UNIVERSITÄT
KOBLENZ · LANDAU
Fachbereich 4: Informatik

Fraunhofer
IDM@NTU
Fraunhofer IDM@NTU Singapur

# Robust Statistical Shape Modeling from Routine Clinical Data

Masterarbeit

zur Erlangung des Grades eines Master of Science (M.Sc.)
im Studiengang Computervisualistik

vorgelegt von

Katharina Lentzen &
Jonas Honsdorf

| | |
|---|---|
| Erstgutachter: | Prof. Dr.-Ing. Stefan Müller<br>(Institut für Computervisualistik, AG Computergraphik) |
| Zweitgutachter: | Dr.-Ing. Marius Erdt<br>(Fraunhofer IDM@NTU Singapur) |

Koblenz, im März 2016

# Erklärung

Ich versichere, dass ich den Abschnitt / die Abschnitte ........ der vorliegenden Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

| | Ja | Nein |
|---|---|---|
| Mit der Einstellung der Arbeit in die Bibliothek bin ich einverstanden. | □ | □ |
| Der Veröffentlichung dieser Arbeit im Internet stimme ich zu. | □ | □ |

..........................................................................

(Ort, Datum) (Unterschrift)

Ich versichere, dass ich den Abschnitt / die Abschnitte ........ der vorliegenden Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

| | Ja | Nein |
|---|---|---|
| Mit der Einstellung der Arbeit in die Bibliothek bin ich einverstanden. | □ | □ |
| Der Veröffentlichung dieser Arbeit im Internet stimme ich zu. | □ | □ |

..........................................................................

(Ort, Datum) (Unterschrift)

Alle Abschnitte, die nicht gekennzeichnet sind, gelten als gemeinsam verantwortet.

Institut für Computervisualistik
AG Computergraphik
Prof. Dr. Stefan Müller
Postfach 20 16 02
56 016 Koblenz
Tel.: 0261-287-2727
Fax: 0261-287-2735
E-Mail: stefanm@uni-koblenz.de

UNIVERSITÄT
KOBLENZ · LANDAU

Fachbereich 4: Informatik

Aufgabenstellung für die Masterarbeit
Jonas Honsdorf
(Mat. Nr. 209 210 132)

**Topic: Robust statistical shape modeling from routine clinical data – Model building approaches**

Statistical Shape Models (SSMs) are a well-established tool for 3D image analysis and especially medical image segmentation. For example, they have been successfully applied to model all major organs and bone structures as well as to all important imaging modalities like CT, MRI, ultrasound and other. Building a high quality SSM requires manually generated ground truth data from clinical experts. Unfortunately, the acquisition of such data is a time-consuming process and prone to error.

The aim of this work is to construct a SSM with reasonable quality, without the need of manual image interpretation. By using any segmentation algorithm, the idea is to exploit the multitude of daily created clinical image data. Although, the output can contain errors to a higher or lower degree, the statistics inherent in such data can be utilized.

In particular, approaches of model building should be developed and robustified against corrupted data. Especially, the question if the statistical information in the erroneous data is sufficient to reconstruct shapes similar to the ground truth should be answered.

Key aspects of the work:

1. Research about Statistical Shape Models and handling unreliable data
2. Data acquisition of a particular organ
3. Developing a framework for robust model building
4. Research about evaluation techniques of statistical shape models
5. Developing a framework for evaluation
6. Evaluation
7. Documentation and discussion about the results

The work is implemented in cooperation with Fraunhofer IDM@NTU in Singapore.

Supervisor: Dr. Marius Erdt (Fraunhofer IDM@NTU)

Koblenz, 10.09.2015

Jonas Honsdorf

– Jonas Honsdorf –

– Prof. Dr. Stefan Müller –

Institut für Computervisualistik
AG Computergraphik
Prof. Dr. Stefan Müller
Postfach 20 16 02
56 016 Koblenz
Tel.: 0261-287-2727
Fax: 0261-287-2735
E-Mail: stefanm@uni-koblenz.de

UNIVERSITÄT
KOBLENZ · LANDAU

Fachbereich 4: Informatik

# Aufgabenstellung für die Masterarbeit
# Katharina Lentzen
# (Mat. Nr. 209 210 058)

**Topic: Robust statistical shape modeling from routine clinical data – Evaluation techniques**

Statistical Shape Models (SSMs) are a well-established tool for 3D image analysis and especially medical image segmentation. For example, they have been successfully applied to model all major organs and bone structures as well as to all important imaging modalities like CT, MRI, ultrasound and other. Building a high quality SSM requires manually generated ground truth data from clinical experts. Unfortunately, the acquisition of such data is a time-consuming process and prone to error.

The aim of this work is to construct a SSM with reasonable quality, without the need of manual image interpretation. By using any segmentation algorithm, the idea is to exploit the multitude of daily created clinical image data. Although, the output can contain errors to a higher or lower degree, the statistics inherent in such data can be utilized.

The focus of this work is to find appropriate evaluation techniques for assessing the quality of a SSM, constructed from erroneous data. Hence, different quality measures shall be used to compare this model to a SSM build from ground truth.

Key aspects of the work:

1. Research about Statistical Shape Models and handling unreliable data
2. Data acquisition of a particular organ
3. Developing a framework for robust model building
4. Research about evaluation techniques of statistical shape models
5. Developing a framework for evaluation
6. Evaluation
7. Documentation and discussion about the results

The work is implemented in cooperation with Fraunhofer IDM@NTU in Singapore.

Supervisor: Dr. Marius Erdt (Fraunhofer IDM@NTU)

Koblenz, 10.09.2015

– Katharina Lentzen –

– Prof. Dr. Stefan Müller –

## Abstract

*Statistical Shape Models* (SSMs) are one of the most successful tools in 3D-image analysis and especially medical image segmentation. By modeling the variability of a population of training shapes, the statistical information inherent in such data are used for automatic interpretation of new images. However, building a high-quality SSM requires manually generated ground truth data from clinical experts. Unfortunately, the acquisition of such data is a time-consuming, error-prone and subjective process. Due to this effort, the majority of SSMs is often based on a limited set of this ground truth training data, which makes the models less statistically meaningful. On the other hand, image data itself is abundant in clinics from daily routine. In this work, methods for automatically constructing a reliable SSM without the need of manual image interpretation from experts are proposed. Thus, the training data is assumed to be the result of any segmentation algorithm or may originate from other sources, e.g. non-expert manual delineations. Depending on the algorithm, the output segmentations will contain errors to a higher or lower degree. In order to account for these errors, areas of low probability of being a boundary should be excluded from the training of the SSM. Therefore, the probabilities are estimated with the help of image-based approaches. By including many shape variations, the corrupted parts can be statistically reconstructed. Two approaches for reconstruction are proposed - an *Imputation method* and *Weighted Robust Principal Component Analysis* (WRPCA). This allows the inclusion of many data sets from clinical routine, covering a lot more variations of shape examples. To assess the quality of the models, which are robust against erroneous training shapes, an evaluation compares the generalization and specificity ability to a model build from ground truth data. The results show, that especially WRPCA is a powerful tool to handle corrupted parts and yields to reasonable models, which have a higher quality than the initial segmentations.

## Zusammenfassung

Statistische Formmodelle sind eine der erfolgreichsten Methoden für 3D-Bildanalysen und insbesondere für die Segmentierung von medizinischen Bilddaten geeignet. Durch die Modellierung der Abweichungen eines Organs in einem Trainingsdatensatz können die statistischen Informationen genutzt werden, um neue Bilddaten automatisch zu interpretieren. Um ein qualitativ hochwertiges statistisches Formmodell zu erstellen, werden jedoch manuell generierte Ground Truth-Daten eines Experten benötigt. Diese Datenbeschaffung ist mit einem enormen Zeitaufwand verbunden und ist außerdem fehleranfällig und subjektiv. Aus diesem Grund basieren die meisten Formmodelle auf einer begrenzten Anzahl an Trainingsdaten, welche das Modell weniger statistisch aussagekräftig machen. Andererseits sind medizinische Bilddaten in Kliniken reichlich vorhanden. In dieser Arbeit werden automatische Methoden zur Erstellung eines statistischen Formmodells ohne die manuelle Interpretation von Bilddaten eines Experten vorgestellt. Die benötigten Trainingsdaten werden als Ergebnis eines jeden Segmentierungsalgorithmus angenommen. Abhängig von der Wahl des Algorithmus, sind die Segmentierungen mit Fehlern verbunden. Diese Bereiche sollten bei der Modellbildung nicht berücksichtigt werden. Aus diesem Grund werden jedem Punkt in einer Trainingsform mittels Bild-basierten Verfahren Wahrscheinlichkeiten zugeordnet, wie sicher die Segmentierung ist. Unter Hinzunahme einer Vielzahl von Formvariationen können die fehlerhaften Daten dann statistisch rekonstruiert werden. Zwei Verfahren zur Rekonstruktion werden vorgestellt - eine sogenannte *Imputation method* und *Weighted Robust Principal Component Analysis* (WRPCA). Diese Methoden ermöglichen die Einbeziehung vieler Datensätze aus der klinischen Routine, welche zu mehr Variationen in dem statistischen Modell führen. Um die Modelle bewerten zu können, vergleicht eine Evaluation die Ergebnisse zu einem Modell aus Ground Truth-Daten. Die Ergebnisse zeigen, dass besonders WRPCA eine robuste Methode liefert, um fehlerhafte Daten zu verarbeiten und gleichzeitig eine Qualitätsverbesserung gegenüber den kaputten Eingangsdaten aufweist.

# Contents

## List of Figures

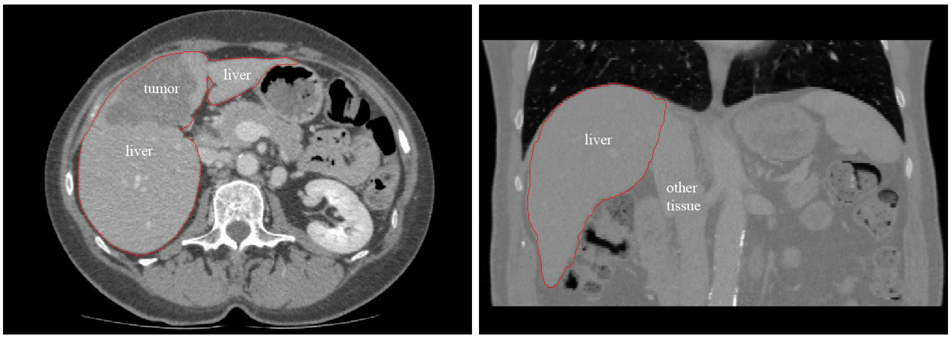
## Basic Symbols and Abbreviations

| | |
|---|---|
| $\lambda_k$ | The $k^{\text{th}}$ eigenvalue in some set of eigenvalues. |
| $\bar{\mathbf{x}}$ | The mean of a set of shapes. |
| $\hat{\mathbf{x}}$ | A generated shape of a shape model. |
| $\mathbf{S}$ | An entire shape in three dimensions. |
| $\mathbf{x}$ | A shape vector, containing $n_p$ points to describe a shape. |
| $\mathcal{D}$ | The singular value thresholding operator. |
| $\mathcal{G}(m)$ | Generalization ability for $m$ retained modes. |
| $\mathcal{L}$ | The augmented Lagrangian function. |
| $\mathcal{S}$ | The shrinkage operator. |
| $\mathcal{S}(m)$ | Specificity ability for $m$ retained modes. |
| $\mu_k$ | A positive and monotonically increasing penalty scalar. |
| $\sigma_i$ | The $i^{\text{th}}$ singular value of a matrix. |
| $b$ | A shape parameter vector. |
| $C$ | The common Covariance matrix. |
| $D$ | The observation matrix of size $3n_p \times n_S$, containing the training data of a set of shapes. |
| $E$ | A matrix formed by the eigenvectors. |
| $e_k$ | The $k^{\text{th}}$ eigenvector of a matrix. |
| $I$ | A 3D-image. |
| $L$ | A low-rank matrix. |
| $N$ | A perturbation matrix, containing small and i.i.d. Gaussian noise. |
| $n_m$ | Number of modes of variation of a model. |
| $n_p$ | Number of points describing a shape. |
| $n_S$ | Number of shapes in a data set. |
| $n_v$ | Number of voxels inside a segmentation. |
| $P_i$ | The $i^{\text{th}}$ probability for a point $p_i$. |

| | |
|---|---|
| $p_i$ | The $i^{\text{th}}$ point of a shape in Cartesian coordinates. |
| $P_{\text{D}}$ | A probability of being a boundary, based on the distance. |
| $P_{\text{GR}}$ | A probability of being a boundary, based on the gradient. |
| $P_{\text{HU}}$ | A probability of being a boundary, based on the HU-values. |
| $R$ | Rotation matrix in alignment. |
| $S$ | A sparse matrix. |
| $s$ | Scaling factor in alignment. |
| $t$ | Translation factor in alignment. |
| $u_i$ | A vector describing the spacing of the $i^{\text{th}}$ image. |
| $v_i$ | The $i^{\text{th}}$ voxel in an image. |
| $X$ | A set of $n_S$ shape vectors. |
| $Y$ | A Lagrangian multiplier. |
| ADM | Alternating Direction Method |
| ALM | Augmented Lagrange Multipliers |
| GPA | Generalized Procrustes Alignment |
| HU | Hounsfield Unit |
| MAE | Mean Absolute Error |
| PCA | Principal Component Analysis |
| PDF | Probability Density Function |
| PDM | Point Distribution Model |
| PPCA | Probabilistic Principal Component Analysis |
| RMSE | Root Mean Square Error |
| RPCA | Robust Principal Component Analysis |
| SDM | Signed Distance Map |
| SSM | Statistical Shape Model |
| SVD | Singular Value Decomposition |
| WRPCA | Weighted Robust Principal Component Analysis |

# 1 Introduction

The influence of medical image analysis is constantly increasing over the past 30 years. In healthcare, doctors find diseases earlier and thus, the appropriate treatments become more effective. However, automatic methods for image interpretation is a challenging task, still to modern times. In medicine, this interpretation is often done manually, although, the process is time-consuming and prone to error. In addition, manual interpretation can be subjective, when different clinicians have different opinions. Early work on automatic image interpretation found out, that pixel-based operations, such as edge detection and region growing, are less practicable in medical images [DTT08]. These images are often noisy and contain artifacts due to occlusion. Figure 1 demonstrates the additional challenge of classifying a tumor as part of the organ and a case where little contrast between nearby organs makes the interpretation even more difficult. Due to this, higher-level analysis is needed to separate adjacent organs. A more promising approach is to incorporate a priori knowledge about the expected structure of interest into the interpretation process.



**(a)** Transversal view

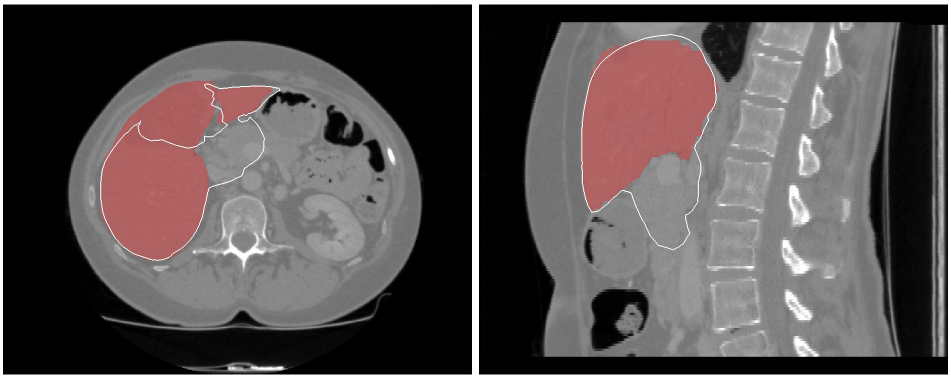
**(b)** Coronal view

**Figure 1:** A tumor in the liver is shown in **(a)** and similar tissue structures in **(b)**, which makes boundary detection difficult. The red outline shows the ground truth border extraction from an expert.

In recent years, automated operations such as model-based segmentation approaches have been appeared, where a model with information about the expected shape and appearance of a particular object is used. Often, the model is build from only a single reference shape, e.g. in industrial applications. However, due to the natural variability of human organs, a single template is not sufficient in medical images. The whole variance of a population of template shapes needs to be included. This leads to a more flexible and specific model - the *Statistical Shape Model* (SSM). The idea of using a SSM in medical image interpretation was first introduced by Cootes *et al.* [CTCG95]. By now, SSMs are one of the most successful tools in 3D image analysis and especially medical image segmentation. For example, they have been successfully applied to model all major organs, such as the liver or the heart, and to bone structures like vertebrae and pelvic bones [KW10]. All important imaging modalities like CT, MRI, ultrasound and other can be used to build a model for segmentation purposes.

Modeling the statistics of a particular organ of interest requires accurate data acquisitions, i.e. gathering training sets with enough shape variations. Manually generated ground truth data has to be provided to achieve a high-quality SSM. However, getting sufficient high-quality manually generated data is in need of clinical experts or specialist knowledge is required. Unfortunately, this process is time-consuming, error-prone and not explicit. For example, the effort to manually outline a single shape in a set of 256 CT-slice images is huge and strenuous. In addition, collecting a plenty of training shapes to cover the whole variability of a organ is important. Due to this effort, the majority of existing SSMs is often based on a limited set of this ground truth training data, which makes the models less statistically meaningful. On the other hand, image data itself is abundant in clinics from daily routine.

In this work, methods for automatically constructing a reliable SSM without the need of manual image interpretation from experts will be introduced. The training data is assumed to be the result of any segmentation algorithm or may originate from other sources, e.g. non-expert manual delineations. This allows low-quality data gained during clinical routine to be used for data acquisition. Usually such images are noisy, incomplete or include artifacts, to minimize harm of the patient. Depending on the algorithm, the output segmentations will contain errors to a higher or lower degree, as Figure 2 points out.



**(a)** Transversal view

**(b)** Sagittal view

**Figure 2:** A segmentation algorithm is used to extract the boundary of a liver in a set of CT images. The red shading visualizes the correct ground truth segmentation. The differences are clearly visible.

The idea behind this work is that the statistics inherent in such data can be exploited and the corrupted shapes can be statistically reconstructed. In contrast to use only a few ground truth samples, lots of data sets can be incorporated in the model building process and new applications benefit from more shape examples. Thus, a SSM which is robust against corrupted shape examples can be build without the need of costly manual interactions, as long as the anatomical variance is covered from the training data. Commonly, SSMs are rarely available in public, thus, an initial model building step is often required in new applications [GMS$^+$14]. This encourages the need of a robust SSM pipeline.

## 1.1 Structure of the Work

The structure of this work is divided as follows. At first, the principles of building a statistical shape model is provided in Section 2. The state of the art in Section 3 addresses two related approaches of building a SSM from low-quality data. To handle corrupted data, the points in each training shape is assigned a probability of being a boundary. Therefore, two methods are proposed in Section 4. The reconstruction of the corrupted data and how the model is build from such data is explained in Section 5. An evaluation of the proposed methods can be found in Section 6. Furthermore, the outcome of this evaluation yields to an outlook and further work in Section 7 and a final conclusion in Section 8.

# 2 Statistical Shape Models

Building a statistical shape model of a particular organ of interest usually starts with a training set of segmented images. In this work, these are the results of any existing segmentation algorithm or may are originated from other sources, e.g. manual delineations. The segmentation is transferred to a mesh representation by extracting the volume data, e.g. by using Marching Cubes algorithm [LC87]. The first step to cover the whole shape variability, a representation of the sample shapes has to be chosen.

## 2.1 Landmark-based Shape Representation

Landmark-based meshes are the simplest and most widely used representation of surface meshes, that can be found in SSM literature [HM09]. A landmark is defined as a specific point $p_i$, distributed along the surface of a shape $\mathbf{S}$:

$$p_i \in \mathbf{S}, \quad i = 1, \ldots, n_p, \tag{1}$$

where $n_p$ is the amount of landmarks used to describe $\mathbf{S}$. All $n_p$ landmark coordinates of a particular shape form the shape vector $\mathbf{x}$:

$$\mathbf{x} = (x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_{n_p}, y_{n_p}, z_{n_p})^T, \tag{2}$$

where $(x_i, y_i, z_i)$ are the Cartesian coordinates of a particular landmark point $p_i$ in $\mathbb{R}^3$. This kind of representation has been extensively used in the statistical analysis of biological shapes and is often denoted as a *Point Distribution Model* (PDM) as a synonym [CTCG92]. Gathering a sample set of $n_S$ training shapes is given by:

$$X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_S}\} \tag{3}$$

This yields to the basis input data for SSM construction. The next step is to find corresponding positions of landmarks across the set of $n_S$ examples.

## 2.2 Point Correspondences

By building statistical shape models, the landmarks in the training meshes must form a dense groupwise correspondence. That means, every mesh is represented by the same amount of landmarks and each single landmark describes approximately the same feature in all training shapes. Establishing these correspondences is one of the main challenges in the construction of a SSM. A reasonable distribution of landmarks is important, as it will affect the quality of the resulting SSM [HM09].

Manually establishing landmarks is a very time-consuming and error-prone process and is depending on expert knowledge. Identifying an appropriate feature point in 3D is also often difficult and the choice of the position from several annotators varies.

To solve the so-called *correspondence problem*, several automatic methods have been proposed. A review is given in the book of Davies *et al.* [DTT08]. All approaches basically perform a registration between the training set, where meshes can have a variable amount of points. The state-of-the-art algorithms map the shapes to a parameter space to build correspondences. By defining topological primitives in the parameter space with the same amount of points for each mesh, these points can be manipulated in an optimization algorithm and mapped back to the original space to establish corresponding shapes. Most shapes in medical imaging (e.g. liver, heart ventricles and kidney) have a genus-0[1] topology and can therefore be represented as a unit-sphere. However, these algorithms are sensitive to inconsistent parameterization. This leads to different mapping regions in the parameter space of corresponding areas in the original space. Due to this, they can fail and the convergence time can increase. [HWM06]

Kirschner et al. presented a groupwise shape parameterization algorithm in [KW10], which results in a consistent high-quality parameterization. The approach starts by choosing a mesh as a reference shape and mapping it to the unit sphere by using an area-preserving spherical parameterization. To reduce the area distortion of the parameterization, the parameters are optimized. The final parameterization is transfered to the other shapes by first aligning the actual mesh with the reference shape, defining an approximate correspondence and deducing the parameterization of the reference mesh to the considered shape. In the following correction phase, incorrect triangles like flipped or folded ones are repaired. A last refinement step reduces the local distortion.

Due to the high-quality parameterization arising of the described algorithm, the approach is used in this work to compute $n_p$ landmarks between all training shapes. Once a dense correspondence have been established, the meshes are aligned in the next section.

## 2.3 Shape Alignment

In most cases, the training meshes have different size, position and orientation, depending on the structure of interest and the used segmentation method. The property of *shape*, however, does not change under these similarity transformations. Therefore, the degrees of freedom, such as translation $t$, scaling $s$ and rotation $R$ are arbitrary factors and are non-relevant for shape variation analysis. The shape alignment steps remove the unnecessary transformations. Figure 3 visualizes this for the case of two unaligned shapes in 2D.

Considering the set of unaligned training shapes in 3D, a common coordinate frame has to be found. One of the most popular approaches to

[1] Genus-0 shapes are topologically equivalent to spheres.

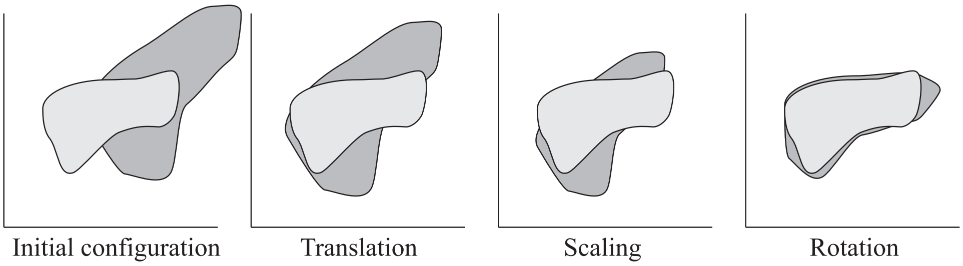


**Figure 3:** Shape alignment of two shapes in 2D.

solve the alignment problem for PDMs is the *Generalized Procrustes Alignment* (GPA) [Gow75], where a similarity transformation on a shape vector $\mathbf{x}$ is performed as:

$$\mathbf{x} \mapsto sR(\mathbf{x} - t) \tag{4}$$

The translation parameter $t \in \mathbb{R}^3$ centers all shapes around the origin. Therefore, the center of mass of each shape is computed:

$$t = \frac{1}{n_p} \sum_{i=1}^{n_p} p_i, \tag{5}$$

and subtracted from each point in Equation 4. After centering all shapes, scaling and rotation can then factored out about the origin. To eliminate the scaling factor $s \in \mathbb{R}^+$, any scale metric can be used. One example to calculate $s$ is an average unit distance scaling:

$$s = \left( \frac{1}{n_p} \sum_{i=1}^{n_p} \|p_i\| \right)^{-1}, \tag{6}$$

where $\| \cdot \|$ denotes the Euclidean norm of a point in $\mathbb{R}^3$. The inverting of the metric is necessary to fit in Equation 4. Shapes are scaled such that the average Euclidean distance of their points to the origin is $1$, i.e. $\|\mathbf{x}\| = 1$. Often, other measures can be found in literature [SG02], e.g. the *Root Mean Square Distance* (RMSD):

$$s = \left( \sqrt{\sum_{i=1}^{n_p} \|p_i\|^2} \right)^{-1} \tag{7}$$

Removing the rotational parameter is a more challenging task. First, a reference shape $\mathbf{x_{ref}}$ is selected to get an initial orientation. This can be arbitrarily chosen from the set of training shapes or computed as the mean $\bar{\mathbf{x}}$ of all $n_S$ shapes:

$$\bar{\mathbf{x}} = \frac{1}{n_S} \sum_{i=1}^{n_S} \mathbf{x}_i \tag{8}$$

Note that if the mean is chosen as the reference shape, it should be scaled as before, i.e. the average distance to the origin is 1. To find the optimal rotation matrix $R$ for a particular shape, which best fits the reference orientation, the sum of the squared distances between corresponding landmarks has to be minimized:

$$\text{minimize} \qquad \|\mathbf{x_{ref}} - \mathbf{x}\|^2 \tag{9}$$

This can be solved by first calculating the covariance matrix $C$ of the two shapes:

$$C = \sum_{i=1}^{n_p} \mathbf{x}_i \mathbf{x_{ref}}_i^T \tag{10}$$

Next, the matrix $C$ is factorized by *Singular Value Decomposition* (SVD) into the form:

$$C = U \Sigma V^T \tag{11}$$

where $U$ and $V$ are the matrices of left- and right-singular vectors and $\Sigma$ contains the positive singular values on its diagonal. Finally, the optimal $3 \times 3$ rotation matrix $R$ that fits in Equation 4 can be extracted:

$$R = V U^T \tag{12}$$

In literature, this procedure is often called the *orthogonal Procrustes problem*. [ELF97]

After optimally aligning all shapes to the reference shape in a least squares sense, the GPA algorithm computes the mean of the shapes. Again, the mean should have the same scaling as the other shapes, i.e. $\|\bar{\mathbf{x}}\| = 1$. Finally, the distance between the mean and the reference shape is computed, e.g. with the Euclidean distance. If this distance exceeds a threshold, the reference shape is replaced with the mean. In this case, the algorithm iteratively repeats the alignment of all shapes to their mutual mean and verifies the distance metric. This process is repeated a few times until convergence. Algorithm 1 summarizes the whole alignment procedure.

**Algorithm 1** Alignment procedure

```
Input: Set of unaligned shapes
Center all shapes around the origin.
Normalize all shapes.
Select a reference shape $\mathbf{x_{ref}}$.
Normalize $\|\mathbf{x_{ref}}\| = 1$
repeat
    Align all shapes to $\mathbf{x_{ref}}$
    Compute the mean shape $\bar{\mathbf{x}}$
    Normalize $\|\bar{\mathbf{x}}\| = 1$
    Distance $d = \|\mathbf{x_{ref}} - \mathbf{x}\|$
    Set $\mathbf{x_{ref}} = \bar{\mathbf{x}}$
until $d$ > threshold
Output: Set of mutually aligned shapes
```

## 2.4 Shape Space

After establishing correspondences and aligning the training data set, the shape vectors can be arranged as the $n_S$ columns of a large data matrix $D$:

$$D = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n_S} \\ y_{1,1} & y_{1,2} & \cdots & y_{1,n_S} \\ z_{1,1} & z_{1,2} & \cdots & z_{1,n_S} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n_S} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,n_S} \\ z_{2,1} & z_{2,2} & \cdots & z_{2,n_S} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_p,1} & x_{n_p,2} & \cdots & x_{n_p,n_S} \\ y_{n_p,1} & y_{n_p,2} & \cdots & y_{n_p,n_S} \\ z_{n_p,1} & z_{n_p,2} & \cdots & z_{n_p,n_S} \end{pmatrix} \in \mathbb{R}^{3n_p \times n_S} \tag{13}$$

Each row represents corresponding coordinates across the set of shapes. Considering all rows of $D$ as a unique dimension, this leads to the *Shape Space*, which is spanned by $3n_p$ dimensions. A single training shape is now represented as a point in the new space and on the other hand, a point in shape space corresponds to a physical shape. The training set of $n_S$ shape examples $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_S}\}$ forms a point cloud in this shape space (cf. Figure 4).

To model the statistical shape variability of the whole training data, the distance between points in shape space can be considered as a measurement of shape variation. A high distance of shapes in shape space means a high variation in physical space. The interpolation of points in shape space can be used to generate an arbitrary large number of new shape instances.

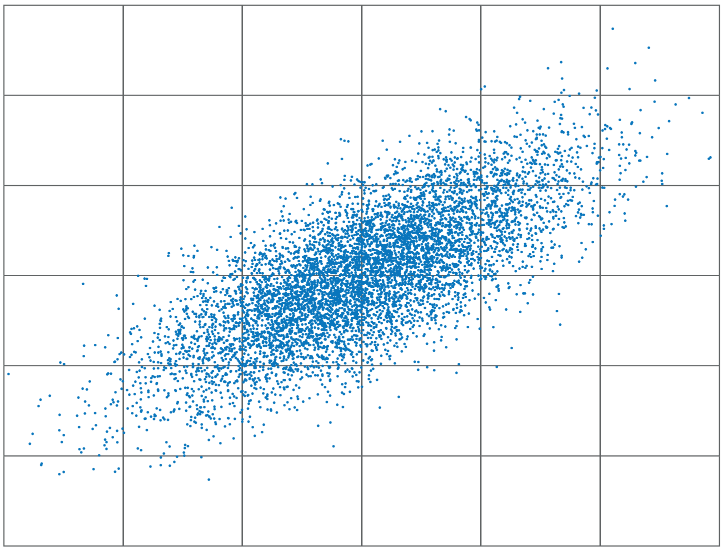


**Figure 4:** The shape space is spanned by $3n_p$ dimensions. A set of training shapes forms a point cloud in this space.

One of the main difficulties which arises with a huge amount of data, is the curse of dimensionality[2]. Consider the following problem. A training set where each shape consists of $n_p = 2500$ landmark points in $\mathbb{R}^3$, yields to a total amount of $7500$ dimensions in shape space. To alleviate this way of looking at the problem, one must reduce the dimensionality of the set of sample shapes.

### 2.5 Principal Component Analysis

*Principal Component Analysis* (PCA) is probably the most commonly used statistical tool for data analysis and dimensionality reduction [Jol02]. With the help of PCA, the most relevant information can be extracted from high-dimensional data. It seems likely, that in some dimensions only small changes across the set of training shapes occur, e.g. due to aligning the data. Thus, those dimensions can be rejected and a new set of axes, that better reflects the actual data distribution needs to be found. In other words, the assumption is, that high-dimensional data lie near a linear subspace of much lower dimensionality. PCA tries to find an estimate of this low-dimensional subspace by finding a new set of orthogonal axes, the directions where the most variance occur. These axes are called the *Principal Components* and are linear transformations of the original axes. Dimensionality reduction is performed by keeping only a subset of $k$ principal components, where $k < 3n_p$.

[2] The curse of dimensionality, termed by Richard E. Bellman, are phenomena, that arise when high-dimensional data is analyzed. For example, by adding more dimensions in a mathematical space, the volume of the space rapidly increases, making the data sparse.

Mathematically, the principal components correspond to the eigenvectors $e_k$ of the data matrix and their respective variances are the eigenvalues $\lambda_k$.

To find the new set of orthogonal axes in the shape space, that best describes the observed variation, the first step is to calculate a new origin. This origin is set to the mean shape $\bar{\mathbf{x}}$ and is calculated by averaging all $n_S$ shape vectors as before in Equation 8. The mean shape can be considered as the center of mass of all shape examples, as showed in Figure 5.

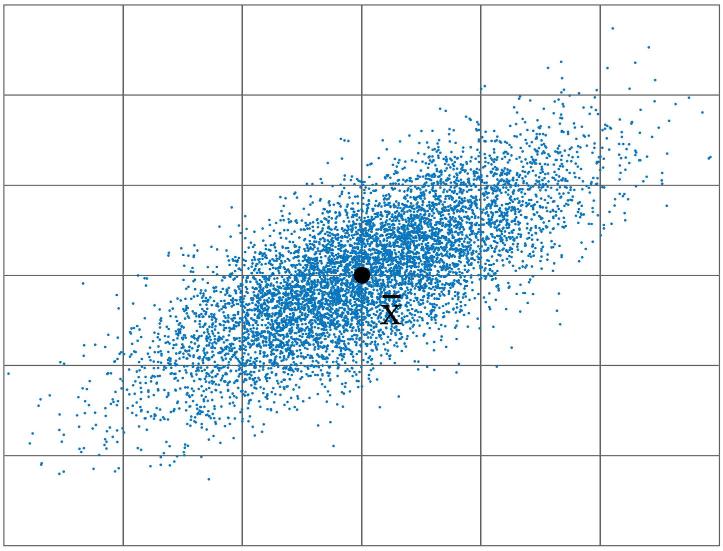


**Figure 5:** The mean shape $\bar{\mathbf{x}}$ is the center of mass of all shapes in shape space.

The next step is the calculation of eigenvectors and eigenvalues of the data set, which are considered as the mapping from shape space to the underlying low-dimensional subspace. Therefore, several methods exist. By subtracting the mean of each variable from the set, the sample covariance matrix $C$ of size $3n_p \times 3n_p$ can be computed:

$$C = \frac{1}{n_S - 1} \sum_{i=1}^{n_S} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \tag{14}$$

The set of eigenvectors of $C$ and the according eigenvalues are computed by *Eigendecomposition*, i.e. deconstructing the square matrix $C$ into the form:

$$C = Q \Lambda Q^{-1}, \tag{15}$$

where the columns of the square matrix $Q$ contains the $3n_p$ eigenvectors of $C$ and $\Lambda$ is a diagonal matrix with corresponding eigenvalues on the diagonal. [Jol02]

A more general solution to find $e_k$ and $\lambda_k$ is to perform a singular value decomposition directly on the original data. At first, all shape vectors in the aligned data matrix $D$ from Equation 13 are centered around the origin by subtracting the mean shape:

$$D = ((\mathbf{x}_1 - \bar{\mathbf{x}}), (\mathbf{x}_2 - \bar{\mathbf{x}}), \dots, (\mathbf{x}_{n_S} - \bar{\mathbf{x}})) \tag{16}$$

Next, the SVD factorizes the matrix $D$ into the form $D = U\Sigma V^T$, where the columns of $U$ and $V$ are the orthonormal eigenvectors of $DD^T$ and $D^T D$ respectively, and the rectangular diagonal matrix $\Sigma$ contains the square roots of eigenvalues from $U$ or $V$ in descending order. The SVD approach offers a higher numerical stability in contrary to the covariance method. Thus, the SVD approach is preferred in practical implementations with high dimensional data. [HM09]

However, both cases yield to a set of at most $k \leq n_S - 1$ non-zero eigenvalues $\lambda_k$. The corresponding eigenvectors $e_k$ form the new axes, that better reflect the original data. These axes are linearly independent and are called the principal components. This means, that dimensionality reduction was performed, since $k < 3n_p$. [DTT08]

The set of non-zero eigenvalues are ordered by descending size:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k, \tag{17}$$

where the largest possible variance corresponds to the first principal component. Figure 6 visualizes the first two principal components in the shape space.

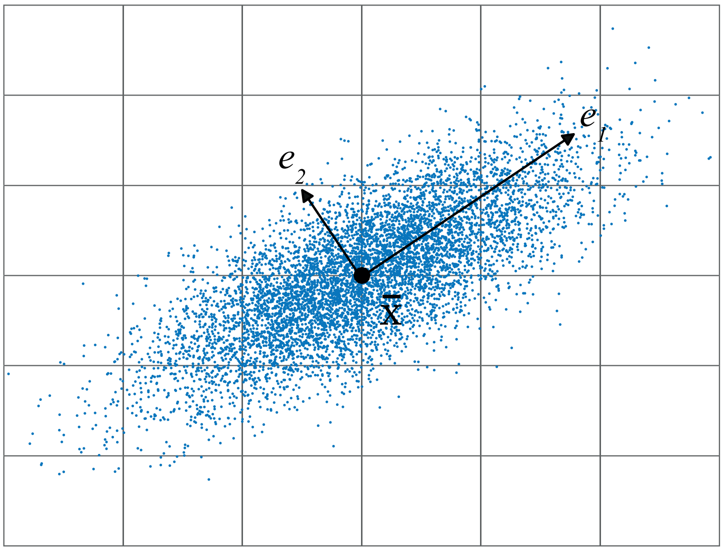


**Figure 6:** The eigenvector $e_1$ that correspond to the largest eigenvalue $\lambda_1$, represents the direction where the most variance occur. The second eigenvector $e_2$ is perpendicular to the first one. These axes span a new low-dimensional vector space.

The whole variance of the data can be covered with the sum of all nonzero eigenvalues:

$$\sum_{i=1}^{k} \lambda_i \tag{18}$$

In practice, only a certain portion of the total variance is retained, such that the dimensionality can be further reduced:

$$\sum_{i=1}^{n_m} \lambda_i, \tag{19}$$

where $n_m < k$. Usually, the smallest dimension $n_m$ is chosen, such that $90\% - 99\%$ of the total variance of the training data is captured [HM09]. The residual terms can be considered as noise. By keeping the most relevant principal modes of variation, new shape instances $\hat{\mathbf{x}}$ can be generated by a linear combination of those $n_m$ modes:

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{i=1}^{n_m} b_i e_i, \tag{20}$$

where $b \in \mathbb{R}^{n_m}$ defines a set of shape parameters, which should be restricted to a certain interval, allowing only plausible shapes to be reconstructed. Usually, $b_i$ is chosen to lie inside $\left[-3\sqrt{\lambda_i}, 3\sqrt{\lambda_i}\right]$. This limitation ensures, that a generated shape in Equation 20 is similar to those from the original training set. The vector $b$ is therefore used to vary a shape, making the model deformable.

A statistical shape model is fully described by these $n_m$ retained eigenvectors and eigenvalues. Figure 7 visualizes the largest two modes of variation for the example of a liver SSM. Notice that PCA results in global modes, this means, each mode will have an impact on every landmark point. [HM09]

Defining the retained eigenvectors $(e_1, e_2, \ldots, e_{n_m})$ as columns of a matrix $E$ of size $3n_p \times n_m$, Equation 20 can be rewritten:

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + Eb, \tag{21}$$

Then, the matrix $E$ performs the mapping from the low-dimensional subspace to the shape space. On the other hand, the transposed matrix $E^T$ is used for the backwards mapping, i.e. the back projection of a shape vector $\mathbf{x}$ to the low-dimensional subspace is defined as:

$$\mathbf{x} \mapsto E^T(\mathbf{x} - \bar{\mathbf{x}}) \tag{22}$$

Figure 8 illustrates the mapping from the shape space to the underlying subspace and vice versa.

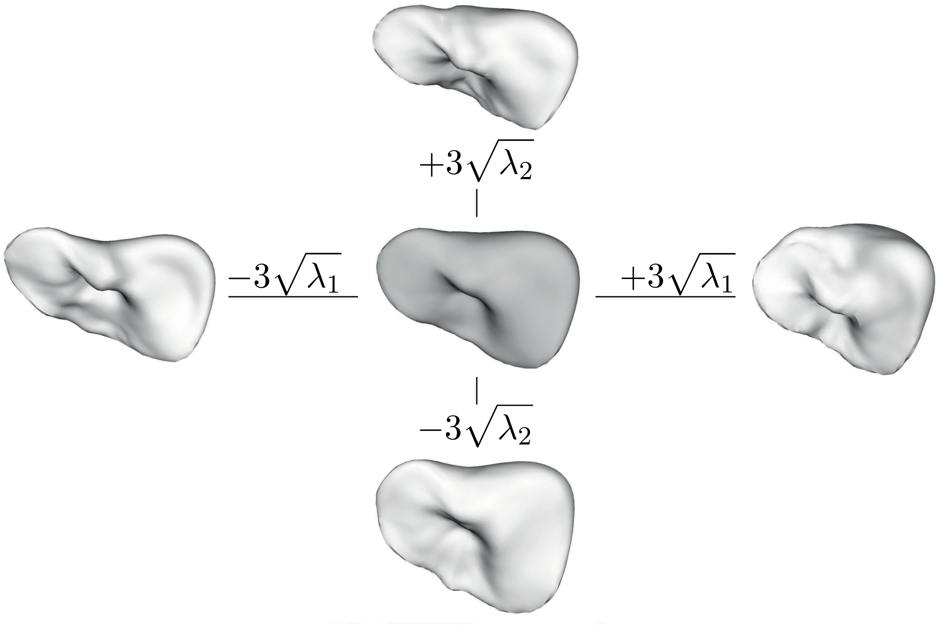


**Figure 7:** The first two principal modes of variation of a liver SSM. The mean shape is pictured in the middle. Retaining only the largest mode is shown on the horizontal axis and the second largest mode on the vertical axis.

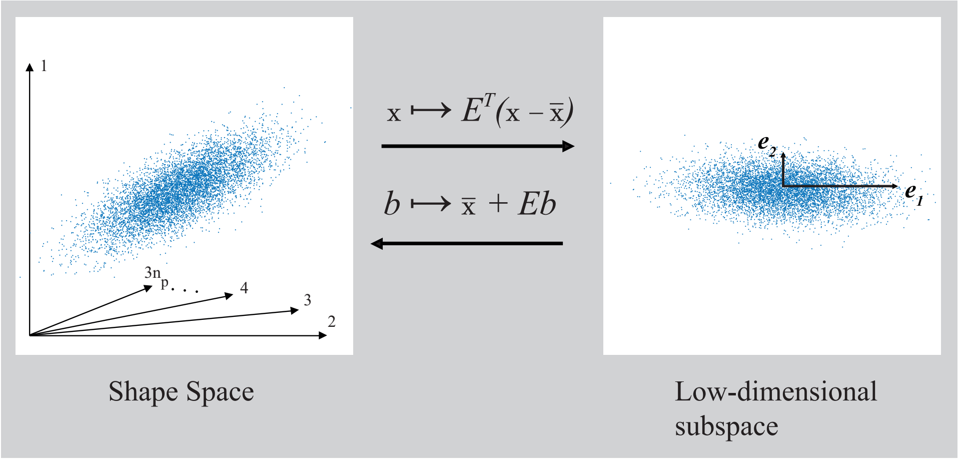

1
$3n_p$ . . .
4
3
2
$\mathrm{x} \longmapsto E^T(\mathrm{x} - \bar{\mathrm{x}})$
$b \longmapsto \bar{\mathrm{x}} + Eb$
$e_2$
$e_1$
Shape Space
Low-dimensional subspace

**Figure 8:** A shape vector **x** in the shape space is mapped to the underlying low-dimensional subspace by subtracting the mean shape $\bar{\mathrm{x}}$ and multiplying the transposed eigenvector matrix $E^T$. This is called the back projection. On the other hand, new shape instances can be generated with the vector $b$, by defining a set of shape parameters.

It has to be mentioned, due to the different dimensionality of spaces, the mappings cannot be the inverse to each other. This means, a reconstruction of an original training shape vector in shape space $\mathbf{x}_i \approx \bar{\mathbf{x}} + Eb$, is only an approximation, since the available number of principal modes is $n_m < 3n_p$.

To summarize, PCA is used to build a statistical shape model by modeling the distribution of a training set of shapes. The most relevant principal modes of variation are retained to reduce the dimensionality in the shape space. Eigenvectors and their corresponding eigenvalues define the new axes, which are centered around the mean shape of the training data. Thus, a SSM can be used to evaluate new shapes, in reference to the similarity to the training set. In addition, a SSM can be used to generate new shape instances inside the distribution by a linear combination. In a next step, this distribution is analyzed.

## 2.6 Model Distribution

Since $\bar{\mathbf{x}}$ and $E$ are fixed after applying PCA, a generated shape using Equation 21 is explicitly defined by the parameter vector $b$. Suppose, the parameter vectors $\{b_1, \ldots, b_{n_S}\}$, that describe the set of $n_S$ training shapes in the low-dimensional subspace, follow a certain distribution $\mathrm{p}(b)$. This distribution can be considered as a *Probability Density Function* (PDF). Sampling the parameter vectors with $\mathrm{p}(b)$ can be used to represent new shape examples, that follow the same distribution as the original training set. If the shapes in the shape space form a unimodal distribution, PCA finds the new low-dimensional coordinate system, centered in the distribution and spanned by the most relevant principal modes of variation. A simple unimodal distribution is the Gaussian distribution. This PDF forms an ellipsoid in the remaining $n_m$ dimensions. Every interpolation between training shapes models valid shapes. In contrast, if the distribution in the shape space is multimodal or rather non-linear, PCA can be used to reduce the dimensionality, but the axes of the coordinate system do not necessarily correspond to the largest variances of the training data. Therefore, constraining the parameter vector $b$ as before will possibly yield to invalid shapes. In this case, other modeling techniques, e.g. kernel methods have to be used. [DTT08] However, this work is restricted to Gaussian distributed input data.

## 2.7 Problems with PCA

Despite the attractive feature of reducing the dimensionality of a data matrix and projecting the data onto a set of lower dimensional vectors, PCA has several shortcomings. To fit the principal components to the set of points in the shape space, PCA operates in an least square sense. Suppose PCA as the deconstruction of $D$ into the sum of a low-rank matrix $L$ and a small perturbation or noise matrix $N$:

$$D = L + N, \tag{23}$$

where $L$ has sufficiently smaller rank than $D$. The assumption holds that the entries of $N$ are small and independent and identically distributed (i.i.d) Gaussian random variables. Performing dimensionality reduction is done by following least squares constrained optimization:

$$\begin{aligned} &\text{minimize} && \|D - L\|_2 \\ &\text{subject to} && \text{rank}(L) \leq r, \quad D = L + N, \end{aligned} \tag{24}$$

where $\|\cdot\|_2$ is the spectral norm of a matrix, that is, the largest singular value $\lambda_{max}$ and $r \ll \min(3n_p, n_S)$ is the target rank of the underlying subspace. Here, PCA seeks the optimal solution of $L$ in an $\ell^2$ sense. In Section 2.5 for example, this problem can be solved by the SVD of $D$.

The most serious problem with least square minimization is the non-robustness to outliers. If the data matrix $D$ contains large corruptions, PCA yields to arbitrarily false results of the estimated low-rank matrix $L$. In particular, a single corrupted entry in the data matrix can have a strong influence on $L$, leading to an overfitting of this error. Figure 9 illustrates the result of PCA with only one extremely corrupted shape example.

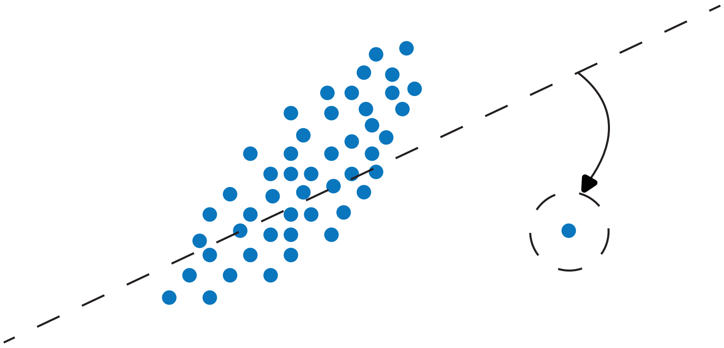


**Figure 9:** A single outlier can lead to an extremely bad estimation of the underlying subspace. The dashed line shows the first principal component.

If the amount of data points explaining large outliers in the training set increases, some modes of variation will focus on representing these corrupted parts. Therefore, it is necessary to detect and handle such corrupted data in a pre-processing step before applying PCA to the training data. [CLMW11]

# 3 State of the Art

Robust statistical shape modeling based on incomplete or corrupted data is a rarely covered topic in the literature. Most attention is given to solve the correspondence problem properly [HM09], as this is directly associated with the quality of the model. Robust modifications are usually applied to the model-based segmentation task, i.e. to the search step after model building [ANJY06]. In the first place, corrupted data is avoided and usually manually segmented ground truth data is used.

The first work to address the problem of using training shapes from *lousy* data to build a SSM is the approach of Lüthi *et al.* [LAV09]. In their approach, they choose a reference shape which is free of outliers, i.e. a manually segmented ground truth shape. This reference is divided into arbitrary parts, preferably anatomically significant patches. Later, each part is tested individually if it is used for model building. Lüthi *et al.* used a non-rigid registration algorithm to create a vector field among all training shapes. Thus, correspondence is established, where each shape can be represented as a warp of the reference with the vector field. If a shape contains outliers, this transformation will cause unnatural deformations. Lüthi *et al.* statistically identify the outlier patches by using a algorithm called *PCOut*. PCOut basically rescales the data with robust estimators, e.g. the median, and performs PCA. Additionally, each part of a shape is assigned a probability, depending on how well this patch fits into the PCA model. Those parts whose probability is below a user-specified threshold are removed before the model is build. One method to deal with such incomplete data sets to build a SSM is *Probabilistic PCA* (PPCA) [TB99].

One drawback of this approach from Lüthi *et al.* is that a single corrupted landmark can cause the rejection of a whole patch. This would mean a loss of statistical information. Furthermore, the method is depending on the size and position of the subdivided parts. This can lead to artifacts in the reconstruction step and on the variation modes of patch boundaries [GMS$^+$14]. In addition, they need a reference shape, which is free of arbitrarily corrupted data, i.e. a manually segmented shape. Finally, PPCA is designed to handle missing data, but automatic segmentation algorithms results typically in an appearance of more ore less corrupted points, rather than missing points.

The second work to mention is the method of Gutierrez *et al.* [GMS$^+$14]. They are the first addressing the problem of building a SSM from both, outlier and incomplete training data. In contrast to use robust statistical methods, their work relies on advances in sparse optimization in recent years. Using a modern technique called *Robust PCA* (RPCA), the shape data matrix is modeled as the addition of a low-rank matrix and a sparse matrix. The assumption is, that the low-rank matrix can be recovered, leaving the corrupted data points in the sparse matrix. The reconstructed low-rank matrix can be used with standard PCA, assuming that the data is free of outliers. They achieve significantly better results in terms of robustness to missing data in comparison to Lüthi *et al.* and standard PCA.

The method of Gutierrez *et al.* is the first work to address the problem of building a SSM, which is robust against outliers and missing data. However, the drawback of this method is, that non-outlier high frequency information may get lost. That is the result of using RPCA to recover the outlier-free data. Furthermore, in their work, a reference shape is needed to establish correspondences. This reference image is assumed to contain the boundary of the organ completely, i.e. a manually segmented shape which can induce bias towards this reference.

However, the approach from Gutierrez *et al.* is closely related to the approach presented in this work and the idea of Robust PCA is introduced in Section 5.2. To avoid rejecting non-outlier high-frequency information, the affected points should be declared as safe. Therefore, the training set of shapes is analyzed and each landmark is assigned a probability of being a true boundary segmentation, as inspired by the approach of Lüthi *et al.*

# 4 Outlier Detection by Boundary Probabilities

Depending on the data acquired during clinical routine, the structure of interest will contain errors to a higher or lower degree. Accepting different automatic segmentation algorithms, this can lead to more or less corrupted output meshes. In order to account for these errors, areas of low probability of being a boundary should be excluded from the training of a statistical shape model. Knowing the probability of a landmark to be an outlier, allows for robustifying the model building step. Therefore, two separate image-based measures are used to assign each landmark point $p_i$ a boundary probability $P_i$ - the *Hounsfield Unit* (HU) and the *Gradient Magnitude*.

Since the output of the segmentation algorithm is producing a mesh, the mapping to the original image has to be found. If we define the origin point of a composed 3D-image $I$ at zero in all three axes, the corresponding voxel $v_i$ for the $i^{\text{th}}$ landmark point is calculated by a 3D-index $k_i$:

$$v_i = I[k_i], \quad k_i = \frac{p_i}{u}, \tag{25}$$

where $p_i$ is the $i^{\text{th}}$ landmark from a mesh and $u$ contains the spacings in each direction of the 3D-image. These $n_p$ voxels are used in the following to assign boundary probabilities for $n_p$ landmarks.

## 4.1 Hounsfield Unit

In CT imaging, the Hounsfield Unit is the normalized attenuation of X-ray radiation in tissue. The radiodensity of distilled water is defined as 0 HU and air as -1000 HU (see Table 1). Each voxel in the data set is assigned a value in the range of -1000 (Air) and 1000 (compact bone). To calculate

| Type | HU |
|---|---|
| Compact bone | $1000$ |
| **Liver** | **40** to **60** |
| White matter | $\sim$ 20 to 30 |
| Grey matter | $\sim$ 37 to 45 |
| Blood | $40$ |
| Muscle | $10$ to $40$ |
| Kidney | $30$ |
| Cerebrospinal fluid | $15$ |
| Water | $0$ |
| Fat | $-50$ to $-100$ |
| Soft Tissue | $-100$ to $-300$ |
| Air | $-1000$ |

**Table 1:** Overview of the common differentiation of HU-values.

probabilities for each landmark based on HU-values, first, the global mean HU-value $h_{mean}$ of the segmented organ is computed. For example, in the case of the liver as a target organ, all $n_v$ voxels inside the segmentation are considered and tested if they correspond to liver values from Table 1:

$$h_{mean} = \frac{1}{M}\sum_{i=1}^{n_v}\psi(v_i)HU(v_i), \quad \psi(v) = \begin{cases} 1 & \text{if } HU(v) \in [40, 60] \\ 0 & \text{otherwise} \end{cases}, \tag{26}$$

where $HU(v)$ returns the HU-value of the considered voxel and $M$ is the number of voxels where $\psi(v)$ is 1. Only liver HU-values between 40 and 60 are taken into account to calculate the global mean[3]. In the next step, a box $B$ of size $b \times b \times b$ is sampled around each voxel, specified by the precomputed index. All voxels inside the segmented region of the box are considered and a local mean HU-value inside the box is calculated:

$$h_{local} = \frac{1}{W}\sum_{i=1}^{B}\Gamma(v_i)HU(v_i), \quad \Gamma(v) = \begin{cases} 1 & \text{if } HU(v) \in [\text{-}1000, 1000] \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

The denominator $W$ is the number of voxels where $\Gamma(v)$ is 1. The difference of $h_{local}$ and $h_{mean}$ is illustrated in Figure 10.

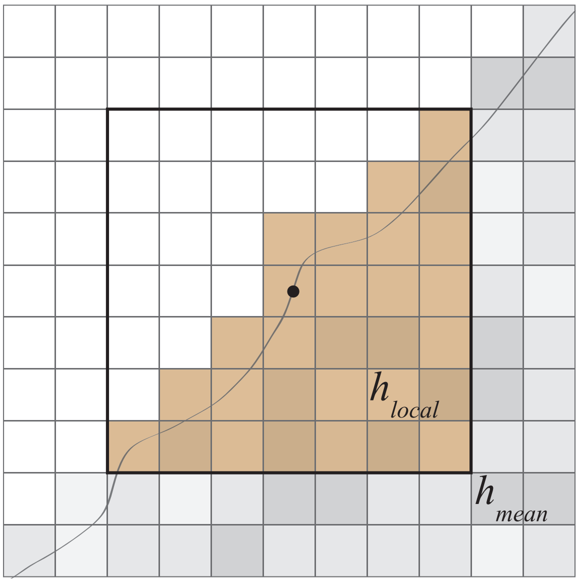


**Figure 10:** To calculate $h_{local}$, a box is sampled around a landmark. All HU-values inside the segmentation are used to compute the mean within the box. For better visualization, the Figure shows the two-dimensional case with a box size of 7 × 7.

[3] If contrast medium is used, the values should be adapted.

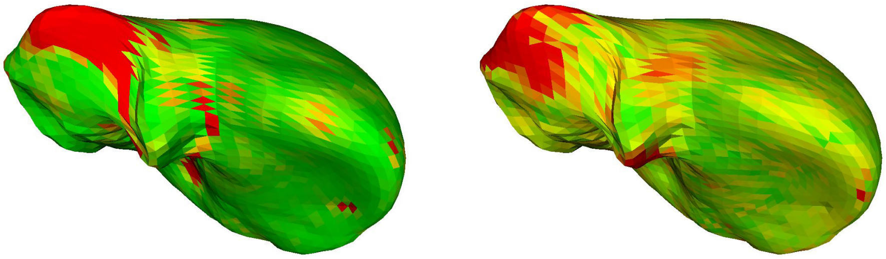
To assess a probability, the Euclidean distance $E_{\mathrm{HU}}$ between the global mean $h_{mean}$ and the local mean $h_{local}$ serves as a measure:

$$E_{\mathrm{HU}} = \|h_{mean} - h_{local}\| = |h_{mean} - h_{local}| \tag{28}$$

The HU probability for a landmark is formed with a threshold $T$, to penalize high distances:

$$P_{\mathrm{HU}} = \frac{T - E_{\mathrm{HU}}}{T} \tag{29}$$

Figure 11 compares a liver mesh, colored with the HU probabilities and colored with the optimal probabilities, computed by using the distance to the ground truth. The optimal probability distance function is described later in Section 4.4



**(a)** HU probabilities

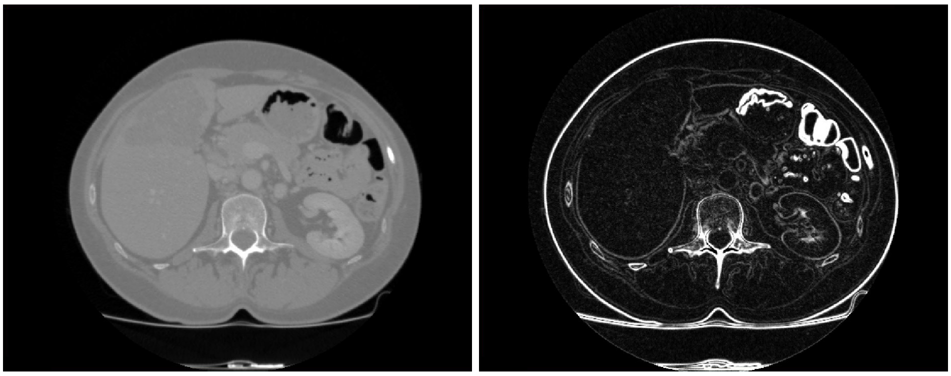
**(b)** Optimal probabilities

**Figure 11:** A liver mesh is colored with **(a)** the HU probabilities and with **(b)** the optimal probabilities. The coloring encodes good probabilities (green) and bad probabilities (red) of being a boundary.

## 4.2 Gradient Magnitude

Another measure to define a boundary probability is the image gradient. In a preprocessing step, the original 3D-image is filtered with a gradient magnitude filter, to strengthen the contour of the organ and to separate homogeneous regions [JMIC15]. Figure 12 visualizes the result of the filter. Next, the segmented image region is converted to a binary image by defining all entries inside the segmentation as 1, and 0 for outside respectively. With the binary image as input, a *Signed Distance Map* (SDM) is computed with the algorithm from Maurer *et al.* [MQR03]. The result of a SDM is another image with the Euclidean distance of each voxel to the boundary of the segmentation. The boundary is considered as the zero level line of the SDM. The inside has negative distance values and outside positive distances.

With the use of the precomputed index, a box $B_1$ of size $b_1 \times b_1 \times b_1$ is centered around a considered landmark. By using the SDM, all boundary

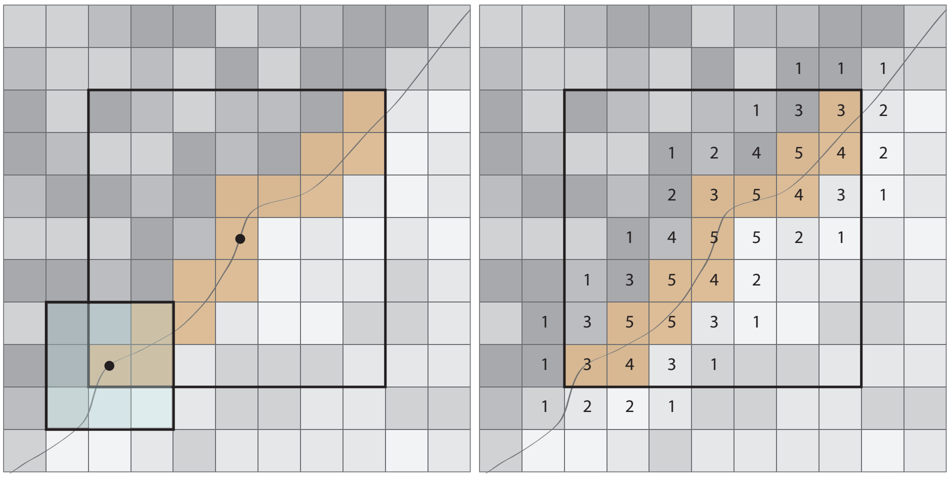


**(a)** Before

**(b)** After

**Figure 12:** The input image **(a)** and the effect of a gradient magnitude filter **(b)** is pictured for a single slice.

voxels are found inside the box. A second box $B_2$ of size $b_2 \times b_2 \times b_2$ is used to calculate the local gradient magnitude mean of all boundary voxels. The two boxes are shown in Figure 13a. By including the neighbors of a supposed boundary voxel with the second box, a better sampling of the gradient is obtained (cf. Figure 13b). Mathematically, this is given by:



**(a)** Boxes $B_1$ and $B_2$

**(b)** Implicit weighting of voxels

**Figure 13:** Two boxes $B_1$ and $B_2$ are used to compute the gradient probability **(a)**. $B_1$ is centered around a landmark point and $B_2$ is centered around each boundary voxel inside $B_1$. This leads to an implicit weighting of voxels **(b)**, where voxels close to or at the boundary are sampled more often.

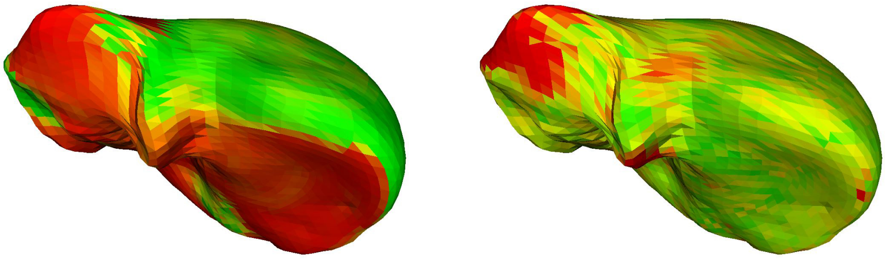
$$g_{local} = \frac{1}{Q} \sum_{i=1}^{B_1} \left( \Phi(v_i) \sum_{j=1}^{B_2} g(v_j) \right), \quad \Phi(v) = \begin{cases} 1 & \text{if } SDM(v) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

The factor $Q$ is the number of voxels where $\Phi$ is $1$, multiplied by the size of $B_2$. The function $g(v)$ returns the gradient magnitude of $v$ and $SDM(v)$ returns the value in the SDM at the voxel $v$. Due to this oversampling of $B_2$, an implicit weighting is gained.

To form the gradient probability $P_{\text{GR}}$, the local gradient $g_{local}$ is mapped to $[0, 1]$:

$$P_{\text{GR}} = \frac{g_{local} - g_{min}}{g_{max} - g_{min}}, \tag{31}$$

where the factor $g_{min}$ is the global minimum of the gradient magnitude of all investigated landmarks and $g_{max}$ the maximum respectively. With Equation 31, only local gradients which equals $g_{max}$ can reach a probability of 1. To attenuate this condition, $g_{max}$ can be set empirically and values $P_{\text{GR}} > 1$ are clamped to 1. Figure 14 compares the liver mesh, colored with the gradient probabilities and the optimal probabilities.



**(a)** Gradient probabilities

**(b)** Optimal probabilities

**Figure 14:** A liver mesh is colored with **(a)** the gradient probabilities and with **(b)** the optimal probabilities. The coloring encodes good probabilities (green) and bad probabilities (red) of being a boundary.

### 4.3 Combined Probability

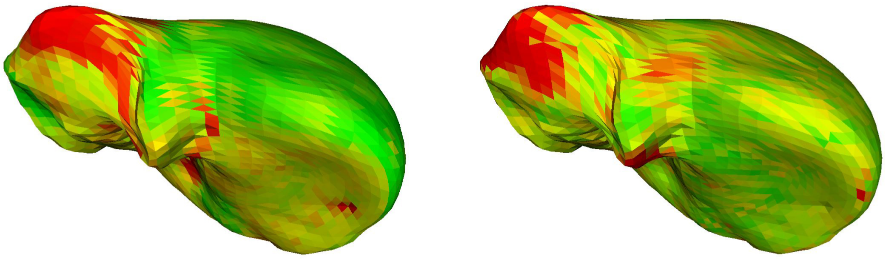
The combined probability $P_i$ for the $i^{\text{th}}$ landmark is the combination of Equation 29 and Equation 31 with an additional weighting parameter $\alpha$, to balance the influence of $P_{\text{GR}}$ and $P_{\text{HU}}$:

$$P_i = (1 - \alpha) \cdot P_{\text{GR}_i} + \alpha \cdot P_{\text{HU}_i} \tag{32}$$

Additionally, the boundary probability is weighted with a S-curve, to penalize bad results and strengthen good probabilities:

$$P_i = \frac{\sin((P_i - 0.5) \cdot \pi) + 1}{2} \tag{33}$$

Figure 15 visualizes the final probabilities with an equally weighting of $P_{\text{GR}}$ and $P_{\text{HU}}$, i.e. where $\alpha$ = 0,5. An evaluation of the estimated boundary probabilities can be found in Section 6.2, where the deviation to the optimal probabilities based on the ground truth is computed.



**(a)** Combined probabilities

**(b)** Optimal probabilities

**Figure 15:** A liver mesh is colored with **(a)** the combined HU and gradient probabilities and with **(b)** the optimal probabilities. In addition, the estimated probabilities are weighted with a S-curve. The coloring encodes good probabilities (green) and bad probabilities (red) of being a boundary.

## 4.4 Optimal Boundary Probabilities

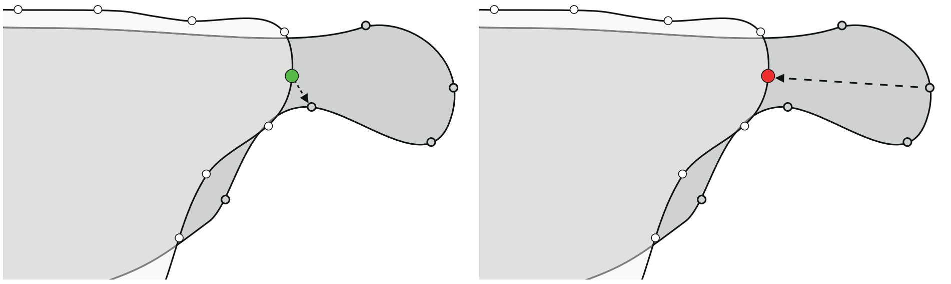
To test whether the probabilities computed from the HU-values and the gradient are a reasonable guess, the true distance of each landmark point to the ground truth is represented as probabilities as well. High distance points should be assigned a low probability and low distances should get a high probability. Hence, for a particular point $p$ of a training shape $\mathbf{x_t}$, the distance to the nearest point $q$ in the corresponding ground truth shape $\mathbf{x_{gt}}$ is calculated. Generally, the minimum distance $d_{\text{min}}$ is given by:

$$d_{\text{min}} = \delta(p, \mathbf{x_{gt}}) = \min_{i \in \{1,\dots,n_p\}} \|p - q_i\| \tag{34}$$

A probability can be computed by defining a threshold $T_{\text{D}}$ to describe the maximal allowed extent of the distance. Distances exceeding the threshold are clamped to $T_{\text{D}}$, such that $d_{\text{min}} \in [0, T_{\text{D}}]$. To get values in the range $[0, 1]$, the distances are divided by $T_{\text{D}}$. To form a probability such that small distances are correspond to higher probabilities, the distance must be inverted:

$$P_{\text{D}} = 1 - \left(\frac{d_{\text{min}}}{T_{\text{D}}}\right) \tag{35}$$

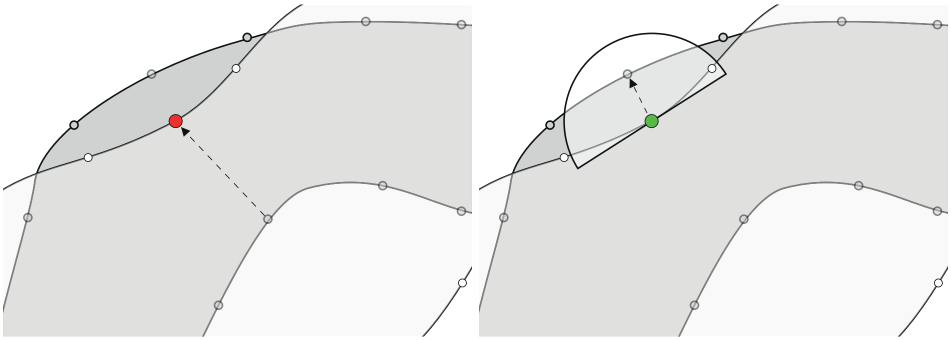
This representation of the probabilities has one drawback. A special case is visualized in Figure 16a, where a ground truth shape contains a bulge. To overcome the problem of false probability assignment in this particular case, Equation 34 is additionally performed for each ground truth



**(a)** Bad probability assignment **(b)** Better probability assignment

**Figure 16:** If the ground truth shape contains a bulge, the probabilities based on the nearest distance can lead to bad estimates **(a)**. To overcome this problem, one should additionally iterate over the ground truth shape **(b)**.

point. Each point, found in the second iteration, is tested if the precomputed distance differs to the new computed distance, as seen in Figure 16. If this is the case, the maximum of both distances is chosen to be the true distance. Thus, in Figure 16b, the point is assigned a bad probability. However, another special case appears with this approach. Consider a ground truth shape with a deep concave part (cf. Figure 17a), where a bad probability is misleadingly assigned.

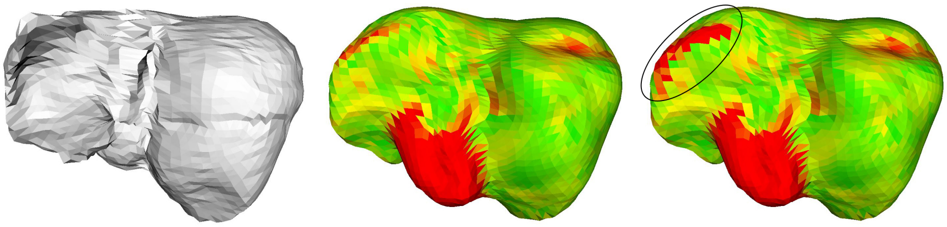


**(a)** Bad probability assignment **(b)** Better probability assignment

**Figure 17:** In strong concave regions, the probability assignment can fail **(a)**. To overcome this problem, the distance from a point on the ground truth to the training shape is only replaced, if the point lies in the front hemisphere of the training shape **(b)**.

To avoid wrong probability assignment in this case, the distance of the ground truth to a particular point $p_i \in \mathbf{x_t}$ is only taken into account, if the ground truth point $q$ lies in the front hemisphere of $p_i$. This can be verified by the dot product of the normal[4] of $p_i$ and the normalized vector between $p_i$ and $q \in \mathbf{x_{gt}}$. A positive dot product corresponds to points in the front hemisphere and therefore the point in Figure 17b is assigned a reasonable probability.

This approach is more robust than simply calculating the closest point to the ground truth, as before in Equation 34. Finally, each point in $\mathbf{x_t}$ is assigned a probability by using Equation 35. Figure 18 compares the improvement of this approach to simply applying Equation 34 only. The procedure is summarized in Algorithm 2.



**(a)** Ground truth **(b)** Training mesh **(c)** Training mesh

**Figure 18:** Visualization of the optimal probabilities of a training mesh, based on the distance to the corresponding ground truth shape **(a)**. The nearest distance is computed directed from the training mesh to the ground truth **(b)**, and from both directions with additional front hemisphere checking **(c)**. Notice, that the described method yields slightly better estimated probabilities in the marked area.

[4] It has to be mentioned, that a point does not have a normal. However, a normal at a particular point can be interpolated from its adjacent faces.

**Algorithm 2** Computing optimal probabilities with ground truth

```
**Input:** $\mathbf{x_t}$, $\mathbf{x_{gt}}, T_D$
Initialize distance array $d_{\mathbf{t}}[n_p]$
Initialize index $i = 0$
Initialize minimal distance $d_{\text{min}} = 0$
for all $p \in \mathbf{x_t}$ do
    Find point $q$ with minimum distance in $\mathbf{x_{gt}}$
    $d_{\text{min}} = \|p - q\|$
    $d_{\mathbf{t}}[i] = d_{\text{min}}$
    $i = i + 1$
end for
for all $q \in \mathbf{x_{gt}}$ do
    Find point $p$ with minimum distance in $\mathbf{x_t}$
    $d_{\text{min}} = \|q - p\|$
    $i =$ getIndex$(p)$
    if $d_{\mathbf{t}}[i] < d_{\text{min}}$ then
        if $(getNormal(p) \cdot \vec{pq}) > 0$ then
            $d_{\mathbf{t}}[i] = d_{\text{min}}$
        end if
    end if
end for
for all $d \in d_{\mathbf{t}}$ do
    Clamp $d$ to $[0, T_D]$
    Normalize distance: $d = d \cdot T_D^{-1}$
    Invert distance: $d = 1 - d$
end for
Output: Optimal probabilities $d_{\mathbf{t}}$
```

# 5 Handling Corrupted Training Data

Handling outliers and missing entries in high-dimensional data is a well known problem in the field of statistical investigation [IR10]. For example, *Imputation methods* exist, where each uncertain point is replaced with a reasonable guess, e.g. the mean. These methods carry out the analysis as no corrupted data is existent. Thus, an imputation method is proposed in the following subsection to sort out outliers, by using the estimated boundary probabilities, and to create a statistical shape model.

## 5.1 Imputation Approach

The construction of a statistical shape model starts with a set of segmented images. Surface meshes with an arbitrary amount of points are extracted from the volume data, e.g. by Marching Cubes algorithm [LC87]. The first critical step to build the model is to find corresponding landmarks throughout the data set. A state-of-the-art groupwise consistent shape parameterization [KW10] is used to generate shapes with $n_p$ corresponding landmarks. Deformations caused by corrupted data should be replaced with the information of corresponding points. Thus, to estimate such outliers, each landmark point $p_i$ is assigned a probability $P_i$ of being a boundary, as described before in Section 4. After aligning the shapes with the Procrustes method (see Section 2.3), the landmark coordinates with a bad result from the probabilities are substituted with reasonable points from the remaining data. A brute force approach is to take the mean of all high probability points. With the probabilities, a weighted mean of landmark points is computed as visualized in Figure 19.

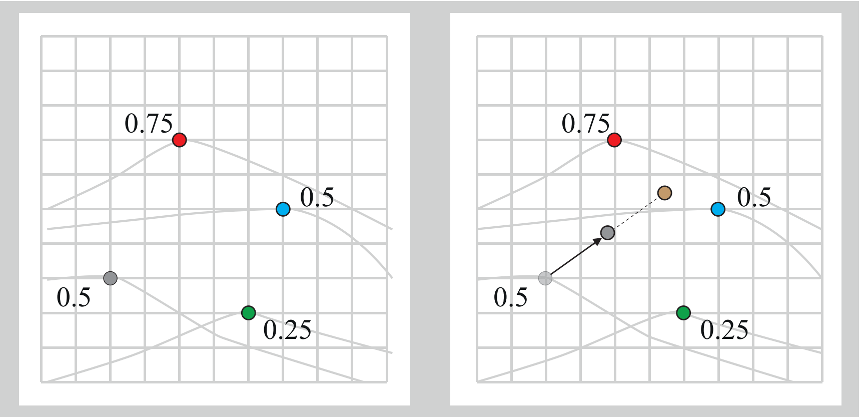


**Figure 19:** The weighted shifting of landmarks in 2D. The point in the bottom left corner (gray) has a probability of $0.5$. Thus, the transformation to the mean is only affected half. The mean itself is calculated from the weighted sum of all corresponding landmarks.

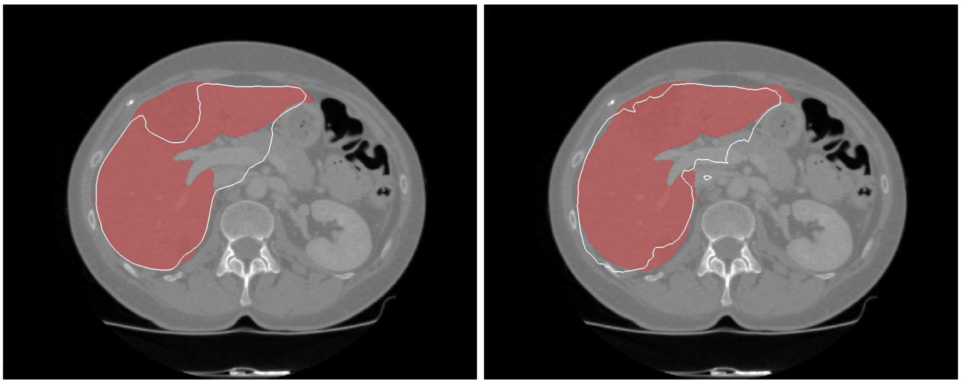
Mathematically, for the $i^{\text{th}}$ point in the $j^{\text{th}}$ shape vector, the weighted mean $\bar{p}_{ij}$ of all corresponding points in the other shapes is given by:

$$\bar{p}_{ij} = \frac{1}{\sum\limits_{k=1,k\neq j}^{n_S} P_{ik}} \cdot \sum_{k=1,k\neq j}^{n_S} \left(P_{ik} \cdot p_{ik}\right), \tag{36}$$

where $P_{ik}$ is the probability of the $i^{\text{th}}$ point in the $k^{\text{th}}$ shape. Thus, points with low probability have less influence on the mean, than points with a high probability. The point $p_{ij}$ is then shifted towards the mean $\bar{p}_{ij}$. However, if the own probability of the omitted point is already reflecting a good segmentation result, the point should stay unchanged. In other words, the shifting to $\bar{p}_{ij}$ is weighted with its own probability:

$$p'_{ij} = p_{ij} + (1 - P_{ij}) \cdot (\bar{p}_{ij} - p_{ij}) \tag{37}$$

The probability $P_{ij}$ has to be inverted in Equation 37, because a high probability point from the segmentation is already a good result and the influence of shifting $p_{ij}$ to $\bar{p}_{ij}$ is low. Thus, a single point is completely replaced by the weighted mean, if its probability is 0. This procedure is performed for every shape, where every corrupted landmark is shifted with respect to its probability towards the corresponding mean. Figure 20 compares the difference between a transversal slice of a mesh before and after the weighted shifting procedure, with the estimated boundary probabilities from Section 4.3, is applied.



**(a)** Before

**(b)** After

**Figure 20:** The effect of the imputation method **(a)** before and **(b)** after the weighted shifting approach.

Due to the fact that the probabilities are estimates, the modified meshes can contain some spikes. Smoothing these parts could be a possibility to overcome problems in the PCA model building, where some modes could focus on explaining these spikes. In order to create a reasonable shape model, the statistics inherent in the data are used to smooth out the meshes. First, a statistical shape model is build in an iterative leave-one-out fashion. By excluding one mesh in each iteration, the statistics are captured with $n_S - 1$ training shapes by PCA. Therefore, the eigenvectors and eigenvalues are computed as described in Section 2.5. In the next step, the omitted shape is projected back onto the low-dimensional subspace, spanned by the $n_m$ retained principal components of the SSM (cf. Figure 21).

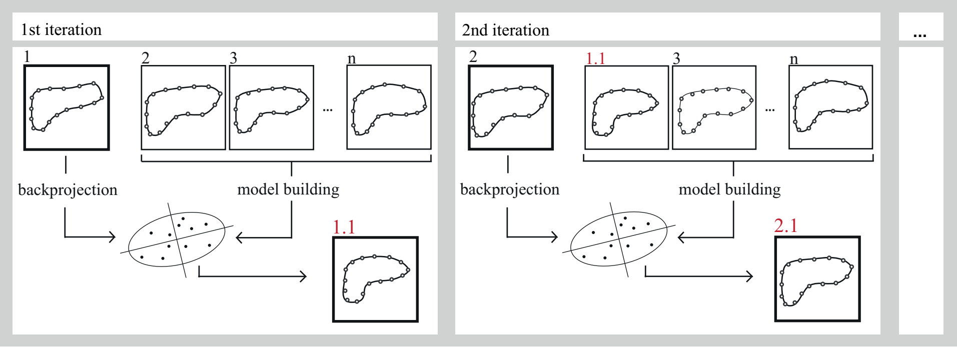
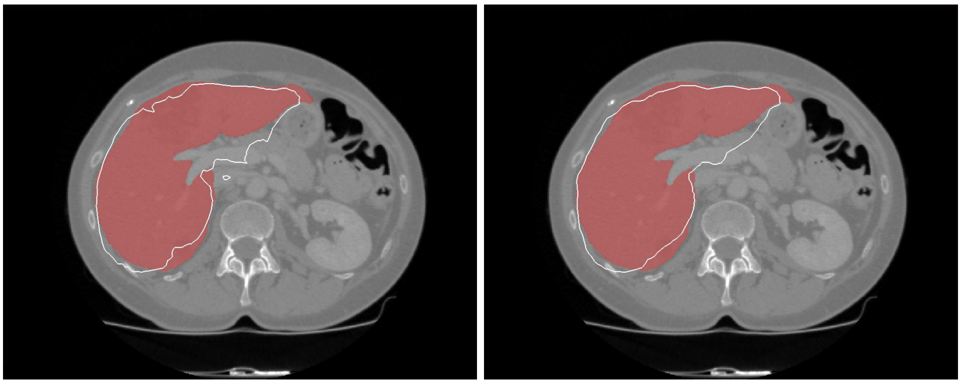


**Figure 21:** An omitted shape is back-projected onto the model of all other shapes. This process is iteratively performed for each shape.

Here, the transformation to the back-projected mesh is only applied to points with low probability, i.e. the back projection is again weighted with probabilities. This procedure is performed for all other shape examples, where the previously back-projected shape is used for the next model building iteration. This is performed until each shape was projected back onto the underlying subspace. The overall process is repeated a few times, until the spikes are decreased. Figure 22 demonstrates the effect of this it-



**(a)** Before **(b)** After

**Figure 22:** The effect **(a)** before and **(b)** after the iterative model building approach. The result is a slightly smoother shape representation.

erative leave-one-out model building approach. Finally, all reconstructed shapes are assumed to be free of outliers and therefore used to create a SSM. Figure 23 gives an overview of the proposed work flow.

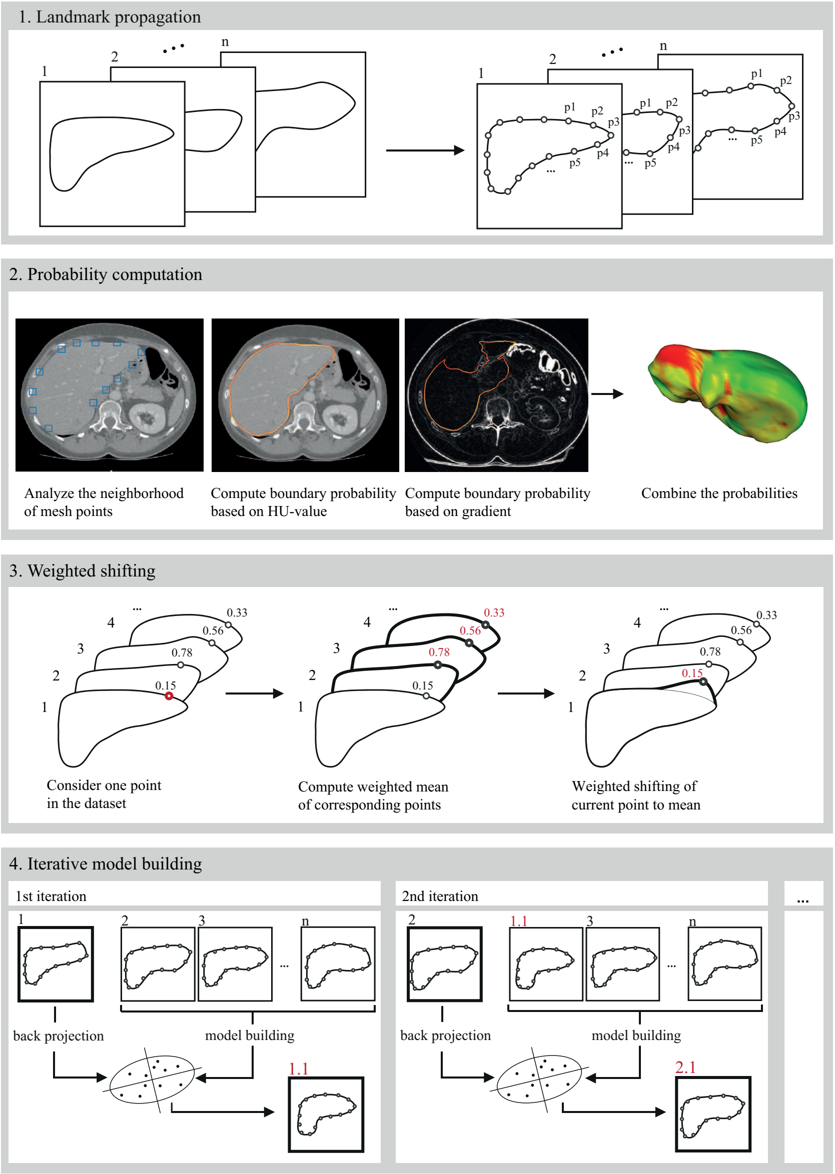

1. Landmark propagation
1
2
n
p1
p2
p3
p4
p5
...
2. Probability computation
Analyze the neighborhood of mesh points
Compute boundary probability based on HU-value
Compute boundary probability based on gradient
Combine the probabilities
3. Weighted shifting
1
2
3
4
...
0.15
0.78
0.56
0.33
Consider one point in the dataset
Compute weighted mean of corresponding points
Weighted shifting of current point to mean
4. Iterative model building
1st iteration
2nd iteration
...
back projection
model building
1.1
2.1

**Figure 23:** Overview of the imputation method.

This imputation approach is published under the title *Statistical Shape Modeling from Gaussian Distributed Incomplete Data for Image Segmentation* at the 4th MICCAI workshop on clinical image-based procedures in 2015 [MLH+15].

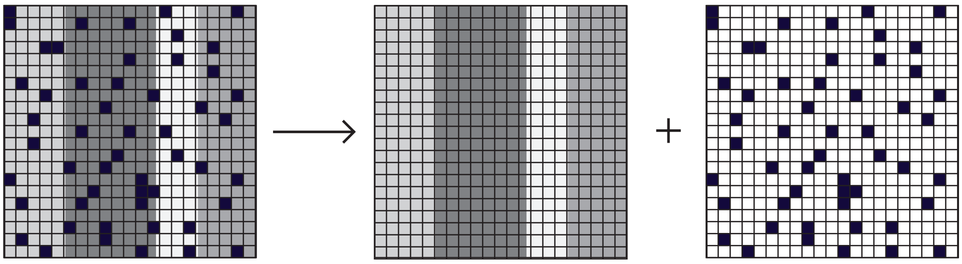
## 5.2 Robust PCA

In classical PCA, the large training data matrix $D$ in Equation 23 can be considered as a decomposition of a low-rank matrix $L$ and a small noise matrix $N$. The low-rank matrix $L$ can be recovered under the assumption, that the magnitude of perturbation is small and caused by i.i.d. Gaussian noise. Unfortunately, allowing corrupted shapes in the training set yields to arbitrarily corrupted entries of unknown magnitude in the perturbation matrix $N$ and classical PCA is not practicable, as Section 2.7 pointed out. Several approaches have been proposed in the literature, addressing the topic of *robustifying* PCA. As mentioned in Section 3, an idealized version of Robust PCA exists by Wright *et al.* [CLMW11]. The method relies on advances in sparse optimization in recent years and its robustness has been proved in several areas of application of computer science, such as the separation of moving objects from the background in video surveillance [BZ14].

In contrast to PCA, the data matrix $D \in \mathbb{R}^{3n_p \times n_S}$ is supposed to be the result of a low-rank matrix $L \in \mathbb{R}^{3n_p \times n_S}$ and an arbitrarily corrupted, but sparse[5] matrix $S \in \mathbb{R}^{3n_p \times n_S}$:

$$D = L + S \tag{38}$$

In Figure 24, this decomposition of a corrupted observation matrix is illustrated. Recovering the underlying low-rank matrix $L$, while $S$ holds the deformations caused by outliers, falls into the class of *constrained optimization problems*. The robust PCA problem can be formulated as:



Corrupted data matrix $D$ Underlying low-rank matrix Sparse error matrix

**Figure 24:** A visualization of splitting a corrupted data matrix $D$ into their underlying low-rank structure and a sparse component. Most of the entries in the error matrix are 0, but all non-zero entries are of arbitrary magnitude, i.e. unbounded.

$$\begin{aligned} \text{minimize} \quad & \operatorname{rank}(L) + \gamma \|S\|_0 \\ \text{subject to} \quad & D = L + S, \end{aligned} \tag{39}$$

[5] In a sparse matrix, only a fraction of the entries are affected.

where $\|\cdot\|_0$ is the $\ell_0$-norm[6], i.e. the total number of non-zero elements in the matrix. The parameter $\gamma$ is a positive weighting factor, balancing the contribution of the terms $L$ and $S$. This formulation is a highly non-convex[7] optimization problem, which is NP-hard and difficult to solve efficiently without approximation [WGR$^+$09]. To obtain a tractable optimization problem, the objective function is relaxed with a convex formulation of Equation 39. This is done by replacing the rank of $L$ with the nuclear norm $\|\cdot\|_*$, i.e. the sum of its singular values:

$$\|L\|_* = \sum_{i=1}^{\min(n_p,n_S)} \sigma_i \tag{40}$$

and the $\ell_0$-norm of $S$ with the $\ell_1$-norm, i.e. the maximum absolute column sum of the matrix:

$$\|S\|_1 = \max_{j=1,\ldots,n_S} \sum_{i=1}^{n_p} |a_{ij}|, \tag{41}$$

where $a_{ij}$ are the entries of the matrix $S$. Hence, this relaxation of the robust PCA problem yields to the following convex representative:

$$\begin{aligned} &\text{minimize} && \|L\|_* + \gamma\|S\|_1 \\ &\text{subject to} && D = L + S \end{aligned} \tag{42}$$

Wright *et al.* proved in [CLMW11], that each component of the matrices $L$ and $S$ can be exactly recovered. The only assumption is, that the matrix $L$ is of low-rank and not sparse, and the sparse components in $S$ are uniformly and randomly distributed. However, the entries in the matrix $S$ are allowed to be of arbitrarily large magnitude, in contrast to the small noise term $N$ in classical PCA.

The weighting parameter $\gamma$ is used to control the influence of the sparse component. For example, a high value for $\gamma$ would yield to a stronger influence on $S$, that means, rejecting more outliers in the data. A low value would behave like classical PCA [GMS$^+$14]. However, it is explicitly mentioned in [CLMW11], that whatever the choice of $\gamma$, the recovered matrices $L$ and $S$ are exact solutions of the problem. In practice, a universal value for the parameter $\gamma$ can be chosen from the maximum dimension of the data matrix $D$:

$$\gamma = \frac{1}{\sqrt{\max(3n_p, n_S)}} \tag{43}$$

[6] The $\ell_0$-norm is not actually a norm. It is defined as $\|a\|_0 = \sqrt[0]{\sum_i a_i^0}$, where zeroth-power and zeroth-root appear.

[7] A non-convex optimization problem may have multiple local optima, while a local optimum in a convex problem is at the same time a global optimal solution.

Several methods exist to solve constrained optimization problems. One of the most efficient approaches is the method of *Augmented Lagrange Multipliers* (ALM) in [Ber82], which seeks a solution by a set of unconstrained subproblems. The general method of ALM solves equality constrained optimization problems in the form:

$$\begin{aligned} &\text{minimize} && f(X) \\ &\text{subject to} && h(X) = 0, \end{aligned} \tag{44}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $h : \mathbb{R}^n \to \mathbb{R}^m$. By introducing a Lagrangian multiplier matrix $Y \in R^{m \times n}$, the equality constraint $h(X) = 0$ can be removed. The augmented Lagrangian function $\mathcal{L}$, which has to be minimized, is then defined as:

$$\mathcal{L}(X, Y, \mu) = f(X) + \langle Y, h(X) \rangle + \frac{\mu}{2} \|h(X)\|_F^2, \tag{45}$$

where $\mu$ is a positive scalar, penalizing the violation of the linear constraint. The notation $\langle \cdot, \cdot \rangle$ denotes the standard trace inner product between two matrices[8] of the same size, i.e. $\langle A, B \rangle = \text{trace}(A^T B) = \sum_{i,j} A_{i,j} B_{i,j}$. The induced Frobenius norm $\| \cdot \|_F$ of a real matrix $A \in \mathbb{R}^{m \times n}$ is defined as:

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2} \tag{46}$$

Thus, the method replaces the original constrained problem in Equation 44, by a sequence of unconstrained subproblems in Equation 45. Lin *et al.* showed in [LCM10], that ALM is applicable to the robust PCA problem with the following substitutions:

$$X = (L, S), \qquad f(X) = \|L\|_* + \gamma \|S\|_1, \qquad h(X) = D - L - S \tag{47}$$

According to this, the augmented Lagrangian function is given by:

$$\mathcal{L}(L, S, Y, \mu) = \|L\|_* + \gamma \|S\|_1 + \langle Y, D - L - S \rangle + \frac{\mu}{2} \|D - L - S\|_F^2, \tag{48}$$

To minimize Equation 48, a general ALM algorithm would minimize $L$ and $S$ simultaneously by iteratively setting:

$$(L_k, S_k) = \arg\min_{L,S} \mathcal{L}(L, S, Y_k, \mu_k) \tag{49a}$$

$$Y_{k+1} = Y_k + \mu_k (D - L_k - S_k), \tag{49b}$$

where $k$ describes the $k^{\text{th}}$ iteration. Lin *et al.* showed in [LCM10], that when the penalty parameter $\mu_k$ is progressively increasing, the Lagrange

[8] The trace of a product of two matrices is similar to the dot product of vectors.

multiplier $Y_k$ converges to the exact optimal solution. Hence, the optimal step size for updating $Y_k$ is $\mu_k$.

However, in the low-rank and sparse decomposition problem, $D = L + S$, the direct usage of the general ALM method ignores the fact, that the objective function and the constraint are separable [YY13]. A practical improvement of the ALM method splits the minimization of $\mathcal{L}$, with respect to $L$ and $S$, into two subproblems. This sequential variant of ALM is called the *Alternating Direction Method* (ADM) [YY13]. With ADM, minimizing these subproblems is done iteratively by repeatedly updating $L$ and $S$ as follows:

$$L_{k+1} = \arg\min_{L} \mathcal{L}(L, S_k, Y_k, \mu_k) \tag{50a}$$

$$S_{k+1} = \arg\min_{S} \mathcal{L}(L_{k+1}, S, Y_k, \mu_k) \tag{50b}$$

$$Y_{k+1} = Y_k + \mu_k(D - L_{k+1} - S_{k+1}) \tag{50c}$$

Both subproblems, Equations 50a and 50b, have simple and efficient solutions. In [CLMW11], the estimate of $S_{k+1}$ is solved as:

$$S_{k+1} = \mathcal{S}_{\gamma\mu_k^{-1}}\left[D - L_{k+1} + \mu_k^{-1}Y_k\right], \tag{51}$$

where $\mathcal{S}_\tau : \mathbb{R} \rightarrow \mathbb{R}$ is the so-called *shrinkage operator* with $\tau$ as a positive thresholding parameter, given by $\tau = \gamma\mu_k^{-1}$. This operator performs a selection for each element $a_{ij}$ of a matrix of being part of the sparse matrix:

$$\mathcal{S}_\tau\left[a_{ij}\right] = \text{sgn}(a_{ij}) \cdot \max(|a_{ij}| - \tau, 0) = \begin{cases} a_{ij} - \tau & \text{if } a_{ij} > \tau \\ a_{ij} + \tau & \text{if } a_{ij} < -\tau \\ 0 & \text{otherwise} \end{cases}, \tag{52}$$

where $\text{sgn}(a_{ij})$ tests the sign of $a_{ij}$:

$$\text{sgn}(a_{ij}) = \begin{cases} 1 & \text{if } a_{ij} > 0 \\ 0 & \text{if } a_{ij} = 0 \\ -1 & \text{if } a_{ij} < 0 \end{cases} \tag{53}$$

In this manner, each element of the matrix is proofed to be an outlier of the data (cf. Figure 25). Since the denominator $\mu_k$ is an increasing value in Equation 51, the threshold $\tau$ in $\mathcal{S}_\tau$ is decreasing in each iteration. The geometrical meaning of this is, that for example in the first iteration, the sparse matrix is filled with the largest outliers of the data.

The second subproblem in Equation 50a, estimating $L_{k+1}$, can be solved via a singular value thresholding operator $\mathcal{D}_\tau\left[A\right]$:

$$L_{k+1} = \mathcal{D}_{\mu_k^{-1}}\left[D - S_k + \mu_k^{-1}Y_k\right], \tag{54}$$

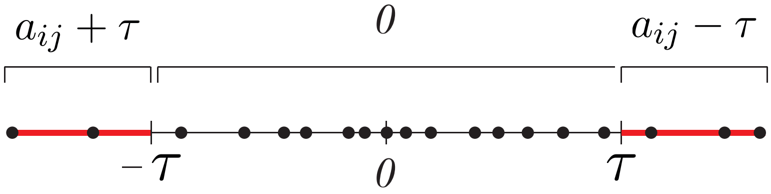


**Figure 25:** The influence of the shrinkage operator $\mathcal{S}$. All values in the range of $[-\tau, \tau]$ are set to zero in the sparse matrix $S$. Entries outside the interval, i.e. $[-\infty, -\tau]$ and $[\tau, \infty]$, correspond to outliers. These outliers are included in the sparse matrix and the values are set to the magnitude of the violation of the threshold, as indicated with the red lines.

where $\mathcal{D}_\tau$ limits the number of retained singular values in a matrix, with $\tau$ as the positive thresholding parameter. This operator computes the SVD of the matrix, followed by applying the shrinkage operator $\mathcal{S}_\tau$ onto $\Sigma$:

$$\mathcal{D}_\tau [A] = U \mathcal{S}_\tau [\Sigma] V^T \tag{55}$$

Thus, this thresholding operator tests, if the singular values of a matrix fall below the threshold $\tau$. Since singular values are all positive, the alternate expression of the shrinkage operator in Equation 52 can be reduced to $\mathcal{S}_\tau[a_{ij}] = \max(x - \tau, 0)$. Finally, the guess of $L_{k+1}$ is build from the inverse SVD, where all singular values below $\tau$ are set to 0 and the threshold is subtracted from the $sv$ valid singular values. This means, the estimation of $L_{k+1}$ is given by:

$$L_{k+1} = U \Sigma' V^T, \quad \Sigma' = \begin{pmatrix} \sigma_1 - \tau & & \\ & \ddots & \\ & & \sigma_{sv} - \tau \end{pmatrix} \in \mathbb{R}^{sv \times sv} \tag{56}$$

The meaning of this is, that the amount of such large singular values is bounded by the rank of $L$. As the threshold $\tau = \mu_k^{-1}$ decreases, the number of valid singular values increases. Thus, the rank of $L_k$ is monotonically increasing and converges to the true rank, since $\mu_k$ in an increasing sequence. [LCM10]

In practice, in the first iteration $k = 0$, the matrices $L_0$ and $S_0$ are filled with zeros. As an initialization of the Lagrange multiplier matrix $Y_0$, the authors in [LCM10] suggest to choose:

$$Y_0 = \frac{D}{\max\left(\|D\|_2, \gamma^{-1}\|D\|_\infty\right)}, \tag{57}$$

where $\|\cdot\|_2$ is the spectral norm of a matrix, i.e. the largest singular value[9] of the matrix:

$$\|D\|_2 = \sigma_{\max}(D), \tag{58}$$

[9] The positive singular values are the square roots of the eigenvalues.

and $\| \cdot \|_\infty$ is the maximum absolute row sum of the matrix:

$$\|D\|_\infty = \max_{j=1,\dots,n_p} \sum_{j=1}^{n_S} |a_{ij}| \tag{59}$$

The following assignments of matrix $Y_{k+1}$ in Equation 50c are based on the residual matrix $D - L - S$. The parameter $\gamma$ is universally chosen, such as in Equation 43 and $\mu_0$ is initially set to $1.25/\|D\|_2$, as suggested in [LCM10]. This procedure is summarized in Algorithm 3.

**Algorithm 3** Solving the robust PCA problem with ADM

```
Input: Observation matrix $D$, parameter $\gamma$
initialize $k = 0$
initialize $L_0, S_0 = 0$
initialize $Y_0 =$ Eq. 57
initialize $\mu_0 > 0$
while not converged do
    $L_{k+1} = \mathcal{D}_{\mu_k^{-1}} \left[ D - S_k + \mu_k^{-1} Y_k \right]$
    $S_{k+1} = \mathcal{S}_{\gamma\mu_k^{-1}} \left[ D - L_{k+1} + \mu_k^{-1} Y_k \right]$
    $Y_{k+1} = Y_k + \mu_k(D - L_{k+1} - S_{k+1})$
    update $\mu_k$ to $\mu_{k+1}$
    update $k$ to $k + 1$
end while
Output: $L_k, S_k$
```

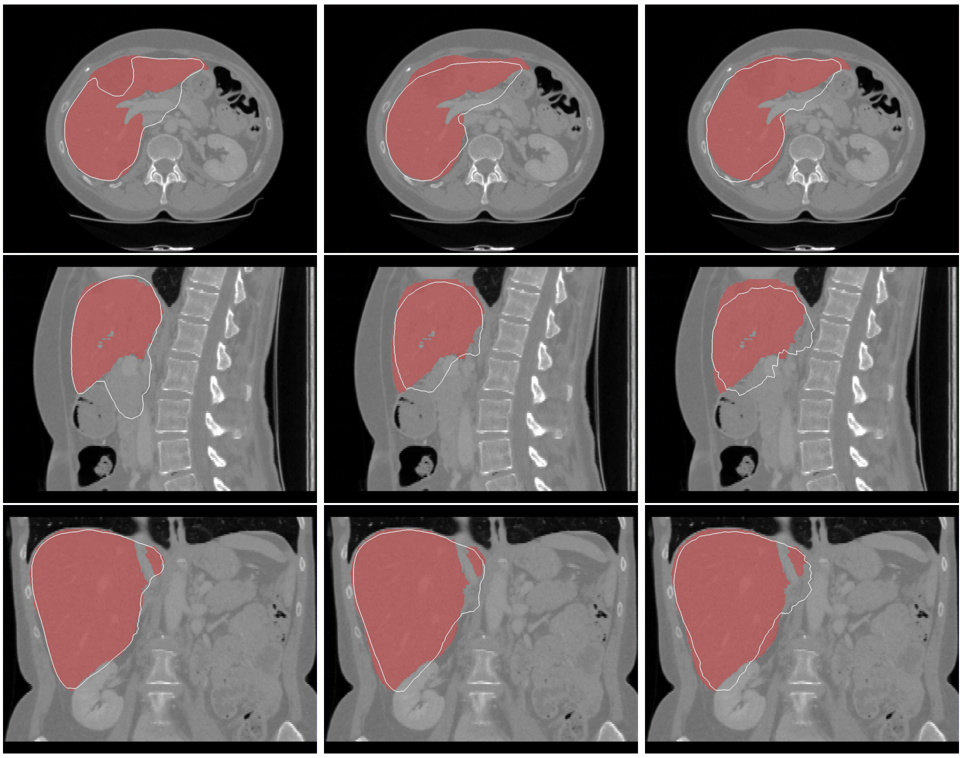
As proposed in [LCM10], a stopping criterion to terminate the algorithm could be:

$$\frac{\|D - L_k - S_k\|_F}{\|D\|_F} < \varepsilon, \tag{60}$$

where $\varepsilon$ is a small tolerance value. To confirm convergence, a value of $1 \times 10^{-7}$ is adequate. The parameter $\mu_k$ is assumed to be monotonically increasing after each iteration. Thus, any positive multiplication $\rho > 1$ is sufficient:

$$\mu_{k+1} = \rho \mu_k \tag{61}$$

In this work, the approach of robust PCA is used for outlier detection and correction. Solving the optimization problem via ADM, the deformations from corrupted landmarks in the training data matrix $D$ are placed in the sparse matrix $S$. The underlying low-rank structure of $D$ is recovered and stored in $L$. Figure 26 shows a shape example from the reconstructed low-rank matrix, compared to the initial and ground truth segmentation, as well as to the imputation method. Assuming that the data in $L$ is error-free, standard PCA is finally applied to build a statistical shape model.



**(a)** Before **(b)** After RPCA **(c)** After imputation

**Figure 26:** Three slices of the input image **(a)**, compared to the effect of RPCA **(b)** and the imputation approach **(c)** from Section 5.1. The top row is the transversal view, the middle row is the sagittal view and the bottom row is the coronal view. The red shading encodes the manually generated ground truth segmentation. RPCA yields to a smoother mesh reconstruction, compared to the imputation approach. Both approaches handles the wrong bulges in the transversal and sagittal views better than the initial representation.

In addition to the outlier correction using RPCA, the probabilities from Section 4 can be incorporated into the method, as some prior knowledge where the errors probably appear. One simple approach is to weight the transformations from all landmarks in the training data $D$, to the reconstructed low-rank matrix $L$ with their probabilities:

$$L'_{ij} = D_{ij} + (1 - P_{ij}) \cdot (L_{ij} - D_{ij}) \,, \tag{62}$$

where $i \in \{1, \ldots, n_p\}$ and $j \in \{1, \ldots, n_S\}$ are the indices of the entries of the matrices $L$ and $D$ and their boundary probability $P$. In this work, this is referred to *Weighted Robust PCA* (WRPCA) with outer weighting.

Another approach is to directly weight the selection of entries in the sparse matrix. Bringing Equation 52 into a different form:

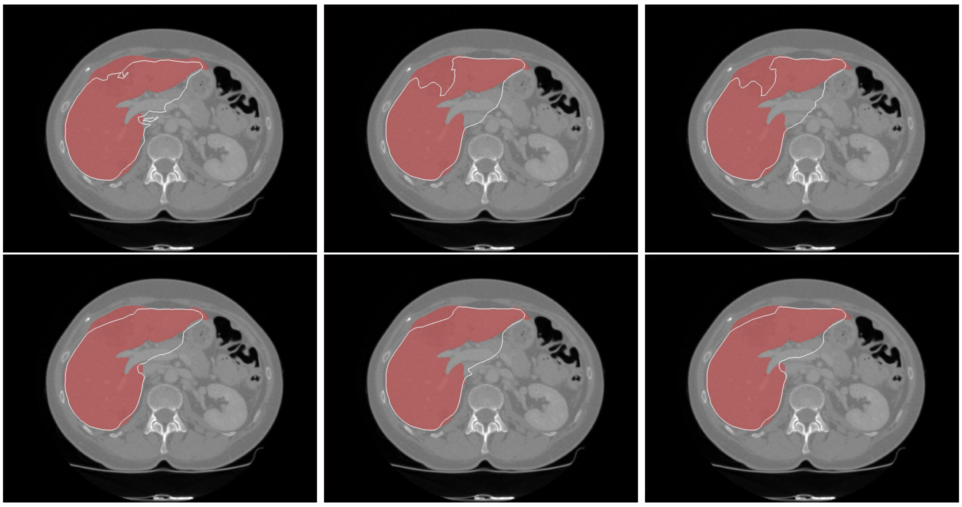
$$\begin{align} \mathcal{S}_\tau \left[a_{ij}\right] &= \text{sgn}(a_{ij}) \cdot \max(|a_{ij}| - \tau, 0) \tag{63} \\ &= \max(a_{ij} - \tau, 0) + \min(a_{ij} + \tau, 0) \tag{64} \end{align}$$

Then, the weights are applied by influencing the result of the minima and maxima parts:

$$\mathcal{S}_\tau \left[a_{ij}\right] = \max(a_{ij} - \tau - P_{ij} \cdot a_{ij}, 0) + \min(a_{ij} + \tau - P_{ij} \cdot a_{ij}, 0) \tag{65}$$

Thus, if the coordinate of the point in entry $a_{ij}$ has a high probability of being a boundary, i.e. $P \approx 1$, the results of the minima and maxima parts becomes 0, since $\tau$ is a positiv value. On the other hand, probabilities nearly 0, yield to the same result as standart RPCA. In this work, this is referred to WRPCA with inner weighting. Notice, that due to the additional weighting in $\mathcal{S}_\tau$, the amount of non-zero entries in $S$ decreases. Hence, the parameter $\gamma$ that regulates this, needs to be adapted, to get the same amount of non-zero entries in $S$, as in standard RPCA. Figure 27 compares the different results of WRPCA with built-in probabilities.

Notice, that the estimated probabilities provides better results with the outer weighting approach. In case of inner weighting, the accuracy of the estimated probabilities are not enough to get acceptable reconstructions. On the other hand, if optimal boundary probabilities are existent, the inner weighting gives slightly better boundary reconstructions, than the outer weighting. However, both of these approaches to incorporate the probabilities in RPCA are further evaluated in the next Section.



**(a)** Outer weighting **(b)** Inner weighting **(c)** Adapted $\gamma$

**Figure 27:** Comparison between outer weighting **(a)**, inner weighting **(b)** and inner weighting with adapted $\gamma$ **(c)** of a transversal slice. The top row shows the result with the estimated boundary probabilites from Section 4.3 and the bottom row shows the result with the optimal boundary probabilities from Section 4.4. The parameter $\gamma$ is chosen, such that the number of entries in the sparse matrix $S$ is roughly the same as in standart RPCA. Notice, that the inner weighting with estimated probabilities provides the poorest results, whereas the inner weighting with optimal probabilities produces the best results.

# 6 Evaluation

The intention of this work was to build reliable statistical shape models from routine clinical data, instead of using time-consuming ground truth data. The set of automatic segmented images from a particular organ of interest, is the results of any segmentation algorithm or non-professional manual delineations of the organs contour. The challenge was to find corrupted regions in the segmented meshes and manipulate them with the help of the statistics in the training set, to better reflect the class of corresponding ground truth data. The quality of the built SSM should be similar to a model built by ground truth data.

First, the estimated probabilities from Section 4.3 are rated, by reference to the optimal boundary probabilities from Section 4.4. Then, the evaluation of the two approaches, developed in 5.1 and 5.2, is divided in two different quality measurements - a model and a mesh evaluation. In the model evaluation, the associated SSMs of the two approaches have to be rated, in matters to the similarity to a SSM, built by the ground truth. To assess the statistical model, built by these reconstructed meshes, the measures *Generalization* and *Specificity* are considered. In the mesh evaluation, simply the reconstruction of the corrupted meshes are compared to the ground truth shapes. More precisely, the change in the error is measured before and after the corrupted data is handled. Different distance metrics are used to compare two shapes from the different data sets.

In both evaluations, the two approaches are performed with different settings. To rate the imputation approach, the algorithm is performed with 1, 3 and 5 iterations and without the iterative back-projection step. The RPCA evaluation is divided into the different weighting approaches, i.e. outer and inner weighting. In addition, the standard RPCA procedure without including the weights and the WRPCA with inner weighting and corrected $\gamma$ is computed. The available input training data for both approaches is introduced next.

## 6.1 Dataset

For evaluation, a training set of 63 clinical CT scans have been used. 19 data sets were taken from the public 3D-IRCAD data base[10], 17 training data sets from the MICCAI liver challenge [HvGS09] in 2007 and the remaining 27 were additional non-public data sets. These data sets have a slice dimension of $512 \times 512$ voxels and varying extent in the $z$-dimension in between $[129, 183]$. For evaluation purposes, ground truth data is available for each example. An existing segmentation algorithm is used to create the initial outlines of the structure of interest. Here, a liver segmentation was cho-

[10] www.ircad.fr

sen [EK10], since the data sets contain CT liver scans. By extracting the image data with Marching Cubes algorithm [LC87], the delineations are transferred to a mesh representation with 2402 points in $\mathbb{R}^3$. After finding corresponding points across the set of ground truth shapes and the initial segmented mesh data[11], each shape vector contains 2562 landmark points. In order to test whether the anatomical variance of the liver follows a Gaussian distribution in shape space, the 63 ground truth shape examples have been used for building a SSM. The projection of the first two principal components is visualized in Figure 28.

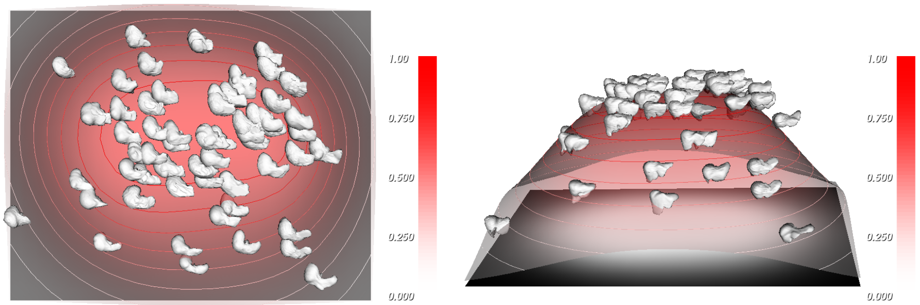


**Figure 28:** A liver SSM with underlying Gaussian distributed PDF. The shading encodes the probability of being a plausible liver shape.

The shading encodes the underlying probability density function of the model, where the saturation indicates the probability of a shape to be a plausible liver shape. The shapes cluster around the point with the highest probability, i.e. the mean shape is very representative for the distribution of the training data. The PDF continuously decreases from the point with the highest probability, therefore, a Gaussian normal distribution is assumed to be sufficient.

## 6.2 Probability Evaluation

To get a measure for evaluation of the estimated boundary probabilities $P$, based on the combined HU and gradient magnitude values, the *Mean Absolute Error* (MAE) to the optimal boundary probabilities $P_\text{D}$ from Section 4.4 is calculated. For a single shape, this is given by:

$$d_\text{MAE} = \frac{1}{n_p} \sum_{i=1}^{n_p} |P_i - P_{\mathbf{D}_i}| \tag{66}$$

Thus, the average magnitude of the absolute deviation of the probabilities is measured. In addition, the standard deviation $\sigma$ of $d_\text{MAE}$ is computed to

[11] Establishing correspondences together for both, the ground truth and the initial segmentations, is necessary for the model evaluation.

quantify the amount of variation of each boundary probability to the MAE:

$$\sigma_{d_{\mathrm{MAE}}} = \sqrt{\frac{1}{n_p} \sum_{i=1}^{n_p} (|P_i - P_{\mathbf{D}_i}| - d_{\mathrm{MAE}})^2} \tag{67}$$

In the computation of the boundary probabilities, the box sizes for the HU and the gradient were chosen to be $7 \times 7 \times 7$ and the inner box of the gradient is $9 \times 9 \times 9$. In order to form the probability based on the HU, a threshold $T = 30$ is chosen. This means, that distances between $h_{mean}$ and $h_{local}$ higher or equal $30$ are assigned a probability of $0$. For the probabilities based on the gradient, the maximum gradient $g_{max}$ is set to $60$. All gradient values that exceed $g_{max}$ have a probability of $1$. Both probability estimates are weighted equally with $\alpha = 0.5$. The threshold $T_{\mathrm{D}}$ for the optimal boundary probability calculations is set to $10$. Since the distances are measured with the meshes in the original size, the units correspond to *mm*. That means, distances exceeding $10$ *mm* are assigned the worst probability. In Figure 29, the results of $d_{\mathrm{MAE}}$ for each training shape are plotted with their corresponding standard deviation. Lower values indicate better results. The total average of all training shapes, i.e. $n_S^{-1} \cdot \sum_{i=1}^{n_S} d_{\mathrm{MAE}_i}$, yields a deviation of approximately $22\%$.

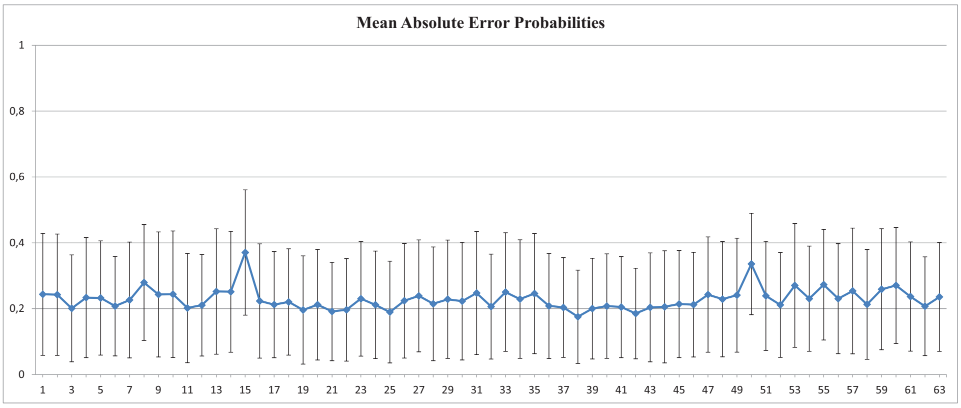


**Figure 29:** The Mean Absolute Error of the boundary probability estimation to the optimal distance probabilities, plotted for 63 shape examples. The vertical lines indicate the standard deviation of a particular shape. The average error is $22\%$.

### 6.3 Model Evaluation

A statistical shape model is described by the underlying PDF of the training meshes in the shape space. For Gaussian distributed meshes, PCA finds the coordinate system, centered in the mean and spanned by the axes with the highest variance. To evaluate these models, the whole region in shape space defined by the PDF has to be considered [DTT08]. Common specifications to assess the quality of a SSM are:

1. **Generalization**: The model is able to represent any instance of the PDF and not only the training data.
2. **Specificity**: The model represents only valid instances of the class of the training data.
3. **Compactness**: The model is described with the minimum amount of possible parameters. This is already given, since PCA reduced the dimensionality and a low-dimensional subspace is found [DTT08].

**Generalization** A statistical shape model should be able to generate any new instances of the class of the training shapes, not only the training shapes itself. Considering Figure 30a, a PDF with a high generalization ability scatters between and around the data points. If the generalization is low, the PDF does not cover the whole class of input data (cf. Figure 30b). This fundamental property permits learning the characteristics of the training meshes and utilizes the statistics to generate new shapes.

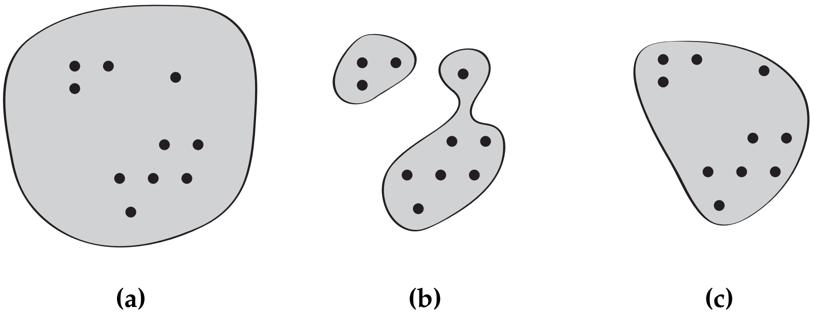


**Figure 30:** The PDF (grey) of different models is illustrated, where the data is represented as points (black). A model with **(a)** high generalization and low specificity overestimates the space spanned by the data points. A model with **(b)** high specificity and low generalization ability does not cover the whole space between the data points. A specific model which has also a high generalization ability is shown in **(c)**.

Generalization is usually computed using a leave-one-out measure, by building a model of all but one training mesh and reconstructing the omitted shape with the model PDF [GB10]. This procedure is repeated for all training shapes. The reconstructed meshes are reviewed, relating to the accuracy of the reconstruction to the original training mesh. This error is averaged over the whole distribution. The leave-one-out method has one drawback, it only probes the models PDF at the data points [DTT08]. As mentioned above, the entire class of the training shapes should be considered in a meaningful evaluation. Hence, another way to compute the generalization ability, is to generate samples within the PDF and compare these samples to the training data. With the set of input training shapes $X_\mathbf{t} = \{\mathbf{x_t}_1, \mathbf{x_t}_2, \ldots, \mathbf{x_t}_{n_S}\}$ and a set of samples $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_K\}$, generated by sampling the PDF $\mathrm{p}(b)$ of $X_\mathbf{t}$ (see Equation 21), the generalization $\mathcal{G}(m)$ for $m$ retained modes can be computed by:

$$\mathcal{G}(m) = \frac{1}{n_S} \sum_{i=1}^{n_S} \min_{\mathbf{y} \in Y} \left( d(\mathbf{x}_i, \mathbf{y}) \right) \tag{68}$$

where $d(\cdot, \cdot)$ is any distance metric between two shapes. Thus, the generalization is just the mean of all minimum distances for each training mesh to the nearest sample in $Y$. Due to this, the appearance of the sample affects the quality of the resulting generalization value. Notice, if each generated sample is similar to one of the training shapes, the result will be low. Thus, to minimize the generalization ability, samples preferable similar to the training meshes have to be generated. The variations of the input shapes are described by the eigenvectors and the eigenvalues of the model. To cover most of the distribution of training shapes with the samples, different modes of variation have to be considered. For example, if only a single mode is retained, the parameter vector $b$ is build from the first principal component by randomly sampling in the range $\left[-3\sqrt{\lambda_1}, 3\sqrt{\lambda_1}\right]$. In the interpretation of the results, smaller values stand for a higher generalization. The standard deviation of $\mathcal{G}(m)$ can be computed:

$$\sigma_{\mathcal{G}(m)} = \sqrt{\frac{1}{n_S} \sum_{i=1}^{n_S} \left( \min_{\mathbf{y} \in Y} d(\mathbf{x}_i, \mathbf{y}) - \mathcal{G}(m) \right)^2} \tag{69}$$

**Specificity** A specific model only generates plausible examples, i.e. shapes similar to the training shapes. Figure 30b represents a PDF with a high specificity ability, where the PDF is centered around the data points. Though, in 30a, the PDF represents more than the space spanned by the data points. Thus, the model can generate shape instances different to the data points. The property of generating specific shapes can be described similar as the generalization ability:

$$\mathcal{S}(m) = \frac{1}{K} \sum_{i=1}^{K} \min_{\mathbf{x} \in X} (d(\mathbf{x}, \mathbf{y}_i)), \tag{70}$$

where $K$ is the amount of randomly generated samples. Different to $\mathcal{G}$, the mean of all minimum distances from the population of samples to the training shapes is computed. The smaller the values, the more specific is the model PDF. However, a small result of $\mathcal{S}$ does not yield to a coverage of the complete training data [DTT08]. Therefore, the evaluation of generalization and specificity should be considered and evaluated in combination. Figure 30c illustrates a model with both, a high generalization as well as a high specificity ability. The standard deviation of $\mathcal{S}(m)$ is given by:

$$\sigma_{\mathcal{S}(m)} = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(\min_{\mathbf{x}\in X} d(\mathbf{x},\mathbf{y}_i) - \mathcal{S}(m)\right)^2} \tag{71}$$

In summary, the computation of $\mathcal{G}(m)$ and $\mathcal{S}(m)$ calculates the distance between the set of training shapes to a population of instances within its class. The two specifications rate the quality of one model. The evaluation in this work however, should achieve a rating of one model referring to the model built by the ground truth data. Therefore, the procedure of computing generalization and specificity is modified. Additionally to the training set $X_{\mathbf{t}}$ and their sample set $Y$, the corresponding ground truth set $X_{\mathbf{gt}} = \{\mathbf{x_{gt}}_1, \mathbf{x_{gt}}_2, \ldots, \mathbf{x_{gt}}_{n_S}\}$ has to be involved. Instead of using $X_{\mathbf{t}}$ and $Y$ for the computation of $\mathcal{G}(m)$ and $\mathcal{S}(m)$, $X_{\mathbf{gt}}$ and $Y$ are used. For example, consider the resulting shapes of the RPCA approach as $X_{\mathbf{t}}$. A model is build from this training data and samples $Y$ are generated within this model. Then $Y$ is compared to the ground truth set $X_{\mathbf{gt}}$ by computing $\mathcal{G}(m)$ and $\mathcal{S}(m)$.

### 6.3.1 Surface-based evaluation

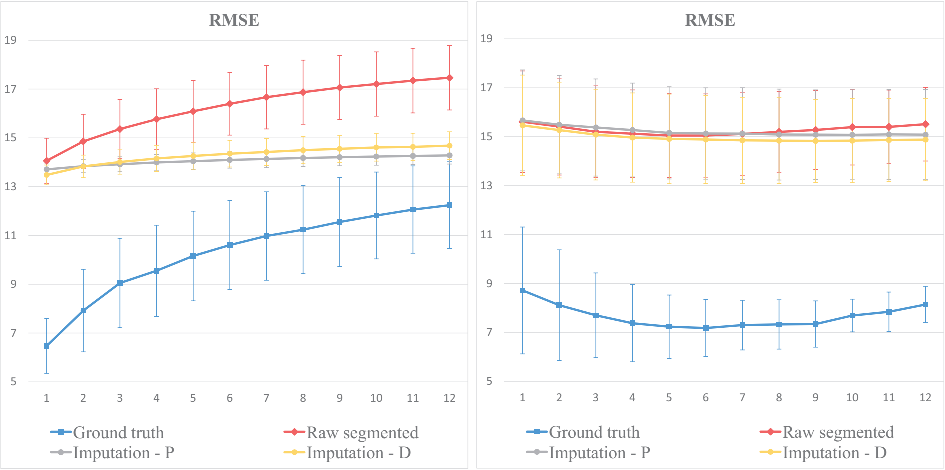
As mentioned before, any distance metric can be used in the evaluation of $\mathcal{G}(m)$ and $\mathcal{S}(m)$, to compare the surfaces of two shape vectors. One simple approach is a point-to-point distance measure, based on the corresponding landmarks of the two shapes $\mathbf{y}$ and $\mathbf{x_{gt}}$. With the linear MAE measure from Equation 66, all differences between the points would be weighted equally. In order to give a relatively high weight to large differences of corresponding landmarks, the *Root Mean Square Error* (RMSE) $d_{\text{RMSE}}$ can be used instead for comparison:

$$d_{\text{RMSE}}(\mathbf{y}, \mathbf{x_{gt}}) = \sqrt{\frac{1}{n_p}\sum_{i=1}^{n_p} \|p_i - q_i\|^2}, \tag{72}$$

where $p \in \mathbf{y}$ and $q \in \mathbf{x_{gt}}$. The RMSE indicates the square root of the mean of the squared Euclidean distances between the two surfaces $\mathbf{y}$ and $\mathbf{x_{gt}}$. Notice, that point correspondences between $Y$ and $X_{\mathbf{gt}}$ are needed. By finding corresponding landmarks between $X_{\mathbf{t}}$ and $X_{\mathbf{gt}}$, implicitly correspondence between the generated samples in $Y$ and $X_{\mathbf{gt}}$ exists.

In the following, the models of the proposed meshes from Sections 5.1 and 5.2 are evaluated. The two approaches are performed with different settings and in each case with the estimated image-based (P) and the optimal distance probabilities (D). Furthermore, the ground truth model itself is rated, by generating samples directly in this model, as a reference. The RMSE is used as a distance metric and the amount of retained modes of variation $m$ in $\mathcal{G}(m)$ and $\mathcal{S}(m)$ is set to 12. To attempt a high coverage of the PDF, the number of generated samples $K$ is set to $10000$.

The results of specificity and generalization ability for the imputation approach with 3 iterations is shown in Figure 31. Varying the amount of iterations do not show significant differences. Hence, the graph of the imputation method is taken as a representative, due to better visualization. Notice, since a SSM is scaled during model building, the units of generalization and specificity are arbitrary. However, by keeping track of the average scaling factor, it is possible to rescale the outcome. Thus, the original unit of *mm* is reobtained.



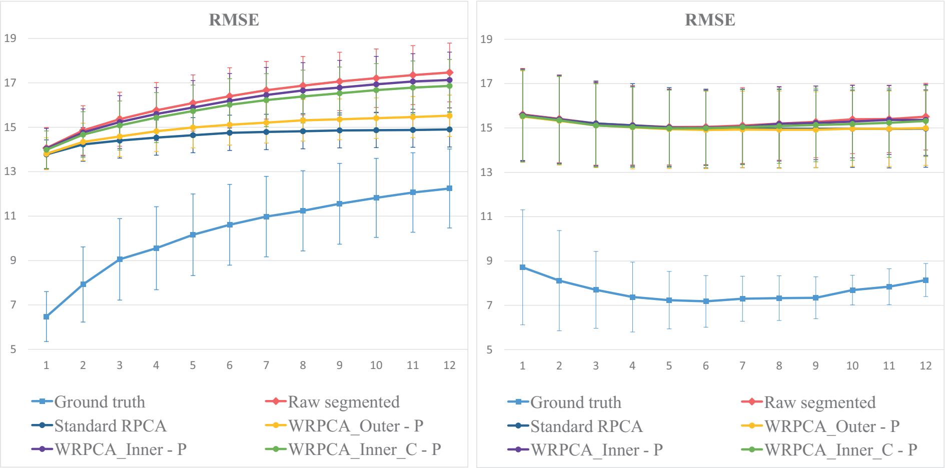
**(a)** Specificity

**(b)** Generalization

**Figure 31:** The specificity and generalization ability of the imputation approach, computed with the RMSE as internal shape metric.

In both cases, the ground truth achieves the best results, as the samples are generated directly in the model of $X_{\mathbf{gt}}$. The imputation method yields to a more specific model in the relation to the ground truth, than the raw segmented meshes. Both, estimated and optimal probabilities, show similar results. Unlike the specificity of the raw segmented and the ground truth, the specificity of the imputation only increases slightly with the number of retained modes and the standard deviations are smaller. Here the reconstruction accuracy of the ground truth model is about $2-7$ *mm* better. However, the generalization ability of the imputation approach provides only slightly improvements to the raw segmented meshes with a reconstruction accuracy to the ground truth of about $6,7\ mm$.

The evaluation results for RPCA are plotted in Figure 32. Again for better visualization, only the estimated probabilities are shown. On average, the approaches with the optimal distance probabilities are $0,2\ mm$ better than with the image-based estimates. Again, all variations of RPCA yield to a higher specificity and generalization than the raw segmented meshes. Against the expectations, the standard RPCA approach achieves slightly better specificity results than WRPCA. This issue will be discussed later in this section. Concerning generalization, the values are all similar, with no significant improvements.



**(a)** Specificity

**(b)** Generalization

**Figure 32:** The specificity and generalization ability of the RPCA approach, computed with the RMSE as internal shape metric.

### 6.3.2 Volume-based evaluation

The major drawback of the model evaluation is the dependence on corresponding landmark points. As these are obtained from all training and ground truth shapes, perfect correspondence cannot be assumed to exist between the data. Thus, the distribution of the landmarks can cause better results for worse models, where slightly deviations occur. Ideally, correspondence is established again for each pair of investigated sample $\mathbf{y}$ and ground truth shape $\mathbf{x_{gt}}$. From a computational point of view, this would be impractical though. Therefore, other measures without the need of corresponding features are required. However, if $10000$ samples with each $2562$ points are used, the approach proposed in Section 4.4 and other closest point algorithms are still to complex. [HWM06]

By changing the distance metric within the computation of $\mathcal{G}(m)$ and $\mathcal{S}(m)$ to a volumetric difference, the measure becomes independent of the underlying landmark distribution [CCH06]. One method to define such shape similarity in a volumetric representation is the *Tanimoto coefficient (TC)*, also known as Jaccard coefficient. The TC is defined by the ratio of the number of voxels in the intersecting set of two binary images $A$ and $B$, to the number of voxels in the union:

$$d_{\mathrm{O}}(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{73}$$

where $d_{\mathrm{O}} \in [0,1]$. An overlap of $d_{\mathrm{O}} = 1$ means, the meshes are exactly the same. If $d_{\mathrm{O}} = 0$, the two images do not coincide at all. A distance metric is obtained by computing the volumetric error $d_{\mathrm{VE}}$:

$$d_{\mathrm{VE}} = (1 - d_{\mathrm{O}}) \cdot 100 \tag{74}$$

To integrate the TC in the proposed evaluation method, the meshes have to be converted into a binary volume representation. This is done by stenciling each shape with a reference volume, where $1$ indicates the foreground and $0$ the background. According to [HWM06], this procedure to obtain the overlap of two shapes is less time consuming than the complex surface distance computation, even if the meshes first have to be converted to a volume representation. Since a volume representation requires more memory space, the amount of randomly generated samples is set to $800$.

To generate this reference volume for a set of input shapes, the region covered by the meshes has to be found. Since the data is centered around the origin, the extent $h$ in each dimension $x, y$ and $z$ can be computed by finding the minimum and maximum bounds in this data set. For example, the extent in direction $x$ is computed by $h_x = max_x - min_x$. The spacing $u$ is defined to be equal in all three dimensions and is described by:

$$u = \frac{\max(h_x, h_y, h_z)}{N_v} \tag{75}$$

where the value $N_v$ is a predefined maximal number of voxels in one dimension. The actual amount of voxels $(dim_x, dim_y, dim_z)$ is computed for each dimension by:

$$dim_x = \frac{h_x}{u}, \quad dim_y = \frac{h_y}{u}, \quad dim_z = \frac{h_z}{u} \tag{76}$$

The resulting volume has an equally spacing, but a different amount of voxels in each dimension (cf. Figure 33).

Applying the volumetric error measurement to the $\mathcal{G}(m)$ and $\mathcal{S}(m)$ evaluation, the ground truth meshes $X_{\mathbf{gt}}$, as well as the sample set $Y$ have to be transformed into a volume representation. The result for the imputation

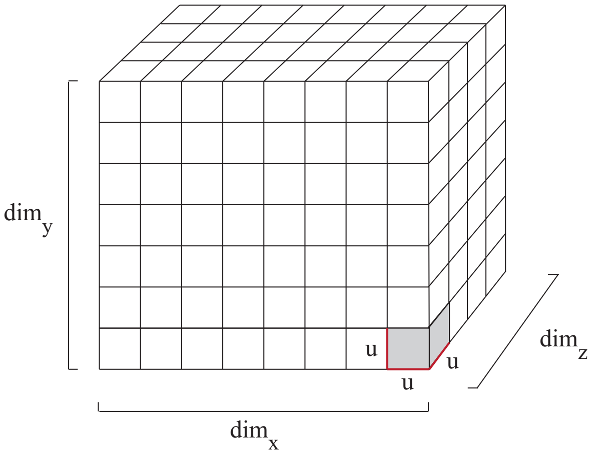
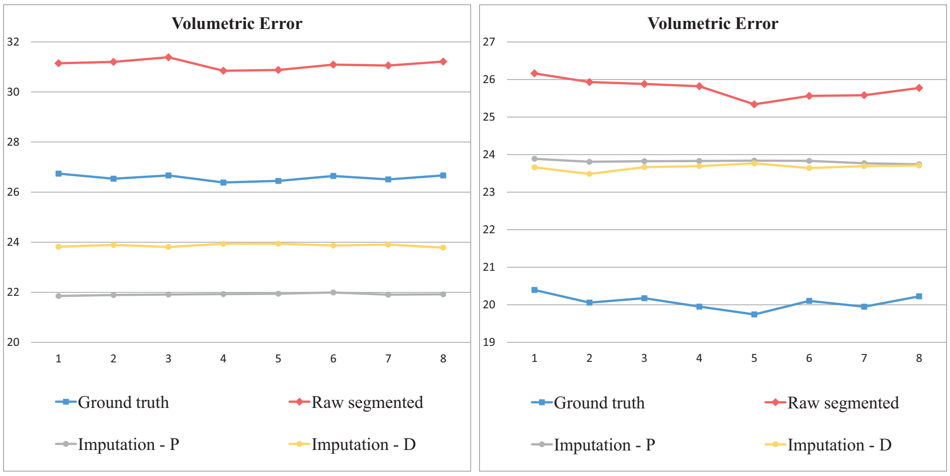


**Figure 33:** The generated reference volume for the evaluation is defined by the triple $(dim_x, dim_y, dim_z)$. The voxel spacing $u$ is equal in each dimension.

approach is visualized in Figure 34, where the results for three iterations are used as a representative. Due to the limited memory space, the value $N_v$ is set to 200 and the number of generated samples to 800.

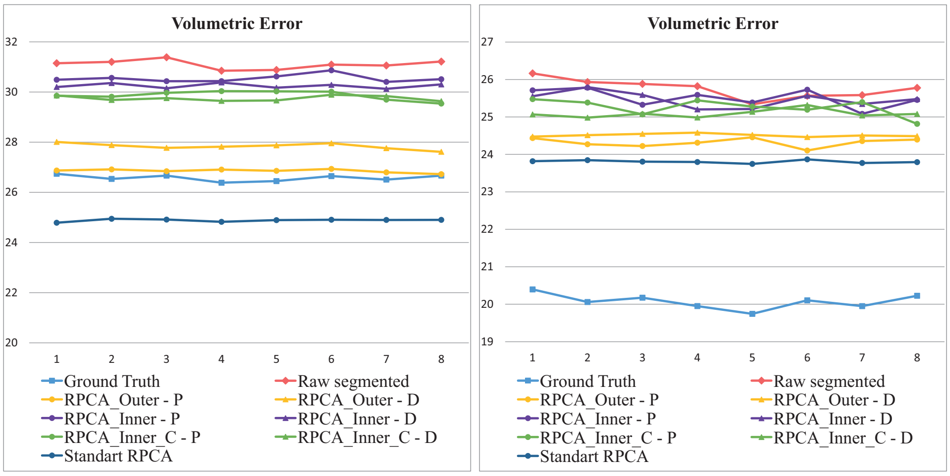


**(a)** Specificity

**(b)** Generalization

**Figure 34:** The specificity and generalization ability of the imputation approach, computed with the volumetric error as internal shape metric.

Directly noticeable, the results indicate greater differences between the developed approaches and the raw segmented meshes. With the volumetric error as the shape distance metric, the generalization ability shows greater improvements. However, the specificity show some unexpected results, where both imputation methods are more specific than the ground truth samples. Same with the RPCA approaches in Figure 35, where the standard RPCA achieves the best specificity, even compared to the ground truth. Furthermore, standard RPCA yields to the result with the closest generalization graph to the ground truth. However, all developed approaches show improvements compared to the raw segmented model samples.



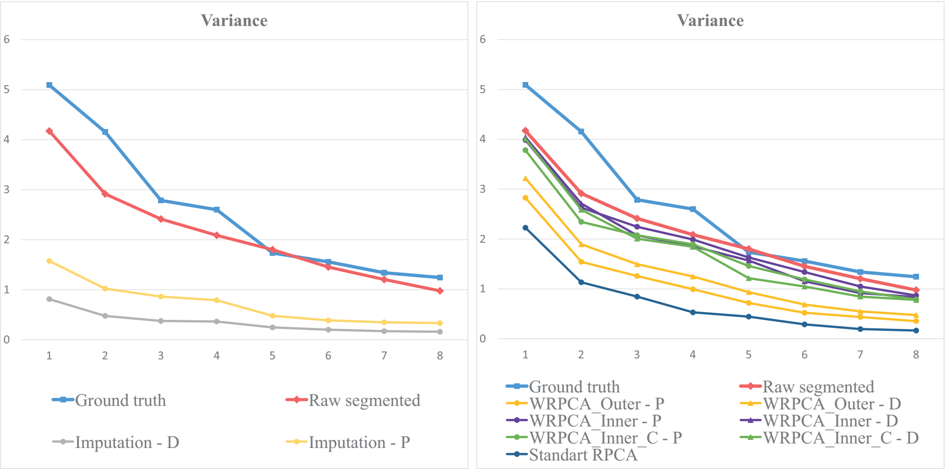
**(a)** Specificity

**(b)** Generalization

**Figure 35:** The specificity and generalization ability of the RPCA approach, computed with the volumetric error as internal shape metric.

### 6.3.3 Discussion

Concerning the unexpected results from the model evaluation with the RMSE and the volumetric error, some arguments need to be considered. The main problem arises, when different datasets are used, such as the imputation approach, RPCA and ground truth, where samples are generated within the fixed interval of $\left[-3\sqrt{\lambda_k}, 3\sqrt{\lambda_k}\right]$. The eigenvalues of the models of those different training sets for generating samples highly deviate, as Figure 36 points out. The ground truth data has the highest variance in the first modes, followed by the raw segmented data set. Noticeable is the low variance of the imputation methods and the standard RPCA approach. Exactly these meshes achieved a high specificity and generalization in the previous evaluation. In Figure 37, the ellipses, constructed by the variances of the first two principal components are drawn. The space spanned by the ground truth is much bigger than the space spanned by the imputation method. In the evaluation of $\mathcal{G}(m)$ and $\mathcal{S}(m)$, samples are generated in this small space and compared to the ground truth meshes. Hence, the generated samples cannot cover the whole variance of the ground truth. In the



**(a)** Imputation approach

**(b)** RPCA approach

**Figure 36:** The eigenvalues of the proposed approaches show smaller values, compared to the ground truth and raw segmented data. The imputation approach in **(a)** suffers the most loss of variance, whereas the WRPCA preserves more shape variability, i.e. higher eigenvalues.

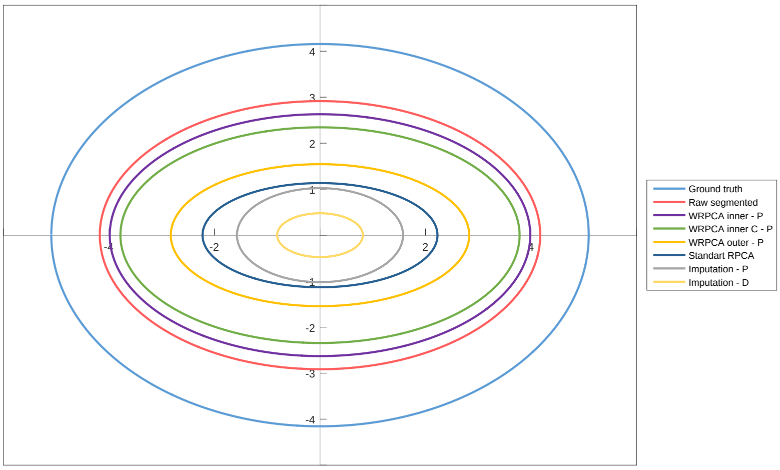


**Figure 37:** The ellipses are drawn from the eigenvalues of the first two principal components. The legend is sorted by decreasing order of the amount of variance.

imputation approach, the meshes are all shifted towards the mean. The resulting meshes as well as the generated samples are similar to the mean and do not show a high variance. Due to this, it is reasonable, that the imputation method achieves a high specificity compared to the ground truth. In the computation of $\mathcal{S}(m)$ for each of these mean-like meshes, the most similar ground truth is found. Thus, the PDF of the ground truth follows a Gaussian distribution, the chance that a sample of the imputation method finds a similar ground truth in the mean area is high. The same applies for the standard RPCA approach. Due to this, considering only generalization and specificity as a model quality measurement, is not sufficient in this modified evaluation procedure. The properties of the PDF have to be taken into account in the rating. Notice, that all proposed approaches in Figure 37 yield to a decreasing variance to a lower or higher degree, compared to the original raw segmented shape data. However, all WRPCA approaches preserve the retained variance better than the imputation and standart RPCA. Therefore, by considering all given facts from evaluating specificity, generalization and the variance, the methods of the weighted RPCA obtain the most reasonable results and can be recommended to build a SSM from erroneous data. By correcting $\gamma$ in the inner weighting, the model can be slightly improved, however, deciding between outer and (corrected) inner weighting is difficult. The outer weighting achieves better values in the specificity and generalization ability, whereas the inner weighting has the smallest loss of variance.

All experiments were performed on an Intel Core i5 3570k CPU desktop PC with 8 GB RAM. The computation for the surface-based evaluation with $10000$ samples took 18 minutes for 12 modes. The evaluation of the volume-based shape difference with $800$ samples took 110 minutes for 8 modes. It could be argued, that more samples would improve the accuracy of the evaluation. However, $800$ samples was the limit under the conditions of the hardware.

## 6.4 Mesh Evaluation

The described model evaluation probes the PDF of a SSM in the entire class of the training shapes. According to this, the actual physical shapes of the reconstructed meshes need to be considered independently [DTT08]. This is done by computing the distance between these training meshes and the corresponding ground truth.

### 6.4.1 Surface-based evaluation

For mesh evaluation, first, the distance metric $d_{\text{RMSE}}$ is performed for each pair of the $63$ training meshes from the initial segmentation to the corresponding ground truth. Each of these pairs are centered around the origin and mutually aligned with the Procrustes method from Section 2.3. The average of all distances is taken as the reference error $d_{\text{ref}}$ of the raw segmented mesh data, i.e. before the approaches from Sections 5.1 and 5.2 are applied:

$$d_{\text{ref}} = \frac{1}{n_S} \cdot \sum_{i=1}^{n_S} d_{\text{RMSE}_i} \tag{77}$$

Then, the mean RMSE in the above equation is recomputed after the meshes have been reconstructed using the proposed methods from this work. Here, the estimated boundary probabilities (P) and the optimal boundary probabilities (D) are both considered again. Finally, Figure 38 shows the average RMSE for the imputation approach and Figure 39 the same for RPCA. The lower the values, the smaller is the deviation to the ground truth data.

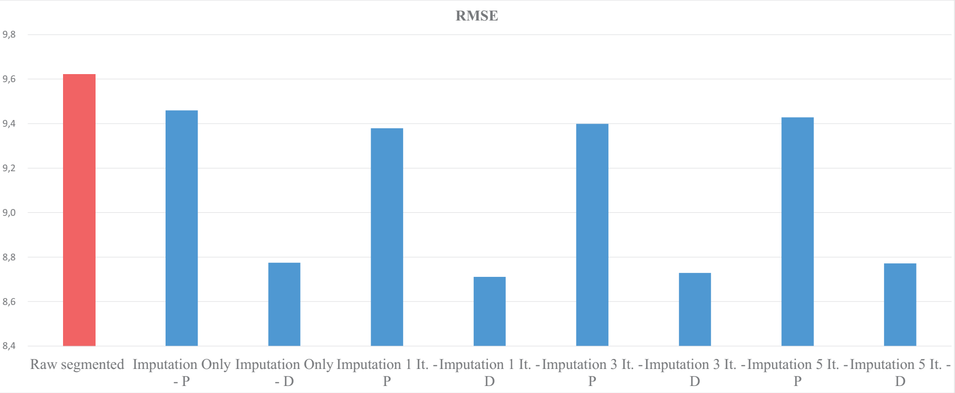


**Figure 38:** The average RMSE of all meshes after the imputation approach. Smaller values indicate better results, i.e. a smaller error.

Since the results in both figures are less than the reference error, both approaches generate on average meshes, more similar to the ground truth than the raw segmented meshes. Trivially, the results from the imputation method in Figure 38 show, that the optimal probabilities provide lower results and therefore a smaller error to the ground truth than the estimated image-based probabilities. Furthermore, the influence of the iterative back-projection steps are limited to small changes, as recognized in the model evaluation. However, the best result is given by 1 iteration.

The RPCA results in Figure 39 yield to the best outcome with the outer weighting. Notice, that the outer weighting with probabilities achieve a higher value than without using any weights. By varying the parameter $\gamma$ in the inner weighting as suggested, slightly better results are obtained.

Since this assessment is based on point correspondences again, plausible results are not ensured. As mentioned in the model evaluation, closest point algorithms are a better choice. The computational complexity in the

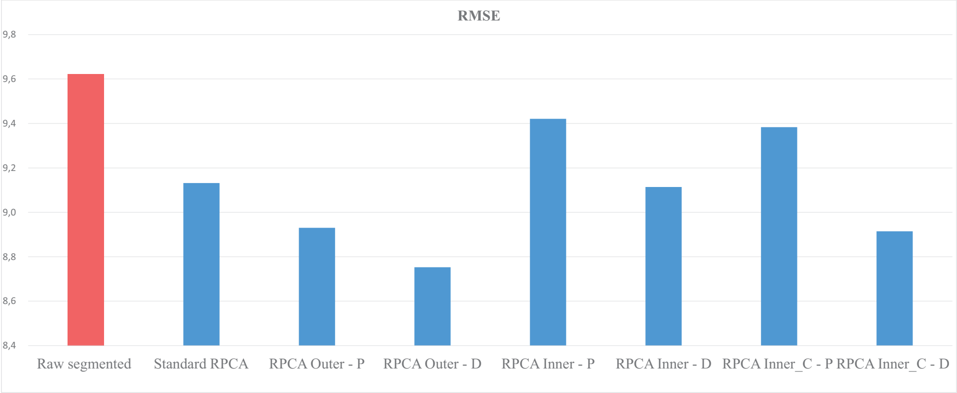


**Figure 39:** The average RMSE of all meshes after the RPCA approach. Smaller values indicate better results, i.e. a smaller error.

mesh evaluation is much lower, as only $63$ shape comparisons are computed, instead of $63 \cdot 10000$ in the model evaluation. One possible closest point metric is the Hausdorff distance, which is described below.

The *Hausdorff distance* is a common technique to compute the difference between sets of points. It is based on the Euclidean-norm and is very sensitive to outliers [GB10]. The metric is given by computing twice the *directed Hausdorff distance* and finding the maximum. The directed Hausdorff distance is described as the maximum distance to the nearest points on another mesh:

$$d_{\mathrm{dHD}}(\mathbf{x_t}, \mathbf{x_{gt}}) = \max_{p \in \mathbf{x_t}} \min_{q \in \mathbf{x_{gt}}} \|p - q\| \tag{78}$$

This means, for every point in $\mathbf{x_t}$, a landmark in $\mathbf{x_{gt}}$ is found, where the Euclidean norm is minimized. The directed Hausdorff distance $d_{\mathrm{dHD}}$ arises from the maximum of these smallest Euclidean norms. Equation 78 computes an oriented distance from one shape to another. It is obvious that $d_{dHD}(\mathbf{x_t}, \mathbf{x_{gt}}) \neq d_{dHD}(\mathbf{x_{gt}}, \mathbf{x_t})$. The general Hausdorff distance $d_{\mathrm{HD}}$ between $\mathbf{x_t}$ and $\mathbf{x_{gt}}$ is formed from the maximum of $d_{\mathrm{dHD}}$ in both directions:

$$d_{\mathrm{HD}}(\mathbf{x_t}, \mathbf{x_{gt}}) = \max\left(d_{\mathrm{dHD}}(\mathbf{x_t}, \mathbf{x_{gt}}), d_{\mathrm{dHD}}(\mathbf{x_{gt}}, \mathbf{x_t})\right). \tag{79}$$

The Hausdorff distance is applied to the evaluation of the reconstructed meshes. The averaged results for the imputation approach are visualized in Figure 40 and for the RPCA approach in Figure 41. The Hausdorff distance is always positive and smaller values represent better results, i.e. higher similarity to the ground truth. Same as the RMSE evaluation, all approaches yield to smaller distance values than the raw segmented data.

By considering the different imputation method performances, the lowest value is reached by the weighted shifting. Unlike the RMSE evaluation, the smoothing iteration steps do not further improve the Hausdorff distance.

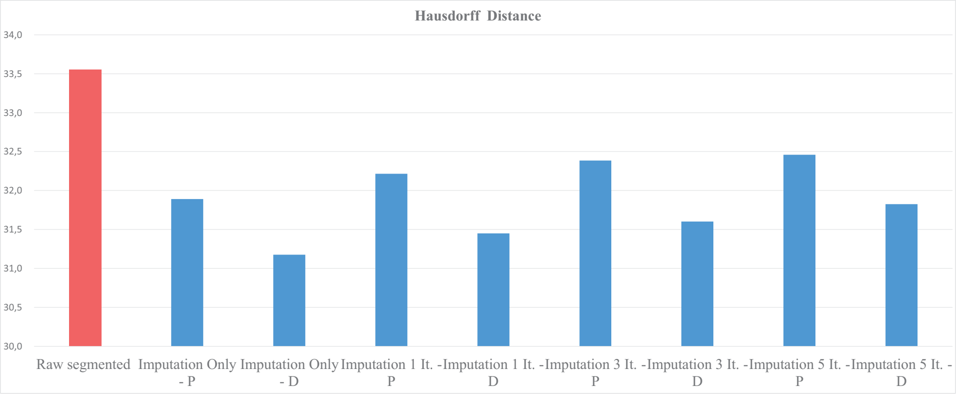


**Figure 40:** The average Hausdorff distance of all meshes after the imputation approach. Smaller values indicate better results, i.e. a smaller error.

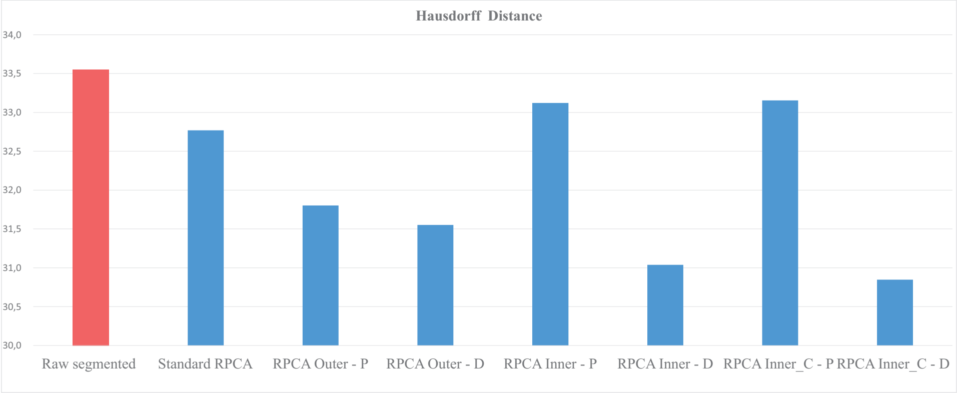


**Figure 41:** The average Hausdorff distance of all meshes after the RPCA approach. Smaller values indicate better results, i.e. a smaller error.

In the Hausdorff evaluation of the different RPCA performances, the meshes of the RPCA with optimal inner weighting and corrected $\gamma$ achieve the best results. However, the worst performance, i.e. the lowest similarity to the ground truth, is the RPCA with estimated inner weighting. This corresponds to the previously findings from Figure 27. Since the Hausdorff distance is very sensitive to outliers, the results show that the estimated probabilities are too inaccurate to reduce all corrupted parts. This sensibility has to be handled carefully. In particular, a case of similar meshes, a single high distance value results in a high Hausdorff distance. Compared to the standard RPCA, the use of estimated probabilities, however, can enhance the mesh reconstruction in the case of outer weighting.

### 6.4.2 Volume-based evaluation

Same as with the model evaluation, the meshes can be transferred to a volume-based representation, to get rid of the dependence of the underlying point structure. Thus, it is possible to use the Hausdorff distance as a measure in a volume representation. However, the results did not show significant differences to the surface-based Hausdorff distances. Therefore, the volumetric overlap is used to get a different perspective of shape comparison.

After transforming all 63 training meshes and all 63 ground truth meshes to a binary volume representation, as described in Section 6.3.2, each overlap $d_O$ from Equation 73 is computed between the $n_S$ shape pairs. In Figure 42 and 43, the mean of the overlap, i.e. $n_S^{-1} \sum_{i=1}^{n_S} d_{O_i}$, is shown for both proposed approaches and the raw segmented data.

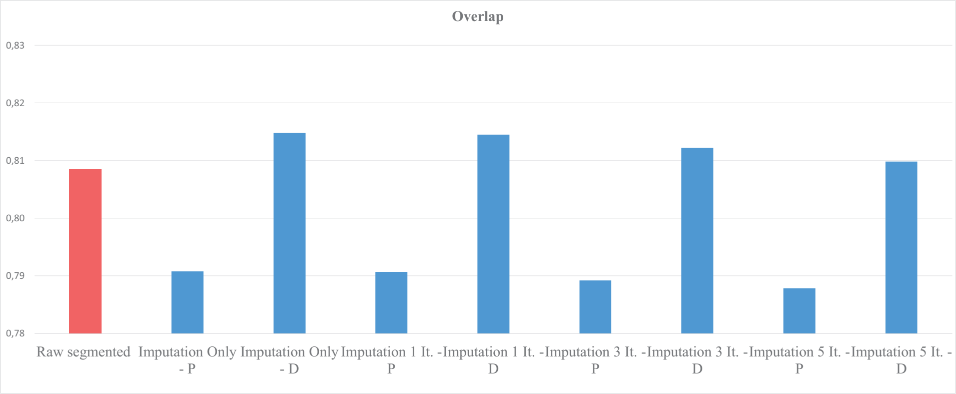


**Figure 42:** The average overlap of all meshes after the imputation approach. Higher values indicate better results, i.e. a better match with the ground truth.

The imputation results from Figure 42 show a decreasing mesh quality with the estimated probabilities. In contrast, the optimal boundary prob-

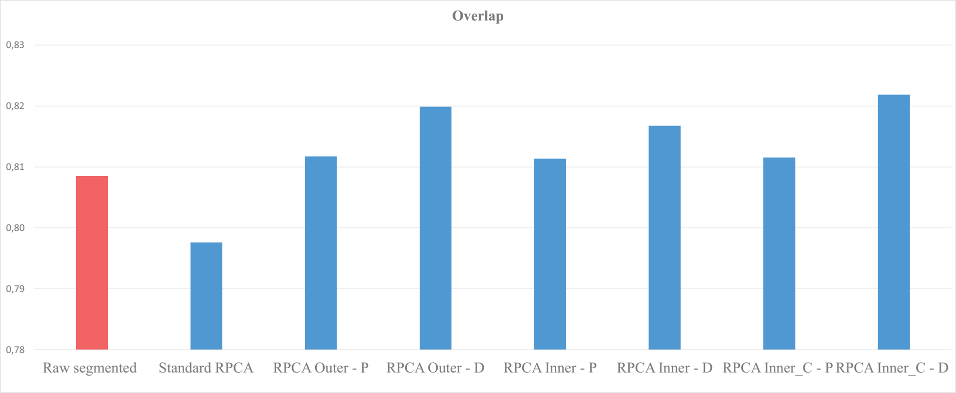


**Figure 43:** The average overlap of all meshes after the RPCA approach. Higher values indicate better results, i.e. a better match with the ground truth.

abilities achieve a higher overlap to the ground truth. This strengthen the speculation, that the estimated probabilities in the imputation method are not sufficient. As discovered in Section 6.2, the estimated probabilities vary about 22% to the optimal probabilities. It can be argued, that due to this, actually good segmented regions however have a 22%-higher chance of being shifted towards the mean shape. In this case, the structure of the physical shape is hard to preserve.

On the other hand, the WRPCA approaches from Figure 43 improve the overlap to the ground truth, even with the estimated probabilities. Here, the WRPCA approach with inner weighting achieves the best result on average. Furthermore, standard RPCA yields to a degradation of the overlap, where the reconstruction provides mean-like meshes again.

Finally, in Figure 44, the different proposed approaches with the estimated boundary probabilities are compared with three example shapes and in Figure 45 with the optimal probabilities respectively.

## 6.5 Conclusion

Summarizing the results from the proposed evaluation methods, the reconstructed meshes of both, the imputation as well as the RPCA approach, achieve a higher quality, i.e. a lower distance to the ground truth. In every case, the meshes weighted by the estimated probabilities have a lower agreement to the ground truth, than with the optimal probabilities. As shown in Evaluation 6.2, the image-based probabilities deviate 22% from the optimal probabilities. By using these estimates, only the WRPCA approach with outer weighting yields to acceptable results. Using the optimal weights, both approaches show reasonable outcomes. The imputation approach achieves a good specificity and generalization ability, the eigenvalue analysis in 6.3.3 however, shows a high loss of variance. The meshes

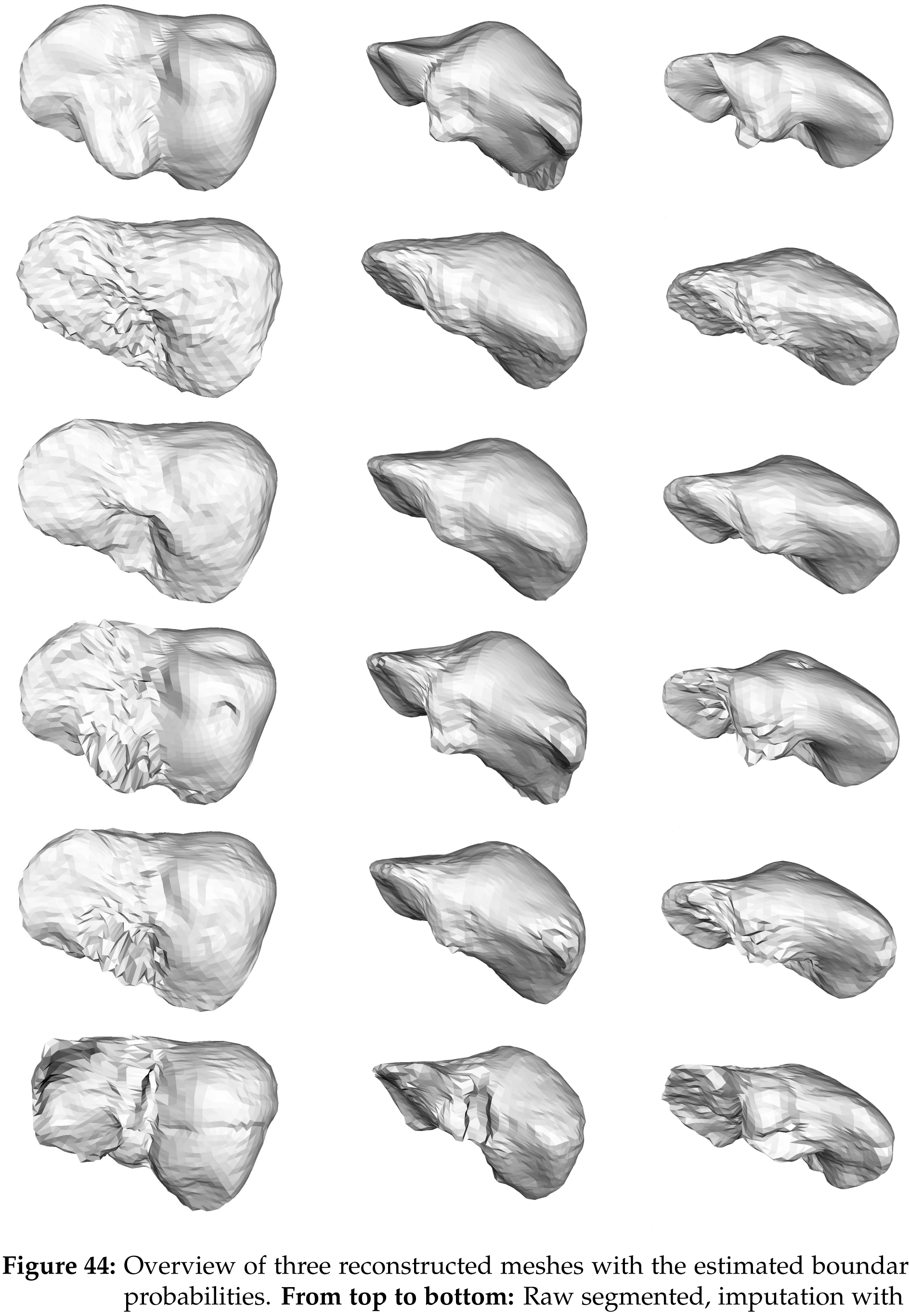


**Figure 44:** Overview of three reconstructed meshes with the estimated boundary probabilities. **From top to bottom:** Raw segmented, imputation with 1 iteration, standard RPCA, inner WRPCA with corrected $\gamma$, outer WRPCA, ground truth.

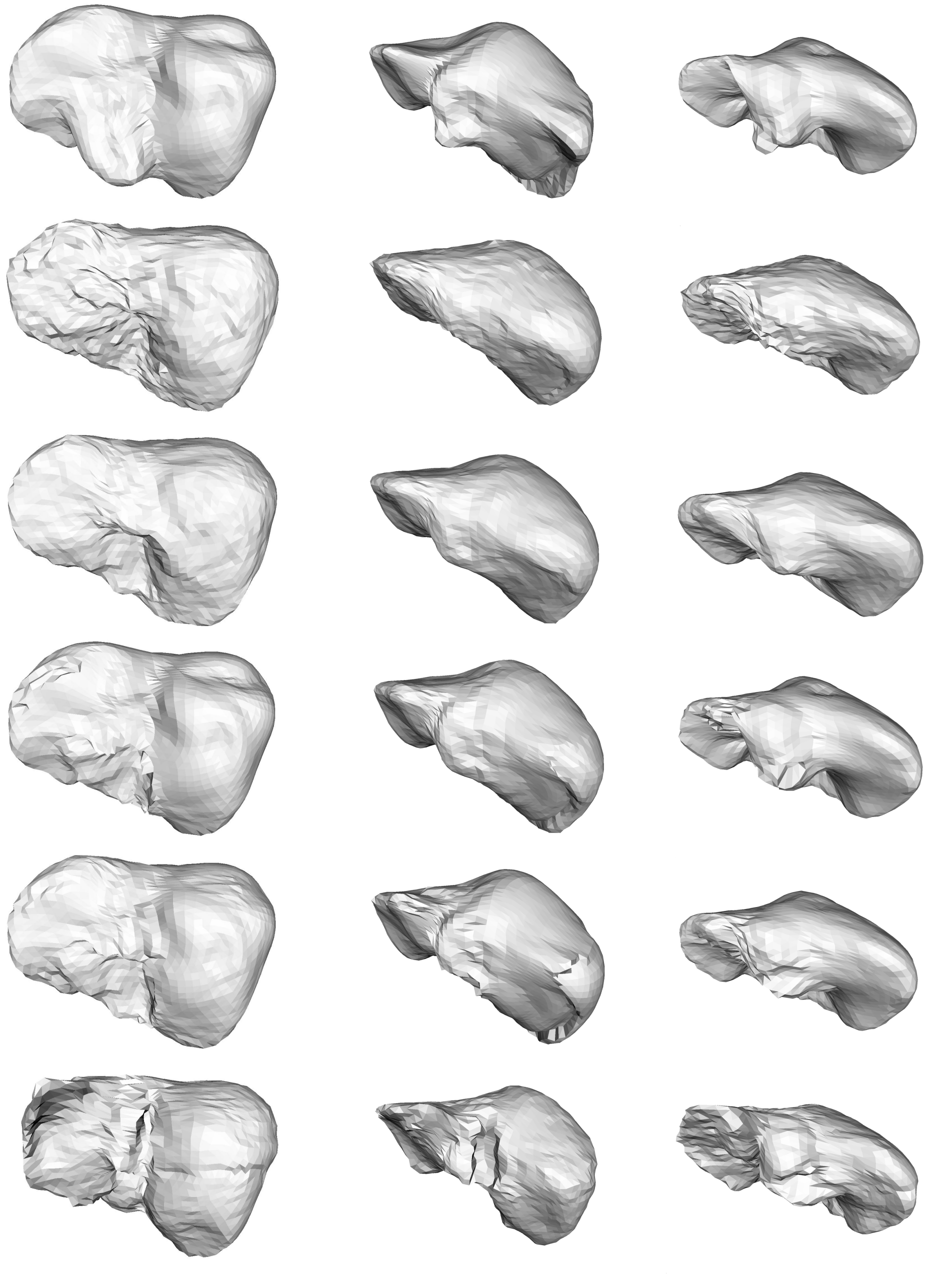


**Figure 45:** Overview of three reconstructed meshes with the optimal boundary probabilities. **From top to bottom:** Raw segmented, imputation with 1 iteration, standard RPCA, inner WRPCA with corrected $\gamma$, outer WRPCA, ground truth.

are more similar to each other than the ground truth meshes, so they cannot cover the whole variance of the ground truth. Considering the different iteration steps, the meshes with one iteration achieves slightly better results than the others. Comparing the two developed approaches, the RPCA with the optimal weighting scores in nearly every case the best outcome. By rating the different settings in the RPCA approach, the standard RPCA has a higher dissimilarity to the ground truth than the WRPCA. In the model evaluation, the outer weighting has the highest specificity and generalization ability, on the other hand, in the mesh evaluation, the inner weighting with the corrected $\gamma$ attains in most cases the best results. Regarding the retained variance, the standard RPCA has the highest loss from the class of RPCA. The method with inner weighting retained the most variance.

# 7 Outlook and Further Work

Regarding the evaluation results, it can be argued, that the output of the segmentation algorithm used to create the corrupted training data may already be of good quality. By using the distance method from Section 4.4, without clamping the values to a certain threshold, the average error of the input segmentations and the ground truth is 4,41 *mm*. From the 63 shape examples, the maximum mean point deviation is 8,34 *mm* and the minimum is 2,92 *mm*. Hence, other segmentation algorithms might lead to training data with a higher degree of errors and a poorer initial shape model. It would be interesting to see the improvement by using the meshes of other segmentation algorithms. Furthermore, the resulting SSM could be integrated into a model-based segmentation algorithm and the methods from this work could be iteratively applied to see if the quality of the model can be further enhanced.

The amount of ground truth training shapes is typically limited and rarely publicly available. With the proposed approaches from this work, potentially hundreds or thousands of corrupted training shapes can be incorporated in the model building procedure. It would be also interesting to see, how many data sets are needed in order to create a SSM of the same or better quality, than a shape model generated from a limited number of ground truth data.

In this work, the construction of a statistical shape model is based on a landmark representation. Thus, point correspondences are needed in order to model the variability of a population of shape examples. As mentioned in the evaluation, a volume-based representation is a better choice for the internal shape comparison metric, to get rid of the problems that arise from the point correspondences. Building correspondences means, relocating points to specific positions of the shapes in the data set. Thus, the distribution of the landmarks of the meshes could be suboptimal (cf. Figure 46). The density of the mapped points can slightly vary and the distribution of the landmarks can become sparse in certain regions [HWM06]. Thus, it may happen that the represented mesh varies from the original mesh. Hence, building correspondences can cause errors. Additionally in this work, the proposed reconstruction approaches can cause some changes in the landmark distribution and it is not ensured, that dense correspondence exists anymore.

By considering the evaluation in a volume-representation, the problem was only solved partially. However, in the steps of building a model from landmark meshes, the correspondence error is already included. By transferring the meshes exclusively in the evaluation into a volume-based representation, the correspondence error cannot be undone, it simply preserves the error of increasing. To overcome this correspondence error, the whole pipeline of model building has to be transferred into a volume-based proce-

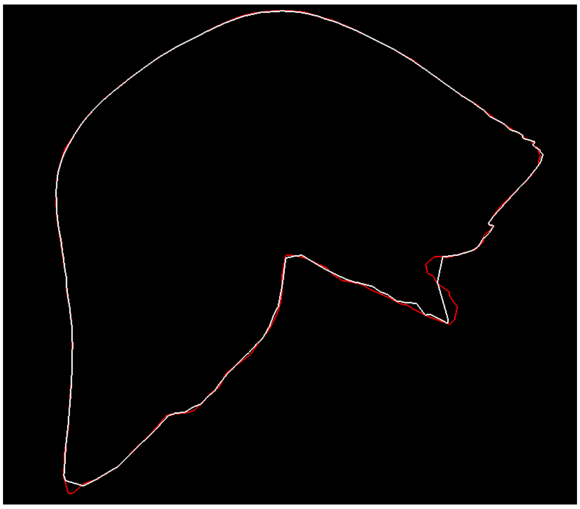


**Figure 46:** A problem of landmark-based shape representation is shown. The distribution of points varies after correspondence is established. Especially, the deviations appear in regions with a high curvature. The red outline shows the initial segmentation and the white outline the mesh after correspondence is established.

dure, e.g. where the boundary shape is implicitly described with a signed distance map. However, there are also some limitations. Considering the pipeline developed in this work, one problem arises with the handling of boundary probabilities. In the landmark-based representation, the 3D-points were shifted towards a specific point, in relation to their probability. Assuming a probability for each voxel, a simple shifting of voxels is not possible. Furthermore, by changing the value of voxels in the signed distance map representation, it could happen, that the zero-level line becomes sparse and the SDM is not representative anymore. Another shortcoming is, that building a shape model with PCA can become erroneous, if the volume is defined by a SDM. As said in [DRT08], building linear PCA on SDMs can cause unrealistic shapes by linear combination of the SDMs. To solve this issue, Cremers *et al.* proposed a method in [COS06], where kernel PCA and a Parzen estimator is used for modeling the shape distribution.

# 8 Conclusion

In this work, methods for constructing a statistical shape model without the need of manually delineated ground truth data have been proposed. The training data is assumed to be the result of any segmentation algorithm or may originated from non-expert annotators. Depending on this data acquisition, the shape examples will contain regions with erroneous boundary segmentations. In order to handle such corrupted parts, each landmark point is assigned a probability of being a boundary. These estimated probabilities rely on image-based methods and have an average deviation to the optimal probabilities of about 22%. During the further model building procedure, two different approaches were introduced, to treat the erroneous data appropriately, before PCA is applied. The statistics inherent in the training data is used for reconstruction of the corrupted shapes. The imputation method is a rather brute-force approach, where low probability points are replaced with the mean of corresponding landmarks. Recent advances in sparse optimization yield to a robustified version of PCA, where a low-rank matrix is recovered from the corrupted training data matrix. Outliers are separated from the data, by solving a convex optimization problem. By incorporating the boundary probabilities into the RPCA method, the prior knowledge can be exploited, by weighting the selection of the outliers in RPCA. After the training data is reconstructed, using either the imputation or RPCA approach, the data is assumed to be free of outliers. By applying PCA to the data, the low-dimensional linear subspace is found, to perform dimensionality reduction and model the variability of the training data.

In order to test whether the imputation and RPCA approaches enhance the quality of the resulting shape model to the initial segmentations, an evaluation of 63 liver CT scans compared the generalization and specificity ability and the difference between the training shapes and ground truth. Both approaches showed improvements of the mesh reconstruction, where the WRPCA achieves the best outcome. By rating the different weighting methods, the outer, as well as the inner with the corrected $\gamma$, result in good reconstructions. The model built by these meshes is of reasonable quality and preserves a high amount of variation. However, using these models, built of 63 reconstructed meshes from routine clinical data instead of 63 high-quality ground truth, leads to a degraded quality. It would be interesting to see, how the quality of the model is affected by using potentially hundreds or thousands of corrupted training shapes and compared to a model, built with only a limited amount of ground truth data.

The approaches developed in this work could be further integrated in a model-based segmentation algorithm, by first building a SSM of a set of reconstructed meshes in an initialization step and than using this model as a reference for new segmentations. The underlying shape model can be updated in every segmentation phase, in order to increase the amount of training shapes in the model. However, the proposed approaches allow the inclusion of even low-quality data, gained during clinical routine, to enlarge the amount of shape variations in a SSM. Accepting, that the quality of a ground model cannot be reached, corrupted shape examples can be used to create a reasonable SSM. Hence, new applications can benefit from the robust framework to build reliable statistical shape models.

# References

[ANJY06] Julien Abi-Nahed, Marie-Pierre Jolly, and Guang-Zhong Yang. Robust active shape models: A robust, generic and simple automatic segmentation tool. In *Proceedings of the 9th International Conference on Medical Image Computing and Computer-Assisted Intervention - Volume Part II*. Springer-Verlag, 2006.

[Ber82] Dimitri P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982. ISBN 1-886529–04-3.

[BZ14] Thierry Bouwmans and El Hadi Zahzah. Robust pca via principal component pursuit: A review for a comparative evaluation in video surveillance. *Computer Vision and Image Understanding*, pages 22–34, 2014.

[CCH06] William R. Crum, Oscar Camara, and Derek L. G. Hill. Generalized overlap measures for evaluation and validation in medical image analysis. *IEEE Transactions on Medical Imaging*, 2006.

[CLMW11] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, June 2011.

[COS06] Daniel Cremers, Stanley J. Osher, and Stefano Soatto. Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *International Journal of Computer Vision*, pages 335–351, September 2006.

[CTCG92] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. *BMVC92: Proceedings of the British Machine Vision Conference*, chapter Training Models of Shape from Sets of Examples, pages 9–18. Springer London, 1992.

[CTCG95] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, January 1995.

[DRT08] S. Dambreville, Y. Rathi, and A. Tannenbaum. A framework for image segmentation using shape models and kernel space shape priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1385–1399, August 2008.

[DTT08] Rhodri Davies, Carole Twining, and Chris Taylor. *Statistical Models of Shape: Optimisation and Evaluation*. Springer Publishing Company, Incorporated, first edition, 2008. ISBN 1848001371, 9781848001374.

[EK10] M. Erdt and M. Kirschner. Fast automatic liver segmentation combining learned shape priors with observed shape deviation. In *IEEE 23rd International Symposium on Computer-Based Medical Systems (CBMS)*, October 2010.

[ELF97] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3-d rigid body transformations: A comparison of four major algorithms. *Machine Vision and Applications - Special issue on performance evaluation*, March 1997.

[GB10] Sebastian T. Gollmer and Tharsten M. Buzug. A method for quantitative evaluation of statistical shape models using morphometry. In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. IEEE Press, April 2010.

[GMS$^+$14] B. Gutierrez, D. Mateus, E. Shiban, B. Meyer, J. Lehmberg, and N. Navab. A sparse approach to build shape models with routine clinical data. In *IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, April 2014.

[Gow75] J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 1975.

[HM09] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3d medical image segmentation: A review. *Medical image analysis*, 2009.

[HvGS09] T. Heimann, B. van Ginneken, and M. Styner. Comparison and evaluation of methods for liver segmentation from CT datasets. *IEEE Transactions on Medical Imaging*, 2009.

[HWM06] Tobias Heimann, Ivo Wolf, and Hans-Peter Meinzer. Optimal landmark distributions for statistical shape model construction. In *Proc. SPIE Medical Imaging: Image Processing*, 2006.

[IR10] Alexander Ilin and Tapani Raiko. Practical approaches to principal component analysis in the presence of missing values. *Journal of Machine Learning Research*, August 2010.

[JMIC15] Hans J. Johnson, M. McCormick, L. Ibáñez, and The Insight Software Consortium. *The ITK Software Guide Book 2: Design and Functionality*. Kitware, Inc., fourth edition, 2015. ISBN 1-930934-28-9.

[Jol02] I.T. Jolliffe. *Principal Component Analysis*, chapter Mathematical and Statistical Properties of Sample Principal Components, pages 29–61. Springer New York, 2002.

[KW10] Matthias Kirschner and Stefan Wesarg. Construction of groupwise consistent shape parameterizations by propagation. In *Medical Imaging 2010: Image Processing. Part One*, 2010.

[LAV09] Marcel Lüthi, Thomas Albrecht, and Thomas Vetter. Building shape models from lousy data. In *Proceedings of the 12th International Conference on Medical Image Computing and Computer-Assisted Intervention: Part II*. Springer-Verlag, 2009.

[LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87. ACM, 1987.

[LCM10] Zhouchen Lin, Minming Chen, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Mathematical Programming*, 2010.

[MLH$^+$15] Jingting Ma, Katharina Lentzen, Jonas Honsdorf, Lin Feng, and Marius Erdt. Statistical shape modeling from gaussian distributed incomplete data for image segmentation. *MICCAI Workshop on Clinical Image-based Procedures: Translational Research in Medical Imaging*, 2015.

[MQR03] Calvin R. Maurer, Jr., Rensheng Qi, and Vijay Raghavan. A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 2003.

[SG02] Mikkel B. Stegmann and David Delgado Gomez. A brief introduction to statistical shape analysis. *Informatics and mathematical modelling, Technical University of Denmark, DTU*, March 2002.

[TB99] Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 1999.

[WGR$^+$09] John Wright, Arvind Ganesh, Shankar Rao, Yigang Peng, and Yi Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In *Advances in Neural Information Processing Systems 22*. Curran Associates, Inc., 2009.

[YY13] Xiaoming Yuan and Junfeng Yang. Sparse and low-rank matrix decomposition via alternating direction methods. *Pacific Journal of Optimization*, 2013.