HAL
open science

Discrete rigid transformation graph search for 2D image registration

Phuc Ngo, Akihiro Sugimoto, Yukiko Kenmochi, Nicolas Passat, Hugues Talbot

To cite this version:

Phuc Ngo, Akihiro Sugimoto, Yukiko Kenmochi, Nicolas Passat, Hugues Talbot. Discrete rigid transformation graph search for 2D image registration. PSIVT Workshops, 2013, Guanajuato, Mexico. pp.228-239, 10.1007/978-3-642-53926-8_21 . hal-00866285

HAL Id: hal-00866285

https://hal.science/hal-00866285v1

Submitted on 30 Sep 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Discrete rigid transformation graph search for 2D image registration

P. Ngo[1,2], A. Sugimoto[1], Y. Kenmochi[2], N. Passat[3], and H. Talbot[2]

[1] National Institute of Informatics, Japan
[2] Université Paris-Est, LIGM, France
[3] Université de Reims Champagne-Ardenne, CReSTIC France

Abstract. Rigid image registration is an essential image processing task, with a large body of applications. This problem is usually formulated in the continuous domain, most often in an optimization framework. This strategy is suitable whenever the estimated transformation is associated with interpolation. In the case of purely discrete applications, *e.g.*, for template-based segmentation or classification, it is however preferable to avoid digitizing the result again after transformation. In this article, we deal with this point of view in the 2D case. Based on a fully discrete framework, we explicitly explore the parameter space of rigid transformations. This exploration leads to a local search scheme that can be involved in combinatorial optimization strategies.

Keywords: Rigid registration, digital image, combinatorial optimisation on graph, parameter space subdivision.

1 Introduction

Image registration is a crucial task in various fields of computer vision and image analysis (*e.g.*, medical imaging [1], remote sensing [2], object tracking [3]). Essentially, such registration problems consist of estimating a deformation field or a geometric transformation matrix so that a source image can be transformed into a target image [4].

This widely studied problem is often formulated as an optimisation problem, for instance via minimizing a distance measure between the two images. The search space for this procedure is the parameter space of the transformation, which is most often continuous, *e.g.*$[0, 2\pi[\times\mathbb{R}^2$ for 2D rigid transformations.

This approach is suitable in cases where the resulting transformed image is further interpolated, *e.g.*when it is meant to be visualized. However, in specific applications, *e.g.*template-based segmentation or classification [5, 6], it is required to guarantee to preserve some properties under transformations, such as topological properties. In this context, a continuous transformation approach imposes a re-digitization of the obtained result that may bias the results. In such cases, it may be preferable to directly compute the discrete transformation.

In this article, we investigate solutions to this discrete problem in the case of 2D rigid registration. Recently, a fully discrete approach to rigid transformations

on $\mathbb{Z}^2$ was proposed in [7]. In this work, the whole parameter space of rigid transformations is modeled by a combinatorial structure, namely a *discrete rigid transformation (DRT) graph*. This structure models a neighbouring relationship between transformations. As well, using this structure, the actual deformation of transformed images can be handled pixel by pixel. Using DRT graphs, we can explicitly explore the parameter space of all the rigid transformations defined on any finite subset of $\mathbb{Z}^2$ of size $N \times N$. Exhaustive search on DRT graphs can be handled with a polynomial spatial complexity of $\mathcal{O}(N^9)$, which limits practical use. In this article, our aim is to reduce the complexity of some operations in this structure. More precisely, we focus on calculating the exact neighbourhood structure in linear time with respect to the image size. This allows us to perform the local search on DRT graph from any starting point in this parameter space using a gradient descent, which leads to a local optimum, and paving the way for integration of this approach into any combinatorial optimization process.

This article is organized as follows. Sec. 2 explains how the DRT graph models the whole parameter space of 2D rigid transformations of a digital image. Sec. 3 formulates the optimisation problem of image rigid registration using neighbouring relations defined on DRT graph. Sec. 4 describes a discrete exploration procedure within this graph to find locally optimal solutions to the registration problem. Sec. 5 describes experimental results, while Sec. 6 concludes the article.

2 Parameter space of rigid transformation

2.1 Rigid transformations on $\mathbb{Z}^2$

In the continuous domain, a rigid transformation is a bijection $\mathcal{T} : \mathbb{R}^2 \to \mathbb{R}^2$, defined for any $\boldsymbol{x} = (x, y) \in \mathbb{R}^2$, as

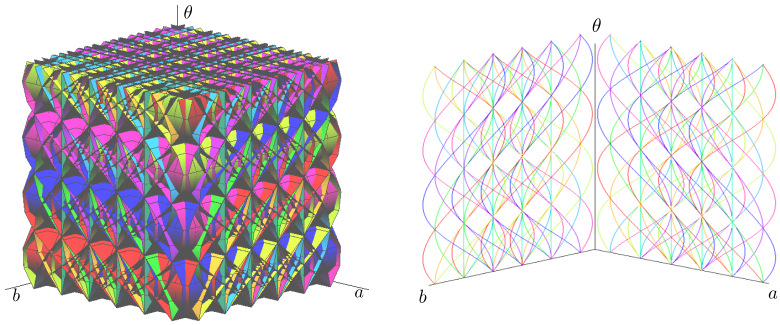
$$\mathcal{T}(\boldsymbol{x}) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix}, \tag{1}$$

with $a, b \in \mathbb{R}$ and $\theta \in [0, 2\pi[$. It will be sometimes noted $\mathcal{T}_{ab\theta}$. When applied on digital images defined on $\mathbb{Z}^2$, rigid transformations require digitization as post-processing step. Practically, this can be handled by defining $T = D \circ \mathcal{T}$, where $D : \mathbb{R}^2 \to \mathbb{Z}^2$ is a standard rounding function. This function $T : \mathbb{Z}^2 \to \mathbb{Z}^2$ is then explicitly defined, for any $\boldsymbol{p} = (p, q) \in \mathbb{Z}^2$, as

$$T(\boldsymbol{p}) = D \circ \mathcal{T}(\boldsymbol{p}) = \begin{pmatrix} [p\cos\theta - q\sin\theta + a] \\ [p\sin\theta + q\cos\theta + b] \end{pmatrix}. \tag{2}$$

2.2 Subdivision of the parameter space

The use of the rounding function D to define T implies that rigid transformations on $\mathbb{Z}^2$ are not continuous with respect to the parameters a, b and θ. More precisely, discontinuities are located at half-integer pixel coordinates, modeling pixel boundaries. This leads to the definition of *discrete rigid transformations*



(a) Tipping surfaces. (b) Tipping curves.

Fig. 1. (a) Tipping surfaces in the parameter space (a, b, θ). (b) Tipping curves, in the 2D planes (a, θ) and (b, θ).

(DRTs), each of which corresponds to the class of all transformations $\mathcal{T}_{ab\theta}$ such that $(a, b, \theta) \mapsto T = D \circ \mathcal{T}_{ab\theta}$ is piecewise constant.

The parameter space $\mathbb{R}^2 \times [0, 2\pi[$ of (a, b, θ) is correspondingly subdivided into 3D cells. Each DRT corresponds to a cell whose boundary is 2D surfaces that model discontinuities. The transformations leading to such discontinuities are those that map an integer point onto a half-integer coordinate point. They can be expressed, for any $\boldsymbol{p} = (p, q) \in \mathbb{Z}^2$ mapped onto either a "vertical" point $\boldsymbol{p}_\Phi = (k + \frac{1}{2}, \lambda)$ or a "horizontal" one $\boldsymbol{p}_\Psi = (\lambda, l + \frac{1}{2})$ (with $k, l \in \mathbb{Z}$ and $\lambda \in \mathbb{R}$):

$$\left| \begin{array}{ll} \Phi_{pqk} : \mathbb{R}^2 & \longrightarrow \mathbb{R} \\ \qquad (b, \theta) & \longmapsto a = \phi_{pqk}(\theta) = k + \frac{1}{2} + q \sin\theta - p \cos\theta, \end{array} \right. \tag{3}$$

$$\left| \begin{array}{ll} \Psi_{pql} : \mathbb{R}^2 & \longrightarrow \mathbb{R} \\ \qquad (a, \theta) & \longmapsto b = \psi_{pql}(\theta) = l + \frac{1}{2} - p \sin\theta - q \cos\theta. \end{array} \right. \tag{4}$$

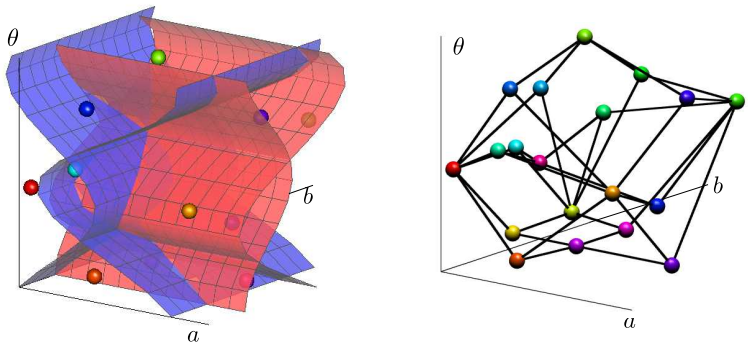
The surfaces Φ_{pqk} (resp. Ψ_{pql}) in the parameter space (a, b, θ) are called *tipping surfaces*. Their cross-sections ϕ_{pqk} (resp. ψ_{pql}) on the 2D plane (a, θ) (resp. (b, θ)) are called *tipping curves*. For an image of size $N \times N$, Φ_{pqk} and Ψ_{pql} (resp. ϕ_{pqk} and ψ_{pql}) verify $p, q \in [\![0, N-1]\!]$ and $k, l \in [\![0, N]\!]$. Tipping surfaces/curves are illustrated in Fig. 1.

2.3 Combinatorial structure of the subdivided parameter space

Each DRT is modeled by a 3D cell, while each discontinuity is modeled by a 2D tipping surface piece. By mapping each cell onto a vertex, and each tipping surface piece onto an edge, we can dually model the subdivided parameter space as a graph, called a DRT graph, as illustrated in Fig. 2.

Definition 1 ([7]) *A* DRT graph $G = (V, E)$ *is defined such that:*

- *each vertex $v \in V$ models a DRT; and*
- *each labelled edge $e = (v, w, f) \in E$, where f is either Φ_{pqk} or Ψ_{pql}, connects two vertices $v, w \in V$ sharing a tipping surface f as the boundary.*



(a) Parameter space. (b) DRT graph.

Fig. 2. (a) Subdivision of the parameter space into 3D cells and 2D surfaces. (b) The associated DRT graph.

In [8], it is proved that the 3D cell associated to a DRT in the parameter space of (a, b, θ) is convex with respect to both the a- and b-axis, and that its boundary consists of an upper and a lower sets of tipping surfaces. These properties are used in Sec. 4.2 and Appendix A to calculate a representative of rigid transformations associated to a DRT.

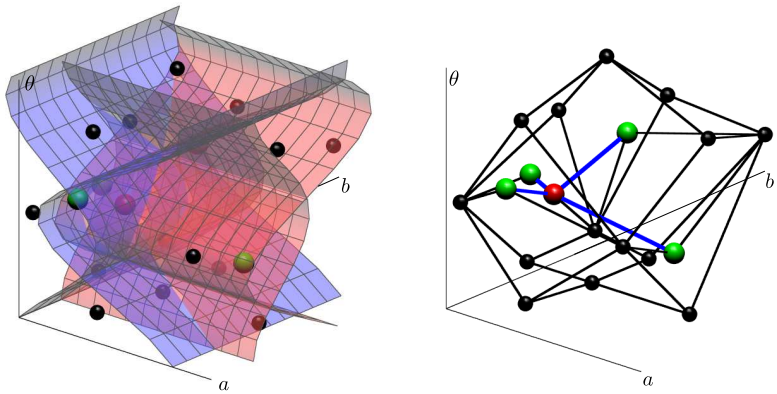
In the DRT graph of a given image I of size $N \times N$, each vertex is associated to a unique transformed image, obtained by any rigid transformation whose parameters are contained in the 3D cell corresponding to the vertex. Moreover, the existence of an edge between two vertices in the DRT graph indicates a *neighbouring* relation between the DRTs, such that the two associated transformed images differ by (at most) one pixel over the N^2 pixels of I; the edge label f contains the information of such a different pixel. This property allows us to use the DRT graph to produce the transformed images via successive elementary (*i.e.*, one-pixel) modifications. Such elementary modification is not easy in the continuous framework due to the uncountably infinite nature of transformations defined on $\mathbb{R}^2$.

3 Registration as a combinatorial optimisation problem

Let us consider two images A and B of same size $N \times N$. The problem of image registration consists of finding a transformation T_v^* within the set $\mathbb{T} = \{T_v \mid v \in V\}$ of all the DRTs, that minimizes a given distance d between the image A and the transformed image $T_v(B)$ of the image B by T_v, *i.e.*

$$T_v^* = \arg\min_{T_v \in \mathbb{T}} d(A, T_v(B)). \tag{5}$$

Several distances d can be used (*e.g.*, cross-correlation, mutual information, least square difference [4], or signed distance [9]). The choice of distance is not addressed in our study; we consider d as a function and focus on the algorithm for computing Eq. (5) by exploring the DRT graph.



(a) Parameter space. (b) DRT graph.

Fig. 3. Neighbours of a DRT. The considered DRT is depicted in red and its neighbours in green. (a) The DRT shares a surface piece with its neighbours in the parameter space. (b) They are adjacent, *i.e.*, they share an edge (in blue), in the DRT graph.

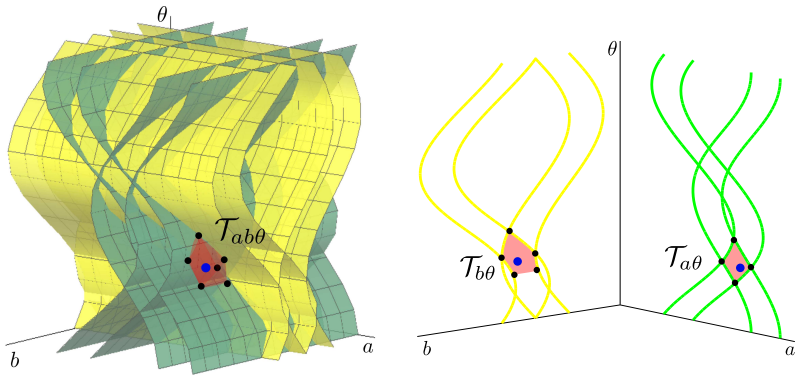
For given images A and B of finite size, the DRT graph $G = (V, E)$ associated to B is a finite structure that models all the existing DRTs for this image. Therefore, there exists $v \in V$ such that $d(A, T_v(B))$ is minimal, where $T_v(B)$ is the transformed image of B associated to the DRT T_v at vertex v. In other words, by considering a brute-force search, a solution (*i.e.*, a global optimum) can be found for Eq. (5). It is proved in [7] that the DRT graph G has a high-polynomial space complexity of $\mathcal{O}(N^9)$ for an image of size $N \times N$, and an exact algorithm is proposed to compute G in a linear time with respect to its size.

Proposition 2 *Exhaustive search on DRT graph costs $\mathcal{O}(N^9)$ in time and space complexities.*

Practically, this limits exploration of the whole structure to very small images. Nevertheless, it is possible to perform a local search of G in order to determine a *local* optimum. In the next section, we show that such exploration can, in particular, be carried out without requiring to build the whole graph G but only its useful parts, thus leading to a much lower algorithmic complexity.

4 Local search on discrete rigid transformation graph

In this section, we describe how to find locally optimal solutions for the image registration problem stated above. Basically, our approach consists of carrying out a gradient descent from an initial vertex in the DRT graph. Despite its apparent simplicity, this process requires to deal with a non-trivial issue, namely the – efficient – construction of a sufficient part of the DRT in the search area.



(a) DRT associated to $\mathcal{T}_{ab\theta}$. (b) Its projections.

Fig. 4. Example of rigid transformation $\mathcal{T}_{ab\theta}$ (blue dot) and its associated DRT (red volume) in the 3D parameter space (a, b, θ) (a), and its projection in the 2D planes (a, θ) and (b, θ) (b).

4.1 Neighbourhood construction in a DRT graph

From Sec. 2, we know that the parameter space of rigid transformations is subdivided into DRTs, and this subdivision is modeled by the combinatorial structure of a DRT graph $G = (V, E)$. We consider the notion of *neighbourhood* $N(v)$ of a DRT $v \in V$ by using its standard definition in graph theory. Two DRTs are neighbours if their respective vertices share an edge, *i.e.*, are adjacent. Equivalently, they share a surface piece in the dual parameter space (see Fig. 3).

As described in Sec. 2.2, when projecting the two families of tipping surfaces on the planes (a, θ) and (b, θ), respectively, we obtain two families of tipping curves ϕ and ψ. Thus, the boundary of a DRT by tipping surfaces in the 3D parameter space (a, b, θ) can be fully described from its two cross-sections in the planes (a, θ) and (b, θ) by the corresponding tipping curves. Using the a- and b-convex property, we propose Algorithm 1 to explicitly determine the bounding tipping curves, and thus tipping surfaces (see Fig. 4). From Definition 1, finding the neighbours of a given DRT is equivalent to finding the tipping curves segment of its boundary. For efficiency of method, Algorithm 1 requires a representative rigid transformation, denoted by (a, b, θ), corresponding to the dual region of an input vertex v, and returns its neighbours $N(v)$. Note that v is modeled by the set of *upper* (resp. *lower*) tipping curves, denoted by Φ^+ and Ψ^+ (resp. Φ^- and Ψ^-), indicating the correspondence between pixels of the image B and those of its transformed image $T_v(B)$. Algorithm 1 consists of finding among these tipping curves in Φ^+, Φ^-, Ψ^+ and Ψ^-, those that form the boundary of v.

Algorithm 1 uses a function F that returns the values θ of the intersections of two given tipping curves f and g such that:

$$F^*_{\theta'}(f, g) = \{\theta * \theta' \mid \theta \in \text{Intersection}(f, g)\},$$

where $* \in \{<, >\}$ and Intersection(f, g) computes the θ coordinates of intersections between f and g (the exact calculation of this function is given in [7]).

Algorithm 1: Finding the nearest neighbours of a given DRT.

Input: A DRT v, *i.e.*, Φ^-, Φ^+, Ψ^- and Ψ^+ with (a, b, θ) associated to v.
Output: The neighbours $N(v)$ of v.

1 $\mathcal{B} \leftarrow \emptyset$ `// Initialize the boundary set` $\mathcal{B}$ `of` v
2 $\theta' \leftarrow \theta$
3 **repeat** `// Sweep on the right side along the` θ`-axis from` θ'
4 $\quad \phi_l \leftarrow \arg\max_{\phi \in \Phi^-} \phi(\theta')$; $\phi_u \leftarrow \arg\min_{\phi \in \Phi^+} \phi(\theta')$ `// In the plane` (a, θ)
5 $\quad s_1 \leftarrow \min_{\phi \in \Phi^-} F^{<}_{\theta'}(\phi_l, \phi)$; $s_2 \leftarrow \min_{\phi \in \Phi^+} F^{<}_{\theta'}(\phi_u, \phi)$; $s_3 \leftarrow \min F^{<}_{\theta'}(\phi_l, \phi_u)$
6 $\quad \psi_l \leftarrow \arg\min_{\psi \in \Psi^-} \psi(\theta')$; $\psi_u \leftarrow \arg\min_{\psi \in \Psi^+} \psi(\theta')$ `// In the plane` (b, θ)
7 $\quad t_1 \leftarrow \min_{\psi \in \Psi^-} F^{<}_{\theta'}(\psi_l, \psi)$; $t_2 \leftarrow \min_{\psi \in \Psi^+} F^{<}_{\theta'}(\psi_u, \psi)$; $t_3 \leftarrow \min F^{<}_{\theta'}(\psi_l, \psi_u)$
8 $\quad \theta' \leftarrow \min\ (t_1, t_2, t_3, s_1, s_2, s_3)$ `// Find the next intersection`
9 $\quad \mathcal{B} \leftarrow \mathcal{B} \cup \{\phi_l, \phi_u, \psi_l, \psi_u\}$
10 **until** $(\min_{i=1..3}(s_i) \neq s_3 \wedge \min_{i=1..3}(t_i) \neq t_3)$ `// Verify if the boundary is closed`;
11 $\theta' \leftarrow \theta$
12 **repeat** `// Sweep on the left side along the` θ`-axis from` θ'
13 $\quad \phi_l \leftarrow \arg\max_{\phi \in \Phi^-} \phi(\theta')$; $\phi_u \leftarrow \arg\min_{\phi \in \Phi^+} \phi(\theta')$ `// In the plane` (a, θ)
14 $\quad s_1 \leftarrow \max_{\phi \in \Phi^-} F^{>}_{\theta'}(\phi_l, \phi)$; $s_2 \leftarrow \max_{\phi \in \Phi^+} F^{>}_{\theta'}(\phi_u, \phi)$; $s_3 \leftarrow \max F^{>}_{\theta'}(\phi_l, \phi_u)$
15 $\quad \psi_l \leftarrow \arg\max_{\psi \in \Psi^-} \psi(\theta')$; $\psi_u \leftarrow \arg\max_{\psi \in \Psi^+} \psi(\theta')$ `// In the plane` (b, θ)
16 $\quad t_1 \leftarrow \max_{\psi \in \Psi^-} F^{>}_{\theta'}(\psi_l, \psi)$; $t_2 \leftarrow \max_{\psi \in \Psi^+} F^{>}_{\theta'}(\psi_u, \psi)$; $t_3 \leftarrow \max F^{>}_{\theta'}(\psi_l, \psi_u)$
17 $\quad \theta' \leftarrow \max\ (t_1, t_2, t_3, s_1, s_2, s_3)$ `// Find the next intersection`
18 $\quad \mathcal{B} \leftarrow \mathcal{B} \cup \{\phi_l, \phi_u, \psi_l, \psi_u\}$
19 **until** $(\max_{i=1..3}(s_i) \neq s_3 \wedge \max_{i=1..3}(t_i) \neq t_3)$ `// Verify if the boundary is closed`;
20 $N(v) \leftarrow \{w \in V \mid (v, w, f) \in E \wedge f \in \mathcal{B}\}$

The complexity of this algorithm is analysed below:

- finding the boundary tipping curves ϕ_l, ϕ_u, ψ_l and ψ_u at θ' (steps 4, 6, 13 and 15) presents a cost of $\mathcal{O}(N^2)$; and
- finding the next intersection of tipping curves on the left (resp. right) with the boundary curves ϕ_l, ϕ_u, ψ_l and ψ_u of θ' (steps 5, 7 and 8 (resp. 14, 16 and 17)) presents a cost of $\mathcal{O}(N^2)$,

while these two steps are repeated until the region surrounding the DRT is obtained. The number of iterations is then the size of the neighbourhood of v, denoted by m. Thus, the total complexity of the proposed algorithm is $\mathcal{O}(mN^2)$.

Since there are $\mathcal{O}(N^2)$ tipping curves in Φ^+, Φ^-, Ψ^+ and Ψ^-, $m = \mathcal{O}(N^2)$ in the worst case. However, in practice, we observe that m is bounded by a small constant. We obtain the value m experimentally using Algorithm 1 with different image sizes. The results are given in Fig. 5 for images of sizes varying from 5×5 to 80×80. For each image size, we randomly chose several DRTs, and compute the value of m. We notice $4 \leq m \leq 14$ over all the experiments.

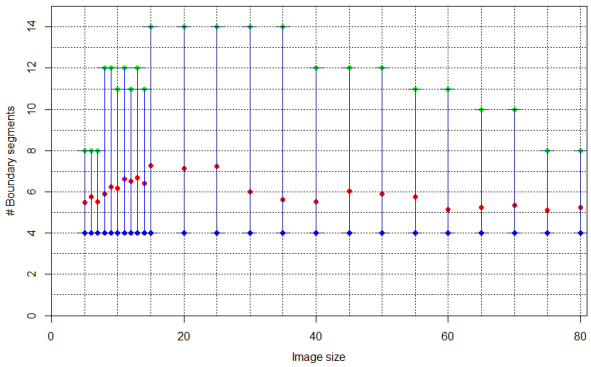


Fig. 5. Experiments on the size m of neighbourhoods of DRTs, calculated by Algorithm 1. The value m is calculated over 960 experiments for image size varying from 5×5 to 80×80, in each of which several DRTs are considered. Green (resp. blue and red) points denote the max (resp. min and average) value of m over the experiments.

4.2 Local search method

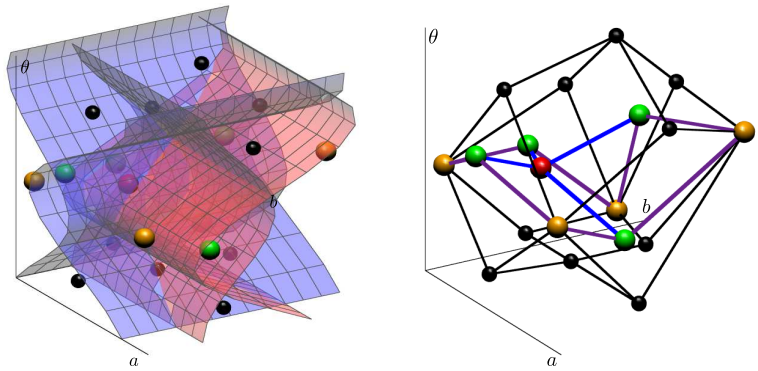
From Algorithm 1, we can now propose a procedure to determine a locally optimal solution of Eq. (5).

Since each DRT, *i.e.*, each vertex of the DRT graph G, corresponds to a *unique* transformed image (see Sec. 2.3), it is possible to obtain the gradient of d between A and the transformed image $T_v(B)$, denoted by Δd. By examining Δd for all neighbouring vertices of $N(v)$ for a given vertex v, we can then find a vertex – or more generally a set of vertices – presenting the minimal value $d(A, T_v(B))$. Then, by choosing a smallest valued vertex (or possibly several) in that set, and repeating this process, it is thus possible to carry out a standard gradient descent that finally leads to a locally optimum when, for its vertex, no neighbours present a better value. Note that the number of vertices of DRT graph of a finite image is finite [7], thus the local search stops at a local optimum.

As described in Sec. 4.1, Algorithm 1 for computing $N(v)$ of a given v requires as the input the set of upper and lower tipping curves. Initially, this set can be calculated from a given rigid transformation associated to (a, b, θ) such that it contains the pairs of tipping curves ϕ_{pqk}, ϕ_{pqk+1} and ψ_{pql}, ψ_{pql+1} as follows.

$$\Phi = \{\phi_{pqk}, \phi_{pqk+1} \mid \phi_{pqk}(\theta) \leq a \leq \phi_{pqk+1}(\theta)\}$$
$$\Psi = \{\psi_{pql}, \phi_{pql+1} \mid \psi_{pql}(\theta) \leq b \leq \psi_{pql+1}(\theta)\}$$

We denote $\Phi^- = \{\phi_{pqk} \in \Phi\}$ (resp. $\Phi^+ = \{\phi_{pqk+1} \in \Phi\}$) and $\Psi^- = \{\psi_{pql} \in \Psi\}$ (resp. $\Psi^+ = \{\psi_{pql+1} \in \Psi\}$) as *lower* (resp. *upper*) tipping curves. Note that, we can calculate $T_w(B)$ from $T_v(B)$, for $w \in N(v)$. This means that the set $\{\Phi^-, \Phi^+, \Psi^-, \Psi^+\}$ with respect to w is updated according to $T_w(B)$. Furthermore, thanks to the a- and b-convex property of the parameter space of rigid transformations (see Sec. 2.3), we can easily compute a new representative transformation (a, b, θ) of the DRT w knowing its boundary. (The details can be found in Appendix A.)



(a) Parameter space. (b) DRT graph.

Fig. 6. Example of k-neighbours of a DRT. The given DRT is depicted in red, its 1-neighbours in green and its 2-neighbours in both green and yellow.

As for any local search procedure, an important question is the choice of the initial vertex, *i.e.*, the seed v_0 of the gradient descent. To determine v_0, we can, *e.g.*, use the solution obtained by a conventional registration method as discussed in [4, 10]. Due to the effect of digitization on the transformed space, there are many local optima and we usually obtain the local optima basins. In order to improve this local search, it is possible to consider not only the direct neighbours, but also the k-neighbours ($k \geq 1$) in G (see Fig. 6). The set $N^k(v)$ of all the k-neighbours of a DRT v is defined, such that:

$$N^k(v) = N^{k-1}(v) \cup \bigcup_{u \in N^{k-1}(v)} N(u) \text{ and } N^1(v) = N(v).$$

To compute $N^k(v)$, we can use Algorithm 1 recursively. Note that the maximum number of k-neighbours of v is m^k, where m is the maximum degree of the DRT graph (*i.e.*, the number of 1-neighbours of a DRT). As stated in Sec. 4.2, finding 1-neighbours of a DRT costs $\mathcal{O}(mN^2)$. The complexity of the algorithm to find k-neighbours is then $\mathcal{O}(m^{k+1}N^2)$, where m has a constant behaviour in practice.

5 Experiments

The main purpose of this section is to illustrate the behaviour of the local search approach described in Sec. 4. These results and methods can be extend to various types of images such as grey-level, label or color, since DRT graphs are defined independently from the value space of the images (see Sec. 3). In such cases, it is necessary to use an appropriate distance. For the sake of readability, we focus here on the case of binary images, and the signed distance function –which gives fewer flat zones for the gradient term of d [9, 11]– to illustrate and analyse the issues related to the digitization on the discrete space of the transformed images.

In order to obtain an initial transformation (or a seed) for the proposed algorithm, we use a SIFT feature-based method [10], and show that our discrete

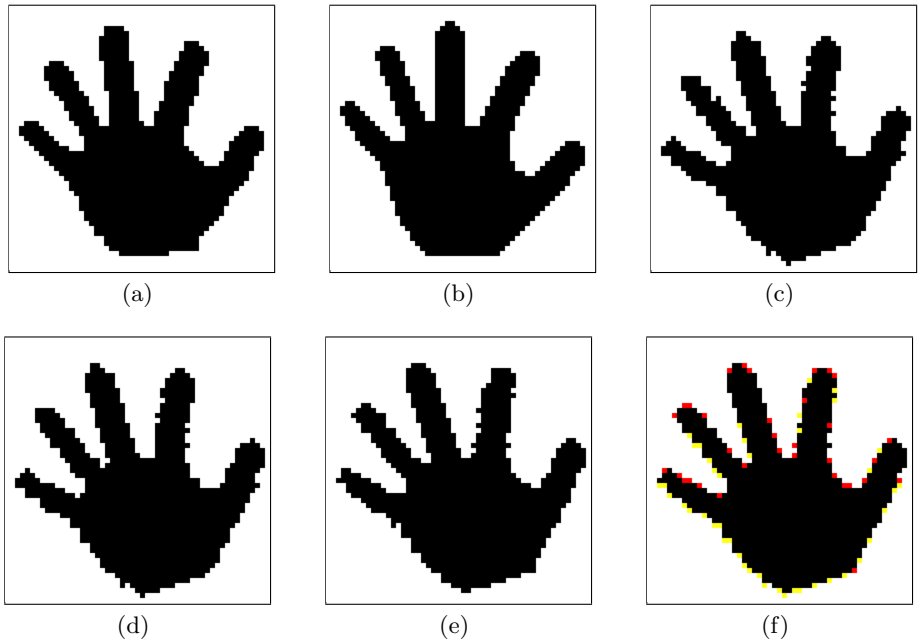
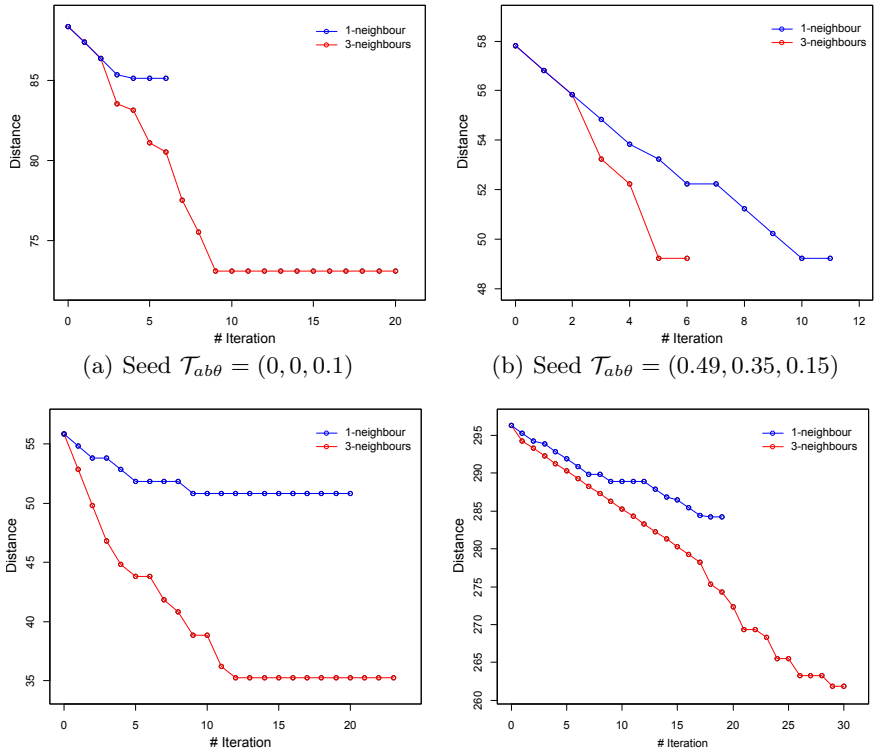


Fig. 7. Input binary images for the local search approach: (a) Reference image, and (b) target image. (c) Transformed image of (b) by the seed $\mathcal{T}_{ab\theta} = (0.52, 0.79, 0.3107)$. (d) and (e) Local optima obtained by using 1-neighbours and 3-neighbours respectively. (f) Different pixels between (d) and (e) in which yellow (resp. red) pixels are black (resp. white) in (d) and white (resp. black) in (e).

method can improve the result of this continuous method. Experiments are first carried out with the direct neighbours of several given seeds on a binary image of size 53×53 (see Fig. 7 for an example). Fig. 8 shows the results achieved with the algorithm described in Section 4.2 using 1-neighbours and 3-neighbours. We observe from these experiments that the local optimum distance and the convergence speed (*i.e.*, number of iterations) for the case of using 3-neighbours is better than those of using 1-neighbours.

6 Conclusion

We have proposed a purely discrete framework to solve the problem of 2D image registration under rigid transformations. This work is based on the recently introduced notion of DRT graph [7], which represents the discrete structure of subdivision of the parameter space of rigid transformations. Based on this discrete representation of the transformation parameter space, we have developed a gradient descent procedure that constructs, on the flight, the part of the space that has to be processed at each step, with a $\mathcal{O}(mN^2)$ time complexity, where $N \times N$ is the image size and m is the maximum degree of vertices in the DRT graph. In practice, m has a constant behaviour with respect to $N \times N$. This



(a) Seed $\mathcal{T}_{ab\theta} = (0, 0, 0.1)$ (b) Seed $\mathcal{T}_{ab\theta} = (0.49, 0.35, 0.15)$

(c) Seed $\mathcal{T}_{ab\theta} = (0.12, 0.05, 0.1314)$ (d) Seed $\mathcal{T}_{ab\theta} = (0.52, 0.79, 0.3107)$

Fig. 8. Local search from different initial transformations (or seeds) for the inputs in Fig. 7(a) and (b).

leads to an efficient algorithm in almost linear time with respect to the image size to compute the neighbours for the local search approach.

In the worst case, we have $m = \mathcal{O}(N^2)$. In future work, we hope to improve this upper bound for m. In addition, since we know how to compute neighbours on DRT graph, we can combine our proposed method with other combinatorial approaches [12] in order to improve our results. From a methodological viewpoint, several extensions to higher dimensions (3D in particular) and to various types of images such as grey-level, label or color images are also being considered for the proposed approach.

The research leading to these results has received funding from the French *Agence Nationale de la Recherche* (Grant Agreement ANR-10-BLAN-0205).

References

1. J. V. Hajnal, D. L. G. Hill, D. J. Hawkes, Medical Image Registration, CRC Press, 2001.
2. R. A. Schowengerdt, Remote Sensing: Models and Methods for Image Processing, 3rd Edition, Elsevier Academic Press, 2007.

3. A. Yilmaz, O. Javed, M. Shah, Object tracking: A survey, ACM Computing Surveys 38 (4) (2006) 1–45.
4. B. Zitová, J. Flusser, Image registration methods: A survey, Image and Vision Computing 21 (11) (2003) 977–1000.
5. P.-L. Bazin, D. L. Pham, Topology-preserving tissue classification of magnetic resonance brain images, IEEE Transactions on Medical Imaging 26 (4) (2007) 487–496.
6. S. Faisan, N. Passat, V. Noblet, R. Chabrier, C. Meyer, Topology preserving warping of 3-D binary images according to continuous one-to-one mappings, IEEE Transactions on Image Processing 20 (8) (2011) 2135–2145.
7. P. Ngo, Y. Kenmochi, N. Passat, H. Talbot, Combinatorial structure of rigid transformations in 2D digital images, Computer Vision and Image Understanding 117 (4) (2013) 393–408.
8. P. Ngo, Y. Kenmochi, N. Passat, H. Talbot, Combinatorial properties of 2D discrete rigid transformations under pixel-invariance constraints, in: IWCIA, Proceedings, Vol. 7655 of LNCS, Springer, 2012, pp. 234–248.
9. Y. Boykov, V. Kolmogorov, D. Cremers, A. Delong, An integral solution to surface evolution PDEs via geo-cuts, in: ECCV, Proceedings, Vol. 3953 of LNCS, Springer, 2006, pp. 409–422.
10. M. Amintoosi, M. Fathy, N. Mozayani, A fast image registration approach based on SIFT key-points applied to super-resolution, Imaging Science Journal 60 (4) (2011) 185–201.
11. F. F. Pedro, P. H. Daniel, Distance transforms of sampled functions, Theory of Computing 8 (19) (2012) 415–428.
12. E. Aarts, J. K. Lenstra, Local Search in Combinatorial Optimization, John Wiley & Sons, Inc., 1997.

A Appendix

Let us first formulate the problem as follows: Given a DRT v modeled by a set S of tipping surface pieces bounding v, report a transformation $\mathcal{T}_{ab\theta}$ of the chosen v. Note that S is described by $S = \{(\phi_i^+, q_i, q_{i+1}), (\phi_i^-, r_i, r_{i+1}), (\psi_i^+, s_i, s_{i+1}), (\psi_i^-, t_i, t_{i+1}), i = 1, \ldots, m \mid \phi_i^+ \in \Phi^+, \phi_i^- \in \Phi^-, \psi_i^+ \in \Psi^+ \text{ and } \psi_i^- \in \Psi^-\}$, where $(q_i, q_{i+1}), (r_i, r_{i+1}), (s_i, s_{i+1})$ and (t_i, t_{i+1}) are the θ-coordinates of the endpoints of the associated tipping curve segments $\phi_i^+, \phi_i^-, \psi_i^+$ and ψ_i^- respectively. Thanks to the a- and b-convexity of the bounded region by S, we propose Algorithm 2 to calculate $\mathcal{T}_{ab\theta}$ of v.

Algorithm 2: Finding a $\mathcal{T}_{ab\theta}$ associated to a DRT v.

Input: A DRT v, with the set S.
Output: A rigid transformation $\mathcal{T}_{ab\theta}$ of v.

1 $\theta_{min} \leftarrow \min_{(\phi_i^-, q_i, q_{i+1}) \in S} r_i$; $\theta_{max} \leftarrow \max_{(\phi_i^-, q_i, q_{i+1}) \in S} r_{i+1}$; $\theta \leftarrow (\theta_{min} + \theta_{max})/2$

2 $a_{min} \leftarrow \max_{(\phi_i^-, q_i, q_{i+1}) \in S} \phi_i^-(\theta)$; $a_{max} \leftarrow \min_{(\phi_i^+, r_i, r_{i+1}) \in S} \phi_i^+(\theta)$; $a \leftarrow (a_{min} + a_{max})/2$

3 $b_{min} \leftarrow \max_{(\psi_i^-, q_i, q_{i+1}) \in S} \psi_i^-(\theta)$; $b_{max} \leftarrow \min_{(\psi_i^+, r_i, r_{i+1}) \in S} \psi_i^+(\theta)$; $b \leftarrow (b_{min} + b_{max})/2$

4 $\mathcal{T}_{ab\theta} \leftarrow (a, b, \theta)$