# FAST RANDOM WALKER IMAGE REGISTRATION USING PRECOMPUTATION

Shawn Andrews, Lisa Tang and Ghassan Hamarneh

Medical Image Analysis Lab, School of Computing Science, Simon Fraser University, Canada {sda56, lisat, hamarneh}@sfu.ca

# ABSTRACT

In this paper, we introduce an extension to the random walker image registration method designed to increase the speed at which a registration is performed. Our method involves precomputing data from one of the images being registered while anticipating the acquisition of the second image, and then using this precomputed data to approximate the final transformation once the second image becomes available. The precomputation scheme incorporates a parameter controlling the trade-off between registration speed and accuracy that can be tuned when the registration is being performed. Our results show that with precomputation, random walker image registration runs 3 to 10 times faster on volumetric images with only 3% to 20% loss in registration accuracy.

Index Terms— Discrete Registration, Precomputation

# **1. INTRODUCTION**

Deformable image registration (DIR) is a fundamental task in medical image analysis, and a key step in many clinical applications [1]. A few of the prominent clinical uses of image registration include fusing information from multiple imaging modalities, tracking the progression of a medical condition by registering images taken at different times, and constructing anatomical atlases for use in segmentation [2].

Recently, several registration techniques have moved towards performing registration in the discrete domain [3–6], where voxels are assigned labels from a predefined set of displacement vectors. The discrete approach allows the use of powerful discrete optimization techniques that avoid problems often encountered in continuous registration formulations, such as becoming stuck in local minima. In this paper we focus on the random walker image registration (RWIR) algorithm proposed by Cobzas and Sen [4], which uses a Gaussian MRF energy that can be globally minimized to provide competitive results for dense registration [7].

One trade-off of discrete techniques is that the predefined set of displacement vectors needs to be quite large, particularly in 3D tasks, in order to achieve adequate spatial resolution, leading to high computational costs. The requirement for a large label set has been mitigated by iterative approaches designed to incrementally update an initially sparse label set [4, 6, 7], though such iterative approaches often sacrifice global optimality. Alternatively, a sparser, image-adaptive discretization of the image domain can be used [8, 9] instead of the standard uniform voxel-based discretization, so fewer spatial locations need to be solved for.

In this paper, we propose a technique for increasing the efficiency of RWIR that is complimentary to the methods described above. Our technique is based on precomputing data using only one of the images, resulting in a significant speedup to the registration with a minimal loss in accuracy. This precomputation technique is motivated by the fact that one of the two images is often available before the second, e.g. when registering images taken of the same patient months apart or when registering a new image to an existing atlas. Our precomputation technique has a parameter that controls the trade-off between speed and accuracy, allowing very fast registrations to accommodate time-sensitive tasks. Our work is based on similar techniques used to increase the efficiency of random walker segmentation tasks [10, 11], which have proven to be successful. While some registration parameters must be chosen before precomputation, several important parameters (e.g. image similarity, regularization strength) can be set afterwards, allowing multiple registrations with different parameter settings to be evaluated quickly.

The remainder of the paper is organized as follows: in Sec. 2.1 we review the RWIR method. In Sec. 2.2 we introduce our precomputation technique. In Sec. 2.3 we discuss adjusting registration parameters after precomputation. In Sec. 3 we present results illustrating the increase in efficiency our method provides.

# 2. FAST RANDOM WALKER FOR IMAGE REGISTRATION

# 2.1. Random Walker Registration Review

To register two images,  $J_1, J_2 : \Omega \to \mathbb{R}$ , where  $\Omega \subset \mathbb{R}^D$  and D is the dimensionality of the images, we find a spatial transformation  $T : \Omega \to \Omega$  such that  $J_1 \circ T$  (that is,  $J_1$  composed with T) is aligned to  $J_2$ . Not all transformations are equally likely, however, so some regularization is imposed on T to ensure a feasible transformation.

In RWIR, we work in the discrete image domain, denoted

 $\hat{\Omega}$ , with  $|\hat{\Omega}| = n$  pixels, and we represent transformations using a discrete set of K displacements,  $\{\mathbf{v}^1, \dots, \mathbf{v}^K\}$ , where  $\mathbf{v}^k \in \mathbb{R}^D$ . Registration is formulated as a labeling problem, where each pixel  $x \in \hat{\Omega}$  is assigned a fuzzy labeling of the displacement vectors,  $\mathbf{u}_x = [u_x^1, \dots, u_x^K] \in \mathbb{P}_K$ .  $\mathbb{P}_K \subset \mathbb{R}^K$ is the unit simplex of positive unit-sum vectors.

RWIR consists of two steps. The first step is to construct a prior term  $\mathbf{p}^k : \hat{\Omega} \to \mathbb{R}$  for each label  $k \in \{1, \dots, K\}$  based on an image similarity function:

$$\mathbf{p}^k = \frac{1}{Z_x} \mathcal{D}(x, \mathbf{v}_k, J_1, J_2) .$$
 (1)

Here,  $Z_x$  is a normalization function ensuring the prior labels sum to one at each pixel.  $\mathcal{D}$  can include various image features or user input, and is largely application dependent.

The second step is to construct a weighted graph based on the moving image,  $J_1$ , and use it to spatially regularize the displacement labels. The image graph consists of a node for each pixel and weighted edges between neighboring pixels, with weights chosen inversely proportional to the difference in intensities between the incident pixels. Specifically, for an edge with incident pixels x and y, we assign edge weight  $w_{x,y} = \exp(-\beta |J_1(x) - J_1(y)|)$ , where  $\beta$  is a scalar parameter. Defining W as the  $n \times n$  sparse matrix with entries corresponding to these edge weights and D as the  $n \times n$  diagonal matrix with entries corresponding to the row sums of W, then L = D - W is known as the combinatorial Laplacian of the image graph.

The fuzzy labels  $\mathbf{u}^k : \hat{\Omega} \to \mathbb{R}$  are calculated by minimizing, for each  $k \in \{1, \dots, K\}$ , the Gaussian MRF energy

$$E_{RW}(\mathbf{u}^k) = \mathbf{u}^{k\top} L \mathbf{u}^k + \left(\mathbf{u}^k - \mathbf{p}^k\right)^\top \Gamma \left(\mathbf{u}^k - \mathbf{p}^k\right), \quad (2)$$

where  $\Gamma$  is an  $n \times n$  diagonal matrix of spatially dependent regularization weights.  $\Gamma$  is useful for increasing regularization strength in homogeneous regions, where the prior labels  $\mathbf{p}^k$  are not very informative.

Denoting  $U = [\mathbf{u}^1, \dots, \mathbf{u}^K]$  and  $P = [\mathbf{p}^1, \dots, \mathbf{p}^K]$  as  $n \times K$  matrices, the minimum of  $E_{RW}$  is calculated by solving for U in the system of linear equations:

$$(L+\Gamma)U = \Gamma P . \tag{3}$$

Solving (3) gives a fuzzy labeling for the displacement vectors at each pixel. Since the rows of P sum to one, so do the rows of U, and thus can be interpreted as fuzzy labels.

Once the fuzzy labels have been calculated, they are used to construct a dense displacement field over the image; e.g. by taking the displacement corresponding to the maximum probability [4]. This results in a globally optimal dense displacement field that can represent a wide range of transformations.

### 2.2. Random Walker Precomputation

A major drawback of RWIR is that it is computationally expensive. The main bottleneck is solving the system of equations in (3), which is time consuming since it must be solved for each of the K displacement labels. To achieve adequate resolution, particularly for 3D registration, K must be very large. As detailed above, several techniques have been proposed to increase the efficiency of RWIR by iteratively running the algorithm, starting with a sparse displacement label set and updating the label set after each iteration. These approaches have proven effective, but the iterative approach sacrifices global optimality, and there is no guarantee that the optimal displacements will be added to the label set.

In this section, we propose an alternative, complimentary approach to improve RWIR efficiency. We take advantage of the fact that in medical applications, one of the images being registered,  $J_1$ , is often available for analysis before the second image,  $J_2$  (e.g. registration to an existing atlas). By performing precomputation on  $J_1$ , we can greatly increase the speed at which (3) is solved. The resulting algorithm will be referred to as fastRWIR.

Rearranging (3) gives

$$U = (L + \Gamma)^{-1} \Gamma P .$$
(4)

In (4), L and  $\Gamma$  are derived from  $J_1$ , and only the similarity term P involves  $J_2$ , so in theory we could precompute  $(L + \Gamma)^{-1}\Gamma$  and then calculate U from P using a single matrix multiplication. Unfortunately, this technique is not effective because L and  $\Gamma$  are sparse matrices with O(n) non-zero entries, but the inverse  $(L + \Gamma)^{-1}$  is not sparse, and working with a matrix with  $O(n^2)$  non-zero entries is infeasible in terms of computational time and memory.

We address this problem by utilizing a technique similar to the one proposed for random walker image segmentation [10,11]. Given  $J_1$ , we calculate L and  $\Gamma$  and then perform an eigenvector decomposition of  $(L + \Gamma)$  to obtain the smallest  $k \ll n$  eigenvalues,  $\lambda_1 \leq \cdots \leq \lambda_k$ , and their corresponding eigenvectors,  $Q = [\mathbf{q}_1, \dots, \mathbf{q}_k]$ . This gives:

$$(L+\Gamma)^{-1} \approx Q\Lambda^{-1}Q^{\top} \tag{5}$$

$$U \approx Q \Lambda^{-1} Q^{\top} \Gamma P .$$
 (6)

Note that all of the  $\lambda$ 's are real (since L and  $\Gamma$  are symmetric) and positive (since the eigenvalues of a Laplacian are nonnegative and the eigenvalues of the positive diagonal matrix  $\Gamma$  are positive), so  $\Lambda$  is always invertible. Since U is approximated using the inverse of  $\Lambda$ , the smallest eigenvalues provide the best approximation. k controls the trade-off between accuracy and speed; for small enough k (6) is much more efficient to compute than (3).

#### 2.3. Parameter Tuning

Properly chosen parameters are key to the success of many registration algorithms. The speedup provided by fastRWIR allows more registrations to be performed with different parameter settings in a given amount of time, increasing the robustness of the results to parameter tuning. However,



Fig. 1. The average distance between the displacements generated by RWIR and fastRWIR for different values of k and the average times taken for both RWIR and fastRWIR. For reference, the average distance between the RWIR and the ground truth displacements was  $2.24 \pm 0.05$  mm.

fastRWIR requires certain parameter values be set at precomputation time, so it is important to note what parameters can be updated without affecting the precomputed matrices.

Clearly, the similarity term P can be updated online, allowing for multiple similarity functions to be tested quickly. Another important parameter that can be set after precomputation is the strength of the regularization. If the matrix  $\Gamma$  is updated to  $\hat{\Gamma} = \Gamma + \alpha I$ , for a scalar  $\alpha$ , (4) becomes

$$U = (L + \Gamma + \alpha I)^{-1} (\Gamma + \alpha I) P$$
(7)

$$\approx Q(\Lambda + \alpha I)^{-1}Q^{\top}(\Gamma + \alpha I)P$$
. (8)

Here, the only inverse is of a  $k \times k$  diagonal matrix, which is a very efficient operation. fastRWIR can be performed quickly with various regularization strengths  $\alpha$ , ensuring that the final transformation is not under-regularized (e.g. if the transformation is irregular) or over-regularized (e.g. if the images are too dissimilar).

# 3. RESULTS

In this section, we provide results comparing our method, fastRWIR, to the original RWIR algorithm, demonstrating the considerable speedup it provides with minimal loss in accuracy. Our experiments are performed on 40 T1-MR volumetric brain images from the LONI dataset [12] using unoptimized MATLAB code run on a machine with 2 Quad Core Intel Xeon 2.33 GHz CPUs. More efficient implementations would provide faster run times for both algorithms, but we are focused on the speedup provided by fastRWIR over RWIR.



Fig. 2. The DSCs of the 56 segmented brain regions change after registration, averaged across all pairs of images. RWIR took  $2716 \pm 1783$  secs. to perform the registrations, while fastRWIR took only  $830 \pm 478$  secs.

#### 3.1. Synthetic Misalignment

In our first experiment, we applied known warps to the images and then attempted to recover these warps using RWIR and fastRWIR. We generated 10 warps for each image by randomly displacing B-spline control points, spaced 18 mm apart, where the displacements were sampled uniformly from vectors up to 6 mm in magnitude.

We registered each image and its warped version using RWIR and fastRWIR with  $k \in \{100, 200, \dots, 1000\}$  precomputed eigenvectors. In Fig. 1, we evaluated the accuracy of fastRWIR by calculating the mean distance between the displacements generated at each voxel by RWIR and by fastRWIR. We compared these distances to the mean distance between the RWIR displacements and the ground truth displacements (i.e. the registration error) of  $2.24 \pm 0.05$  mm. For k = 1000, fastRWIR generated similar results to RWIR, with an average distance between displacements of  $0.07 \pm 0.06$  mm, only 3% of the registration error. For k = 300, the average distance between fastRWIR and RWIR is still only  $0.46 \pm 0.10$ , about one fifth of the error.

Fig. 1 also compares the time taken by RWIR and fastRWIR for different values of k. For k = 1000, fast-RWIR achieved a roughly 3 times speedup, and for k = 300 it achieved a roughly 10 times speedup. As long as a sufficient number of eigenvectors are precomputed, the exact value of k can be chosen at run time to meet any speed requirements.

#### 3.2. Real Data

In our next experiment, we performed registration between each pair in the 40 images (randomly choosing which one to use as the moving image) using both RWIR and fastRWIR with k = 1000. Each image has 56 anatomical brain regions segmented, and we evaluated the accuracy of the registrations



Fig. 3. The increase in DSC of each of the 56 segmented brain regions when registering with RWIR instead of fastRWIR with k = 1000. Some outliers above and below the current bounds have been excluded for readability.

by comparing the Dice similarity coefficients (DSCs) between each of these regions in the pair of images before and after the registration is performed.

Fig. 2 shows a comparison of the DSCs for each of the brain regions before registration and after registration using RWIR and *fast*RWIR. Both registration methods typically increase the DSCs, indicating they improve the correspondence between anatomical regions significantly. As expected, *fast*RWIR performs slightly worse than RWIR on average, though the improvement in DSC when using RWIR instead of *fast*RWIR is small (0.013±0.005) compared to the improvement in DSC before and after registration (0.074 ± 0.041). Thus, the approximations introduced by *fast*RWIR reduce the increase in DSC from registration by less than a fifth. Despite only a small drop in accuracy, *fast*RWIR ran more than 3 times faster than RWIR (830 ± 478 seconds for *fast*RWIR vs. 2716 ± 1783 seconds for RWIR).

Fig. 3 provides a more detailed comparison of the algorithms, showing the increase in DSC when using RWIR instead of fastRWIR for each of the image pairs. For most regions, the difference in DSC is consistently close to 0, though a few regions appear to give fastRWIR trouble. This is likely due to the eigenvector approximation of the Laplacian of the image graph being poor in the neighborhood of these regions.

### 4. CONCLUSION

We have introduced a method for increasing the speed of the RWIR algorithm using a precomputation technique. We have shown that our fastRWIR technique greatly increases the efficiency of RWIR without a significant drop in registration quality. fastRWIR provides a powerful tool for aligning large volumetric images quickly and accurately.

Our future work will involve combining our precomputa-

tion method with existing improvements to RWIR (e.g. iteratively updating a sparse displacement label set) and with standard methods for increasing performance (such as GPU computing) to achieve a real-time, accurate registration method.

Acknowledgements. The authors are grateful to NSERC for partially funding this work.

# 5. REFERENCES

- A. Sotiras, C. Davatzikos, and N. Paragios, "Deformable medical image registration: A survey," *IEEE TMI*, vol. 32, no. 7, pp. 1150–1188, 2013.
- [2] Q. Wang et al., "Groupwise registration based on hierarchical image clustering and atlas synthesis," *Human brain mapping*, vol. 31, no. 8, pp. 1128–1140, 2010.
- [3] B. Glocker et al., "Dense image registration through MRFs and efficient linear programming," *MedIA*, vol. 12, no. 6, pp. 731–741, 2008.
- [4] D. Cobzas and A. Sen, "Random walks for deformable image registration," in *MICCAI*, 2011, pp. 557–565.
- [5] L. Tang, G. Hamarneh, and R. Abugharbieh, "Reliability-driven, spatially-adaptive regularization for deformable registration," in *Biomedical Image Registration*, 2010, pp. 173–185.
- [6] M. Heinrich, M. Jenkinson, M. Brady, and J. Schnabel, "Globally optimal deformable registration on a minimum spanning tree using dense displacement sampling," in *MICCAI*, 2012, pp. 115–122.
- [7] L. Tang and G. Hamarneh, "Random walks with efficient search and contextually adapted image similarity for deformable registration," in *MICCAI*, 2013, pp. 43– 50.
- [8] Karteek Popuri, Dana Cobzas, and Martin Jägersand, "A variational formulation for discrete registration," in *MICCAI*, 2013, pp. 187–194.
- [9] L. Tang and G. Hamarneh, "Random walk image registration with cost aggregation," in *ISBI*, 2014.
- [10] L. Grady and A. Sinop, "Fast approximate random walker segmentation using eigenvector precomputation," in *CVPR*, 2008, pp. 1–8.
- [11] S. Andrews, G. Hamarneh, and A. Saad, "Fast random walker with priors using precomputation for interactive medical image segmentation," in *MICCAI*, 2010, pp. 9–16.
- [12] A. Klein et al., "Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration," *Neuroimage*, vol. 46, no. 3, pp. 786–802, 2009.