A Combinatorial Solution to Non-Rigid 3D Shape-to-Image Matching

Florian Bernard[1,2,3]   Frank R. Schmidt[3]   Johan Thunberg[1]   Daniel Cremers[3]

[1]Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Luxembourg
[2]Centre Hospitalier de Luxembourg, Luxembourg
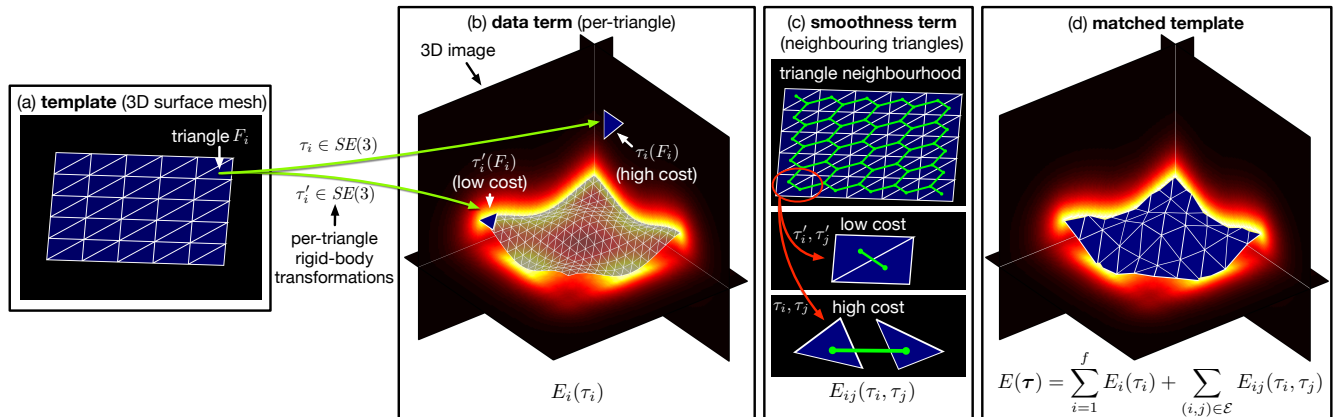[3]Technical University of Munich (TUM), Germany

Figure 1. 3D Shape-to-Image Matching. (a) Given a shape as a triangular mesh, we associate to each triangle F_i its rigid transformation $\tau_i \in \mathrm{SE}(3)$ such that the sum of data and smoothness terms is minimised. (b) The data term $E_i(\tau_i)$ measures how well the transformed triangle $\tau_i(F_i)$ fits into the volumetric image. (c) The smoothness term $E_{ij}(\tau_i, \tau_j)$ penalises the discrepancy between transformed triangles $\tau_i(F_i)$ and $\tau_j(F_j)$. (d) Optimising E provides us with a shape-to-image matching.

Abstract

We propose a combinatorial solution for the problem of non-rigidly matching a 3D shape to 3D image data. To this end, we model the shape as a triangular mesh and allow each triangle of this mesh to be rigidly transformed to achieve a suitable matching to the image. By penalising the distance and the relative rotation between neighbouring triangles our matching compromises between image and shape information. In this paper, we resolve two major challenges: Firstly, we address the resulting large and NP-hard combinatorial problem with a suitable graph-theoretic approach. Secondly, we propose an efficient discretisation of the unbounded 6-dimensional Lie group $\mathrm{SE}(3)$. *To our knowledge this is the first combinatorial formulation for non-rigid 3D shape-to-image matching. In contrast to existing local (gradient descent) optimisation methods, we obtain solutions that do not require a good initialisation and that are within a bound of the optimal solution. We evaluate the proposed method on the two problems of non-rigid 3D shape-to-shape and non-rigid 3D shape-to-image registration and demonstrate that it provides promising results.*

1. Introduction

Matching a shape template to an image is a well studied problem in computer vision and image analysis. It gives rise to a wide range of applications, including image segmentation and object detection. An early approach for the detection of lines and parametrised curves in images is the voting-based Hough transform [14], which was later generalised to the detection of arbitrary shapes [1].

Whilst the Hough transform considers rigid shapes, the utilisation of shape information in image segmentation tasks has also been addressed in the non-rigid case, including methods based on active shape models [10], level sets [12], convex shape spaces [13], multiphase graph cuts [58], or statistical shape models [19, 63]. For the reconstruction of the shape of an object from a single 2D image, shape-from-template approaches aim to match a given 3D template to the image via a 3D-to-2D projection [50, 38, 46]. Several authors have considered combinatorial formulations of the non-rigid shape-to-image matching problem for cer-

tain classes of shapes. For the case of matching contours [11, 53], or 2D chordal graph polygons [16], the resulting optimisation problems can be solved globally. However, a generalisation of these methods to 3D shapes is non-trivial and we are not aware of previous work addressing this issue. The purpose of this work is to fill this gap by presenting a combinatorial formulation for the non-rigid matching of a 3D shape template to a 3D image. For that, we model the shape as a triangular mesh and allow each triangle F_i of this mesh to be independently transformed via a rigid transformation $\tau_i \in \mathrm{SE}(3)$. Using a discretisation of the unbounded 6-dimensional Lie group $\mathrm{SE}(3)$, we formulate the matching task as a manifold-valued multi-labelling problem that can be cast as minimising the energy

$$E(\boldsymbol{\tau}) = \sum_{i=1}^{n} E_i(\tau_i) + \sum_{(i,j)\in\mathcal{E}} E_{ij}(\tau_i, \tau_j). \tag{1}$$

Here, the data term $E_i(\tau_i)$ takes the image information into account, while the smoothness term $E_{ij}(\tau_i, \tau_j)$ measures the dissimilarity between the observed shape and the modelled shape prior. By penalising the distance and the relative rotation of neighbouring triangles our matching compromises between image and shape information. In general, minimising functions of the form in (1) is NP-hard [26].

1.1. Related Work

To the best of our knowledge the present paper is the first one that considers a combinatorial formulation of the non-rigid 3D shape to 3D image matching problem. In the following we will summarise methodologies that are most relevant to our work.

Continuous Optimisation: In many scenarios it is natural to assume that image or shape deformations are spatially continuous and smooth. Frequently, such problems are formulated in terms of optimisation problems over the space of diffeomorphisms [15, 40, 2, 41]. Commonly, gradient descent-like methods are used to obtain (local) optima of the (typically non-convex) problems. However, a major shortcoming of these methods is that a good initial estimate is crucial and in general there are no bounds on the optimality of the solution. To deal with the non-convexity of a 2D shape-to-image matching problem that is formulated in terms of optimal transport, the authors in [52] propose to use a branch and bound scheme.

Shortest Paths and Dynamic Programming: In contrast to the continuous local optimisation methods, many vision problems can be formulated in a discrete manner such that they are amenable to solutions based on graph algorithms and dynamic programming (DP) [17]. Since curves are intrinsically one-dimensional, various curve matching formulations can also be reduced to finding a shortest-path in a particular graph. Moreover, based on a recursive formu-

lation using easier-to-solve subproblems, matching problems with templates that have a tree structure can frequently be tackled by DP. For a deformable matching of an open contour to a 2D image, a global solution based on DP has been proposed in [11]. Also based on DP, in [16] the authors present a method for solving the problem of deformably matching a 2D polygon to a 2D image for chordal graph polygons. In [53], the authors propose a globally optimal approach for matching a closed contour to a 2D image based on cycles in a product graph of the contour and the image. A related formulation that is also based on a product graph has recently been introduced in [30] for deformable contour to 3D shape matching.

Graph-cuts: It is well known (see e.g. [4]) that any cut of a graph can be interpreted as finding a closed manifold of co-dimension 1 in the ambient space (e.g., closed curve in 2D, closed surface in 3D, etc.). One such example is the reconstruction of a 3D shape from a set of sparse 3D points, where the latter is represented on a discrete 3D grid [35].

Labelling Problems: Labelling problems are ubiquitous in computer vision and appear both in continuous and discrete settings [62]. The popular Markov Random Field (MRF) framework offers a Bayesian treatment thereof [37]. Also, linear programming relaxations of MRFs have been studied [59]. The continuous approaches to multi-labelling include various convex relaxations [47, 32, 55, 18], multi-labelling problems with total variation regularisation of functions with values on manifolds [33], as well as sublabel-accurate convex relaxations [42, 31]. Among the discrete multi-labelling methods are the previously-mentioned graph-cuts, which can be used to find global solutions for certain binary labelling problems, including problems with submodular pairwise costs [25]. For a subclass of multi-labelling problems a global solution can also be found [25]. This sub-class includes pairwise costs that are convex in terms of totally ordered labels [22]. In addition, efficient algorithms for finding local optima of general multi-labelling problems have been proposed [6, 27], which even have theoretical optimality guarantees. A more detailed description of the energy functions that can be optimised using graph-cuts is given in [26, 24, 25].

1.2. Main Contributions

The main contribution of this paper is to present for the first time a combinatorial formulation of the non-rigid 3D shape to 3D image matching problem. Whilst our problem is a natural extension to the afore-mentioned "dimension one" matching approaches [11, 16, 53, 30], a generalisation to (intrinsic) *dimension two* problems is more intricate. Our main contributions are:

- By using a surface mesh transformation model that makes use of *per-triangle* rigid transformations, we

formulate the 3D shape to 3D image matching problem in terms of a manifold-valued multi-labelling problem.

- We introduce a pairwise term that defines a metric on the label space SE(3), which itself is a high-dimensional Lie group. With that, our energy function is amenable to be minimised by the α-expansion algorithm [6], which has been shown to work well in practice, is efficient even for very large label spaces, and has theoretical optimality guarantees.

- In contrast to continuous optimisation methods that use gradient descent-like algorithms, our combinatorial method does not require a good initialisation.

- In order to deal with the computationally challenging discretisation of SE(3), we propose to use a coarse-to-fine discretisation of the Lie group.

2. Non-Rigid 3D Shape-to-Image Matching

In this section we first specify our objective, followed by a description of the data term and smoothness term. After introducing the combinatorial problem, we describe the discretisation of the label space and we discuss the algorithmic solution of the problem.

2.1. Objective Function

In the following, we assume that a 3D shape $S \subset \mathbb{R}^3$ is given as a triangular mesh. This means we have $n \in \mathbb{N}$ triangles $F_1, \ldots, F_n \subset \mathbb{R}^3$ such that

$$S = \bigcup_{i=1}^n F_i. \tag{2}$$

We use the set $\mathcal{E} \subset \{1, \ldots, n\}^2$ to define the neighbourhood between pairs of (different) triangles. We assume that for all $(i,j) \in \mathcal{E}$ the neighbouring triangles F_i and F_j are non-disjoint and that the intersection $F_i \cap F_j$ results either in a common edge or a common vertex. Also, w.l.o.g. we assume that for each $(i,j) \in \mathcal{E}$ it holds that $i < j$, *i.e.* $(i,j) \in \mathcal{E} \Rightarrow (j,i) \notin \mathcal{E}$.

Our objective is it now to match the 3D shape S onto a volumetric image $I \colon \Omega \to \mathbb{R}^c$, where $\Omega \subset \mathbb{R}^3$ denotes the compact image domain and $c \in \mathbb{N}$ describes the amount of image channels. While we are interested in a non-rigid shape-to-image matching, we like to favour matchings that are as-rigid-as-possible, similar to the approach in [54] that applies (locally regularised) rigid transformations to *each vertex*. However, in our case this is done by applying to *each triangle* F_i a rigid transformation

$$\tau_i = (\tilde{\tau}_i, \vec{\tau}_i) \in \mathrm{SE}(3) = \mathrm{SO}(3) \ltimes \mathbb{R}^3, \tag{3}$$

where $\tilde{\tau}_i \in \mathrm{SO}(3) \subset \mathbb{R}^{3 \times 3}$ represents the rotational part and $\vec{\tau}_i \in \mathbb{R}^3$ represents the translational part of τ_i. The

task of finding the best matching $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)$ can be formulated as minimising the energy

$$E(\boldsymbol{\tau}) = \sum_{i=1}^n E_i(\tau_i) + \sum_{(i,j) \in \mathcal{E}} E_{ij}(\tau_i, \tau_j). \tag{4}$$

In Section 2.2 we define the data term $E_i(\tau_i)$ that evaluates how well the transformed triangle $\tau_i(F_i)$ fits to the image data. In Section 2.3 we define the smoothness term $E_{ij}(\tau_i, \tau_j)$ that measures the geometric dissimilarity between the shape model S and the transformed shape

$$\boldsymbol{\tau}(S) := \bigcup_{i=1}^n \tau_i(F_i). \tag{5}$$

Using the proposed piecewise rigid transformation model we may end up with a model $\boldsymbol{\tau}(S)$ having (small) gaps or intersections between neighbouring triangles. We will later address this issue and present a simple yet effective way of dealing with this irregularity.

2.2. Data Term

The data term $E_i(\tau_i) \in \mathbb{R}$ measures how well the transformed triangle $\tau_i(F_i)$ fits to the image data I. For that, we introduce the *score image* $J \colon \Omega \to [0,1]$ that is derived from the image I (e.g. a gradient magnitude image, or more advanced predictors based on neural networks). For a triangle $F \subset \Omega$, we define

$$J[F] := \int_F J(x) \mathrm{d}x. \tag{6}$$

With that, the value $J[F]$ indicates how well the triangle F fits to the image data, where a high value of the score image indicates a good fit. The data term is then given by

$$E_i(\tau_i) = -J[\tau_i(F_i)]. \tag{7}$$

In the discrete setting, the data term E_i is computed by a weighted sum of function values $-J(x)$ over the triangle. The weights take the rasterisation of the deformed triangle $\tau_i(F_i)$ in the image into account.

2.3. Smoothness Term

The pairwise term $E_{ij}(\tau_i, \tau_j) \in \mathbb{R}_0^+$ penalises the disagreement between neighbouring triangles F_i and F_j after they have been transformed by τ_i and τ_j, respectively.

For defining the pairwise term we first introduce suitable distances. The expression

$$d_{\mathrm{SO}(3)}(\tilde{\tau}_i, \tilde{\tau}_j) = \sqrt{\frac{1}{2}} \left\| \log(\tilde{\tau}_i^T \tilde{\tau}_j) \right\|_F \tag{8}$$

is the geodesic distance between the rotations $\tilde{\tau}_i$ and $\tilde{\tau}_j$ on $\mathrm{SO}(3)$ [21], with matrix logarithm $\log(\cdot)$. For q_i

and q_j being the quaternion representations of $\tilde{\tau}_i$ and $\tilde{\tau}_j$, one can efficiently compute the distance as $d_{\mathrm{SO(3)}} = 2\cos^{-1}(|\langle q_i, q_j \rangle|)$, where $\langle \cdot, \cdot \rangle$ is the quaternion inner-product [21].

For defining a distance between neighbouring triangles, we make use of the concept of group actions. To be more specific, we define

$$d_{\mathrm{SE(3)},X}(\tau_i, \tau_j) = \max_{x \in X} \|\tau_i(x) - \tau_j(x)\|_2, \qquad (9)$$

where the group $\mathrm{SE(3)}$ acts on the non-empty compact set $X \subseteq \mathbb{R}^3$. In our case we use $X = F_i \cap F_j$ such that $d_{\mathrm{SE(3)}, F_i \cap F_j}(\tau_i, \tau_j)$ can be seen as distance between the deformed triangles $\tau_i(F_i)$ and $\tau_j(F_j)$. In this case the maximum in $d_{\mathrm{SE(3)}, F_i \cap F_j}(\tau_i, \tau_j)$ is achieved at the common vertices of F_i and F_j, which is attractive from a computational point of view.

Using the introduced distances, we define our pairwise term as a weighted sum thereof, i.e.

$$\begin{aligned} E_{ij}(\tau_i, \tau_j) = \qquad\qquad\qquad\qquad (10) \\ \lambda_B d_{\mathrm{SO(3)}}(\tilde{\tau}_i, \tilde{\tau}_j) + \lambda_S d_{\mathrm{SE(3)}, F_i \cap F_j}(\tau_i, \tau_j). \end{aligned}$$

The purpose of the bending term, weighted by $\lambda_B > 0$, is to ensure that the rotations of neighbouring triangles are similar. The stretching term, weighted by $\lambda_S > 0$, ensures that neighbouring triangles stay close together.

2.4. Combinatorial Formulation

A matching $\boldsymbol{\tau}$ of shape S to the image I is given by a solution of the optimisation problem

$$\min_{\boldsymbol{\tau} \in \mathrm{SE(3)}^n} E(\boldsymbol{\tau}). \qquad (11)$$

Due to the non-convexity of the feasible set $\mathrm{SE(3)}^n$, it follows that Problem (11) is non-convex. This non-convexity makes it difficult to solve the problem directly over the unbounded continuous space $\mathrm{SE(3)}^n$. Our approach is to optimise instead over a discretisation of the search space. With that, we obtain a multi-labelling problem, for which efficient and effective algorithms are available.

For the discretisation of $\mathrm{SE(3)}$ we make use of the fact that it is a product space of $\mathrm{SO(3)}$ and $\mathbb{R}^3$. Thus, we define $\mathcal{L} \subset \mathrm{SO(3)} \ltimes \mathbb{R}^3 = \mathrm{SE(3)}$ to be the (finite) manifold-valued label space that contains $\ell = |\mathcal{L}|$ elements of $\mathrm{SE(3)}$.

Translations: The Lie group $\mathrm{SE(3)}$ is non-compact due to the translational part being encoded by $\mathbb{R}^3$. However, since the image domain Ω is compact, the image size provides natural bounds for a discretisation of the translations. Let n_x, n_y, n_z be the number of voxels of the image I and ℓ_x, ℓ_y, ℓ_z be the number of labels for the x, y, z translations. For convenience, and w.l.o.g., we assume that our template is defined relative to the centre of the image domain, i.e. the template's centre-of-gravity coincides with

the centre of the image. Moreover, w.l.o.g., we assume that we are looking for a matching such that a substantial part of the (transformed) template lies inside the image[1]. Let us define $\mathcal{Z}_m(n) = \{-\frac{n}{2}, \ldots, 0, \ldots, \frac{n}{2}\}$ to be the set containing m evenly-spaced elements with centre 0, where m is an odd positive integer. The *diameter* n defines the difference between the largest and the smallest elements. A discretisation of the translations is given by the set $\vec{\mathcal{L}} = \mathcal{Z}_{\ell_x}(n_x) \times \mathcal{Z}_{\ell_y}(n_y) \times \mathcal{Z}_{\ell_z}(n_z)$ with $|\vec{\mathcal{L}}| = \ell_x \cdot \ell_y \cdot \ell_z$.

Rotations: Various works that are related to the discretisation of $\mathrm{SO(3)}$ have previously been presented. These include sampling strategies for rigid-body path planning [29], an approximation of the neighbourhood in $\mathrm{SO(3)}$ based on vector distances [36], or an analysis of various metrics for 3D rotations [21]. Our discretisation of $\mathrm{SO(3)}$ is based on the *Hopf fibration*, which describes $\mathrm{SO(3)}$ in terms of the circle $\mathbb{S}^1$ and the 2-sphere $\mathbb{S}^2$. The intuition of this approach is to transfer a discretisation of $\mathbb{S}^1$ and $\mathbb{S}^2$ to the space of rotations. We refer the interested reader to [61] for a detailed description. Let $\tilde{\mathcal{L}}$ denote the so-obtained set of a uniform sampling of $\mathrm{SO(3)}$ containing $\tilde{\ell} = |\tilde{\mathcal{L}}|$ elements.

By optimising E over the label space $\mathcal{L}^n$, we now obtain the combinatorial optimisation problem as

$$\min_{\boldsymbol{\tau} \in \mathcal{L}^n} E(\boldsymbol{\tau}). \qquad (12)$$

3. Algorithm

In order to solve Problem (12), we use α-expansion [6, 26, 5], which greedily updates only one label at a time. We note that there are also potential alternatives to α-expansion (e.g. for non-metric pairwise terms [27, 55], or fusion moves [34]). Whilst α-expansion has the requirement that the pairwise term is a metric, it is appealing both from a practical and a theoretical point of view. To be more specific, it is efficient, robust with respect to initialisation, the obtained local optimum is guaranteed to lie within a factor of the global optimum, and an efficient implementation that supports the online computation of the smoothness term is available [6, 26, 5], which is crucial for the size of problems that we are solving. We now show that our pairwise term is a metric and thus α-expansion is applicable.

Lemma 1 *Let* $X \subseteq \mathbb{R}^3$ *be a non-empty compact set and* $\tau_i, \tau_j, \tau_k \in \mathrm{SE(3)}$. *The stretching term*

$$d_{\mathrm{SE(3)},X}(\tau_i, \tau_j) = \max_{x \in X} \|\tau_i(x) - \tau_j(x)\|_2,$$

is a pseudometric, i.e. *it satisfies*

(i) $d_{\mathrm{SE(3)},X}(\tau_i, \tau_j) \geq 0$, $d_{\mathrm{SE(3)},X}(\tau_i, \tau_i) = 0$,

(ii) symmetry: $d_{\mathrm{SE(3)},X}(\tau_i, \tau_j) = d_{\mathrm{SE(3)},X}(\tau_j, \tau_i)$, *and*

---

[1]If this is not the case, one can increase the image size accordingly.

(iii) the triangle inequality:
$$d_{\text{SE}(3),X}(\tau_i, \tau_k) \le d_{\text{SE}(3),X}(\tau_i, \tau_j) + d_{\text{SE}(3),X}(\tau_j, \tau_k).$$

Proof: (i) and (ii) follow directly from the definition. The triangle inequality holds, since

$$d_{\text{SE}(3),X}(\tau_i, \tau_k) = \max_{x \in X} \|\tau_i(x) - \tau_k(x)\|_2$$
$$= \max_{x \in X} \|\tau_i(x) - \tau_k(x) + \tau_j(x) - \tau_j(x)\|_2$$
$$\le \max_{x \in X} \left(\|\tau_i(x) - \tau_j(x)\|_2 + \|\tau_j(x) - \tau_k(x)\|_2 \right)$$
$$\le \max_{x \in X} \|\tau_i(x) - \tau_j(x)\|_2 + \max_{x \in X} \|\tau_j(x) - \tau_k(x)\|_2$$
$$= d_{\text{SE}(3),X}(\tau_i, \tau_j) + d_{\text{SE}(3),X}(\tau_j, \tau_k). \qquad \blacksquare$$

Proposition 1 *For $\lambda_S, \lambda_B > 0$ and $(i,j) \in \mathcal{E}$, the pairwise term $E_{ij}(\cdot, \cdot)$ defined in (10) is a metric.*

Proof: Due to the assumption in Section 2.1, for $(i,j) \in \mathcal{E}$ it follows that $X := F_i \cap F_j \ne \emptyset$ is compact. Thus, $d_{\text{SE}(3),X}(\cdot, \cdot)$ is a pseudometric (Lemma 1). Whilst $d_{\text{SO}(3)}(\cdot, \cdot)$ is known to be a metric on SO(3), it is only a pseudometric on SE(3). Since $E_{ij}(\cdot, \cdot)$ is a positive linear combination of two pseudometrics, $E_{ij}(\cdot, \cdot)$ is also a pseudometric. To show that $E_{ij}(\cdot, \cdot)$ is a metric, we show that $E_{ij}(\tau_i, \tau_j) = 0$ implies $\tau_i = \tau_j$. For $E_{ij}(\tau_i, \tau_j) = 0$, it holds that $d_{\text{SO}(3)}(\tilde{\tau}_i, \tilde{\tau}_j) = 0$, which implies $\tilde{\tau}_i = \tilde{\tau}_j$. Moreover, with $E_{ij}(\tau_i, \tau_j) = 0$ and $\tilde{\tau}_i = \tilde{\tau}_j$, it holds that $d_{\text{SE}(3),X}(\tau_i, \tau_j) = \max_{x \in X} \|(\tilde{\tau}_i(x) + \vec{\tau}_i) - (\tilde{\tau}_j(x) + \vec{\tau}_j)\|_2 = \|\vec{\tau}_i - \vec{\tau}_j\|_2 = 0$, which implies $\vec{\tau}_i = \vec{\tau}_j$. Hence $\tau_i = \tau_j$. $\qquad \blacksquare$

3.1. Coarse-to-Fine Processing

In practice, for a reasonably large number of labels $\ell = \tilde{\ell} \cdot \ell_x \cdot \ell_y \cdot \ell_z$, a direct solution of Problem (12) is intractable. In order to cope with this issue we propose to use a coarse-to-fine strategy that (approximately) solves Problem (12) at different levels s of the label space. Let $s=0$ denote the coarsest (initial) level and $s = s_{\max} \ge 0$ the finest (final) level. Once a solution $\boldsymbol{\tau}^{(s)}$ has been obtained at level s, for running the algorithm at level $s+1$ the labelling is initialised with $\boldsymbol{\tau}^{(s)}$ and the label space is updated accordingly. For computational efficiency, in the coarse-to-fine approach each triangle has its own feasible label space. Let $\mathcal{L}_i^{(s)}$ denote this feasible label space for the i-th triangle at level s. Initially, on the base level $s=0$, the label spaces are the same for each triangle, *i.e.* $\mathcal{L}_i^{(0)} = \mathcal{L}^{(0)}$. The general idea for obtaining $\mathcal{L}_i^{(s+1)}$ is to consider a (uniform) discretisation of the neighbourhood of the transformation $\tau_i^{(s)}$ at level s, where the radius of the neighbourhood decreases across the levels. Let us introduce a neighbourhood on SO(3):

Definition 1 *(ϵ-ball on SO(3))*
The ball on SO(3) with radius ϵ and centre $\tilde{\tau} \in \text{SO}(3)$ is defined as $\mathcal{B}_\epsilon^{\text{SO}(3)}(\tilde{\tau}) = \{\tau \in \text{SO}(3) : d_{\text{SO}(3)}(\tau, \tilde{\tau}) < \epsilon\}$.

Next, we describe the coarse-to-fine structure of the label space based on its product space nature. Since each label can be written as

$$\tau_i^{(s)} = (\tilde{\tau}_i^{(s)}, \vec{\tau}_i^{(s)}) \in \tilde{\mathcal{L}}_i^{(s)} \times \vec{\mathcal{L}}_i^{(s)} \subset \text{SO}(3) \ltimes \mathbb{R}^3, \quad (13)$$

we can consider the translations and rotations independently. Note that $\tilde{\mathcal{L}}_i^{(s)} \times \vec{\mathcal{L}}_i^{(s)} \subset \text{SE}(3)$ is not necessarily a group anymore. By enforcing that $\tau_i^{(s)} \in \mathcal{L}_i^{(s+1)}$, the solution $\boldsymbol{\tau}^{(s)}$ at level s is also contained in the new label space. Thus, the energy cannot increase from level s to $s+1$.

Translations: We define $\vec{\mathcal{L}}^{(0)} := \vec{\mathcal{L}}$ (cf. Section 2.4). For obtaining the set of translations at level $s+1$ for triangle i, the new translation grid at level $s+1$ is centred at $\vec{\tau}_i^{(s)}$. Moreover, the diameter from level s is reduced by a factor of two, leading to

$$\vec{\mathcal{L}}_i^{(s+1)} := \qquad\qquad\qquad\qquad\qquad\qquad (14)$$
$$\vec{\tau}_i^{(s)} + \left(\mathcal{Z}_{\ell_x}(\frac{n_x^{(s)}}{2}) \times \mathcal{Z}_{\ell_y}(\frac{n_y^{(s)}}{2}) \times \mathcal{Z}_{\ell_z}(\frac{n_z^{(s)}}{2}) \right),$$

where the vector-set addition is element-wise. Initially, $n_x^{(0)} = n_x, n_y^{(0)} = n_y$ and $n_z^{(0)} = n_z$.

Rotations: Let $\tilde{\mathcal{L}}^{[r]}$ denote a discretisation of (the entire) SO(3) at resolution r containing $\tilde{\ell}^{[r]}$ elements, where $\tilde{\ell}^{[r]}$ increases with increasing r. $\tilde{\mathcal{L}}^{[r]}$ should not be confused with $\tilde{\mathcal{L}}_i^{(s)}$, which is the (rotation) label space of the i-th triangle at level s that is to be defined below. Following the construction in [61], we obtain 5 resolutions of the SO(3) discretisation $\tilde{\mathcal{L}}^{[r]}$ for $r = 0, \dots, 4$. After including the identity in $\tilde{\mathcal{L}}^{[r]}$, the number of elements ranges from $\tilde{\ell}^{[0]} = 577$ to $\tilde{\ell}^{[4]} \approx 2 \cdot 10^6$. Let us define a set that contains a fixed number of p elements from $\tilde{\mathcal{L}}^{[r]}$ that are "closest" to the identity, *i.e.*

$$\tilde{\mathcal{L}}_p^{[r]} := \tilde{\mathcal{L}}^{[r]} \cap \mathcal{B}_\epsilon^{\text{SO}(3)}(\text{Id}), \qquad\qquad (15)$$

where for each resolution r the smallest ϵ that fulfils $|\tilde{\mathcal{L}}_p^{[r]}| \ge p$ is used as radius.

Now, we define $\tilde{\mathcal{L}}^{(0)} := \tilde{\mathcal{L}}^{[0]}$, and

$$\tilde{\mathcal{L}}_i^{(s+1)} := \{\tau \tilde{\tau}_i^{(s)} : \tau \in \tilde{\mathcal{L}}_p^{[s+1]} \subseteq \text{SO}(3)\}, \qquad (16)$$

which is the set of compositions of $\tilde{\tau}_i^{(s)}$ with all rotations in $\tilde{\mathcal{L}}_p^{[s+1]}$. We use $p = \tilde{\ell}^{[0]} = 577$ for all levels s. For the predefined SO(3) griddings at resolutions $r = 0, \dots, 4$ there always existed an ϵ such that the above inequality is tight, *i.e.* $|\tilde{\mathcal{L}}_p^{[r]}| = \tilde{\ell}^{[0]} = 577$.

By including the identity in $\tilde{\mathcal{L}}_p^{[s+1]}$, we make sure that $\tilde{\tau}_i^{(s)} \in \tilde{\mathcal{L}}_i^{(s+1)}$. Since $0 \in \mathcal{Z}.(\cdot)$, it follows that $\vec{\tau}_i^{(s)} \in \vec{\mathcal{L}}_i^{(s+1)}$. Thus, we have $\tau_i^{(s)} \in \mathcal{L}_i^{(s+1)}$, ensuring that the energy cannot increase when moving from level s to $s+1$.

3.2. Practical Considerations

In this section we describe some aspects for the application of the proposed method in practice.

3.2.1 Mesh Connectivity

After applying an individual rigid-body transformation to each triangle of the template mesh, in general the resulting mesh may have gaps or intersections between neighbouring triangles. However, due to the introduced regulariser, these gaps or intersections can be expected to be rather small. For recovering the original mesh topology, we replace each subset of vertices having the same position in the mesh template by their centre-of-gravity after the transformation.

3.2.2 Memory Requirements

For running our algorithm we pre-compute the data term, requiring memory of $\mathcal{O}(n \cdot \ell)$ (the online computation has constant memory requirements but leads to a significantly increased runtime). Since pre-computing the full pairwise term requires memory of $\mathcal{O}(n^2 \cdot \ell^2)$, we only precompute the bending term $d_{\mathrm{SO}(3)}$, requiring memory of $\mathcal{O}(\tilde{\ell}^2)$. The stretching term $d_{\mathrm{SE}(3),X}$ is computed online.

4. Results

For the evaluation of our method we focus on demonstrating the general applicability of our approach. Both point-set registration [3, 48, 8, 36, 44, 43, 23, 20, 49, 39] and the related correspondence problem for 3D shapes [56, 60, 28, 7] can be tackled using our method by recasting them as shape-to-image matching problem. Hence, in our evaluation below, in addition to 3D image segmentation, we also consider the case of deformable mesh registration.

4.1. Deformable Mesh Registration

In the first set of experiments we demonstrate that our method can be used to perform deformable mesh registration. To emphasise that our method is insensitive to initialisation, we compare it exemplarily with the Coherent Point Drift (CPD) algorithm [44, 43], a widely-used point-cloud registration method based on Expectation Maximisation.

Template and Target: For the evaluation we use a low-resolution mesh of the *Stanford bunny* as template ($n = 498$), as shown in Fig. 2 (top left). A total of 20 deformed versions of the bunny mesh, each with a random pose, are used as registration targets. For that, we synthetically create deformed versions of a high-resolution bunny mesh (Fig. 2, top right) based on random 3D displacement vectors defined at 8 control points on a cubic grid. These displacement vectors are then transferred to the mesh using a spline-based

interpolation in order to achieve a smooth and nonlinear deformation. One such deformed mesh is shown in Fig. 2 (bottom left). Eventually, a random pose transform is applied to the deformed shape, as shown in Fig. 2 (bottom right).
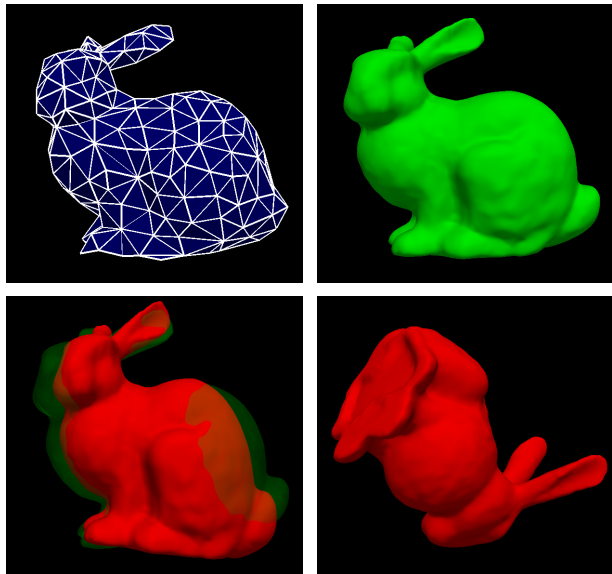

Figure 2. Bunny meshes. Top left: template. Top right: high-resolution target. Bottom left: deformed target (with original target overlay). Bottom right: deformed target with random pose.
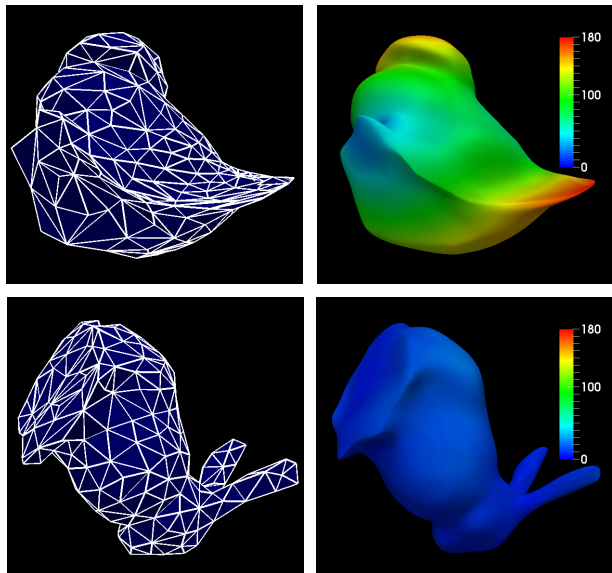

Figure 3. Qualitative results for registrations of the bunny template to the deformed target with random pose (cf. Fig. 2, bottom right). Top left: CPD result (shape destroyed). Top right: CPD error. Bottom left: our result. Bottom right: our error.

Score Image: In order to use our method for mesh registration we create a score image for each target mesh and then fit the template mesh to the score image. For $d : \Omega \to \mathbb{R}_0^+$ we denote by $d(x)$ the distance of position x to the boundary of the target mesh. Now, we define the score image as $J(x) = \exp(-\frac{d(x)}{\beta})$, where we used $\beta = 2$.
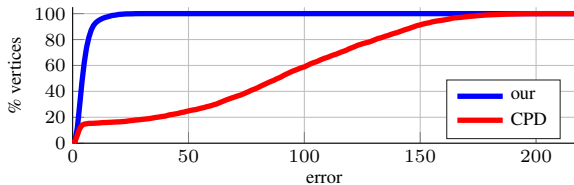
Figure 4. Percentage of vertices (vertical axis) which have an error that is smaller than or equal to the value on the horizontal axis.

Parameters: We set $\lambda_S = 54 \cdot \frac{g}{\max(\{n_x, n_y, n_z\})}$ and $\lambda_B = 27 \cdot \frac{g}{\pi}$, where the normalisation factor $g = \frac{nc_{\max}}{2|\mathcal{E}|}$ takes the problem size into account. The positive number $c_{\max}$ is an upper bound for the largest possible absolute value of the data term for a single triangle (cf. eq. (6)), which we compute by multiplying the largest value of the score image by the area of the largest triangle. The size of the label space is $|\mathcal{L}^{(0)}| = 9^3 \cdot 577 = 420{,}633$, and the dimensions of the score image range from 208^3 to 262^3, which resulted in an average processing time of ≈ 92 minutes per registration on a MacBook Pro (2.5GHz, 16GB).

Results: In the case of CPD, we first solve for a rigid registration and then for a non-rigid registration (performing a non-rigid registration directly performed worse). Since CPD is highly initialisation-sensitive it fails in 17 out of the 20 evaluated cases, where one representative failure case is depicted in Fig. 3 (top row). This extreme amount of corrupt registrations emphasises the necessity of a method that is robust with respect to initialisation. In contrast, in all 20 cases our method is able to achieve a good registration, see Fig. 3 (bottom row) for a representative result. In Fig. 4 we present a quantitative evaluation.

Discussion: Whilst we do not claim to present an exhaustive evaluation of mesh registration methods, we demonstrated the insensitivity to initialisation of our method in a proof of concept manner. One advantage of our approach is that we neither have the necessity of compatible mesh topologies, nor of compatible mesh discretisations, since the target is represented in terms of the score image. Since the score image is a discrete representation of the target shape *surface*, our approach amounts to a surface-based registration, rather than a point-based registration as CPD, which is biased towards aligning points to be as close as possible. Moreover, the score image offers further flexibility since additional information can be integrated (e.g. uncertainties, mesh texture, shape features, etc.).

4.2. Segmentation

In the second set of experiments we apply our method to the segmentation of four brain structures (*substantia nigra & subthalamic nucleus* as single object and the *nucleus ruber*, both bilaterally) in 16 multi-modal 3T magnetic resonance images. The delineation of the *subthalamic nucleus* is known to be a challenging task, even for humans [51].

The main difficulties include weak image contrasts and the small size of the brain structures (the structures shown in Fig. 5 are contained in a bounding box of $\approx 6 \times 3 \times 2.5 \text{cm}^3$, with the MRI image covering a volume of $\approx 20^3 \text{cm}^3$).

Template: For capturing the inter-relation between the brain structures, we use a multi-object template ($n{=}379$), as shown in Fig. 5. The template connects neighbouring brain structures by (degenerate) triangles, referred to as "phantom triangles", which are used only for the smoothness term and are "free" with respect to the data term.
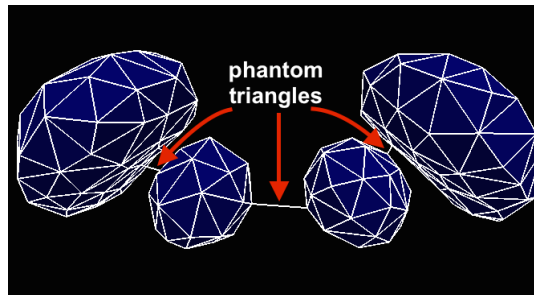

Figure 5. Brain structure template.

Parameters: We set $\lambda_S = 135 \cdot \frac{g}{\max(\{n_x, n_y, n_z\})}$ and $\lambda_B = 90 \cdot \frac{g}{\pi}$, where g is defined as before. The size of the label space is $|\mathcal{L}^{(0)}| = 11^3 \cdot 577 = 767{,}987$, and the dimension of all score images is $364 \times 436 \times 364$, which resulted in an average processing time of ≈ 58 minutes per fitting.

Score Image: In order to perform image segmentation with our method, we use a data term that is based on the recently proposed 3D U-Net CNN [9]. For all 16 images we train the network in a leave-one-out manner for the prediction of volumetric segmentations. In the centre column of Fig. 7 three examples of so-predicted volumetric segmentations are shown. For this challenging segmentation task the U-Net is able to identify the (rough) location of the brain structures, but does in many cases not produce an output that resembles the shape of the brain structures (the first two rows in Fig. 7). Thus, we complement the U-Net segmentations with geometric information using our method.

For each of the four brain structures $o \in \{1, 2, 3, 4\}$ we use an individual score image J_o. Given the binary U-Net segmentation $I_o^{\text{unet}} : \Omega \to \{0, 1\}$ for brain structure o, we first extract the (predicted) boundary using morphological operations. For $d_o : \Omega \to \mathbb{R}_0^+$ we denote by $d_o(x)$ the distance of position x to the so-extracted boundary. Then, we use a Gaussian kernel to define the score image as

$$J_o(x) = w_o \exp\left(-\frac{-d_o^2(x)}{2\sigma_o^2}\right). \qquad (17)$$

The weight w_o and bandwidth σ_o are used for incorporating the confidence about the U-Net segmentation I_o^{unet} of brain structure o. To this end, for

$$\mathcal{Y}_o = \{x \in \mathbb{R}^3 : I_o^{\text{unet}}(x) = 1\} \qquad (18)$$

being the one-level set of I_o^{unet} that represents the point-cloud of segmented voxels, we use the average of the (per-coordinate) *median absolute deviation* (MAD) as robust *dispersion* measure, computed as

$$\overline{MAD}(\mathcal{Y}_o) = \frac{1}{3}\|MAD(\mathcal{Y}_o)\|_1, \tag{19}$$

for $MAD(\mathcal{Y}_o) = \mathrm{median}(|\mathcal{Y}_o - \mathrm{median}(\mathcal{Y}_o)|) \in \mathbb{R}^3$.

The median is understood in a per-coordinate sense, and both the set-vector difference and the absolute value are understood element-wise. Now, for $\mathcal{Y}'_o$ denoting the point-cloud of segmented voxels of structure o as given by the template, the absolute value of the average MAD difference is given by $h_o = |\overline{MAD}(\mathcal{Y}_o) - \overline{MAD}(\mathcal{Y}'_o)|$. With that, we define $\sigma_o = \rho(h_o+1)$, scaled by $\rho=3$. Thus, if the average MAD for the U-Net segmentation and the template are equal, the bandwidth corresponds to ρ, whereas a larger difference in dispersion leads to a larger bandwidth, accounting for more uncertainty in the U-Net segmentation. Moreover, we define $w_o = \frac{1}{h_o+1}$ such that an increased uncertainty in the U-Net segmentation of brain structure o leads to a decreased weight for its data term.

Results: For the evaluation of the segmentation we use the Dice Similarity Coefficient (DSC) as volumetric overlap measure, which is defined as $\frac{2|\mathcal{Y} \cap \mathcal{Y}'|}{|\mathcal{Y}|+|\mathcal{Y}'|}$, for $\mathcal{Y}$ and $\mathcal{Y}'$ each being point-clouds of segmented voxels. We compute the DSC for each individual brain structure, and then report the average of the four DSC values. Quantitative results comparing the plain U-Net segmentation and our obtained segmentations are presented in Fig. 6. The boxplot on the left reveals that in overall our method achieves higher volumetric overlaps across the 16 cases. Moreover, the plot of sorted DSC differences on the right emphasises that applying our method improves the DSC in most cases (the values above zero), and in only a few cases it is reduced slightly. In Fig. 7
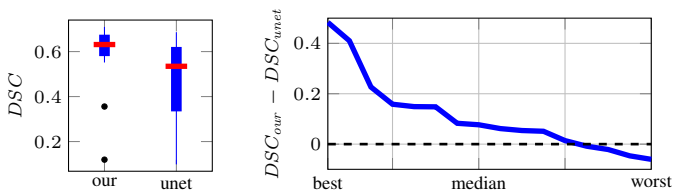

Figure 6. Left: Boxplot of the DSC of our method versus the U-Net segmentation. Right: Sorted DSC differences for the 16 cases (values above zero indicate an improvement upon U-Net).

qualitative results for three segmentation cases that correspond to the *best*, *median* and *worst* cases in Fig. 6 (right) are shown. In the first row of Fig. 7 it can be seen that in some cases our method is even able to achieve a reasonable segmentation based on a poor U-Net segmentation. This was possible by putting a stronger emphasis on the shape information relative to the data term, which also biases the

method towards the shape information. A related discussion on the biasedness of model-to-data fitting approaches in structure-from-motion can be found in [45].
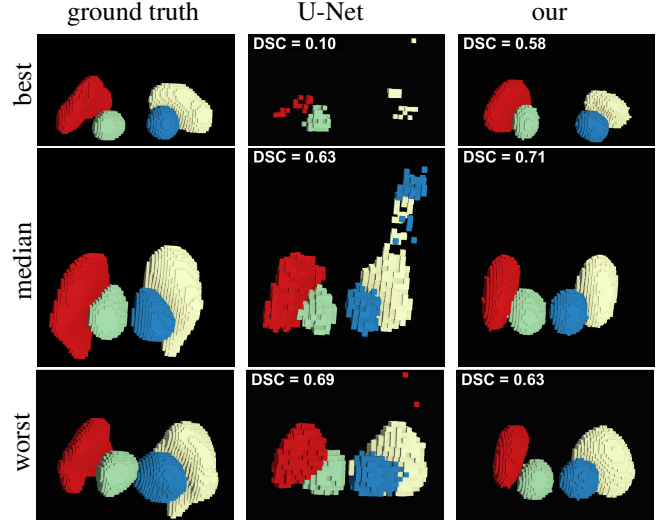

Figure 7. Qualitative results for the brain structure segmentation experiments. Each row shows a different instance of results, where from top to bottom we present the *best*, *median* and *worst* DSC differences (cf. Fig. 6, right).

5. Conclusion

We introduced the first combinatorial method for non-rigidly matching a 3D shape to a 3D image. The key idea is to represent the 3D shape as a triangular mesh and to solve a manifold-valued multi-labelling problem on the set of triangles. We determine an assignment of a rigid-body transformation associated with each triangle by minimising a cost function where the unary terms encode the local matching cost in the image and the pairwise terms penalise the amount of non-rigidity in the deformation. In particular, we propose an efficient discretisation of the unbounded 6-dimensional Lie group of rigid motions. Moreover, we solve the large and NP-hard optimisation problem with a graph theoretic algorithm that is insensitive to initialisation and has the guarantee that the obtained solution is within a factor of the global optimum [6]. Experimental validation confirms these benefits.

References

[1] D. H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 1

[2] M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, 2005. 2

[3] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *TPAMI*, 14(2):239–256, 1992. 6

[4] Y. Boykov and V. Kolmogorov. Computing Geodesics and Minimal Surfaces via Graph Cuts. *ICCV*, 2003. 2

[5] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *TPAMI*, 26(9):1124–1137, 2004. 4

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *TPAMI*, 23(11):1222–1239, 2001. 2, 3, 4, 8

[7] Q. Chen and V. Koltun. Robust Nonrigid Registration by Convex Optimization. *ICCV*, 2015. 6

[8] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114–141, 2003. 6

[9] Ö. Cicek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In *MICCAI*, 2016. 7

[10] T. F. Cootes and C. J. Taylor. Active Shape Models - Smart Snakes. In *BMVC*, pages 266–275, 1992. 1

[11] J. Coughlan, A. Yuille, C. English, and D. Snow. Efficient deformable template detection and localization without user initialization. *Computer Vision and Image Understanding*, 78(3):303–319, 2000. 2

[12] D. Cremers. Dynamical statistical shape priors for level set-based tracking. *TPAMI*, 28(8):1262–1273, 2006. 1

[13] D. Cremers, F. R. Schmidt, and F. Barthel. Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. In *CVPR*, 2008. 1

[14] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972. 1

[15] P. Dupuis, U. Grenander, and M. I. Miller. Variational problems on flows of diffeomorphisms for image matching. *Quarterly of applied mathematics*, pages 587–600, 1998. 2

[16] P. F. Felzenszwalb. Representation and detection of deformable shapes. *TPAMI*, 27(2):208–220, 2005. 2

[17] P. F. Felzenszwalb and R. Zabih. Dynamic Programming and Graph Algorithms in Computer Vision. *TPAMI*, 33(4):721–740, 2011. 2

[18] B. Goldluecke, E. Strekalovskiy, and D. Cremers. Tight convex relaxations for vector-valued labeling. *SIAM Journal on Imaging Sciences*, 6(3):1626–1664, 2013. 2

[19] T. Heimann and H.-P. Meinzer. Statistical shape models for 3D medical image segmentation: A review. *Medical Image Analysis*, 13(4):543–563, 2009. 1

[20] R. Horaud, F. Forbes, M. Yguel, G. Dewaele, and J. Zhang. Rigid and articulated point registration with expectation conditional maximization. *TPAMI*, 33(3):587–602, Mar. 2011. 6

[21] D. Q. Huynh. Metrics for 3D Rotations: Comparison and Analysis. *Journal of Mathematical Imaging and Vision*, 35(2):155–164, June 2009. 3, 4

[22] H. Ishikawa. Exact optimization for Markov random fields with convex priors. *TPAMI*, 25(10):1333–1336, 2003. 2

[23] B. Jian and B. C. Vemuri. Robust Point Set Registration Using Gaussian Mixture Models. *TPAMI*, 33(8):1633–1645, 2011. 6

[24] V. Kolmogorov and Y. Boykov. What Metrics Can Be Approximated by Geo-Cuts, Or Global Optimization of Length/Area and F. *ICCV*, 2005. 2

[25] V. Kolmogorov and C. Rother. Minimizing non-submodular functions with graph cuts – a review. *TPAMI*, 29(7):1274–1279, 2007. 2

[26] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *TPAMI*, 26(2):147–159, Feb. 2004. 2, 4

[27] N. Komodakis and G. Tziritas. Approximate labeling via graph cuts based on linear programming. *TPAMI*, 29(8):1436–1453, 2007. 2, 4

[28] A. Kovnatsky, M. M. Bronstein, X. Bresson, and P. Vandergheynst. Functional correspondence by matrix completion. In *CVPR*, 2015. 6

[29] J. J. Kuffner. Effective sampling and distance metrics for 3D rigid body path planning . In *ICRA*, Jan. 2004. 4

[30] Z. Lähner, E. Rodolà, F. R. Schmidt, M. M. Bronstein, and D. Cremers. Efficient Globally Optimal 2D-to-3D Deformable Shape Matching. *CVPR*, 2016. 2

[31] E. Laude, T. Möllenhoff, M. Möller, J. Lellmann, and D. Cremers. Sublabel-Accurate Convex Relaxation of Vectorial Multilabel Energies. *ECCV*, 2016. 2

[32] J. Lellmann and C. Schnörr. Continuous multiclass labeling approaches and algorithms. *SIAM Journal on Imaging Sciences*, 4(4):1049–1096, 2011. 2

[33] J. Lellmann, E. Strekalovskiy, S. Koetter, and D. Cremers. Total variation regularization for functions with values in a manifold. In *ICCV*, 2013. 2

[34] V. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion moves for markov random field optimization. *TPAMI*, 32(8):1392–1405, 2010. 4

[35] V. S. Lempitsky and Y. Boykov. Global Optimization for Shape Fitting. *CVPR*, 2007. 2

[36] H. Li and R. Hartley. The 3D-3D Registration Problem Revisited. In *ICCV*, 2007. 4, 6

[37] S. Z. Li. *Markov random field modeling in computer vision*. Springer Science & Business Media, 2012. 2

[38] A. Malti, A. Bartoli, and R. Hartley. A Linear Least-Squares Solution to Elastic Shape-From-Template. *CVPR*, 2015. 1

[39] H. Maron, N. Dym, I. Kezurer, S. Kovalsky, and Y. Lipman. Point registration via efficient convex relaxation. *ACM Transactions on Graphics (TOG)*, 35(4):73, 2016. 6

[40] P. W. Michor and D. Mumford. Riemannian Geometries on Spaces of Plane Curves. In *Journal of the European Mathematical Society*, 2004. 2

[41] M. I. Miller, A. Trouvé, and L. Younes. Geodesic shooting for computational anatomy. *Journal of Mathematical Imaging and Vision*, 24(2):209–228, 2006. 2

[42] T. Möllenhoff, E. Laude, M. Moeller, J. Lellmann, and D. Cremers. Sublabel-Accurate Relaxation of Nonconvex Energies. In *CVPR*, 2016. 2

[43] A. Myronenko and X. Song. Point Set Registration: Coherent Point Drift. *TPAMI*, 32(12):2262–2275, Dec. 2010. 6

[44] A. Myronenko, X. Song, and M. A. Carreira-Perpinán. Non-rigid point set registration: Coherent Point Drift. In *NIPS*, 2007. 6

[45] I. Nurutdinova and A. Fitzgibbon. Towards Pointless Structure from Motion: 3D Reconstruction and Camera Parameters from General 3D Curves. In *ICCV*, 2015. 8

[46] S. Parashar, D. Pizarro, A. Bartoli, and T. Collins. As-Rigid-as-Possible Volumetric Shape-from-Template. *ICCV*, 2015. 1

[47] T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers. A convex formulation of continuous multi-label problems. In *ECCV*, 2008. 2

[48] A. Rangarajan, H. Chui, and F. L. Bookstein. The softassign procrustes matching algorithm. In *IPMI*, 1997. 6

[49] A. Rasoulian, R. Rohling, and P. Abolmaesumi. Group-wise registration of point sets for statistical shape models. *TMI*, 31(11):2025–2034, Nov. 2012. 6

[50] M. Salzmann and P. Fua. Reconstructing sharply folding surfaces: A convex formulation. In *CVPR*, 2009. 1

[51] J. R. Schlaier, C. Habermeyer, J. Warnat, M. Lange, A. Janzen, A. Hochreiter, M. Proescholdt, A. Brawanski, and C. Fellner. Discrepancies between the MRI- and the electrophysiologically defined subthalamic nucleus. *Acta Neurochirurgica*, 153(12):2307–2318, July 2011. 7

[52] B. Schmitzer and C. Schnörr. Globally optimal joint image segmentation and shape matching based on Wasserstein modes. *Journal of Mathematical Imaging and Vision*, 52(3):436–458, 2015. 2

[53] T. Schoenemann and D. Cremers. A combinatorial solution for model-based image segmentation and real-time tracking. *TPAMI*, 32(7):1153–1164, 2010. 2

[54] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. *Symposium on Geometry Processing*, 2007. 3

[55] E. Strekalovskiy, B. Goldlücke, and D. Cremers. Tight convex relaxations for vector-valued labeling problems. *ICCV*, 2011. 2, 4

[56] O. Van Kaick, H. Zhang, G. Hamarneh, and D. Cohen Or. A survey on shape correspondence. In *Computer Graphics Forum*, pages 1681–1707, 2011. 6

[57] S. Varrette, P. Bouvry, H. Cartiaux, and F. Georgatos. Management of an academic hpc cluster: The ul experience. In *Proc. of the 2014 Intl. Conf. on High Performance Computing & Simulation (HPCS 2014)*, 2014. 8

[58] N. Vu and B. S. Manjunath. Shape prior segmentation of multiple objects with graph cuts. In *CVPR*, 2008. 1

[59] T. Werner. A linear programming approach to max-sum problem: A review. *TPAMI*, 29(7):1165–1179, 2007. 2

[60] T. Windheuser, U. Schlickewei, F. R. Schmidt, and D. Cremers. Geometrically consistent elastic matching of 3D shapes: A linear programming solution. *ICCV*, 2011. 6

[61] A. Yershova, S. Jain, S. M. LaValle, and J. C. Mitchell. Generating Uniform Incremental Grids on SO(3) Using the Hopf Fibration. *The International journal of robotics research*, 29(7):801–812, June 2010. 4, 5

[62] C. Zach, C. Hane, and M. Pollefeys. What is optimized in convex relaxations for multilabel problems: Connecting discrete and continuously inspired map inference. *TPAMI*, 36(1):157–170, 2014. 2

[63] S. Zhang, Y. Zhan, M. Dewan, J. Huang, D. N. Metaxas, and X. S. Zhou. Sparse shape composition: A new framework for shape prior modeling. In *CVPR*, 2011. 1