Shape-aware Surface Reconstruction from Sparse 3D Point-Clouds

Florian Bernard[1,2,*], Luis Salamanca[2], Johan Thunberg[2], Alexander Tack[3], Dennis Jentsch[3], Hans Lamecker[3,4], Stefan Zachow[3,4], Frank Hertel[1], Jorge Goncalves[2], Peter Gemmar[2,5]

Abstract

The reconstruction of an object's shape or surface from a set of 3D points plays an important role in medical image analysis, e.g. in anatomy reconstruction from tomographic measurements or in the process of aligning intra-operative navigation and preoperative planning data. In such scenarios, one usually has to deal with *sparse* data, which significantly aggravates the problem of reconstruction. However, medical applications often provide contextual information about the 3D point data that allow to incorporate prior knowledge about the shape that is to be reconstructed. To this end, we propose the use of a statistical shape model (SSM) as a prior for surface reconstruction. The SSM is represented by a point distribution model (PDM), which is associated with a surface mesh. Using the shape distribution that is modelled by the PDM, we formulate the problem of surface reconstruction from a probabilistic perspective based on a Gaussian Mixture Model (GMM). In order to do so, the given points are interpreted as samples of the GMM. By using mixture components with anisotropic covariances that are "oriented" according to the surface normals at the PDM points, a surface-based fitting is accomplished. Estimating the parameters of the GMM in a maximum a posteriori manner yields the reconstruction of the surface from the given data points. We compare our method to the extensively used Iterative Closest Points method on several different anatomical datasets/SSMs (brain, femur, tibia, hip, liver) and demonstrate superior accuracy and robustness on sparse data.

Keywords: Sparse shape reconstruction, statistical shape model, point distribution model, Gaussian mixture model, expected conditional maximisation.

1. Introduction

The reconstruction of an object's shape or surface from a set of 3D points is a highly relevant problem in medical image analysis. It appears for example in image segmentation, where images provide implicit information on the location of anatomical structures via intensity levels, which is frequently converted into geometric information via some kind of feature extraction method. Another scenario is computer-assisted surgery, where a pre-operative therapy plan is transferred to the operating room by means of a navigation system.

In contrast to mere 3D point-clouds that may represent virtually any object, image data of medical objects yield additional contextual information that can be used to adopt prior knowledge about the anatomical structures to be reconstructed. Heckel et al. (2011) make use of the variational interpolation method (Turk and O'Brien, 1999), which essentially uses a general prior on the surface smoothness. Going one step further, Pauly et al. (2005)

and Gal et al. (2007) have considered templates for 3D scan completion that are matched to measurements. However, these methods are limited since the available measurements are assumed to be sufficiently dense (Berger et al., 2014). To tackle this limitation, Bernard et al. (2015) suggested the use of a statistical shape model (SSM) for surface reconstruction. Because the class of anatomical structures is known for clinical routine tasks such as segmentation, registration, or intra-operative navigation, it is possible to use their shapes as geometric priors.

Our main contribution of this work is the introduction of a *surface-based* SSM fitting procedure in order to reconstruct a surface from a sparse 3D point-cloud. Using a point distribution model (PDM) (Cootes and Taylor, 1992), we incorporate a prior into our reconstruction framework that captures the likely shape of the object to be extracted. In doing so, we reformulate the problem of surface reconstruction from a probabilistic perspective, embedding the prior distribution of the SSM parameters that explain plausible shapes into the objective function. Our evaluation considers several different anatomical structures and SSMs (brain, femur, tibia, hip, liver) and sparse data point scenarios, which may occur in different applications, ranging from interactive segmentation to intra-operative registration for navigation. We are able to show superior accuracy and robustness compared to the extensively used Iterative Closest Point (ICP)

---
[*]Corresponding author
[1]Centre Hospitalier de Luxembourg, Luxembourg
[2]Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Esch-sur-Alzette, Luxembourg
[3]Zuse Institute Berlin (ZIB), Germany
[4]1000shapes GmbH, Berlin, Germany
[5]Trier University of Applied Sciences, Trier, Germany

method (Besl and McKay, 1992). Rather than restricting ourselves to particular applications by investigating application-specific aspects, our goal is to demonstrate the *general applicability* of the proposed approach in order to emphasize that the method may be useful in a wide range of settings. The surface-based fitting procedure is achieved by the following methodological contributions:

- By extending existing point-set registration procedures based on Gaussian Mixture Models (GMMs) (Myronenko et al., 2007; Myronenko and Song, 2010; Horaud et al., 2011; Zheng, 2013) such that anisotropic covariances are used in combination with a PDM as transformation model, we obtain a shape-aware surface reconstruction method that is superior to ICP with respect to robustness and accuracy.

- Before, only spherical (isotropic) GMMs accounting for a *point-based* matching have been used (Zheng, 2013; Bernard et al., 2015). We now complement these works by presenting a formulation that is based on anisotropic GMMs that are "oriented" by the surface normals, accounting for a *surface-based* fitting.

- A *rigorous and self-contained derivation* of the surface-based fitting method is presented, leading to an Expected Conditional Maximisation (ECM) algorithm (Meng and Rubin, 1993). ECM shares the same convergence properties as the Expectation Maximisation (EM) method (Dempster et al., 1977) while being more general.

- We develop a fast approximation of the ECM-based fitting method that has the same computational complexity as the spherical GMM-based method. Numerical simulations show that it is less prone to unwanted local optima compared to the original ECM-based method.

This article is organised as follows: section 2 summarises previous research relevant to our methodology. In sections 3 and 4, we introduce our notation and formally state the considered surface reconstruction problem. In addition, we recapitulate the background of PDMs, probabilistic point-set registration, and the Expectation Maximisation method. In section 5 we present our novel shape-aware surface reconstruction method, including a time complexity and convergence analysis. Section 6 comprises experiments conducted using the proposed methods. In section 7 we conclude this work.

2. Related Work

A plethora of methods for general surface reconstruction has been presented in the literature so far (see e.g. Raya and Udupa (1990); Bolle and Vemuri (1991); Herman et al. (1992); Hoppe et al. (1992); Edelsbrunner and Mücke (1994); Bajaj et al. (1995); Amenta et al. (1998);

Bernardini et al. (1999); Treece et al. (2000); Kazhdan et al. (2006); Schroers et al. (2014)). Many of them are summarised and described in the State-of-the-Art Report (STAR) by Berger et al. (2014). In the remainder of the present section, we will discuss only those surface reconstruction methods that go beyond pure smoothness assumptions and make use of more explicit shape priors.

For the completion of 2D shapes, Guo et al. (2012, 2013) incorporate templates from a shape database as (geometric) priors into a Bayesian framework. Similarly, a database of 3D shapes is used by Pauly et al. (2005) for completing 3D surface scans. For increased flexibility compared to static priors, Gal et al. (2007) use a context-specific database of *local* shape priors, where the input data is matched by (dynamically) combining multiple local shape priors into a global prior. As pointed out by Berger et al. (2014), both approaches described by Pauly et al. (2005) and Gal et al. (2007) are limited by the assumption that the point-clouds are assumed to be sufficiently dense. A unified framework for repairing the geometry and texture of meshes has been presented by Park et al. (2006). They employ context-based geometry filling for filling holes in the surface, where available local patches of the mesh are used to fill its missing parts. The task of obtaining high-resolution 3D meshes from low-quality inputs is tackled by Shen et al. (2012) by dynamically assembling object templates from a database of object parts. The 3D shape completion methods discussed above have a common focus on completing (mostly small) missing parts of meshes obtained from 3D scans. However, our interest lies primarily in methods that go beyond patching or improving low-resolution input.

Blanz et al. (2004) have presented a closed-form solution for SSM-based 3D surface reconstruction from a sparse set of points, which relies, however, on the assumption that such a set of points is already in correspondence to the model. Albrecht et al. (2013) introduced posterior shape models that have the objective to model the distribution of a whole shape given only partial information. This method assumes that the corresponding model points of the available partial data are known. In their experiments this issue is either solved manually or using the ICP method. Similarly, for shape prediction from sparse observations, Blanc and Székely (2012) use a variant of ICP that evaluates multiple initialisations. Anguelov et al. (2005) introduced the shape completion and animation of people (SCAPE) method, where one model for pose deformations and one model for shape variations are learned separately. The main objective of this method is the completion of body shapes based on a small number of known positions for some of the model points. Applied to bone models, Rajamani et al. (2007) fit an SSM to a small number of anatomical landmarks that correspond to some of the model points. Baka et al. (2010) fit an SSM to sparse data points that are in correspondence with the model, applied to 2D heart datasets. By producing confidence intervals as output, their method is able to incorporate uncertainties

in the input data. Instead of using a trained SSM, Lu et al. (2011) formulate a low-rank matrix recovery problem for restoring missing parts of objects in archaeological studies. Considering that a set of (incomplete) objects of the same class is available, and that correspondences between common parts are known, their approach is based on the assumption that all shapes are approximately linearly correlated. A shortcoming of the methods discussed so far is that they all *assume known correspondences.* However, if the objects do not exhibit a sufficient amount of distinct features, the identification of exact correspondences is very difficult or even infeasible in practice.

In the literature, there are several methods published addressing this difficulty in (automatically) detecting the correspondence between sparse data points and a model. Due to its simplicity, the ICP algorithm, where correspondences and transformations are estimated in an alternating manner, is a very popular method for the registration of two shapes represented as point sets. Numerous variants of the originally-proposed method have since been developed, e.g. by Rusinkiewicz and Levoy (2001); Granger and Pennec (2002); Segal et al. (2009); Maier-Hein et al. (2012); Bouaziz et al. (2013); Billings et al. (2015). For example, Granger and Pennec (2002) propose the EM-ICP algorithm where hidden variables are used to model unknown correspondences in a surface registration problem. Based on this work, Hufnagel et al. (2008) have proposed a method for learning a PDM from unstructured point-sets. For that, the authors establish probabilistic correspondences first, followed by the computation of the mean shape and the modes of variability. The surface reconstruction method by Stoll et al. (2006) is able to deform a given template to fit point-cloud data. To do so, the user defines initial correspondences between the template and a set of points, which are then refined iteratively in an ICP-like manner. Along the lines of Stoll et al. (2006), for knee surgery, Fleute and Lavallée (1998); Fleute et al. (1999) have presented a methodology to find pose and shape deformation parameters in order to fit an SSM to very sparse data points. Their approach resembles ICP due to the alternating closest point estimation and pose/deformation model parameter updates. Similarly, Chan et al. (2003) reconstruct 3D models of bones in orthopaedic surgery based on a sparse set of points obtained from ultrasound. Here, shapes are repeatedly instantiated using a PDM, where each shape instance is used as input to ICP in order to (rigidly) fit the sparse points. This work has been extended by Barratt et al. (2008), who use a PDM defined on a regular grid instead of the shape surface. Zheng et al. (2006) have proposed a three-stage procedure for sparse shape reconstruction in computer-assisted orthopaedic surgery. In the first stage, the pose of the sparse points is adapted using ICP so that they best fit the mean shape. In the second stage, the shape deformation parameters are estimated for the given correspondences, which are eventually refined using a further deformation based on thin-plate splines. Based on Gaussian Mixture Models with heteroscedas-

tic covariances, Zheng (2013) has proposed a method for deformable shape registration. A promising approach of fitting a hand pose model to points that are sufficiently densely sampled from depth images is presented by Taylor et al. (2014, 2016). The authors formulate a continuous optimisation problem that solves simultaneously for surface-based correspondences and model parameters using a nonlinear sum-of-squares objective.

In the work of Chang and Zwicker (2009), the registration of articulated shapes in two range scans is based on a reduced deformation model defined on a regular grid. In this approach, the deformations are modelled by a convex combination of rigid transformations, where the weights are spatially varying. The registration is performed by alternatingly updating the rigid transformations and their weights, where closest point correspondences are recomputed after each step.

Despite the immense amount of literature on applications and variations of statistical shape models (for an overview the interested reader is referred to the works by Cootes and Taylor (1992); van Ginneken et al. (2002); Heimann and Meinzer (2009)), there have been a limited number of investigations into the use of SSMs for interactive segmentation. In the work of van Ginneken et al. (2003), a user directly manipulates points of the PDM, which has the disadvantage that the user needs to estimate the (unknown) corresponding position of the considered model point in the image. This is particularly difficult if the object does not exhibit distinct features. In their slice-wise SSM-based segmentation of abdominal aortic aneurysms, de Bruijne et al. (2004) initialise the current slice's PDM fitting from the segmentation result of the previous slice, with the option of manually correcting segmentations on a per-slice basis. Liu and Udupa (2009) present Oriented Active Shape Models where the semi-automatic live-wire technique is coupled with an Active Shape Model. Other authors present tools for the posterior correction of model-based segmentations, such as Timinger et al. (2003); Schwarz et al. (2008); Tan and Acharya (2014). van den Hengel et al. (2007) present an interactive procedure based on a set of rules and 2D sketches for completing partial 3D models in structure from motion.

To summarise, existing approaches performing surface reconstruction using SSMs either assume known correspondences or estimate point-based correspondences in an alternating manner. One exception is the work by Taylor et al. (2016), where, for sufficiently dense data, a surface-based hand pose model fitting is performed.

3. Background

In this section we introduce the notation and define PDMs.

3.1. Notation

$\mathbf{1}_p$ and $\mathbf{0}_p$ denote the p-dimensional column vectors containing ones and zeros, respectively. $\mathbf{I}_p$ denotes the $p \times p$

identity matrix and diag($\mathbf{x}$) is the $p \times p$ matrix with the elements of the vector $\mathbf{x} \in \mathbb{R}^p$ on its diagonal. For a matrix $\mathbf{A}$, $\mathbf{A}_{i,j}$ is the (scalar) element in row i and column j. The colon-notation is used to denote all rows or columns, e.g. $\mathbf{A}_{:,j}$ is the j-th column of $\mathbf{A}$. For a matrix $\mathbf{A}$, the concatenation of all columns is given by vec($\mathbf{A}$).

By $\mathfrak{p}(x)$ we denote the *probability density function* (p.d.f.), or *probability mass function* in the discrete case, where x can indicate both a random variable and a realisation of it, depending on the context. $\mathcal{N}(x|\mu, \Sigma)$ denotes the Gaussian p.d.f. with mean μ and covariance matrix Σ.

3.2. Point Distribution Models

Statistical shape models based on PDMs (Cootes and Taylor, 1992) are an established technique to capture non-linear shape deformations of shapes in $\mathbb{R}^3$ from training data by using a linear model in the higher-dimensional shape space. Let $\{\mathbf{X}_k : 1 \leq k \leq K\}$ be the set of K training shapes, where each shape is represented by N points (or vertices) in 3 dimensions given by the rows of the matrix $\mathbf{X}_k \in \mathbb{R}^{N \times 3}$. For processing multiple shapes in a meaningful way, the N vertices of all K shapes have to be in correspondence, i.e. the rows for all $\mathbf{X}_k$ are corresponding to each other.

The PDM is obtained by finding an (affine) subspace close to the subspace spanned by the training shapes, commonly performed by Principal Components Analysis (PCA). For that, we define $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_K] \in \mathbb{R}^{3N \times K}$, with $\mathbf{x}_k = \text{vec}(\mathbf{X}_k) \in \mathbb{R}^{3N}$. This allows to compute the modes of shape variability as the eigenvectors of the sample covariance matrix $\mathbf{C} = \frac{1}{K-1}(\mathbf{X} - \bar{\mathbf{x}}\mathbf{1}_K^T)(\mathbf{X} - \bar{\mathbf{x}}\mathbf{1}_K^T)^T$, where $\bar{\mathbf{x}} = \frac{1}{K}\sum_{k=1}^K \mathbf{x}_k$ is the mean of all shapes in the $3N$-dimensional shape space. Let $\mathbf{\Phi} \in \mathbb{R}^{3N \times M}$ be the matrix of the first M eigenvectors of $\mathbf{C}$ with largest eigenvalues. For $\boldsymbol{\alpha}$ being a variable in $\mathbb{R}^M$, the PDM $\mathbf{y}(\boldsymbol{\alpha}) : \mathbb{R}^M \to \mathbb{R}^{3N}$ is a vector-valued function defined as

$$\mathbf{y}(\boldsymbol{\alpha}) = \bar{\mathbf{x}} + \mathbf{\Phi}\boldsymbol{\alpha}, \tag{1}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^M$ is the shape deformation parameter. The deformation of vertex i through $\boldsymbol{\alpha}$ is denoted by

$$y_i(\boldsymbol{\alpha}) = \bar{x}_i + \mathbf{\Phi}_i\boldsymbol{\alpha}, \tag{2}$$

where the three rows of vertex i are selected from $\bar{\mathbf{x}}$ and $\mathbf{\Phi}$ to obtain $\bar{x}_i \in \mathbb{R}^3$ and $\mathbf{\Phi}_i \in \mathbb{R}^{3 \times M}$.

A common assumption is that $\boldsymbol{\alpha}$ follows a zero-mean Gaussian distribution, i.e. $\mathfrak{p}(\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\alpha}|\mathbf{0}_M, \Lambda)$, where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_M)$ with λ_m being the m-th eigenvalue of $\mathbf{C}$. Thus, thanks to imposing a distribution upon $\boldsymbol{\alpha}$, we obtain a distribution over shapes (Albrecht et al., 2013).

Usually, in addition to the PDM accounting for shape deformations, a rigid transformation is employed in order to account for the absolute pose of the shape $\mathbf{y}(\boldsymbol{\alpha})$ with respect to some reference (e.g. the image coordinate system). Here we assume that the predominant part of the pose has already been normalised. Minor pose variations can be (approximately) captured implicitly in the PDM.

4. Problem Formulation

Given is a PDM that serves as prior for plausible shapes. Also, we assume that (for fixed $\boldsymbol{\alpha}$) the PDM points are vertices of an oriented triangular surface mesh $\mathcal{M}$. Additionally, we are given the set $\mathcal{P} = \{p_j : 1 \leq j \leq P\}$ of P surface points of the shape that is to be reconstructed, where $\mathcal{P}$ is sparse in the sense that it only contains few points p_j lying on the object's surface. The objective is to find the deformation parameter $\boldsymbol{\alpha}$ such that $\mathbf{y}(\boldsymbol{\alpha})$ "best agrees" with the points in $\mathcal{P}$.

4.1. A Generative Model

Our work is motivated by the widely used Coherent Point Drift (CPD) approach for general point-set registration (Myronenko et al., 2007; Myronenko and Song, 2010). It also resembles the approach by Zheng (2013) for deformable shape registration, which however is based on heteroscedastic GMMs with isotropic covariances, and it is related to the EM-ICP algorithm (Granger and Pennec, 2002) since we also make use of hidden variables to model the unknown correspondences.

In the following we present the probabilistic model, where a distribution is imposed over each vertex i by using a GMM. Given the set $\mathcal{P}$ and a PDM, we consider the following assumptions:

1. For $i = 1, \ldots, N$, each PDM vertex position $y_i(\boldsymbol{\alpha}) \in \mathbb{R}^3$ is considered as the mean of a 3D Gaussian distribution with covariance $\mathbf{\Sigma}_i$.

2. Each point $p_j \in \mathcal{P}$ can be uniquely mapped to one specific vertex index i, its *generating component*, from whose distribution it is drawn.

As such, each point p_j follows the distribution

$$\mathfrak{p}(p_j|i, \boldsymbol{\alpha}, \mathbf{\Sigma}_i) = \mathcal{N}(p_j|y_i(\boldsymbol{\alpha}), \mathbf{\Sigma}_i), \tag{3}$$

where all p_j for $j = 1, \ldots, P$ are independent. The corresponding graphical model is depicted in Fig. 1. Note that all of the N Gaussian components are parametrised by $\boldsymbol{\alpha}$ and the covariances $\hat{\mathbf{\Sigma}} := \{\mathbf{\Sigma}_i\}$. We assume that each component is chosen with equal probability, i.e. $\mathfrak{p}(i) = \frac{1}{N}$.

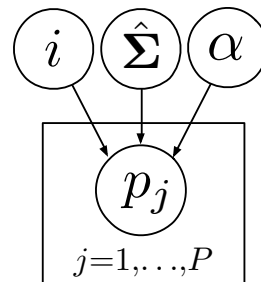Figure 1: Graphical model of the generating process of $\mathcal{P}$.

Our objective is to find the parameters $\boldsymbol{\alpha}$ and $\hat{\mathbf{\Sigma}}$ that are most likely to have generated the points $\mathcal{P}$. Since the

generating component i_j of p_j is unknown, the indices of the generating components are treated as *latent* variables. To incorporate this uncertainty, we consider a GMM for the distribution of p_j, leading to

$$\mathfrak{p}(p_j|\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}}) = \sum_{i=1}^{N} \mathfrak{p}(i)\, \mathfrak{p}(p_j|i,\boldsymbol{\alpha},\boldsymbol{\Sigma}_i) \qquad (4)$$

$$= \frac{1}{N} \sum_{i=1}^{N} \mathcal{N}(p_j|y_i(\boldsymbol{\alpha}),\boldsymbol{\Sigma}_i)\,. \qquad (5)$$

Using Bayes' theorem, one can derive the probability that the i-th mixture component has generated the point p_j, given also $\boldsymbol{\alpha}$ and $\hat{\boldsymbol{\Sigma}}$, as

$$\mathfrak{p}(i|p_j,\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}}) = \frac{\mathfrak{p}(p_j|i,\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}})\mathfrak{p}(i)\mathfrak{p}(\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}})}{\sum_{i'} \mathfrak{p}(p_j|i',\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}})\mathfrak{p}(i')\mathfrak{p}(\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}})} \qquad (6)$$

$$= \frac{\mathfrak{p}(p_j|i,\boldsymbol{\alpha},\boldsymbol{\Sigma}_i)\mathfrak{p}(i)}{\sum_{i'} \mathfrak{p}(p_j|i',\boldsymbol{\alpha},\boldsymbol{\Sigma}_{i'})\mathfrak{p}(i')} \qquad (7)$$

$$= \frac{\mathcal{N}(p_j|y_i(\boldsymbol{\alpha}),\boldsymbol{\Sigma}_i)}{\sum_{i'} \mathcal{N}(p_j|y_{i'}(\boldsymbol{\alpha}),\boldsymbol{\Sigma}_{i'})}\,. \qquad (8)$$

4.2. Optimisation using EM

If the generating component i_j of p_j is unknown, according to eq. (5) all points $p_j \in \mathcal{P}$ are independent and identically distributed (i.i.d.). Thus, the log-likelihood as a function of the model parameters $\boldsymbol{\theta} := (\boldsymbol{\alpha},\hat{\boldsymbol{\Sigma}})$ reads

$$L(\boldsymbol{\theta}) = \ln \mathfrak{p}(\mathcal{P}|\boldsymbol{\theta}) = \ln \prod_{j=1}^{P} \sum_{i=1}^{N} \mathfrak{p}(i)\mathfrak{p}(p_j|i,\boldsymbol{\theta}) \qquad (9)$$

$$= \sum_{j=1}^{P} \ln \sum_{i=1}^{N} \mathfrak{p}(i)\mathfrak{p}(p_j|i,\boldsymbol{\theta})\,.$$

The maximisation of L w.r.t. $\boldsymbol{\theta}$ cannot be solved readily due to the sum appearing inside the logarithm. Therefore, a common approach is to employ an iterative method for the maximisation. We denote the estimate of $\boldsymbol{\theta}$ at iteration n as $\boldsymbol{\theta}^{(n)}$ and rewrite eq. (9) as

$$L(\boldsymbol{\theta},\boldsymbol{\theta}^{(n)}) = \sum_{j=1}^{P} \ln \sum_{i=1}^{N} \mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)}) \frac{\mathfrak{p}(i)\mathfrak{p}(p_j|i,\boldsymbol{\theta})}{\mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)})}\,. \quad (10)$$

Applying Jensen's inequality (Jensen, 1906) leads to

$$L(\boldsymbol{\theta},\boldsymbol{\theta}^{(n)}) \geq \qquad (11)$$

$$\sum_{j=1}^{P} \sum_{i=1}^{N} \mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)}) \ln \frac{\mathfrak{p}(i)\mathfrak{p}(p_j|i,\boldsymbol{\theta})}{\mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)})} =: Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(n)})\,.$$

As such, the right-hand side of eq. (11), which we denote by $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(n)})$, is a lower bound for $L(\boldsymbol{\theta},\boldsymbol{\theta}^{(n)})$. Maximising this lower bound is the key idea of the EM algorithm. In the E-step, the probabilities $\mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)})$ are evaluated for fixed $\boldsymbol{\theta}^{(n)}$ by using eq. (8). Then, in the M-step, Q in eq. (11) is maximised w.r.t. $\boldsymbol{\theta}$ for the fixed $\mathfrak{p}(i|p_j,\boldsymbol{\theta}^{(n)})$ computed before.

5. Methods

In section 5.1 below, we present the main novelty of this paper, the anisotropic GMM-based fitting approach that employs covariance matrices that "oriented" according to the surface normals at the PDM points. This allows to move from a purely point-based matching (Myronenko et al., 2007; Myronenko and Song, 2010; Zheng, 2013; Bernard et al., 2015) to a more surface-based fitting. Subsequently, we describe a fast approximation of the anisotropic GMM method. By using an extension to this approximation, one can ensure that it is an instance of the Generalised Expectation Maximisation (GEM) method and thus the convergence is guaranteed.

5.1. Surface Reconstruction using an Anisotropic GMM

When using spherical covariances for each of the N Gaussian components, a purely point-based fitting is conducted. However, in a vast amount of medical applications of SSMs, the points of the PDM represent the vertices of a *surface mesh*. This surface mesh is in general only an approximation of a *continuous surface*. Whilst the sparse points $\mathcal{P}$ are assumed to lie on this continuous surface, in general they do not coincide with the PDM vertices. Hence, matching the *surface*, depending on the PDM deformation parameter $\boldsymbol{\alpha}$, is more appropriate. A didactic 2D example is presented in Fig. 2.

5.1.1. Surface-aligned Covariance Matrices

We now formalise our surface-based fitting method using a GMM with anisotropic covariance matrices. In the GMM, the covariance matrix of each component i, i.e. each vertex of the PDM, is defined as

$$\boldsymbol{\Sigma}_i(\sigma,\boldsymbol{\alpha}) := \sigma^2 \mathbf{C}_i(\boldsymbol{\alpha})\,. \qquad (12)$$

The scalar parameter σ^2 can be seen as a global scaling factor, whereas the matrix $\mathbf{C}_i(\boldsymbol{\alpha})$ models the anisotropy of the surface structure locally by using a larger variance in the directions of vectors lying in the tangent plane of the PDM surface, compared to the variance along the PDM normal direction (cf. Fig. 2 (c)).

Assuming that the surface mesh $\mathcal{M}$ of the underlying shape of the PDM is given in the form of oriented triangles (cf. section 4), with i_2 and i_3 we denote the index of the "left" and "right" neighbour vertex of i, respectively. With that, the surface normal at vertex i is given by

$$n_i(\boldsymbol{\alpha}) = \frac{(y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})) \times (y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha}))}{\|(y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})) \times (y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha}))\|}\,. \quad (13)$$

The matrix $\mathbf{C}_i$ is defined as

$$\mathbf{C}_i(\boldsymbol{\alpha}) := (\frac{1}{\eta} - 1)n_i(\boldsymbol{\alpha})n_i^T(\boldsymbol{\alpha}) + \mathbf{I}_3\,, \qquad (14)$$

where the parameter $\eta \geq 1$ weights the variances of vectors along the normal direction compared to the tangential direction (note that we use the same value of η for
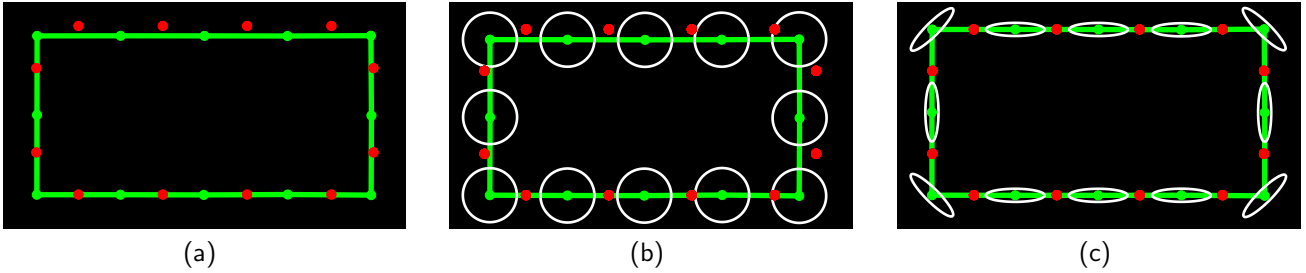
Figure 2: Anisotropic covariance matrices to achieve a surface-based fitting. The sparse points $\mathcal{P}$ are shown in red, the PDM of a rectangle is shown in green, where the green points define the PDM vertices ($N = 12, M = 2, P = 12$). The orientation of the covariance matrices is shown as white ellipses. The objective is to deform the rectangle PDM such that it fits the red points by adjusting $\boldsymbol{\alpha}$. The initialisation is shown in (a). Since the red points are sampled between the PDM points (cf. *shape approximation problem* (Hill et al., 1995)), using spherical covariance matrices results in a fit that is even worse than the initialisation (b), whereas using anisotropic covariances results in a more accurate fit than the initialisation (c).

all points). A motivation for eq. (14) can be found in the work of Hill et al. (1995). For the covariance matrix $\boldsymbol{\Sigma}_i(\sigma, \boldsymbol{\alpha})$, the variance along the normal $n_i(\boldsymbol{\alpha})$ is given by $\frac{\sigma^2}{\eta}$, and the variance in the direction of any vector in the tangent plane is σ^2. As such, for $\eta = 1$ one obtains an isotropic GMM fitting, in analogy to the CPD algorithm (Myronenko et al., 2007; Myronenko and Song, 2010). We refer to the isotropic method with η set to 1 as *ISO*. The two main differences between CPD and our approach are that in our case the transformations are parametrised by a PDM, and that we also allow for anisotropic covariances. Using transformations that are parametrised by a PDM has also been done by Zheng (2013) for deformable shape registration. Choosing $\eta > 1$ achieves the desired behaviour of modelling a larger variance in the tangent plane. Note that for $\eta > 0$, the matrix $\mathbf{C}_i(\boldsymbol{\alpha})$ is symmetric and positive definite with the spectrum $\{\frac{1}{\eta}, 1, 1\}$. The inverse of $\mathbf{C}_i$, the *precision matrix*, is

$$\mathbf{W}_i(\boldsymbol{\alpha}) := \mathbf{C}_i^{-1}(\boldsymbol{\alpha}) = (\eta - 1)n_i(\boldsymbol{\alpha})n_i^T(\boldsymbol{\alpha}) + \mathbf{I}_3 \,. \quad (15)$$

5.1.2. Maximum A Posteriori (MAP) Solution

We can cast the log-likelihood from eq. (9) into a Bayesian view, leading to

$$L^{\text{posterior}}(\boldsymbol{\theta}) = \ln[\mathfrak{p}(\mathcal{P}|\boldsymbol{\theta})\mathfrak{p}(\boldsymbol{\theta})] \,, \quad (16)$$

where additional knowledge in the form of the prior distribution $\mathfrak{p}(\boldsymbol{\theta})$ is incorporated. By assuming $\mathfrak{p}(\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\alpha}|\mathbf{0}_M, \Lambda)$ as described in section 3.2, and choosing a uniform prior for $\mathfrak{p}(\sigma)$, the prior distribution results in $\mathfrak{p}(\boldsymbol{\theta}) \propto \mathfrak{p}(\boldsymbol{\alpha})$. The Q-function for the MAP solution reads

$$Q(\boldsymbol{\alpha}, \sigma, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) = \quad (17)$$
$$\text{const} - \frac{1}{2}\boldsymbol{\alpha}^T \Lambda^{-1} \boldsymbol{\alpha} - \frac{3P}{2} \ln \sigma^2$$
$$- \frac{1}{2\sigma^2} \sum_{i,j} \mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) \cdot$$
$$(p_j - y_i(\boldsymbol{\alpha}))^T \mathbf{W}_i(\boldsymbol{\alpha})(p_j - y_i(\boldsymbol{\alpha})) \,.$$

As already described, the E-step is solved by evaluating eq. (8). Then, the M-step comprises maximising Q in eq. (17) w.r.t. $\boldsymbol{\alpha}$ and σ. Due to the dependence of $\mathbf{W}_i$ on $\boldsymbol{\alpha}$ (for $\eta \neq 1$), finding $\boldsymbol{\alpha}$ that maximises Q does not admit a simple closed-form solution. This is in contrast to the isotropic case ($\eta = 1$), where $\boldsymbol{\alpha}$ that maximises Q can be found by solving a linear system of equations. Instead, $\boldsymbol{\alpha}$ is now obtained using the BFGS quasi-Newton method (Nocedal and Wright, 2006). The idea is to start with the old value $\boldsymbol{\alpha}^{(n)}$, and then iteratively move along directions that increase Q. Whilst the ordinary Newton method requires the gradient of Q as well as its Hessian, the BFGS quasi-Newton method uses an approximation of the Hessian that is cheap to compute. A derivation of the gradient $\nabla_{\boldsymbol{\alpha}} Q$ of Q w.r.t. $\boldsymbol{\alpha}$ can be found in Appendix A.

In order to obtain $\boldsymbol{\alpha}$, one option is to run the BFGS quasi-Newton procedure until convergence, where one obtains an $\boldsymbol{\alpha}$ that (locally) maximises Q. With that, the updates of σ on $\boldsymbol{\alpha}$ depend on each other and the procedure reverts to the ECM algorithm (Meng and Rubin, 1993). An alternative is to run only a single quasi-Newton step in each M-step. With that, the obtained $\boldsymbol{\alpha}$ is not a local maximiser of Q; however, one still has the guarantee that Q is non-decreasing. As such, this procedure reverts to the GEM algorithm (Dempster et al., 1977).

Finally, the σ-update for fixed $\boldsymbol{\alpha}$ is given by

$$\sigma^2 = \frac{1}{3P} \sum_{i,j} \mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) \cdot \quad (18)$$
$$(p_j - y_i(\boldsymbol{\alpha}))^T \mathbf{W}_i(\boldsymbol{\alpha})(p_j - y_i(\boldsymbol{\alpha})) \,.$$

The pseudocode of the anisotropic GMM fitting procedure is presented in Algorithm 1.

5.1.3. Fast Approximate Anisotropic GMM

Now we introduce an approximation of the $\boldsymbol{\alpha}$-update that is a much faster alternative to the quasi-Newton method. The main idea is to use the previous value $\boldsymbol{\alpha}^{(n)}$ instead of $\boldsymbol{\alpha}$ for computing the anisotropic covariance matrices $\mathbf{C}_i(\boldsymbol{\alpha}^{(n)})$ during the $\boldsymbol{\alpha}$-update in the M-

6

Input: $\bar{x}, \mathbf{\Phi}, \mathcal{P}, \eta, \mathcal{M}$
Output: $\boldsymbol{\alpha}, \sigma^2$
Initialise: $\boldsymbol{\alpha} = 0$, $\sigma^2 = \frac{1}{3NP}\sum_{i,j}\|p_j - \bar{x}_i\|^2$, $\mathbf{P} \in \mathbb{R}^{P \times N}$,
$\quad\quad\quad\quad\quad \mathbf{y} = \bar{x} + \mathbf{\Phi}\boldsymbol{\alpha}$

1 **foreach** $i = 1, \ldots, N$ **do**
$\quad$ // compute $\mathbf{W}_i$
2 $\quad$ $n_i = \frac{(y_{i_2} - y_i) \times (y_{i_3} - y_i)}{\|(y_{i_2} - y_i) \times (y_{i_3} - y_i)\|}$
3 $\quad$ $\mathbf{W}_i = (\eta - 1)n_i n_i^T + \mathbf{I}_3$
4 **repeat**
$\quad$ // E-step
5 $\quad$ **foreach** $j = 1, \ldots, P$ **do**
6 $\quad\quad$ $t = 0$
7 $\quad\quad$ **foreach** $i = 1, \ldots, N$ **do**
8 $\quad\quad\quad$ $\mathbf{P}_{ji} = \exp(-\frac{1}{2\sigma^2}(p_j - y_i)^T \mathbf{W}_i(p_j - y_i))$
9 $\quad\quad\quad$ $t = t + \mathbf{P}_{ji}$
10 $\quad\quad$ $\mathbf{P}_{j,:} = \frac{1}{t}\mathbf{P}_{j,:}$
$\quad$ // M-step
11 $\quad$ $\boldsymbol{\alpha} = \text{quasi-Newton}(Q, \nabla Q, \boldsymbol{\alpha})$
12 $\quad$ $\mathbf{y} = \bar{x} + \mathbf{\Phi}\boldsymbol{\alpha}$
13 $\quad$ **foreach** $i = 1, \ldots, N$ **do**
$\quad$ // compute $\mathbf{W}_i$
14 $\quad\quad$ $n_i = \frac{(y_{i_2} - y_i) \times (y_{i_3} - y_i)}{\|(y_{i_2} - y_i) \times (y_{i_3} - y_i)\|}$
15 $\quad\quad$ $\mathbf{W}_i = (\eta - 1)n_i n_i^T + \mathbf{I}_3$
16 $\quad$ $\sigma^2 = \frac{1}{3P}\sum_{i,j}\mathbf{P}_{ji}(p_j - y_i)^T \mathbf{W}_i(p_j - y_i)$
17 **until** *convergence*

Algorithm 1: Pseudocode of the anisotropic GMM fitting method. The notation "quasi-Newton$(Q, \nabla Q, \boldsymbol{\alpha})$" denotes running the quasi-Newton method for maximising Q w.r.t. $\boldsymbol{\alpha}$, where ∇Q is its gradient and the third argument is the initial value of $\boldsymbol{\alpha}$. If the GEM approach is used, the quasi-Newton method is run only for a single iteration. Note that the surface mesh $\mathcal{M}$ is used for the normal computations.

step. Our key assumption is that the PDM is *well-behaved* in the sense that neighbouring vertices vary smoothly during deformation; thus, *locally* the deformation of an individual triangle is nearly a translation. Since surface normals are *invariant* to translations it follows that $\|n_i(\boldsymbol{\alpha}) - n_i(\boldsymbol{\alpha}^{(n)})\|$ is small, which implies that $\|W_i(\boldsymbol{\alpha}) - W_i(\boldsymbol{\alpha}^{(n)})\|$ is also small.

The resulting Q-function using the proposed approximation is now given by

$$\tilde{Q}(\boldsymbol{\alpha}, \sigma, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) = \tag{19}$$
$$\text{const} - \frac{1}{2}\boldsymbol{\alpha}^T \Lambda^{-1}\boldsymbol{\alpha} - \frac{3P}{2}\ln \sigma^2$$
$$- \frac{1}{2\sigma^2}\sum_{i,j}\mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) \cdot$$
$$(p_j - y_i(\boldsymbol{\alpha}))^T \mathbf{W}_i(\boldsymbol{\alpha}^{(n)})(p_j - y_i(\boldsymbol{\alpha})),$$

where the difference to Q in eq. (17) is that the constant $\mathbf{W}_i(\boldsymbol{\alpha}^{(n)})$ is now used in place of the function $\mathbf{W}_i(\boldsymbol{\alpha})$. As such, the $\boldsymbol{\alpha}$-update in the M-step is a quadratic concave problem that can be maximised efficiently. The solution for $\boldsymbol{\alpha}$ is found by solving the linear system $\mathbf{A}\boldsymbol{\alpha} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{M \times M}$ is given by

$$\mathbf{A} = \sigma^2 \Lambda^{-1} + \sum_{i,j}\mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)})\mathbf{\Phi}_i^T \mathbf{W}_i(\boldsymbol{\alpha}^{(n)})\mathbf{\Phi}_i,$$
$$\tag{20}$$

and $\mathbf{b} \in \mathbb{R}^M$ by

$$\mathbf{b} = \sum_{i,j}\mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)})\mathbf{\Phi}_i^T \mathbf{W}_i(\boldsymbol{\alpha}^{(n)})(p_j - \bar{x}_i). \tag{21}$$

The pseudocode for this approximate method is similar to Algorithm 1, except for line 11, where $\boldsymbol{\alpha}$ is computed by solving a linear system.

In order to guarantee that the approximate method converges, it is necessary that in the M-step the value of the *exact* Q in eq. (17) is non-decreasing, i.e. the new $\boldsymbol{\alpha} = \mathbf{A}^{-1}\mathbf{b}$ obtained using $\tilde{Q}$ must fulfil

$$Q(\mathbf{A}^{-1}\mathbf{b}, \sigma^{(n)}, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) \geq Q(\boldsymbol{\alpha}^{(n)}, \sigma^{(n)}, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}). \tag{22}$$

For $\eta = 1$ the methods reverts to the isotropic case and condition (22) vacuously holds. However, for $\eta > 1$ this is not true in general. One way to ensure that Q is non-decreasing is to evaluate the condition in eq. (22) in each iteration, and, in the case of a violation, revert to one of the quasi-Newton methods for the $\boldsymbol{\alpha}$-update. Specifically, we consider a single quasi-Newton step for updating $\boldsymbol{\alpha}$. We denote the approximate method without this convergence check as *ANISO*, and the method with the convergence check and the quasi-Newton step as fall-back as *ANISOc*.

In Fig. 3 we illustrate the behaviour of Q and compare it with $\tilde{Q}$ for various choices of η. Note that for visualisation purposes we have chosen $M = 2$, whereas in higher-dimensional cases the effect of an increasing η on the non-concavity can be expected to be more severe.

5.2. Performance Analysis

Table 1 summarises the computational complexity of the presented methods. In Fig. 4 we plot the mean of the normalised value of Q as a function of the processing time for all four anisotropic fitting methods. For each random run we sample a shape instance by drawing $\boldsymbol{\alpha}$ (cf. section 3.2), select P points randomly from the mesh surface (cf. section 6), and run the four methods. The obtained Q for the four methods are then normalised such that in each run the smallest Q corresponds to 0 and the largest Q corresponds to 1 (normalisation w.r.t to all four methods simultaneously). We have found that the single-step quasi-Newton method (GEM) is faster compared to the full quasi-Newton procedure (ECM). Moreover, compared to both quasi-Newton methods, the approximate methods are much faster. Since the ANISOc method makes use of elements both of the GEM and the ANISO method, the total time complexity of the ANISOc method is the combined time complexity for GEM and ANISO (cf. Table 1). Nevertheless, in our simulations the ANISOc method comes close to the ANISO method in terms of convergence speed. This is because in the early stages of the iterative procedure the ANISOc method satisfies condition (22) in most cases. A violation of (22) happens more frequently in the later stages. Since these results suggest that the ANISO
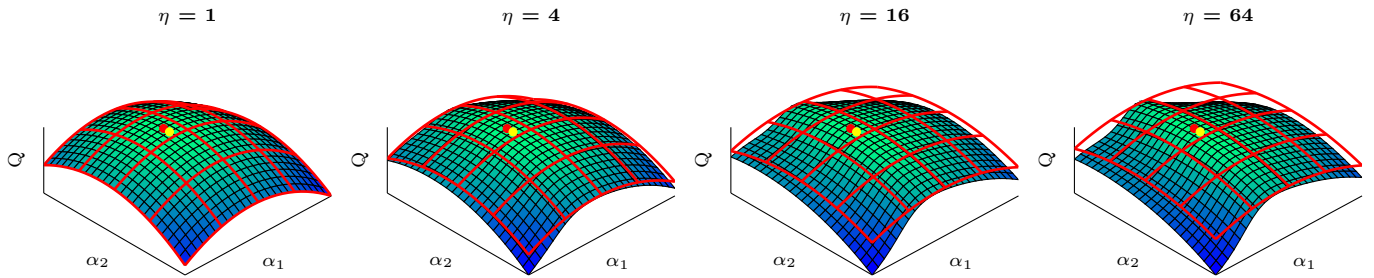
Figure 3: Illustration of the behaviour of Q for various η. The height and the colour of the surface both show the value of Q, eq. (17), depending on $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$. The red grid shows its concave approximation $\tilde{Q}$ as presented in eq. (19). The red dot denotes the value of Q at $\boldsymbol{\alpha}^{(n)}$; at this position $Q = \tilde{Q}$. The yellow dot indicates the maximum of Q. For the trivial case of $\eta = 1$ it can be seen that $Q = \tilde{Q}$ everywhere, whereas an increasing η leads to a larger discrepancy between Q and $\tilde{Q}$ as well as to an "increased non-concavity" of Q.

6. Experiments

In this section we evaluate the proposed fitting procedures on five datasets with the parameters being shown in Table 2. For the generation of the set of sparse points $\mathcal{P}$, we sample sparse points randomly on the shape *surfaces*. To do so, we first select a triangle from the surface mesh with a probability proportional to its area. Then, we uniformly sample a point lying within the triangle according to the procedure presented by Osada et al. (2002). Moreover, in order to evaluate how well our method is able to cope with uncertainties in the given points, we considered noisy versions of these points by adding spherical Gaussian noise with covariance $\tilde{\sigma}^2 \mathbf{I}_3$ to each point individually. For each experiment we have taken the scale of the object in the shape model into account for choosing $\tilde{\sigma}$. The considered values of $\tilde{\sigma}$ are presented in Table 2.

Table 2: Summary of parameters for the datasets. N is the number of points in the PDM ("ds" denotes the downsampled PDM), K the number of training shapes, M the number of modes of variation for leave-all-in (LAI) and leave-one-out (LOO) experiments, η the anisotropy parameter, and $\tilde{\sigma}$ the standard deviation of the noise added to the points.

	N	N (ds)	K	M (LAI)	M (LOO)	η	$\tilde{\sigma}$
brain	1792	371	17	16	96 (kPCA)	8	2 mm
femur	3800	759	60	59	58	4	5 mm
tibia	4701	814	60	59	58	4	5 mm
hip	5603	1120	48	47	46	4	5 mm
liver	4542	908	112	111	110	2	1 cm

In addition to the proposed probabilistic fitting methods presented in this paper, we also evaluate two non-probabilistic ICP approaches. The first approach is the regularised isotropic ICP method as outlined in Algorithm 2, which we denote *ICP* in the experiments. The second approach is an anisotropic version thereof, which we denote *AICP* in the experiments. Similarly to Maier-Hein et al. (2010), for the anisotropic ICP we compute the nearest neighbours in the third line of Algorithm 2 by

Table 1: Computational complexity table. The complexity of the $\boldsymbol{\alpha}$-update for one iteration of the BFGS quasi-Newton methods is $\mathcal{O}(M^2)$ plus the complexity of the evaluation of Q and ∇Q (Nocedal and Wright, 2006) (we use n to denote the number of iterations of the quasi-Newton method). The complexity of the $\boldsymbol{\alpha}$-update of the remaining methods comprises the computation of $\mathbf{A}$ and $\mathbf{b}$, as well as solving a linear system of equations of size $M \times M$, for which we present the complexity $\mathcal{O}(M^3)$ due to the matrix inversion involved. Note that in $\star$ we present the complexity for general Λ, for diagonal Λ the quadratic time complexity in M reduces to linear complexity.

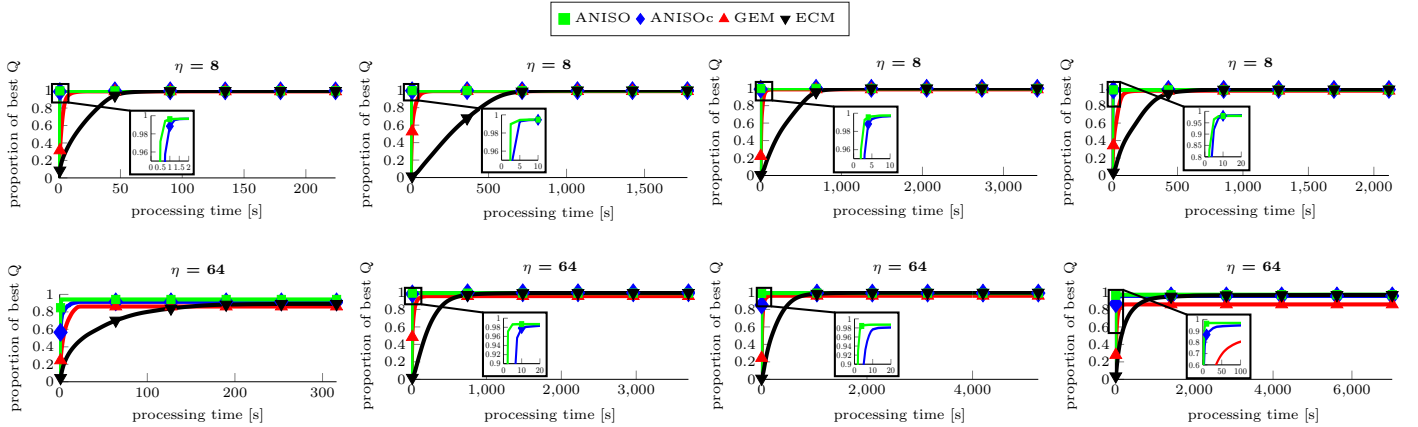|		anisotropic				isotropic
		ECM	GEM	ANISOc	ANISO	ISO
update $\mathbf{y}$		$\mathcal{O}(MN)$				
compute $\{\mathbf{W}_i\}$		$\mathcal{O}(N)$				-
E-step		$\mathcal{O}(NP)$				
$\boldsymbol{\alpha}$-update	evaluate Q, eq. (17)	$\mathcal{O}(MN + M^2 + NP)^\star$			-	
	evaluate ∇Q, eq. (A.4)	$\mathcal{O}(M^2 + MNP)^\star$			-	
	construct $\mathbf{A}$	-		$\mathcal{O}(M^2N + NP)$		
	construct $\mathbf{b}$	-		$\mathcal{O}(MN + NP)$		
	total $\boldsymbol{\alpha}$-update	$\mathcal{O}(n(M^2 + MNP))$	$\mathcal{O}(M^2 + MNP)$	$\mathcal{O}(M^3 + MNP + M^2N)$		$\mathcal{O}(M^3 + M^2N + NP)$
σ-update		$\mathcal{O}(NP)$				
total	(per outer iteration)	$\mathcal{O}(n(M^2 + MNP))$	$\mathcal{O}(M^2 + MNP)$	$\mathcal{O}(M^3 + MNP + M^2N)$		$\mathcal{O}(M^3 + M^2N + NP)$

Figure 4: Proportion of best value of Q versus processing time averaged over 100 random runs for the four anisotropic methods for two choices of η (in the rows). In each column a different dataset has been used to produced the results, from left to right we show results produced by the brain shapes dataset ($N = 1792, M = 16, P = 30$, cf. section 6.4), the femur dataset ($N = 3800, M = 59, P = 30$, cf. section 6.1), the tibia dataset ($N = 4071, M = 59, P = 30$, cf. section 6.1), and the hip dataset ($N = 5603, M = 47, P = 30$, cf. section 6.3).

Input: $\bar{\mathbf{x}}, \mathbf{\Phi}, \mathcal{P}, \Lambda$
Output: $\boldsymbol{\alpha}$
Initialise: $\boldsymbol{\alpha} = \mathbf{0}$, $\mathbf{p} = \text{vec}([p_1, \ldots, p_P]) \in \mathbb{R}^{3P}$
1 **repeat**
2 $\quad$ $\mathbf{y} = \bar{\mathbf{x}} + \mathbf{\Phi}\boldsymbol{\alpha}$
$\quad$ // Find nearest neighbours
3 $\quad$ $\mathcal{N} = \text{findNearestNeighbourIndices}(\mathbf{y}, \mathcal{P})$
$\quad$ // Solve linear system for $\boldsymbol{\alpha}$
4 $\quad$ $\mathbf{A} = \mathbf{\Phi}_{\mathcal{N},:}$
5 $\quad$ $\mathbf{b} = \mathbf{p} - \bar{\mathbf{x}}_{\mathcal{N}}$
6 $\quad$ $\boldsymbol{\alpha} = (\mathbf{A}^T\mathbf{A} + \Lambda^{-1})^{-1}\mathbf{A}^T\mathbf{b}$ $\quad\quad$ // Tikhonov regularisation
7 **until** *convergence*

Algorithm 2: Pseudocode of the ICP baseline method. The notation $\mathbf{\Phi}_{\mathcal{N},:}$ and $\bar{\mathbf{x}}_{\mathcal{N}}$ means selecting the appropriate rows from $\mathbf{\Phi}$ and $\bar{\mathbf{x}}$ according to the indices of the nearest neighbours $\mathcal{N}$.

taking the anisotropic covariance matrices (eq. (14)) into account. For both, ICP and AICP, the regulariser Λ corresponds to the covariance matrix of the PDM parameter $\boldsymbol{\alpha}$ (cf. section 3.2). Moreover, in our evaluation we compare the ground truth data to the mean shape, i.e. in this setting we do not run any fitting procedure at all, which amounts to setting $\boldsymbol{\alpha} = \mathbf{0}$.

The anisotropic method requires to set the parameter η accounting for the amount of anisotropy. We have manually chosen the values of η for each dataset, as shown in Table 2. We have empirically found that for an increasing amount of uncertainty $\tilde{\sigma}$ in the sparse points, it is advantageous to use a lower value of η.

We consider leave-all-in (LAI) and leave-one-out (LOO) experiments. The LAI experiments measure the performance of our method given a perfect model, whereas the LOO experiments evaluate the generalisation ability to unseen data.

We use the Dice Similarity Coefficient (DSC) as volumetric overlap measure, which is defined as

$$DSC(\mathcal{V}_x, \mathcal{V}_y) = \frac{2|\mathcal{V}_x \cap \mathcal{V}_y|}{|\mathcal{V}_x| + |\mathcal{V}_y|} \quad (23)$$

for the volumetric segmentations $\mathcal{V}_x$ and $\mathcal{V}_y$. Additional results considering surface-based measures for all datasets are presented in the supplementary material.

6.1. Knee Bones: Femur and Tibia

For the femur and tibia datasets we assumed that the pose has already been normalised and we worked directly in the space of the SSM. In practice, this can for example be tackled in a similar manner as by Seim et al. (2010), who proposed an automated SSM-based knee bone segmentation, where initially the model is positioned inside the three dimensional CT or MR image via Generalised Hough Transform (Ballard, 1981).

We present experiments for a PDM of the femur with $N = 3800$ points (cf. Fig. 5 (a)) and a PDM of the tibia with $N = 4701$ points (cf. Fig. 5 (c)). Additionally, we evaluated the ANISO method using the downsampled PDMs, denoted ANISO-ds, where only a subset of the original PDM vertices are used (cf. Fig. 5 (b) and Fig. 5 (d)). Random sparse points are generated according to the procedure described in section 6, where for each training shape 10 instances of sparse points $\mathcal{P}$ are sampled. In Fig. 5 such random instances of $\mathcal{P}$ are shown for the mean shapes of both bones. Summaries of the results are shown in Fig. 6 for the femur and in Fig. 7 for the tibia.

It can be seen that for both bone PDMs if only $P = 9$ points are available, the ICP methods perform slightly better than the ANISO method. Once more points become available, the ANISO method outperforms the ICP method. Surprisingly, the ANISO-ds method, which uses a downsampled PDM, outperforms the ANISO method for $P = 9$. We assume that this is because the original PDMs, comprising $N = 3800$ vertices for the femur and $N = 4701$ vertices for the tibia, contain fine local details that lead to an overfitting when reconstructing the surface from only $P = 9$ points. In contrast, the downsampled PDM contains less details that may impede the surface reconstruc-
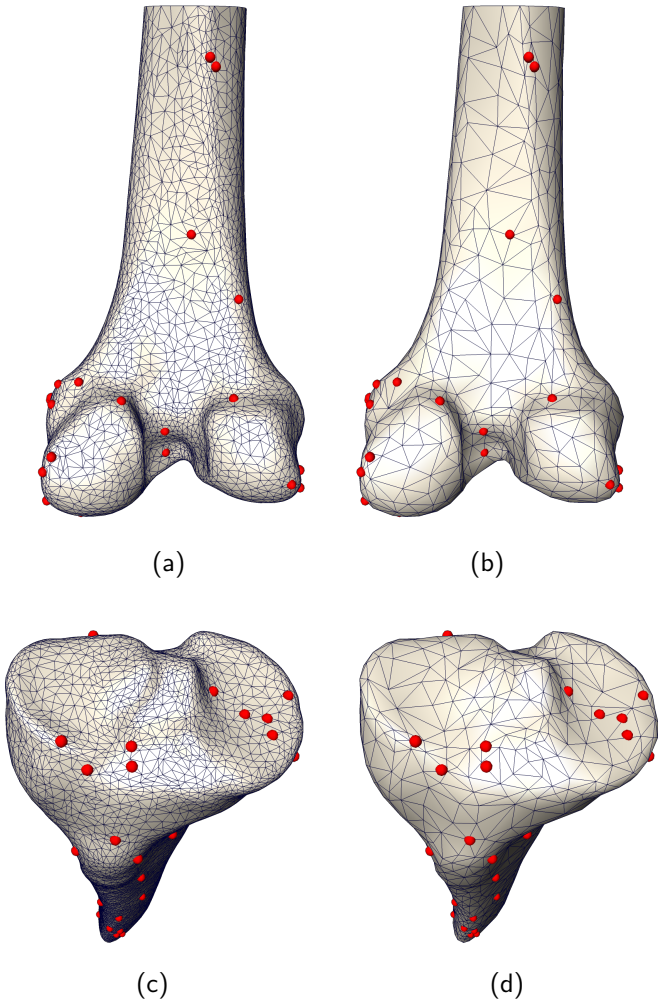
Figure 5: Femur and tibia datasets. (a) Femur mean shape with $N = 3800$ vertices. (b) Downsampled mean shape with 759 vertices. (c) Tibia mean shape with $N = 4701$ vertices. (d) Downsampled mean shape with 814 vertices. For both bone models $P = 36$ sparse points have been randomly drawn from the original surface according to the procedure described in section 6.

tion. Moreover, the ANISO-ds method outperforms the ISO-ds method, which confirms our elaborations in Fig. 2 on real data. Moreover, for both bone PDMs, the AICP method performs very similar to ICP.

6.2. Liver

We carried out LAI and LOO experiments using a liver PDM with $N = 4542$ points (cf. Fig. 8 (a)). Figure 8 (b) shows the downsampled PDM. For each training shape 10 instances of sparse points $\mathcal{P}$ are sampled. Summaries of the results are shown in Fig. 9.

For both settings, LAI and LOO, without any noise, i.e. $\tilde{\sigma} = 0$ cm, the anisotropic method outperforms both ICP methods with respect to the mean DSC regardless of how many random points have been sampled. Especially for 36 and 90 points the accuracy of the anisotropic method becomes increasingly superior compared to ICP. Consid-
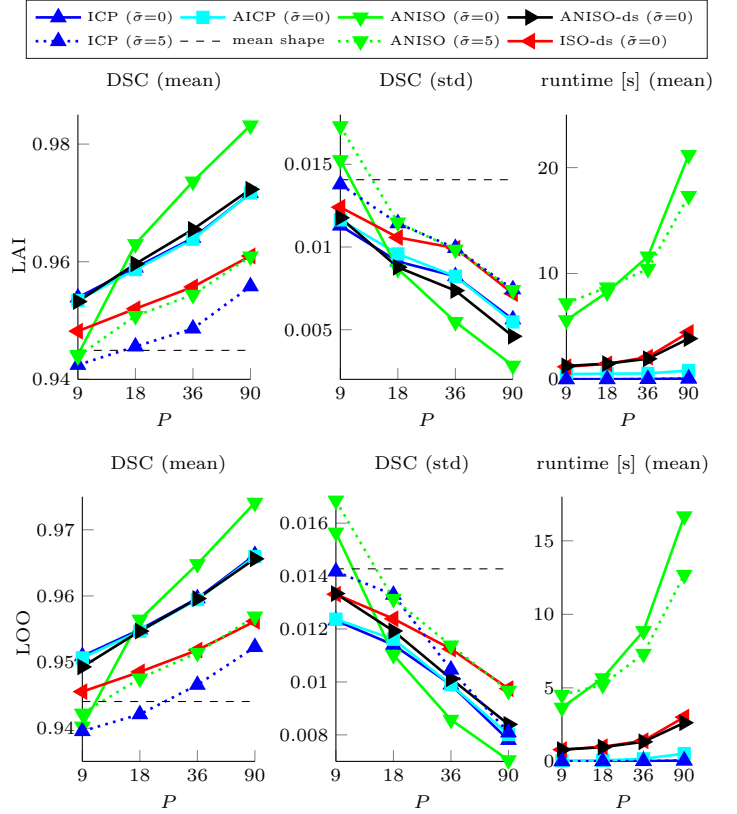
Figure 6: DSC and runtime for femur LAI and LOO results.
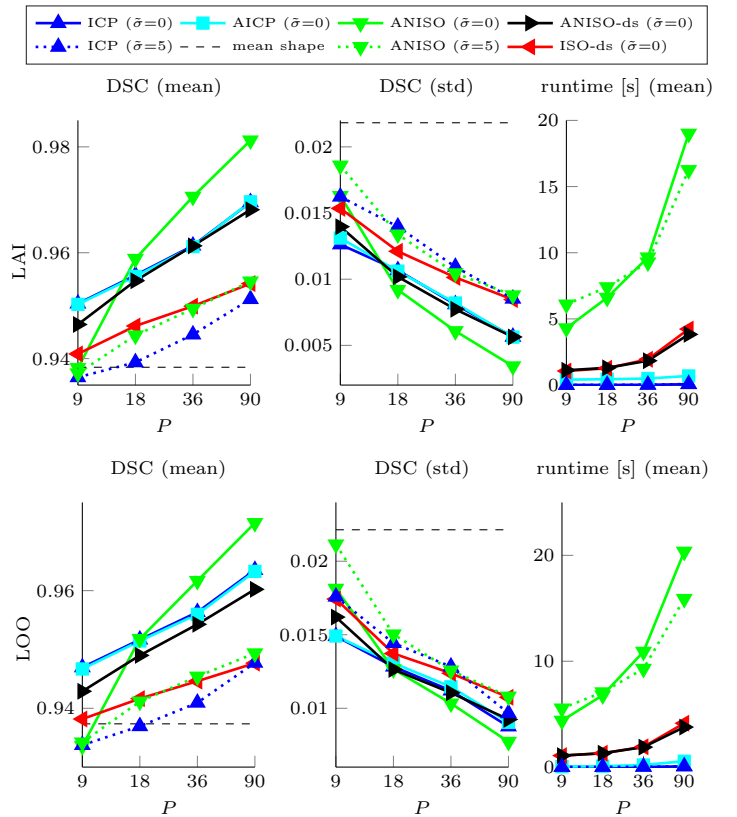
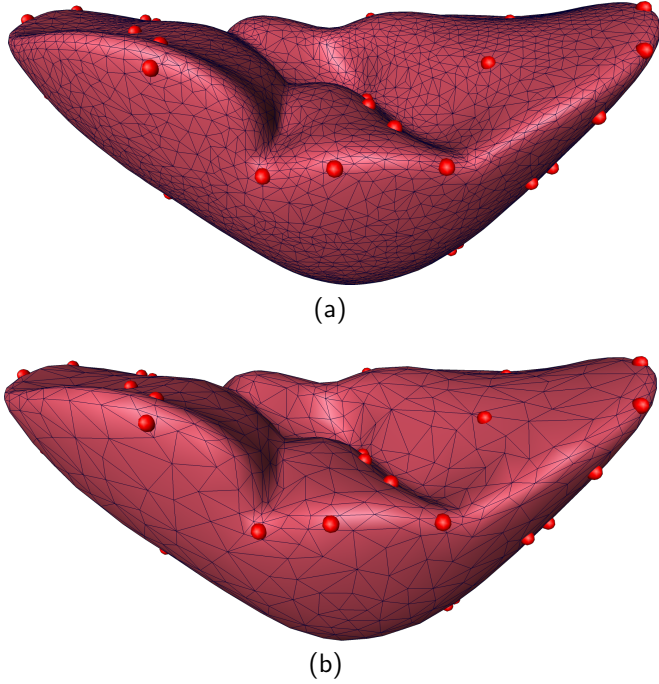Figure 7: DSC and runtime for tibia LAI and LOO results.

10

(a)

(b)

Figure 8: Liver dataset. (a) Mean shape with $N = 4542$ vertices. (b) Downsampled mean shape with 908 vertices. $P = 36$ sparse points have been randomly drawn from the original surface according to the procedure described in section 6.

Figure 9: DSC and runtime for liver LAI and LOO results.

ering the random points disturbed by Gaussian noise with $\tilde{\sigma} = 1$ cm, the ICP method yields better results for 9 and 18 points. With noisy points, for the LAI and the LOO experiments at least 36 points seem to be necessary for the anisotropic method to achieve results similar to the ICP methods with respect to the DSC. Similarly to the femur and tibia cases, the AICP method is on par with the ICP method.

6.3. Hip

We carried out LAI and LOO experiments using a hip PDM with $N = 5603$ points (cf. Fig. 10 (a)). Figure 10 (b) shows the downsampled PDM. For each training shape 10 instances of sparse points $\mathcal{P}$ are sampled. Summaries of the results are shown in Fig. 11. For both settings, LAI and LOO, without any Gaussian disturbance, i.e. $\tilde{\sigma} = 0$ mm, the anisotropic method outperforms the ICP method with respect to the mean Dice Similarity Coefficient for more than 9 points. Again, for the ANISO method the accuracy is increasing with more sampled points. Considering the random points disturbed by Gaussian noise with $\tilde{\sigma} = 5$mm, the ICP method yields better results for 9 points, whereas for 18 points or more the ANISO method
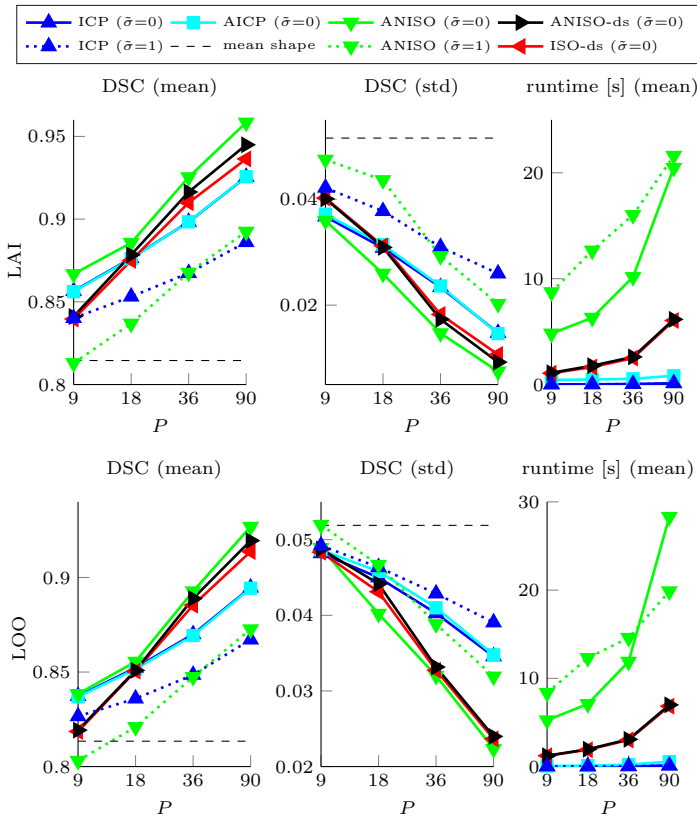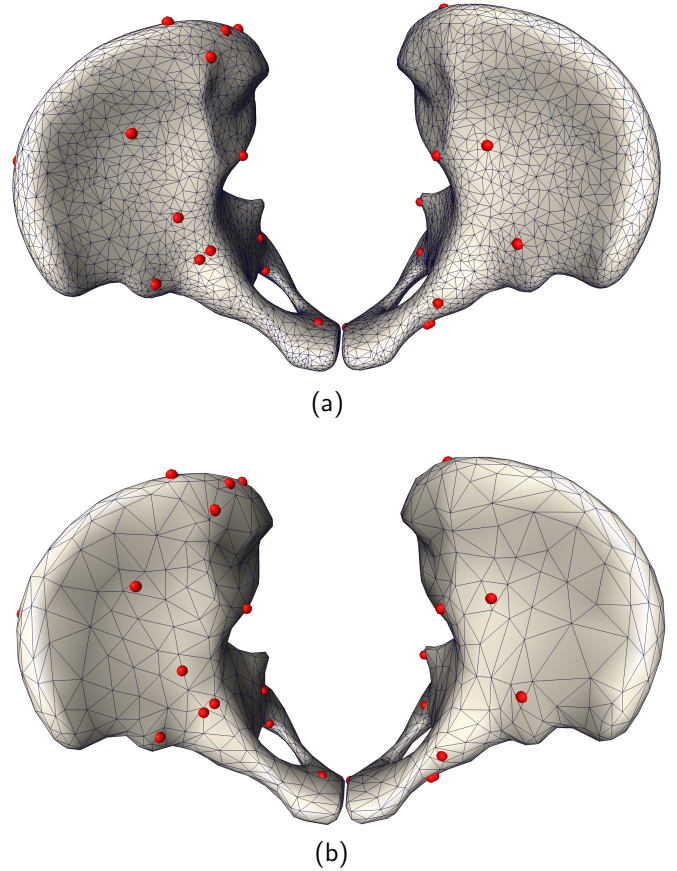

(a)

(b)

Figure 10: Hip dataset. (a) Mean shape with $N = 5603$ vertices. (b) Downsampled mean shape with 1120 vertices. $P = 36$ sparse points have been randomly drawn from the original surface according to the procedure described in section 6.

11

performs better. Comparing the ICP and the AICP approach, both methods are similar with respect to the DSC.
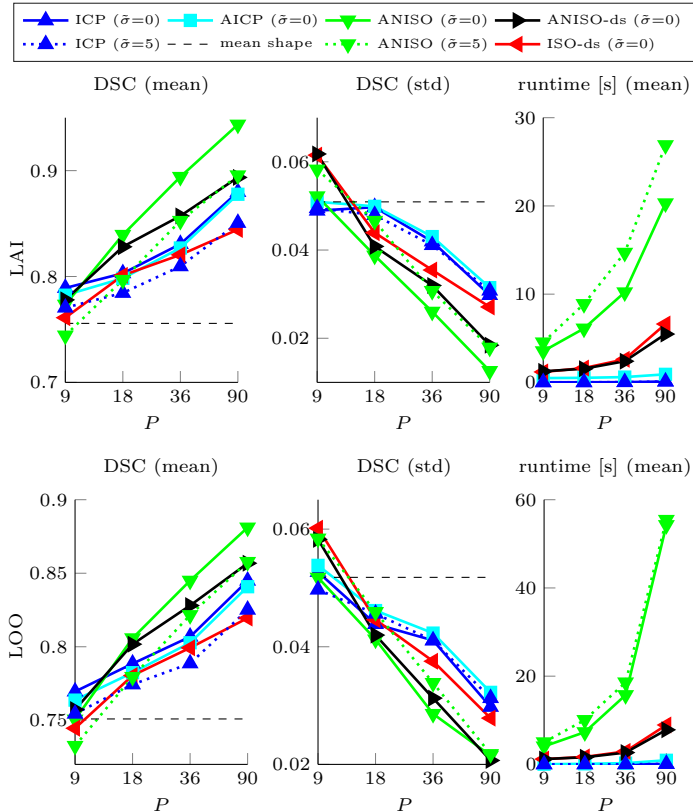

Figure 11: DSC and runtime for hip LAI and LOO results.

6.4. Brain Structures

In this experiment we consider a multi-object PDM that captures the inter-relation between multiple brain structures, namely *Substantia Nigra & Subthalamic Nucleus* (SN+STN, as compound object), *Nucleus Ruber* (NR), *Thalamus* (Th) and *Putamen & Globus Pallidus* (Put+GP, as compound object), where all structures are considered bilaterally. The mean of the PDM is shown in Fig. 12 (a).

The PDM is learned from multi-label segmentations that are all represented in a common coordinate system, the MNI ICBM 152 (Fonov et al., 2009) template space in our case (more details on the manual annotation and the establishment of correspondences can be found in our previous work (Bernard et al., 2014, 2016b)). The alignment of the patient images into the MNI template space is conducted using the rigid image registration method FLIRT (Jenkinson and Smith, 2001). Hence, thanks to this alignment, the orientation and position are already approximately normalised. Consequently, for a new patient image that is to be segmented, a registration to the MNI template space is sufficient.

6.4.1. Interactive Segmentation

One interesting perspective of our presented shape-aware surface reconstruction method might be its integration into an interactive segmentation setting. This could be implemented by alternating between the user annotating object boundaries, and running our fitting method in order to reconstruct a surface from the user-input. As a first step into this direction, in addition to random sparse points, for the brain structure dataset we also evaluate a fitting to partial contours. We decided to focus on *partial* contours instead of full contours since for some of the structures some regions may be difficult to delineate. In order to perform this evaluation we synthetically generated contours according to the procedure described in the supplementary material. In Fig. 12 (c) we show the sparse points that constitute these partial contours. The main idea of the contour generation is as follows. First, we randomly select a 2D slice of the binary 3D segmentation image for a particular object. Next, from the 2D slice of the ground truth segmentation a subcontour of the entire boundary is randomly selected. Combining the chosen slice index with the subcontour leads to a planar 3D contour. Eventually, this contour is subsampled and the points are added to $\mathcal{P}$. Whilst our synthetic contour generation does not substitute a proper study involving user-drawn contours, we found that the resulting contours look plausible to be drawn by a human operator. For the generation of the partial contours we considered two settings, $\mathfrak{c}_1$ and $\mathfrak{c}_2$. For $\mathfrak{c}_1$, we have two contours in four of the eight brain structures, as shown in Fig. 12 (c), where the number of points ranges from 58 to 106, with a median of 80. For $\mathfrak{c}_2$, we have a single contour for each of the eight brain structures, where the number of points ranges from 58 to 81, with a median of 68. Note that when considering partial contours, for each $p_j \in \mathcal{P}$ we assume that it is known to which of the eight brain structures it belongs, which is used to constrain the E-step in our fitting methods (and the nearest-neighbour routine for the ICP methods).

In order to evaluate the robustness of our presented method with respect to noisy inputs, we also created noisy contours. For that, each partial contour is translated in the image plane by a random vector that has a zero-mean Gaussian distribution with covariance $\tilde{\sigma}^2 \mathbf{I}_2$. Our motivation for using *in-plane* translations is that when the user draws a contour in the image, the particular image plane is fixed and thus the only uncertainty occurs in-plane.

6.4.2. Results

For each of the $K = 17$ training shapes we sample 20 instances of sparse points $\mathcal{P}$. Following concepts introduced by Cootes and Taylor (1995); Wang and Staib (1998, 2000), in the LOO experiments we increase the flexibility of the resulting shape model by extracting $M = 96$ eigenvectors of a modified covariance matrix. In our case, we used the sum of the (scaled) covariance matrix $\mathbf{C}$ and a Gaussian kernel with standard deviation of approximately 5mm. We refer the interested reader to our previous work
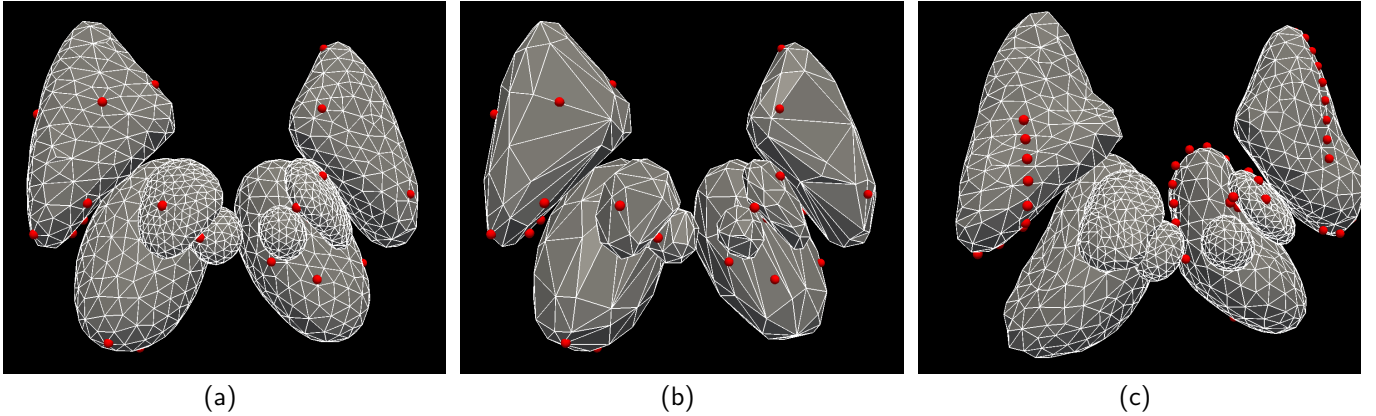
Figure 12: Brain shapes dataset. (a) Mean shape with $N = 1792$ vertices. (b) Downsampled mean shape with 371 vertices. In (a,b) $P = 36$ sparse points have been randomly drawn from the original surface according to the procedure described in section 6. (c) A shape instance from the training set with partial contours.

for details (Bernard et al., 2016a), where the method is referred to as *kPCA*. We also demonstrated that this method is able to improve the generalisation ability of the PDM in this small training dataset comprising $K = 17$ shapes. Summaries of the results are shown in Fig. 13.
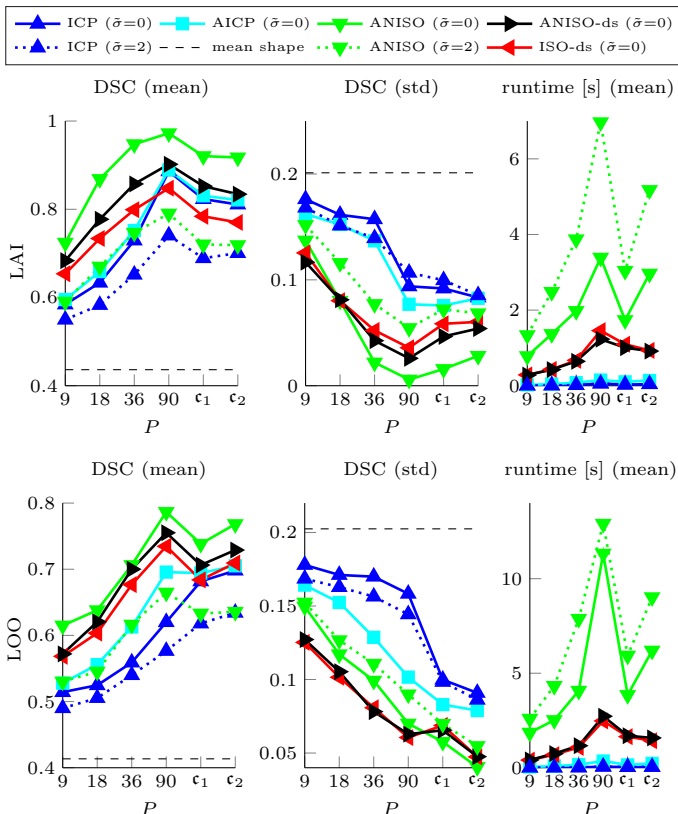

Figure 13: DSC and runtime for brain shape LAI and LOO results.

As anticipated, the plots confirm that with respect to fitting accuracy an increasing number of measurements P improves the results. Moreover, it can be seen that the ANISO method outperforms the ICP methods in all cases, where the standard deviation of ICP is much larger. In all

cases, running *any* fitting method is superior compared to simply using the mean shape. Due to the simplicity of the ICP method, its runtime is much lower compared to the proposed fitting methods. However, by using the ANISO method with a downsampled PDM, the runtime can be reduced compared to the ANISO method, whilst still having superior fitting accuracy compared to ICP.

When using the method for interactive segmentation, the individual annotation of a moderate amount of random points, e.g. $P = 90$, is rather tedious and time-demanding. Thus, in the settings $\mathfrak{c}_1$ and $\mathfrak{c}_2$ we have evaluated the alternative of using points that can be derived from a very few number of partial contours. With that, one can obtain a reasonable number of points, cf. section 6.4.1, with much less effort. Our results suggest that this is in general preferable over using a small amount of random points, say $P \leq 18$. Nevertheless, the case of having $P = 90$ random points outperformed the considered partial contours. We believe the reason is that the random points contain more diverse and scattered information compared to contours containing many correlated points.

7. Conclusion and Outlook

In this paper we have presented a methodology for a shape-aware surface reconstruction from sparse surface points. The proposed methodology is superior compared to the standard approach of ICP with respect to accuracy and robustness on a wide range of datasets. In this method, the likely shape of the object that is to be reconstructed is captured by a PDM associated with a surface mesh. By interpreting the available sparse surface points as samples drawn from a GMM, the surface reconstruction task is cast as the maximisation of the posterior likelihood, which we tackle by variants of the EM algorithm. In order to achieve a surface-based fitting, we use a GMM with anisotropic covariance matrices, which are "oriented" by the surface normals at the PDM points. However, this re-

sults in a non-concave optimisation problem that needs to be solved in each M-step. We deal with this by maximising a concave approximation that considers the surface normals of the PDM computed from the previous value of the shape deformation parameter. As stated before, this approximation makes sense with the assumption that neighbour PDM vertices vary smoothly and the fact that surface normals are invariant to translations. We empirically demonstrated that finding a global maximum of this approximation leads to better results compared to finding a local optimum during the exact (non-concave) M-step. Moreover, our proposed concave approximation results in an algorithm that has the same time complexity as the isotropic fitting procedure.

The proposed surface reconstruction method deals exclusively with shape deformations. Thus, the normalisation of the pose must be solved a-priori in an application-dependent manner. In the example of the multi-object brain shape reconstruction we dealt with this issue by first performing rigid image registration in order to align the data into a common coordinate system. Dealing with the limitation of not explicitly considering a rigid transformation in order to model the pose of the object is the next step for achieving an even broader applicability. Whilst in principal one can formulate an analogous problem that considers the pose, the resulting problem is much more difficult to solve. This is because a simultaneous maximisation must be performed with respect to the rigid transformation and the shape deformation parameter. This is usually done iteratively, as in Active Shape Model search (Cootes and Taylor, 1992). With that, particular challenges to be dealt with are that the resulting surface reconstruction procedure would be much more sensitive to unwanted local optima as well as much slower.

We have conducted an evaluation of the proposed algorithmic tools on a wide range of datasets in order to demonstrate their general applicability in the field of medical image analysis. For the evaluation we have considered a general noise model, i.e. Gaussian noise that is independent for each point. In addition, for the contour case we also considered in-plane (Gaussian) noise. Since we focus on demonstrating the general applicability, a detailed evaluation of certain application-specific aspects (e.g. specific noise models) has not been studied in this paper. One interesting direction for future work is to consider outliers in the sparse points, which is relevant if the sparse points are automatically generated (e.g. using feature extraction methods). This could for example be tackled by integrating an additional uniform component into the mixture model (Myronenko and Song, 2010). Another approach is to use a RANSAC-like procedure (Fischler and Bolles, 1981). In order to encourage the integration of our method into application-specific medical imaging workflows, we make our method publicly available[6]. We expect

that the public availability of the method will stimulate the commencement of interesting new research questions. One such question may be which points are most useful for the reconstruction of surfaces.

Appendix A. Gradient of Q in (17)

In the following, we derive the gradient of Q w.r.t. $\boldsymbol{\alpha}$, i.e.

$$\nabla_{\boldsymbol{\alpha}} Q = \left[\frac{\partial Q}{\partial \boldsymbol{\alpha}_m}\right]_m. \tag{A.1}$$

For brevity, we write $\partial\cdot$ to denote the partial derivative $\frac{\partial\cdot}{\partial\boldsymbol{\alpha}_m}$ w.r.t. $\boldsymbol{\alpha}_m$, where the dependence on m is implicit. First, we note that the cross-product $u \times v$ of two vectors $u, v \in \mathbb{R}^3$ can be written as the matrix multiplication $[u]_\times v$, where the operator $[\cdot]_\times : \mathbb{R}^3 \to \mathbb{R}^{3\times3}$ creates a skew-symmetric matrix from its input vector by

$$\left[\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}\right]_\times := \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}. \tag{A.2}$$

Introducing

$$b_i(\boldsymbol{\alpha}) := (y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})) \times (y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})), \tag{A.3}$$

we can write $n_i(\boldsymbol{\alpha}) = \frac{b_i(\boldsymbol{\alpha})}{\|b_i(\boldsymbol{\alpha})\|}$. Now, by representing the cross product in (A.3) as a matrix product with the notation from (A.2), and by using the product rule, the partial derivative of $b_i(\boldsymbol{\alpha})$ is given by

$$\partial b_i(\boldsymbol{\alpha}) = \{\partial[y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})]_\times\}(y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})) \tag{A.4}$$
$$+ [y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})]_\times\{\partial(y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha}))\}$$
$$= [\boldsymbol{\Phi}_{i_2,m} - \boldsymbol{\Phi}_{i,m}]_\times(y_{i_3}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})) \tag{A.5}$$
$$+ [y_{i_2}(\boldsymbol{\alpha}) - y_i(\boldsymbol{\alpha})]_\times(\boldsymbol{\Phi}_{i_3,m} - \boldsymbol{\Phi}_{i,m}).$$

Moreover,

$$\partial\|b_i(\boldsymbol{\alpha})\| = \frac{b_i^T(\boldsymbol{\alpha})\{\partial b_i(\boldsymbol{\alpha})\}}{\|b_i(\boldsymbol{\alpha})\|}. \tag{A.6}$$

By using the quotient rule, the partial derivative of $n_i(\boldsymbol{\alpha})$ is given by

$$\partial n_i(\boldsymbol{\alpha}) = \frac{\|b_i(\boldsymbol{\alpha})\|\{\partial b_i(\boldsymbol{\alpha})\} - b_i(\boldsymbol{\alpha})\{\partial\|b_i(\boldsymbol{\alpha})\|\}}{\|b_i(\boldsymbol{\alpha})\|^2}. \tag{A.7}$$

---

[6]`https://github.com/fbernardpi/sparsePdmFitting`

14

Using $\partial n_i(\boldsymbol{\alpha})$, we can write

$$\partial \mathbf{W}_i(\boldsymbol{\alpha}) = (\eta-1)(\{\partial n_i(\boldsymbol{\alpha})\}n_i^T(\boldsymbol{\alpha}) + n_i(\boldsymbol{\alpha})\{\partial n_i^T(\boldsymbol{\alpha})\})\,. \tag{A.8}$$

Now, given the expression for $\partial \mathbf{W}_i(\boldsymbol{\alpha})$, we can finally compute the partial derivative of Q w.r.t. $\boldsymbol{\alpha}_m$, which is

$$\partial Q(\boldsymbol{\alpha}, \sigma, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)}) = \tag{A.9}$$
$$-(\Lambda^{-1})_{m,:}\boldsymbol{\alpha} - \frac{1}{2\sigma^2}\sum_{i,j}\mathfrak{p}(i|p_j, \boldsymbol{\alpha}^{(n)}, \sigma^{(n)})\cdot$$
$$[(p_j - y_i(\boldsymbol{\alpha}))^T\{\partial \mathbf{W}_i(\boldsymbol{\alpha})\}(p_j - y_i(\boldsymbol{\alpha}))-$$
$$2\boldsymbol{\Phi}_{i,m}^T\mathbf{W}_i(\boldsymbol{\alpha})(p_j - y_i(\boldsymbol{\alpha}))]\,.$$

References

Albrecht, T., Lüthi, M., Gerig, T., Vetter, T., 2013. Posterior shape models. Medical Image Analysis 17, 959–973.

Amenta, N., Bern, M., Kamvysselis, M., 1998. A new Voronoi-based surface reconstruction algorithm, in: SIGGRAPH.

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J., 2005. SCAPE: shape completion and animation of people, in: SIGGRAPH, ACM. pp. 408–416.

Bajaj, C.L., Bernardini, F., Xu, G., 1995. Automatic reconstruction of surfaces and scalar fields from 3D scans, in: SIGGRAPH.

Baka, N., de Bruijne, M., Reiber, J., 2010. Confidence of model based shape reconstruction from sparse data, in: Biomedical Imaging: From Nano to Macro.

Ballard, D.H., 1981. Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognition 13, 111–122.

Barratt, D.C., Chan, C.S., Edwards, P.J., Penney, G.P., Slomczykowski, M., Carter, T.J., Hawkes, D.J., 2008. Instantiation and registration of statistical shape models of the femur and pelvis using 3D ultrasound imaging. Medical Image Analysis 12, 358–374.

Berger, M., Tagliasacchi, A., Seversky, L., Alliez, P., 2014. State of the art in surface reconstruction from point clouds. EUROGRAPHICS STAR .

Bernard, F., Gemmar, P., Hertel, F., Goncalves, J., Thunberg, J., 2016a. Linear Shape Deformation Models with Local Support Using Graph-based Structured Matrix Factorisation, in: Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV.

Bernard, F., Gemmar, P., Husch, A., Saleh, C., Neb, H., Dooms, G., Hertel, F., 2014. Improving the Consistency of Manual Deep Brain Structure Segmentations by Combining Variational Interpolation, Simultaneous Multi-Modality Visualisation and Histogram Equilisation. Biomedical Engineering / Biomedizinische Technik 59, 131–134.

Bernard, F., Salamanca, L., Thunberg, J., Hertel, F., Goncalves, J., Gemmar, P., 2015. Shape-aware 3D Interpolation using Statistical Shape Models, in: Proceedings of Shape Symposium, Delemont.

Bernard, F., Vlassis, N., Gemmar, P., Husch, A., Thunberg, J., Goncalves, J., Hertel, F., 2016b. Fast correspondences for statistical shape models of brain structures, in: Proc. SPIE Medical Imaging, San Diego.

Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., Taubin, G., 1999. The ball-pivoting algorithm for surface reconstruction. IEEE Transactions on Visualization and Computer Graphics 5, 349–359.

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14, 239–256.

Billings, S.D., Boctor, E.M., Taylor, R.H., 2015. Iterative mostlikely point registration (IMLP): a robust algorithm for computing optimal shape alignment. PloS one 10.

Blanc, R., Székely, G., 2012. Confidence regions for statistical model based shape prediction from sparse observations. IEEE Transactions on Medical Imaging 31, 1300–1310.

Blanz, V., Mehl, A., Vetter, T., Seidel, H.P., 2004. A statistical method for robust 3D surface reconstruction from sparse data, in: 3D Data Processing, Visualization and Transmission, pp. 293–300.

Bolle, R.M., Vemuri, B.C., 1991. On three-dimensional surface reconstruction methods. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 1–13.

Bouaziz, S., Tagliasacchi, A., Pauly, M., 2013. Sparse iterative closest point. Computer Graphics Forum 32, 1–11.

de Bruijne, M., van Ginneken, B., Viergever, M.A., Niessen, W.J., 2004. Interactive segmentation of abdominal aortic aneurysms in CTA images. Medical Image Analysis 8, 127–138.

Chan, C.S., Edwards, P.J., Hawkes, D.J., 2003. Integration of ultrasound-based registration with statistical shape models for computer-assisted orthopaedic surgery, in: SPIE Medical Imaging, pp. 414–424.

Chang, W., Zwicker, M., 2009. Range Scan Registration Using Reduced Deformable Models. Computer Graphics Forum 28, 447–456.

Cootes, T.F., Taylor, C.J., 1992. Active Shape Models - Smart Snakes, in: British Machine Vision Conference, pp. 266–275.

Cootes, T.F., Taylor, C.J., 1995. Combining point distribution models with shape models based on finite element analysis. Image and Vision Computing 13, 403–409.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39, 1–38.

Edelsbrunner, H., Mücke, E.P., 1994. Three-dimensional alpha shapes. ACM Transactions on Graphics 13, 43–72.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24.

Fleute, M., Lavallée, S., 1998. Building a complete surface model from sparse data using statistical shape models: Application to computer assisted knee surgery, in: International Conference on Medical Image Computing and Computer-Assisted Intervention.

Fleute, M., Lavallée, S., Julliard, R., 1999. Incorporating a statistically based shape model into a system for computer-assisted anterior cruciate ligament surgery. Medical Image Analysis 3, 209–222.

Fonov, V.S., Evans, A.C., McKinstry, R.C., Almli, C.R., Collins, D.L., 2009. Unbiased nonlinear average age-appropriate brain templates from birth to adulthood. Neuroimage 47, S102.

Gal, R., Shamir, A., Hassner, T., Pauly, M., Cohen Or, D., 2007. Surface reconstruction using local shape priors., in: Symposium on Geometry Processing, pp. 253–262.

van Ginneken, B., de Bruijne, M., Loog, M., Viergever, M.A., 2003. Interactive shape models, in: SPIE Medical Imaging, pp. 1206–1216.

van Ginneken, B., Frangi, A.F., Frangi, R.F., Staal, J.J., Ter Haar Romeny, B.M., Viergever, M.A., 2002. Active Shape Model Segmentation with Optimal Features. IEEE Transactions on Medical Imaging 21, 924–933.

Granger, S., Pennec, X., 2002. Multi-scale EM-ICP: A fast and robust approach for surface registration. European Conference on Computer Vision .

Guo, G., Jiang, T., Wang, Y., Gao, W., 2012. Recovering Missing Contours for Occluded Object Detection. IEEE Signal Processing Letters 19, 463–466.

Guo, G., Jiang, T., Wang, Y., Gao, W., 2013. 2-D shape completion with shape priors. Chinese Science Bulletin 58, 3430–3436.

Heckel, F., Konrad, O., Karl Hahn, H., Peitgen, H.O., 2011. Interactive 3D medical image segmentation with energy-minimizing implicit functions. Computers & Graphics 35, 275–287.

Heimann, T., Meinzer, H.P., 2009. Statistical shape models for 3D medical image segmentation: A review. Medical Image Analysis 13, 543–563.

van den Hengel, A., Dick, A.R., Thormählen, T., Ward, B., Torr, P.H.S., 2007. Interactive 3D Model Completion, in: Digital Image

Computing Techniques and Applications, pp. 175–181.

Herman, G.T., Zheng, J., Bucholtz, C.A., 1992. Shape-based interpolation. IEEE Computer Graphics and Applications 12, 69–79.

Hill, A., Cootes, T.F., Taylor, C.J., 1995. Active Shape Models and the Shape Approximation Problem. Image and Vision Computing 14, 601–607.

Hoppe, H., DeRose, T., Duchamp, T., McDonald, J.A., Stuetzle, W., 1992. Surface reconstruction from unorganized points. SIGGRAPH , 71–78.

Horaud, R., Forbes, F., Yguel, M., Dewaele, G., Zhang, J., 2011. Rigid and articulated point registration with expectation conditional maximization. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 587–602.

Hufnagel, H., Pennec, X., Ehrhardt, J., Ayache, N., Handels, H., 2008. Generation of a statistical shape model with probabilistic point correspondences and the expectation maximization-iterative closest point algorithm. International Journal of Computer Assisted Radiology and Surgery 2, 265–273.

Jenkinson, M., Smith, S., 2001. A global optimisation method for robust affine registration of brain images. Medical Image Analysis 5, 143–156.

Jensen, J.L.W.V., 1906. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. Acta mathematica 30, 175–193.

Kazhdan, M., Bolitho, M., Hoppe, H., 2006. Poisson surface reconstruction, in: Eurographics Symposium on Geometry Processing.

Liu, J., Udupa, J.K., 2009. Oriented active shape models. IEEE Transactions on Medical Imaging 28, 571–584.

Lu, M., Zheng, B., Takamatsu, J., Nishino, K., 2011. 3D shape restoration via matrix recovery, in: Computer Vision–ACCV Workshops, pp. 306–315.

Maier-Hein, L., Franz, A.M., dos Santos, T.R., Schmidt, M., Fangerau, M., Meinzer, H.P., Fitzpatrick, J.M., 2012. Convergent iterative closest point algorithm to accomodate anisotropic and inhomogenous localization error. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 1520–1532.

Maier-Hein, L., Schmidt, M., Franz, A.M., dos Santos, T.R., Seitel, A., Jähne, B., Fitzpatrick, J.M., Meinzer, H.P., 2010. Accounting for anisotropic noise in fine registration of time-of-flight range data with high-resolution surface data, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 251–258.

Meng, X.L., Rubin, D.B., 1993. Maximum likelihood estimation via the ECM algorithm: A general framework. Biometrika 80, 267–278.

Myronenko, A., Song, X., 2010. Point Set Registration: Coherent Point Drift. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 2262–2275.

Myronenko, A., Song, X., Carreira-Perpinán, M.A., 2007. Non-rigid point set registration: Coherent Point Drift, in: Neural Information Processing Systems, pp. 1009–1016.

Nocedal, J., Wright, S.J., 2006. Numerical Optimization. Springer.

Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D., 2002. Shape distributions. ACM Transactions on Graphics (TOG) 21, 807–832.

Park, S., Guo, X., Shin, H., Qin, H., 2006. Surface completion for shape and appearance. The Visual Computer 22, 168–180.

Pauly, M., Mitra, N.J., Giesen, J., Gross, M.H., Guibas, L.J., 2005. Example-Based 3D Scan Completion., in: Symposium on Geometry Processing, pp. 23–32.

Rajamani, K.T., Styner, M.A., Talib, H., Zheng, G., Nolte, L.P., Ballester, M.A.G., 2007. Statistical deformable bone models for robust 3D surface extrapolation from sparse data. Medical Image Analysis 11, 99–109.

Raya, S.P., Udupa, J.K., 1990. Shape-based interpolation of multidimensional objects. IEEE Transactions on Medical Imaging 9, 32–42.

Rusinkiewicz, S., Levoy, M., 2001. Efficient variants of the ICP algorithm, in: 3-D Digital Imaging and Modeling.

Schroers, C., Setzer, S., Weickert, J., 2014. A Variational Taxonomy for Surface Reconstruction from Oriented Points, in: Computer Graphics Forum, pp. 195–204.

Schwarz, T., Heimann, T., Tetzlaff, R., Rau, A.M., Wolf, I., Meinzer, H.P., 2008. Interactive surface correction for 3D shape based segmentation, in: SPIE Medical Imaging, pp. 69143O–69143O.

Segal, A., Haehnel, D., Thrun, S., 2009. Generalized-icp, in: Robotics: Science and Systems.

Seim, H., Kainmüller, D., Lamecker, H., Bindernagel, M., Malinowski, J., Zachow, S., 2010. Model-based Auto-Segmentation of Knee Bones and Cartilage in MRI Data. Proc. of Medical Image Analysis for the Clinic: A Grand Challenge , 215–223.

Shen, C.H., Fu, H., Chen, K., Hu, S.M., 2012. Structure recovery by part assembly, in: SIGGRAPH.

Stoll, C., Karni, Z., Rössl, C., Yamauchi, H., Seidel, H.P., 2006. Template deformation for point cloud fitting. SPBG , 27–35.

Tan, J.H., Acharya, U.R., 2014. Active spline model: A shape based model—interactive segmentation. Digital Signal Processing 35, 64–74.

Taylor, J., Bordeaux, L., Cashman, T., Corish, B., Keskin, C., Sharp, T., Soto, E., Sweeney, D., Valentin, J., Luff, B., 2016. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. ACM Transactions on Graphics (TOG) 35, 143.

Taylor, J., Stebbing, R., Ramakrishna, V., Keskin, C., Shotton, J., Izadi, S., Hertzmann, A., Fitzgibbon, A., 2014. User-Specific Hand Modeling from Monocular Depth Sequences, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE. pp. 644–651.

Timinger, H., Pekar, V., von Berg, J., Dietmayer, K., Kaus, M., 2003. Integration of interactive corrections to model-based segmentation algorithms, in: Bildverarbeitung für die Medizin 2003. Springer, pp. 171–175.

Treece, G.M., Prager, R.W., Gee, A.H., 2000. Surface interpolation from sparse cross sections using region correspondence. IEEE Transactions on Medical Imaging 11, 1106–1114.

Turk, G., O'Brien, J.F., 1999. Shape transformation using variational implicit functions, in: SIGGRAPH, pp. 335–342.

Wang, Y., Staib, L.H., 1998. Boundary finding with correspondence using statistical shape models, in: Computer Vision and Pattern Recognition (CVPR), pp. 338–345.

Wang, Y., Staib, L.H., 2000. Boundary finding with prior shape and smoothness models. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 738–743.

Zheng, G., 2013. Expectation conditional maximization-based deformable shape registration, in: International Conference on Computer Analysis of Images and Patterns, Springer. pp. 548–555.

Zheng, G., Rajamani, K.T., Nolte, L.P., 2006. Use of a Dense Surface Point Distribution Model in a Three-Stage Anatomical Shape Reconstruction from Sparse Information for Computer Assisted Orthopaedic Surgery: A Preliminary Study. Computer Vision–ACCV .