Unsupervised Shape Correspondence Estimation for Anatomical Shapes

Master's Thesis

Computational Engineering Science
TU Berlin

Faculty V - Mechanical Engineering and Transport Systems
Institute for Medical Engineering

Author:	Lisa Bautz
Matriculation Number:	372855
First Examiner:	Prof. Dr. Marc Kraft
Second Examiner:	Steven Mücke, M.Sc.
Advisors Zuse Institute Berlin:	Dr. Stefan Zachow
	Tamaz Amiranashvili, M.Sc.
Submission Date:	April 6, 2023

Sworn Affidavit

in accordance with section § 60 Abs. 8 AllgStuPO:

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Berlin, 6.4.2023

L. Bautz

Lisa Bautz

Masterthesis Computational Engineering Science

Technische
Universität
Berlin

Fakultät V Verkehrs- und
Maschinensysteme
Institut für Maschinenkonstruktion
und Systemtechnik
Fachgebiet Medizintechnik
Prof. Dr.-Ing. Marc Kraft

For
Lisa Bautz, Matr.Nr. 372855, Fakultät V

Topic: Unsupervised Shape Correspondence Estimation for Anatomical Shapes

Shape correspondence enables us to transfer the location of labeled points of one shape to the surface of another unlabeled homological shape. While sparse shape correspondence registers only a small set of surface points, dense shape correspondence aims to register the whole discretized surface of a shape to the surface of another shape. There are many possible applications for dense shape correspondence in the medical context. A set of registered shapes can be used to build a statistical shape model (SSM) of bones, organs or other parts of the human body. These models yield the possibility to quantify and visualize shape variations in healthy and pathological populations. For therapy planning it is also desirable to have a fully labeled 3D-model of the area of interest, a so called "anatomical atlas".

Shape correspondence is often established on basis of machine learning methods in a semi-automatic way. This process often involves manual labelling of a sparse set of landmarks to initialize the dense registration. Since the labeling process is time-consuming and often in need of a medical expert, a fully automatic registration process is favorable. More recent approaches aim to solve the shape correspondence task in an unsupervised way. These models can be trained on a set of training meshes without correspondence and generate correspondence by registering a template to a patient-specific shape.

The objective of this thesis is to compare different methods for unsupervised correspondence generation as well as to potentially improve the state of the art. Since there is no sufficiently accurate ground truth for dense correspondence, the evaluation of the methods is challenging. Therefore, a systematic performance analysis regarding criteria related to dense correspondence on different data sets is necessary.

The thesis includes the following tasks:

1. Literature review regarding
    - Unsupervised methods used for shape correspondence estimation
    - Evaluation of dense correspondences in a medical context
2. Development of a concept to systematically evaluate generated shape correspondences
3. Comparison of selected methods used for shape correspondence estimation
4. Performance analysis of the selected models on different data sets to ensure generalizability
5. Adaptation of the methods to the medical context and data sets at hand
6. Further development of one Method
7. Documentation of all tasks

The thesis is supervised by the Department of Medical Engineering at the TU Berlin in cooperation with the Zuse Institute Berlin. The first examiner is Prof. Kraft, and the second examiner is M.Sc. Steven Mücke. Supervisors of the Zuse Institute are M.Sc. Tamaz Amiranashvili and Dr. Stefan Zachow. After completion of the thesis, an abstract in German and English must be written and a (graded) presentation must be given.

Berlin, 08.11.2022

M. Kraft

Prof. Dr.-Ing. M. Kraft

Abstract

The concept of shape correspondence describes a relation between two or more shapes of the same class. It often consists of a mapping between points on semantically similar locations of all shapes. One possible application for shape correspondence in medicine is the automatic location of anatomical landmarks. Another popular application is the construction of statistical shape models. These models are an established way to represent geometric variation of anatomical shapes in a compact way. Possible applications range from the generation of shapes and reconstruction tasks to disease classification.

This thesis aims to investigate unsupervised methods that can be used to estimate such a correspondence on anatomical shapes. While most methods used in the medical domain focus on classical optimization algorithms to establish correspondence, the broader computer vision domain developed a versatile field of data-driven methods. Recently, the new shape model FlowSSM was introduced, which does not require predefined correspondences for training as it generates them itself. As the performance of the shape model is quite competitive, it is natural to assume that the generated correspondences are of high quality as well.

For this reason, we evaluate the quality of the correspondences generated by FlowSSM within this thesis. Furthermore, we modify the method by adding a second loss term that minimizes geodesic distortions. This is done to favor isometric deformations which can lead to better correspondences. We compare the results with two established methods from the medical domain, LDDMM and Meshmonk. Furthermore, we investigate the performance of a fourth method called Neuromoph. This data-driven method comes from the wider computer vision field and was not tested on anatomical data yet.
All methods are evaluated with a set of different metrics. This includes metrics to assess the quality of the resulting meshes, a sparse correspondence error on anatomical landmarks, and metrics to measure the quality of the resulting shape models. Furthermore, we test all methods on three datasets with different degrees of geometric variation, namely liver, distal femur and face.

We show that FlowSSM produces correspondences with state-of-the-art quality. Moreover, our modification further improved the quality of correspondences at a global level. Nevertheless, there is no clear ranking between all methods, as the results differ between metrics and datasets. Thereby, we can show that there are different qualities to a proper correspondence which are reflected in the different metrics. It is therefore strongly recommendable to choose a correspondence estimation method specifically for the problem at hand.

Zusammenfassung

Das Konzept der Formkorrespondenz zwischen 3D-Objekten einer Klasse beschreibt eine Beziehung zwischen den Instanzen (oft Punkten) der unterschiedlichen Objekten. Hierbei werden Punkte, die an semantisch gleichwertigen Orten liegen, miteinander in Verbindung gebracht. Eine mögliche Anwednung der Formkorrespondenz im medizinischen Bereich ist daher die automatisierte Lokalisierung von anatomischen Landmarken. Eine weitere Anwendung ist das Erstellen von statistischen Formmodellen. Mit diesen kann die geometrische Variation anatomischer Formen kompakt abgebildet werden. Medizinische Anwendungen reichen dabei von der einfachen Formgenerierung zu komplexeren Rekonstruktionsaufgaben und der Klassifizierung von gesunden und pathologischen Formen.

In dieser Arbeit werden unterschiedliche Methoden zur Erzeugung von Formkorrespondenzen untersucht. Die entsprechende Literatur im medizinischen Bereich verwendet hierzu meist Methoden, die das klassische Optimierungsproblem einer nichtrigiden Transformation lösen. Im Computer Vision Bereich wurden in den letzten Jahren auch einige datengetriebene Methoden zur Korrespondenzgenerierung veröffentlicht. Im letzten Jahr wurde außerdem die Methode FlowSSM zur Erstellung statistischer Formmodelle vorgestellt, die nicht auf korrespondierenden Oberflächen basiert, sondern diese selbst erzeugt. Da FlowSSM trotzdem konkurenzfähige Ergebnisse erzielt, ist naheliegend, dass auch die zugrundeliegenden, selbst generierten Korrespondenzen von hoher Qualität sind.

Innerhalb dieser Arbeit wird daher die Qualität der von FlowSSM erzeugten Korrespondenzen evaluiert. Außerdem wird die Methode um eine zusätzliche Kostenfunktion erweitert, die geodetische Verzerrungen verhindern soll. Dadurch sollen nichtisometrische Deformationen vermieden werden, wodurch die Qualität der resultierenden Korrenspondenzen gesteigert werden kann. Die Ergebnisse von FlowSSM werden mit zwei etablierten Methoden aus dem medizinischen Bereich, LDDMM und Meshmonk, verglichen. Außerdem wird NeuroMorph, eine aktuelle, datengetriebene Methode aus dem Bereich des maschinellen Sehens getestet. Letztere wurde bisher noch nicht auf medizinischen Daten evaluiert. Die Bewertung aller generierten Korrespondenzen basiert auf ausgewählten indirekten Metriken. Hierzu gehört auch die Performance bei konkreten Anwendungsfällen wie der Lokalisierung von Landmarken und dem Erstellen von statistischen Formmodellen.

Im Rahmen der Arbeit wird gezeigt, dass FlowSSM Korrespondenzen produziert, deren Qualität dem aktuellen State-of-the-art entspricht. Durch das Hinzufügen der zweiten Kostenfunktion wird die Qualität der Korrespondenzen auf einem globalen Level noch weiter gesteigert. Prinzipiell lässt sich jedoch keine Hierarchie zwischen den Methoden ableiten, da die Performance stark innerhalb der untersuchten Metriken und Datensätzen schwankt. Die Auswahl einer passenden Methode sollte sich daher vor allem am Anwendungsfall orientieren.

Contents

List of Figures

List of Tables

Acronyms

CT	Computer Tomography
GDPL	Geodesic Distance Preservation Loss
GPA	Generalized Procrustes Analysis
GPU	Graphics Processing Unit
ICP	Iterative Closest Points
ICA	Independent Component Analysis
IPV	Intra Person Variability
LDDMM	Large Diffeomorphic Deformation Metric Mapping
MRI	Magnetic Resonance Imaging
ML	Machine Learning
MLP	Multi Layer Perceptron
NICP	Non-rigid Iterative Closest Points
PA	Procrustes Analysis
PCA	Principal Component Analysis
PDM	Point Distribution Model
SIF	Self-Intersecting Faces
SIM	Self-Intersecting Meshes
SSM	Statistical Shape Model

1 Introduction

Due to the widespread use of tomographic image data and advances in image processing methods as well as available computing capacities, an ever growing variety of 3D shapes is captured. These three-dimensional measurements of anatomical shapes contain valuable information regarding geometric shape and size. A logical conclusion is the need to compare two or more shapes with each other. In order to do so, a relation between semantically similar points of two shapes is needed. A concept to describe such a relation is the concept of shape correspondence. This consists of a mapping that matches points on shape $\mathcal{X}$ to semantically similar points on shape $\mathcal{Y}$. On the human face, this could be a mapping between the tip of the nose from person $\mathcal{X}$ to the tip of the nose from person $\mathcal{Y}$. An example is visualized in Figure 1.1, where semantically similar points of two faces are matched. This thesis aims to investigate different methods that can be used to establish such a relation.

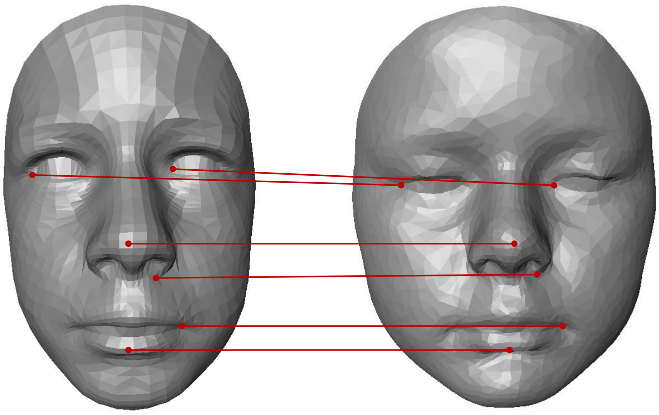


Figure 1.1: Corresponding points (red) between two faces. As the points are placed on anatomical landmarks such as the tip of the nose, it is easy to determine their correct location. For points on the forehead or on the cheeks the correspondence estimation task is more complex.

One possible application of shape correspondence is the automatic localization of anatomical landmarks or measurements between them. Furthermore, as shape correspondence enables us to compare two shapes with each other, it also enables us to compare shapes of a whole population. This opens the field to population wide statistical analyses and classification of e.g., pathological shapes. An established way to capture the geometric variance of a whole population is the use of Statistical Shape Models (SSMs). Hereby, the geometric variability of a shape population is modeled as a combination of (mean) template and a hierarchical set of modes to capture the variability [Coo+95]. Therefore, SSMs can be utilized for generative modeling i.e., the generation from a learned population-based shape space. This opens a large field of applications in the medical domain whenever pathological shapes or shapes with sparse data have to be reconstructed ([Alo+22], [SWZ14]). Another area of application emerges from the potential to represent the whole population in a low dimensional parameter space. When clustering or classification algorithms are applied to this space, the results therefrom can be utilized for diagnosis ([AZT21], [SWZ14]).

Most methods used for SSM construction rely on 3D shape representations that are homologous and in semantic correspondence towards each other. While there are newer methods that do not require corresponding shapes (i.e., [Lüd+22]), the classical SSM approach is still popular, as it is an easy and computationally efficient method that yields robust and well understood results. As different authors (e.g., [HM09], [Lam08]) mentioned, the estimation of this correspondence is the hardest part of SSM construction and yields the most influence on the resulting quality.

Methods for correspondence estimation in the medical domain often solve the classic optimization problem of a non rigid registration between different shapes. With the rise of deep learning methods during the last years, there was a steep growth of data-driven methods for correspondence estimation in the broader computer vision community. Especially unsupervised methods such as [Eis+21] are of interest, as they do not require a predefined correspondence for training. Most of these methods have not been tested on anatomical data yet. As medical data differs from data typically used in the computer vision field, it is unknown how these methods would perform.

Recently, the new shape model FlowSSM ([Lüd+22]) was introduced, which does not require shapes with a predefined correspondence for training as it generates the correspondences itself. As the shape model itself has a strong performance, the hypothesis that the generated correspondences are of high quality is natural. This hypothesis is therefore of research interest and the subject of this thesis. These are its major contributions:

1. We show that FlowSSM is able to generate correspondences with state-of-the-art quality by comparing it to different other methods. This includes established methods from the medical domain ([Dur+14], [Whi+19]).
2. We evaluate how NeuroMorph [Eis+21], a method from the wider computer vision domain, performs on anatomical shapes.

3. We investigate how the implementation of an additional loss term referring to geometrical constraints can help to improve the correspondence quality of FlowSSM.

As there is no proper ground truth for shape correspondence, we have to use different indirect metrics to assess the quality of the generated correspondences. This includes the evaluation of sparse correspondences from anatomical landmarks as well as the quality of resulting SSMs. In order to ensure a better generalization of the results, different datasets with individual challenges were used. On basis of this evaluation setting, we can report that there is no clear ranking between the different investigated methods. It is therefore important to choose a suiting method with regard to the use-case at hand. Furthermore, we can show that the additional loss term implemented in FlowSSM is able to improve the resulting correspondences on most metrics used in this thesis.

Thesis Structure This thesis is structured in the following way:
In Chapter 2 we provide the reader with the theoretical background needed for this thesis. We start with a short introduction on 3D shape representation. Afterwards, we review methods used for correspondence estimation with a focus on data-driven, unsupervised methods. Lastly, we take a look at statistical shape modeling and give a brief overview regarding methods for construction, possible applications and metrics used to assess their quality.

Chapter 3 describes the the methods and materials used in this thesis. At first, we characterize the methods selected for a further investigation. Afterwards, we explain how an additional loss term is added to FlowSSM in order to improve the resulting correspondences. Lastly, we describe the datasets and metrics used for the evaluation of the correspondences.

The performance of all methods is discussed in Chapter 4. First we state the plain results on each dataset and each experiment. Afterwards, we discuss their meanings and place them in a wider context.

The last chapter of this thesis summarizes our findings and gives suggestions for future work.

2 Background and Related Work

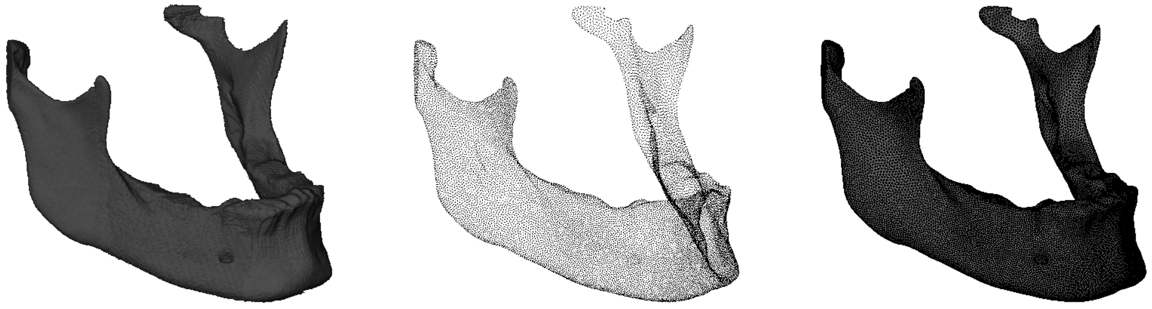
The main goal of this chapter is to present the state-of-the-art regarding methods used for correspondence estimation and to give some background information on Statistical Shape Models (SSMs).
It starts with some basic information on the digital representation of 3D shapes. The vocabulary learned here is useful for large parts of this thesis. Afterwards, we take a look on classical methods for correspondence estimation, as well as the state-of-the-art regarding unsupervised methods. This is necessary to gain an understanding of the variety on different approaches but also their similarities. Lastly, since the construction of SSMs is an important application for shape correspondence in medicine and a good proxy of its quality, a broad overview of construction and applications of SSMs is given. This deepens the understanding of the need for corresponding shapes, as well as measurements to evaluate the quality of SSMs.

2.1 Three Dimensional Shapes

This section gives a short summary of concepts and methods needed throughout this thesis regarding 3D shapes and their digital representations. The aim of this chapter is therefore to provide background information and understanding on secondary methods and concepts used in this thesis, as a more comprehensive overview would be beyond the scope of this thesis.

Shape Representation A 3D object may be represented by its outer boundary which often refers to the shape of an object. In order to represent a surface in a digital format, many methods and representations have been developed. All representations can at least approximately be converted into each other. Since most representations come with certain (dis-)advantages, the choice of a suitable representation is still important. Figure 2.1 visualizes the different representations, using the lower jaw bone (med. mandible) as an example.

Most anatomical shapes result from segmentations of volumetric data, such as Computer Tomography (CT)-scans and Magnetic Resonance Imaging (MRI) data. Therefore, the initial representation is usually *voxel*-based [HM09]. Since voxel data consists of a 3D grid of



(a) Discretization by a regular hexahedral grid (i.e. voxel representation)

(b) 3D pointset representation of the outer boundary of the shape.

(c) Triangulation of the 3D object boundary(i.e. surface mesh representation)

Figure 2.1: Possibilities for digital representation of 3D objects, visualized on the lower jaw bone (e.g. mandible).

values, it requires a lot of memory to store data in a high resolution [Par+19]. Another form of representation that is closely related to the raw data of sensors are *point clouds*, which are sampled on the surface of an object. Unfortunately this representation lacks topology information [Par+19]. If a point set gets expanded by a connectivity information, we call it a *mesh* [HM09]. In meshes the *vertices* are connected by *edges* to form *faces*. The normal vectors associated to each face give information on the in- and outside direction of a shape. According to Ambellan et al. [Amb+19b] the most common representations for building SSMs are meshes and point sets. Throughout this thesis, shapes will be represented as cursive and upper case variables, such as $\mathcal{X}$. The points or vertices of this shape will be represented as a bold and upper case variable of the same letter, e.g. $\boldsymbol{X}$ and the coordinates of one single point as a bold and lower case variable $\boldsymbol{x}$.

Shape Transformations During this thesis the transformation of shapes is a recurring task, especially the deformation of shapes. Therefore, it is important to define the relevant terminology beforehand.

A *rigid* transformation preserves the shape information and can therefore be used to align shapes towards each other. The deformation can be decomposed in reflection, rotation and translation [Kai+11]. This transformation sustains the pairwise distance between points on the surface.

An *isometric* transformation only preserves the geodesic distances between points on the surface. The geodesic distance is defined as the shortest distance between two points along the surface [KS98][1]. Thereby, an isometric transformation also allows to bend the shape [Sah20].

If stretching or squeezing is involved, the deformation is *non-isometric*. Since the shape information gets altered, the term deformation can be used as a synonym for all non-rigid transformations.

A *homeomorph* transformation is bijective and continuous in both directions. If it is also differentiable in both directions, it is called *diffeomorph* [Lam08].

Metrics for Shape Comparison In order to evaluate a shape transformation it is often useful to measure the distance between the surfaces of two shapes, e.g. the distance between source and target shape. While there are a lot of metrics found in literature, we focus on the *Chamfer Distance* as it does not rely on predefined correspondences. It finds corresponding points simply based on the smallest distance. Therefore, it is only a suitable metric for comparing shapes that are rather similar to each other.

When applied to point sets, the metric searches for pairs of points from set $\mathbf{X}$ and $\mathbf{Y}$ with minimal distance and averages over the Euclidean distances between these pairs [Lüd+22]:

$$\mathcal{CD}_{PP}\left(\boldsymbol{X}, \boldsymbol{Y}\right) = \frac{1}{2\left|\boldsymbol{X}\right|} \sum_{\boldsymbol{x} \in \boldsymbol{X}} \min_{\boldsymbol{y} \in \boldsymbol{Y}} \left\|\boldsymbol{x} - \boldsymbol{y}\right\|_F + \frac{1}{2\left|\boldsymbol{Y}\right|} \sum_{\boldsymbol{y} \in \boldsymbol{Y}} \min_{\boldsymbol{x} \in \boldsymbol{X}} \left\|\boldsymbol{x} - \boldsymbol{y}\right\|_F \tag{1}$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm. The Chamfer distance can also be evaluated between two surfaces. Instead of point-pairs with minimal distance, we search for the point on surface $\mathcal{Y}$ which is closest to a vertex $\boldsymbol{x}$ from shape $\mathcal{X}$ and vice versa. The surface metric is more exact, but also computationally expensive since we search for the nearest points within a triangle as opposed to the nearest points within a set of discrete points. We denote the surface-to-surface Chamfer distance as $\mathcal{CD}_{SS}$. If the distance from $\mathcal{X}$ to $\mathcal{Y}$ and from $\mathcal{Y}$ to $\mathcal{X}$ are evaluated, as above, the metric can be called *symmetric*.

Methods for Shape Alignment In order to compare two or more shapes, it is often necessary to remove all rigid transformations, as they do not carry any relevant shape information. This is often done by rigidly aligning these shapes towards each other.
In this thesis, two popular methods are used for this purpose: the Iterative Closest Points (ICP)-method and the Procrustes Analysis (PA). The PA can only be used if the points are

[1] Whenever the geodesic distance is computed within these thesis, we use the following python library: `https://pypi.org/project/pygeodesic/`

in correspondence and is computed in a closed form, whereas the ICP works on correspondence free shapes in an iterative manner.
The PA aims to minimize the least-square distance between a source shape $\boldsymbol{X}$ and a target shape $\boldsymbol{Y}$ by the application of a transformation matrix $\boldsymbol{T}$ [Gow10]:

$$\min \|\boldsymbol{XT} - \boldsymbol{Y}\|^2 \tag{2}$$

If no constraints are opposed on $\boldsymbol{T}$, the solution is given by

$$\boldsymbol{T} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y} \tag{3}$$

The aligned source shape $\boldsymbol{X}_T$ is then given by $\boldsymbol{X}_T = \boldsymbol{XT}$. A special case of the PA is the Generalized Procrustes Analysis (GPA), where the classical PA is expanded in order to align more than two shapes. Hereby, a mean shape is computed in an iterative manner [Gow75].

The ICP method was proposed by Besl and McKay [BM92] to align two point sets. It consists of the following steps:

1. Match the closest source point to each reference point.
2. Compute the rigid transformation that minimizes the average Euclidean distance between the point pairs.
3. Apply the transformation to the source point set.
4. Start at 1. or finish if the Euclidean distance between the matched point pairs is below a predefined threshold.

Note, that the ICP can end in a local minimum and therefor fail, if the shapes are not roughly aligned beforehand.

2.2 Correspondence Estimation

According to Kaick et al. [Kai+11] the correspondence estimation task can be stated as the search for a meaningful relation between the elements of a set of shapes. The outcome is a mapping for each element of one shape to a semantically similar element of another shape. In order to find a meaningful relation it is necessary to understand the local as well as the global geometry of each shape and sometimes even the functionality of some shape parts. The nature of the elements mapped to each other depends on the representation of the shape. If the shapes are depicted as meshes or point clouds, the elements used for correspondence are usually (vertex-)points or the faces of the mesh.

If the output of a correspondence estimation includes only a set of selected elements (e.g. landmarks such as the tip of the nose and the corners of the eye), the correspondence is called *sparse*. If, on the other hand, the output includes all elements of the shape (e.g. all points on the surface of a face) we call it a *dense* correspondence ([Kai+11], [Sah20]). The concept of dense and spare correspondence is visualized in Figure 2.2. This thesis focuses on the estimation of dense correspondence for a set of (vertex-)points. Therefore, all other variants are neglected in the following pages. Meanwhile, Kaick et al. [Kai+11] and Sahillioğlu [Sah20] review a broader set of methods used for correspondence estimation.

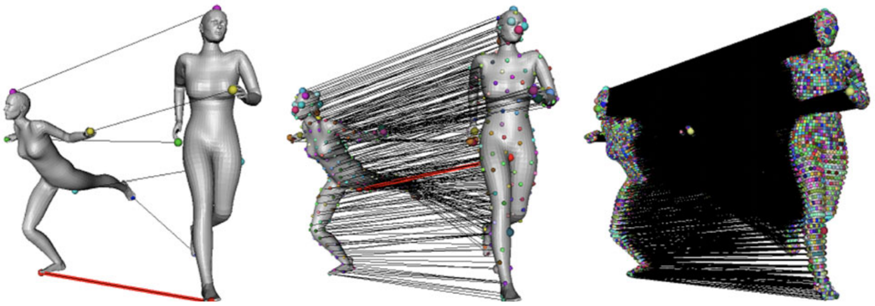


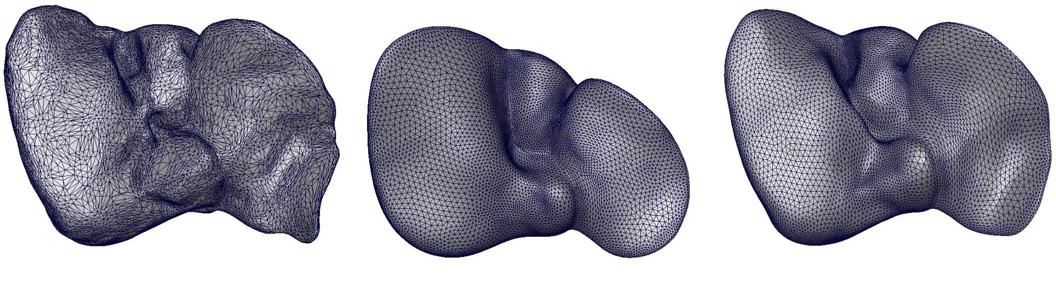
Figure 2.2: Sparse (left and middle image) and dense correspondence (right image). Images taken from [Sah20].

This chapter gives a brief overview of the classical methods used to establish shape correspondence and their strategies. Later, we focus on recent advances in unsupervised deep learning methods used for correspondence estimation. Both is important, as it generates a pool from which the methods used in this thesis are chosen. Last but not least, we discuss different criteria for the evaluation of such methods and possible applications.

2.2.1 Methods for Correspondence Estimation

Methods for correspondence estimation can be distinguished in two categories and their combination: those based on the similarity of elements ("feature mapping") and methods using the proximity between the points of aligned shapes ("registration") [Kai+11]. Similarity-based methods use descriptors to find matching elements. Descriptors can be simple characteristics such as geodesic distances [Kai+11] or more complex, such as functions based in the Laplace-Beltrami base [Ovs+12]. The descriptors get matched to each other with a so called functional map. Registration-based methods use deformations to align shapes to each other. One classical approach is the use of the ICP algorithm and its variants (e.g. [BM92], [BT00]). Affine transformations, such as the ICP method, pose a large restriction for possible deformations. This can lead to inadequate correspondences and non-homeomorphic mappings for shapes with high variances [HM09]. In order to avoid this problem, non-rigid methods have been developed, such as [BR07] and [Whi+19].

Another important distinction lies between *pair-wise* and *group-wise* approaches. While pair-wise methods consecutively match one shape to another, group-wise methods take the whole population into account when establishing correspondence. As Ravikumar et al. [Rav+18] states, group-wise methods are preferable since they are less biased, less prone to outliers and therefore more robust in general.



(a) Target shape (b) Template shape (c) Deformed template

Figure 2.3: Possible learning objective for unsupervised correspondence estimation: Deform template shape (b) to match the surface of the target shape (a), while preserving the mesh structure of the template (c). The goal is the minimization of the surface distance between deformed template and target.

Unsupervised deep learning methods Recent advances in deep learning have influenced the shape correspondence field [Sah20]. All learning-based methods are trained on a set of shapes and therefore are group-wise methods. This leads to the advantages mentioned above. Learning-based methods can be divided into supervised and unsupervised methods. Supervised methods require large datasets consisting of corresponding shapes. Since the

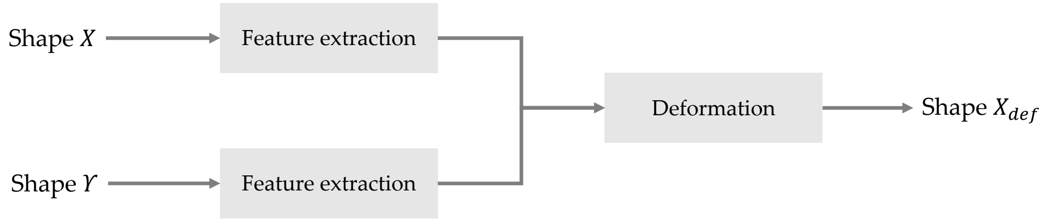


Figure 2.4: Common framework for unsupervised correspondence estimation: Firstly, features from shape $\mathcal{X}$ and shape $\mathcal{Y}$ get extracted. Based on those features, shape $\mathcal{X}$ is deformed to match the surface of $\mathcal{Y}$.

generation of this data is tedious and time-consuming, it is favorable to circumvent this problem by applying unsupervised methods. Hence, this section focuses on recent advances in unsupervised learning-based methods for correspondence estimation.

Since unsupervised methods assume that there are no labels given at all, the only information available is the geometry of the set of training shapes. But arguably a few labels are utilized, since data is often roughly aligned (e.g. top/down, left/right) or cut to the relevant shape part (e.g. only distal femur) beforehand. Most methods strive to deform one mesh into the geometry of another *target* mesh. This is essentially based on the assumption that each point moves to a semantically similar location, generating a global correspondence. One can distinguish between *target-to-target* deformations, where random shapes are deformed into each other, and *template-to-target* deformation, where only a template shape gets deformed.
The template usually has to be defined in advance, a common choice is the average shape of the training set (e.g. in [Whi+19]). Lebrat et al. [Leb+22] generate their template by taking and remeshing the convex hull of all surfaces. Thereby, most methods for template generation require some kind of correspondence beforehand, since the shapes have to be aligned towards each other. Lüdke [Lüd22] use a hub-and spokes approach for template generation that does not require a correspondence. Figure 2.3 shows the template-to-target deformation process: The objective is to deform the template shape (b) to match the surface of the target shape (a). The resulting deformed template (c) clearly has the surface of the target and the mesh structure of the template.

A typical framework used by many approaches is the one pictured in Figure 2.4. It combines the similarity-based strategy with the deformation-based strategy mentioned above: At first, features get extracted from one or both shapes, and afterwards the shapes get deformed. Differences lie in the tools and architectures used for both tasks and the terms

for the loss function. Most loss functions consist of a Chamfer distance between deformed shape and target in combination with different regularization terms (e.g. [Gro+18], [Uy+], [Lan+21], [Tra+21]).
Groueix et al. [Gro+18] were one of the firsts to employ this framework. Here, an encoder network is used for feature extraction of only the target and a decoder is trained to deform the template into the target shape, based on the extracted features. The method "NeuroMorph" proposed by Eisenberger et al. [Eis+21] is quite similar. It is based on neural networks and uses the extracted features of both shapes to build a correspondence matrix which feeds into the network used for deformations. Both networks for feature extraction and the deformation network share the same architecture based on EdgeConv-Layers ([Wan+19b]) and a simple Multi Layer Perceptron (MLP). While Neuromorph works with meshes, Zeng et al. [Zen+21] utilize only point clouds and an even simpler network architecture, but otherwise their method is quite similar to NeuroMorph. Lang et al. [Lan+21] also use a neural network for feature extraction. But afterwards they utilize the features for cross- and self-construction instead of a simple deformation.

Other authors took on the idea of functional maps and transferred it to a deep learning framework. Roufosse et al. [RSO19] employ a neural network to learn descriptor functions in the Laplace Beltrami space. Afterwards a functional map is optimized to minimize the difference between the descriptor functions. The functional map then can be viewed as correspondence matrix. Halimi et al. [Hal+19] follow a similar approach but start with SHOT descriptors [STD14] of the initial meshes. Aygün et al. [ALC20] expand this approach with geodesic distances as features and compute these distances using heat kernels. Eisenberger et al. [Eis+20] combine functional maps with extrinsic deformations in their Deep Shells Framework. Starting with SHOT descriptors as well, they use a neural network to learn features of both shapes.

Jiang et al. [Jia+21] use an entirely different approach following a *hub-and-spoke model*: At first the source and target shape get projected into a learned latent space. Here, they search for nearest template-neighbors and thereby a possible deformation from source to target is found. Another important characteristic of their "Shapeflow" method is the application of a continuous flow-field to compute the deformations. This leads, according to the authors, to smoother results with less intersections. Uy et al. [Uy+] also utilize a hub-and-spoke model in the way Shapeflow does, however they simply train their deformations on a neural network. Trappolini et al. [Tra+21] employ an attention mechanisms for template-target registration in an auto-encoder setting. Deng et al. [DYT21] use an auto-decoder setting but base their training on implicit fields instead of meshes or point clouds.

Both, deformation- and similarity-based approaches have limitations regarding very dissimilar shapes [Gro+19]. The concept of cycle-consistency tries to address this challenge. In order to enforce cycle-consistency, a point has to be mapped to its original location after a cycle of deformations [Che+21]. Zhou et al. [Zho+16] apply this concept to 2D images and Groueix et al. [Gro+19] for 3D point clouds.

Shape Correspondence in Medicine The methods mentioned above are applied to general computer vision tasks. The objects under examination are usually human bodies in different poses or everyday items such as chairs, cars or different animals. These objects can have a lot of dissimilarities (think of chairs with or without armrests, with three or four legs etc.). Most medical shapes have a higher degree of similarity. This paragraph therefore focuses on methods used to estimate correspondences of anatomical shapes, such as different organs or bones.

For the sake of constructing SSMs, there are some off-the-shelf tools for correspondence estimation. According to Goparaju et al. [Gop+22], Deformetrica, SPHARM-PDM and Shapeworks are widely used today. Deformetrica is based on the Large Diffeomorphic Deformation Metric Mapping (LDDMM) framework proposed by Durrleman et al. [Dur+14]. SPHARM-PDM utilizes the SPHARM method developed by Brechbühler et al. [BGK95] and Shapework apply their own particle-based method [CEW17]. The mathematical background of these papers is beyond the scope of this thesis and can on interest be explored in the cited works.
There are a lot of other classical approaches used to bring anatomical shapes in correspondence. However, there is a significant loss of group-wise methods. A lot of works focus on the registration of 2D images, which is not the scope of this thesis. Mambo et al. [MDH18] reviews these methods. Some works base their training on images (mostly CT and MRI data) and output 3D data such as point clouds ([Rav+18], [Agi+20], [Dal+19], [Bay+19]).

2.2.2 Applications for Correspondence

A big field of application for shape correspondence is the field of virtual reality, where correspondence can be used to transfer detected motions onto the avatar or other objects [ALC20]. Since shape correspondence is a classical problem in computer vision in general, it is needed for a lot of downstream tasks [DYT21]. This includes the morphing of shapes ([Kai+11], [Eis+21]), the editing of shapes [DYT21], or information transfer. The latter includes the transfer of deformations [SP04], style [Xu+10] and texture ([KS04], [DYT05], [DYT21]). Other possible utilizations include mobility analysis [Wan+19a] and robotic grasping [Mil+03]. The biggest field of application in a medical context is the use for SSMs and their various applications mentioned in Section 2.3.2. But shape correspondence can also be utilized for change detection, as done e.g. by Nguyen et al. [Ngu+14] for the skeletal changes of the mandible.

2.2.3 Evaluation of Dense Correspondence

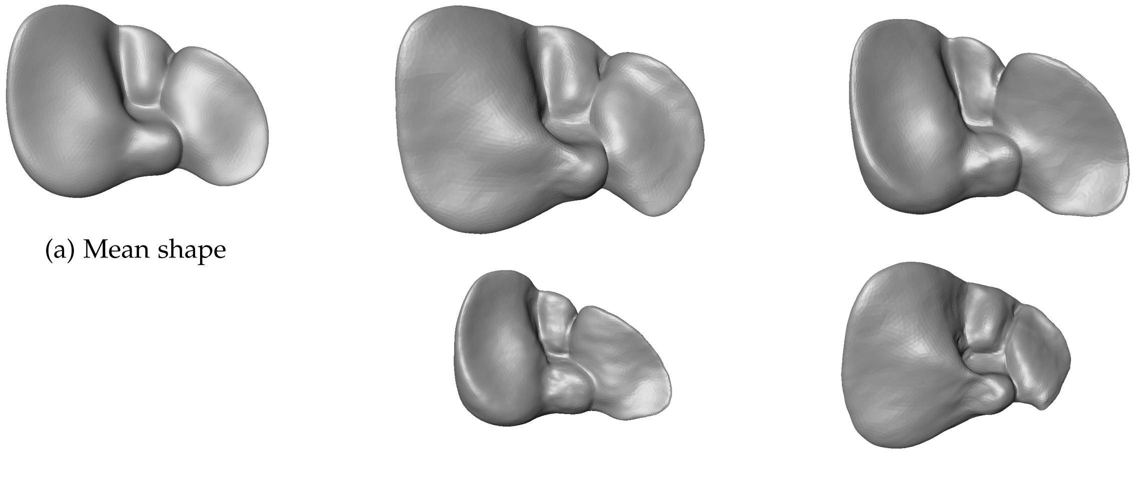
The easiest way to evaluate correspondence is the use of a so called ground truth correspondence. According to Kaick et al. [Kai+11] this can be achieved with two different metrics: The Hamming loss (number of correctly matched points) and the endpoint error (distance from matching point to its known ground truth). Especially in the medical field, the use of ground truth correspondence is controversial, since it is generally not known and results of handmade correspondence are often not reproducible [HM09].
Kaick et al. [Kai+11] also offer a set of other (indirect) metrics to assess the quality of a correspondence. As we lack ground truth information for the data used in this thesis, some of these metrics are applied later on. Note, that most metrics rely on the datasets at hand. Some, like the previously mentioned ground truth, even require special annotations, so-called *labels*.

One possible metric is the use of the output of the objective function used to generate the correspondence. The underlying assumption is that this metric is proportional to the accuracy of the estimated correspondence. Obviously, this can only be applied when the methods under observation use similar objective functions.
Especially in cases where a comparison between different methods for correspondence estimation is desired, the ground truth correspondence on benchmarking datasets can be utilized for evaluation. Standard datasets used for this purpose include FAUST [Bog+14] and SHREC'20 [Dyk+20].

According to Kaick et al. [Kai+11], it is common to assess the quality of generated correspondences on anatomical data on behalf of the quality of resulting SSMs. Metrics to do so are covered in Section 2.3.3.
Another possibility is the evaluation of the performance of the application itself. Depending on the use-case, the evaluation is often only qualitative e.g., the quality of transferred textures. As this thesis focuses on anatomical data, the construction of SSMs is an important application. Another possible application is the automatic location of anatomical landmarks. The evaluation of such a sparse correspondence requires appropriate labels for the whole dataset.

2.3 Statistical Shape Models

Statistical Shape Models are used to describe a class of semantically similar objects in a compact way. They capture the average shape as well as the geometric variations that occur in the training population, the latter usually in the form of eigenmodes [HM09]. Figure 2.5 shows parts of an SSM of the liver. While (a) shows the mean shape, (b) and (c) visualize possible variations imposed by the first two eigenmodes. This section covers the construction of SSMs as well as a broad overview of medical applications and metrics to evaluate the quality of different SSMs.



(a) Mean shape

(b) First eigenmode

(c) Second Eigenmode

Figure 2.5: Statistical Shape Model of the liver: Mean shape (a) and first two eigenmodes (b and c) of the Principal Component Analysis (PCA). The first eigenmode has a large impact on the size of the left part of the liver, whereas the second mode mainly effects the size of the right part.

2.3.1 Construction of SSMs

Most methods for the construction of surface based SSMs rely on training shapes with dense correspondence. Methods to establish dense correspondence are discussed in Section 2.2. In order to find the shape variations based on the actual geometry and not on the position in Euclidean space, it is necessary align the shapes beforehand. The rigid transformations applied for shape alignment are usually rotation and translation and sometimes even rescaling. Depending on the medical application, size is an important biological variation and should therefore be preserved during shape alignment [HM09]. The most popular method to align corresponding shapes is the GPA mentioned above [SWZ14].

Point Distribution Models In 1995 Cootes et al. [Coo+95] introduced Point Distribution Models (PDMs), which are still the most common SSMs in use today ([Ber+17], [Li+21], [Tót+20], [Amb+19a]). PDMs are based on a PCA to reduce the dimensionality of the shape representation and result in a mean shape $\boldsymbol{M}$ and a set of sorted eigenmodes $\boldsymbol{e}_m$ to capture the variations of the population. Each (un-)seen shape can now be approximated by a linear combination of weighted eigenmodes.

The mean shape $\boldsymbol{M}$ of N training shapes is defined as the average of the coordinates of each of the k vertices $\boldsymbol{X}_i$ over all training meshes:

$$\boldsymbol{M} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{X}_i \in \mathbb{R}^{3k}. \tag{4}$$

Here it is important to note that this formula only applies if the training shapes are in dense correspondence, i.e. if each vertex is at a semantically similar position on every shape. With the mean shape $\boldsymbol{M}$ it is possible to compute the covariance matrix $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Sigma} = \frac{1}{N-1}\sum_{i=1}^{N}(\boldsymbol{X}_i - \boldsymbol{M})(\boldsymbol{X}_i - \boldsymbol{M})^T. \tag{5}$$

The eigendecomposition of $\boldsymbol{\Sigma}$ leads to the eigenvectors $\boldsymbol{e}_m$ and -values λ_m needed to represent the shape variations. The $s = \min(3k, N) - 1$ eigenvalues have to be sorted in descending order: $\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_s$, as higher eigenvalues correspond to more important variations in shape. The sum of all eigenvalues can be interpreted as the total variance V_T of the model:

$$V_T = \sum_m \lambda_m \tag{6}$$

An important aspect in the design of a PDM is the number of eigenvectors t used to build the model. The parameter t is often chosen in order to capture an accumulated variance that represents a defined ratio p_t of the total variance [Coo20]:

$$\sum_{m=1}^{t}\lambda_m = p_t V_T. \tag{7}$$

Common values for p_t range between 0.9 and 0.98. Other strategies for the selection of t are mentioned in Heimann and Meinzer [HM09]. The linear combination of t eigenvectors used to describe unseen shapes can be written as:

$$\boldsymbol{Y}_i = \boldsymbol{M} + \sum_{m=1}^{t} b_m \boldsymbol{e}_m, \tag{8}$$

with shape weights $b \in \mathbb{R}^t$:

$$b = \boldsymbol{Q}^T (\boldsymbol{Y}_i - \boldsymbol{M}) \tag{9}$$

and

$$\boldsymbol{Q} = (\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_t) \in \mathbb{R}^{3m \times t}. \tag{10}$$

When all eigenvectors are used, the training shapes can be represented perfectly. Equation (8) can also be used to generate new shapes by varying the shape weights b_m. In order to get realistic shapes similar to the training set, the range for b_m has to be limited. A popular way for doing so is to limit b_m within the range of three standard deviations around the mean [Coo+95]:

$$-3\sqrt{\lambda_m} \leq b_m \leq 3\sqrt{\lambda_m} \tag{11}$$

Other Methods for SSM Construction While the PDM proposed by Cootes et al. [Coo+95] is the most popular SSM to this date, there are other linear and non-linear methods for SSM construction available.
PDMs face a lot of challenges when applied to medical data. One challenge regards the global nature of the PCA: variations are captured on a global level. Meanwhile a lot of medical use-cases are interested in local shape anomalies, indicating diseases and other pathologies. Among others Lecron et al. [Lec+12] try to solve this issue with a multilevel part-based PCA approach. Another way to solve this issue is the use of sparse mode PCAs, as applied by Sjöstrand et al. [SSL06] or the use of Independent Component Analysis (ICA) [Sui+04]. Here it is important to note that methods which are not PCA-based usually do not allow for an ordering of modes which hinders a compact model representation.

Another challenge is the limitation on possible deformations of the mean shape since twisting and bending cannot be captured with linear statistical methods [HM09]. An obvious solution to meet this problem is to expand the linear PCA to a non-linear version. This is often done using kernel PCA ([Wan14], [Ma+19], [RDT06], [KBW11]). A downside of the kernel PCA is the choice of a suiting kernel, which is often non-trivial and dependent on the application. Other non-linear approaches make use of the so called *Principal Geodesic Analysis* (e.g. [Fle+04], [Tyc+18]) which considers geodesic distances between the shapes instead of Euclidean distances [Bru+14].

In contrast to all the other methods mentioned above Lüdke et al. [Lüd+22] do not require training shapes in predefined dense correspondence. It is sufficient, if all shapes are roughly aligned according to the principal axes. Their method "FlowSSM" uses a neural network to estimate correspondence and performs a PCA in latent space to construct an SSM. A more detailed description is given in Section 3.1.1, as it is part of the comparison in this thesis.

2.3.2 Medical Applications for the Use of SSMs

While PDMs initially were developed to locate (anatomical) shapes in 2D images [Coo+95], today there is a wide range for medical applications of SSMs. Therefore, this section gives only a broad overview of possible applications.

Segmentation and Reconstruction As mentioned above, SSMs can be used to generate plausible unseen shapes. This generative capability can be used in segmentation and reconstruction tasks, where SSMs serve as a so-called geometric prior that regularizes the deformation to stay within the respective shape space of the learned population. Heimann and Meinzer [HM09] give an overview of SSMs used in segmentation tasks. Shape completion is needed, when the underlying data is sparse or erroneous. SSMs can be used to interpolate the data in accordance with the known population. Another possible reconstruction task is the generation of a healthy version of a pathological shape. Sarkalkan et al. [SWZ14] gathered various examples for SSM-based bone reconstructions. Lecron et al. [Lec+12] fit SSMs to a spine with instruments installed to deform and straighten the spine. The resulting model gives insights on the geometry of the spine without these instruments. Ma et al. [Ma+19] use SSMs to deal with erroneous data of kidneys and ankle bones. As Ambellan et al. [Amb+19b] pointed out it is also possible to approximate 3D shapes based on only a few 2D images. This can lead to a significant reduction of radiation exposed to the patient.

Shape Analysis and Diagnosis The compact encoding of SSMs can be used to capture characteristic alterations of shapes. In combinations with Machine Learning (ML) methods this can be utilized for a better analysis of the shapes. As outlined by Ambellan et al. [Amb+19b] unsupervised methods can be used to cluster the data into different subgroups. This can be diseases-specific groups, but also groups based on demographic features such as age and sex. Supervised methods on the other hand can train a classifier for disease classification tasks. In literature SSMs were applied to classify osteoarthritis ([Tyc+18], [AZT21], and papers mentioned in [SWZ14]) and Alzheimer's disease [AZT21]. On the other hand Bruse et al. [Bru+17] use an SSM to cluster different aortic arch shapes.

Other Applications Sarkalkan et al. [SWZ14] and Goparaju et al. [Gop+22] show multiple ways where SSMs can be used for surgery planning and implant design. And as Ambellan et al. [Amb+19b] pointed out, SSMs can also be utilized for education as it could help students to get a better understanding of the variations of "normal" anatomical structures. Tang et al. [Tan+19] have used SSMs for data augmentation.

2.3.3 Evaluation of the Quality of SSMs

Later on, we compare different methods used for correspondence estimation of anatomical shapes. As discussed previously, we use the quality of the emerging SSMs as a proxy to evaluate the quality of the underlying correspondences. It is therefore important to be able to evaluate the quality of the resulting SSMs as an indirect metric to measure the underlying correspondence quality.

According to Davies et al. [DCT01] an ideal shape model has the following properties: generalization ability, specificity and compactness. These *intrinsic* criteria are the ones most often used for the evaluation of SSMs:

- **Generalization Ability:** The ability of the SSM to represent unseen shapes. In order to compute this measure, a previously unseen shape is projected into the PCA-space and reconstructed. Now we compute the Euclidean distance between the original and reconstructed points [Bru+14].
- **Specificity:** The ability of the SSM to generate new and valid instances of the shape family. Specificity can be quantified by randomly generating a large number of samples and computing the Euclidean distance to the corresponding points of the closest training shape [Gop+22].
- **Compactness:** The number of parameters needed to model the variability of the model. This criterion mirrors Occam's razor principle "a simple explanation is more likely to be better than a complicated explanation" [Gop+22], indicating that fewer parameters (eigenvectors) are preferable. The compactness can be computed as the sum of all eigenvalues whose corresponding eigenvectors were used to build the model ([Gop+22]).

Unfortunately, Ericsson and Karlsson [EK07] proved that a low-quality correspondence can still lead to excellent results in the standard metrics used to measure the quality of SSMs mentioned above. In order to reduce this risk, Ravikumar et al. [Rav+18] proposed other metrics to evaluate the quality of SSMs and of the underlying correspondence. One possibility is the use of anatomical landmarks defined by a medical expert. The underlying assumption is that a good correspondence on these points indicates a good correspondence on all other points and that the ground truth of the sparse annotations holds truth. Another option is the evaluation of the SSM performance of an actual application, such as a clustering or classification task. Munsell et al. [MDS08] try to circumvent the risk of good SSM results based on a low quality correspondence by applying the correspondence estimation method to an synthetic 2D dataset with known ground truth correspondence. But as Ravikumar et al. [Rav+18] pointed out, it is not feasible to expand this approach to the 3D medical domain.

2.4 Summary

Shape correspondence can be described as a semantically meaningful mapping between instances of two or more shapes. Methods used for correspondence estimation are either similarity-based, deformations-based or a combination of both. Affine registrations, such as the ICP, fail to generate adequate results for shapes with high variations. Recent advances in computational power and deep learning methods opened the field to data-driven methods. Those methods are trained on the whole population of the training set and therefore called group-wise methods. It is assumed that these group-wise methods are preferable to classical pair-wise methods [Rav+18]. Since the manual generation of ground truth correspondence labels is quite tedious and not reproducible, unsupervised learning-based methods are favorable as they do not need these labels. Most of these methods deform shapes into each other and use the Chamfer distance in combination with regularization terms as a loss function. Advances in unsupervised learning based methods have not been evaluated on the medical data yet.

The evaluation of correspondence is quite complicated, since the ground truth is generally unknown. Metrics used in literature include the evaluation towards handmade correspondence, the evaluation of arbitrary metrics such as the Chamfer distance or the evaluation of performance on the application. The main applications for correspondences of anatomical shapes are hereby the construction of SSMs and automatic landmark localization. The quality of SSMs is mostly evaluated in terms of generality, specificity and compactness.

3 Materials and Methods

The goal of this thesis is the estimation of shape correspondences for anatomical shapes. This is done with two experiments: a comparison of a selection of existing methods used for correspondence estimation and the further development of FlowSSM. The methods selected for the comparison are presented in Section 3.1, the modification of FlowSSM is described in Section 3.2.

In order to evaluate the resulting correspondences, an experimental set-up is needed and to ensure a better comparability, the set-up for both experiments is the same. The datasets used in this set-up are described in Section 3.3. Afterwards, we take a look at the metrics that can be used to evaluate the resulting correspondence in Section 3.4. A proper assessment is quite a challenging task, as there is no ground truth available.

3.1 Methods from Literature considered for Comparison

We presented a wide field of different approaches to estimate shape correspondences in Section 2.2.1. In this section we select a few of these methods to investigate their performance on anatomical data. As stated before, there is no reproducible ground truth for correspondence on anatomical shapes. Thereby, all methods selected in this section do not require such a ground truth and are therefore unsupervised.

In order to ensure a better comparability, all methods investigated follow the template-to-target deformation approach. The goal of these methods is therefore to input randomly meshed target shapes and output a deformed template which fits to the surface of the target. It is then assumed that the templates consist of vertices which are in correspondence to the vertices of all other deformed templates and this correspondence is evaluated and discussed in the next chapter.

3.1.1 FlowSSM

The first method chosen for the comparison is "FlowSSM" by Lüdke et al. [Lüd+22]. The method was initially proposed to construct SSMs, but can also be used to generate dense correspondences. Since the resulting SSMs were of high quality, it is a promising approach to evaluate the underlying correspondence. It is also one of the only data-driven and therefore group-wise methods that can be used for correspondence estimation in the medical field known to the author. Furthermore, the method is closely related to "ShapeFlow" by Jiang et al. [Jia+21], as both methods rely on the integration of a deformation flow. In the broader computer vision domain, ShapeFlow attained good results when compared to other unsupervised methods used for correspondence estimation (see e.g. [Eis+21], [DYT21], [Jia+21]). This raises expectations on the quality of the correspondences estimated by FlowSSM.

FlowSSM takes meshes as input, but works with points evenly sampled on the surface. The idea is to deform each point on a template $\boldsymbol{x}_0$ to represent the surface of the target shape. The deformation function Φ describes the deformation trajectory of each point $\boldsymbol{x}_0$ during the time $\tau \in [0,1]$:

$$\Phi(\boldsymbol{x}_0, \tau) = \boldsymbol{x}_0 + \int_0^\tau \boldsymbol{v}_\theta(\boldsymbol{x}(t))dt \tag{12}$$

Here $\boldsymbol{v}(\cdot)$ represents the velocity field, which can be described as

$$\boldsymbol{v}_\theta(\boldsymbol{x}(t), t) = \boldsymbol{f}_\theta(\boldsymbol{x}(t), t \cdot \boldsymbol{z}) \cdot \|\boldsymbol{z}\|_2. \tag{13}$$

The flow function $\boldsymbol{f}$ is parametrized by an MLP and $\boldsymbol{z}$ represents a shape-specific latent vector. In order to the broaden the deformation's frequency spectrum, the process is divided into a global and a local deformation, each with their own MLP. The global deformation uses global latent vectors, whereas the local latent vectors are defined by a sum of M weights z_k:

$$\begin{aligned} \boldsymbol{z}(\boldsymbol{x}) &= \sum_{k=1}^{M} z_k \varphi_k \left(\|\boldsymbol{c}_k - \boldsymbol{x}\|_2 \right) \\ \varphi_k(r) &= e^{-(\varepsilon_k \cdot r)^2} \end{aligned} \tag{14}$$

where $\boldsymbol{c}_k$ are control points and ε_k is an inverse Gauss kernel width. While z_k and $\boldsymbol{c}_k$ are learned in training, ε_k has to be determined by the user. The loss function used for training is the pointset to pointset Chamfer distance.

After training, an SSM is build with the common PDM approach. In contrast to the common versions, the PCA is performed on the latent vectors and not the vertices of the deformed template. However, this thesis uses only the deformed template meshes and neglects the resulting SSM. This enables us to only evaluate the quality of the resulting correspondences and not the (possible) advantage of performing a PCA in latent space. Furthermore, the

SSM build by FlowSSM requires unseen shapes to be processed in an inference-step in order to optimize the latent vectors. This additional step is time-consuming and requires a Graphics Processing Unit (GPU). It can be avoided when only the learned correspondences are used for SSM construction as done in this thesis.

Implementation The python implementation used in this thesis is a predecessor of the version publicly available on Github[1]. As the method was already used with the distal femur and liver dataset, we reused the recommended settings. The faces dataset lacks this advantage. We re-used the settings from the liver data, as parameter-tuning is a time and energy intensive task. As we gain results comparable to other state of the art methods such as Grewe and Zachow [GZ16] (see Section 4.1.3), we argue that the parameter choice is appropriate. All deviations from the default settings can be found in Table 3.1. The parameter ε_k refers to the inverse Gauss kernel width mentioned in equation (14) while the acronym LOD represents the level of detail of the local and global deformer, respectively.

3.1.2 NeuroMorph

Another groupwise method with competitive results when compared in literature is "NeuroMorph" by Eisenberger et al. [Eis+21]. Since it has never been applied to anatomical data, it holds great potential for this comparison. Furthermore, its structure is related to FlowSSM, which simplifies the comparison.

The method itself is not template-based but strives to deform random shapes (e.g. $\mathcal{X}$ and $\mathcal{Y}$) into each other. As a first step, a neural network is used to extract features based on the vertices of each shape (e.g. $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively). A correspondence matrix $\boldsymbol{\Pi}$ matches the pairs with most similar features using a softmax operator. Afterwards, an interpolator network is used to compute the deformation. The input variables for the interpolator are the vertices $\boldsymbol{X}$, the correspondence based offset $\boldsymbol{\Pi Y} - \boldsymbol{X}$ and a time variable t. The output consists of the deformed vertices $\boldsymbol{X}_T$. The neural networks for feature extraction and interpolation are of the same architecture, which consists mainly of five EdgeConv-Layers [Wan+19b] and an MLP. The loss term used for training is composed of three different loss functions:

1. The registration loss: $||\boldsymbol{\Pi Y} - \boldsymbol{X}_T||_2^2$.
2. The as-rigid-as-possible loss, which strives to restrict the interpolation to possible sequences by minimizing distortions (see [SA07] for more details).

[1] https://github.com/davecasp/flowssm

3. The geodesic distance preservation loss: $\|\boldsymbol{\Pi} \boldsymbol{D}_{\mathcal{Y}} \boldsymbol{\Pi}^T - \boldsymbol{D}_{\mathcal{X}}\|_2^2$, where $\boldsymbol{D}_{\mathcal{X}}$ and $\boldsymbol{D}_{\mathcal{Y}}$ are geodesic distance matrices which have to be computed before training.

The output of the method consists of deformed vertices on every step of the interpolation and the correspondence matrix $\boldsymbol{\Pi}$. If the vertices are used for further downstream tasks, it is recommended to apply the post-processing method "Smooth Shells" by Eisenberger et al. [ELC20].

Implementation We use the python implementation of NeuroMorph found on Github[2] and apply the default settings. As the pre-processing is implemented in Matlab, we re-implemented it in python. For the post-processing we used the Matlab implementation of SmoothShells[3] with the default settings.

When applied to the use-case of building corresponding meshes for SSMs, it is only necessary to deform a template towards varying targets. Therefore, the method was adapted to only train deformations of the template and not all training shapes. Both versions are tested within this thesis.

NeuroMorph deforms every possible data combination in each training epoch. If a dataset has n targets, this results in 2^n possible combinations. Since the version with the template has a reduced variability of training data as there are only n possible combinations, we increased the number of training epochs to account for this disadvantage. Furthermore, the training procedure of NeuroMorph changes some settings during the training. The parameter n_{inc} determines when these changes are applied and was adjusted towards the increased number of epochs. As the time needed for training increases dramatically with the number of training shapes and therefore possible combinations, we limited the maximum training time to one week. This leads to the reduced number of epochs of the distal femur training. The changed settings are summarized in Table 3.1.

3.1.3 LDDMM

The third method chosen for comparison is the LDDMM Framework as used by Deformetrica [Dur+14]. Since this method is already established in the medical domain, it serves as a baseline for the state of the art in use today. In contrast to the other two methods, LDDMM is not data-driven but rather a simple optimization problem. It is therefore a pair-wise

[2] https://github.com/facebookresearch/neuromorph

[3] https://github.com/marvin-eisenberger/smooth-shells

method. The goal is to minimize the Varifold distance d_W between target $\mathcal{Y}$ and deformed template $\mathcal{X}$:

$$\min d_W(\phi(\mathcal{X}), \mathcal{Y}) = \min(\langle \phi(\mathcal{X}), \phi(\mathcal{X}) \rangle_{W^*} + \langle \mathcal{Y}, \mathcal{Y} \rangle_{W^*} - 2 \langle \phi(\mathcal{X}), \mathcal{Y} \rangle_{W^*}) \tag{15}$$

The inner product between two meshes is given as:

$$\langle \mathcal{S}, \mathcal{S}' \rangle_{W^*} = \sum_p \sum_q K^W \left(\boldsymbol{c}_p, \boldsymbol{c}'_q \right) \frac{\left(\boldsymbol{n}_p^T \boldsymbol{n}'_q \right)^2}{\left| \boldsymbol{n}_p \right| \left| \boldsymbol{n}'_q \right|} \tag{16}$$

where K^W is a kernel with width σ_W and $\boldsymbol{c}_p$ and $\boldsymbol{n}_p$ denote the centers and normals of the faces.

The *diffeomorphism* ϕ in equation (15) represents the deformation applied to the template and is defined for a point $\boldsymbol{x}$ on the surface as $\phi(\boldsymbol{x}) = \boldsymbol{x} + v(\boldsymbol{x})$. The velocity field v at a given time $t \in [0,1]$ is defined as:

$$\dot{\mathbf{X}}(t) = v_t(\boldsymbol{X}(t)) = K(\boldsymbol{X}(t), \boldsymbol{c}(t))\boldsymbol{\alpha}(t) \text{ with } \boldsymbol{X}(0) = \boldsymbol{X}_0 \tag{17}$$

where K represents a kernel. It is dependent on the control points $\boldsymbol{c}$ and weights $\boldsymbol{\alpha}$. Therefore, this method is again based on the integration of a deformation flow.

Implementation The implementation used in this thesis is Deformetrica[4]. Only the kernel width σ_W was altered to gain better results on each dataset and use the default parameters otherwise. The chosen values are listed in Table 3.1.

3.1.4 Meshmonk

Last but not least we evaluate the performance of Meshmonk [Whi+19]. Meshmonk was developed to provide a publicly available tool for dense correspondence estimation which is even for non-experts easy to use. This low-level usability in combination with promising results on anatomical shapes reported in literature were the reason for its selection in this thesis.

Meshmonk strives to deform a template towards a target by applying both, an ICP and a Non-rigid Iterative Closest Points (NICP) algorithm. It is therefore a pairwise method. The ICP is used to roughly align both shapes. The NICP starts with a symmetrical weighted

[4] https://www.deformetrica.org/

k-nearest neighbor search to initialize the correspondence. Afterwards possible outliers are detected and removed. Finally, a visco-elastic transformation is applied. This process is repeated in a iterative manner, until either a pre-defined surface distance between template and target is reached, or a pre-set number of iterations was performed.

Implementation We used the C++ implementation available on Github[5]. As all shapes are already aligned towards the template with an ICP, we did not repeat this step and applied only 80 iterations of NICP. Other than this adjustment, we used the default settings on all other options.

Table 3.1: Parameter used on each dataset for each method.

Method	**Parameter**	**Distal Femur**	**Liver**	**Face**
FlowSSM	ε_k	0.8421233433207136	0.4615228342424673	0.4615228342424673
	LOD	[1,7]	[1,5]	[1,5]
	$n_{samples}$	15000	12000	12000
Neuromorph	# epochs	240	600	600
	n_{inc}	150	300	300
Neuromorph on temp.	# epochs	1200	1200	1200
	n_{inc}	600	600	600
LDDMM	σ_W	7	2	7
Meshmonk	n_{it}	80	80	80

[5] https://github.com/TheWebMonks/meshmonk

3.2 FlowSSM with Geodesic Distance Preservation Loss

FlowSSM has a few advantages when compared to NeuroMorph. For one, there is no need for elaborate pre- and postprocessing methods. Furthermore, it generates smooth, intersection-free meshes due to the flow integration. As FlowSSM was initially not conceived to learn shape correspondence, there is still some potential to improve its results. For these reasons, the following section examines an attempt to improve the correspondences generated by FlowSSM.

3.2.1 Methodology

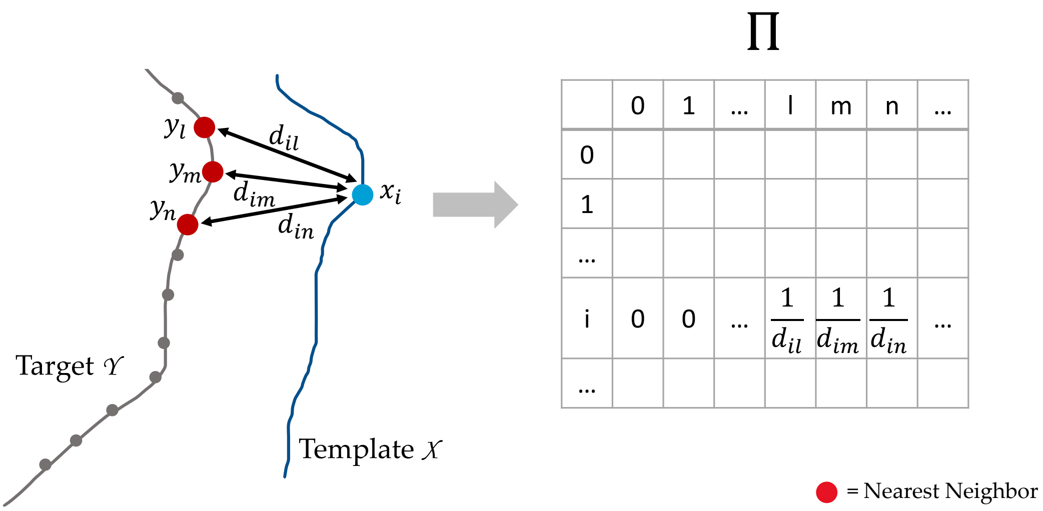
There are not many shape features in the literature that can be used for unsupervised training of correspondence estimation. One approach commonly followed is the usage of the Geodesic Distance Preservation Loss. Since it is not yet implemented for FlowSSM, we want to test this approach.

The idea behind the Geodesic Distance Preservation Loss (GDPL) is to promote correspondences that generate isometric maps between template and target shapes. This is done by the minimization of the difference between the pairwise geodesic distances between points on the template and the target under a given correspondence. The loss term can lead the neural network to prefer isometric deformations to non-isometric ones, as geodesic distances are maintained during isometric deformations (see Section 2.1). In an extreme example, the loss function causes a transformation of the vertices of the nose to a new location rather than flattening the nose on the originally location and recreating it somewhere else on the face. As such the loss can encourage a good correspondence, as the original "nose vertices" will stay on the nose. Furthermore, FlowSSM does not directly optimize any correspondence-specific properties. This loss function fills this gap by adding the optimization of a correspondence matrix $\mathbf{\Pi}$ to the training objective.

Regarding the implementation of the loss function, we apply a formulation frequently found in literature (e.g. [Eis+21], [Hal+19]):

$$\mathcal{L}_{geod} = \|\mathbf{\Pi} \boldsymbol{D}_{\mathcal{Y}} \mathbf{\Pi}^T - \boldsymbol{D}_{\mathcal{X}}\|_2^2, \tag{18}$$

where $\boldsymbol{D}_{\mathcal{X}}$ and $\boldsymbol{D}_{\mathcal{Y}}$ represent the pairwise geodesic distance matrices of shape $\mathcal{X}$ and $\mathcal{Y}$, respectively. In our implementation the shape $\mathcal{X}$ represents the template, while the shape $\mathcal{Y}$ denotes the different target shapes.



	0	1	...	l	m	n	...
0							
1							
...							
i	0	0	...	$\frac{1}{d_{il}}$	$\frac{1}{d_{im}}$	$\frac{1}{d_{in}}$	...
...							

Figure 3.1: Illustration of the general procedure to estimate Π: at first, the k nearest neighbors of template vertex x_i on the target y_l, y_m, y_n are found. Π is initially filled with zeros. In the i-th row of Π, the indices of the nearest neighbors l, m and n are filled with the inverse distances between the template vertex and nearest neighbors on the target mesh.

In order to be able to use Equation 18 as a loss term, it has to be differentiable. As the geodesic distance matrices are pre-computed, the differentiability only depends on the correspondence matrix $\mathbf{\Pi}$.

Estimation of the Correspondence Matrix The matrix $\mathbf{\Pi}$ can be interpreted as a correspondence matrix, e.g. a matrix mapping each vertex of shape $\mathcal{X}$ to its corresponding vertex on shape $\mathcal{Y}$. In literature the correspondence matrix is often based on extracted features. Since FlowSSM does not use feature extraction directly, we approximate a soft correspondence matrix based on the k nearest neighbors of the deformed templates vertices on the target surface:

We initialize $\mathbf{\Pi}$ as a $n \times m$ zero matrix to define the mapping between n template vertices and m target vertices. Afterwards, we enter values for the k nearest neighbors of each template vertex in order to approximate each point on the deformed template by its nearest neighbors on the target. In order to do so, we fill in the inverse distance to the nearest neighbors, e.g. if vertex y_j of the target is one of the k nearest neighbors of vertex x_i on the template shape, we fill $\mathbf{\Pi}$ the following way:

$$\mathbf{\Pi}_{i,j} = \frac{1}{(\|x_i - y_j\|_F)^{\sigma}} \tag{19}$$

Here, $\|\cdot\|_F$ represents the Frobenius norm and the parameter σ can be used to shift more weight to the closest neighbor. Finally, we normalize each row of $\mathbf{\Pi}$ to get a sum of one. The process for the estimation of $\mathbf{\Pi}$ is illustrated in Figure 3.1.

It is important to note, that this loss function can never reach zero. That is, because the correspondence matrix is not one-hot but maps each vertex of $\mathcal{X}$ to its k nearest neighbors. Furthermore, even with a perfect mapping the difference between the geodesic distance matrices cannot be zero unless they are exactly the same. This would only be the case if $\mathcal{X}$ is an isometric mapping of $\mathcal{Y}$, which is highly unlikely for anatomical shapes of two different patients.

We use the GDPL in combination with the previously applied Chamfer distance loss $\mathcal{L}_{CD}$ of FlowSSM and utilize the parameter α to adjust the weight of the GDPL:

$$\mathcal{L}_{total} = \mathcal{L}_{CD} + \alpha \mathcal{L}_{geod} \tag{20}$$

3.2.2 Implementation

The implementation of the GDPL faces a few challenges. While FlowSSM originally samples points on the surfaces of each mesh, the GDPL requires a (fixed) set of vertex points with a pre-computed pairwise geodesic distance matrix. FlowSSM benefits from the sampled points as they differ in each epoch and therefore increase variation within the training set. In order to keep this advantage, we apply the Chamfer distance loss $\mathcal{L}_{CD}$ to the sampled points, and use the vertices of the original meshes to compute the GDPL:

For the calculation of $\mathbf{\Pi}$ we utilize the deformed sampled points of the template and search for the nearest neighbors between the original target vertices. Since $\mathbf{\Pi}$ is now defined towards the sampled points of the template and not its vertices we cannot use the pairwise geodesic distance matrix of the template $\boldsymbol{D}_{\mathcal{X}}$ as originally intended. Instead, we have to adapt it by using another correspondence matrix $\mathbf{\Pi}_X$:

$$\hat{\boldsymbol{D}}_{\mathcal{X}} = \mathbf{\Pi}_{\mathcal{X}} \boldsymbol{D}_{\mathcal{X}} \mathbf{\Pi}_{\mathcal{X}}^T \tag{21}$$

The matrix $\mathbf{\Pi}_{\mathcal{X}}$ is similarly computed as $\mathbf{\Pi}$ and therefore mainly relies on the nearest vertices of each sampled point on the surface of the template. Contrary to $\mathbf{\Pi}$, the computation of $\mathbf{\Pi}_{\mathcal{X}}$ always relies on the 3 nearest neighbors. The adapted distance matrix $\hat{\boldsymbol{D}}_{\mathcal{X}}$ is computed in the dataloading step and therefore never alters during training.

Another advantage of the use of points sampled on the surface is the independence to the numbers of vertices on the mesh. As data is often processed batch-wise during training, it is important that all matrices have the same size for every target mesh. Therefore, when using the GDPL we have to decrease every mesh to the size of the smallest mesh in the dataset. The vertices to be removed are randomly selected in each epoch. The number of vertices used for every dataset is listed in Table 3.2.

The number of gradients used for backpropagation is increased by the GDPL dramatically, as $\mathbf{\Pi}$ is a $n \times m$ matrix. This is beyond the computing capacity used in this thesis, especially for the liver and distal femur dataset. An obvious action would be to decrease the batch size. However, it was observed that this alteration decreases the quality of the training results significantly. Therefore, we choose another strategy and decrease the number of points sampled on the template surface by 75% when the liver and distal femur datasets are used. Unfortunately, this still leads to a small decrease in the training results. Therefore we use this setting without the active GDPL to compute a new baseline model for each dataset.

Selection of Hyperparameters The employment of the GDPL leads to five new data-specific hyperparamters. The first is the number of vertices of the smallest mesh in the dataset $n_{samples}$. The second is the weight α of the GDPL in the total loss term. This parameter is chosen in order to bring the GDPL approximately to the same level as the Chamfer distance loss. Similar to Eisenberger et al. [Eis+21] it was observed that the GDPL can restrict necessary non-isometric deformations. This effect can be reduced by deactivating the loss after a certain amount of epochs. The parameter n_{epochs} defines for how many epochs the loss function is active. The last two parameters are the number of nearest neighbors between deformed template and target k and the "temperature" σ used to adapt the weight of the closest neighbor (see Equation 19). Those are chosen based on different training results with alterations of the parameter. In order to evaluate the results of the GDPL in a unsupervised manner, metrics based on labels were not taken into account during this evaluation. Table 3.2 summarizes the hyperparameter used for each dataset. As the validation data was utilized for hyperparameter tuning, we report the final results based on the test data split.

Table 3.2: Hyperparameter used for the geodesic distance preservation loss on every dataset.

Parameter	Distal Femur	Liver	Face
$n_{samples}$	11942	12974	1827
α	0.5	1	0.5
n_{epochs}	300	250	250
k	15	10	15
σ	1	2	1

3.3 Data

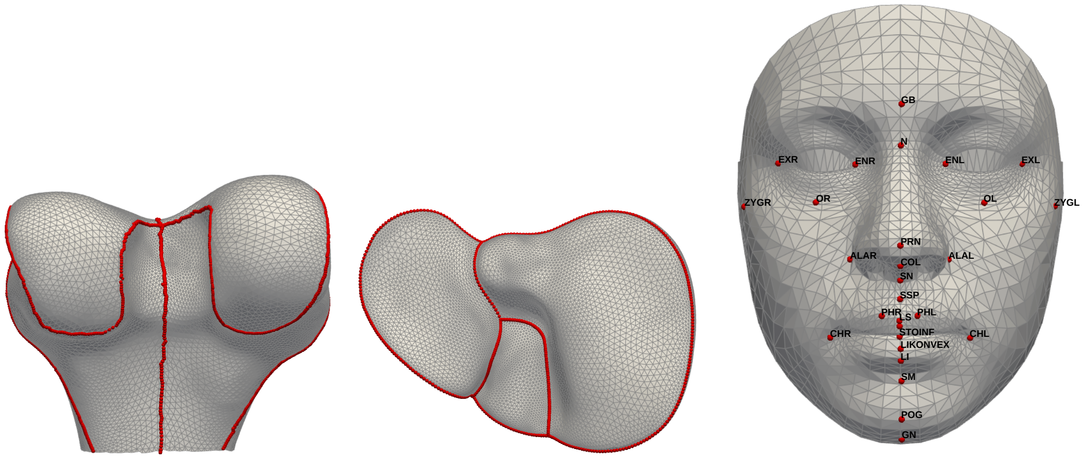
This section presents the datasets used in this thesis. As we want to find the limits of each correspondence estimation method, the objective here is to find challenging datsets that cover a wide range of possible use-cases. Furthermore, Section 2.2.3 showed that some evaluation metrics for dense correspondence require annotations, i.e. labels of some kind. As these labels relate to the data, we need datasets with different types of labels to increase the number of metrics that can be used for evaluation.
Every dataset was split in approximately 70% training data, 10% validation and 20% test data. The datasets used in this thesis and their associated labels are briefly described below. All characteristics of the datasets are summarized in Table 3.3. After the presentation of the datasets we take a look at the origin of the templates and the preprocessing applied to each dataset before usage.

3.3.1 Datasets

Distal Femur The distal femur region refers to the part of the femoral bone, which comprises the knee joint. As a rigid tissue structure, the exhibited variability is limited. The round prominence at the end of the bone, called condyle, shows the most variation [Tyc+18], especially in pathological samples. As the dataset is cut just above the knee joint, the surfaces are bordered.

The data used in this thesis is a subset of the data generated by Ambellan et al. [Amb+19a] on basis of the Osteoarthritis Initiatives data[6]. It consists of segmented MRI scans of healthy samples and those affected from varying degrees of arthritis. Since the data was initially

[6] https://nda.nih.gov/oai/



(a) Distal femur (b) Liver (c) Face

Figure 3.2: Patch borders (a and b) and anatomical landmarks (c) in red defined on the template surface of each dataset.

processed to build SSMs, the triangulation of these meshes is parametrized to represent a semi-automatically generated correspondence.

The process of semi-automated establishment of correspondences between a set of femoral shapes includes the manual annotation of sparse landmarks, the construction of patch borders connecting these landmarks and a supervised post-processing to mesh the surface parts between the patch borders. All of this was done by Ambellan et al. [Amb+19a]. Figure 3.2 (a) shows the template shape of the distal femur data set as well as the patch borders used for the semi-automatic correspondence definition. The semi-automatic correspondence can be understood as some kind of correspondence ground truth. Therefore, it can be used as a label for evaluation throughout this thesis. The vertices placed on the patch borders can be comprehended as a set of sparse landmarks. Since most of those vertices were not placed manually on anatomical landmarks, they lack the significance of landmarks placed on actual anatomical features. Therefore, they are marked as (✓) in Table 3.3.
For the application of methods used for correspondence estimation it is obviously necessary to get meshes without correspondence. Therefore, we use the remeshed version from Lüdke [Lüd22] for training and evaluation.

Liver The liver is a soft tissue organ with a high degree in geometric variability. This can even include twisting and bending [Fas+98]. The liver dataset is the only dataset used in this thesis, where the surfaces have no boundary.

We use the dataset from Kainmuller et al. [KLL07]. Just as the distal femur dataset, this data was initially used to build an SSM. The required correspondences have been created using a method similar to the one that was applied to the distal femur data. Figure 3.2 (b) shows the surface of the template shape and the patch borders used for correspondence estimation. Again, we use the shapes with predefined correspondence as ground truth labels and the remeshed versions from Lüdke [Lüd22] as training and evaluation data. Once more, we can comprehend the vertices placed on the patch borders as a set of sparse landmarks with reduced meaning.

Faces The human face allows for an easy location of many anatomical landmarks. The set of available landmarks is one of the reasons why this dataset is included in the thesis. Furthermore, it is the dataset with the lowest resolution and lowest number of samples used in this thesis. Even though faces are no typical "anatomical shapes", there are possible reconstruction tasks for SSMs (e.g. [Alo+22]) apart from the classical computer vision applications. The dataset represents only the facial area of the head. Thereby, the surfaces have a boundary, just like the distal femur data.

The facial data was originally recorded by Prof. Dr. Dr. Bernd Lapatki and Dr. med. dent. Johanna Radeke at the University of Ulm and consists of 100 different faces with neutral expressions. Grewe and Zachow [GZ16] utilized the dataset to establish a correspondence in an automatic manner beforehand. Since we need medhes without correspondece, we use a remeshed version of the meshes from Grewe and Zachow [GZ16]. Contrary to the other two datasets, we do not use the correspondences generated by [GZ16] as a ground truth label, because they were generated fully automatically.
The sparse landmarks were annotated by experts from the University of Ulm. We use the landmark locations averaged over the nine expert ratings as the ground truth labels. Since some labels are not located on the facial area used in this thesis, we only use 29 of the originally 32 annotated landmarks. Figure 3.2 (c) shows the template shape and the location and abbreviation for every landmark in use. The exact definition of each landmark location can be found in the appendix at Table A1.

3.3.2 Pre-processing and Template Generation

Since all datasets have already been used for SSM construction, the mean shapes of the previously built SSMs are used as templates for all datasets in this thesis. This is arguably a controversial choice, since this mean shape is not available in a real world use-case.

However, this choice will reduce the effects that a poor template could have on the correspondence estimation. Furthermore, the annotated landmarks are already available for the different mean shapes, which simplifies the evaluation process. Last but not least, we can use the semi-automatically generated correspondences to build SSMs and compare their quality to those of automatically generated correspondences. For future applications where a template is not available beforehand, we refer to the related work (e.g. [Dur+14], [Lüd22], [Leb+22], [Whi+19]).

As a result of the recording process, all shapes are already roughly aligned regarding the principal axes. In order to equalize the starting conditions, all shapes were additionally aligned towards their template shape. This was done using the ICP algorithm mentioned in Section 2.1. The prior alignment and the prior cutting of some shapes (e.g. femur to distal femur, region of the face) can be seen as a kind of weak label used by the training methods later on. This would contradict to the definition of *unsupervised* methods. However, almost all methods found in literature that call themselves unsupervised, are trained on data that is at least roughly aligned. Even the datasets used for benchmarking, for instance the shapes of FAUST [Bog+14], are oriented in a similar way. Furthermore, most medical data originated from either CT or MRI scans where orientation of the patient is always known and saved during recording.

Table 3.3: Overview of different features of all datasets. This includes the split between train, validation and test data as well as the available labels for each dataset.

Dataset	Distal Femur	Liver	Face
Avg. number of vertices	11,968	12,974	1,969
Number of training meshes	177	78	67
Number of validation meshes	25	11	10
Number of test meshes	51	23	19
Semi-automatic correspondence	✓	✓	-
anatomical landmarks	(✓)	(✓)	✓

3.4 Evaluation of Dense Correspondence

Since a single, unambiguous ground-truth dense correspondence cannot be defined for anatomical shapes, there is no proper ground truth that can be used for evaluation in this thesis. Therefore, we have to identify indirect ways of estimating the quality of generated correspondence between shapes. Therefore, it is important to include multiple metrics and datasets in the evaluation, since individual metrics can be misleading and draw an incomplete picture. In Section 2.2.3 we already covered a lot of possible metrics. The following section justifies the choice of metrics used in this thesis and covers their implementation.

3.4.1 Selection of Criteria

The selection of criteria highly depends on the available data. Some metrics mentioned in Section 2.2.3, such as the evaluation on a benchmarking dataset with ground truth, could therefore not be included in this thesis. This section summarizes the reasons for the selection of criteria used in this thesis.

Ground truth / Semi-automatic Correspondence Literature often refers to semi-automatically generated correspondences as a "ground truth" used for evaluating correspondences (e.g. [Amb+19a], [AZT21], [Tyc+18], [KLL07]). Since there is data available that has semi- automatically generated correspondence labels and due to the lack of a better alternative, we want to follow this example. Nevertheless, the procedure which was used to generate these correspondences included an automated meshing part which was applied on the areas between the patch borders. Since the correspondences in this area therefore contain a certain degree of ambiguity, their meaningfulness is reduced. Therefore, we refrain from calling it "ground truth" and use the term "semi-automatic" instead.

Landmarks This metric is particularly important, as it enables us to directly evaluate a sparse correspondence. Furthermore, the location of landmarks is an application of shape correspondence in itself and therefore recommended for evaluation by Kaick et al. [Kai+11]. As mentioned in Section 2.3.3, the localization of landmarks can also help us to detect whether the standard metrics used for SSM evaluation are misleading. Therefore, all landmarks available will be used for evaluation. This includes the patch borders used to generate semi-automatic correspondences as well as the actual anatomical landmarks of the face dataset.

Shape Approximation Accuracy As all methods strive to deform a template into different targets, it is important to assess the quality of these deformations. Therefore, we evaluate the Chamfer distance between deformed templates and targets. Furthermore, we take a look at the number of self-intersections on the resulting meshes, as self-intersections are unnatural on anatomical shapes and therefore undesirable.

Quality of Resulting SSMs As mentioned before, the construction of SSMs is an important application for shape correspondence as well as a good indicator for its quality. In accordance to the standard of literature, we evaluate the resulting SSMs on the terms of generality, specificity and compactness. Depending on the application of the SSM later on, it is also important that the meshes generated by the SSM exhibit certain qualities. Therefore, we also take a look at the self-intersecting faces on meshes projected into the SSM and on meshes generated by the SSM.

3.4.2 Metric Details

All automatic methods investigated in this thesis provide correspondence by deforming a template towards a target shape. As a first step, we align the resulting deformed templates to their original targets using the ICP algorithm (see Section 2.1). Furthermore, the test data split is used for the evaluation. This section describes details of all applied criteria and also coins the terms for different evaluation metrics used in this thesis.

Surface Error The "Surface Error" used to measure the quality of the deformation is computed as the symmetric surface to surface Chamfer distance between the deformed template $\mathcal{X}$ and its target shape $\mathcal{Y}$. We compute the distances and average over all M shapes of the test split:

$$\text{Surface Error} := \frac{1}{M}\sum_{i=1}^{M}\mathcal{CD}_{SS}(\mathcal{X},\mathcal{Y}) \tag{22}$$

Self-intersections The intersections within each deformed template mesh are evaluated with two metrics: percentage of Self-Intersecting Meshes (SIM) and number of Self-Intersecting Faces (SIF).

Semi-automatic Correspondence The "Correspondence error" measures the distance between deformed template and their corresponding points on the shapes with semi-automatic correspondence. In order to compute it, the vertices of the deformed template were projected onto the surface of the target mesh in correspondence. Now the Euclidean distance between the projected points $\boldsymbol{X}_{proj}$ and their corresponding points on the target $\boldsymbol{Y}_{corr}$ is measured and averaged over all M shapes with the Frobenius norm $||\cdot||_F$:

$$\text{Correspondence Error} := \frac{1}{M}\sum_{i=1}^{M} ||\boldsymbol{X}_{proj_i} - \boldsymbol{Y}_{corr_i}||_F \tag{23}$$

Landmarks The anatomical landmarks are located on the surface of the template. Most landmarks are not defined on a vertex, but are rather located anywhere on a triangular face of the surface mesh. Therefore, a barycentric coordinate system[7] can be used to transfer the points from the location on the original face to the corresponding location on the deformed face. Afterwards, the landmarks of the deformed template are projected onto the surface of the target. The "Landmark Error" describes the averaged Euclidean distance between the projected landmark locations on the deformed template $\boldsymbol{L}_{proj}$ and the ground truth locations $\boldsymbol{L}_{gt}$:

$$\text{Landmark Error} := \frac{1}{M}\sum_{i=1}^{M} ||\boldsymbol{L}_{proj_i} - \boldsymbol{L}_{gt_i}||_F \tag{24}$$

Since the patch borders of the liver and distal femur dataset are defined on the vertex positions, they can be seen as sparse correspondence. Therefore, they are evaluated in the same way as the semi-automatic correspondence.

SSM Quality In order to only evaluate the underlying correspondence and not the method to build an SSM, all SSMs are build in the same way. Thereby, we use the standard procedure proposed by Cootes et al. [Coo+95] and summarized in Section 2.3.1. At first, all corresponding meshes will be aligned towards the template using the Procrustes Analysis (see Section 2.1). Here we use the deformed training meshes of each method as this increases the number of eigenmodes. Afterwards, a PCA is applied to these meshes, and the resulting SSM consists of its eigenmodes. Figure 3.3 visualizes the different algorithms and data used in the evaluation process for an SSM.

[7] When applied to triangles, barycentric coordinates make use of the fact that any point on the triangle can be expressed as a linear combination of the vertices which span the triangle. The weights of this linear combination are the barycentric coordinates, which can be applied to any triangle [YS19].

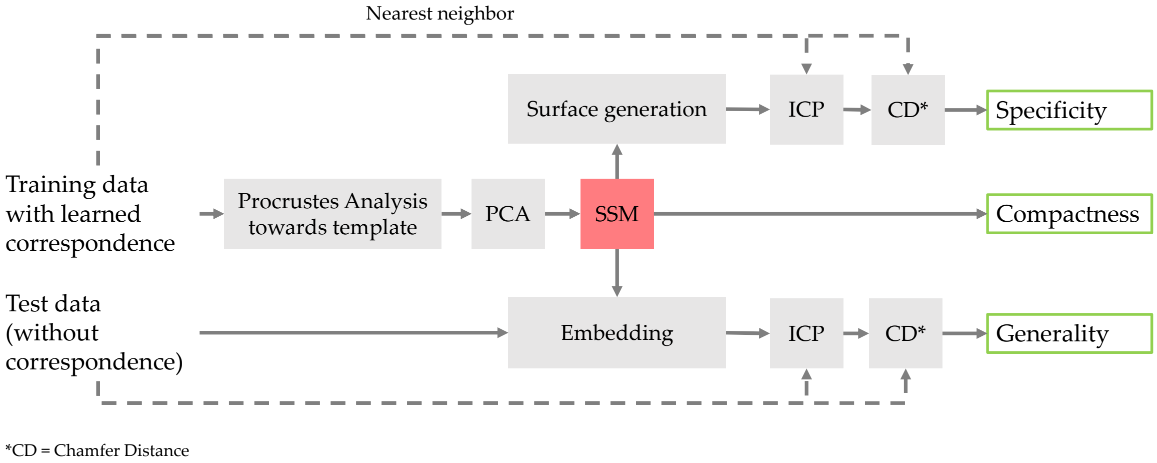


Figure 3.3: Flowchart for SSM evaluation. The training data with learned correspondences is used to build the SSM and for the evaluation of the specificity. The correspondence-free test data is used for the generality evaluation, whereas the compactness stems from the SSM itself.

In literature, the generalization ability is often computed as a leave-one-out test of all shapes. Since this would lead to a lot of computationally demanding re-trainings of all correspondence estimation methods examined in this thesis, we refrain from this approach. Instead we compute this metric as a hold-out study, which is commonly used in deep learning. To evaluate the generality, we use the test dataset without correspondence. In contrast to the shapes with trained correspondence, this reduces the bias towards smoother deformations and creates the same conditions for all methods under examination. The shapes get embedded into the SSM and afterwards aligned to their original shape with the ICP algorithm. The latter is done to reduce the influence of rigid motion. Afterwards, the symmetric surface to surface Chamfer distance between the embedded $\mathcal{X}_{proj}$ and original shape $\mathcal{X}_{org}$ is computed and averaged over all M shapes of the test split:

$$\text{Generality} := \frac{1}{M} \sum_{i=1}^{M} \mathcal{CD}_{SS}(\mathcal{X}_{org_i}, \mathcal{X}_{proj_i}) \tag{25}$$

The poses of the face-shapes have a high variability. This can lead to problems during the generality experiment. In order to reduce this risk, we add an ICP alignment step after every 5 projection steps. This is only done for the face dataset.

In order to evaluate the specificity, 2000 random meshes are sampled based on the SSM. Again, the sampled meshes get aligned towards every training shape, using the ICP algorithm. Afterwards, we compute the pointset to pointset Chamfer distance between the generated mesh $\mathcal{X}_{gen}$ and its closest match in the training set $\mathcal{Y}_{nn}$. We refrain from using the surface to surface Chamfer distance and use the pointset to pointset Chamfer distance instead, since it is faster to compute. Finally, we average the Chamfer distances over all 2000 shapes:

$$\text{Specificity} := \frac{1}{2000} \sum_{i=1}^{2000} \mathcal{CD}_{PP}(\mathcal{X}_{gen_i}, \mathcal{Y}_{nn_i}) \tag{26}$$

The compactness is evaluated in the following way: We sum the eigenvalues and plot the variance over the number of eigenmodes needed. It is therefore computed as

$$\text{Compactness}(K) = \sum_{i=1}^{K} \lambda_i. \tag{27}$$

Here K represents the number of eigenvectors used to build the model, and λ_i indicates the eigenvalue of the i-th eigenvector [Gop+22].

The intersections within each generated mesh are evaluated with two metrics already mentioned above: percentage of SIM and SIF. We compute these metrics for all embedded shapes of the generality experiment and all generated shapes of the specificity experiment.

3.5 Summary

The main goal of this thesis is to examine different methods used for shape correspondence estimation of anatomical shapes. This is done with two experiments: the comparison of different methods found in literature and the further development of one method. All methods strive to deform a template towards the targets surface. The underlying assumption is that points are deformed to semantically similar locations, which ensures a proper correspondence.

Four methods were chosen for the comparison: Meshmonk and LDDMM, which follow classical optimization problems, as well as the two group-wise methods FlowSSM and NeuroMorph. The method FlowSSM is also used in the second experiment, as we implemented an additional loss term, the GDPL. This loss term prefers isometric, distortion-free deformations and can therefore help to improve the correspondence quality. As the loss-term increases the memory usage, other minor changes from the original FlowSSM had to be implemented, resulting in new baseline models.

All methods are evaluated with the same experimental set-up, consisting of datasets and evaluation metrics. Three different datasets are used, namely distal femur, liver and face. The datasets differ in resolution and geometrical variance in order to expand the challenge. As there is no ground truth available, different indirect metrics were chosen to evaluate the correspondence. This includes metrics based on labels such as the correspondence and landmark error, metrics based on the quality of the resulting meshes (surface error and self-intersections) as well as metrics to asses the quality of the resulting SSMs, namely generality, specificity and compactness.

4 Results and Discussion

In this chapter we present the results of all experiments and discuss their meanings. The goal of this chapter is to answer the following questions:

- Which methods produce good shape correspondences and why?
- Which metrics are suitable to evaluate correspondence?
- How do the different characteristics of the three datasets affect the results of each method?

The first section of this chapter deals with the first experiment, which is the comparison of different methods from literature. Starting with the plain results of each dataset, we put them into context and analyze the (dis-)advantages of each method afterwards. The second section covers the second experiment and therefore evaluates the GDPL. At first, we discuss the deviations towards the results of FlowSSM. Afterwards, we try to explain them.

4.1 Experiment 1: Comparison of different Methods from Literature

In the first experiment, we compare the correspondence quality of the different correspondence estimation methods presented in Section 2.2.1. We report the results of all criteria sorted by the different datasets. Since the datasets distal femur and liver include a semi-automatically generated correspondence for all shapes, these meshes were used to construct an SSM. This gives us the opportunity to evaluate whether time-consuming process of the semi-automatic correspondence generation is actually worthwhile.

There are four different variants of the method NeuroMorph evaluated in this section. The first variation results from the use of the postprocessing method SmoothShells [ELC20] as suggested in the original publication. Results after postprocessing are noted with the ending "pp.". The other variation stems from the shapes deformed during the training od the model. As already mentioned in Section 3.1.2, the initial method trains by deforming every possible data combination. The variation implemented in this thesis, where only the template shape gets deformed toward every target shape, is marked with the ending "o.T." (on template).

Computation Time Apart from the quality of the resulting correspondences, the time needed for the estimation is an important factor when it comes to the usability of the different methods. Table 4.1 gives an overview on the different methods, datasets and used hardware. It is important to note, that the data-driven methods often take a lot of time for training, but the actual time needed for the correspondence estimation of unseen shapes is rather short. As the prediction time from NeroMorph is negligible when compared to the duration of training, it is not listed in the table. In contrast, the classical methods Meshmonk, LDDMM and the method used for post-processing of the NeuroMorph data need the same amount of time for every shape, as they optimize the pairwise deformation individually. The prediction durations listed in Table 4.1 are based for the test split of every dataset.

Table 4.1: Computation times needed for training and predictions of different methods.

Method	train. / pred.	Distal Femur	Liver	Face
FlowSSM	training	4 h★	2 h▲	45 min★
	prediction	2 h♦	0.5 h★	1.5 h★
NeuroMorph	training	7 d 22 h▲	4d 22.5 h★	3 h★
NeuroMorph o.T.	training	2 d 3.5 h★	5.5 h▲	2.5 h♦
SmoothShells	prediction	2 h*	1 h*	15 min*
LDDMM	prediction	10 min*	5 min*	5 min*
Meshmonk	prediction	4.5 h*	2 h*	15 min*

★ = Nvidia A40 RTX 48GB ▲ = Nvidia Tesla V100 PCIe 32GB
♦ = Nvidia A100 SXM4 80GB * = Nvidia GeForce RTX 3080 10GB

4.1.1 Results on Distal Femur Data

The results of all methods applied to the distal femur dataset are listed in Table 4.2 and Table 4.3. The first table shows the evaluation metrics directly based on the deformed meshes, while the second table shows the results of the constructed SSMs. A short view on the tables reveals that there is no clear ranking. Rather, different methods excel on a few metrics, but underperform on other criteria. Furthermore, the results are quite similar between all methods and hardly show any variation.

The surface error is lowest on the meshes deformed by the LDDMM approach. The versions of NeurmoMorph without postprocessing perform worst. Pairwise methods (i.e. Meshmonk and LDDMM) deform the template without the generation of intersections. The classical NeuroMorph approach leads to the most self-intersections, with and without postprocessing. The correspondence error is exceptionally low on the templates deformed by Meshmonk. This also applies to the landmark error.

All SSMs build on automatically generated correspondences have a better generalization ability, than the SSM build on semi-automatic correspondence. The generality error is lowest if the SSM was build on the LDDMM data or the data generated by the postprocessed NeuroMorph version trained to only deform the template. However, the SSMs build on all NeuroMorph results produce a lot of self-intersections in both generality and specificity results. The specificity error itself is lowest on the un-postprocessed NeuroMorph versions. Figure 4.4 (a) shows the compactness of the SSMs build on the estimated correspondences. It is obvious that the LDDMM framework leads to the lowest compactness. The NeuroMorph versions without postprocessing reach the best results.

Table 4.2: Evaluation metrics on the deformed templates of the distal femur data for all methods.

Method	**Surface Error in mm**	**SIM**	**SIF**	**Corresp. Error in mm**	**Landmark Error in mm**
FlowSSM	0.12 ± 0.09	8 %	31	1.53 ± 0.21	1.54 ± 0.24
NeuroMorph	0.51 ± 0.06	24 %	8	1.77 ± 0.31	1.83 ± 0.32
NeuroMorph pp.	0.10 ± 0.02	69 %	10	1.60 ± 0.29	1.70 ± 0.31
NeuroMorph o.T.	0.52 ± 0.05	4 %	10	1.74 ± 0.27	1.81 ± 0.29
NeuroMorph o.T. pp.	0.10 ± 0.02	4 %	10	1.77 ± 0.24	1.86 ± 0.26
LDDMM	0.07 ± 0.01	0 %	0	1.51 ± 0.34	1.51 ± 0.32
Meshmonk	0.11 ± 0.03	0 %	0	1.05 ± 0.27	1.10 ± 0.26

Table 4.3: SSM quality on distal femur data in terms of generality and specificity as different methods were used for correspondence estimation.

Method	**Generality**			**Specificity**		
	Error in mm	**SIM**	**SIF**	**Error in mm**	**SIM**	**SIF**
Semi-automatic	0.30 ± 0.07	8 %	24	1.12 ± 0.13	2 %	11
FlowSSM	0.27 ± 0.05	0 %	0	1.12 ± 0.12	0 %	0
NeuroMorph	0.25 ± 0.05	86 %	39	0.95 ± 0.14	56 %	8
NeuroMorph pp.	0.26 ± 0.05	84 %	14	1.13 ± 0.14	79 %	12
NeuroMorph o.T.	0.25 ± 0.05	80 %	50	0.97 ± 0.12	21 %	25
NeuroMorph o.T. pp.	0.24 ± 0.04	57 %	26	1.12 ± 0.13	24 %	19
LDDMM	0.24 ± 0.04	2 %	74	1.14 ± 0.15	2 %	84
Meshmonk	0.26 ± 0.06	2 %	122	1.08 ± 0.12	1 %	19

4.1.2 Results on Liver Data

The results for the liver dataset are listed in Table 4.4 and Table 4.5. Again, there is no clear ranking between the different methods. But in contrast to the distal femur data, the results of the different methods differ severely.

The surface error of the NeuroMorph variants is exceptionally high, while the postprocessing leads to a significant decrease of the error. Only LDDMM reaches a similar low surface error. Again, the meshes processed by all NeuroMorph variants have a lot of self-intersecting faces. While Meshmonk also generates a lot of self-intersections, FlowSSM hardly generates any and LDDMM does not create any self-intersections at all. The correspondence error and landmark error of Meshmonk are again particularly low. FlowSSM generates an even lower landmark error. The NeuroMorph variants without postprocessing generate remarkably high landmark errors.

When the estimated correspondences are used for SSM construction, the resulting SSMs have the following qualities: Again, the semi-automatically generated correspondences lead to the highest generality error. The NeuroMorph variants without postprocessing lead to the lowest generality error, as well as the Meshmonk and LDDMM approach. In terms of specificity, the NeuroMorph varaiants without postprocessing reach by far the best results, while LDDMM performs worst. All Neuromorph variants, the semi-automatic correspondences and Meshmonk lead to a lot of self-intersections in the resulting meshes of both experiments, generality and specificity. The compactness of all SSMs is visualized in Figure 4.4 (b). Noticeably, all methods yield quite similar results. On closer inspection, we can see that the NeuroMorph versions with postprocessing need the most modes to capture 100 % variance. The SSM constructed with the correspondences generated by LDDMM is the least compact SSM when only a few eigenmodes are used. Surprisingly, it is also the first SSM to reach the full variance.

Figure 4.6 shows a concrete example of the generality experiment. The semi-automatic correspondences (c) lead to high frequency details, which are not part of the original shape (a). The NeuroMorph variants, especially without postprocessing, as well as Meshmonk, lead to many unnatural hard edges and an irregular mesh. The projections made by FlowSSM and LDDMM produce the smoothest mesh, but lack some geometrical details such as the bulge in the top left corner.

Table 4.4: Evaluation metrics on the deformed templates of the liver data for all methods.

Method	Surface Error in mm	SIM	SIF	Corresp. Error in mm	Landmark Error in mm
FlowSSM	0.57 ± 0.11	9 %	20	11.04 ± 4.10	8.97 ± 4.23
NeuroMorph	1.93 ± 0.28	100 %	158	11.74 ± 4.00	12.30 ± 4.82
NeuroMorph pp.	0.34 ± 0.05	100 %	160	11.27 ± 4.08	11.08 ± 5.11
NeuroMorph o.T.	1.71 ± 0.22	96 %	129	11.38 ± 4.07	12.63 ± 4.80
NeuroMorph o.T. pp.	0.33 ± 0.05	96 %	116	11.33 ± 4.05	11.31 ± 4.94
LDDMM	0.36 ± 0.04	0 %	0	11.05 ± 3.73	10.26 ± 4.53
Meshmonk	0.62 ± 0.21	91 %	226	10.23 ± 4.28	9.86 ± 4.99

Table 4.5: SSM quality on liver data in terms of generality and specificity as different methods were used for correspondence estimation.

Method	Generality			Specificity		
	Error in mm	SIM	SIF	Error in mm	SIM	SIF
Semi-automatic	1.92 ± 0.39	100 %	113	5.05 ± 0.64	88 %	60
FlowSSM	1.82 ± 0.38	35 %	70	5.04 ± 0.67	5 %	64
NeuroMorph	1.74 ± 0.36	100 %	492	4.23 ± 0.57	100 %	184
NeuroMorph pp.	1.81 ± 0.37	100 %	297	4.99 ± 0.62	100 %	204
NeuroMorph o.T.	1.67 ± 0.37	100 %	387	4.20 ± 0.61	99 %	133
NeuroMorph o.T. pp.	1.78 ± 0.40	96 %	178	4.89 ± 0.66	100 %	134
LDDMM	1.75 ± 0.32	57 %	106	5.10 ± 061	56 %	103
Meshmonk	1.73 ± 0.23	100 %	342	4.84 ± 0.65	96 %	106

4.1.3 Results on Face Data

The results for the deformed face meshes are listed in Table 4.6. Similar to the other datasets, there is no clear ranking between the methods. Once more, we observe high differences between the different methods.

The surface error, again, is highest on the NeuroMorph versions without postprocessing. It is exceptionally low on the meshes deformed by LDDMM. While FlowSSM leads to no self-intersections whatsoever, all NeuroMorph versions create not even a single mesh without self-intersections. The landmark error is lowest on the meshes deformed by FlowSSM. The standard NeuroMorph implementation leads to the highest landmark error. It also is to be highlighted that the postprocessing step increases the landmark error when applied to NeuroMorph version trained on deforming only the template mesh.
As the dataset was initially used by Grewe and Zachow [GZ16] to generate unsupervised

correspondences, we can compute their landmark error on our test split. In order to ensure a better comparability, we used the results based on a version of their proposed method which does not use texture information. With on average 2.38 mm, the error is slightly higher than for the meshes generated with FlowSSM. However, the comparability is limited, as they trained their model on the original dataset with an increased depth of field and the used template has an open mouth, as well as eyes. As the method uses landmark detection for an initial alignment, it can only be used on face data.

The landmark error is visualized for all individual landmarks in Figure 4.1. While the left side shows the landmark error of the landmarks located on the vertical centerline of the template, the right side shows all other landmarks. It is obvious that the ranking between the methods differs from landmark to landmark. Especially on the centerline, the error increases if the landmark is located closer the edge of the surface. Only FlowSSM seems to disregard this trend. The same trend also occurs on the right plot in weakened form. Here, the landmarks are plotted based on their horizontal location from right to left. The errors of Meshmonk even seem to be nearly symmetrical in regard to the vertical centerline, which is located between the landmarks PHR and PHL. While LDDMM reaches the highest errors, FlowSSM yields by far the best results. Sometimes, the error is even lower than the Intra Person Variability (IPV). In those cases, the method performs on the same level as the expert annotators.

The generality and specificity results for the face dataset are listed in Table 4.7. Meshmonk, LDDMM and the NeuroMorph versions with postprocessing lead to the lowest generality errors. The specificity error again is lowest on the NeuroMorph versions without postprocessing, but FlowSSM also reaches a comparatively low result. Once more, all NeuroMorph versions lead to a lot of self-intersections. While Meshmonk creates a lot of self-intersections in the generality experiment, the specificity experiment creates only a few self-intersections. The compactness results for all methods are visualized in Figure 4.4 (c). The methods LDDMM and Meshmonk need the most eigenmodes to capture the variance of the population. FlowSSM and the NeuroMorph version trained to deform only the template need the fewest.

Figure 4.5 shows a concrete example of the generality experiment. The original structure of the template mesh is best preserved with FlowSSM (c). The other methods obviously have problems with the illustration of the eyes, as the shape is often very unnatural. NeuroMorph without postprocessing leads to a large nose and Meshmonk seems to have problems with the border of the face.

Table 4.6: Evaluation metrics on the deformed templates of the face data for all methods.

Method	Surface Error in mm	SIM	SIF	Landmark Error in mm
FlowSSM	0.46 ± 0.12	0 %	0	2.25 ± 1.46
NeuroMorph	0.83 ± 0.08	100 %	175	4.30 ± 3.17
NeuroMorph pp.	0.30 ± 0.03	100 %	167	4.17 ± 2.85
NeuroMorph o.T.	0.97 ± 0.07	100 %	273	3.40 ± 2.21
NeuroMorph o.T. pp.	0.30 ± 0.04	100 %	220	3.87 ± 2.48
LDDMM	0.17 ± 0.02	84 %	26	3.79 ± 3.49
Meshmonk	0.78 ± 0.38	42 %	11	3.13 ± 2.79

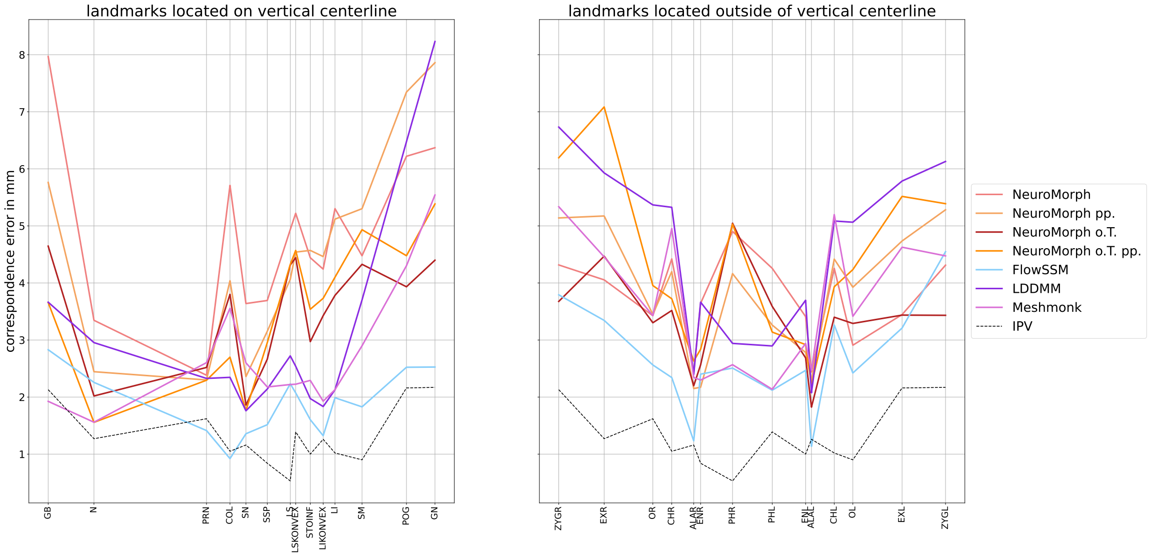


Figure 4.1: Landmark errors of the different methods. The left figure shows the landmarks located on the vertical centerline of the template from top to bottom. The right figure shows the landmarks located outside of the centerline of the template from right to left. The distance between the landmarks is proportional to their distance on the template in the respective direction. The IPV data was taken from [Tiw21].

Table 4.7: SSM quality on face data in terms of generality and specificity as different methods were used for correspondence estimation.

Method	Generality			Specificity		
	Error in mm	SIM	SIF	Error in mm	SIM	SIF
FlowSSM	1.25 ± 0.20	11 %	40	1.82 ± 0.20	0 %	0
NeroMorph	1.10 ± 0.26	100 %	286	1.64 ± 0.19	100 %	185
NeuroMorph pp.	1.02 ± 0.25	100 %	210	2.14 ± 0.18	100 %	181
NeuroMorph o.T.	1.14 ± 0.31	100 %	312	1.68 ± 0.19	100 %	275
NeuroMorph o.T. pp.	1.04 ± 0.28	100 %	211	2.31 ± 0.26	100 %	219
LDDMM	0.94 ± 0.28	79 %	62	2.11 ± 0.21	63 %	34
Meshmonk	1.00 ± 0.21	100 %	49	2.03 ± 0.18	36 %	8

4.1.4 Discussion

The results on all dataset showed that there is no clear ranking between the methods, as the results differ on each metric and each dataset. In literature, it is often assumed that group-wise methods are favorable compared to pair-wise methods, as they are less prone to outliers. This assumption can not be confirmed by our results, as there is no clear ranking between those categories. Furthermore, the selected metrics used for evaluation often contradict each other. This highlights that the quality of a correspondence can best be evaluated by using more metrics, and that the metrics should be chosen according to the use case at hand. Nonetheless, there are a few general observations to be made on the performance of every method:

Semi-automatic Correspondences The semi-automatically generated correspondences of the distal femur and liver data were used to build SSMs. It is remarkable that the resulting generality errors are the highest when compared to all other methods. The specificity errors also range in the higher levels. The high frequency details in Figure 4.6 can only partly be explained by the fact that the meshes have no surface error and therefore contain all details of the original shapes. This of course leads to eigenmodes that capture more details, resulting in more high frequency details on the projections. But since the projection also shows details that are not part of the original shapes, this indicates that the eigenmodes are somehow faulty. The observations described above induce the realization that the time-consuming task of semi-automatically generating correspondences is no longer necessary. On the other hand, the observations question the reliability of the correspondence labels for distal femur and liver data, as these labels were generated in the same way. For this reason, these labels will be given a lower priority in the remainder of this thesis.

FlowSSM For most metrics the method scores somewhere in the middle field. However, there are a few exceptions to be made. First of all, the meshes generated by FlowSSM and the resulting SSMs have the fewest intersections. This can be explained by the underlying approach, as the integration of a deformation flow is known to hardly create self-intersections ([Lüd+22], [Jia+21]). Furthermore, FlowSSM yields by far the best results on the landmark errors on the face and liver dataset. This could be explained by its independence from the template meshing: As FlowSSM samples points evenly on the surface of each mesh, it is not influenced by irregularities of the mesh itself. Since for example the face template is densely meshed around the eyes, other methods "use" these vertices on other parts of the face, resulting in a bad representation of the eyes and a high landmark error. This could also explain why the landmark error tends to be higher on the outer landmarks for nearly all methods (compare Figure 4.1).

NeuroMorph Two metrics stand out when evaluated on the NeuroMorph results: the exceptionally high numbers of self-intersections and the low specificity errors when computed without post-processing. Since self-intersections are scarcely evaluated in the broader computer vision domain, this disadvantage is probably not known to the authors. The good specificity results however can easily be explained: The surface error of the variants without postprocessing is also quite high. This leads to smoother meshes without many details. The specificity error evaluates the similarity between shapes generated by the SSM and shapes used to build the SSM. As the latter lack many details, the generated meshes will also be very smooth. And since no details have to be illustrated in this setting, the resulting error is exceptionally low. The good specificity results are therefore misleading and the postprocessing step highly recommendable whenever the surface error is high. Another observation of the NeuroMorph variants is notable: The results when trained in the standard setting hardly differ from the results when the deformations are only trained on the template. As the latter is much faster during training (see Table 4.1), it is recommendable to use this version.

LDDMM Similar to FlowSSM, meshes generated by LDDMM tend to have only a few self-intersections. Again, this can be explained by the integration of a deformation flow field which is part of the method. LDDMM produces the lowest surface errors and good generality results. However, the specificity error is often among the highest and the compactness is also on the lower end. The landmark error on the face data is also one of the highest. These contradicting results lead to the assumption that the generated correspondences are somehow faulty. Erroneous correspondences that yield good SSM results have also been reported in literature (i.e. [Gop+22], [EK07], [MDS08]). The low performance for the landmark errors of the face dataset can probably again be explained by the construction of the template mesh. As LDDMM takes vertices from the densly meshed area around the eyes and moves them somewhere else, the surface error is reduced, as more vertices are available to represent the other areas of the face. Unfortunately, this means that the verices are not moved to semantically similar locations, which leads to a high landmark error. But as all shapes are deformed in the same (presumably wrong) way, the results are consistent in itself. This, on the other hand, could explain the good generality results.

Meshmonk Regarding all datasets, Meshmonk tends to have an especially low landmark error, but the SSM quality is always somewhere in the middle range. In the concrete examples of the generality experiment on the liver (Figure 4.6) and face dataset (Figure 4.5), we can see that resulting shapes have an irregular mesh with unnaturally sharp edges. As the authors hardly give any background information on the mathematical background of the method, it is difficult to interpret the results. Nevertheless, as the method is easy to use and quite quick in its computations, it might be a good choice for some applications, especially if these include the annotation of landmarks.

4.2 Experiment 2: Evaluation of FlowSSM with the Geodesic Distance Preservation Loss

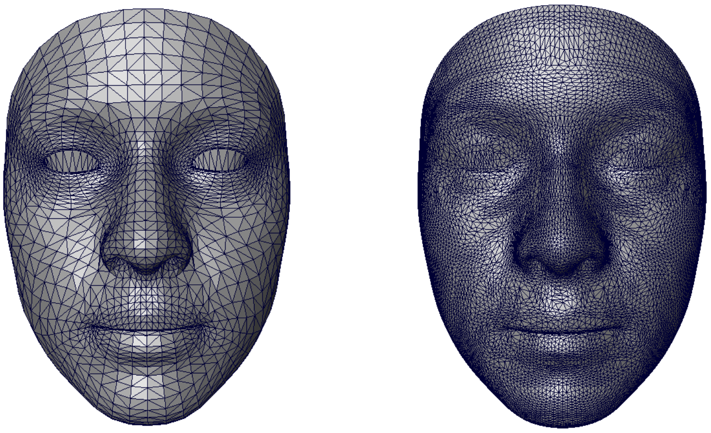
In the second experiment we evaluate the performance of the GDPL implemented in FlowSSM. As the implementation required us to decrease the number of points considered on the template surface, a new baseline model was trained for each dataset. The differences observed on the error metrics between the original FlowSSM model and the new baseline model also influence the model trained with the new loss term and therefore the final results. It is assumed that the results of the GDPL could be altered by those differences if a large enough memory was available. The main hypothesis to be tested in this experiment is whether the addition of the GDPL leads to a better correspondence and if so, why.

Remeshed Face Data The GDPL relies on a search of up to 15 nearest neighbors during the estimation of the correspondence matrix. This can be challenging if there are not many neighbors available, which is the case for the face dataset due to its low resolution. On the original template, as shown in Figure 4.2 (a), a set of 15 nearest neighbors would probably span over half the area of the forehead. And a "wrong" neighbor, that could for instance occur if the hneighborhood is influenced by the edge of the surface, would have a huge impact. In order to weaken this challenge, the face dataset was remeshed to increase its resolution. Thereby, the previous average resolution of 1,969 vertices per shape was increased to 10,101 vertices per shape. Figure 4.2 (b) shows the remeshed template. It is obvious that the resolution of the area around the eyes is approximately the same as on all other areas, as opposed to the original template.

Computation Time Table 4.8 lists the computation times needed for the trainings and predictions of the different FlowSSM versions. It is obvious that the use of the GDPL leads to a steep increase in training but especially in prediction time. While the memory management of the method could be improved, the difference partly stems from the many nearest neighbor searches. Therefore, the use of the GDPL will always be slower. The fact that an additional pre-processing step is needed for the computation of the geodesic distance matrices increases the computation time of the GDPL even more.

4.2.1 Results on all Datasets

Table 4.9 lists the results on the deformed meshes of all datasets. We can see that the addition of the GDPL term leads to small increases of all metrics, especially on the landmark



(a) Original template (b) Remeshed template

Figure 4.2: Original and remeshed face template

Table 4.8: Computation times needed for training and predictions of different FlowSSM versions.

Method	train. / pred.	Distal Femur	Liver	Face	Face remeshed
Original	training	4 h$^\bigstar$	2 h$^\blacktriangle$	45 min$^\bigstar$	45 min$^\bigstar$
	prediction	2 h$^\blacklozenge$	0.5 h$^\bigstar$	1.5 h$^\bigstar$	1.5 h$^\bigstar$
New Baseline	training	2h$^\bigstar$	1 h$^\bigstar$	45 min$^\bigstar$	45 min$^\bigstar$
	prediction	1 h$^\bigstar$	15 min$^\bigstar$	1.5 h$^\bigstar$	30 min$^\bigstar$
with GDPL	training	1 d 8 h$^\bigstar$	16.5 h$^\bigstar$	1.5 h$^\blacklozenge$	8.5 h$^\bigstar$
	prediction	1d 1.5 h$^\bigstar$	1d 4h$^\bigstar$	30 min$^\bigstar$	2d 10h$^\blacklozenge$

★ = Nvidia A40 RTX 48GB ▲ = Nvidia Tesla V100 PCIe 32GB

♦ = Nvidia A100 SXM4 80GB

error. It is also noticeable that the remeshing of the face dataset leads to an overall lower surface error. In order to see whether the deviation between new baseline and GDPL are statistically significant, a pairwise t-test was applied to all error metrics. Except for the liver data, the addition of the GDPL leads to statistically significant higher errors ($p < 0.05$).

In Figure 4.3 we see the error on each individual landmark of the face dataset. As before, the figure is split in two different plots. The left plot lists the errors located on the vertical centerline of the template, while the right plot covers all other landmarks from right to left. Most errors are significantly higher when the GDPL is added. The errors can be reduced, if the shapes have a higher resolution (remeshed versions). Again, we can see that the error increases if the landmark is closer to the edge of the surface. This effect is especially striking for the GDPL variants on the right plot. The differences between the methods tend to be smaller if the landmarks are located on the centerline of the template. Here, the remeshed GDPL version produces nearly the same results as the baseline trainings and sometimes it is even better (e.g. on LIKONVEX, COL, STOINF).

In Table 4.10 the generality and specificity results are listed. The GDPL leads to a small increase of self-intersections on the generality meshes of the distal femur data. Otherwise the results seem quite consistent. On the liver dataset we see a decrease of the generality and specificity error, as well as a decrease of self-intersections on the generality meshes. The latter changes on the face data, where the GDPL leads to a steep increase of self-intersections on the generality meshes. It also leads to a small decrease of the specificity error. Again, paired t-tests were conducted to evaluate the statistical significance of the deviations between the new baseline model and the FlowSSM version with GDPL. We can see that the addition of the GDPL has no significant influence on the femur SSM. However, the improvements we can see on the liver and remeshed face dataset are all statistically significant. The compactness for all datasets is plotted in Figure 4.4. While the GDPL leads to a reduced compactness on the distal femur data, this is not observable on the other datasets. The compactness on the remeshed face dataset is even improved by the addition of the GDPL.

Figure 4.5 shows a concrete example of the generality experiment of the face data. Akin to the similar generality error, it is hard to state which result is better. It is, however remarkable that the version with the GDPL moved the characteristic rhombus on the templates mesh structure further up the forehead. On the exemplary generality result of the liver data (Figure 4.6), the resulting meshes differ in many regions. Unfortunately, these differences are hard to interpret.

4.2.2 Discussion

When we compare the results of the FlowSSM version with GDPL and the results of the new baseline model we can summarize the following observations:

1. On the distal femur dataset and the face variants we get significantly worse results on the metrics applied to the deformed meshes when adding the GDPL.
2. The results on the deformed liver meshes hardly deviate from the new baseline.
3. We get significantly lower generality and specificity errors on the liver and face remeshed dataset if we add the GDPL. Furthermore, these metrics hardly deviate on the other datasets.

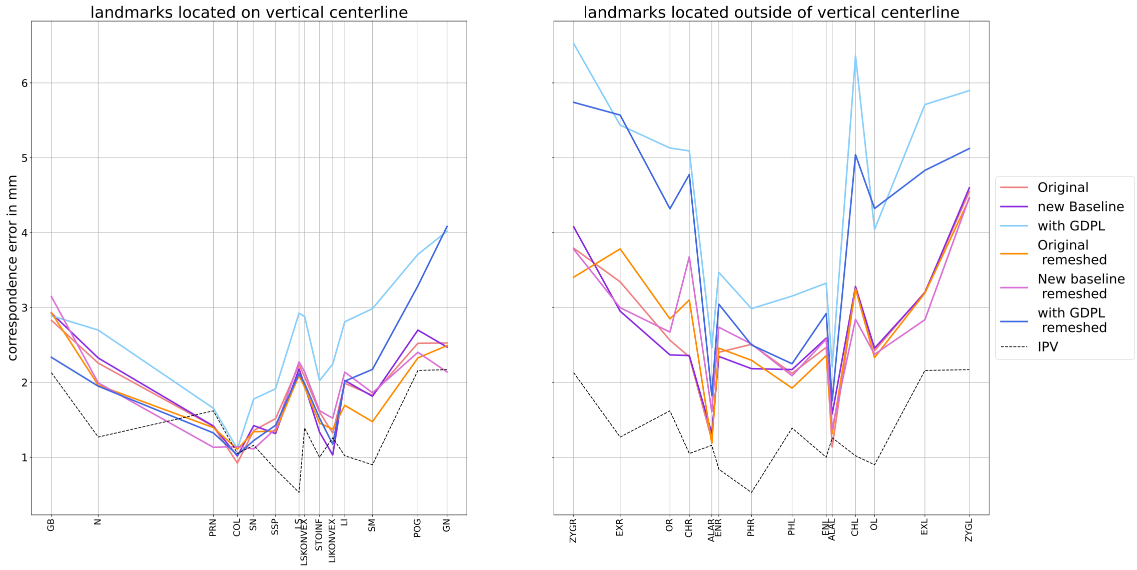


Figure 4.3: Landmark errors of the different FlowSSM variants. The left figure shows the landmarks located on the vertical centerline of the template from top to bottom. The right figure shows the landmarks located outside of the centerline of the template from right to left. The distance between the landmarks is proportional to their distance on the template in the respective direction. The IPV data was taken from [Tiw21].

Table 4.9: Evaluation metrics on the deformed templates of all datasets for the different variants of FlowSSM. If the addition of the GDPL leads to statistically significant deviations (paired t-test, $p < 0.05$) from the new baseline, the results are marked with a "*". Only error metrics were tested.

Dataset	Method	Surface Error in mm	SIM	SIF	Corresp. Error in mm	Landmark Error in mm
Distal Femur	Original	0.12 ± 0.09	8 %	31	1.53 ± 0.21	1.54 ± 0.24
	New baseline	0.14 ± 0.09	6 %	26	1.89 ± 0.23	1.86 ± 0.27
	with GDPL	0.15 ± 0.09*	8 %	29	2.10 ± 0.64*	2.11 ± 0.65*
Liver	Original	0.57 ± 0.11	9 %	20	11.04 ± 4.10	8.97 ± 4.23
	New baseline	0.68 ± 0.14	9 %	31	10.45 ± 3.98	8.87 ± 4.49
	with GDPL	0.68 ± 0.12	4 %	25	10.60 ± 5.29	9.92 ± 6.25
Face	Original	0.46 ± 0.12	0 %	0	-	2.25 ± 1.46
	New baseline	0.48 ± 0.15	0 %	0	-	2.24 ± 1.45
	with GDPL	0.81 ± 0.11*	0 %	0	-	3.43 ± 2.21*
Face remesh	Original	0.34 ± 0.16	0 %	0	-	2.22 ± 1.48
	New baseline	0.34 ± 0.17	0 %	0	-	2.28 ± 1.48
	with GDPL	0.40 ± 0.16*	0 %	0	-	2.87 ± 2.10*

The liver obviously displays the largest benefits of the GDPL. This could be explained in two different ways. The liver is, contrary to the other datasets, a soft-tissue organ and could thereby exhibit more deformations with isometric nature. This would fit the formulation of the geodesic distance preservance and could therefore explain the improved results.
Another possible explanation is based on the nearest neighbor search. As we search for up to 15 nearest neighbors for each template vertex, we assume them to be equally distributed in the area around the original vertex. However, if the vertex is located at or near the edge of the surface, this assumption cannot be made. If, for example, the vertex is located on the right edge of the surface, most neighbors will be located further left, which distorts the resulting adapted geodesic distances. Thereby, the loss function does not function properly. As the surface of the liver has no border, this problem doesn't occur at all. This observation could also partly explain, why the landmark errors are higher on the outside of the face: it is more likely that the neighborhood of these landmarks touches the edge of the surface.

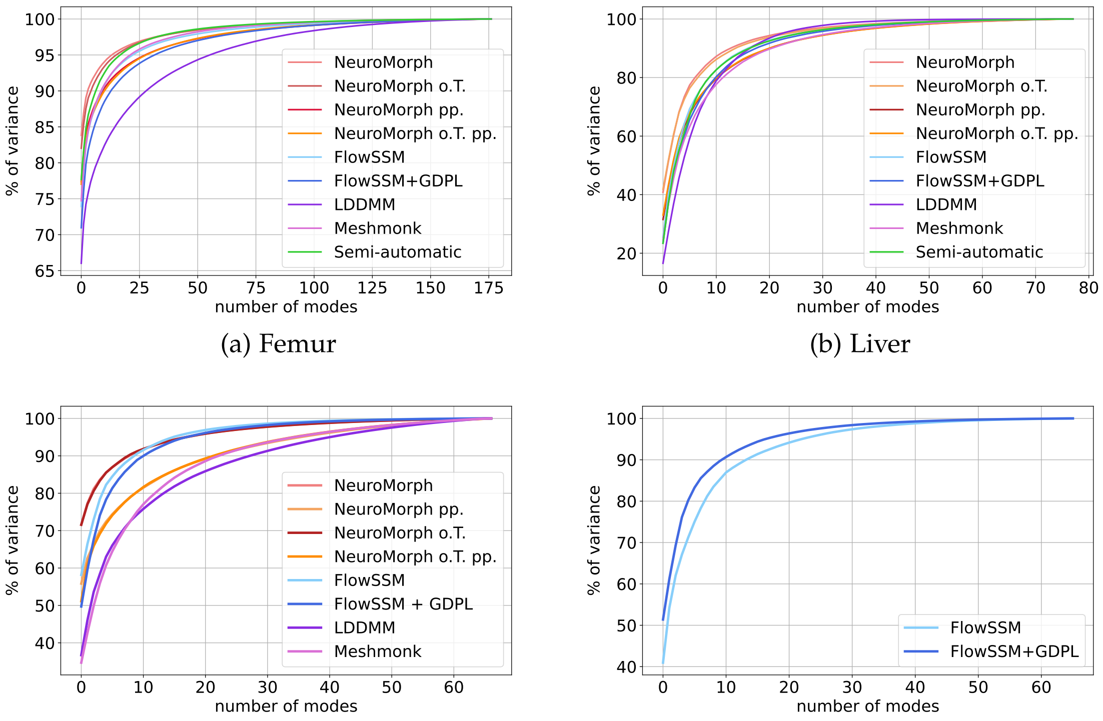
The remeshing of the face dataset leads to an decrease of all error metrics. This is not surprising, as it is easier to display a surface if more vertices are available. However, the results with the GDPL are improved significantly, as the proportion of vertices whose neighborhood is affected by the surfaces edge is reduced significantly.
One big advantage of FlowSSM in its original form is its independence towards differences

Table 4.10: SSM quality of different FlowSSM variants on all datasets. If the addition of the GDPL leads to statistically significant deviations (paired t-test, $p < 0.05$) from the new baseline, the results are marked with a "*". Only error metrics were tested.

Dataset	**Method**	**Generality**			**Specificity**		
		Error in mm	**SIM**	**SIF**	**Error in mm**	**SIM**	**SIF**
Distal Femur	**Original**	0.27 ± 0.05	0 %	0	1.12 ± 0.12	0 %	0
	New baseline	0.25 ± 0.05	0 %	0	1.11 ± 0.12	0 %	0
	with GDPL	0.25 ± 0.05	6 %	25	1.11 ± 0.12	0 %	0
Liver	**Original**	1.82 ± 0.38	35 %	70	5.04 ± 0.67	5 %	64
	New baseline	1.80 ± 0.39	4 %	111	5.01 ± 0.65	2 %	54
	with GDPL	1.68 ± 0.28*	9 %	70	4.95 ± 0.62*	5 %	62
Face	**Original**	1.13 ± 0.25	11 %	40	1.82 ± 0.20	0 %	0
	New baseline	1.14 ± 0.20	11 %	25	1.82 ± 0.21	0 %	0
	with GDPL	1.13 ± 0.28	37 %	37	1.73 ± 0.20*	1 %	14
Face remesh	**Original**	1.09 ± 0.25	26 %	101	1.99 ± 0.28	37 %	145
	New baseline	0.95 ± 0.25	42 %	195	1.58 ± 0.21	0 %	0
	with GDPL	0.83 ± 0.22*	42 %	245	1.73 ± 0.20*	3 %	68

in the mesh resolutions, as stated in Section 4.1.4. The advantage gets diminished by the use of the GDPL since the original vertices are used in the loss function. This could partly explain the rise of the landmark error on the face dataset.

Subsuming we can state that the use of the GDPL improves the correspondences on a global level, but individual landmarks could suffer from a reduced correspondence accuracy. At the same time the method loses robustness, as it is more influenced by the meshing of the template in terms of general resolution and differently meshed regions. The functionality is also decreased when applied to data whose surfaces are bordered.



(a) Femur

(b) Liver

(c) Face

(d) Face remeshed

Figure 4.4: Compactness of the different SSMs per dataset. An ideal SSM captures the whole variation of the population in only a few eigenmodes.

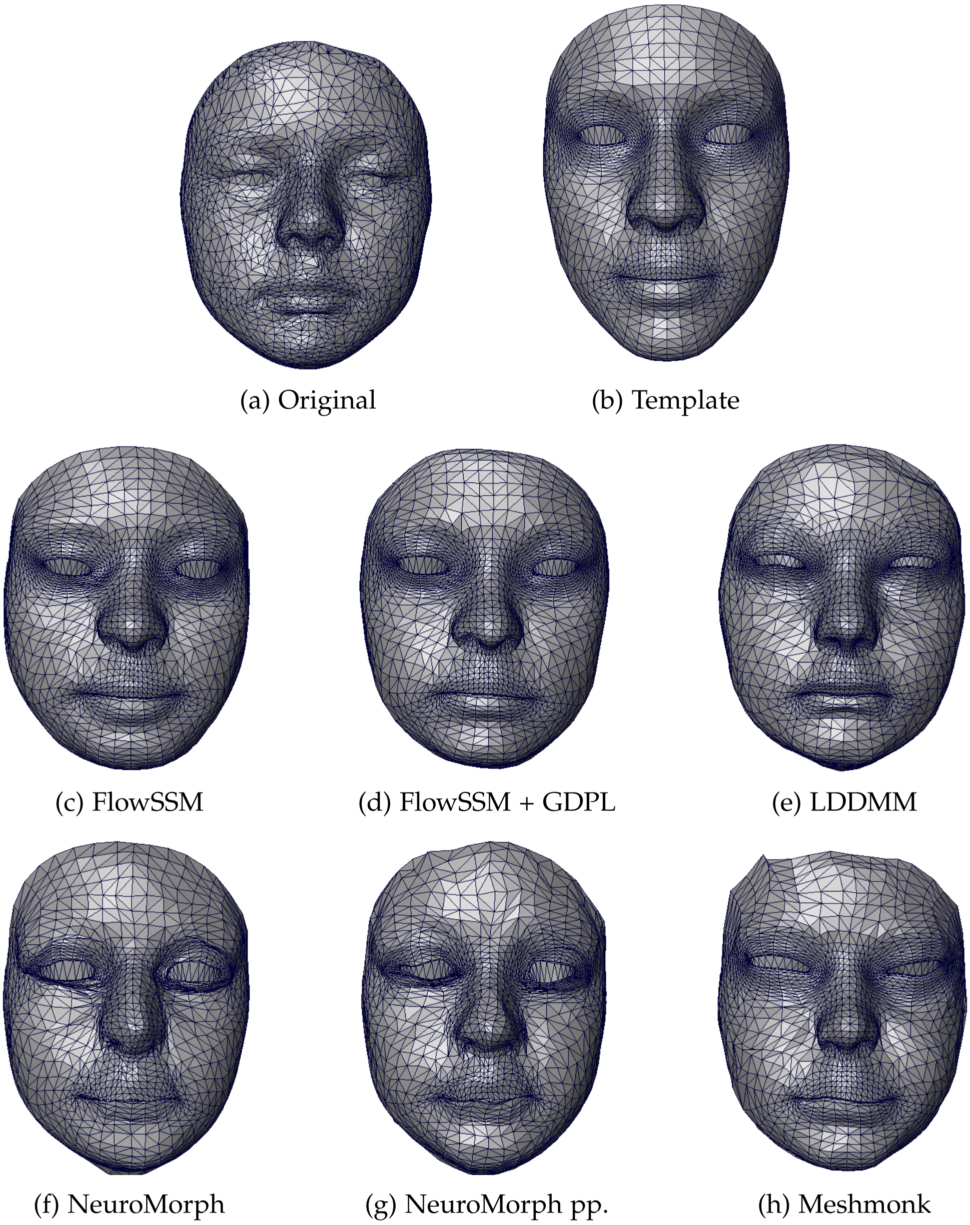


Figure 4.5: Example of the generality experiment on the face data. The original shape (a) was projected into the SSM. Thereby, the eigenmodes were wheigted in a way to transform the template (b) surface towards the original shape (a). The images (c) - (h) show the results of the different methods.

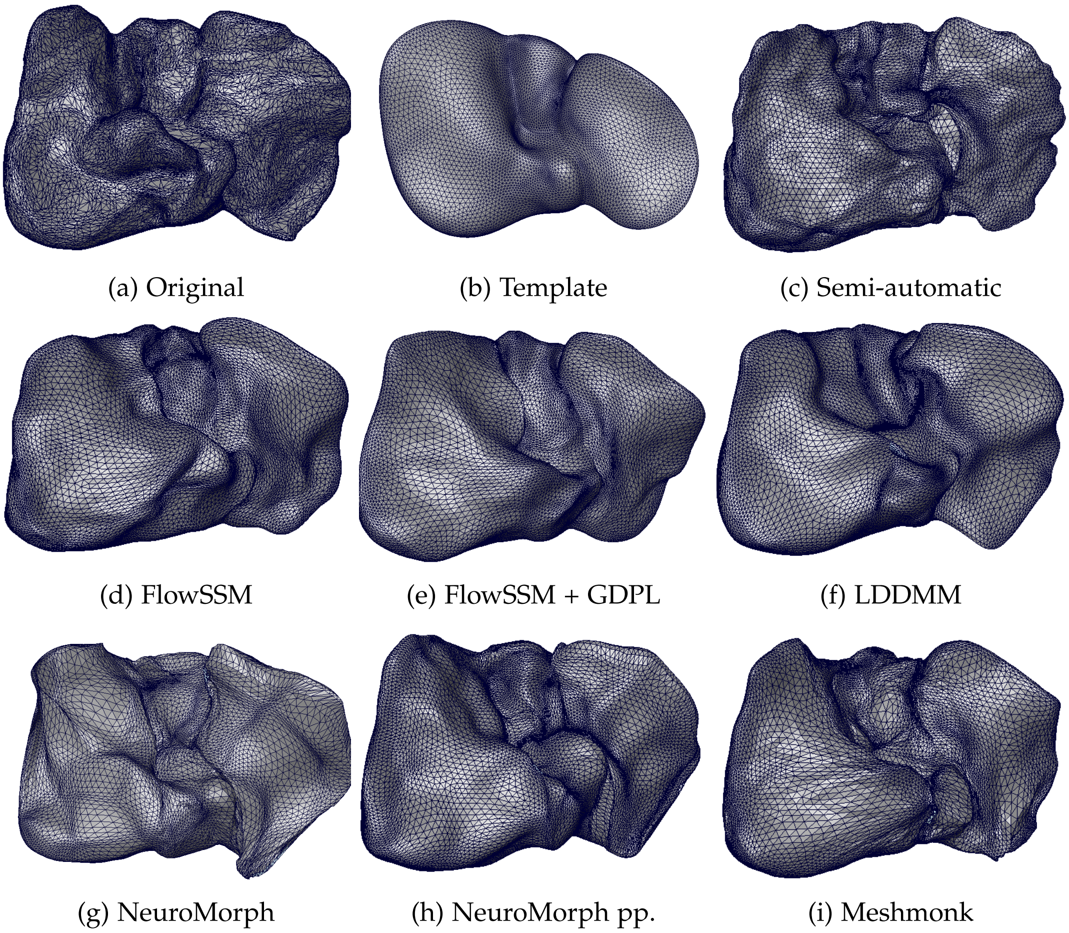

(a) Original
(b) Template
(c) Semi-automatic
(d) FlowSSM
(e) FlowSSM + GDPL
(f) LDDMM
(g) NeuroMorph
(h) NeuroMorph pp.
(i) Meshmonk

Figure 4.6: Example of the generality experiment on the liver data. The original shape (a) was projected into the SSM. Thereby, the eigenmodes were wheigted in a way to transform the template (b) surface towards the original shape (a). The images (c) - (i) show the results of the different methods.

5 Conclusion and Future Work

The aim of this thesis was to investigate different methods that can be used for correspondence estimation of anatomical shapes. In order to do so, we evaluated the performance of the following methods: LDDMM, Meshmonk, NeuroMorph and FlowSSM. The first two methods are already established in the medical domain. NeuroMorph was never tested on anatomical data before, as it is a data-driven method from the wider computer vision community. FlowSSM on the other hand, was initially developed to build shape models of anatomical shapes, but the resulting correspondences were not evaluated beforehand. Furthermore, the GDPL was added to FlowSSM in order to improve the quality of the generated correspondences.

Since there is no reproducible ground truth for the correspondence of anatomical shapes, the evaluation itself is a challenge. It is therefore mandatory to evaluate the correspondence quality with a set of indirect metrics. The metrics used in this thesis include the Chamfer distance, the sparse correspondence of anatomical landmarks and the quality of resulting SSMs. The evaluation was performed on three datasets with different degrees of geometric variation, namely distal femur, liver and face.

In this final chapter of this thesis we want to summarize the findings of all experiments and give an outlook towards possible future work. To this end, we divide our findings and suggestions into the following categories: Statements regarding the evaluation of correspondences in general, recommendations regarding the suitable selection of a method used for correspondence estimation and special findings regarding the method FlowSSM and the addition of the GDPL.

5.1 Conclusion

Evaluation of Shape Correspondence: During the evaluation we found that the quality of the results varied across all methods, datasets and metrics. We can thereby conclude that it is important to evaluate correspondence on datasets with different features such as topology and degree of geometric variation when benchmarking various methods against each other. Furthermore, we observed that the metrics can sometimes contradict each other.

In order to ensure a good correspondence, it is therefore recommended to evaluate a set of metrics, ideally related to the field of application afterwards.

The assumption of many methods that the deformation process moves vertices to semantically corresponding positions could not be unanimously confirmed in this work. Thus, a deformation that minimises the surface distance between the deformed template and the target does not automatically produce a good correspondence. This statement mainly relies on the LDDMM results on the face dataset, where a low surface error between deformed template and target surfaces still lead to an above average landmark error. The observations on the postprocessed NeuroMorph variants, where a low surface error produced a low SSM quality support this statement.

Moreover, we saw that surface error, landmark error and SSM quality often contradict each other. This observation could be explained by different expressions of correspondence quality. The SSMs mostly require a consistent correspondence, i.e. that each point on the template is deformed towards the same location on every mesh. The source location on the template and the destination locations do not necessarily have to be the semantically same. In order to evaluate the sparse correspondence on special landmarks on the other hand, the deformations have to move those points to their anatomically similar location. Here, it is not enough if the correspondences are consistent, they have to be objectively correct.

FlowSSM and GDPL: The method FlowSSM has certain advantages when compared to the other methods tested in this thesis. As it does not directly use the vertices of each mesh, it is independent towards the resolution of the meshes. It can also create meshes with different resolutions of the same surface. If the GDPL is added, this advantage is reduced, as the loss function needs a predefined set of vertices. The loss term also works better, if the data is of high resolution. Furthermore, problems occur if the loss function is applied in an area close to the border of a surface. Lastly, training and predictions with GDPL need more time than with the standard setting.

However, the addition of the GDPL can lead to a significant improvement of the generalization ability and specificity of the resulting SSM. If these qualities are important to the use-case and the dataset is of high resolution, it is recommendable to apply the GDPL.

Selection of a Method used for Shape Correspondence Estimation: Since the SSM quality based on the semi-automatically generated correspondences was on the same level or even lower than quality based on fully automatically generated correspondences, the time-intensiv semi-automatic generation of correspondence does not seem worthwhile.

As opposed to literature, we could not see a clear superiority of groupwise methods. Because the performance of the different methods varied for each dataset and metric, this observation could also result from other method properties than the group- or pairwise training.
The performance of the different methods hardly varied, when the dataset exhibited only low levels of geometric variations (e.g. the distal femur). On all other datasets, we saw significant differences between the methods but no clear ranking. If the data has a lot of geometric variance, it is therefore recommendable to choose the method according to the downstream task. For applications that require a objectively correct correspondence, the method FlowSSM seems suitable. If not enough data for training is available, Meshmonk could be used. If the downstream task is in need of an SSM, FlowSSM with the GDPL might be a good choice. The method LDDMM is a good alternative for building SSMs if less data is available. NeuroMorph, on the other hand, did not produce convincing results in the evaluation of this thesis.

5.2 Future Work

Evaluation of Shape Correspondence While this thesis already used a lot of metrics for the evaluation of the correspondence quality, there are still some suggestions from literature not yet implemented. Goparaju et al. [Gop+22] suggested to evaluate the correspondence on an actual SSM applications such as a reconstruction task or a disease classification. Kaick et al. [Kai+11] recommend to evaluate correspondence on a synthetic dataset with ground truth annotations. The challange here lies in finding or creating a synthetic dataset of anatomical data with a sufficient degree of realism.

As the results significantly differed on each dataset, it seems recommendable to test on even more datasets. New datasets should pose new challenges, such as a different topology (e.g. a torus or a surface with holes).
We previously discussed, that there might be different qualities of correspondence responsible for contradicting results on different metrics. Future work is needed to prove this assumption.

FlowSSM and GDPL: While the addition of the GDPL improves the resulting correspondence of FlowSSM on a global scale, it also brings some downsides. However, there are some ideas for future improvements: The use of the GDPL is limited at the edge of a surface. One easy way to circumvent this problem would be to not apply the GDPL on

vertices located on the edge of a surface.
Another possible improvement for the nearest neighbor search could include the connectivity information of the mesh. Since the nearest neighbor search is based on Euclidean distances, the current formulation has a risk of adding a wrong neighbor in an area with high curvature. A geodesic-based neighborhood search would eliminate this issue and could thereby minimize the number of self-intersections created on the deformed template.

In the previous chapter it was stated, that FlowSSM is more invariant towards the mesh structure than other methods. The use of the GDPL could limit this advantage, as it depends on the given surface discretization. Future work could investigate, how far different mesh structures influence the results of FlowSSM with GDPL.

Other Methods used for Shape Correspondence Estimation: As this thesis did not find a clear superiority of groupwise methods, this topic could be investigated in future research. An easy way to prove groupwise superiority would be to train a learning based method (i.e. FlowSSM or NeroMorph) for each target and use only a single target mesh as training data. If the resulting deformed templates have a lower quality than the ones trained on the whole dataset, the method clearly profits from the groupwise setting.

One way to improve the performance of the learning based methods, could be the addition of a supervised landmark error loss term. This could improve the sparse correspondence on these points. But as labels are needed, the method would not be unsupervised anymore. Furthermore, most of the methods strive to minimize a symmetrical surface distance. If template and target consist of the same corresponding points, this is a good choice. If, however, one of the two covers anatomical areas that are not present in the other this leads to problems, as there is no good correspondence possible in both directions. For those cases a one-sided loss term would be more suitable and make the method more robust in future work.

Bibliography

[Agi+20] R. Agier et al. "Hubless keypoint-based 3D deformable groupwise registration". In: *Medical Image Analysis* 59 (2020), p. 101564. DOI: 10.1016/j.media.2019.101564.

[ALC20] M. Aygün, Z. Lähner, and D. Cremers. "Unsupervised Dense Shape Correspondence using Heat Kernels". In: *2020 International Conference on 3D Vision (3DV)* (2020), pp. 573–582.

[Alo+22] A. Alomar et al. "Reconstruction of the fetus face from three-dimensional ultrasound using a newborn face statistical shape model". In: *Computer Methods and Programs in Biomedicine* 221 (2022), p. 106893. DOI: 10.1016/j.cmpb.2022.106893.

[Amb+19a] F. Ambellan et al. "Automated segmentation of knee bone and cartilage combining statistical shape knowledge and convolutional neural networks: Data from the Osteoarthritis Initiative". In: *Medical Image Analysis* 52 (2019), pp. 109–118. DOI: 10.1016/j.media.2018.11.009.

[Amb+19b] F. Ambellan et al. "Statistical Shape Models: Understanding and Mastering Variation in Anatomy". In: *Springer International Publishing* (2019), pp. 67–84. DOI: 10.1007/978-3-030-19385-0_5.

[AZT21] F. Ambellan, S. Zachow, and C. von Tycowicz. "Rigid Motion Invariant Statistical Shape Modeling based on Discrete Fundamental Forms". In: *Medical Image Analysis* 73 (2021), p. 102178. DOI: 10.1016/j.media.2021.102178.

[Bay+19] S. Bayer et al. "Registration of vascular structures using a hybrid mixture model". In: *International Journal of Computer Assisted Radiology and Surgery* 14.9 (2019), pp. 1507–1516. DOI: 10.1007/s11548-019-02007-y.

[Ber+17] F. Bernard et al. "Shape-aware Surface Reconstruction from Sparse 3D Point-Clouds". In: *Medical Image Analysis* 38 (2017), pp. 77–89. DOI: 10.1016/j.media.2017.02.005.

[BGK95] C. Brechbühler, G. Gerig, and O. Kübler. "Parametrization of Closed Surfaces for 3-D Shape Description". In: *Computer Vision and Image Understanding* 61.2 (1995), pp. 154–170. DOI: 10.1006/cviu.1995.1013.

[BM92] P. J. Besl and N. D. McKay. "Method for registration of 3-D shapes". In: Robotics - DL tentative. Boston, MA, 1992, pp. 586–606. DOI: 10.1117/12.57955.

[Bog+14] F. Bogo et al. "FAUST: Dataset and Evaluation for 3D Mesh Registration". In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2014, pp. 3794–3801. DOI: 10.1109/CVPR.2014.491.

[BR07] B. J. Brown and S. Rusinkiewicz. "Global non-rigid alignment of 3-D scans". In: *ACM Transactions on Graphics* 26.3 (2007), p. 21. DOI: 10.1145/1276377.1276404.

[Bru+14] A. Brunton et al. "Review of statistical shape spaces for 3D data with comparative analysis for human faces". In: *Computer Vision and Image Understanding* 128 (2014), pp. 1–17. DOI: 10.1016/j.cviu.2014.05.005.

[Bru+17] J. L. Bruse et al. "Detecting Clinically Meaningful Shape Clusters in Medical Image Data: Metrics Analysis for Hierarchical Clustering Applied to Healthy and Pathological Aortic Arches". In: *IEEE Trans. Biomed. Eng.* 64.10 (2017), pp. 2373–2383. DOI: 10.1109/TBME.2017.2655364.

[BT00] A. D. Brett and C. J. Taylor. "A Method of Automated Landmark Generation for Automated 3D PDM Construction". In: *Image and Vision Computing* 18.9 (2000), pp. 739–748.

[CEW17] J. Cates, S. Elhabian, and R. Whitaker. "Chapter 10 - ShapeWorks: Particle-Based Shape Correspondence and Visualization Software". In: *Statistical Shape and Deformation Analysis*. Academic Press, 2017, pp. 257–298. DOI: 10.1016/B978-0-12-810493-4.00012-2.

[Che+21] A.-C. Cheng et al. "Learning 3D Dense Correspondence via Canonical Point Autoencoder". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6608–6620.

[Coo+95] T. Cootes et al. "Active Shape Models-Their Training and Application". In: *Computer Vision and Image Understanding* 61.1 (1995), pp. 38–59. DOI: 10.1006/cviu.1995.1004.

[Coo20] T. Cootes. "Linear statistical shape models and landmark location". In: *Handbook of Medical Image Computing and Computer Assisted Intervention*. Elsevier, 2020, pp. 575–598. DOI: 10.1016/B978-0-12-816176-0.00029-6.

[Dal+19] A. V. Dalca et al. "Unsupervised learning of probabilistic diffeomorphic registration for images and surfaces". In: *Medical Image Analysis* 57 (2019), pp. 226–236. DOI: 10.1016/j.media.2019.07.006.

[DCT01] R. H. Davies, T. F. Cootes, and C. J. Taylor. "A Minimum Description Length Approach to Statistical Shape Modelling". In: *Information Processing in Medical Imaging*. Vol. 2082. Springer Berlin Heidelberg, 2001, pp. 50–63.

[Dur+14] S. Durrleman et al. "Morphometry of anatomical shape complexes with dense deformations and sparse parameters". In: *NeuroImage* 101 (2014), pp. 35–49. DOI: 10.1016/j.neuroimage.2014.06.043.

[Dyk+20] R. M. Dyke et al. "SHREC'20: Shape correspondence with non-isometric deformations". In: *Computers & Graphics* 92 (2020), pp. 28–43. DOI: 10.1016/j.cag.2020.08.008.

[DYT05] H. Q. Dinh, A. Yezzi, and G. Turk. "Texture transfer during shape transformation". In: *ACM Transactions on Graphics* 24.2 (2005), pp. 289–310. DOI: 10.1145/1061347.1061353.

[DYT21] Y. Deng, J. Yang, and X. Tong. "Deformed Implicit Field: Modeling 3D Shapes with Learned Dense Correspondence". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 10281–10291. DOI: 10.1109/CVPR46437.2021.01015.

[Eis+20] M. Eisenberger et al. "Deep Shells: Unsupervised Shape Correspondence with Optimal Transport". In: *Advances in Neural information processing systems* 33 (2020), pp. 10491–10502.

[Eis+21] M. Eisenberger et al. "NeuroMorph: Unsupervised Shape Interpolation and Correspondence in One Go". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 7469–7479. DOI: 10.1109/CVPR46437.2021.00739.

[EK07] A. Ericsson and J. Karlsson. "Measures for Benchmarking of Automatic Correspondence Algorithms". In: *Journal of Mathematical Imaging and Vision* 28.3 (2007), pp. 225–241. DOI: 10.1007/s10851-007-0018-5.

[ELC20] M. Eisenberger, Z. Lahner, and D. Cremers. "Smooth Shells: Multi-Scale Shape Registration With Functional Maps". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 12262–12271. DOI: 10.1109/CVPR42600.2020.01228.

[Fas+98] J. H. Fasel et al. "Segmental anatomy of the liver: poor correlation with CT." In: *Radiology* 206.1 (1998), pp. 151–156. DOI: 10.1148/radiology.206.1.9423665.

[Fle+04] P. Fletcher et al. "Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape". In: *IEEE Trans. Med. Imaging* 23.8 (2004), pp. 995–1005. DOI: 10.1109/TMI.2004.831793.

[Gop+22] A. Goparaju et al. "Benchmarking off-the-shelf statistical shape modeling tools in clinical applications". In: *Medical Image Analysis* 76 (2022), p. 102271.

[Gow10] J. C. Gower. "Procrustes methods: Procrustes methods". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.4 (2010), pp. 503–508. DOI: 10.1002/wics.107.

[Gow75] J. C. Gower. "Generalized procrustes analysis". In: *Psychometrika* 40.1 (1975), pp. 33–51. DOI: 10.1007/BF02291478.

[Gro+18] T. Groueix et al. "3D-CODED: 3D Correspondences by Deep Deformation". In: *Computer Vision – ECCV 2018*. Vol. 11206. Springer International Publishing, 2018, pp. 235–251.

[Gro+19] T. Groueix et al. "Unsupervised cycle-consistent deformation for shape matching". In: *Computer Graphics Forum* 38.5 (2019), pp. 123–133.

[GZ16] C. M. Grewe and S. Zachow. "Fully Automated and Highly Accurate Dense Correspondence for Facial Surfaces". In: *Computer Vision – ECCV 2016 Workshops*. Vol. 9914. Springer International Publishing, 2016, pp. 552–568.

[Hal+19] O. Halimi et al. "Unsupervised Learning of Dense Shape Correspondence". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 4365–4374. DOI: 10.1109/CVPR.2019.00450.

[HM09] T. Heimann and H.-P. Meinzer. "Statistical shape models for 3D medical image segmentation: A review". In: *Medical Image Analysis* 13.4 (2009), pp. 543–563. DOI: 10.1016/j.media.2009.05.004.

[Jia+21] C. " Jiang et al. "ShapeFlow: Learnable Deformations Among 3D Shapes". In: *Advances in Neural Information Processing Systems* 33 (2021), pp. 9745–9757.

[Kai+11] O. van Kaick et al. "A Survey on Shape Correspondence". In: *Computer Graphics Forum* 30.6 (2011), pp. 1681–1707. DOI: 10.1111/j.1467-8659.2011.01884.x.

[KBW11] M. Kirschner, M. Becker, and S. Wesarg. “3D Active Shape Model Segmentation with Nonlinear Shape Priors”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011* 6892 (2011), pp. 492–499.

[KLL07] D. Kainmuller, T. Lange, and H. Lamecker. “Shape Constrained Automatic Segmentation of the Liver based on a Heuristic Intensity Model”. In: *Proc. MICCAI Workshop 3D Segmentation in the Clinic: A Grand ChallengeProc.* 109 (2007), p. 116.

[KS04] V. Kraevoy and A. Sheffer. “Cross-parameterization and compatible remeshing of 3D models”. In: *ACM Transactions on Graphics (ToG)* 23.3 (2004), pp. 861–869.

[KS98] R. Kimmel and J. A. Sethian. “Computing geodesic paths on manifolds”. In: *Proceedings of the National Academy of Sciences* 95.15 (1998), pp. 8431–8435. DOI: 10.1073/pnas.95.15.8431.

[Lam08] H. Lamecker. “Variational and statistical shape modeling for 3D geometry reconstruction”. PhD thesis. Freie Universität Berlin, 2008.

[Lan+21] I. Lang et al. “DPC: Unsupervised Deep Point Correspondence via Cross and Self Construction”. In: *2021 International Conference on 3D Vision (3DV)*. 2021, pp. 1341–1350. DOI: 10.1109/3DV53792.2021.00141.

[Leb+22] L. Lebrat et al. “CorticalFlow: A Diffeomorphic Mesh Deformation Module for Cortical Surface Reconstruction”. In: *arXiv preprint arXiv:2206.02374* (2022), p. 15.

[Lec+12] F. Lecron et al. “Fast 3D Spine Reconstruction of Postoperative Patients Using a Multilevel Statistical Model”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*. Vol. 7511. Springer Berlin Heidelberg, 2012, pp. 446–453.

[Li+21] J. Li et al. “AutoImplant 2020-First MICCAI Challenge on Automatic Cranial Implant Design”. In: *IEEE Transactions on Medical Imaging* 40.9 (2021), pp. 2329–2342. DOI: 10.1109/TMI.2021.3077047.

[Lüd+22] D. Lüdke et al. “Landmark-Free Statistical Shape Modeling Via Neural Flow Deformations”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Vol. 13432. Springer Nature Switzerland, 2022, pp. 453–463.

[Lüd22] D. Lüdke. *Neural Flow-based Deformations for Statistical Shape Modelling*. Master’s Thesis. 2022.

[Ma+19] J. Ma et al. “A novel robust kernel principal component analysis for nonlinear statistical shape modeling from erroneous data”. In: *Computerized Medical Imaging and Graphics* 77 (2019), p. 101638. DOI: 10.1016/j.compmedimag.2019.05.006.

[MDH18] S. Mambo, K. Djouani, and Y. Hamam. “A Review on Medical Image Registration Techniques”. In: *International Journal of Computer and Information Engineering* 12.1 (2018), pp. 48–55.

[MDS08] B. Munsell, P. Dalal, and Song Wang. “Evaluating Shape Correspondence for Statistical Shape Analysis: A Benchmark Study”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 30.11 (2008), pp. 2023–2039. DOI: 10.1109/TPAMI.2007.70841.

[Mil+03] A. Miller et al. “Automatic grasp planning using shape primitives”. In: *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*. IEEE, 2003, pp. 1824–1829. DOI: 10.1109/ROBOT.2003.1241860.

[Ngu+14] T. Nguyen et al. “Use of shape correspondence analysis to quantify skeletal changes associated with bone-anchored Class III correction”. In: *The Angle Orthodontist* 84.2 (2014), pp. 329–336. DOI: 10.2319/041513-288.1.

[Ovs+12] M. Ovsjanikov et al. “Functional maps: a flexible representation of maps between shapes”. In: *ACM Transactions on Graphics* 31.4 (2012), pp. 1–11. DOI: 10.1145/2185520.2185526.

[Par+19] J. J. Park et al. “DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 165–174. DOI: 10.1109/CVPR.2019.00025.

[Rav+18] N. Ravikumar et al. “Group-wise similarity registration of point sets using Student’s t-mixture model for statistical shape models”. In: *Medical Image Analysis* 44 (2018), pp. 156–176. DOI: 10.1016/j.media.2017.11.012.

[RDT06] Y. Rathi, S. Dambreville, and A. Tannenbaum. “Statistical shape analysis using kernel PCA”. In: *Electronic Imaging 2006*. San Jose, CA, 2006, 60641B. DOI: 10.1117/12.641417.

[RSO19] J.-M. Roufosse, A. Sharma, and M. Ovsjanikov. “Unsupervised Deep Learning for Structured Shape Matching”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2019, pp. 1617–1627. DOI: 10.1109/ICCV.2019.00170.

[SA07] O. Sorkine and M. Alexa. “As-Rigid-As-Possible Surface Modeling”. In: *Symposium on Geometry processing* 4 (2007), pp. 109–116.

[Sah20] Y. Sahillioğlu. “Recent advances in shape correspondence”. In: *The Visual Computer* 36.8 (2020), pp. 1705–1721. DOI: 10.1007/s00371-019-01760-0.

[SP04] R. W. Sumner and J. Popovic. “Deformation transfer for triangle meshes”. In: *ACM Transactions on graphics (TOG)* 23.3 (2004), pp. 399–405.

[SSL06] K. Sjöstrand, M. B. Stegmann, and R. Larsen. “Sparse principal component analysis in medical shape modeling”. In: *Medical Imaging 2006: Image Processing*. Vol. 6144. 2006, pp. 1579–1590. DOI: 10.1117/12.651658.

[STD14] S. Salti, F. Tombari, and L. Di Stefano. “SHOT: Unique signatures of histograms for surface and texture description”. In: *Computer Vision and Image Understanding* 125 (2014), pp. 251–264. DOI: 10.1016/j.cviu.2014.04.011.

[Sui+04] A. Suinesiaputra et al. “Extraction of Myocardial Contractility Patterns from Short-Axes MR Images Using Independent Component Analysis”. In: *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*. Vol. 3117. Springer Berlin Heidelberg, 2004, pp. 75–86.

[SWZ14] N. Sarkalkan, H. Weinans, and A. A. Zadpoor. “Statistical shape and appearance models of bones”. In: *Bone* 60 (2014), pp. 129–140. DOI: 10.1016/j.bone.2013.12.006.

[Tan+19] Z. Tang et al. “An Augmentation Strategy for Medical Image Processing Based on Statistical Shape Model and 3D Thin Plate Spline for Deep Learning”. In: *IEEE Access* 7 (2019), pp. 133111–133121. DOI: 10.1109/ACCESS.2019.2941154.

[Tiw21] S. Tiwari. *Facial Landmark Detection on 3D Surface Scans using Geometric Deep learning*. Master’s Thesis. 2021.

[Tót+20] K. Tóthová et al. “Probabilistic 3D surface reconstruction from sparse MRI information”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer* (2020), pp. 813–823.

[Tra+21] G. Trappolini et al. “Shape registration in the time of transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 5731–5744.

[Tyc+18] C. von Tycowicz et al. “An efficient Riemannian statistical shape model using differential coordinates”. In: *Medical Image Analysis* 43 (2018), pp. 1–9. DOI: 10.1016/j.media.2017.09.004.

[Uy+] M. A. Uy et al. “Joint Learning of 3D Shape Retrieval and Deformation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (), pp. 11713–11722.

[Wan+19a] X. Wang et al. “Shape2Motion: Joint Analysis of Motion Parts and Attributes From 3D Shapes”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 8868–8876. DOI: `10.1109/CVPR.2019.00908`.

[Wan+19b] Y. Wang et al. “Dynamic Graph CNN for Learning on Point Clouds”. In: *ACM Transactions on Graphics* 38.5 (2019), pp. 1–12. DOI: `10.1145/3326362`.

[Wan14] Q. Wang. “Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models”. In: *arXiv preprint arXiv:1207.3538* (2014).

[Whi+19] J. D. White et al. “MeshMonk: Open-source large-scale intensive 3D phenotyping”. In: *Scientific Reports* 9.1 (2019), p. 6085. DOI: `10.1038/s41598-019-42533-y`.

[Xu+10] K. Xu et al. “Style-content separation by anisotropic part scales”. In: *ACM SIGGRAPH Asia 2010 papers on - SIGGRAPH ASIA '10*. 2010, p. 1. DOI: `10.1145/1882262.1866206`.

[YS19] Z. Yan and S. Schaefer. “A Family of Barycentric Coordinates for Co-Dimension 1 Manifolds with Simplicial Facets”. In: *Computer Graphics Forum*. Vol. 38. 5. Wiley Online Library. 2019, pp. 75–83.

[Zen+21] Y. Zeng et al. “CorrNet3D: Unsupervised End-to-end Learning of Dense Correspondence for 3D Point Clouds”. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 6048–6057. DOI: `10.1109/CVPR46437.2021.00599`.

[Zho+16] T. Zhou et al. “Learning Dense Correspondence via 3D-Guided Cycle Consistency”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, 2016, pp. 117–126. DOI: `10.1109/CVPR.2016.20`.

Appendix

Table A1: Definitions of anatomical landmarks on face dataset by experts from Universitätsklinikum Ulm, taken from [Tiw21].

Abbreviation	**Name**	**Definition**
ALAL	Alare curvature	Most lateral point of attachment of the nasal wing to the cheek (on left side of the face).
ALAR	Alare curvature	Most lateral point of attachment of the nasal wing to the cheek (on right side of the face).
CHL	Cheilion	Lateral point at the corner of the mouth (on left side of the face).
CHR	Cheilion	Lateral point at the corner of the mouth (on right side of the face).
COL	Columnella	Most convex point of the nose bridge.
ENL	Endocanthion	Point at the medial commissure of the palpebral fissure (on left side of the face).
ENR	Endocanthion	Point at the medial commissure of the palpebral fissure (on right side of the face).
EXL	Exocanthion	Point at the lateral commissure of the palpebral fissure (on left side of the face).
EXR	Exocanthion	Point at the lateral commissure of the palpebral fissure (on right side of the face).
GB	Glabella	Medial point at greatest convexity above the root of the nose.
GN	Gnathion	Most anterior and caudal point on the chin.
LI	Labiale inferius	Medial point at transition from lower lip red to lower lip white.
LIKONVEX	Labiale inferius konvex	Anteriormost point on lower lip red.

Abbreviation	Name	Definition
LS	Labiale superius	Medial point at the transition from upper lip white to upper lip red.
LSKONVEX	Labiale superius konvex	Anteriormost point on upper lip red.
N	Nasion	Medial point at deepest retraction between forehead and bridge of nose.
OL	Orbitale	Point at the lower edge of the orbit, one eyelid slit width below the pupil of the unconstrained, straight-eyed eye (on left side of the face).
OR	Orbitale	Point at the lower edge of the orbit, one eyelid slit width below the pupil of the unconstrained, straight-eyed eye (on right side of the face).
PHL	Philtrum	Transition lip red lip white (on left side of the face).
PHR	Philtrum	Transition lip red lip white (on right side of the face).
POG	Pogonion	Medial most anterior point on the chin.
PRN	Pronasale	Most anterior center of the nasal tip.
SM	Supramentale	Medial point at most concave location between lower lip and chin.
SN	Subnasale	Medial point at the transition of the columella into the philtrum.
SSP	Subspinale	Medial point at greatest concavity of philtrum.
STOINF	Stomion inferior	Medial point at the oral fissure.
STOSUP	Stomion superior	Medial point at the oral fissure.
ZYGL	Zygion	Point at widest part of face at zygomatic arch, constructed by crossing lines exocanthion to lowermost point at earlobe and orbital to porion (on left side of the face).

Abbreviation	Name	Definition
ZYGR	Zygion	Point at widest part of face at zygomatic arch, constructed by crossing lines exocanthion to lowermost point at earlobe and orbital to porion (on right side of the face).