}$, two blocks V' and V'' in B' are dependent precisely if they are adjacent, i.e., there is a pair of adjacent voxels in \mathcal{N} for which one voxel is in V' and the other is in V'' . This allows us to use a *red-black ordering* to parallelize the iteration over a single set of blocks; the set of blocks B' is further divided into two groups, identified as *red* and *black* blocks, such that black blocks are only adjacent to – and thus dependent on – red blocks, and vice versa. See Fig. 1. All black blocks can then be updated independently, as can all red blocks. Our implementation uses OpenMP² to distribute the work among multiple CPU cores.

The second computational optimization is based on the observation that after a few iterations over the blocks, most blocks have already converged to their locally optimal state, and remain unchanged for subsequent iterations. Thus, a large amount of redundant work is spent on trying to improve the deformation field within blocks that have already reached a locally optimal state. Specifically, if a block V' was not changed at one iteration, it is guaranteed to remain unchanged also at the next iteration unless a dependent block $V'' \in \beta$ was changed since the last update of V' .

Keeping track of these conditions allows to implement *early termination*, i.e., not performing the graph cut optimization step for blocks that will not be changed anyway.

6. Experiments and results

Several previous studies have established the usefulness of graph-cut based optimization for deformable image registration (Tang and Chung, 2007; So et al., 2011). Thus, the focus of this section is not to study the performance of graph-cut based registration per se. Instead, we focus on evaluating how the proposed division into subregions affects the optimality of the result. Additionally, we perform an empirical comparison between our proposed method and the method of Glocker et al. (2011). We evaluate the performance of the proposed optimization method on a set of 21 whole body MR images of female subjects from the POEM cohort (www.medsci.uu.se/poem).

All subjects were imaged on a 1.5T clinical MRI system (Philips Achieva, Philips Healthcare, Best, Netherlands) in the supine position using the body coil and a whole body water-fat imaging protocol that used a spoiled 3D multi-gradient echo sequence. Scan parameters were: TR/TE1/ Δ TE = 5.9/1.36/1.87 ms, 3 unipolar echoes, flip angle 3. Imaged field of view (FOV) 530 mm \times 377 mm \times 2000 mm, reconstructed voxel size 2.07 mm \times 2.07 mm \times 8.0 mm in sagittal \times coronal \times axial directions. The imaging protocol and the water-fat image reconstruction have been described previously (Berglund et al., 2010; Kullberg et al., 2009).

Fat- and water fraction volume images were calculated from the MR image data as described by Berglund et al. (2010). Background noise outside the body was suppressed by multiplication with a body mask, obtained using the approach described by Strand et al. (2017). We treat each pair of fat- and water fraction images as a single two-channel volume image, and use these as input to the registration algorithm.

One subject was selected as the reference subject, to which the other twenty subjects were registered. The registration parameters used were a step length of $\epsilon = 0.5$ (specified in image coordinates), $k = 6$ resolution levels in the multiresolution pyramid, regularization parameter $\alpha = 0.1$, and the tolerance level Υ was set to 10^{-5} . All pairs were registered in both forward and reverse directions. The quality of the results was assessed using the value of the matching criterion $f(\mathbf{u})$ of the final deformation field and the inverse consistency vector magnitude error (VME) (Christensen and Johnson, 2001), computed as

$$\text{VME} = \frac{1}{|V|} \sum_{x \in V} |x - T_{B \rightarrow A} \circ T_{A \rightarrow B}(x)|, \quad (16)$$

where $T_{B \rightarrow A}(T_{A \rightarrow B}(x))$ is the composite of the deformation fields in forward and reverse directions.

6.1. Evaluation of the effect of sub-region size

Registrations were performed on the 20 subject pairs with varying size on the sub-regions and compared to registration without sub-regions, i.e., direct α -expansion on the full image as proposed by Tang and Chung (2007). Block-shaped sub-regions of size $\kappa \in \{1^3, 8^3, 16^3, 32^3\}$ were used. During the experiment, the computation time for registration with direct α -expansion was shown to exceed 5 days per subject and for practical reasons, these computations were thus terminated. The results of the remaining experiments are shown in Fig. 3. As the figure shows, the computation time increases dramatically as we increase in the sub-region size. In terms of VME and matching criterion, $f(\mathbf{u})$, the effect is less noticeable. Both measures are drastically improved as we move

² <http://openmp.org>.

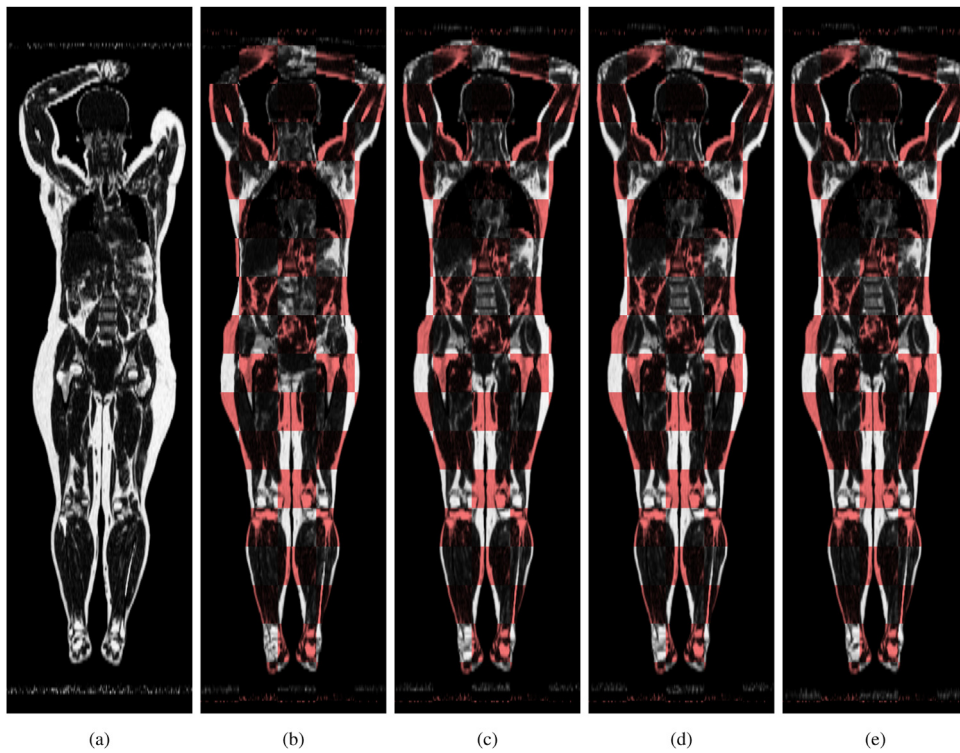


Fig. 2. Coronal slices from the fat percent images from one of the subjects registered in the experiment. Presented are the unregistered fat percent image (a) and checkerboards showing the registered fat images (white) and the reference images (red) for sub-region sizes 1^3 (b), 8^3 (c), 16^3 (d), and 32^3 (e). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

from a sub-region size of 1^3 (equivalent to ICM) to a size of 8^3 , but increasing the sub-region size further only affects the VME and matching criterion marginally.

Fig. 2 shows a sample from the registered subjects from the experiment. Presented are the original unregistered fat percent image and checkerboards showing the registered fat images (white) and the reference images (red) for the various sub-region sizes. The checkerboards show the similarity between the registered subject and the reference subject. Visually, these results agree with the VME and matching criterion results. The registration result with sub-region size 1^3 (b) exhibits visual errors throughout the image, most notably in arms and lower legs. For the other sub-region sizes (c–e) there were small errors around the calves but none of the other large errors from (b) are visible. In comparison, there are no evident differences between (c–e), indicating that the choice of sub-region size, within a reasonable range, have little or no effect on the visual end result.

To facilitate a comparison between our proposed method and direct α -expansion, we repeated the experiment but performed the registration to the second-last level in the resolution pyramid, i.e., we register the volumes downsampled by a factor two along each dimension. In this experiment, registrations were performed using sub-regions of size $\kappa \in \{1^3, 8^3, 16^3, 32^3, 64^3\}$, as well as using direct α -expansion. The results of this second experiment are shown in Fig. 4. The results are in agreement with the previous experiment; computation time increases dramatically with increased sub-region size, but the matching criterion and VME are not substantially decreased for sub-region sizes beyond 8^3 .

For this experiment, all registrations were performed on an Intel Xeon W-2133 with 6 cores and hyperthreading enabled (12 hardware threads).

6.2. Comparison with the method of Glocker et al. (2011)

In a second experiment, we compare the performance of our proposed method to the method proposed by Glocker et al. (2011), using a freely available reference implementation (*DROP registration*, <http://www.mrf-registration.net/>). As this particular implementation only supports single-channel images, all registrations in this experiment were performed on the water fraction images only.

Both methods were configured to optimize, as closely as possible, the same matching criterion. For our method, we used the same parameters as for the experiment in Section 6.1, with sub-regions of size 16^3 . An important difference between the two implementations is that our proposed method iterates the optimization procedure until convergence (as determined by the tolerance level Υ), while the DROP registration software performs a user-defined number of iterations at each resolution level. By recommendation from the authors of the DROP registration software, the number of iterations was set to 10 for this experiment.

The results of the comparison, in terms of computation time, inverse consistency (VME), and SSD are shown in Fig. 5. As the figure shows, the computation time for our proposed method is lower by almost a factor 2. The VME is slightly higher for our proposed method, while the SSD is somewhat lower. Fig. 6 shows a sample from the registered subjects from the experiment, using both methods. In a visual comparison, no clear difference is visible between the two methods. We conclude that in this experiment setup, both methods produced comparable registration results, while the computation time was substantially lower for our proposed method.

For this experiment, all registrations were performed on an Intel Core i7-7700K 4.2GHz, 32GB RAM, with 4 cores and hyperthreading enabled (8 hardware threads).

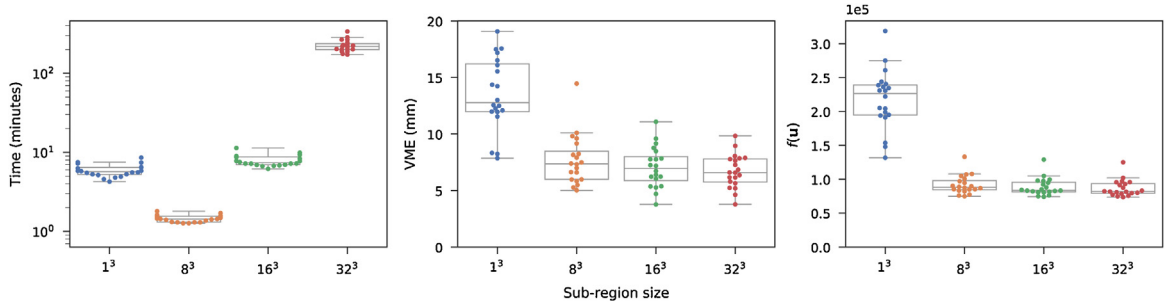


Fig. 3. Total computation time, VME, and resulting matching criterion $f(\mathbf{u})$ presented for registrations using sub-region size parameters 1^3 , 8^3 , 16^3 , and 32^3 . Note that the Y-axis in the time plot is shown in logarithmic scale.

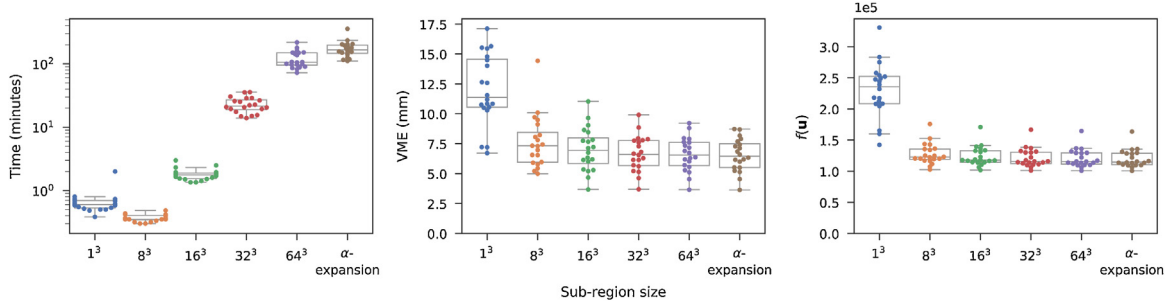


Fig. 4. Total computation time, VME, and resulting matching criterion $f(\mathbf{u})$ for registrations using sub-region size parameters 1^3 , 8^3 , 16^3 , 32^3 , 64^3 , and registrations with direct α -expansion, with image resolution halved along each dimension. Note that the Y-axis in the time plot is shown in logarithmic scale.

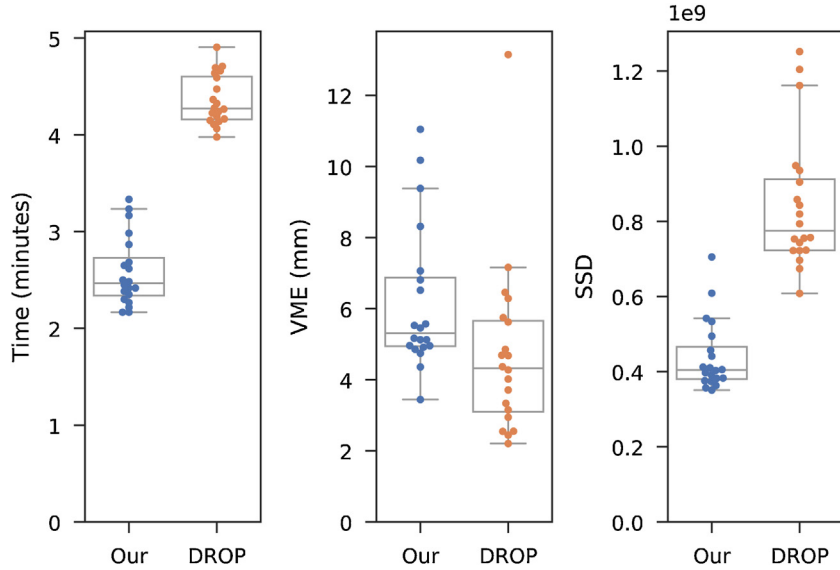


Fig. 5. Total computation time, VME, and resulting SSD ($D(\mathbf{u})$) for registrations using our proposed method and the DROP registration software.

7. Conclusion

Graph-cut based optimization with α -expansion has previously been shown to be a powerful tool for deformable image registration. The high computational cost of this approach, however, has limited the utility of this approach for registration of large volume images. The high computational cost was illustrated in our experiments, where the registration time of a single pair of images by direct application of α -expansion exceeded five days. We have presented an approach for fast discrete optimization for deformable registration of volume images. By dividing the image into subregions and restricting the α -expansion moves to a single sub-region at a time, we achieve a large reduction in computation time – from days to minutes – with only a small penalty in terms of solution optimality.

The division into subregions also facilitates additional computational optimizations – parallelization and early termination – that both reduce the computation time substantially.

We have demonstrated the utility of the proposed method in a case study on registration of whole-body fat-water MRI images, and conclude that the reduction in computation time facilitated by the proposed method makes graph-cut based optimization a viable option for registration of large volume images.

Authors' contribution

Simon Ekström and Filip Malmberg: conceptualization, methodology, software, validation, writing – original draft, writing – review & editing, formal analysis. Håkan Ahlström: resources, project

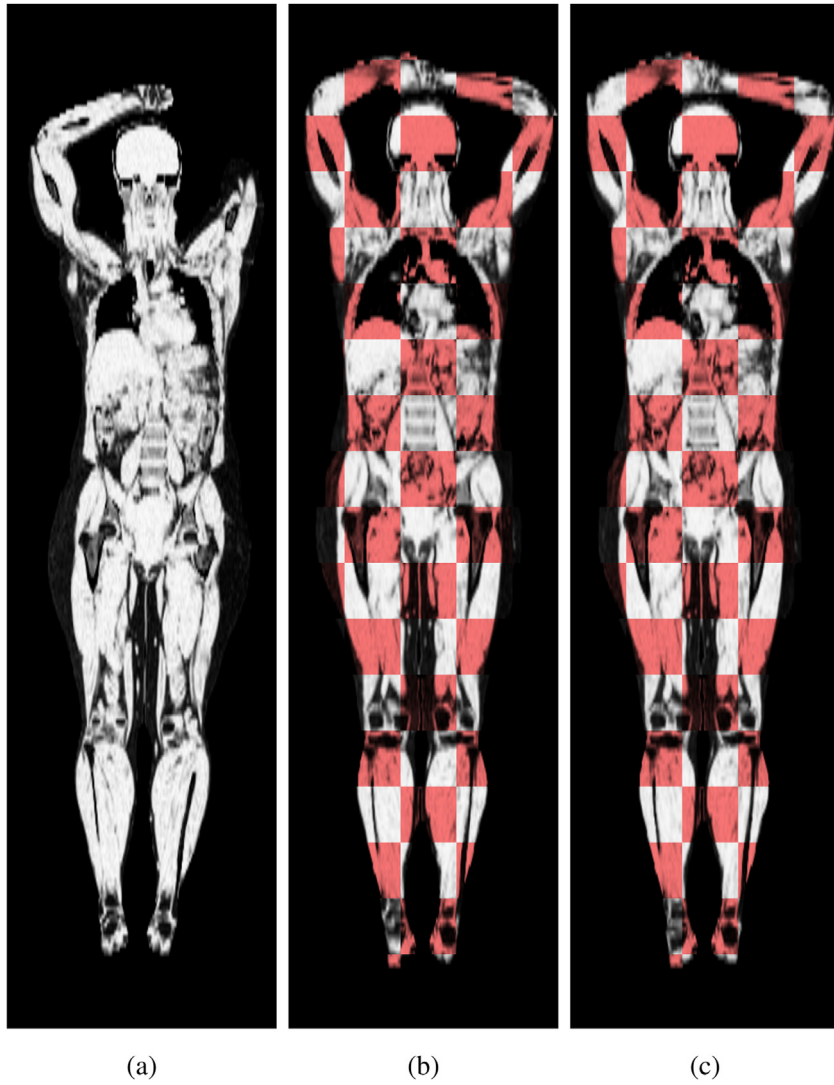


Fig. 6. Coronal slices from the water percent images from one of the subjects registered in the experiment. Presented are the unregistered water percent image (a), and checkerboards showing the registered water images (white) and the reference images (red) using DROP registration (b) and our proposed method (c), respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

administration, funding acquisition, supervision, writing – review & editing. Joel Kullberg: conceptualization, resources, supervision, writing – review & editing. Robin Strand: conceptualization, methodology, writing – review & editing, formal analysis.

Acknowledgements

Funding was received from Swedish Research Council (2016-01040). The authors wish to thank Prof. Nikos Paragios for his guidance on using the DROP registration software.

Conflict of interest: None declared.

Appendix A. Proof of submodularity

A globally optimal solution to the binary labeling problem given in Eq. (6) can be found by solving a maximum flow/minimum cut problem, provided that all binary terms are *submodular* (Kolmogorov and Zabini, 2004). In this appendix, we prove that the binary terms in given by Eqs. (9)–(12) are submodular for any \mathbf{u} and δ .

A binary term $\phi_{v,w}(L(v), L(w))$ is *submodular* if it satisfies the inequality

$$\phi_{v,w}(0, 0) + \phi_{v,w}(1, 1) \leq \phi_{v,w}(0, 1) + \phi_{v,w}(1, 0). \quad (\text{A.1})$$

Here, we have

$$\phi_{v,w}(0, 0) = \alpha \|\mathbf{u}(v) - \mathbf{u}(w)\|^2, \quad (\text{A.2})$$

$$\phi_{v,w}(1, 1) = \alpha \|(\mathbf{u}(v) + \delta) - (\mathbf{u}(w) + \delta)\|^2, \quad (\text{A.3})$$

$$\phi_{v,w}(1, 0) = \alpha \|(\mathbf{u}(v) + \delta) - \mathbf{u}(w)\|^2, \quad (\text{A.4})$$

$$\phi_{v,w}(0, 1) = \alpha \|\mathbf{u}(v) - (\mathbf{u}(w) + \delta)\|^2. \quad (\text{A.5})$$

Noting that $\phi_{v,w}(0, 0) = \phi_{v,w}(1, 1)$, the left hand side of Eq. (A.1) can be rewritten as

$$\phi_{v,w}(0, 0) + \phi_{v,w}(1, 1) = 2\alpha \|\mathbf{u}(v) - \mathbf{u}(w)\|^2. \quad (\text{A.6})$$

The right hand side of Eq. (A.1) can be rewritten as

$$\phi_{v,w}(0, 1) + \phi_{v,w}(1, 0) = 2\alpha \|\mathbf{u}(v) - \mathbf{u}(w)\|^2 + 2\alpha \|\delta\|^2. \quad (\text{A.7})$$

Since $2\alpha \|\delta\|^2 \geq 0$, we have

$$\phi_{v,w}(0, 0) + \phi_{v,w}(1, 1) \leq \phi_{v,w}(0, 1) + \phi_{v,w}(1, 0) \quad (\text{A.8})$$

This completes the proof.

References

- Berglund, J., Johansson, L., Ahlström, H., Kullberg, J., 2010. Three-point Dixon method enables whole-body water and fat imaging of obese subjects. *Magn. Reson. Med.* 63, 1659–1668.
- Boykov, Y., Kolmogorov, V., 2004. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* 26, 1124–1137.
- Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* 23, 1222–1239.
- Christensen, G.E., Johnson, H.J., 2001. Consistent image registration. *IEEE Trans. Med. Imaging* 20, 568–582.
- Glocker, B., Komodakis, N., Tziritis, G., Navab, N., Paragios, N., 2008. Dense image registration through MRFs and efficient linear programming. *Med. Image Anal.* 12, 731–741.
- Glocker, B., Sotiras, A., Komodakis, N., Paragios, N., 2011. Deformable medical image registration: setting the state of the art with discrete methods. *Annu. Rev. Biomed. Eng.* 13, 219–244.
- Heinrich, M.P., Jenkinson, M., Brady, M., Schnabel, J.A., 2013. MRF-based deformable registration and ventilation estimation of lung CT. *IEEE Trans. Med. Imaging* 32, 1239–1248.
- Kolmogorov, V., Zabih, R., 2004. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.* 26, 147–159.
- Kullberg, J., Johansson, L., Ahlström, H., Courivaud, F., Koken, P., Eggers, H., Börnert, P., 2009. Automated assessment of whole-body adipose tissue depots from continuously moving bed MRI: a feasibility study. *J. Magn. Reson. Imaging* 30, 185–193.
- Maintz, J.A., Viergever, M.A., 1998. A survey of medical image registration. *Med. Image Anal.* 2, 1–36.
- So, R.W., Tang, T.W., Chung, A.C., 2011. Non-rigid image registration of brain magnetic resonance images using graph-cuts. *Pattern Recogn.* 44, 2450–2467.
- Sotiras, A., Davatzikos, C., Paragios, N., 2013. Deformable medical image registration: a survey. *IEEE Trans. Med. Imaging* 32, 1153–1190.
- Strand, R., Malmberg, F., Johansson, L., Lind, L., Sundbom, M., Ahlström, H., Kullberg, J., 2017. A concept for holistic whole body MRI data analysis, imiomics. *PLOS ONE* 12, e0169966.
- Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C., 2006. A comparative study of energy minimization methods for Markov random fields. *European Conference on Computer Vision*, 16–29.
- Szumil, A., Papiez, B.W., Bates, R., Hallack, A., Schnabel, J.A., Grau, V., 2016. Graph cuts-based registration revisited: A novel approach for lung image registration using supervoxels and image-guided filtering. *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 152–159.
- Szumil, A., Hallack, A., Grau, V., Schnabel, J.A., 2017. Supervoxels for graph cuts-based deformable image registration using guided image filtering. *J. Electron. Imaging* 26, 61607.
- Tang, T.W., Chung, A.C., 2007. Non-rigid image registration using graph-cuts. *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2007*, 916–924.