

Pelvis Runner

Vergleichende Visualisierung Anatomischer Änderungen

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Visual Computing

eingereicht von

Nicolas Grossmann, BSc

Matrikelnummer 01325103

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller

Mitwirkung: Univ.Ass. Renata Raidou, MSc PhD

Wien, 5. August 2019

Nicolas Grossmann

Eduard Gröller

Pelvis Runner

Comparative Visualization of Anatomical Changes

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Visual Computing

by

Nicolas Grossmann, BSc

Registration Number 01325103

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller

Assistance: Univ.Ass. Renata Raidou, MSc PhD

Vienna, 5th August, 2019

Nicolas Grossmann

Eduard Gröller

Erklärung zur Verfassung der Arbeit

Nicolas Grossmann, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 5. August 2019

Nicolas Grossmann

Acknowledgements

More than anyone else, I want to thank my supervisor Renata Raidou for her constructive feedback and her seemingly endless stream of ideas. It was because of her continuous encouragement that I was able to take my first steps towards a scientific career. I also appreciate the guidance of Eduard Gröller, who read my complete thesis, no matter how long the sentences were, and provided me with tons of helpful advice.

Since this thesis is only the peak of my current academic path, I also want to express my gratitude towards all the people that supported me along the way. Without the support of my family, partner, and friends, my studies would have been neither possible nor as enjoyable as they were.

Kurzfassung

Organe, wie die Blase, die Prostata oder das Rektum, zeichnen sich vor allem durch ihre weiche und flexible Struktur aus. Diese ist auch der Grund dafür, dass sie ihre äußere Form laufend verändern, dies geschieht vor allem in Abhängigkeit ihres aktuellen Füllstandes. Klinische Studien deuten darauf hin, dass diese Schwankungen den Erfolg von Strahlungstherapien für Patienten mit Prostatakrebs beeinflussen könnten. In der Praxis werden die täglichen Formveränderungen der zu behandelten Organe zwar durch simple Neuausrichtung der bestrahlten Region berücksichtigt, dennoch könnte eine detaillierte Analyse der exakten Schwankungen eine höhere Genauigkeit der Therapieplanung ermöglichen. Um die Therapieeffekte zu evaluieren, werden sogenannte Kohortenstudien durchgeführt, in welchen eben jene Form- und Positionsschwankungen und ihre jeweiligen Effekte analysiert werden.

Im Laufe dieser Arbeit präsentieren wir eine webbasierte Anwendung, welche in der Lage ist, einen medizinischen Datensatz der Beckenorgane von 24 Patienten mit Prostatakrebs über einen Verlauf von 13 Behandlungseinheiten zu analysieren. Wir verfolgten hierbei zwei Ziele: Einerseits versuchen wir, medizinische Experten bei der Durchführung von Kohortenstudien zu unterstützen und andererseits wollen wir Ärzten eine individuelle Strahlungstherapieplanung auf Patientenbasis ermöglichen. Unsere Anwendung ist in der Lage, große Datensätze von Organen zu verarbeiten, indem wir sie zuerst in einen niedrig-dimensionalen Raum transformieren, in welchem die Formschwankungen analysiert werden können. Dadurch können wir den Anwendern einen Überblick über den gesamten Datensatz zur Verfügung stellen. In einem nächsten Schritt können Teile des Datensatzes ausgewählt und detaillierter analysiert werden. So kann man zum Beispiel die exakten Formveränderungen der Organe von einzelnen Patienten oder Gruppen von Patienten zu untersuchen. Wir präsentieren die Anwendbarkeit unserer Methode anhand zahlreicher Szenarien und evaluieren ihren Erfolg durch Interviews mit zwei Fachexperten. Die Anwendung, die in dieser Arbeit präsentiert wird, kann als erster Schritt gesehen werden, Ärzte bei der Planung von Strahlungstherapieeinheiten zu unterstützen, welche auch die täglichen Formschwankungen von Organen berücksichtigen.

Abstract

Pelvic organs such as the bladder, rectum or prostate have highly variable shapes that change over time, due to their soft and flexible tissue and varying filling. Recent clinical work suggests that these variations might affect the effectiveness of radiation therapy treatment in patients with prostate cancer. Although in clinical practice small correction steps are performed to re-align the treated region if the organs are shifted, a more in-depth understanding and modeling might prove beneficial for the adaptation of the employed treatment planning strategy. To evaluate the viability and to account for the variability in the population of certain treatment strategies, cohort studies are performed analyzing the shape and position variability of pelvic organs.

In this thesis, we propose a web-based tool that is able to analyze a cohort of pelvic organs from 24 patients across 13 treatment instances. Hereby we have two goals: On the one hand, we want to support medical researchers analyzing large groups of patients for their shape variability and the possible correlations to side effects. On the other hand, we want to provide support for medical experts performing individual patient treatment planning. Our tool offers both the option to analyze a large cohort of different organ shapes, by first modeling them in a shape space and then analyzing the shape variations on a per-patient basis. While this first part aims at providing users with an overview of the data, we also give them the option to perform a detailed shape analysis, where we highlight the statistically aggregated shape of a patient or a specified group using a contour variability plot. Finally, we demonstrate several possible usage scenarios for our tool and perform an informal evaluation with two medical experts. Our tool is the first significant step in supporting medical experts in demonstrating the need for adaptation in radiation therapy treatments to account for shape variability.

Contents

Kurzfassung	ix
Abstract	xi
Contents	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Aim of the Work	2
1.2.1 Tasks	3
1.2.2 Requirements	4
1.3 Methodological Approach	4
1.4 Outline of the Thesis	6
2 Clinical Background	7
2.1 Pelvic Organs	7
2.2 Prostate Cancer	7
2.3 Radiation Therapy	9
2.4 Organ Variability	10
2.5 Therapy Workflow	12
2.5.1 Diagnosis	12
2.5.2 Imaging	13
2.5.3 Target Volume and Organ at Risk Segmentation	13
2.5.4 Treatment Planning	14
2.5.5 Treatment Evaluation	14
2.5.6 Treatment Adaptation	15
2.6 Used Data	15
3 Related Work	19
3.1 Classic Approaches	19
3.2 Shape Analysis	21
3.2.1 Descriptor-based Approaches	21
3.2.2 Statistical Approaches	23
3.3 Ensemble Visualization	26
	xiii

3.4	Comparative Visualization	29
3.5	Cohort Analysis	33
4	Visualization Design	39
4.1	Shape Analysis	39
4.1.1	Data Transformation	39
4.1.2	Shape Space	43
4.1.3	Registration	46
4.1.4	Vectorization	48
4.1.5	Analysis & Dimensionality Reduction	54
4.2	Cohort Visualization	59
4.2.1	Overview First	62
4.2.2	Zoom and Filter	72
4.2.3	Distribution on Demand	76
4.2.4	Missing Data on Demand	91
4.3	Shape Visualization	94
4.3.1	Statistical Analysis	96
4.3.2	Shape Display	99
4.3.3	Overview Visualization for Shape Comparison	103
4.3.4	Details-on-Demand Visualization for Shape Comparison	107
5	Implementation	115
5.1	General Structure	115
5.2	MeVisLab	116
5.3	MATLAB	117
5.4	C# Server	119
5.5	User Interface	119
5.5.1	D3.js	120
5.5.2	three.js	121
6	Results & Evaluation	125
6.1	Usage Scenarios	125
6.1.1	Cohort Exploration	125
6.1.2	Single Organ Shape Exploration	127
6.1.3	Multiple Organ Shape Exploration	127
6.1.4	Shape Type Identification	130
6.1.5	Individual Patient Analysis	133
6.1.6	Toxicity Factor Analysis	136
6.2	Evaluation	139
7	Conclusions & Future Work	141
7.1	Summary	141
7.2	Limitations & Future Work	143

List of Figures	147
List of Tables	155
Bibliography	157

Introduction

What is out of the common is usually
a guide rather than a hindrance.

Sherlock Holmes, A Study in Scarlet
Sir Arthur Conan Doyle

This chapter aims at giving a short overview of the most important concepts presented in this thesis. We focus in particular on the motivation, which drove us as well as our goal. The chapter also includes a short guide explaining the structure of the following thesis.

1.1 Motivation

Prostate cancer is one of the leading cancer types in men, accounting for one out of four of all new cases of cancer in Austrian men [AUS19a]. Radiation therapy using an external beam is an often employed approach in combating it [WL15]. It aims at destroying cancer cells in a non-invasive way through high radiation doses, and therefore destroying the tumor inside the body. This method requires an extensive amount of planning to calculate precisely where the tumor is placed and how to treat it effectively. During the course of the radiation treatment not only the tumor is damaged but also the surrounding healthy tissue, leading to several side effects commonly known as toxicity. While modern treatment techniques like Intensity Modulated Radiation Therapy (IMRT) and Volumetric Arc Therapy (VMAT) allow for more precise treatments, there are also other factors leading to negative side effects [WL15].

Several recent works suggest that the shape changes and variations of patients can lead to increased radiation doses being delivered to healthy organs like the bladder or the rectum [CMMH⁺17]. Although a simple correction in terms of alignment is made before each

treatment to align the treated area with the tumor, this seems often not to be sufficient. Adapting the current workflow to encompass even more complex changes in organ shape could allow for an even better treatment precision with less damage to other organs [THLM⁺13]. To achieve this, medical experts, which we define in the context of this work as physicists working on robust treatment plans, require a better understanding of both the general shape and position variability of certain organs as well as the variability of individual patients. To evaluate the overall robustness of the used treatment plans, individual patient exploration might not be sufficient, therefore a general analysis of the cohort is also necessary.

Currently, the exploration of shape variations is only done in clinical research and is not integrated into clinical practice due to the induced cost and time requirements. By providing a simple tool to estimate and visualize the shape variability of pelvic organs, we want to support clinical researchers in their task to demonstrate the significance of adaptation of the dose plan. To this end, Visual Analytics can be used in the treatment pipeline to enhance the precision of treatment plans.

We can condense the main research question of this work as: *Which strategies can we employ to analyze and visualize the anatomical changes of organs to support medical experts in their research and treatment planning?*

1.2 Aim of the Work

The aim of this work is to implement a tool to be used by medical experts to analyze a set of segmented pelvic organs of multiple patients across the whole radiation therapy treatment procedure. For this purpose we analyzed what recent works in this field tried to achieve and transformed their approaches into seven different research questions, which a medical expert might want to answer:

- (Q1) Which patients or organs experience high amounts of variability inside the cohort?
- (Q2) What are common types of shapes and variations inside the cohort?
- (Q3) Which parts of the cohort are missing?
- (Q4) How do organs vary in shape and position between patients or across treatment instances?
- (Q5) Do certain parts of an organ experience especially high variations?
- (Q6) Are there overlaps between different organs inside the cohort?
- (Q7) How do certain organs differ in shape between patients?

1.2.1 Tasks

To enable researches to find answers to the previously defined questions we defined two basic tasks that our final tool aims at answering.

- (T1) Exploration and analysis of the shape variability and types across the cohort (Q1-Q3)
- (T2) Exploration and analysis of local shape and position variations for individual patients and organs (Q4-Q7)

(T1) Exploration and Analysis of the Shape Variability and Types Across the Cohort

With the provided organ shape data stretching across patients and timesteps, the resulting cohort can provide deeper insights than the analysis of only single patients. The main information researchers are interested in extracting from the provided data is the amount of change certain organs are subject to both between all patients but also over time for single patients. While the definition of statistic variability is precisely defined for sets of numbers, it is not as readily extensible to segmented organ shapes [MWK14]. Therefore, we need to find a way to calculate a simple representation (descriptor) for each organ in the cohort that allows us to measure the similarity between organs. With such a combination of descriptor and metric, we would be able to find groups of similar organs and estimate their variability.

After the measures are calculated, we need to visualize them for the whole cohort. Datasets including patients and timesteps have a quadratic growth in size as more patients are added. Therefore exploration and analysis must be possible in such a way that users are both able to quickly gain an overview, as well as identifying patients or organs with high variability.

For analyzing groups of similar patients and organs, appropriate quality measures need to be provided to inform users how common the group is and how large the inner group variations and missing data percentage is.

(T2) Exploration and Analysis of Local Shape and Position Variations for Individual Patients and Organs

After interesting partitions of the cohort (i.e. patients or subgroups) have been identified, users might want to perform a more detailed exploration of the individual patients and organs. While the previous step focused on showing the average variability for each organ for easy exploration, the detailed exploration should be able to show the variability of a group of organs in 3D. This should allow users to understand which regions of certain organs are prone to variations and how large these differences are. The user should be able to differentiate between changes in position and shape.

Segmentations of certain organs are often manual and can include margins to account for motion and other inaccuracies. Therefore, two adjacent segmentations might overlap. Being able to see how large these overlaps are and how they are influenced by organ shape variations can provide valuable insights into the causes of induced toxicity.

While the analysis of the variability of a partition might provide knowledge about individual variations and behaviors, it should also be possible for an expert to compare multiple partitions to each other, providing knowledge about how the same part may have different variations.

1.2.2 Requirements

The result of this work should not only be able to fulfill the previously defined tasks but it also has to achieve some requirements ensuring its usability as an application. The extent to which these requirements are achieved is discussed in Chapter 6.

Scalability

The underlying approach should be able to process large numbers of patients, timesteps, and organs efficiently. Users should be able to easily add the data of new patients to the full cohort. The resulting visualization has to be able to provide interactive responses to the user's actions to facilitate exploration.

Generalizability

While the main focus of the tool is to provide medical experts with information about pelvic organs, the basic algorithm used for the variability calculation should be independent of the underlying use case. This should ensure an easy extendability of the tool to other parts of the body or even other completely different use cases.

Usability

The final tool should be provided in a form that users can employ it without the need for an extensive setup process. Additionally, the tool and the information it provides needs to be understandable as well as usable by medical experts, who might have varying degrees of experience with interactive visualizations and Visual Analytics.

1.3 Methodological Approach

The initial basis for all further tasks in this thesis is the development of an approach to quantify the similarity of volumetric shapes and estimate the variability. While using contours to visualize shapes and their differences works for a small number of compared items it does not scale well and therefore appropriate abstraction methods need to be found to allow for the comparison of large shape collections. To achieve this an

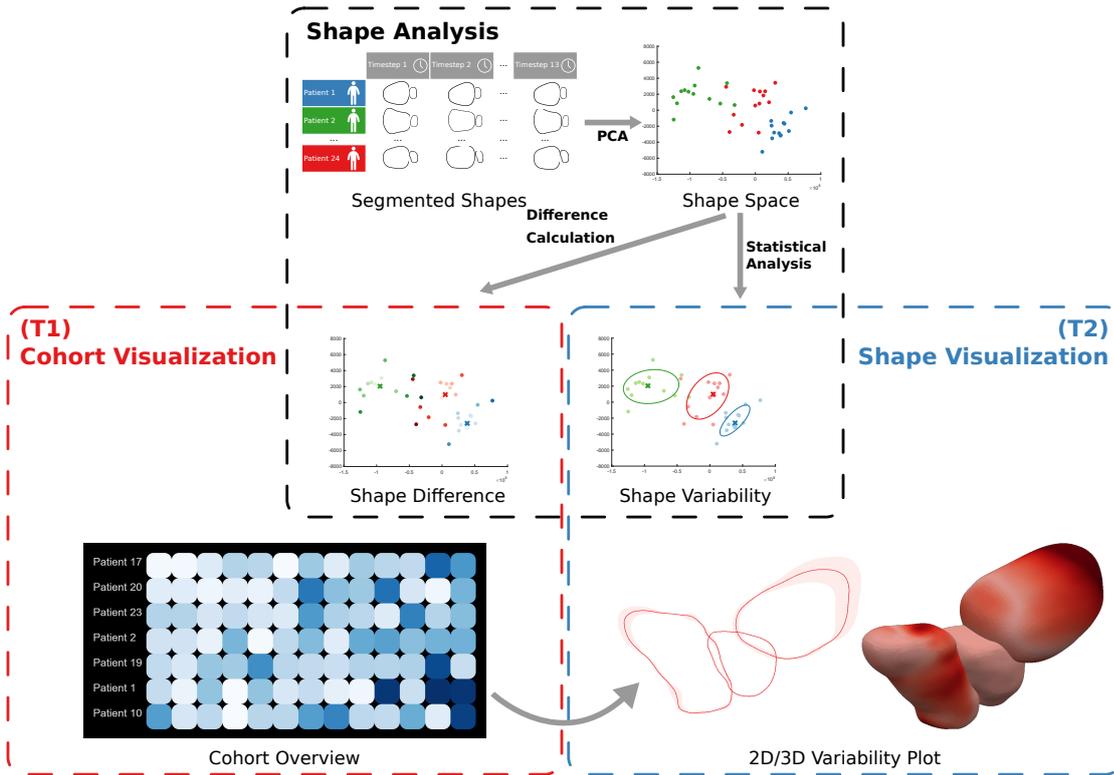


Figure 1.1: Overview of our general approach.

aggregation approach previously only used for weather data [FKRW16] is employed to the organs in the cohort.

For the first task, the resulting low-dimensional embedding of each organ is then used to calculate the variability on a per-patient basis and visualized over the whole cohort. Using the provided 2D representation of each organ, similar organs can be grouped using clustering. The provided information is visualized for each data point in a trellis shaped visualization. To comply with Shneidermann’s Mantra *“Overview first, zoom and filter, then details-on-demand.”* [Shn03] at any given time, the data is grouped with detailed patient information only being visible if the user wishes.

The next step was to approach the second task by providing the user with the option to drill down to groups of patients from the cohort and perform a detailed inspection of the organ variations. This is achieved by reconstructing the initial 3D objects from their low-dimensional embeddings. By sampling the embedding space for both the median and the standard deviation, one can reconstruct 3D objects representing the shape variations. Using these objects a contour boxplot is created for the selection.

The main contribution of this thesis is a visual tool for medical researchers working on the design of robust treatment planning methods who are trying to understand how organ

shapes vary across multiple patients and timesteps. To achieve this we used the pelvic organ data of 24 different patients with prostate cancer across 13 treatment instances and visualized it using an aggregation approach for 3D volumes. The final tool not only allows users to explore a large cohort in a quick and easy way, but also enables an in-depth exploration of individual patients.

1.4 Outline of the Thesis

The following chapters are structured as follows: In Chapter 2 we explain the medical background on pelvic organs their inherent variability with an additional focus on cancer. Of all possible treatment methods for cancer radiation therapy is discussed in detail in all its involved stages, after which the chapter concludes in an explanation of the prostate cancer data used as a basis for this thesis. In the following Chapter 3 we focus on how recent works in medicine try to analyze shape variations in organs, hereby we also look at current advances in shape analysis. Another important research area we reflect upon is the field of ensemble and comparative visualization. After the presentation of the works that influenced this thesis, the next step was to take a look at the problems at hand and discuss possible solutions for them in Chapter 4. Starting with ways to analyze shapes, we go on with the visualization of large cohorts of data and conclude with methods to provide a detailed and comparative analysis of local shape changes. Having decided which of the proposed methods work best for our case, we proceed in Chapter 5 with an explanation about the concrete implementation that resulted in our proposed tool. Chapter 6 looks at the resulting tool and investigates possible usage scenarios and findings derived from the data. During our work, we also performed an informal evaluation with three medical experts, which is also described in this chapter, together with a critical analysis of our results. Finally, we conclude our work in Chapter 7 where we not only reiterate briefly about the performed work, but also offer possible ways of advancing the topic in future works.

Clinical Background

While the general approach of analyzing variability in large volumetric datasets can be interesting for many different applications, the use case this thesis is focusing on is the visualization of anatomical changes. The goal of this chapter is therefore to provide some background information on the medical reasons for organ shape variations as well as showing reasons why their understanding can be beneficial for medical experts.

2.1 Pelvic Organs

The part of the body, which this work largely focuses on is the male pelvis and the organs it contains. Understanding the rough anatomical structure of the contained organs is important to comprehend how different parts of the body interact and where certain problems might arise due to illnesses or treatments.

As part of the urinary system (which can be seen in Figure 2.1), the bladder is a muscular organ that collects the urine. When the urine is disposed of, it leaves the bladder via the urethra, which can be seen as a tube. Wrapping around the urethra and positioned just below the bladder is the prostate. It is an essential component of the male reproductive system, responsible for parts of the ejaculate production and its transport during ejaculation. On top of the bladder, the seminal vesicles are placed, which are two glands producing another component needed for semen. While the bladder can be considered in front of the prostate and seminal vesicles, in their back the last part of the colon is lying. Its primary objective is to act as a storage facility for human feces before they are finally transposed [HWS⁺83].

2.2 Prostate Cancer

Prostate cancer generally describes malignant tissue forming in the prostate. Statistically speaking prostate cancer is the most common type of cancer of men in developed countries,

2. CLINICAL BACKGROUND

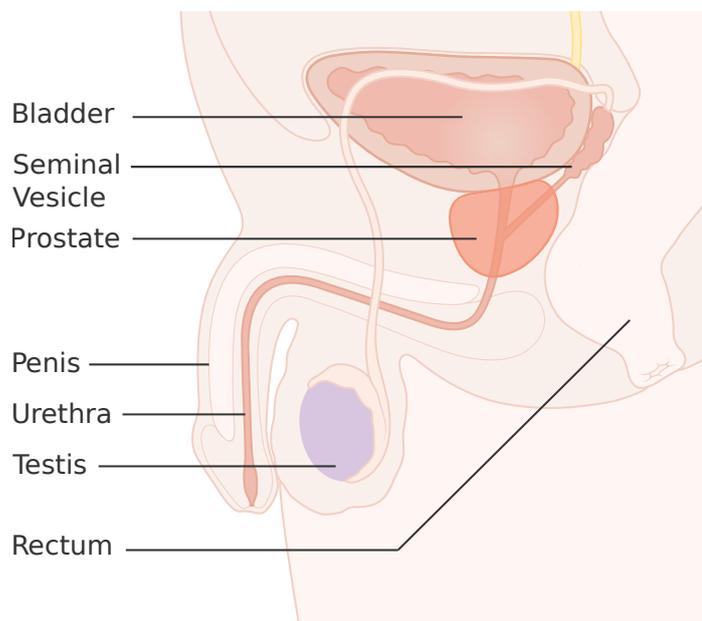


Figure 2.1: Pelvic Organs of the Male Body (Image adapted from Cancer Research UK [UK19])

Organ	Incidence
<i>Pelvic Organs</i>	26%
Prostate	24%
Urinary Tract	2%
<i>Digestive Organs</i>	25%
Colon	8%
Rectum	4%
...	13%
<i>Respiratory Organs</i>	15%
Lung	13%
...	2%

Table 2.1: Cancer incidence rate in Austrian men in 2016. Prostate cancer is the leading type of cancer [AUS19a].

according to recent statistics. As of 2016 around 63.415 cases were reported in Austria, with 5.245 being new cases out of the total population of 8.6 million people [AUS19a]. The overall incidence rate for prostate cancer lies at around 24% of all new cancer cases in men leading in front of lung and all kinds of digestive cancers (as seen in Table 2.1). The risk rises sharply after the age of 50 and reaches its peak at around 70 and 80 [SWD08]. The cumulative chance for a man to suffer from prostate cancer at any time up to the age of 75 lies at around 8.4%. While the general survival rate of prostate cancer is at around 90% after 10 years, this still amounted to 1.225 men dying because of it in 2016 [AUS19b].

Because the early stages of prostate cancer do not lead to any symptoms, early screening measures are used. After a physical rectal examination, the second most important indicators for prostate-related diseases is the amount of prostate-specific antigen (PSA) present in the blood [moy]. If both of these methods seem to indicate cancer a biopsy is performed to take samples of the prostate as final proof. Due to the invasiveness of the method, this is only performed in cases with clear symptoms [EST09].

2.3 Radiation Therapy

For patients suffering from prostate cancer, the right choice of treatment is essential. This decision is influenced by the age of the patient and its remaining live time as well as the previous medical history. Often the choice is made not to perform active treatment and watch the progression. One example of such a case would be a patient who is not expected to develop symptoms in his remaining life [WSM⁺04].

If the choice for active treatment is made, due to the aggressiveness of the tumor or other circumstances, several different treatment options are available [HAB⁺08]. The most common being:

- **Surgery** - The prostate is removed either in parts or as a whole.
- **Radiation Therapy** - The tumorous tissue is destroyed through the application of high radiation doses. The dose can either be applied via an external beam or by using radioactive beads that are implemented.
- **Hormonal Therapy** - The production of certain hormones is decreased, which may slow down the growth of certain tumors (e.g. prostate).
- **Chemotherapy** - The reproduction rate of fast-growing cells in the body is decreased, which also aims at slowing down tumor growth, but due to it attacking all cells healthy ones are also damaged.
- **Combined Therapy** - Often not only a single therapeutic approach is applied but multiple ones to achieve better treatment results.

The treatment method this work focuses on is radiotherapy using an external beam (EBRT). As mentioned before the main goal of radiotherapy is to damage the tumor tissue using high radiation doses. The radiation is applied in the form of several beams around the body intersecting at the tumor, which in turn leads to a high applied dose at the targeted area and a lower dose at the surrounding tissue [ZYF⁺08]. Recent techniques called Intensity Modulated Radiation Therapy (IMRT) and Volumetric Arc Therapy (VMAT) are improvements to this method that use specifically shaped radiation beams and a moving beam to further limit the damage to healthy tissue [QLL⁺12].

2.4 Organ Variability

Generally speaking, the inner anatomy of every human has unique variations. Roughly they can be divided into three different groups:

- Naturally Occurring Variations (i.e. across individuals)
- Pathological Changes (i.e. due to the impact of disease)
- Day-to-Day Changes (i.e. due to filling changes)

Naturally occurring variations might range from simple size and shape differences to different deforming behavior, to varying connectivity patterns of blood vessels. These variations have been thoroughly researched by medical experts and are usually dealt with using imaging data to comprehend each patient's inner structure [RBL⁺09, MMNG15]. Changes induced due to certain diseases like tumor growth are dealt with in a similar way and are explored by analyzing imaging data for each patient individually [AGL⁺13]. Apart from these natural and pathological shape differences, most organs also change on a day-to-day basis, due to some internal circumstances, because all of the aforementioned organs are generally made of soft deformable tissue and are movable up to a certain degree inside the body. As both bladder and rectum are hollow inside and act as temporary storage rooms which are regularly filled and emptied, their general shape largely varies based on these circumstances (e.g. the bladder's volume might range from 100 ml up to 700 ml) [MSD03]. The general amount of flexibility differs from organ to organ and person to person. Large size variations of one organ affect the surrounding ones as well, for example, an enlarged bladder might push away the prostate or seminal vesicles from their usual position or an enlarged prostate (due to some illnesses) might restrict the flow through the urethrae. While the static shape differences are well researched, the dynamic ones occurring each day are only well known on a general basis and not for individual patients.

During the radiation treatment, a patient might experience a wide range of acute side effects linked as an immediate response to the radiation damage like fatigue, increased urinary problems, and digestion problems [FVRS09]. These symptoms should fade away after the treatment procedure is finished. Apart from these short-lived problems, there is

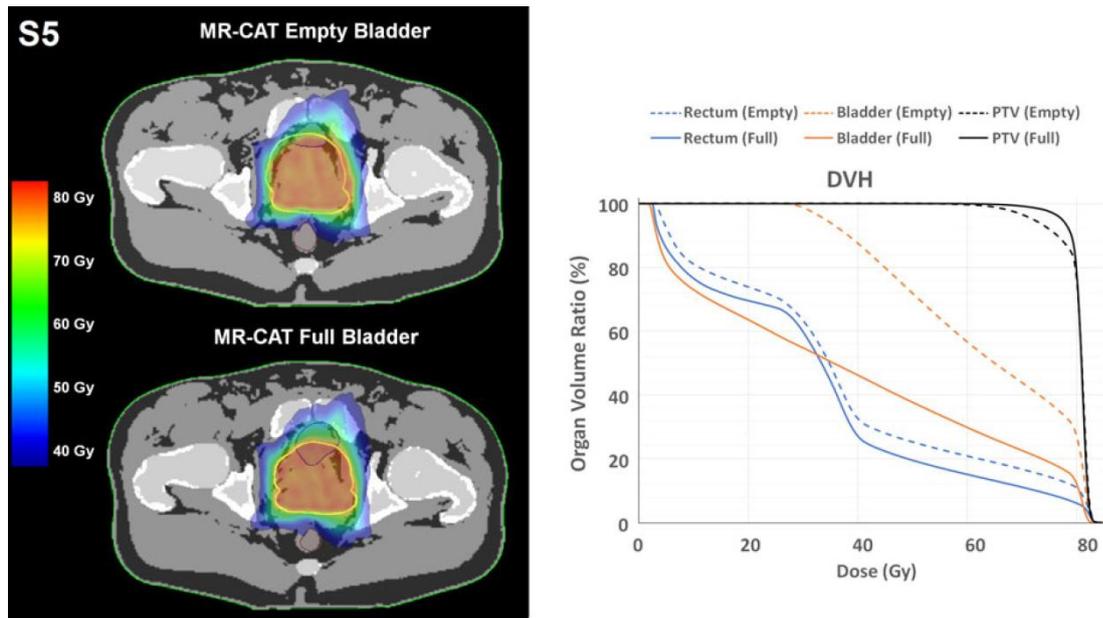


Figure 2.2: Example of a simulated radiation therapy treatment plan applied to a patient with a full and an empty bladder. The left part shows the spatial distribution of the radiation applied to the prostate, while the right part shows the dose-volume-histogram explaining how much each surrounding organ is radiated. It also shows that by having a different bladder filling than the one at the time of the treatment planning the empty bladder receives a higher radiation dose. (Image from *Nejad-Davarani et al.* [NDSM⁺19])

also the chance that a patient might develop long-term symptoms, like rectal bleeding, incontinence, or worse [DKN⁺99, FVRS09]. The reason for this is that often parts of healthy organs are irradiated beyond a certain threshold and suffer therefore permanent damage. This effect is called toxicity and is often associated with the aforementioned longterm symptoms.

While a proper treatment plan aims to minimize the risk for inducing toxicity, recent works suggest a link between the movability and deformability of an organ and its risk of suffering from severe after-effects [CMMH⁺17]. Organs like the bladder whose position and shape varies strongly on a daily basis due to its own and the bowel's filling, are especially prone to this effect [VYM⁺10]. The effect of this can be seen in Figure 2.2, where a treatment, which was initially created for a patient with a full bladder is re-evaluated for the same patient with an empty bladder. It shows how drastically the received dose of the bladder increases, due to this shape change [NDSM⁺19]. Other works also evaluated these effects on other organs like the rectum and came to similar conclusions [FVRS09].

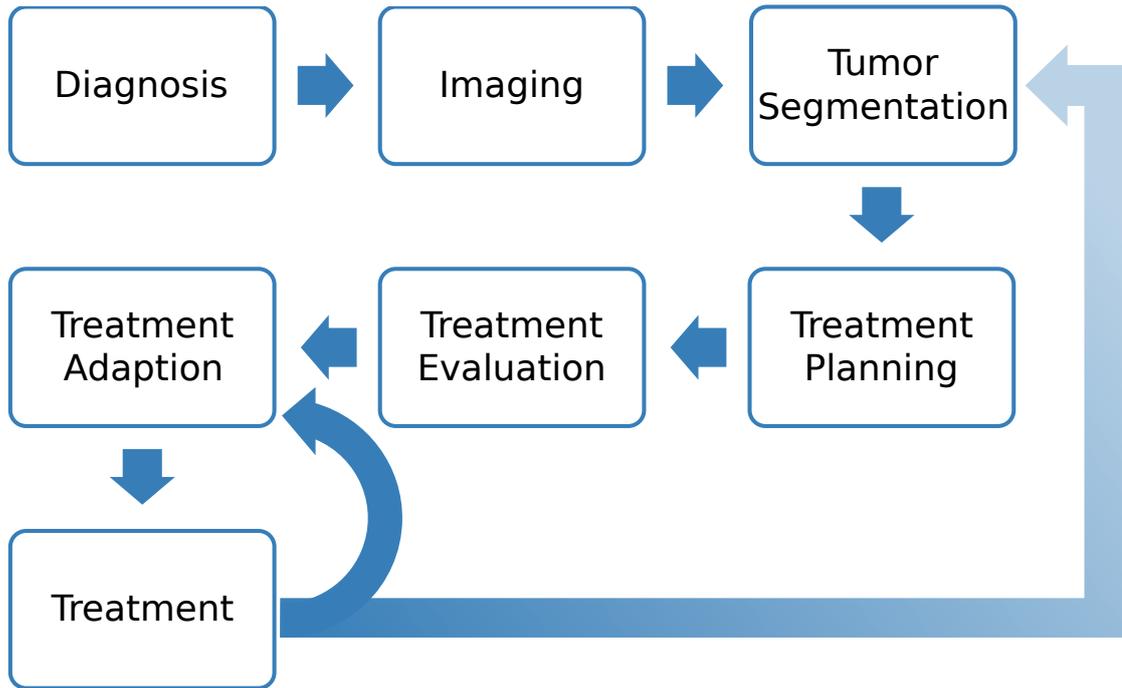


Figure 2.3: Current radiation therapy workflow as described in the work of *Schlachter et al.* [SRM⁺19]. While the current process includes a per treatment adaptation, in terms of position and alignment, future techniques aim at having an even preciser radiated area by re-evaluating a larger part of the workflow for each treatment [SSB⁺05].

2.5 Therapy Workflow

Radiation therapy follows a simple step-by-step workflow, which involves several different people and starts with the diagnosis of a patient with cancer and ends in a post-treatment examination of the patient. The overall goal of this workflow is to ensure the most effective treatment with the least possible side effects. As treatment procedures evolve to be more precise and effective the treatment workflow is also updated continuously. As it is important to understand the processes leading to a patient's treatment and what actions medical experts perform in the course of this planning process, we will present a simplified version of the workflow, ranging from the initial diagnosis up to the treatment of the patient, similar to the ones described in the work of Aselmaa et al. [AGL⁺13] and Schlachter et al. [SRM⁺19].

2.5.1 Diagnosis

The initial diagnosis is often made not by an oncologist but instead by various other physicians, which the patient visits due to some health complaints, like headaches, difficulties during urination, or breathing problems. Follow up tests could include different imaging procedures and blood tests. Ideally, the consulted physician detects

cancer during the testing procedures and refers the patient to colleague specialized on cancer [WL15]. During this transition period, it is of great importance that the medical history and the acquired data is properly shared between the involved parties to ensure proper treatment.

2.5.2 Imaging

After the initial diagnose is made, the next big step is to acquire detailed information about the tumor. For this, imaging data is gathered on the diseased region. Techniques used for this may include Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) to show the internal structure or functional techniques like Positron Emission Tomography (PET) showing the flow of certain markers [NDSM⁺19]. Which modality should be used, largely depends on the type of tumor and its surrounding tissue, the reason for this being that each modality has different up- and downsides, but in general CT images are always used as a basis and then complemented using other modalities.

If a multitude of modalities is used they first need to be linked together, which is done using a process called image fusion, where multiple images of different or the same modality are registered to one another [WTK⁺06]. The goal of this step is to overlay the different images correctly accounting for factors like different image resolutions and patient movement. The CT images often serve in this context as the base image to which the others are aligned [AGL⁺13].

2.5.3 Target Volume and Organ at Risk Segmentation

The next step is to use the images to segment the tumor and its surrounding organs, which might suffer during the treatment. In this process medical experts use the given images to either automatically, semi-automatically or even manually select which parts of the image belong to either the tumor or are parts of surrounding organs [PLV⁺07]. The manual process involves the expert going through the image data slice-by-slice and manually drawing the structures to be segmented [WL15].

Together with the tumor, its surrounding organs are also segmented. They are at risk of being radiated and could suffer under toxicity. Precise knowledge of the position and extent of these organs at risk (OAR) is an essential part of the treatment planning, as many modern tools can use this information to optimize the given dose.

Following this, the tumor is segmented. In this context, it is important to differentiate between the different types of volumes involved in this process (as seen in Figure 2.4) [JYW99, WL15]:

- GTV (Gross Tumor Volume) - Visible Tumor Tissue (as seen on a macroscopic view in the imaging data)
- CTV (Clinical Target Volume) - GTV + Presumed Tumor Tissue (cannot be seen on the images - microscopic - created by a safety margin around the GTV)

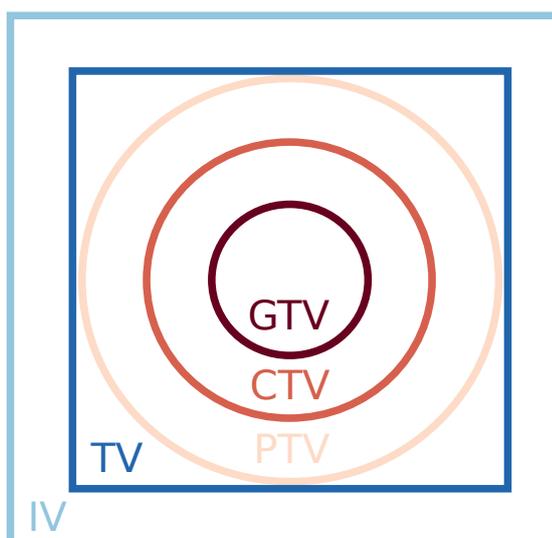


Figure 2.4: Overview of the volume definitions used in radiation therapy planning.

- PTV (Planning Target Volume) - CTV + Safety Margin (to account for organ movement and uncertainties)
- TV (Treated Volume) - Treated Volume with a sufficient radiation dose
- IV (Irradiated Volume) - TV + Affected Healthy Tissue

2.5.4 Treatment Planning

After segmentation, a treatment plan is calculated in a simulation program. The goal of this stage is to provide optimal irradiation of the tumor, while minimizing the damage to the surrounding tissue, as this may lead to acute or late complications as organs like the rectum or bladder are damaged. The planned dose is not administered at once, but is instead split up over several weeks. This allows the regeneration of healthy tissue and prohibits tumor growth [WL15].

The creation of this plan can either be done in a direct way where the user adjusts the beams and evaluates how each part of the body is affected or inversely where the user only defines the area to be treated and the beam positions are calculated.

2.5.5 Treatment Evaluation

Due to the importance of the initial treatment plan, multiple different medical experts evaluate it before it is employed. To do this, different methods have been developed. Two examples of these can be seen in Figure 2.2. Using a spatial view, the experts can directly see how the dose would affect the tumor and its surrounding organs for the given

point in time, showing precisely which parts of each organ are radiated [NDSM⁺19]. In a more abstract way, dose-volume-histograms can show how much radiation is received by how much of each organ, which allows the user to quickly identify organs in risk of induced toxicity [WL15].

2.5.6 Treatment Adaptation

The standard treatment procedure is to use the initially generated plan as a basis for the following treatments. To allow this procedure, it is important that the setting of the initial planning is reproduced during the treatment, for example, prostate treatments are commonly performed with a full bladder [WL15]. There are still many different factors that might lead to shape and position variations throughout the treatment, therefore small adaptations need to be applied to the target volume. This process has to be done continuously to account even for small anatomical movements. These mostly simple translations cannot account for more complex deformations [CMMH⁺17].

The value of adjusting the target volume in prostate cancer therapy on a per treatment basis was highlighted by several recent works, which estimated that a change of bladder filling also leads to an increase in radiation applied to the organ, which in turn leads to increased organ damage and side effects [VYM⁺10]. Recent years lead to many different improvements in radiation therapy, while the machines applying the dose got more and more precise as time progressed, the planning processes also adapted and moved towards a more personalized treatment [SSB⁺05]. Future plans might go towards a 4D treatment plan where the variability and movement of the patient's organs are taken into greater consideration to specify the radiated area, possibly on a per-treatment basis [THLM⁺13]. To achieve this adequate tools need to be integrated into the planning pipeline that allows physicians to inspect and analyze the organ variability.

2.6 Used Data

For this work, we were provided with the data of 24 different patients undergoing radiation therapy for prostate cancer. Around half of the patients, the treated area was restricted to the prostate and seminal vesicles, while for the other half it also included larger portions of the pelvis to include the surrounding lymph nodes. The initial treatment plan was calculated for patients with an empty rectum and full bladder.

Each day of their treatment the patients were instructed to have roughly the same bladder filling. Before each treatment, a cone beam computed tomography was done to align the patient's treatment volume using rigid transformations (only translation).

The provided data includes 13 treatment steps for each patient with the first five of them being the results of the treatments 1-5 and the other ones from every fifth treatment (6, 11, 16, ...) [CMMH⁺17]. While we were not provided with the CT data of each of these timesteps, we were instead given the segmented organs and treatment volumes in the form of contour lines. As mentioned before, medical experts manually delineate the

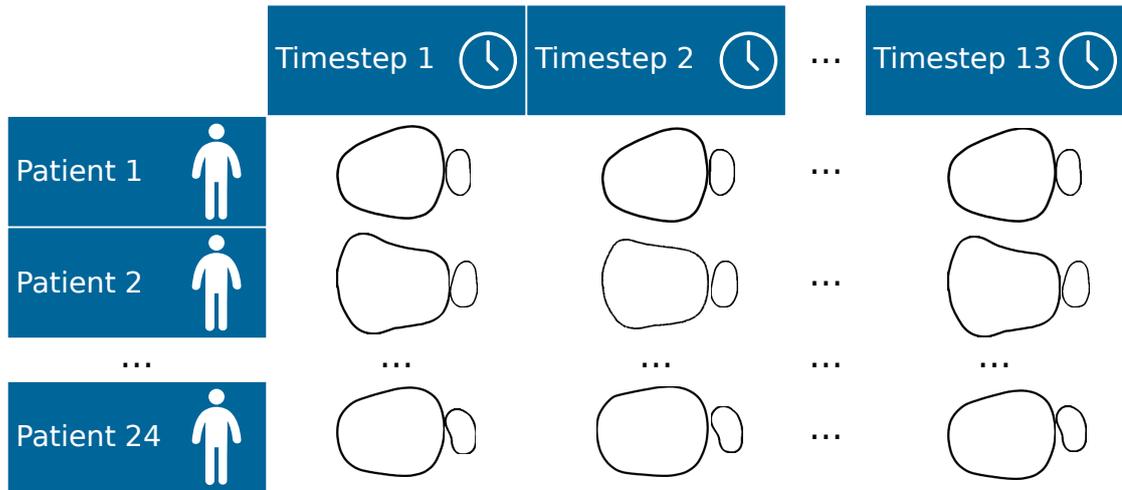
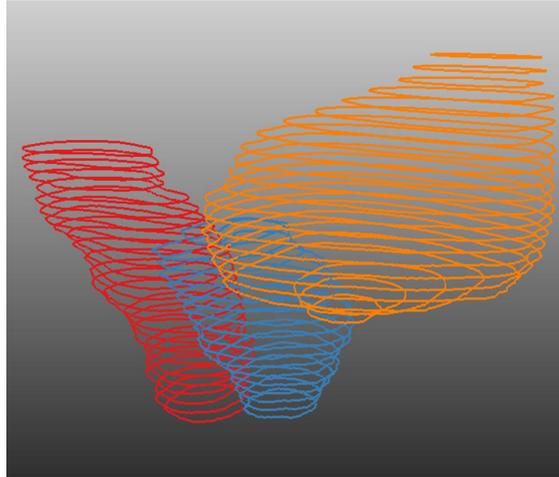


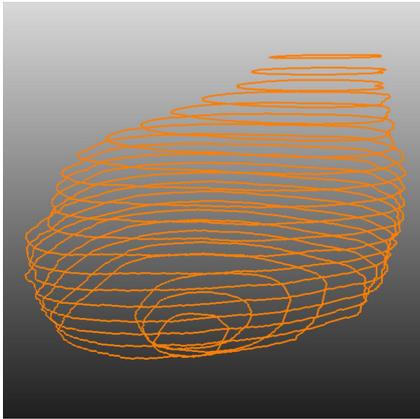
Figure 2.5: General structure of the cohort data used in this work. The segmentation data of 24 patients is used, with each of them having 13 recorded timesteps throughout their treatment. For each point in the cohort multiple different organs are segmented, including bladder, rectum and most importantly the radiated volume.

tumor and organs at risk to calculate a treatment plan. As the data was previously used in a study to determine the effect of bladder changes on the received irradiation, this segmentation process was also done for all timesteps.

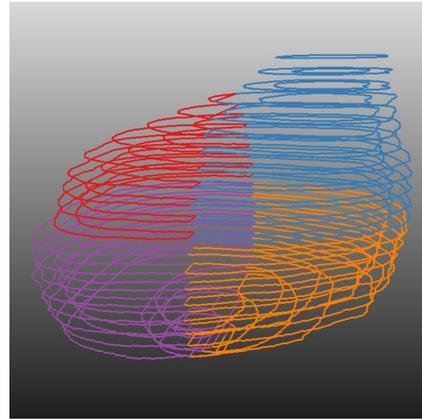
The final data includes the contours of the surrounding organs and the treatment volume of 24 patients with each of them having at least 13 recorded treatment steps. The included organs are bladder and rectum for each one of them, as well in some cases the penile bulb and femur heads. The bladder itself was also split into four different regions (Anterior vs. Posterior and Apex vs. Base). Apart from these organs, the tumor and treatment regions were defined by contours for the CTVs and several different types of PTVs. The 24 patients used in this work were selected from the full cohort of 29 patients as they provided the most consistently labeled organs (as seen and explained in Table 2.2). Selecting only patients with a correctly segmented bladder, rectum and either a combined prostate seminal vesicle (SV) segmentation or a CTV leaves us with 24 patients, which were further used in this work.



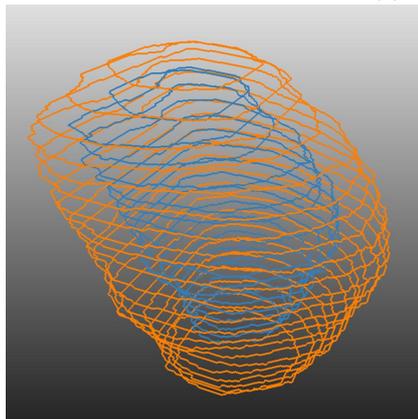
(a) Segmented Organs at Risk (red = Rectum, orange = Bladder) and Prostate (blue)



(b) Segmented Bladder



(c) Four Subparts of the Bladder



(d) Prostate CTV and PTV

Figure 2.6: Examples of the contours present in the provided data

	Occurrence	Percent
Bladder	29	100%
Rectum	29	100%
Fem Head	29	100%
CTV	13	45%
ProstateSV	11	38%
Prostate & SV	2	7%
PTV	29	100%

Table 2.2: Number of segmented parts present in the data per patient ($n = 29$). All patients have a segmented bladder, rectum and femur heads. 26 Patients have either a CTV or a segmented prostate with seminal vesicles (in two cases the prostate and seminal vesicles are separately segmented). Although each patient has some sort of PTV. The type of PTV, defined by the radiation intensity, strongly varies from patient to patient. Each one of them has a unique therapy.

Related Work

The topic of this thesis touches upon a wide range of related works. These works either relate to the presented research questions directly or were used as a basis to develop new solutions for problems that to the best of our knowledge were not presented in any other works.

The last years have seen some growing interest in research regarding the variations and shape differences in pelvic organs, especially in terms of radiation therapy for prostate cancer. Throughout this chapter, we will analyze in detail how these thematically closely related works differ from ours and how we intended to improve upon them.

Apart from these specific visualization topics, we will also take a look at a more general topic that deals with the analysis and representation of volumetric shapes and how to estimate differences and variations between them. The volumetric dataset stretches across both patients and timesteps, their large size can lead to scalability problems and techniques found in cohort and ensemble visualization need to be employed. To allow exploration of shape differences in their original space, methods for comparative visualization need to be used, to gain an understanding of how different patients vary in size and shape. We will discuss how this thesis draws upon these previously done works. The basic workflow and how it touches upon these different fields can be seen in Figure 3.1.

3.1 Classic Approaches

To facilitate understanding of the daily occurring shape variations in pelvic organs and especially how they may have an effect on the side effects of radiation therapy, some works were already performed by medical experts. An example of a work performed by a medical professional is the one of Nejad-Davarani et al. [NDSM⁺19], who analyzed the impact of volume changes to the irradiation of the bladder by simulating a treatment

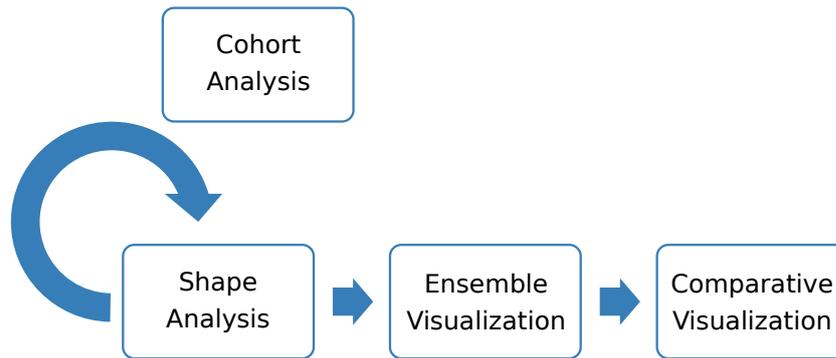


Figure 3.1: Overview of the general topics touched upon in this thesis and how they interact with each other. Starting with an overview visualization of the cohort, interesting shapes are selected and analyzed. Groups of shapes are then treated as an ensemble and need to be visualized jointly. Finally one has to be able to compare multiple different ensembles to each other.

plan for both empty and full bladders of multiple patients. They evaluated the applied dose by using Dose-Volume Histograms (DVHs), which are a representation that shows which percentage of the organ tissue is radiated to which extent in percent. An example of their work can be seen in Figure 2.2, where the applied dose is shown for the PTV, bladder, and rectum. The plot shows that for the full bladder the PTV receives the maximal dose nearly across its whole volume, while the bladder only receives such a high dose for 15% of its volume. This type of visualization is often used during the treatment process as an evaluation tool. However, it is abstracted from the spatial domain and cannot provide any information about the current shape of the involved organs, meaning they only show that a certain dose is administered, but not where. Due to this depiction, the representation of shape changes can only be visualized by showing how the radiation dose changes.

Casares et al. [CMMH⁺17] used another approach to analyze how bladder shape changes affect the administered dose. Although their work did not aim at implementing an analysis tool they proposed some techniques to show the variability of certain organs and their influence on the radiation dose distribution during the treatment. They visualized the shape changes of organs like the bladder over several treatment steps on a per-patient basis by drawing the segmented slices superimposed. In combination with the underlying treatment plan, it can be seen which parts of the bladder intersect with the treatment volume. An example of this can be seen in Figure 3.2. While this approach works well for single patients, it is not extensible to larger cohorts of data and multiple patients.

These previous works of medical experts give insight into what kind of visualizations are common to them, e.g. slice representations for all three axes. Still, they currently do not have any tool for analyzing more than one patient at a time, without any means to look through the cohort patients with certain kinds of variations. While the simple approaches

used to visualize local shape changes are useful when comparing low numbers of patients they do not scale well for larger groups, for example overlaying more segmentation lines would only result in an indistinguishable clutter.

3.2 Shape Analysis

To explore the possible shape variations of organs and to quantify their extent, it is important to use appropriate representation methods and measures, for example, by embedding shapes in a common space. The chosen approach must allow the analysis of large shape cohorts, which in our application are volumes of different organs, while still maintaining interactive frame rates. In this context, the field of shape analysis is of relevance. Due to the explorative nature of tasks analyzing organ variations, visual analytics solutions are often used in this field, where a user is not only shown one visualization for the whole cohort but where he can instead incrementally search through the data to find parts of interest, e.g. patients with especially large variations.

3.2.1 Descriptor-based Approaches

During the last year, several papers were published specifically focused on analyzing topics related to the variability of pelvic organs in patients with prostate cancer. A common trait of them is their use of descriptors calculated based on the shape of the used organs.

One of these tools is *Bladder Runner* by Raidou et al. [RCMA⁺18], which aims at providing information about the amount of irradiation applied to the bladder across the treatment for a patient. For finding similarly shaped organs they use a 14D vector consisting of several characteristics calculated from the organ shape [PI97]. This vector includes the eigenvalues of the inertia tensor, volume, convexity, and several other values. The resulting vector's dimension is reduced by applying a t-Distributed Stochastic Neighbor Embedding (t-SNE) [MH08], after which a clustering [CM02] for similar shapes is performed. The information is then represented in a glyph-based view to show the whole cohort. The variations are visualized by aggregating the organ volumes and drawing isolines at their coverage percentiles, indicating which regions are covered 100%, 75%, etc. (as seen in Figure 3.3a) of the time. While this already shows how strong a certain patient's shape variations are, it allows them to calculate the amount of irradiation received by the bladder. Overall the tool offers a lot of different approaches in regards to our research question, especially as it enables the analysis of the cohort and shape variability at the same time. The main difference is that the aim of this work is to focus on the analysis of more than one organ. We also aim at including the possibility of missing elements in the data, and overlaps between organs. The current approach for analyzing shape variations is not easily expendable to compare more than one or two clusters to each other.

Another tool using a similar approach was implemented by Reiter et al. [RBGR18].

3. RELATED WORK

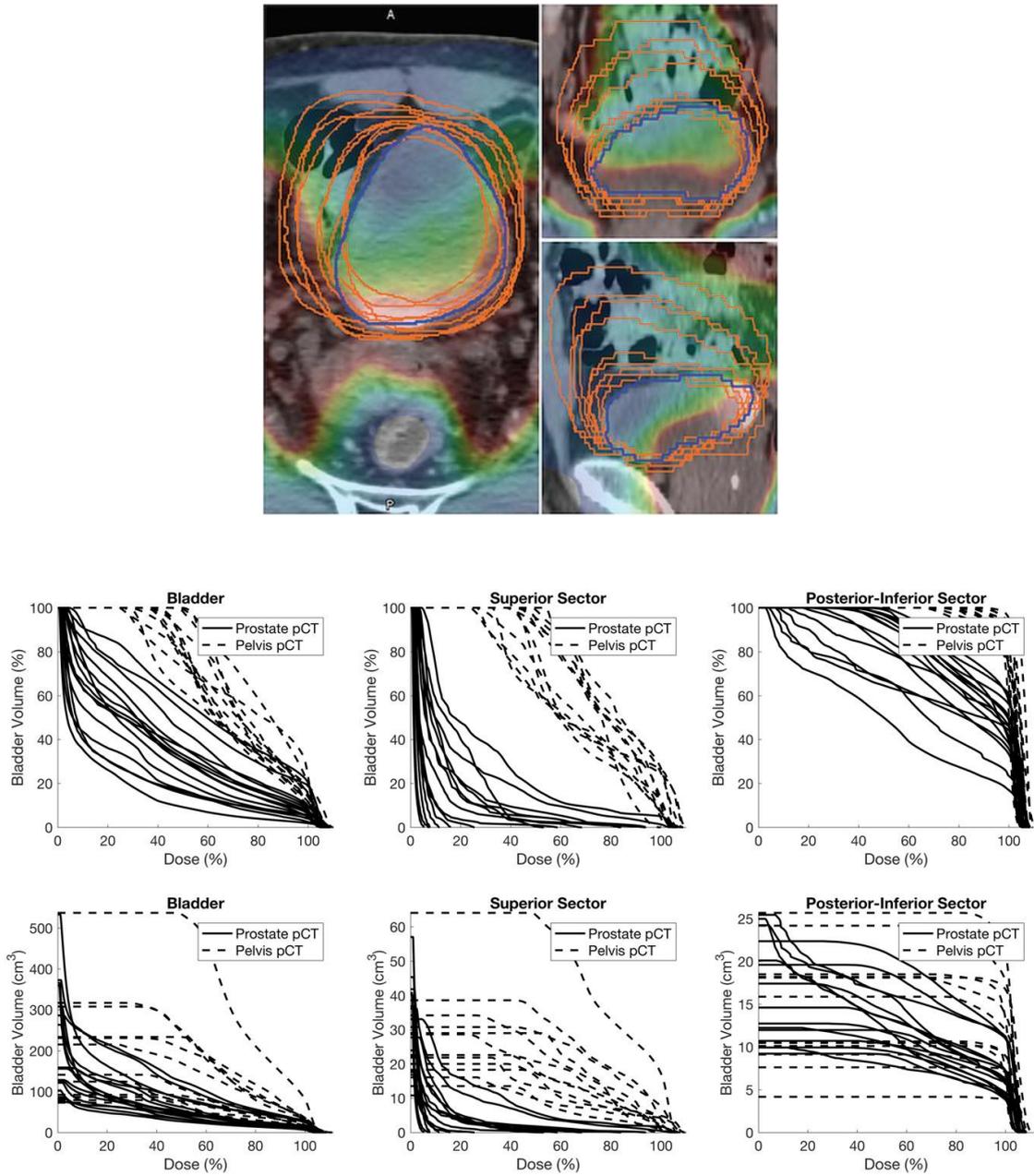


Figure 3.2: Examples of the classic approaches presented (Images from Casares et al. [CMMH⁺17]).

Their goal was to provide domain experts working in the field of segmentation a way to analyze how certain organ variations affect the segmentation results. They described the used segmentation data by scale and rotation invariant descriptors called spherical harmonics [KFR03], iteratively decomposing the shape into a series of coefficients. In contrast to other works, they employ two different dimensionality reduction techniques to reduce their 8D vector to a 2D representation (as seen in Figure 3.3b). The whole cohort containing multiple different organs is transformed using t-SNE [MH08] to differentiate between the different types of organs. If a single organ is viewed independently, its shape vectors are transformed using Principal Component Analysis (PCA) [Shl14]. This allows users to quickly identify outliers created by segmentation errors. A detailed inspection of certain organs is provided in a 3D space where the mean shape is presented with a color-coding indicating the likelihood of shapes being bigger or smaller. This work includes many different organs and places them in a common space. It does not differentiate between patients and timesteps and therefore forgoes most of the valuable cohort meta-information. Additionally, they use automatically segmented meshes, while we use manual segmentation lines, which are not directly convertible. The reason for this being that their segmentation surfaces have precise triangle-to-triangle correspondences. Furthermore, they do not offer a way to show common shape types on a per organ basis.

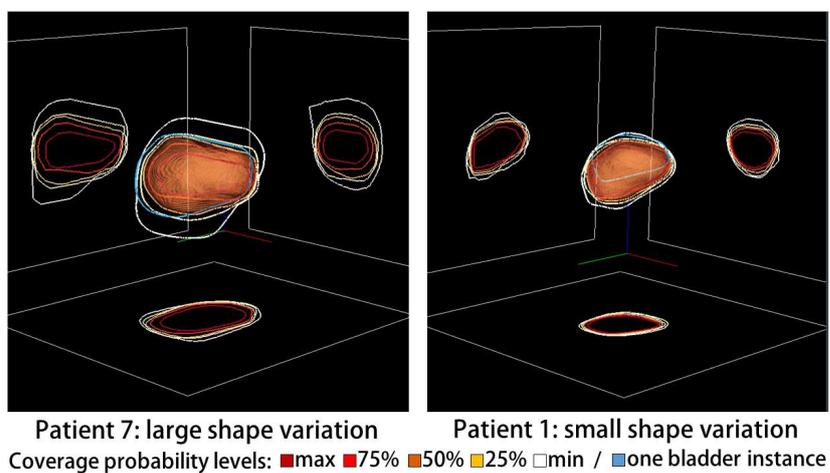
Generally, the use of descriptors as presented in the former works, allows their tool to efficiently differentiate between different shapes. But it lacks the ability to synthesize arbitrary elements in their shape space, restricting the analysis to elements present in the cohort. In regards to representing the shape variability, previous works either used coverage levels or median shapes based on triangle-to-triangle correspondences. The former ones do not represent the actual object shape and only allow the analysis of a single organ, while the latter approach makes an assumption which is not given in our current application.

3.2.2 Statistical Approaches

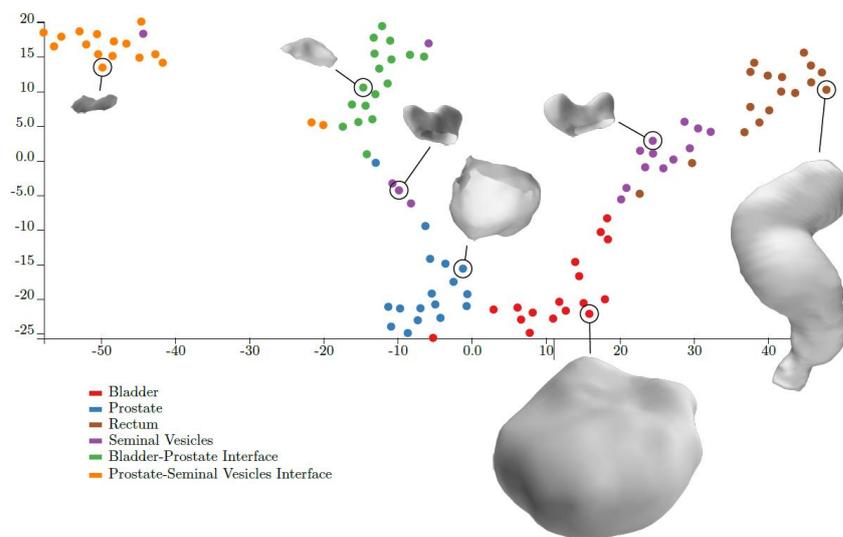
The work of Hermann et al. [HK15] provides an overview of different applications in the field of statistical shape analysis, especially as they deal with medical and biological data. They define three essential steps that a shape analysis tool needs to perform for an effective shape exploration:

1. Registration
2. Analysis
3. Synthesis

In the first step, all shapes are *registered* to a template using translation, rotation, and scaling, the mean shape is often employed as a basis. After this, a statistical *analysis* is performed, which transforms the high-dimensional shapes into a reduced space where the



(a) Raidou et al. [RCMA⁺18]



(b) Reiter et al. [RBGR18]

Figure 3.3: Examples of the descriptor-based shape variability approaches presented.

principal axes describe their main variations. The final step for many applications is to provide a method of *synthesizing* shapes from the reduced space to the original one for inspection. This must work for elements already present in the data and freely picked ones.

One of the first works in this context was the one by Busking et al. [BBP10]. They used a 2D scatter plot to represent the distribution of the elements inside the reduced cohort and allowed the user to not only show concrete elements present in the data but also to synthesize arbitrary objects in the shape space. They calculated and visualized the differences between a selected shape and the mean shape by using a color-coded 3D view. By utilizing a so-called evolution view, where the 3D objects are represented as 2D silhouettes, they want to visualize certain trends in the data. For this, they use three different visual representations: side-by-side, overlaid, or stacked 2D renderings. In another work, they also dealt with the topic of directly comparing intersecting 3D surface meshes [BBF⁺11]. An overview of their tool can be seen in Figure 3.4a. Their work was one of the first to outline how an interactive visual analytics approach can be used in the exploration of shape variations. While the work can analyze a single object in detail it is not extendable to multiple ones. Its 2D plot of the shape distributions shows well which organs are similar but also neglects the notion of patient and timestep. Furthermore, the information on positional variation is also removed due to the initial registration process.

Following this work, the basic idea was picked up by other researchers and applied to other use cases. Landesberger et al [VLBK⁺13] showed its applicability for the analysis of segmentation models to enhance the users understanding of the connection between the underlying data, the model parameters, and the segmentation results. Their tool provided visualizations for the distribution of the cohort elements (as seen in Figure 3.4b), similar to the one presented in Busking et al. [BBP10], as well as color-coded surface models indicating the segmentation quality, and aggregated bar charts providing an overview of the overall segmentation quality in the cohort. Although the use for segmentation is not directly applicable to our case, their visualization approaches for errors could be used to encode variability. They currently do not include any notion for variability for individual patients, and similarly to other works offer no option to analyze the cohort in a temporally coherent manner.

Hermann et al. [HSK11, HSSK14] use a similar approach for the analysis of morphometric differences in animals. In their works, they provide tools to inspect how certain semantic traits affect the structure of elements. To find the differences between two different classes along with a certain trait they used a side-by-side representation for each class and color-coded the difference to their shared mean, highlighting differently shaped regions [HSK11]. In their succeeding work, they analyzed the concept of anatomical covariance, where a change in one region affects another one. Contrary to previous works they use tensor glyphs indicating variation directions as an overview visualization. They also allow the user to select individual regions and manually transform regions to analyze how other parts of the object transform in detail [HSSK14]. Examples of both works can be seen in 3.4c and 3.4d. The analysis based on certain semantic features could offer

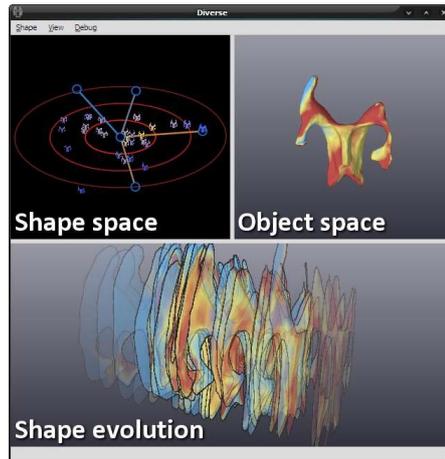
great insight into the underlying reasons for problems like bladder toxicity. Their current approach is largely based on the contrasting differences between two opposing features, which is not the case in the human-based data set we are using.

The general concept of representing cohort data as high-dimensional vectors and reducing them using statistical analysis methods like PCA serves as a good basis for the method presented in this thesis. Our approach differs in a few key instances: Previous methods only focused on analyzing a single type of organ at the same time, we instead incorporate multiple different ones. This also means analyzing their intersections. Secondly, in regards to the used visualization approaches the previous shape space analytics tools heavily rely on allowing the user to manually explore the whole shape space, while in our case the analysis tasks mostly focus on understanding the combined variations of single patients or certain groups across multiple timesteps. Finally, most works describe a common problem in finding an effective way of comparing more than two elements to each other, which we also plan to solve using a combination of aggregation and exploded views.

3.3 Ensemble Visualization

The analysis of shapes allows us to place them in a common space and explore different possible shapes and their similarity. We are more interested in exploring the variations already present in our data. This is especially true if we aim at visualizing the shape variability of a group or a patient. To achieve this we can look at methods from ensemble visualization. Most solutions in this field do not use medical data, but instead large collections of spatiotemporal data, for example from weather simulations. Wang et al. [WHLS18] looked at tools allowing users to understand these large data ensembles, and defined a simple pipeline, which most approaches follow, as seen in Figure 3.5. Starting with the ensemble data, a first step could be to perform aggregation and calculate statistical properties like the mean, median, and standard deviation. After that, the data and statistical properties are rendered. As a final step, multiple visualized elements can be composited. Based on the pipeline they defined, one can differentiate works in this field by looking at which components they employ.

A good example of works that perform an *aggregation before visualization* are the ones of Mirzargar et al. [MWK14] and Ferstl et al. [FBW16, FKRW16, FKRW17]. By redefining the concept of data depth, a measure to estimate how central a data point is inside a group, to be applicable for functions and streamlines, Whitaker et al. [WMK13] were able to estimate the necessary statistical elements for a box plot. Their contour box plot was extended for streamline ensemble data by Mirzargar et al. [MWK14] who were able to show mean, variability, and outliers in simulated hurricane datasets. While rendering an arbitrarily large number of streamlines would only result in a messy spaghetti plot, their aggregated version showed the main trends in a precise and visually clear manner, as seen in Figure 3.6a. As the calculation of their band depth for each streamline in the ensemble has quadratic growth, their implementation might not be suitable for interactive



(a) Busking et al. [BBP10]

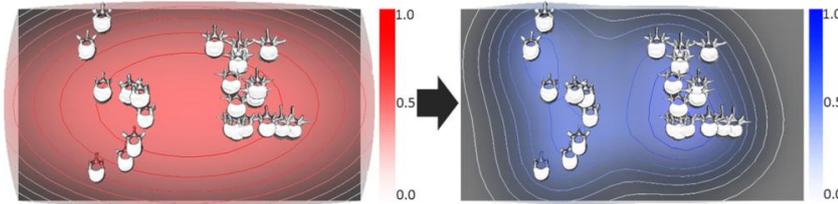
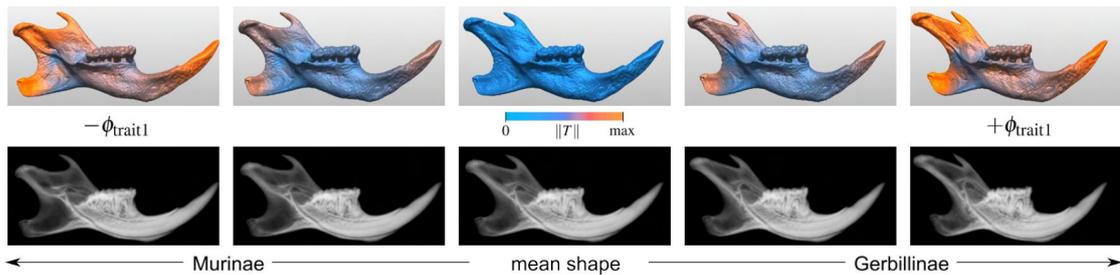
(b) Landesberger et al. [VLBK⁺13]

Figure 3.4: Examples of the statistical shape variability approaches presented.

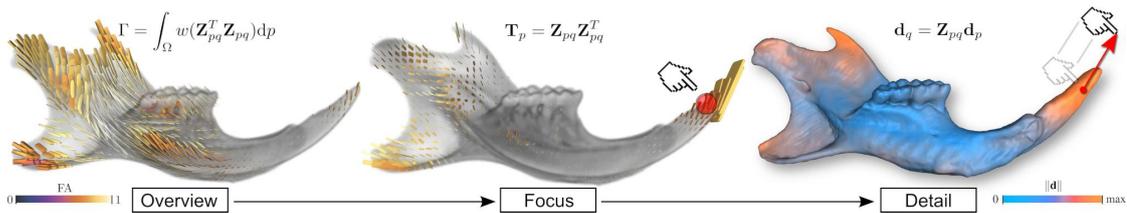
applications.

Following this work, Ferstl et al. [FBW16] created a similar visualization, but instead of calculating the band depth, they used a technique similar to the ones discussed in shape analysis. By sampling streamlines along their path and interpreting the collected points per streamline as a vector, they transformed all streamlines into a dimensionality reduced space using PCA. By sampling the confidence ellipsoid and median in the reduced space and transforming the points back into the original space they were able to calculate a box plot like representation. Their result can be directly compared to the previous work of Mirzargar et al. [MWK14] in Figure 3.6b. In a follow-up work Ferstl et al. [FKRW16] improved upon their previous method by using a signed distance function representation for the streamlines. This, in turn, allows them to perform the calculation of the variability lobes faster, by only sampling two points for the standard deviation. At the same time, this also enables the calculation of correlated streamline parts in the aggregated lobes. The latter can be used to convey a notion of where streamlines that went through one

3. RELATED WORK



(c) Hermann et al. [HSK11]



(d) Hermann et al. [HSSK14]

Figure 3.4: Examples of the statistical shape variability approaches presented (*continued*).

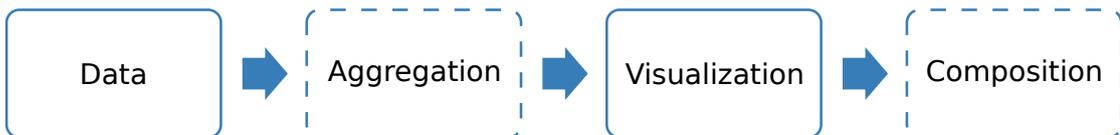


Figure 3.5: Ensemble visualization workflow, adapted from Wang et al. [WHLS18]. The dotted lines show that the respective pipeline parts do not have to be present in all works.

point tended to move afterward. Their workflow and respective results are shown in Figure 3.6c. In further extensions [FKRW17] they showed how to employ these techniques to cluster streamlines in large time-varying ensembles, with the goal to highlight how certain patterns evolve over time. To achieve this they employ different visualization techniques like stacked streamline variability plots, seen in Figure 3.6d. The basic idea for their work has currently been only applied for weather simulation data and not to medical information, where larger variations in shape may occur. Furthermore, while their estimation of shape variability not only works for 2D lines but also for 3D volumes, they have not implemented visualizations that are able to encode information about the 3D median and confidence intervals.

In contrast, Pfaffelmoser et al. [PW13] perform the other way around, visualizing the data, in the form of contour lines over an ensemble first, and then compositing the final image in a way that reveals the underlying structure. Although the initial spaghetti plot of superimposed contours works well for a low number of ensemble members, the

more there are the more complex understanding gets. To solve this problem, they use two techniques: the general direction of the underlying field of values is encoded using a black-gray-white color scheme showing which parts are below and which are above the iso-value. The contours themselves are colored based on the underlying cumulative distribution and the gradient of the value field. Their processing steps and the final result can be viewed in Figure 3.6e. In combination, these two visualizations allow them to differentiate similarly appearing contour ensembles. Their chosen method works well for their selected use case but it assumes similar shapes which we may not always have. It also may be more difficult to interpret by medical experts, as the three different colormaps need a proper explanation.

Looking at the two different categories: aggregation before visualization and composition after visualization, each may offer different benefits. The latter can ensure distinguishability for complex representations, while the former generally tries to achieve a simpler representation by aggregating the base information. Many approaches in this field are solely working with simulated data in the form of streamlines or isocontours and not medical segmentation data, which has to be appropriately processed. Due to the size of the cohort, restricting to only use one of the two categories might restrict the ability of users to compare two different shape ensembles to each other.

3.4 Comparative Visualization

Many of the aforementioned works struggle with finding effective ways to compare multiple visual elements to one another, like for example trying to understand how multiple ensembles of contours or 3D meshes differ. Recently Kim et al. [KCK17] wrote a comprehensive overview of works in this field, categorizing them into four different groups, based on how they chose to compare multiple elements. A visual overview of them is provided in Figure 3.7:

- **Juxtaposition** (Side-by-Side): The elements are placed next to each other, each in its own frame of reference.
 - Pros: Provides a good overview and has no occlusions
 - Cons: Tradeoff between element size and number of elements due to screen space limitations (e.g. lots of small elements vs only a few larger elements)
- **Superimposition** (Overlay): The elements are placed on top of each other in a shared frame of reference.
 - Pros: Direct comparison of full-sized elements
 - Cons: Visibility problems due to occlusions limit the number of elements
- **Interchangeable** (Temporal Superimposition): Only a single element is shown at once on the screen, the view can switch through the elements, either manually or animated.

3. RELATED WORK

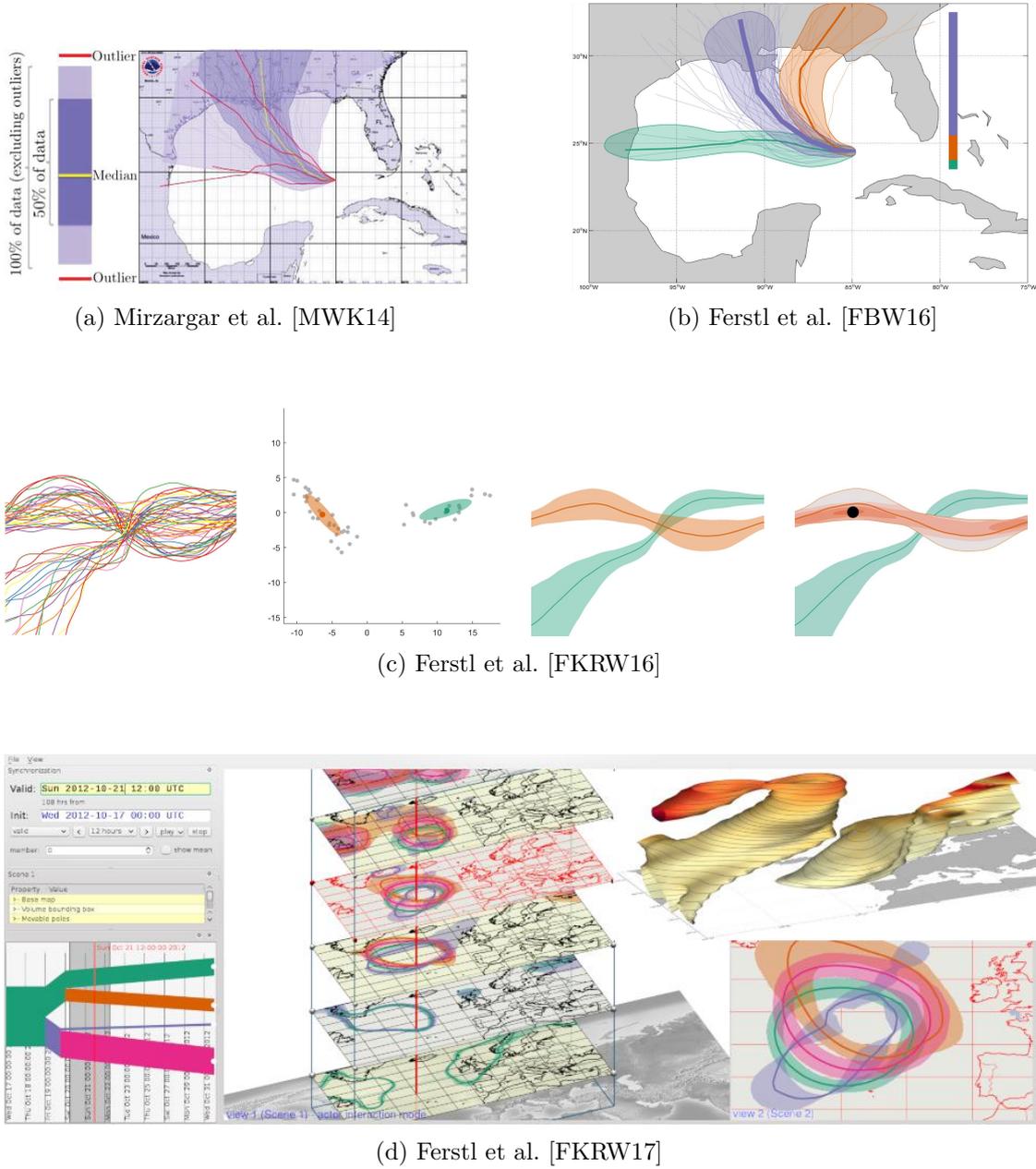


Figure 3.6: Examples of the ensemble visualization approaches presented.

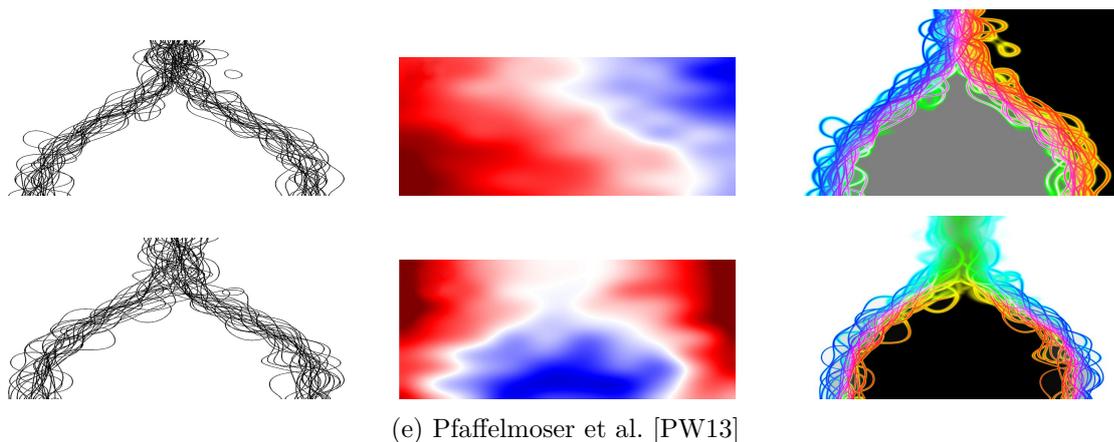


Figure 3.6: Examples of the ensemble visualization approaches presented (*continued*).

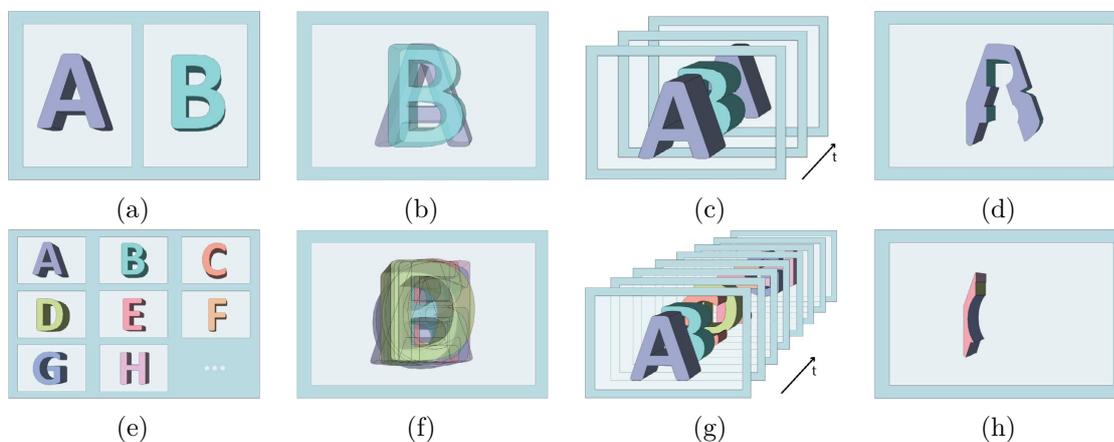


Figure 3.7: Comparison between the fundamental techniques of comparative visualization, showcasing their scalability from two elements to n elements [KCK17].

- Pros: Full-sized view of a single element without occlusions
- Cons: Limited by the number of elements a user can remember
- **Explicit Encoding:** Instead of showing the elements themselves, the differences between them are calculated and shown.
 - Pros: Focused view highlighting even small differences
 - Cons: Original information is lost and the difference is not always meaningful

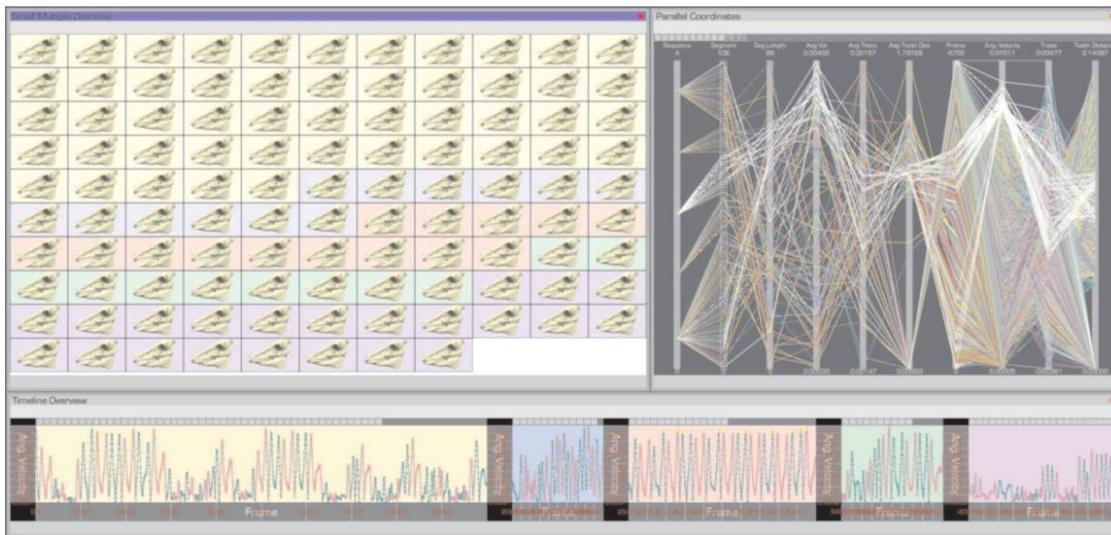
The choice of encoding is largely depended on the application because there is no universally best way. Looking at different examples of mesh comparison approaches in other works, we find uses for all options. Keefe et al. [KER09] for example employs a

juxtapositioned view of small representations of a pig’s jaw to visualize its movement while eating. By explicitly encoding the movement the small representations only show information relevant to the analysis. While only being a part of a more complex visual tool, which includes a parallel coordinate view to perform an in-depth search in the cohort, it serves well at providing a good overview of the available data elements and their respective differences. In Figure 3.8a the basic layout of their tool is presented. This work is a good example of how a combined encoding type is able to achieve better results than a single view. During their work, they show a good way of providing users with an overview of the cohort. For our work, we would also need to include information about the patients behind the data and the temporal order between the cohort elements.

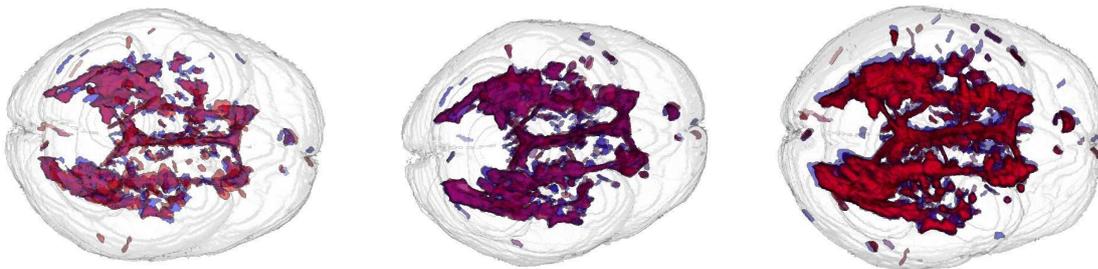
A more classic approach is applied in the work of Tory et al. [TMA01] where the extracted brain lesions from MRI images are superimposed. Their development is visualized either by rendering all of them at once using different colors for each timestep in conjunction with them being transparent or by creating an animated view showing the change over time. While this rather simple approach works well for a small number of timesteps, the analysis gets harder and nearly impossible if more timesteps are added. In their case, they pushed the method to its limit with 11 superimposed elements. The scaling of their technique for 2, 4 and 11 elements can be seen in Figure 3.8b. Their animated view corresponds to the use of an interchangeable representation. The restrictions in terms of scalability are approached by our tool through aggregation as it could be possible for users to analyze arbitrarily large sets of segmentation contours. In general, they investigate the lesions of only one patient at a time, whereas we also want to be able to compare larger groups of patients in a common space at once.

How explicit encoding can be used to highlight the differences between 3D meshes, can be seen in the work of Schmidt et al. [SPA⁺14]. In their work, they focus on how to compare a large number of similar meshes, created through different point cloud surface reconstruction algorithms. To quickly identify regions with high errors in the reconstruction the mesh is rendered textured with a heatmap. As they mention in their work, this kind of approach does not scale well with multiple meshes. Therefore they implemented a parallel coordinate plot visualization, where the errors of interesting regions for different algorithms are drawn on top of each other, which allows for an easier comparison of a high number of meshes (as seen in Figure 3.8c). Using a highlighted map of the surface is a good way to convey where certain deformations are located. As with other works, the use of surfaces is not applicable for our data, and we need to find another way to estimate differences between shapes. Furthermore, their work only focuses on a single object and its deformations, which we want to extend to multiple intersecting organ shapes.

The use of comparative visualization techniques seems to be more and more necessary as we aim at providing insight into the different shape variations present in the cohort. The aspect of being able to appropriately scale a visualization in such a way that the comparison of multiple groups of shape variations is possible, requires the use of a combination of techniques.



(a) Keefe et al. [KER09]



(b) Tory et al. [TMA01]

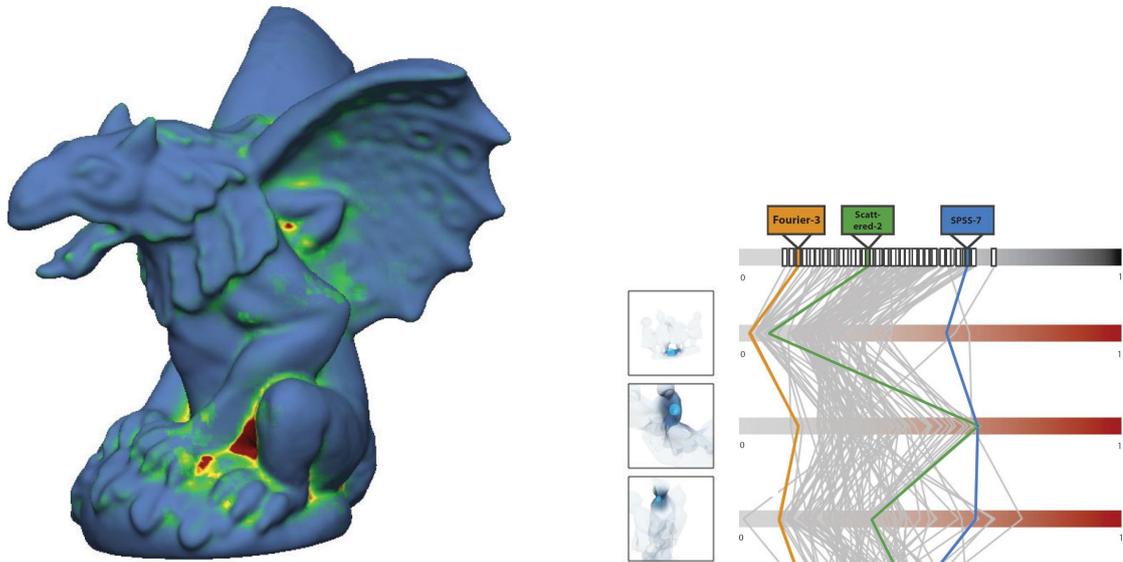
Figure 3.8: Examples of the comparative visualization approaches presented.

3.5 Cohort Analysis

The underlying tasks we approach with our tool can be roughly split into two groups we aim to support:

- *Planning* of a treatment procedure by medical experts trying to analyze and individual patients bladder variability
- *Reasoning* by medical researchers about the general variability of certain organs across a large cohort of patients

While the former can be approached using shape analysis and comparative visualization the latter requires appropriate abstraction and representation methods. These methods

(c) Schmidt et al. [SPA⁺14]Figure 3.8: Examples of the comparative visualization approaches presented (*continued*).

should both be simple enough to provide a good overview and discriminative enough to not hide important details. The works we looked at to solve this problem were from the field of cohort visualization.

Connecting a bit to the topic of shape analysis presented in Section 3.2, Klemm et al. [KLR⁺13, KOJL⁺14] provide several methods on how to analyze pathological deformations of the spine across a cohort. The first work on this topic [KLR⁺13], mostly focused on the extraction of the spine shapes of all patients and the exploration of similarly shaped spines by clustering them in groups. By using a comparative side-by-side visualization, they allow users to comprehend common formations of the spine in the cohort as well as to detect outliers (as seen in Figure 3.9a). In their following work [KOJL⁺14] they generalized the concept of extracting information from images to perform a cohort analysis. The use case and the extracted image information are similar to the previously mentioned work. They are put in relation according to a multitude of other patient parameters like gender and height. The data is aggregated using pivot tables and histograms are used to show the distribution of a certain parameter. For each histogram bar, a 3D representation of the associated mean spine shape is presented, which can be seen in Figure 3.9b. Their approach allows the user to add more parameters to the visualization, changing the histogram to a mosaic plot, showing not only what the average spine shape is for a certain group but also its proportion of the whole population. The works of Klemm et al. [KLR⁺13, KOJL⁺14] show how to effectively reduce image data in a way to analyze its information across a whole cohort to facilitate understanding on

a broader basis. Especially the use of simple tables as a tool for providing information to medical experts in a common form, heavily influenced our work. The used data and analysis methods are not applicable for our data and more importantly, we want to preserve the temporal context of a patients treatment process, which would not be possible if we cluster the involved organs extensively.

Looking beyond the topic of shape analysis, the work of Steenwijk et al. [SMB⁺10] proposed a framework for the visual analysis of patient cohort data based on the simple concepts of features, mappers, and studies to analyze difficult diseases. Features in their work can be any information regarding the state of the patient, be it simple yes or no blood test results or more complex 3D datasets. These features can then be transformed using mappers to extract additional information. By combining multiple features of a patient's state, a study for a disease is created. To support these analytical tasks, a wide range of visualizations is offered, including parallel coordinate plots to show high dimensional features, scatter plots with linked points to provide information about how certain feature points change over time, and even ellipsoid plots to visualize the variability inside a set of points in a scatter plot. Three of their visualizations are shown in Figure 3.9c. The basic idea of presenting information in features and using combinations of them to understand certain diseases is a valuable basis for any cohort analysis tool. Their visualization approaches might be rather complex for domain experts to learn and understand and are therefore not directly applicable to our case.

Considering the high learning curve that many complex visualizations of high dimensional data, like parallel coordinate views or dimensionality reduced views, led Blumenschein et al. [BBS⁺18] to propose a visualization concept aimed at people who are no visualization natives. Their tool exploits the familiarity of most users with tabular data representations. Their approach, as depicted in Figure 3.9d, was to use aggregated tables with color-coded cells to show the average values of the contained elements and also allowed them to make assumptions across multiple cells, i.e. dimensions. Due to the general inflexibility of a tabular view, they added the options to stack the tables by splitting cells to show parts of the aggregated values, added histogram bars to the table column's labels and even encoded the number of missing values for a cell by drawing varying dot patterns. All together their approach provides new users with data analysis tools that are more easily understandable. The general layout of their tool seems well suited to analyze the values and variability of high dimensional datasets, although all example datasets in their work were strictly value vectors and not medical 3D datasets. Additionally, they did provide some methods of manually recognizing clusters using sorting methods, but they did not offer any automatic support for users in this regard.

Looking at works in the field of cohort analysis, the previously presented tools are often intended for the use of experts who already know complex visualization techniques or they are intended exclusively for medical researchers. The representation of temporal progression along a treatment is also often not visually encoded. Our tool aims at solving these problems by being intuitively understandable by users who are only familiar with simple data visualization approaches and whose information is helpful for both researchers

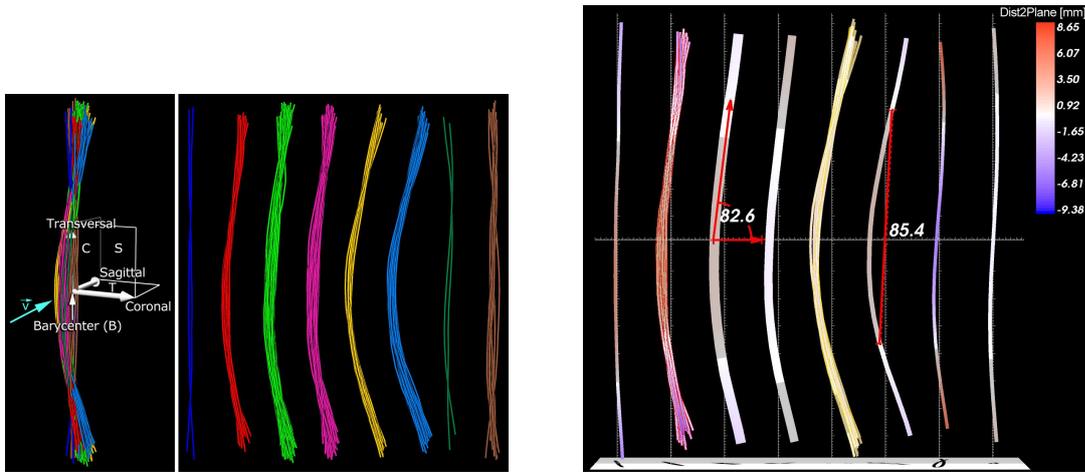
3. RELATED WORK

trying to understand the whole cohort, as well as medical experts creating dose plans for individual patients.

To complete this chapter, we have aggregated all the works we looked at in the course of this chapter in Table 3.1. In this table, we compare all related works to each other based on which of our research questions they can answer. We can see that none of them can answer all at once.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
Raidou et al. [RCMA ⁺ 18]	✓	✓	✗	(✓)	✓	✗	(✓)
Reiter et al. [RBGR18]	✗	(✓)	✗	(✓)	✓	✗	(✓)
Busking et al. [BBP10]	✗	✗	✗	(✓)	✓	✗	✓
Landesberger et al. [VLBK ⁺ 13]	✗	(✓)	✗	(✓)	✓	✗	✓
Hermann et al. [HSK11, HSSK14, HK15]	✗	(✓)	✗	(✓)	✓	✗	(✓)
Mirzagar et al. [MWK14]	✗	✓	✗	✗	(✓)	✗	✗
Ferstl et al. [FBW16, FKRW16, FKRW17]	✗	✓	✗	✓	✓	(✓)	✓
Pfaffelmoser et al. [PW13]	✗	✗	✗	✗	(✓)	✓	✓
Keefe et al. [KERC09]	✓	(✓)	✗	(✓)	✗	✗	(✓)
Tory et al. [TMA01]	(✓)	✗	✗	✗	(✓)	✓	✓
Schmidt et al. [SPA ⁺ 14]	✓	(✓)	✗	✗	✓	✗	✓
Klemm et al. [KLR ⁺ 13, KOJL ⁺ 14]	✓	✓	✗	(✓)	✓	✗	✓
Steenwijk et al. [SMB ⁺ 10]	✓	✓	(✓)	(✓)	✗	✗	✗
Blumenschein et al. [BBS ⁺ 18]	✓	(✓)	✓	✗	✗	✗	✗

Table 3.1: Overview of all related works and how their individual contributions can be used to solve our proposed research questions. (✗ = No, (✓) = Partially, ✓ = Yes). As can be seen no work offers a complete solution to the presented problems.



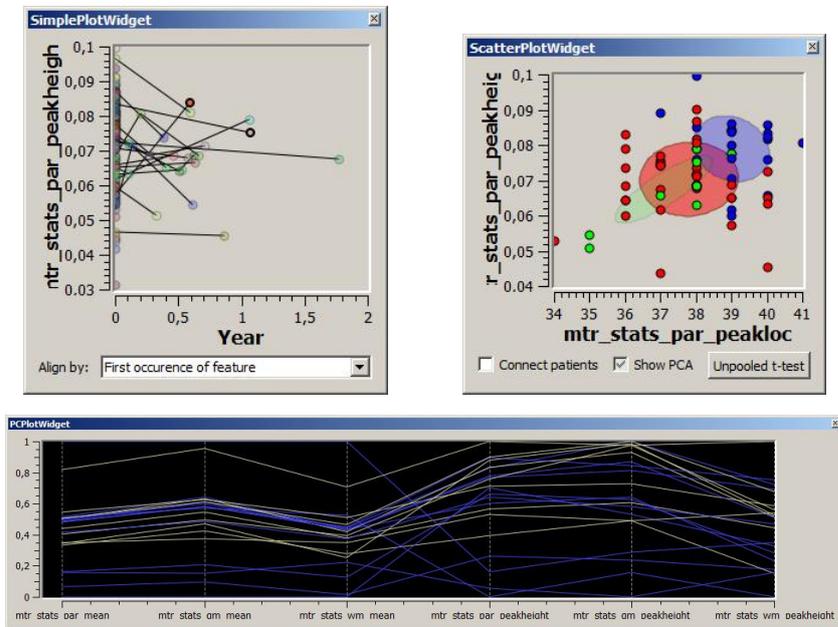
(a) Klemm et al. [KLR⁺13]



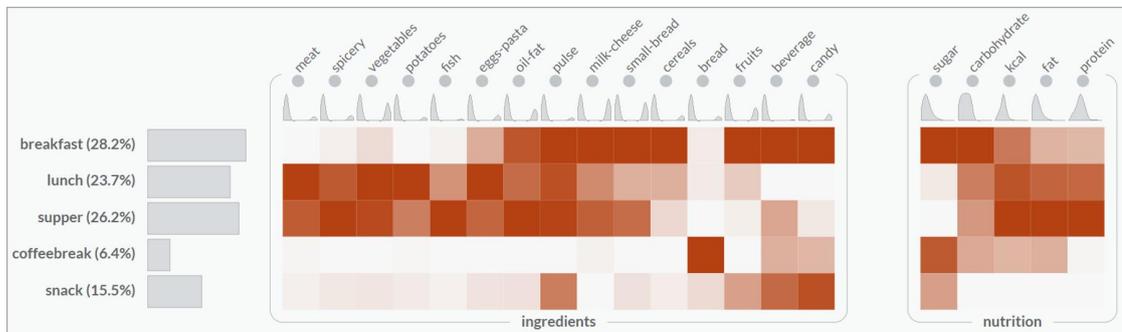
(b) Klemm et al. [KOJL⁺14]

Figure 3.9: Examples of the cohort analysis approaches presented.

3. RELATED WORK



(c) Steenwijk et al. [SMB⁺10]



(d) Blumenschein et al. [BBS⁺18]

Figure 3.9: Examples of the cohort analysis approaches presented (*continued*).

Visualization Design

In the course of this chapter, we will discuss the ways in which our proposed tool tackles the aforementioned tasks of cohort and local exploration in segmented organ data. We will compare the different options available and explain why we think that the chosen encodings are appropriate choices. Section 4.1 presents the underlying shape analysis technique we employ to understand the organ shapes present in the cohort. In Section 4.2 we take a look at how we can solve Task **(T1)** and explore the whole cohort using an overview representation. Section 4.3 approaches Task **(T2)**, the detailed shape exploration. We do this using a, to the best of our knowledge, new visualization application for 3D shape variability based on a medical 2D slice view.

4.1 Shape Analysis

Tasks **(T1)** and **(T2)** both require knowledge about the similarity between the different organ shapes as well as ways to estimate their local shape variations. To allow this we first need to decide which data representation allows us to fulfill these two tasks and how do we convert our segmentation cohort into it. In this section, we will analyze the given data in contrast to other works, as well as how and why we changed it into the currently used representation.

4.1.1 Data Transformation

Before any further processing can be done, the first step that needs to be taken is to find the best representation for the data we are given. The reason for this is, that the form in which the organ shape information is stored has a large impact on how easy the implementation of further processing and visualization steps will be.

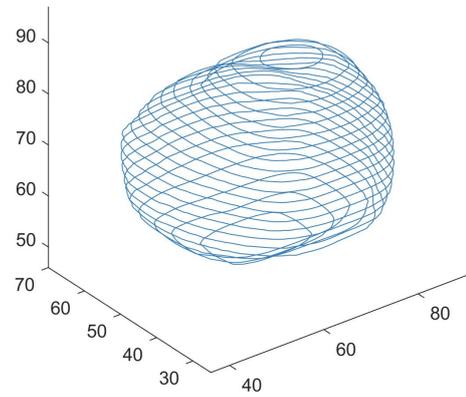
Initially, the shape data is saved in the format of segmentation line ensembles, where each organ is represented by its outer contour. The contour is defined through collections

of lines, defined by numerous points. Multiple lines are stacked on top of each other to encompass the whole organ [SPP00]. These segmentation lines are created manually by medical experts. Another format common in this field are triangle meshes, which are created automatically with segmentation algorithms. Reiter et al. [RBGR18] used this representation in their work. These representations are created by deforming a predefined mesh, consisting of vertices and edges until it fits the organ [MBL⁺02]. Another often employed method to work with organ data, which can be seen in the work of Raidou et al. [RCMA⁺18], is by using a volumetric representation. The information about the organ is stored at multiple points in a three-dimensional grid, where each point can contain a value, indicating either the distance to the organ, a density value, or simply if the point is inside the organ or not [DCH88]. Altogether we found three possible representations for our organ shape data: line ensembles, triangle meshes, and volumes.

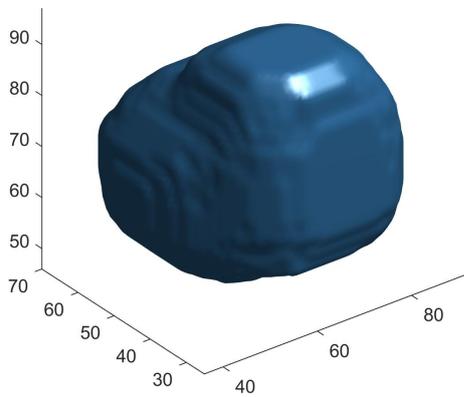
Each of these three methods has different strengths and weaknesses, to determine which representation is the best for our application we want to analyze them based on a few key factors: The first factor we are interested in knowing, is the difficultness with which one can *compare* multiple elements. The comparative capabilities should also apply to substructures of shapes. The possibility to *aggregate* multiple elements in a meaningful way should also exist. The idea behind this is to create a combined or average version, for a group of elements. In the case of intersecting elements there needs to be a way to detect these occurrences and estimate the *overlap* region. Apart from the aspects relating to data manipulation, the data also needs to be *rendered*. We need to consider the efficiency with which this can be done, especially in web applications with a limited resource budget. Last but not least, the general *memory* consumption of the stored data needs to be taken into account. This is important, if the calculation and visualization are performed on a server-client basis.

Starting off with the *line ensemble* shown in Figure 4.1a, the direct comparison between multiple elements or in our case patients is not easily done as the lines themselves have a varying number of points and are not defined for all slices in each volume [SPP00]. The same problem also affects any approach to aggregate these lines, although one could merge multiple lines into one ensemble. The handling of slices with multiple contours or no contours might prove difficult and can require complex algorithms [RF00]. Although there are methods to estimate which contour lines intersect and produce overlaps, this only works if the lines are defined on the same slice and determining precisely the overlapping region, again requires algorithmic computation. Visualizing a line ensemble is rather straightforward as it only requires drawing several lines, one after another. Finally, the size of a line ensemble is rather large, with around 30 MB per element but varies between different elements. Some experts define contours with a low number of points, while others use more, for example, a single organ shape can have between 10 and 40 slices. A slice can have around 100 up to 2000 defined points each with three components. Looking at a typical bladder shape in the cohort, the contour has 23 slices with each of them having around 1200 points, which makes a total of 27.600 points.

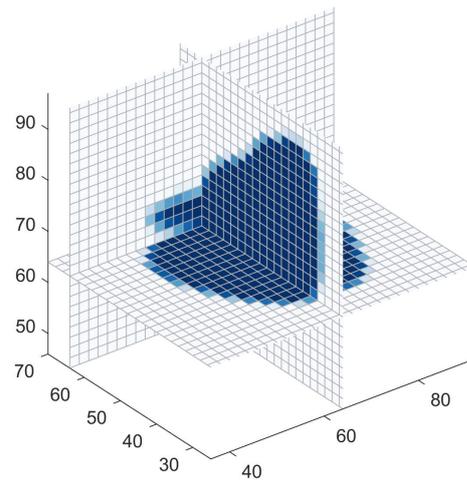
Looking at the *triangle mesh* representation like the one in Figure 4.1b, we can see



(a) Line Ensemble



(b) Triangle Mesh



(c) Volume

Figure 4.1: Three different representation methods of shape data, shown on the example of a segmented bladder.

that a simple direct comparison is only easily doable if a direct triangle-to-triangle correspondence exists. In this case, the comparison of differences in substructures or the aggregation of multiple meshes to an average one is also possible, as shown in Reiter et al. [RBGR18]. If there exists no triangle-to-triangle correspondence these tasks get a lot more difficult, not only that meshes may have varying numbers of vertices and faces but also the corresponding substructures need to be estimated beforehand before any comparisons or aggregation can be performed. Schmidt et al. [SPA⁺14] tackled this problem by finding the shortest distance between two nested surfaces along the surface normal. Estimating the overlap between multiple different triangle meshes is an even more complex problem than for line ensembles as one has not only to estimate which parts are inside both meshes, but also how to connect them to form a proper new mesh [SKB18]. Yet there also exist approximate solutions using bounding volumes that allow for a faster intersection checking [Lun16]. As triangle-meshes are probably the most common rendered elements and the base primitive for most 3D applications, they can be drawn efficiently [AMHH18]. The amount of memory used by them is dependent on how detailed the surface of the object is, i.e. how many vertices are used. For the example of a typical bladder shape, this is around 7000 to 8000 vertices with each having three components.

Finally, *volume* data has no notion of the object's actual shape as it only consists of an array of density or distance values, as indicated in Figure 4.1c. Under the assumption that all volume files are in a common space, direct comparison and especially aggregation is possible on the basis of these values. In the work of Raidou et al. [RCMA⁺18] it was shown that by accumulating and normalizing multiple shape volumes, one can estimate the probability of certain volume parts to be inside the shape. A similar approach can be chosen to estimate overlapping parts of multiple shapes as this can be done by a simple Boolean operation on a per-value basis, delivering precisely the intersecting region [Sig06]. Since current graphics hardware is not aimed at drawing volumes, rendering the data can be computationally more expensive, requiring techniques like tracing rays from the viewpoint through the volume and accumulating the values and along the path [RPSC99]. Last but not least the memory consumption of volumetric files can be rather extensive due to the fact that most of the time the whole volume needs to be saved even if the shape is present in only a small part of it. This makes it independent of the underlying shape complexity, but it can lead to large memory consumption. For example, a volume with a resolution of $450 \times 450 \times 450$ leads to 91.125.000 values per element, or, if a smaller resolution of $128 \times 128 \times 128$ is used, to 2.097.152 values. In some cases, this problem can be tackled by compressing volumes into sparser representations, where only the elements containing information are stored [THGE99].

Looking at all of these representations, we decided to use a hybrid approach with a combination of volumetric data and triangle meshes. The volumetric data representation was chosen since it allows us to directly analyze the data, without the need to know triangle-to-triangle correspondences or the extraction of related surface positions. In this representation each organ shape occupies the same space allowing not only for

direct comparison, but also enforcing a shared resolution, i.e. level of detail, both supporting further calculations. Although the calculations are easier in this space, the visualization is made more difficult. Its large size makes it hard to use in client-server structures. Additionally, the volume rendering techniques needed to display it are also computationally expensive. Therefore we decided to use a surface triangle mesh representation for more efficient visualizations. Volume data can be transformed into a triangle mesh by extracting a so-called isosurface, which is a surface connecting points of equal value inside the volume [LC87]. While the extraction process can take a short amount of time, the created mesh can be transmitted with far lower cost and be visualized with common approaches.

4.1.2 Shape Space

The created volumes contain for each timestep of each patient the coverage mask of the organs present in the cohort. Due to volumes having no inherent notion of which part of a shape corresponds to which part of another one, we need to apply a further transformation to gain knowledge about their shape differences and variability. Therefore the next step is to create a shared space in which the shapes exist based on their similarity, placing similar-looking shapes close to each other. This information is a vital basis on which further analysis tasks build. Referring to recent related works, we identified two main possibilities for the creation of shape spaces: volume-based analysis [Sh14] and descriptor-based analysis [PI97].

Volume-based analysis describes methods like PCA [Sh14], that treat each voxel value as an individual dimension and try to model shape variations by separating data points as good as possible. For example, by either rotating the high dimensional coordinate system in such a way that variance is maximized or by finding a transformation that separates groups of data points as far apart from each other as possible. Descriptor-based methods use instead of statistically derived information, handcrafted features extracted from the underlying data [PI97]. During the creation of the shape space, the developer defines a certain number of features, which can be calculated from the data, like the centroid of the volume or its solidity.

Our decision on what method is the best for our work was based on the analysis of these two methods based on a few important properties. Most importantly the method should be able to *separate* differently shaped structures. Besides, the chosen method is supposed to work for general cases where no a priori *assumptions* to the shape can be made. This also extends to mathematical assumptions made by the method itself. Furthermore, we also look for a method that allows us to create a *reduced representation* of the data, enabling us to store and use large cohorts efficiently. To this regard the computational *performance* of the methods also needs to be considered. In regards to the input and output, the methods should, on the one hand, be able to analyze the volumetric data and be ideally able to *synthesize* volumes from points in the shape space. Finally, we look into the *extendability* of the methods for the analysis of multiple different organ shapes at once

Volume-based analysis methods do not have any previous knowledge about the shapes they analyze but instead assume that there exist correlations between the voxels in the grid [HSSK14]. The analysis and the resulting differentiation quality are impacted strongly if the data is not properly normalized and if there are several outliers in the data. Certain methods like PCA also make the assumption that the data is spread based on a Gaussian distribution, which implies that the mean and variance are descriptive measures. If these assumptions are fulfilled, the result of these methods is not only a space where shape differentiation is optimized, but also allows the selection of the most descriptive variables to create a dimensionally reduced space representation. The initially large volumes can be represented by a small number of variables [Shl14]. Picking up the previous example, for a bladder volume with originally 2.097.152 values a reduced version retaining 95% accuracy, calculated using PCA, has after the transformation only 287 values. The computational requirements for volumetric shape analysis methods largely vary based on the chosen approach. PCA requires all cohort elements to be loaded into memory at once to estimate the covariance matrix, which can lead to a matrix with several gigabytes. A machine learning based approach using Autoencoder networks can work with small batches of data at a time [VLBM08]. A benefit of these methods is that one can invert the transformation from a reduced space to the original one, creating not only elements that were previously present in the data but also valid elements for arbitrary points in the reduced space [AW10]. Extending these methods for multiple organs is possible if they share the same volumetric space. This extension might impact the organ differentiability within organ-type groups.

Descriptor-based analysis methods base their ability to differentiate multiple shapes on the extraction of manually defined characteristics [PI97], including features like volume or perimeter [LL12], or information obtained by a Fourier analysis [VS01]. The quality of the differentiation is largely based on how good these descriptors are at identifying certain shape changes. As most descriptors are created to be used for specific shapes, they may struggle with certain structures with previously unexpected forms and have troubles differentiating them [RHC99]. This introduces the need to add new features to improve the results, which are created on initial assumptions of the shapes found in the data. Similarly to volume-based methods, the extracted features create a reduced space, where the number of dimensions is based on the extracted features. For example, in a previous work, Raidou et al. [RCMA⁺18] extracted 14 features for each bladder. Reversing this process to synthesize shapes from feature points is not possible, as there can be an arbitrarily large number of shapes leading to the same descriptor values [DS08]. One of the biggest benefits of these descriptor-based methods is that they are rather fast to compute, as they do not require knowledge about the whole cohort, but can instead be extracted on a per-patient basis. Concerning its use for multiple different shapes, descriptors can perform better in terms of inter-organ differentiability, as the analysis of each organ is not influenced by others [RBGR18].

Directly comparing the results of both shape analysis methods as depicted in Figure 4.2, we can see that both methods lead to similar results. Both methods can group similar

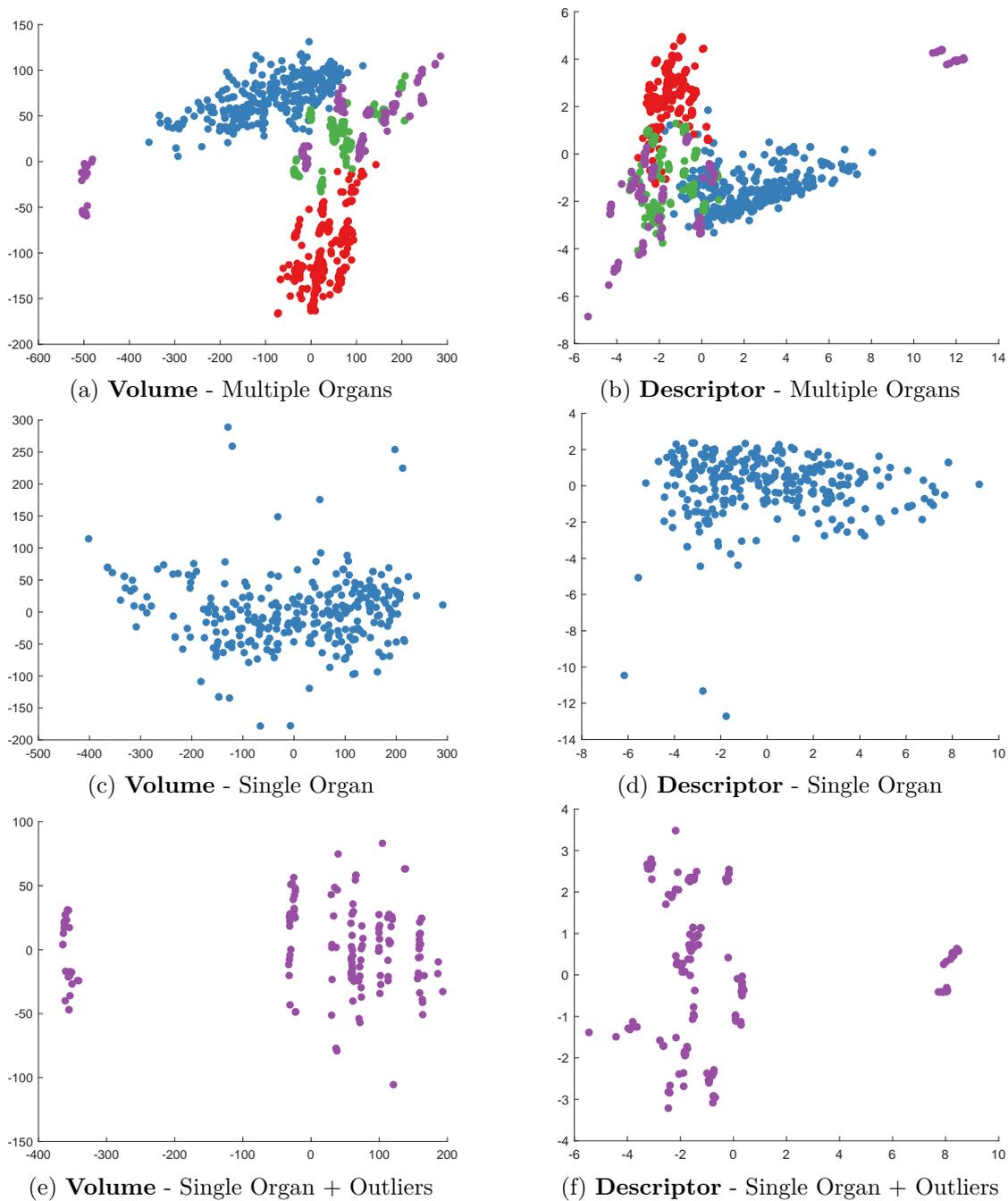


Figure 4.2: ■ Rectum ■ Bladder ■ Prostate ■ CTV Comparison of two different shape spaces, one created directly from volumetric (**Volume**) data and one derived from shape descriptors (**Descriptor**). The former was created from a $128 \times 128 \times 128$ volume sampled $32 \times 32 \times 32$ times and the latter is based on 25 shape descriptors calculated from the volume coverage mask, these descriptors include volume and orientation.

organs close to each other, as well as highlight outliers. Apart from differentiating a multitude of arbitrary organs and finding common patterns for each type, we also aim at recreating the initial shapes after the analysis. Thus, we decided to employ a volume-based analysis method. This is also in accordance with other works in this field showing the benefits of these methods like the one of Busking et al. [BBP10] for the analysis of anatomical shapes. Ferstl et al. [FKRW16] also have shown that using these volume-based approaches, one can perform successive calculations upon them like estimating uncertainty and variance. The initial calculation of the shape space for the whole cohort has high computation and memory demands. The resulting shape space, and especially the dimensionality-reduced elements are able to make up for this as they require far less memory. We think, that the benefits of a volume-based method outweigh the downsides. Especially as it is possible to estimate the transformation into the reduced shape space in a preprocessing step and apply it to the data iteratively, allowing for easy addition of new elements. The creation of a unified shape space for all organs is not feasible using volume methods. The resulting space would largely focus on differentiating the organs from one another, which does not provide us with important new information. We already know the correct organ of each shape and are much more interested in the per-organ variations. We, therefore, decided to perform an individual analysis for each group of organs. This also leaves the option open to add new organ shapes, without the need for recalculating the whole shape space.

4.1.3 Registration

According to Kendall et al. [Ken77], if one can *filter out effects resulting from translations, changes of scale and rotations and declare that shape is "what is left"*. For this reason, most shape analysis methods employ some kind of registration strategy that tries to align the elements in the cohort in such a way that they are most similar to each other and the only remaining differences between them stem from shape variations. Generally, the goal of this step is to remove any factors like position, rotation or other things that we are not interested in understanding before the data is analyzed [HK15]. In addition, this also reduces the amount of information necessary to be modeled in our statistical analysis. The question we must, therefore, ask is: What kind of registration preserves the information we are interested in best?

The simplest approach is to use no registration at all, in which case all elements in the cohort are assumed to already share a common frame-of-reference and that any differences in placement are intentional. It is also a valid option in cases where the data is already registered beforehand. Applying a rigid transformation to the underlying data is done by defining both a fixed image or volume and a moving one. The latter image then is translated and rotated in such a way that it overlaps as good as possible with the fixed image [AF07]. Estimating the ideal position can be done in a multitude of ways, either by using characteristics like the center point and size or by trying to match similar values from both images and move them on top of each other. An affine transformation works similar to the rigid one, although instead of being restricted to only translation

and rotation it also allows the transformation of the image scale and enables shearing [FA96]. This allows for even more precise image alignments while introducing distortions to the moving image. Common algorithms extract interesting features in both images and match similar features to one another, creating a transformation matrix [DSTA95]. All of these approaches are shown in a simple example in Figure 4.3

Based on the data we are provided, and the shape information medical experts expect to extract from the cohort, we are interested in analyzing said methods based on several aspects: Most importantly, the ability to model the *general variabilities* a certain organ undergoes, independent from other influences like organ position. Apart from structural shape variations, we also want to analyze *volumetric changes* and need a method that allows for this. As we are dealing with medical data, the chosen registration algorithm must not introduce *artifacts*, i.e. shape variations that are not present in real humans. Finally, we also need to determine the *reference* to which the images shall be aligned to.

Starting off with the case of simply using *no* registration, the volumetric model would largely focus on the change of position in the organ and model these as they have the largest impact on the data. This would also introduce artifacts if a point between two shapes is reconstructed, even if the two shapes look equal and are only moved apart from each other [HK15]. The statistical analysis would place them apart from each other and the reconstructed point would probably resemble a merged version between the two shapes. Under the assumption that two bladders are placed at their corresponding position in the body, this approach does not hide changes in volume. In addition, this method has no need for defining a registration target towards which the other images are moved.

Applying a *rigid* transformation to our volumetric data would lead to centering the objects inside each other, the resulting analysis would be largely independent of positional variances and would focus on shape and especially volume variations. A problem that might occur with this method is its tendency to equally spread out variations across the whole shape. As the motion of organs inside the body is restricted, volume-based shape changes might only lead to growth in one direction. By using a centroid based alignment strategy, these changes would falsely look as if they occur across the whole object [PF03, CGL⁺10]. In regards to the registration target, any cohort element would suffice, for our case, where only a translation is required simply centering the images in the middle of the volume would be a good option.

Affine transformations are probably the method best suited for focusing solely on shape variations as they tend to remove all influences based on translation, rotation, and scaling, only leaving actual shape variations [PBTM06]. This perfect alignment removes all knowledge about the volume of the organ and only allows the analysis of small local changes. Under the assumption that the registration is able to assign the correct parts to each other, this method produces the best results of possible shape variations with little restoration artifacts. However, the problem is what should be used as the base shape, because a poorly chosen deformed version may lead to unnecessary transformations of the moving organs. Therefore an estimated mean shape is used.

While each of these methods has certain benefits, we can not say that anyone of them is a good solution for the analysis of organ shape variations, especially if the deformation through volume changes should be analyzed. We, therefore, propose a method that only partially applies a similarity transformation, which is a rigid transformation without rotation. This leaves some parts of the cohort in their initial frame-of-reference, meaning that for them no registration is performed. The reason for this being that the organs themselves are already pre-registered on a per-patient basis by medical experts, meaning that all timesteps of a patient share the same space. Therefore we leave all organs of a single patient at their relative position to each other, only correcting small translative changes based on the center point of the CTV or prostate, both of which are considered to have little to none shape changes themselves. All patients are then aligned with each other by estimating the mean center of all organ positions across time. Although this procedure can add small translative variations, it also preserves the volume changes and their main growth directions. The effects of all different registration methods for a concrete patient are shown in Figure 4.4.

4.1.4 Vectorization

Based on the fact that most statistical methods used for volume analysis require the data to be arranged in a two-dimensional matrix with the observations as rows and the features or variables as columns, we need to find a way to transform our cohort in such a representation. Firstly we need to unfold each individual timestep currently represented as a 3D volume in such a way that it can be represented as a 1D row vector. Secondly, we also need to unravel the overall cohort structure of patients and their individual timesteps into a column vector with all timesteps being placed one after another. Therefore we are trying to find good sampling strategies for both the volume unfolding and cohort unfolding. We look into two different curves: scanline curve and Hilbert curve [Hil35]. By sampling these curves we are able to transform higher dimensional spaces into 1D ones.

The most common approach for unfolding a high dimensional structure like a matrix to a vector is the scanline approach. The elements are traversed in the order of their appearance, in case of a volume, this would mean going through the contents slice by slice and for each slice iterating over each element in a row-wise fashion [WFG⁺19]. After one slice is finished, one jumps to the next slice. As a fractal structure, the Hilbert curve has a more complex scanning behavior. It can be best understood by following its creation process, which starts with a simple U shaped element [MJFS01]. The first steps of its creation are shown in Figure 4.5. This process can also be extended for higher dimensions and ensures that the line visits each element of the grid at least once if its side lengths equal 2^k , with k being the iteration level of the Hilbert curve [But69].

One of the factors we take into consideration is the preservation of the *semantic structure* in the unfolded data., e.g. are patient timesteps in order or are previously close volume points also close to each other. Furthermore, the *scalability* of the sampling method is also of importance to us, especially the behavior when appending new elements or

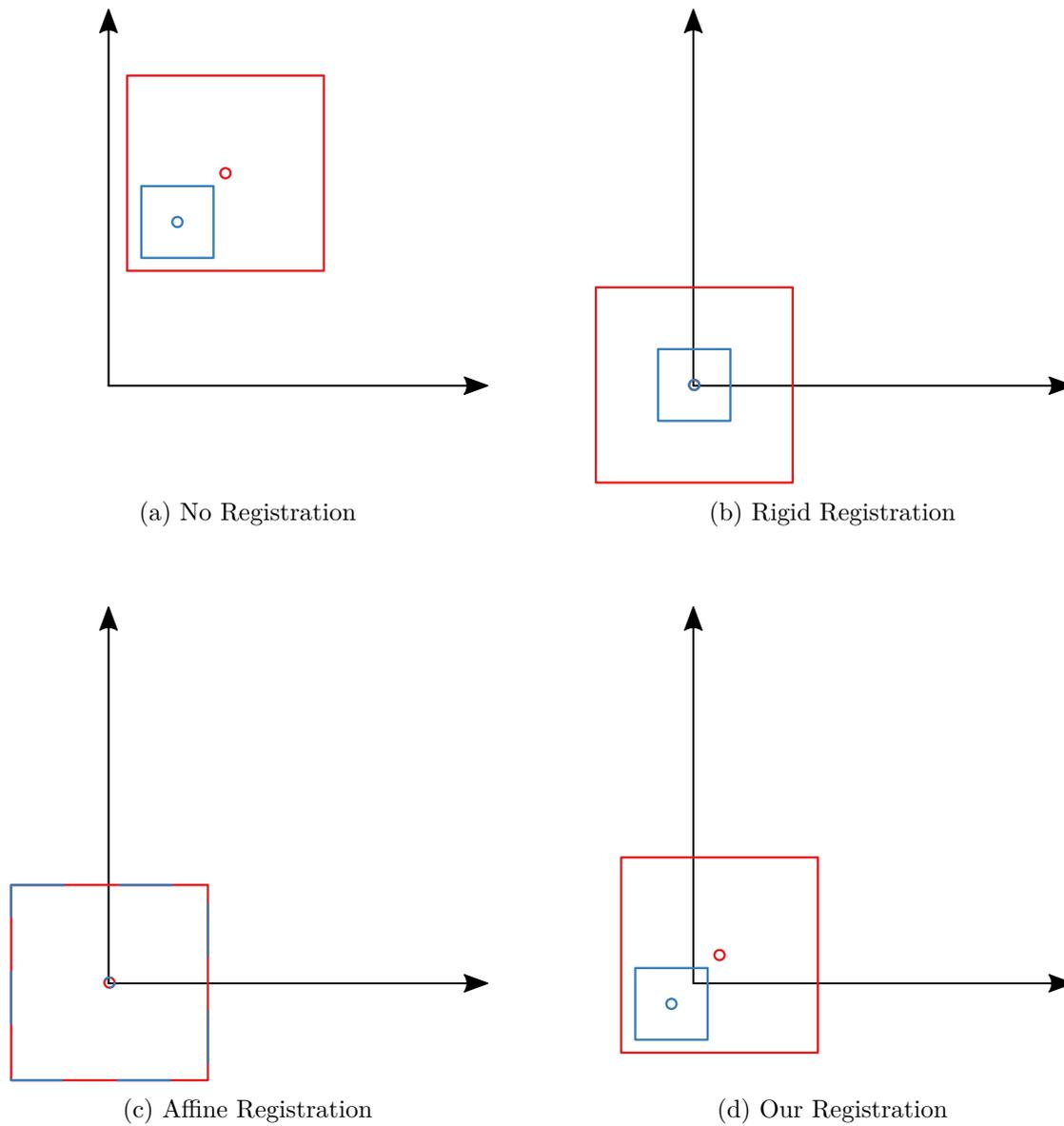


Figure 4.3: Comparison of four different registration methods on a simple 2D example, with the goal being to align the two boxes as good to each other as possible and center them at the origin.

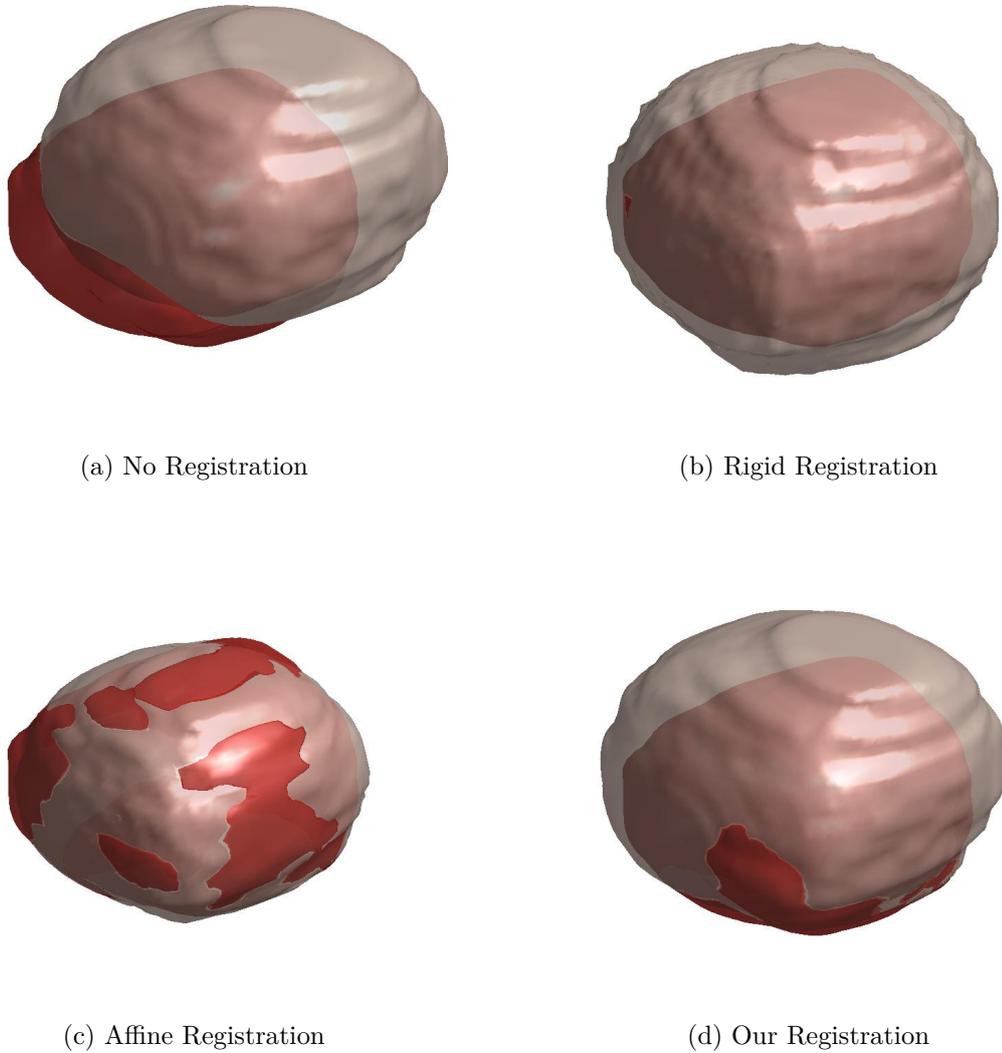


Figure 4.4: Comparison of four different registration methods on the example of two bladders from a single patient at two different points in time. Looking at 4.4d shows the benefits of our method as it neither overly centers the shapes as in 4.4b and does not change the initial volume as in 4.4c.

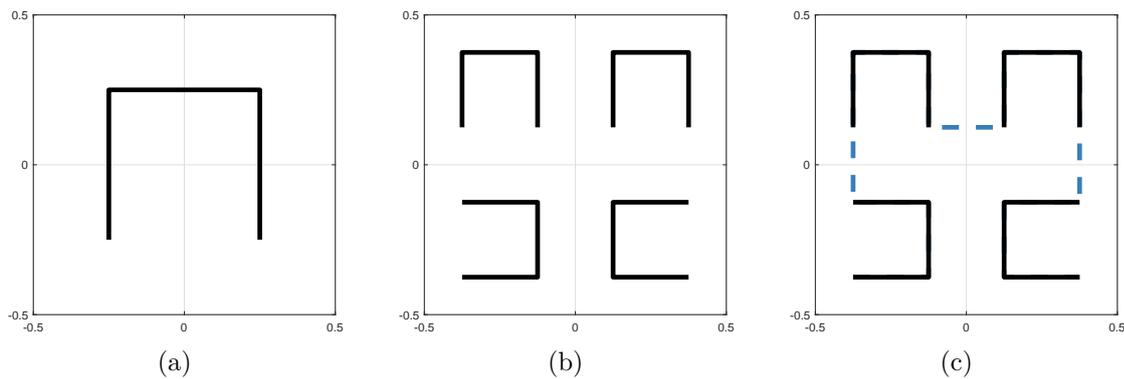


Figure 4.5: A complete iteration includes the following steps: The initial element 4.5a is shrunk to half its size, after which it is replicated four times and placed at the positions and rotations indicated in 4.5b. These initially separated elements are then connected leading to the curve in 4.5c. This element can then undergo the same process to generate the next level of the Hilbert curve [Sag12].

changing the overall number of elements. If desired, there should be a way to change the *sampling frequency* in a way that the aforementioned properties are still preserved. Overall we are also interested in knowing how easy and *efficient* the sampling positions can be estimated.

The basic uniform sampling of the *Scanline* method, as shown in Figure 4.6a, ensures that locality is preserved in the scan direction, meaning that only neighbors along the first dimension are preserved. If the scanline reaches the end of a line a jump follows which resets it to the start of the next line, which results in the large distance between start and end elements [WFG⁺19]. In general, this method scales rather well as its calculation is not bound to any restrictions in regards to the size and aspect ratio of the volume. In addition, the sampling rate can simply be decreased to create varying numbers of lower detailed sampling densities, for example, by only selecting every second or third element. Yet one has to consider alignment issues in other volume slices. Computationally the sampling points are easy to calculate as they are created by iterating along the dimensions, after reaching the end of one the next dimension is increased by one step and the lower ones start at the beginning [TBK03].

Space-filling curves, like the *Hilbert curve* [Hil35] in Figure 4.6b, have the property of preserving locality, meaning that they iterate over all local points, across each dimension in a cluster like fashion, where they only move on when all close points have been sampled. This means that although neighboring points are not always right next to each other in the vector, they are always close by. The Hilbert curve also lacks the jumps the scanline method has at the end of lines. One trade-off is that the method is only defined for spaces of 2^k size. Although one could just scale them to match the aspect ratio, the sampling rate is constant along each dimension [But69]. This also affects the options to create several levels of detail, as the sampling rate can only be doubled or halved equally

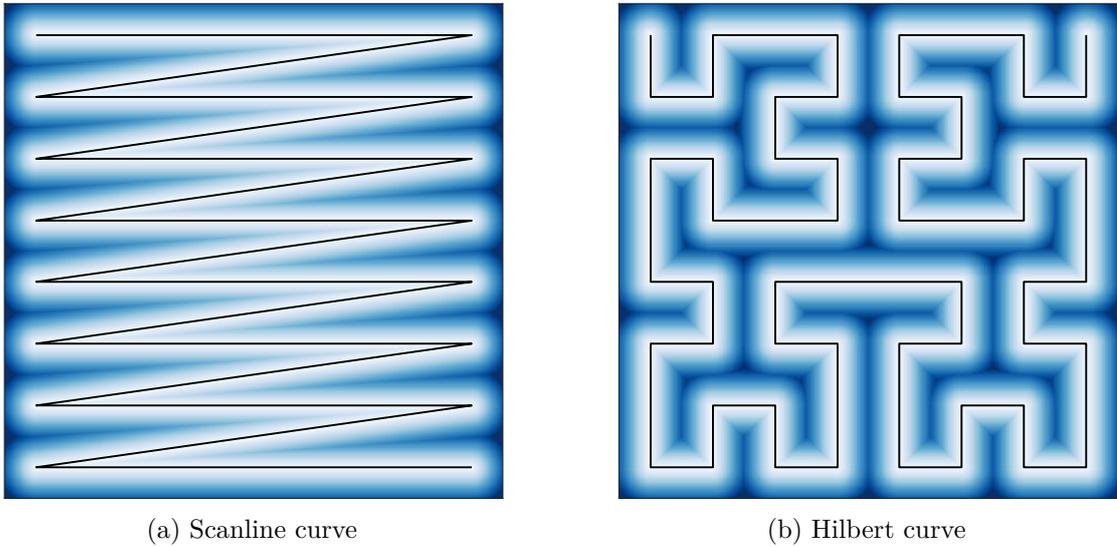
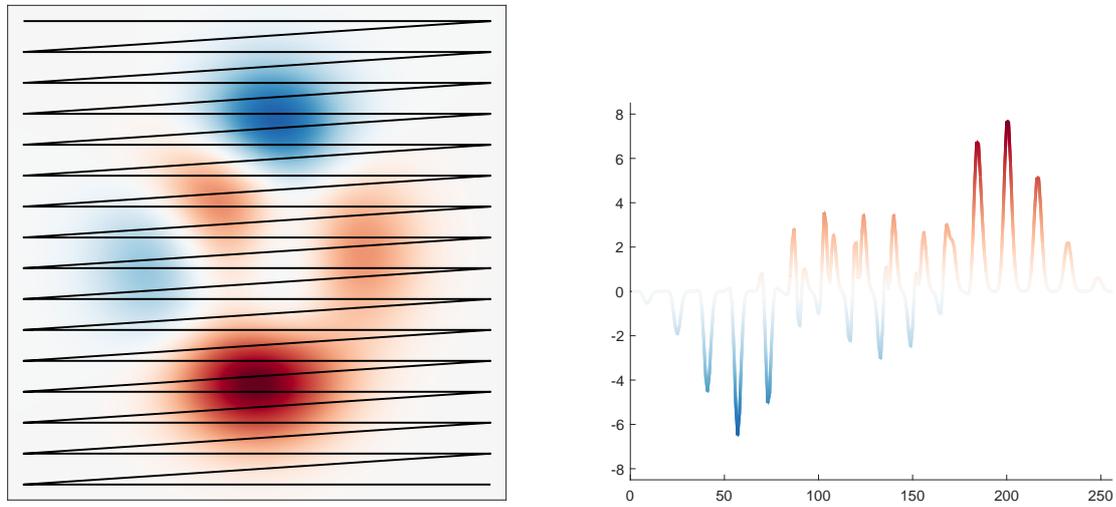


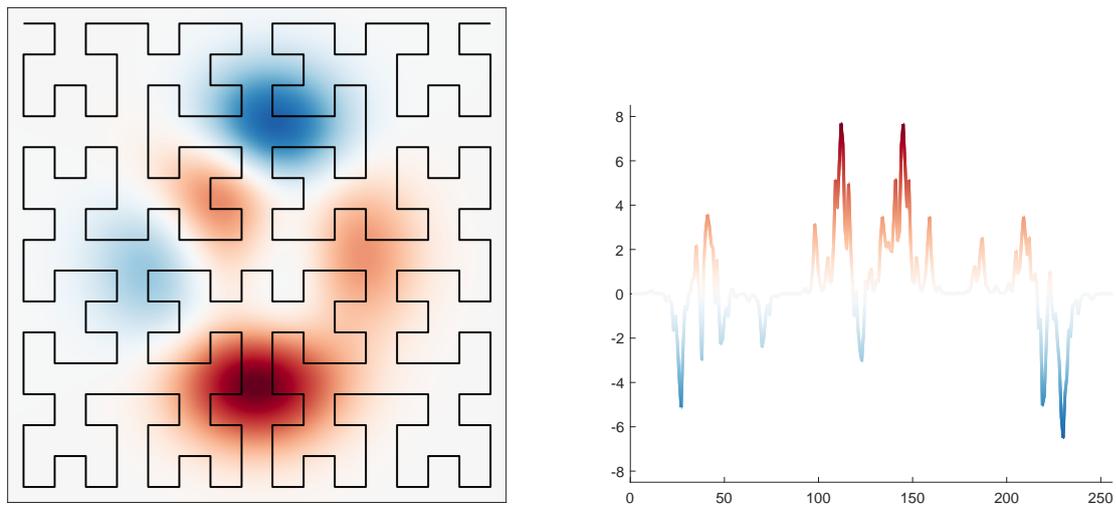
Figure 4.6: General curve patterns for a row-wise scanline curve and the third iteration of a Hilbert curve. Both of which sample the underlying space in an 8×8 pattern. The color encodes the distance to the nearest point on the curve.

across all dimensions. However, all discrete sampling levels, still cross the volume in the same fashion, reducing the risk of alignment errors. The computation of individual sampling positions is expensive as it is an iterative process where for each level k , all previous $k - 1$ levels have to be estimated first [KEB99].

Based on this information, we selected the scanline approach to unravel the individual observation data for patients and timesteps. The statistical analysis in the next step has no need for knowing which data points belong together. Nonetheless, we are interested in preserving the temporal connection, i.e. which timestep comes before which one. By using this approach and sampling across the temporal dimension we can create a simple 1D vector where all timesteps of the first patient are followed by the timesteps of the second patient and so forth. Doing so not only allows us to easily select certain patients and their timesteps by using multiplication, but we can also efficiently add new patients to the data, by appending them in the vector. For the unfolding of the individual 3D volumes, we chose to select the Hilbert curve. This does not have any computational implications for our results. Instead, we used it to analyze how the shape differentiation capabilities of our approach change if the sampling density is reduced. Furthermore, we also looked into different options, based on the work of Weissenbck et al. [WFG⁺19] to use an unfolded 1D line to show differences and overlaps in volumetric structures, which we left open for future improvements. An example of this unfolding process can be seen in Figure 4.7.



(a) Scanline Curve



(b) Hilbert Curve

Figure 4.7: The same density field is sampled and mapped from 2D to 1D, once using a row-major Scanline curve and once using the Hilbert curve. The resulting lines show that the locality preservation of the Hilbert curve leads to fewer jumps and to a closer placement of neighboring parts.

4.1.5 Analysis & Dimensionality Reduction

After the initial segmentation contours are transformed into 1D vectors they can be analyzed. The fundamental goal of this step is to not only transform the initial cohort data into a representation where underlying structures and patterns are made visible, but also to create a lower-dimensional embedding allowing us to create a memory cheap way to store and process large cohorts of patient data. There is a huge variety of different feature extraction methods that try to do just that, including Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Autoencoder Networks, and t-Stochastic Neighborhood Embedding (t-SNE), to name just a few.

Principal Component Analysis is a statistical analysis method that tries to derive new features from the initial ones by linearly combining them in a way that variance along the new features is maximized. Two important factors in this creation are that the new features are uncorrelated and that they are ranked in the order of the variance they encode [Sh14]. A similar approach is followed by the Linear Discriminant Analysis, but in contrast to PCA, which is an unsupervised learning method, LDA is a supervised one, using labeled data. Instead of maximizing the variance encoded in each feature like PCA, LDA tries to maximize the separability between the classes of the data [DHS12]. Both methods allow the creation of a transformation, from which a subset can be chosen to create a dimensionality reduced representation, which maximizes their respective properties. Autoencoders are different from the previous methods based on the fact that they are a neural network-based approach, which calculates non-linear feature combinations. Their basic mechanism is that they try to recreate their original input, all the while their hidden layers have fewer neurons than the input and output [VLBM08]. A simple version can be seen in Figure 4.8. The network tries to model the output in such a way that it matches the input as closely as possible. Extracting the values of the few neurons in the middle gives a lower-dimensional embedding. Another approach trying to model non-linear relations in the data is t-SNE. In a two-step process, the algorithm first calculates a probability distribution between neighboring points and then tries to recreate this distribution in a dimensionality-reduced lower space as good as possible [MH08]. As with many machine learning algorithms, this process is iteratively optimized until a sufficient solution is reached, where the non-linear relationships of the high-dimensional space are reflected. These four methods are by far not the only techniques that can be employed in this case. We restrict us to analyze those methods because they represent a mix of linear and non-linear, as well as supervised and unsupervised techniques, several of which were already employed in other works in this field.

For deciding which of these techniques is the right option we look at several indicators. As each method is based on a variety of mathematical calculations, several of which make *assumptions* on the input data, we, therefore, need to analyze, which of these are fulfilled for our case. The calculation process often requires the creation of large matrices, which might not fit in the memory, or the process itself may take a long time to finish. Therefore, we need to not only know the general *computational requirements*, but also how the approaches scale with more patients and timesteps and even if additional

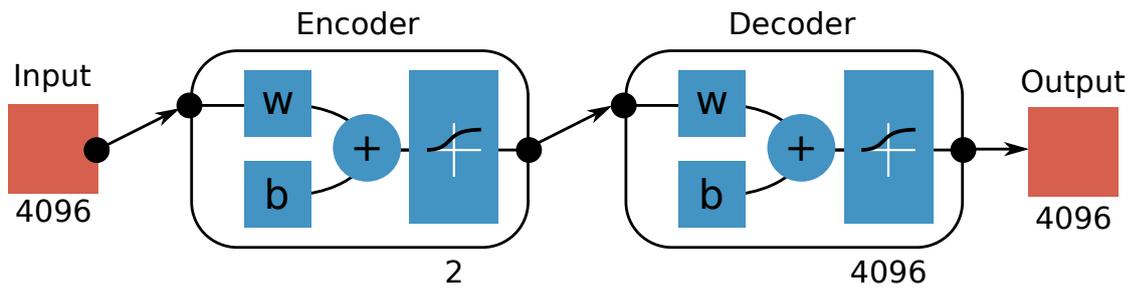
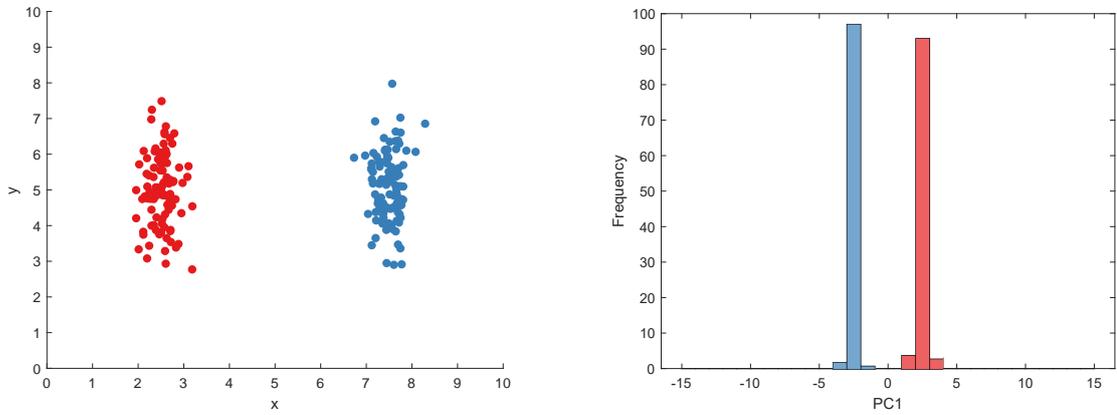


Figure 4.8: Basic structure of an Autoencoder network, consisting of the encoder, which transforms the input image down to a previously defined number of output values, and the decoder which takes said output values as input and tries to recreate the original image as good as possible. Each element consists of the same components: The weight matrix w , the bias b , and an activation function.

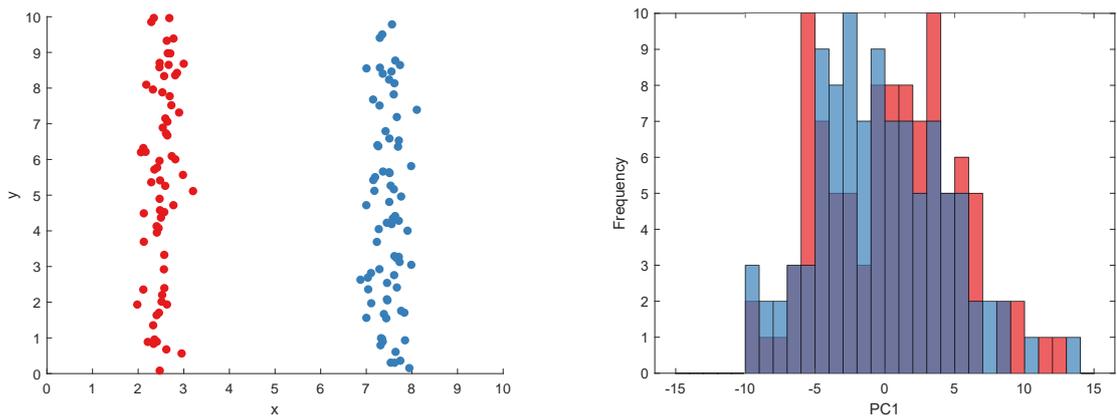
elements can be added later on. Last but not least, as we deal with medical data the result of the analysis should be *reproducible* and *deterministic*, meaning that we get the same result for consecutive computations. Finally, we are interested in knowing if elements can be *synthesized*, allowing us to save the reduced versions and only re-create the original shapes on demand.

PCA is based on the assumption that the data follows a Gaussian distribution and that the direction of greatest variance is the one containing the most information [Sh14]. Although this is often true, it can lead to misleading results if the inner group variance is higher than the one between groups, due to noise (as seen in Figure 4.9). The general computation of the principal components is based on calculating the eigenvalues of the covariance matrix derived from the original data. The estimation of this matrix is often a problem as it has the squared size of the number of points, which can result in several gigabytes of data for an observation with 2.097.152 values created from a $128 \times 128 \times 128$ volume, especially if multiple organs are analyzed. It should be noted, that there exist different implementations that tackle this problem, for example, by making the covariance matrix size depend on the number of observations rather than dimensions or by not computing the covariance matrix at all [Row98]. Although, PCA is sensitive to outliers in the data, it produces for the same input dataset always the same output and is, therefore, deterministic [CLMW11]. The result of PCA is a matrix of all uncorrelated linear combinations of the initial features, from which a subset can be chosen to project the data into a lower-dimensional space. The number of selected features determines how much of the initial variance, and possibly detail, is preserved. This process results in the creation of a transformation matrix which can be used to quickly transform points into the reduced space, but also allows one to reverse this transformation creating shapes from various points [DLMB04, Sh14].

LDA makes similar assumptions to PCA in regards to the distribution of observations, but also requires explicit knowledge about which observations belong together, as it is usually employed in supervised learning [DHS12], an example can be seen in Figure 4.10.



(a) **Intergroup Variance: High** **Intragroup Variance: Low**



(b) **Intergroup Variance: Low** **Intragroup Variance: High**

Figure 4.9: Two examples illustrating how PCA works and showing how data variance affects the results. As PCA determines the principal components based on the dimension encoding the most variance, the data in the two groups in the first example can be easily separated as the difference between them is larger than the one inside each group. This assumption can lead to the problem shown in the second example, where the elements inside the group experience far larger variations and PCA fails to separate the groups appropriately.

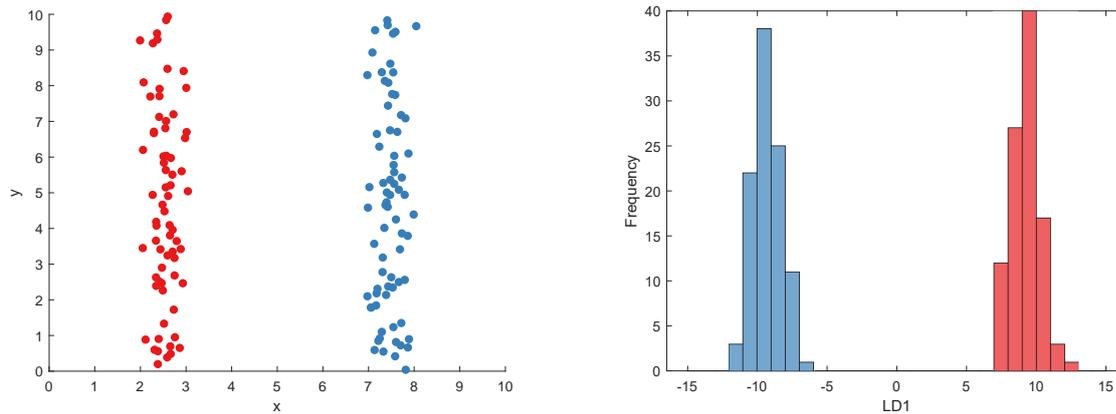


Figure 4.10: An example illustrating how LDA works and how it does not suffer from the restrictions as PCA. Contrastingly to PCA, LDA does not solely rely on the variance present in the data but also has access to the correct group labels. Therefore it can solve cases, which PCA cannot, namely: **Intergroup Variance: Low** **Intragroup Variance: High**

LDA needs a defined set of labels to work with, in our case this could, for example, be the patients with and without induced toxicity or what kind of organ a shape is. An implicit assumption made by the algorithm is that the class sizes are roughly equal and the inner class distributions are also Gaussian [MK01]. A property in which it is similar to PCA is that its performance and memory consumption grow strongly for high-dimensional feature vectors. It also creates matrices with squared dimension sizes for both inner and between-class variance. Furthermore, it requires $O(n^3)$ operations to estimate the eigenvectors for its transformation matrix [GRM15]. The algorithm itself is deterministic and only varies if the input is changed. The resulting transformation does not provide the option to manually select a subset of features and decide the size of the created lower-dimensional space, but instead always reduces the dimensions to the number of classes minus one [McL04]. Similarly to PCA, LDA also provides the reduced points in combination with a transformation matrix as a result. The matrix can be used to transform new observations into the reduced space and to transform points from this space back to the original one, allowing for shape synthesis.

The next option we evaluated was the use of an *Autoencoder* neural network. Different from the previous two methods it requires an underlying neural network data structure, consisting of several layers of neurons [VLBM08]. The training of these networks is performed using backpropagation, where the reconstruction error of the network is used to modify the neurons of the network back-to-front. This process is often computationally expensive and it can require multiple training iterations until the network provides sufficient results [LBH15]. It has been shown that using networks pre-trained for certain image data can speed up this process. While the computation process might be rather lengthy, this method does not have the same scalability issues as the other approaches.

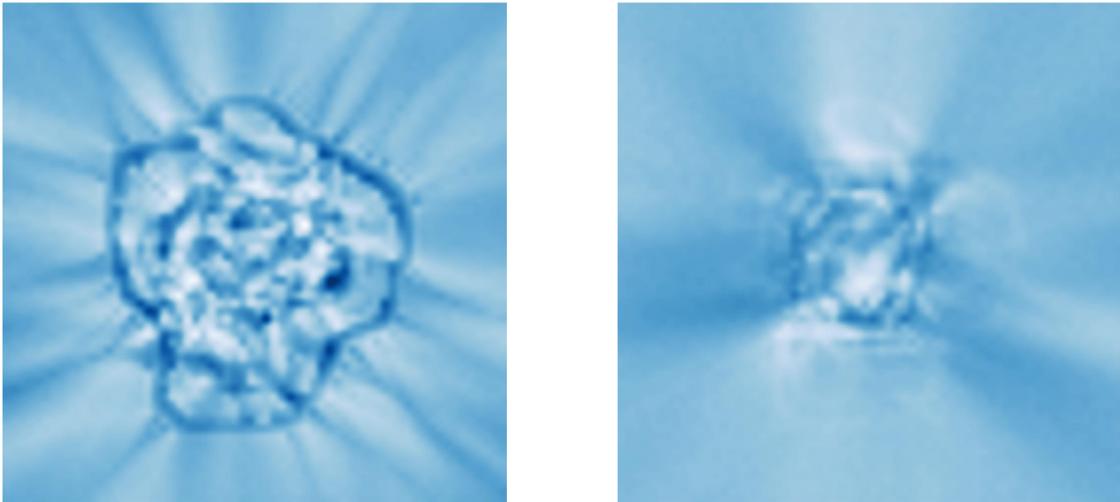


Figure 4.11: By training the Autoencoder network in Figure 4.8 to analyze images of cross-sections of our organ shape cohort, we analyzed how these networks work in principle by looking at their weight matrices in their centermost layer. The two shown images indicate which regions are deemed important. We can see the combined shape of rectum and bladder in the left image and a rough indication of a prostate shape in the right image.

Although it has similar problems with large dimension sizes, the quality of its output increases, the more observations are provided [EBC⁺10]. However, it does not require the whole cohort to be loaded into the memory at once, but it instead can use only one portion, i.e. batch, of the cohort at a time. If the network does not have any pre-trained weights to start with, it is initialized with random values, leading to non-deterministic behavior, meaning that some instances may result in suboptimal solutions. Similarly to PCA, the number of dimensions in the reduced space can be manually chosen by setting the number of neurons in the smallest center layer. The weights of these layers can even be visualized as seen in Figure 4.11. In contrast to PCA, one cannot freely select more or fewer dimensions, as this would require a new training process [MSJ⁺15]. The reconstruction of elements is rather straightforward as the general purpose of the network was to generate these elements with optimal quality. The only restriction in this regard is that, to achieve reasonable performance, the observations used for training and reconstruction are rather small and result in low-resolution shapes.

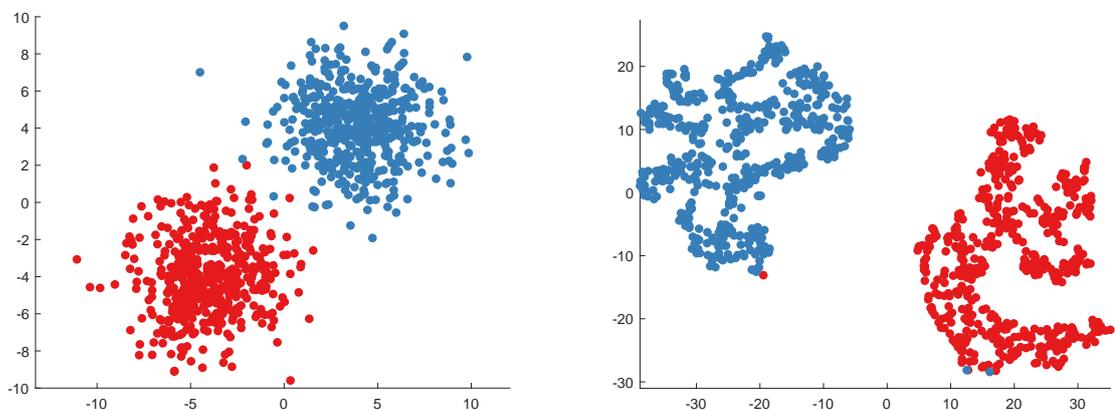
t-SNE is not a mathematical technique like PCA or LDA, but instead a probabilistic, non-linear approach, which tries to model complex data distributions [LBH15]. Although *t-SNE* creates a non-linear projection it still assumes that the distance between points is linear as it uses the Euclidean distance [MH08]. The process is iterative and therefore computationally expensive. This means, that for datasets with large numbers of dimensions another dimensionality-reduction technique like PCA should be employed beforehand [VDM14]. Recent advances like hierarchical *t-SNE* seem to alleviate this

problem [PHL⁺16]. The overall projection process normally has multiple possible local minima, which means that consecutive runs can lead to varying results, without a guarantee that the algorithm has found the optimal solution. The quality of the solution, i.e. how well groups of similar elements are identified, depends largely on the chosen initial parameters of the algorithm and inadequate choices could even lead to false-positive results [WVJ16], which can also be seen in Figure 4.12. In contrast to the other methods, t-SNE only provides the positions of the embedded points and does not offer a direct transformation matrix to add new elements into the space. These axes of the created space also lack an inherent meaning and have no connection to the initial features [MH08]. This also means that the reconstruction of shapes from a point in the space would be both meaningless and impossible.

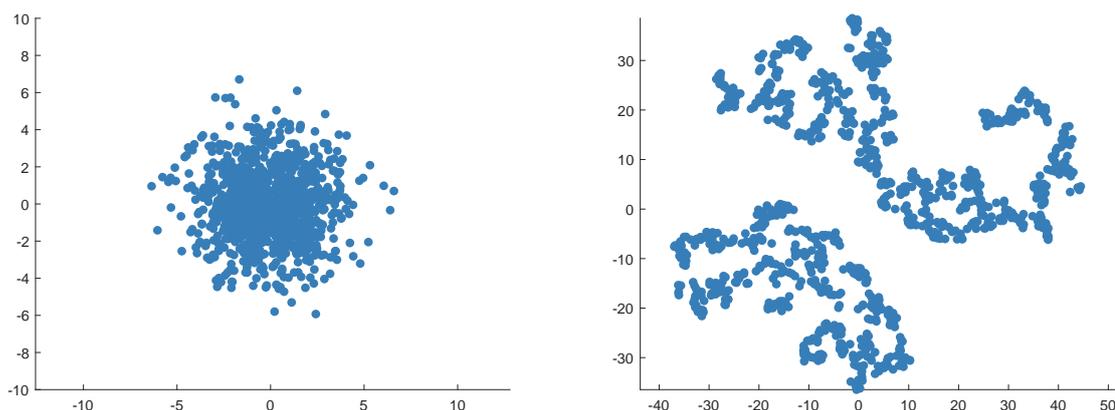
For the analysis of the shapes present in the cohort, we decided to use PCA over the others. We assume that the data is normally distributed and meaning that this analysis tool can be employed. We are also more interested in the global structure of the data within organ types than the overall differences between different organ types, a simple example of the latter is shown in Figure 4.13. A joint shape space of multiple organs, could be investigated in future works. Furthermore, we think that PCA has the best capabilities to deal with the far higher number of dimensions when compared to the number of observations. Due to the relatively small size of the cohort, we did not use a covariance-free version. Using this version could be an option for future extensions, especially if more patients are added to the cohort. Most importantly our decision is supported by previous works of Ferstl et al. [FKRW16], who have also shown in their work the benefits of analyzing shape variability using PCA, as it allows for precise selection of the desired reduction and reconstruction quality. For the identification of groups inside the cohort we do not only rely on PCA, but instead combine its results with t-SNE. It has been shown that these two methods work well in conjunction and allow for a better identification of clusters, as t-SNE analysis more complex, non-linear, relationships [RBGR18]. We discuss the use of t-SNE in more detail in the following section. In general, we found that it produced more pronounced groups of similar patients than other approaches, which results in more stable groupings.

4.2 Cohort Visualization

With the ability to compare different shapes of the same organ to each other, we can now start trying to solve the task of analyzing the whole cohort. The general goal we ought to achieve is allowing researches looking at a large collection of patients and timesteps, to precisely identify patients or organs of interest. We try to make only few a priori assumptions to what might be of interest for a researcher. However, we already provide certain measures based on previous works. Most importantly as the cohort visualization is only one part of an entire analysis tool, we define its main focus on providing both an overview of the whole cohort as well as allowing users to select parts of interest for detailed shape inspection.



(a) Two separate random distributions



(b) One random distribution

Figure 4.12: Two examples of how t-SNE (Iterations: 1000, Perplexity: 20) transforms an input set of random points. In the first case the algorithm is able to clearly separate the two groups ■ ■ from each other, creating two clusters of points. The second example shows a similar result as it also indicates two groups in the data, although the initial data just contains random points from a single distribution ■, highlighting the possibility of false positive results.

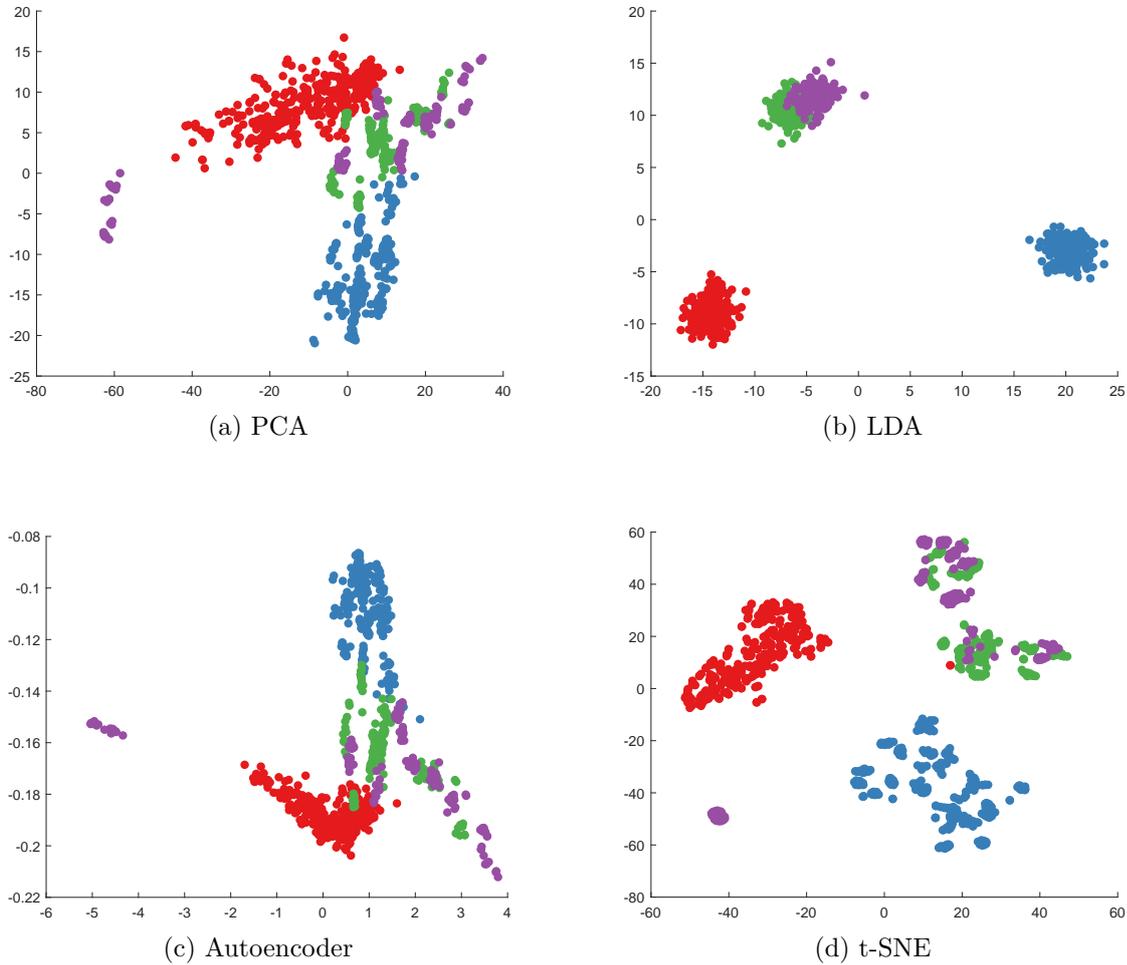


Figure 4.13: ■ Rectum ■ Bladder ■ Prostate ■ CTV Comparison of four different analysis methods applied to the volumetric vectors of all organs of all cohort elements, with the idea of showing how well these methods are able to detect similar organs. Both PCA and the Autoencoder network show similar results, where the organs are spread across the whole space and similar organs are close to each other. LDA and t-SNE are able to show more pronounced groups of organs, hereby it is important to know that LDA was provided with the organ labels initially. In most cases the outlier CTVs, resulting from extremely large treatment volumes filling the whole bowel, are clearly visible.

4.2.1 Overview First

Following the information seeking mantra of Ben Shneiderman [Shn03] the first step we undertake is to provide users with an overview of the whole cohort data. The main idea behind this step is to create a zoomed out, high-level representation. That should convey the general patterns present in the data before the user starts her own more in-depth investigation.

Similarity Estimation

The dimensionality-reduced space creates a representation where each cohort data point is represented by one position in space, where similar shapes are placed close to each other. We are still left with the question what the best way would be to analyze this space, as we strive to provide users with information about the general similarities between shapes and shape groups but also about how an organ shape changes during the treatment process. One possible task requiring this could be the search for treatment instances, where the bladder filling protocol was violated, e.g. the patient had an empty instead of a full bladder. We, therefore, look at three different options allowing us to estimate similarity in one way or the other:

The straightforward way would be to directly compare the positions of shapes in their respective shape spaces. As mentioned before, these shape spaces are created on a per-organ basis by transforming the high dimensional 3D volume vector with $2.097.152(128 \times 128 \times 128)$ dimensions into a lower-dimensional space where each shape is represented by a point. The number of dimensions kept, indicates the amount of detail preserved. In the case of pelvic organs, we need somewhere between 5 to 20 dimensions, depending on the specific organ, to preserve 95% of the shape details. Instead of encoding the point positions directly, another option could be to use a distance metric, to estimate how far certain points are from each other, hereby N dimensions are summarized into one [DD09]. The result can be seen as a measure of similarity. Instead of reducing the point to a distance value, the similarity of all observations could also be estimated by clustering them in groups based on the predominant shape types in the space [Jai10].

Our decision on the best way to tackle the tasks we are facing is based on several factors: Overall the approach should allow a *comparison* of similarity *across timesteps and patients*. An expert should be able to understand how a shape varies across multiple treatment instances, and how common such behavior is for patients. Moreover, it should allow the identification of *outliers*, for example, where a bladder assumed an unusual shape, in relation to the overall shape of a single patient or in general. Finally, the approach should scale appropriately if *more dimensions* are added to the input data, for example, if more principal components are used for the shape space creation. Generally, we also want to reflect upon the extendability of the approach for *multiple organs*. The scaling behavior also extends to the *visualization* itself, as we need to know the number of dimensions the output has.

Using the shape space position of the organs means that the similarity is encoded by

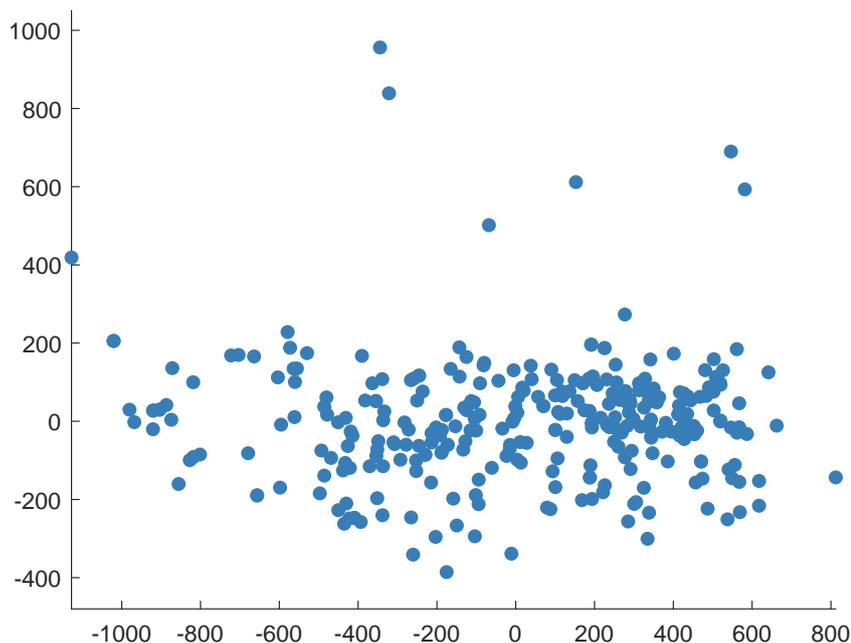


Figure 4.14: Scatter plot of all bladder shapes in a 2D shape space. The similarity of certain shapes is implicitly encoded by their respective proximity.

their *coordinates*, i.e. a 5 to 20 dimensional vector. A simple version can be seen in Figure 4.14. Our shape spaces are still larger than the two or three dimensions we are used to. A problem with high dimensional spaces is that the elements inside them are usually spread out and judging the proximity of elements to estimate their similarity or find outliers gets quite hard. To employ an often-used example finding a single coin on a line of 100 metres does not take long, finding it in a field of $100m \times 100m$, i.e. a 0.714 football field, can take quite a while and finding it inside a volume with a side length of $100m$, i.e. a 45 stories building is nearly impossible [AHK01]. In terms of scalability, as the input and output values are the same, their dimensionality would also grow or shrink with the used principal components [Shl14]. In addition, we could compare more than one organ by joining their high dimensional positions. For this case, it should be noted that most probably different organs have varying numbers of dimensions. This would result in a larger joint shape space position-vector. The employed visualization methods need to process the output of this method, which most certainly ends up with observation vectors that have more than three dimensions.

Using *distance* functions enables a more direct encoding of similarity. It allows for the comparison of two points of arbitrary dimension and transforms them into a 1D value, which describes the distance between them, which is a similarity indicator in shape spaces. Two choices need to be made when using such a metric: the formula used for the calculation and the target point we compare a point to. There is a huge variety of different ways to calculate a distance metric, with varying properties, making the right

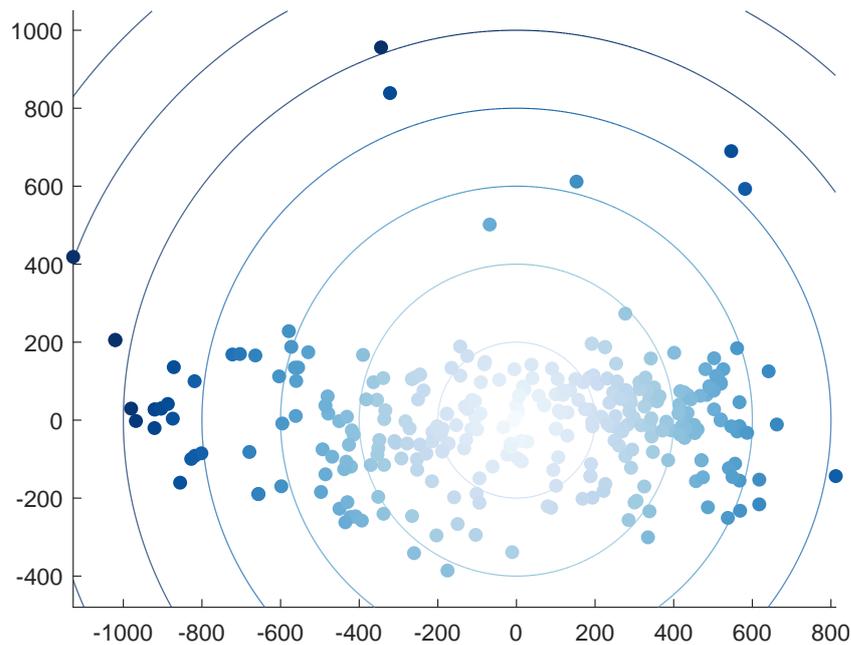


Figure 4.15: Scatter plot of all bladder shapes in a 2D shape space. The similarity of certain shapes is encoded in the distance to the mean shape in the center. The coloring  and rings represent the distance to this point, where dark blue denotes increasing distance.

choice important to ensure valid comparisons [DD09]. Furthermore, while it would be possible to estimate the distance from each point to each other one, this would square the number of values. Therefore a valid target point needs to be chosen for each point to which its distance is estimated. Possible target points could include the mean or median of the shape space or the patient, encoding the difference on a global or a per-patient basis respectively. An example of the distance to the mean estimation using the Euclidean metric can be seen in Figure 4.15. In this representation, especially outliers would be more easily detectable, as they would have an unusually high distance [ZSK12]. Even if the number of dimensions increases this mainly affects the number of calculations, as the resulting distance value is always one dimensional. One could extend this approach to multiple organs, by concatenating their coordinates. The probably different numbers of shape space dimensions would lead to an unequal weighting in the distance calculation. Therefore distance measures should only be calculated on a per-organ basis. The following visualization approaches only have to deal with a single value per cohort element and organ to visualize.

In many ways *clustering* may deliver similar results as the distance calculation, as it also takes the high dimensional shape space embedding of the cohort elements and transforms them into one value per cohort element and organ. The difference is that instead of measuring the precise distance to a defined target, clustering tries to find groups of

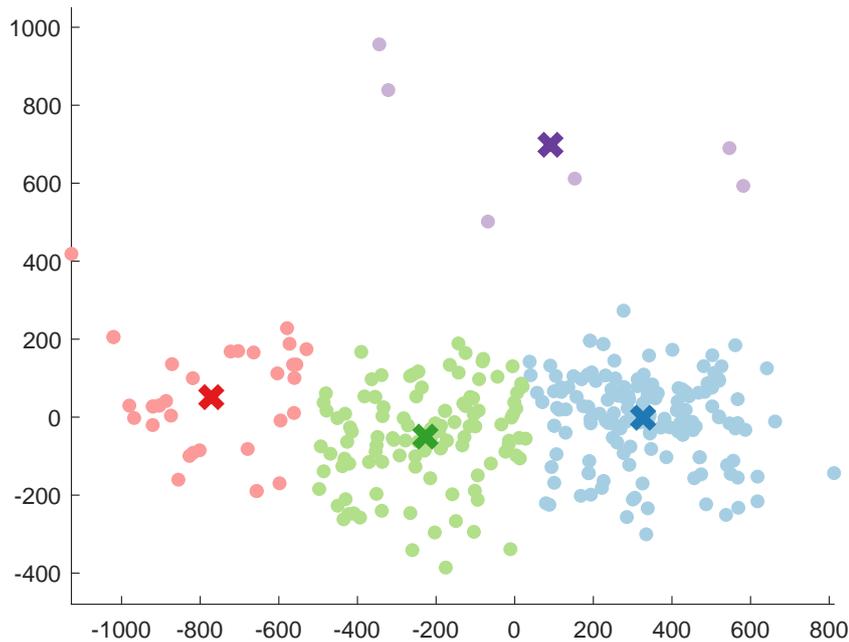


Figure 4.16: Scatter plot of all bladder shapes in a 2D shape space. The similarity is encoded by the cluster each shape belongs to, indicated by the color. The four clusters were found by using k-means, the crosses indicate the respective cluster centers.

similarly shaped organs in the data [Jai10], as seen in Figure 4.16. The similarity would mean in this context belonging to the same group and outliers would be, for example, single timesteps being part of a separate group. To work properly, this process requires similar choices like the distance measure one, including a metric and more importantly a clustering algorithm [XW05]. The latter decides how similar groups of shapes are detected. Increasing the number of dimensions of the shape space coordinates mostly affects the calculation process and not the size of the resulting clustering. Similarly, if combining multiple organs special precautions should be taken that the vector of one organ does not outweigh the ones of the others. Visualizations that aim at showing the estimated shape group categorizations, need to handle one value per cohort element and organ [RCMA⁺18, RBGR18].

In the end, we decided to use two similarity indicators and create visual encodings for both of them. Namely, we employ distance calculation and clustering. We use the former one to directly estimate outliers on a per-patient basis, by measuring the distance to the mean shape of the patient. It also gives a good impression on how much the shape varies across the treatment process of the patient. These subtle differences would not be visible if a clustering would be employed for this task, as this would only offer a binary variability option, i.e. either the shape belongs to a cluster or not. The information, to which shape type a certain cohort element belongs, may be too generalizing for an internal shape analysis for a single patient. Yet, it allows for a better overview of the

whole cohort. Clustering extracts main shape groups of patients, which can be analyzed and compared to one another. This allows users to understand, what shape types exist and how prominent they are. The general similarity of these two methods also makes a combination easy, as both of them deliver a single output value largely independent of the higher dimensional input and they can be visualized in similar ways. It should be noted that we used the Euclidean metric to estimate distances in both cases as it is one of the most common approaches. Its use should in future works be evaluated as many other metrics might provide better results, especially in higher-dimensional spaces [AHK01]. For the clustering, we employed a hierarchical algorithm with complete linkage based on the work of Klemm et al. [KLR⁺13], who already showed its application in the analysis of medical shape types.

Shape-Similarity Visualization

From the previous calculations, we receive a single distance or cluster value per combination of patient, timestep and organ. For a single organ, this would already include three values to encode, meaning that this number only grows, the more organs are inspected. We, therefore, needed to find a way of visually encoding the information to support both the search for highly varying patients, common types of shapes across the cohort, and last but not least, performing all these tasks with parts of the data being missing, as not all organs are present for all patients. With these tasks at hand, we looked into several possible visual encodings, that are commonly used if large amounts of complex data need to be analyzed [FGW02].

Probably one of the more classically used approaches in this area are scatter plots. By drawing the data axes orthogonal to each other, a space is created in which each data element is represented by a single point [Spe01]. Parallel coordinate plots (PCPs) are a commonly employed plot in visual analytics if someone needs to explore a complex data set with more than two dimensions in search of interesting elements or correlations. Instead of plotting all dimensions orthogonally to each other, they are drawn in parallel and observations are represented by polylines traveling across these dimensions [ID90]. By taking the axes of a parallel coordinate plot and arranging them in a circle, the polylines end up looking like stars. This approach is also called a star glyph representation and mainly aims at using our innate perception of shapes to provide us information about the data [War02]. Finally, we also looked into tabular representations, which rely on our knowledge in the use of spreadsheets, we create visual representations of the contained data, for example, by creating a row for each element in the set and use the columns to encode the values in each separate dimension [BBS⁺18].

To judge which of these methods support the visualization of interesting properties in a way that can be understood by medical experts, we evaluated them using a few simple properties. First and foremost the chosen visual encoding needs to *preserve* the *patient* and *timestep information*. The preservation of contextual information is of great importance as we assume that certain shape patterns and variations are characteristic for patients and may also change over time. It should also be possible to extend the chosen

method to more than one organ, i.e. *scalability* to more dimensions. This also includes the need for having the ability to monitor the shape changes for multiple organs of the same patient. Or by formulating this problem a bit more general: Using the encoding one needs to be able to detect patterns and *correlations* across different dimensions, e.g. patient, timestep, and organ. As our cohort already contains a large number of elements, we also need to consider the scalability for *large numbers* of elements, and the handling of *missing information*. We also need to look at the possible *interaction* methods, which should allow users a more in-depth exploration of the data.

Scatter plots are normally only used for 2D data, although in some cases also 3D plots are used. The resulting collective shape of all points often provides a good indication for certain correlation patterns. In general, this approach allows viewers to judge the overall distribution of observations in space [Spe01]. Scatter plots also provide information about possible outliers or groups. This method is largely restricted in terms of what information can be shown and what is lost. One can only take two features out of many and use them to draw the points in the scatter plot. One example could be taking the variability of two organs, e.g. bladder, and rectum. In this case one could see possible connections between the bladder and rectum variations, but the information about which timestep and patient the points belong to is lost. In general, this also limits the amount of information that is displayed, as the number of dimensions, and therefore the possibility to show the variation of multiple organs, is severely restricted in this representation method. One method to overcome this problem is to not only rely on the position in space to encode the information, but also to use other channels like color and shape of the markers, to include information not encoded along the axes, like patient or timestep [KHD⁺10]. Another more common approach would be to create multiple scatter plots, each encoding two out of the complete set of dimensions. This allows the user to see correlations and patterns for every feature combination, as seen in Figure 4.17. A downside of this so-called scatter plot matrix is that it grows quadratically for every new feature, consuming more and more screen space and leaving less space for each plot [Spe01]. The interpretability of scatter plots suffers if lots of data points are drawn, which start overlapping each other. This makes it hard to judge how many points are in one place. Still, there are certain methods like kernel density estimation that try to overcome this problem, by showing how many points are in certain places [Wor89]. Direct interaction methods often come in the form of brushing in combination with linked views. For example, by selecting a region in one scatter plot the chosen elements are highlighted in the other plots of the scatter plot matrix, or even a detailed view may be provided. Tooltips when hovering over points are also employed to provide detailed information.

One of the biggest restrictions of scatter plots and scatter plot matrices is their limited extendability for searching correlations in higher dimensions. This problem is tackled by **parallel coordinate plots** because they allow users to encode larger numbers of dimensions with less visual growth. An example of this is shown in Figure 4.18. Correlations are then judged based on the line patterns between neighboring axes [FGW02]. Although this approach can handle large numbers of dimensions, correlations between

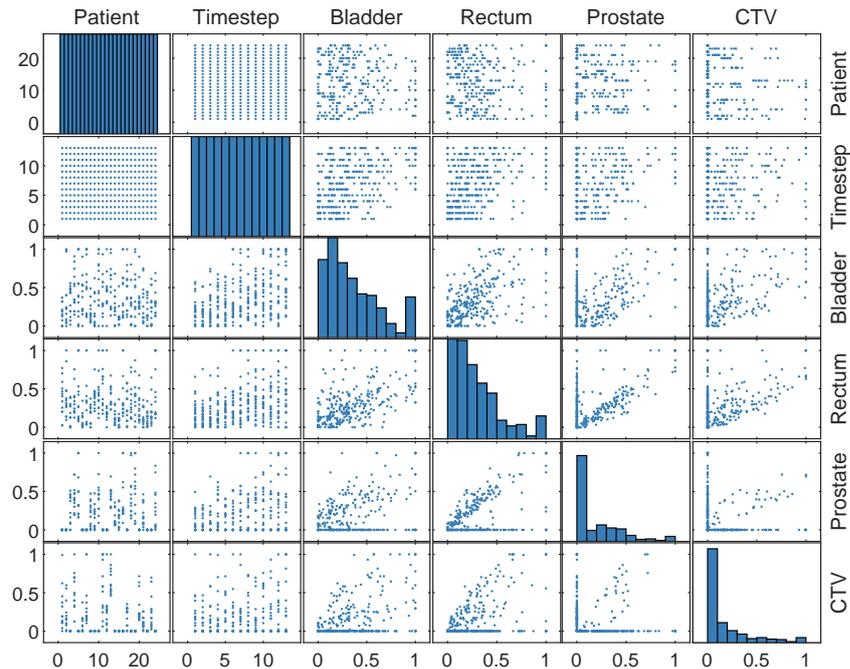


Figure 4.17: A scatter plot matrix encodes the respective shape difference values for all organs and the patient and timestep ids. Each cell takes two of these features and creates a 2D scatter plot. The idea behind this is to uncover relationships between certain features. In the diagonal, histograms of the individual features show their value distributions.

two features are only properly visible if the axes are next to each other. Moreover, recent works have suggested that parallel coordinate plots are harder to interpret for many users than scatter plots, especially if non-linear patterns are present in the data [KARC15]. The encompassing contextual information like patient and timestep id can be encoded as separate axes, but can only be used to provide context information for neighboring axes. Similarly to scatter plots, large numbers of observations can lead to visual clutter that makes the plot hard to understand, as more and more polylines are drawn on top of each other. To tackle this problem, methods were created that aggregate groups of lines in clusters or highlight the density of lines [HW13, JFC14]. For the latter, simply drawing the lines transparently can convey a notion of density. Further information can, for example, be encoded in the line color [GKGR18]. Applying color in these plots can allow users to more easily trace the values of certain points across dimensions and see correlations or patterns concerning non-neighboring axes. In terms of interaction brushing and linking are still often employed methods. For these kinds of plots users can often draw along one axis and filter for polylines that fall exactly inside this value range. This interaction method is an often needed necessity to comprehend large datasets [HW13].

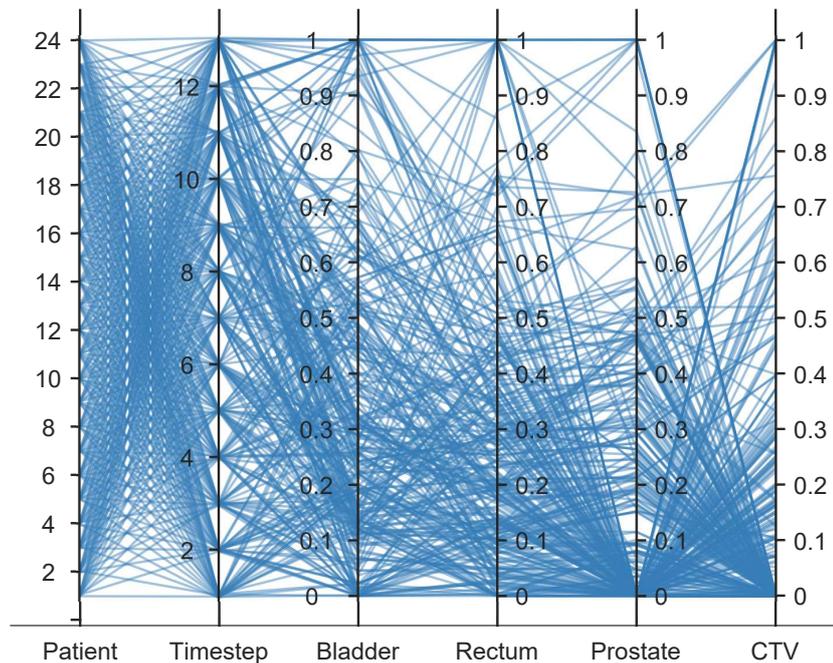


Figure 4.18: The parallel coordinate plot (PCP) has a separate vertical axis for each feature, including the shape difference values for all organs, as well as the patient and timestep ids. For each observation the respective points on the axes are connected using a line. Correlations are encoded by polyline patterns between neighboring axes.

Similar to a PCP, a *glyph plot* can encode higher dimensional data without too many problems. For example, the use of star glyphs allows one to choose the number of dimensions freely, although after a certain limit the distance between the features might become too small and make the glyph harder to interpret. Glyphs are often visually modified using color, blur, and added elements to encode further information [BKC⁺13]. In Figure 4.19 we see simple star glyphs. The preservation of the initial patient and timestep context depends on how the glyphs are created. Producing a glyph for every element in the cohort may create too much information to effectively display. The data is often grouped based on certain problems, e.g. on a per-patient basis and only their average properties are shown as a glyph. This creates a representation that is oriented on providing an intuitive overview of the data, allowing for easy comparisons between multiple cohort elements [KPB14]. A problem with this simplification is that it may hide correlations present in the data, or between multiple glyphs. As the general shape of this representation is the most prominent feature, the arrangement of the axes around the center is very important. In terms of adding more elements and features, we run into similar problems as with the scatter plot matrix and the parallel coordinate plot. Adding more glyphs, reduces the effective size a single glyph can have and makes it hard to interpret individual ones. Increasing the number of lines in a single glyph leads to overlaps and also reduces its understandability [BKC⁺13]. Interaction-wise a user could

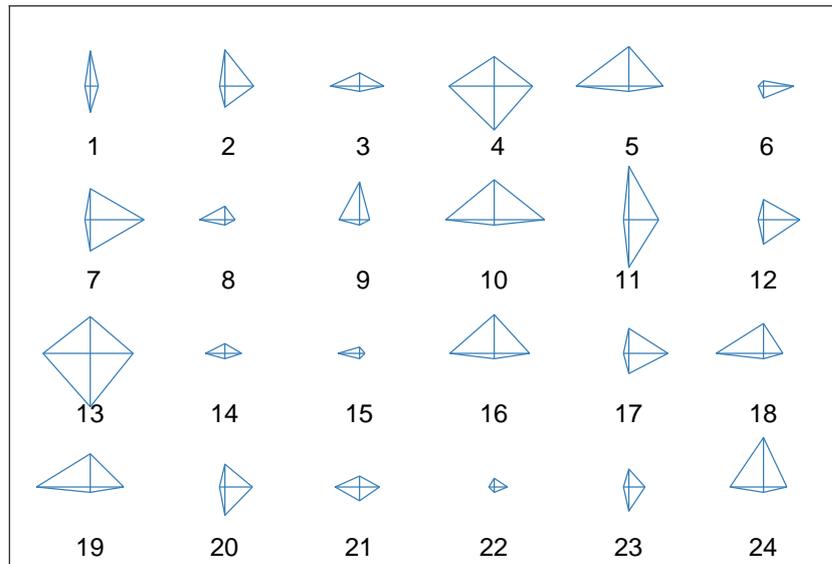


Figure 4.19: In this star glyph plot each patient’s organ variability is encoded in the length of the four lines protruding from the center of each star. The longer the lines are, the more the respective organ varies. Starting from the top and going clock-wise the encoded organs are bladder, rectum, prostate, and CTV.

either select or hover over a glyph for more detailed information, but also rearrange the axes in a different way to change the shape of the glyph.

The fourth and final option we look at are *tabular plots*, shown in Figure 4.20. Generally, these plots struggle, the more dimensions are added, however recent works have suggested that this problem can be circumvented by creating a representation that treats each dimension as a column and therefore allows for a more easy extendability. Individual observations are in this view grouped based on one property, meaning that it can either preserve patient or timestep contextual information, but not both [BBS⁺18]. However, one could only view the values of a single organ or dimension and use the patient and timestep values as the axes. Most commonly a simple color-coding is chosen to show the average value of each cell. There also exists the option to use overlays, blurring, and other options to encode further information [RCG⁺17]. One of the biggest benefits of this view is that tables are commonly known to most users and one can build upon this understanding. Effectively detecting patterns and correlations across and between dimensions requires sorting. By ordering the values of one dimension other dimensions might also reveal an underlying pattern depending on the former. Generally, this approach suffers from adding more observations as each of them would introduce a new line, leading to quadratic growth. To combat this, observations are often aggregated based on certain properties to only create a small number of subgroups [EF09]. This could lead to a loss of details and tools need to be provided to analyze these groups, based, for example, on their inner group variance, as an indicator for how homogeneous a group is. The

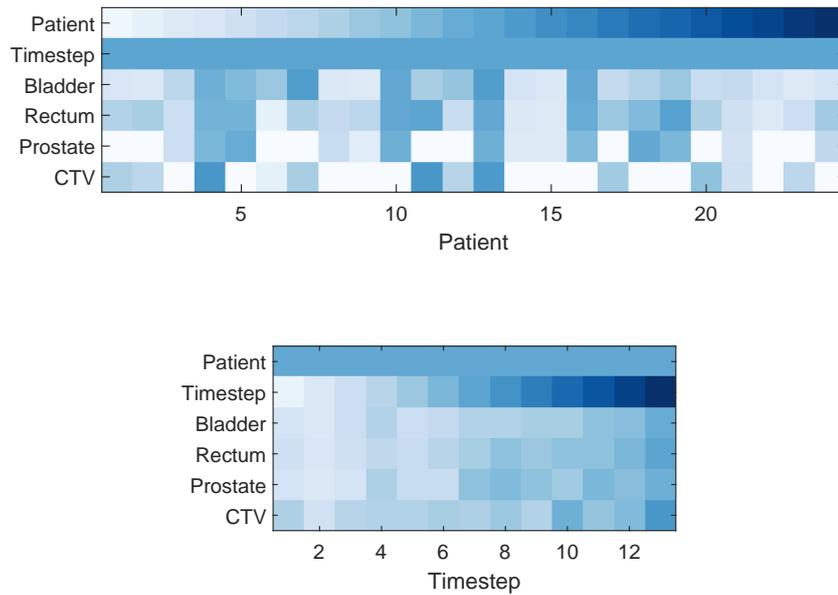


Figure 4.20: In the tabular plots the individual shape difference values are either aggregated based on patient or timestep ids. The average values are encoded as the cell color (low  high). Correlations are seen in increasing or decreasing gradients.

detail-on-demand interaction methods used for the other visualizations can also be used in this one because most of the tabular views correlation recognition depends on the way elements are aggregated and sorted. Interaction tools to change the sorting are also required [FGS⁺17].

Each of the shown methods has its benefits. One of the most important abilities for us was to show the shape change information while at the same time preserving contextual information, like patient and timestep id. Additionally, we wanted to ensure that the visualization itself is readily understandable by users who do not work with visual analytics tools on a regular basis. Therefore we decided to employ a modified tabular view as the main overview tool of our visualization. Instead of providing separate axes for all dimensions and aggregating the observations based on either timestep or patient id, we show both of these values along the axes and highlight the values on a per-organ basis in the grid cells. Depending on if the encoded value represents a distance or a group identifier, we use different colormaps. For the former, we use a sequential white to orange map, indicating small and large distances, and for the latter, we use a qualitative colormap with a pastel color scheme. Both of these maps were taken from Colorbrewer [HB03]. Through this representation the user should be able to compare values across timesteps and patients. To extend this approach to multiple organs, we split each cell into equally sized parts, one for each organ to be shown. The idea behind this being that this should enable users to directly compare values of multiple organs and detect

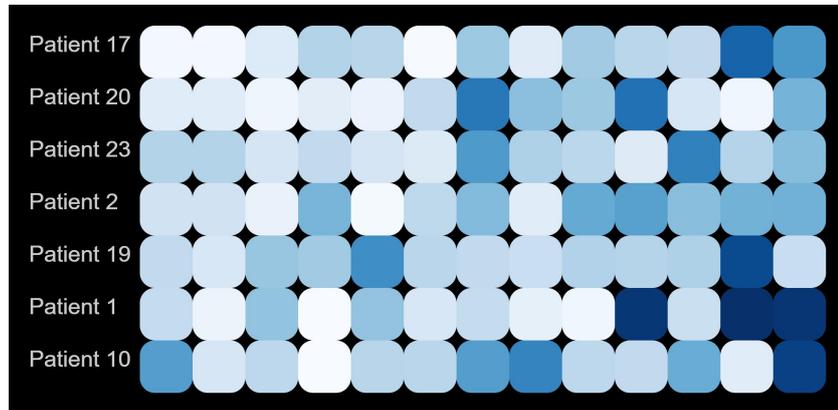
certain patterns and correlations, similar to a glyph-based representation. To make differentiation easier between multiple timesteps and organs, the table cells/timesteps are represented by rounded cells. The user can manually filter, which organs are shown at any given time, enabling to only show the information he is interested in. A screenshot of our solution is shown in Figure 4.21. We decided to use a color-based encoding to show the value of a feature, as we believe that this allows for precise comparisons between non-neighboring elements. At the same time, it preserves the intuitive recognizability of similar elements in the cohort. Compared to a parallel coordinate plot, we think that the resulting representation is more easily understandable by new users. Although we can only encode a limited number of dimensions, we still consider encoding up to four organs at a time to be sufficient. To ensure that the method scales well with more patients, we also use aggregation on a per-patient and timestep level. This possibility is also the reason why we chose this method over scatter plot matrices.

4.2.2 Zoom and Filter

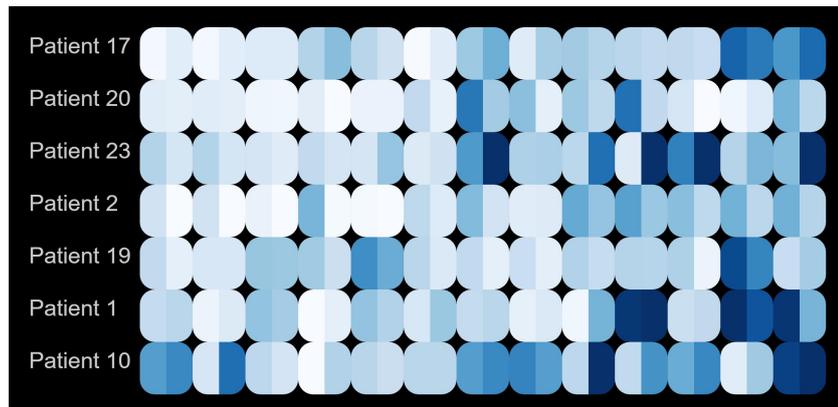
The initial layout of the overview visualization provides users with the option to see the whole cohort at once in such a way that the amount of shape change or the shape type directly encoded for every single combination of timestep and patient. To analyze the data the user would be required to manually scan row-by-row to either detect patients that look similar or to find patients with certain outliers. Doing this even for a relatively small cohort requires looking at large numbers of elements, like in our case where 24 patients each having 13 timesteps lead to a total number of 312 elements per organ. Finding patterns in such a cohort gets increasingly more difficult with every added patient. Therefore, we aim at finding techniques that allow users to drill down from a general level to a more and more precise one until he has gathered sufficient information and can even perform a more detailed shape analysis as a next step. To preserve the usability of our visualization for larger cohorts, we review several interaction methods and visualization techniques commonly used to allow us to make the cohort dataset easier to analyze in both the patient and timestep dimension.

One of the most common methods in this area is the use of filtering, which allows the user to remove uninteresting elements from the data, based on certain selection criteria, e.g. only showing patients without missing values. Instead of removing elements, sorting simply orders all elements based on their average value in one or more dimensions in an ascending or descending manner [BBS⁺18]. A method closely related to sorting is grouping, while the former, orders elements based on their similarity, the latter goes a step further and tries to create distinct groups of elements with similar properties. One way of doing this could include defining bins based on certain intervals inside the data and assigning elements to these groups based on their value [FGS⁺17].

We have several goals that we want to achieve employing these methods. Initially, it is important to know to what extent the method can be used to *reduce* the amount of visual information displayed. Related to this, it is also important to know how the chosen approach affects the capabilities of users to detect *underlying patterns* as well as *outliers*.



(a) One Organ (Bladder)



(b) Two Organs (Bladder and Rectum)

Figure 4.21: Screenshots of the tabular layout used in our tool to represent the similarity of each organ shape to the mean shape of each individual patient. The average shape difference values are encoded by the cell color (low  high).

We mostly work with numerical values, but the chosen approach should also extend to other *types of data*, for example, categorical cluster ids. Since users only have little prior *experience* with visual analytics tools, the required amount of *interaction* should be low.

Filtering is probably the most straightforward way to reduce the number of elements visually displayed. By allowing arbitrarily broad or narrow queries, filtering can allow the users to specifically focus their analysis on a chosen subpart of the overall cohort, enabling them to look at smaller patterns that would otherwise be lost in the complete dataset. Hiding large portions of the data can also lead to a loss of contextual information, e.g. providing information about the general prominence of certain patterns in the complete data [Kei02]. Filtering itself can be used with arbitrary data types, ranging from numerical, over categorical, even to textual data. The interaction needed by the user largely depends on how complex the queries are which the system allows to create. Simple brushing tools only require a few clicks [MW95]. More complex query systems can require textual input using regular expressions. A simple example is shown in Figure 4.22b.

Sorting is one of the most common methods of analyzing tabular data. Although it does not offer any reduction of the presented visual information, its rearrangement of the data elements can allow users to identify several parts of interest, e.g. elements with especially high or low values indicating outliers. This method does not help with increasing the focus on small parts of the data, as much as filtering would, but preserves the contextual information of all other data elements. This can allow the user to detect patterns across elements, like correlations between multiple columns [BBS⁺18]. Figure 4.22c shows the effects of sorting on a simple example. As sorting only defined for numerical and ordinal values, other data types require some kind of mapping function that defines if a certain value is bigger or smaller than another one. In terms of needed user input, sorting can simply be done for single dimensions by selecting a table column. One could also create more complex sorting versions where multiple dimensions are used one after the other, requiring more user input, to define the order [FGS⁺17]. These more complex interaction methods may not be optimal for non-visualization experts.

Lastly, **grouping** is a prerequisite to visual aggregation where groups of elements are combined into a single representative, allowing for large reductions of visual elements [YKSJ08], as shown in Figure 4.22d. Contrastingly to filtering, the aim of creating these group representations is to provide contextual information about the elements removed from the view. The creation of groups based on certain properties can allow users to gain a quick overview of general patterns in the data and perform further inspections based on them. Moreover, this can be used to build semantic groups inside the data and compare them to see where they differ, e.g. patients with and without bladder toxicity [GAHGGPT15]. Depending on the chosen visual encoding, a problem with aggregated representations could be that they can hide outlier values. The significance of the aggregated value largely depends on how dispersed its internal values are [BBS⁺18]. Groups can be created for arbitrary data types, either by simple queries, by creating bins for numerical values or even based on clustering algorithms. Similarly to histogram creation, the number of groups can largely affect the final result. Grouping can be done in

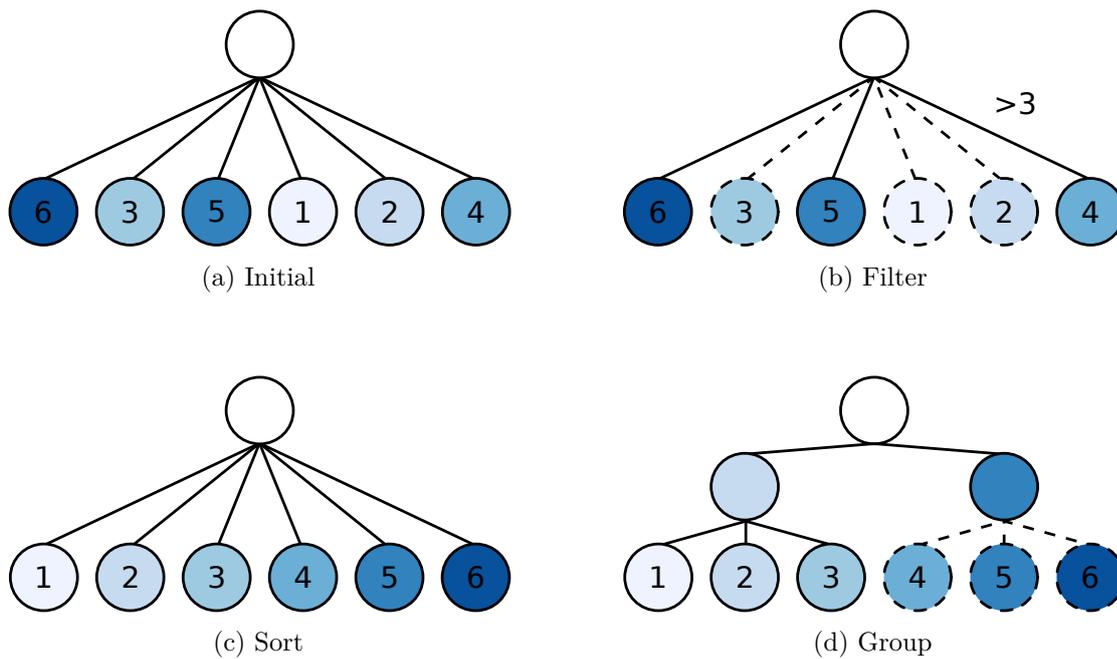


Figure 4.22: Examples of the three different techniques presented to explore the tabular data containing numerical elements.

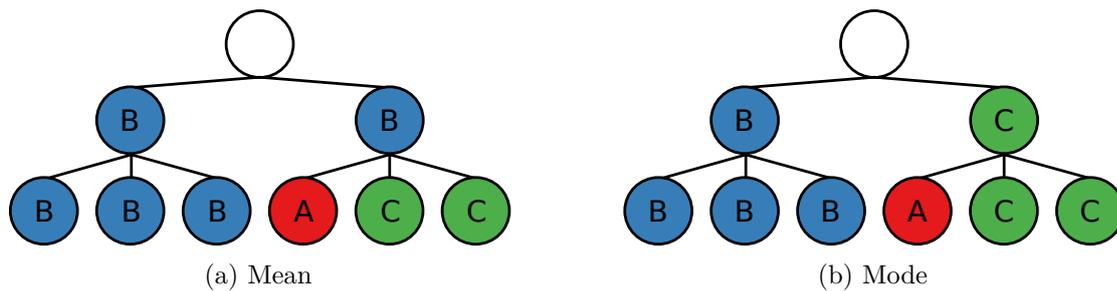


Figure 4.23: Example illustrating why the use of the mean on categorical or ordinal data leads to false results. If, for example, the categorical elements are saved as numbers $A = 1$, $B = 2$ and $C = 3$, which they commonly are, using the mean can lead to false results where the aggregated value depicts something that is not present in the data. In this case the use of the mode, which returns the most common value, is much more indicative of the contained elements.

several different ways, similar to sorting, it can be done on multiple levels. This process requires interaction to select the organs on whose basis the data is grouped.

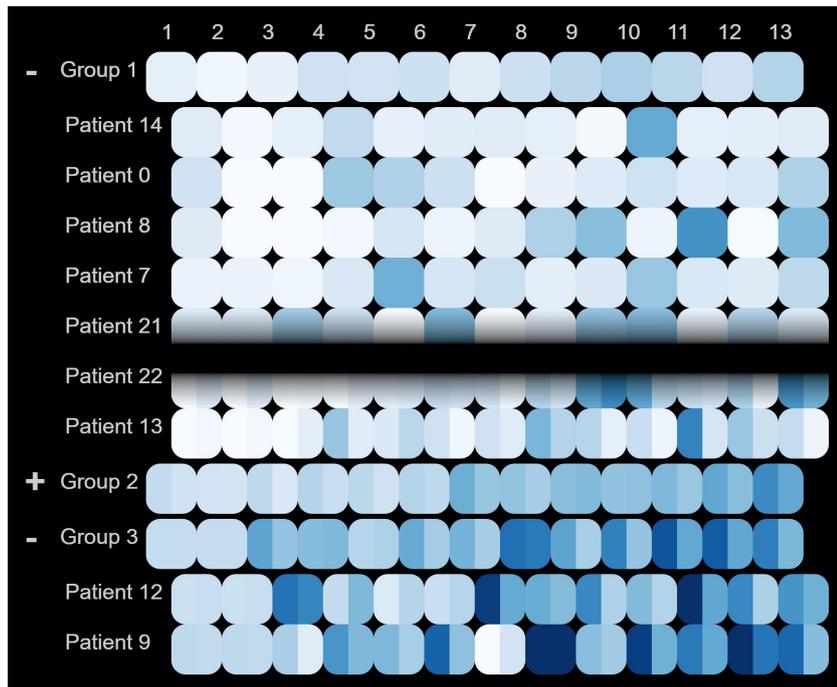
Two of these three methods found application in our tool: sorting and grouping. Sorting is used to order patients based on their use for finding patterns in large tabular representations as shown in the work of Furmanova et al. [FGS⁺17], for example. Users can select

the organ on which the table is sorted and the average shape change is then estimated and used to determine the final order. Missing values are arranged so that they are in front of the smallest ones, at the start of the table. For the aggregation of the categorical cluster ids we use the mode, which returns us the most prominent shape type in a group. An example showcasing why the mode should be used instead of the mean can be seen in Figure 4.23. All of this allows the user to see precisely, which patients in the whole cohort undergo large and which only small shape changes. It also enables the user to find organs with similar shape change behaviors. Grouping is employed in two different ways, for both the timesteps and the patients. As mentioned in Section 2.6, the first five timesteps of each patient are taken at consecutive treatment days during the first week, the rest is taken on a weekly basis. One assumption is that the differences measured in the first five days are an indicator of the overall patient variability. Using this knowledge, we aggregate the 13 table cells for each patient by grouping timesteps 1 – 5 and 6 – 13. The patients themselves are grouped based on their average shape variation values for a single organ or their assigned shape type. If the average variation is used, four different groups are automatically generated, i.e. one for missing, low $< 25\%$, medium $25\% - 75\%$ and high $> 75\%$ average values, based on the interquartile range. The assignment is based on the value range of the respective dimension, with our goal being to enable the user to view only a number of groups, which represent the most common patterns in the data, without the need to manually specify the bin sizes. Each group can also be extended and collapsed to only show the subset of the patients that the user is interested in analyzing. We show the effects of these two methods in Figure 4.24.

Provided that the user was able to analyze the whole cohort using the previous steps, a more detailed exploration might be necessary to both gain additional insights or put previously found patterns into context. Up to now we only analyzed the average similarity values and cluster ids of groups. A next step would be to perform a more in-depth analysis of the groups themselves, to understand their overall aggregation quality, e.g. their value distribution and amount of missing data, as both of these indicate how much the shown average value can be trusted. Going one step further, the user might also be interested in finding out what part of the organ is the one that leads to large shape variations, or how different shape group types compare to each other. All of these detailed inspection methods work in different ways, which we will discuss in the following.

4.2.3 Distribution on Demand

Further analytical processes of the groups could include analyzing how representative a certain shape type is for the current group, or if the shape deviation behavior inside a group is homogeneous or largely varying. Both of which indicate the aggregation quality. Our ability to judge the intragroup distribution is largely dependent on the chosen encoding of the aggregate value. Currently, we employ two aggregation strategies, i.e. the grouping of similar patient rows and combining timesteps into two groups, both of which need to be presented in a way that the user can understand them.



(a) **Patient Grouping** - For every group a row similar to the patient rows is generated, i.e. it contains as many timesteps as the individual patients. The values of the group rows are estimated using the mean of the contained patients. This is shown for both one and two organs.

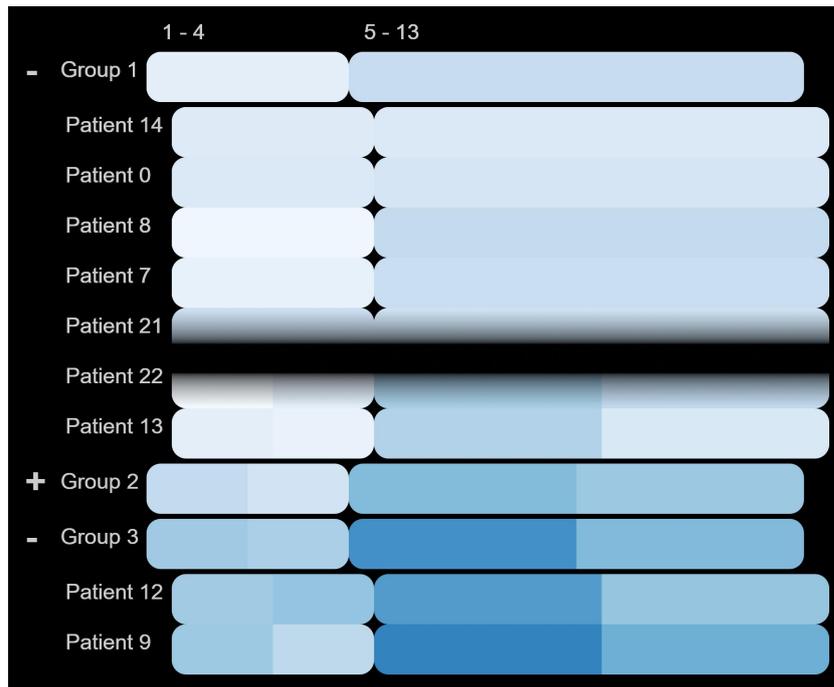
Figure 4.24: Screenshots of the sorted and grouped cohort data.

Distribution Representation

We looked in a first step into different ways that were used in previous works to provide users with a notion of the underlying data distribution. To ensure that our choice of encoding supports the users as good as possible with the tasks at hand we analyze three different ways to estimate the variability of groups: distribution plots, dispersion measures, and statistical tests.

There are many different visualization approaches to render distributed elements. The general idea is to find a representation which is able to render the aggregated elements in such a way that a user can see how spread out they are across an interval. Contrastingly, dispersion measures do not draw the initial data points but instead derive statistical characteristics from them that try to describe the values inside a group, like the standard deviation, or certain quantiles. Statistical tests go even a step further and do not only estimate characteristics, but also evaluate the significance of their findings. ANOVA, for example, evaluates the quality of a grouping based on the differences in overall variance and group variance [RFGD08].

Nearly all of the mentioned methods have some *mathematical requirements* that need to



(b) **Timestep Grouping** - The first five timesteps and the remaining ones are also aggregatable into two groups. The resulting values are also the mean of the contained timesteps for each patient. The size of the extent of the grouped cells is kept the same as the original ones, to preserve the visual context. This is shown for both one and two organs.

Figure 4.24: Screenshots of the sorted and grouped cohort data (*continued*).

be fulfilled, therefore understanding these allows us to determine if they are appropriate for the use in our tool. This also extends to the *data types* these methods are applicable to, as we ideally want to apply the methods to numerical and categorical values. Furthermore, understanding of the output is needed to determine ways to *visualize* them but also to know if they can provide us with knowledge about the overall *distribution shape* and even compare it to other elements across the cohort.

Distribution plots, such as box plots shown in Figure 4.25, do not have large restrictions made to the inherent distribution of the data points, as their main goal is to provide an independent representation [WPK89]. Their output often depends not only on the input data but also on parameters that are chosen beforehand, e.g. the kernel size for violin plots, meaning that the same data points can result in differently looking distributions. Concerning the input data types themselves, there exist approaches that are able to handle everything from numerical data up to categorical data. A benefit of distribution plots is that they preserve the underlying data structure and can allow users to judge the number of elements contained [HN98]. While these plots are well suited for comparing a few groups, they do not scale well for more elements with smaller screen space, as they

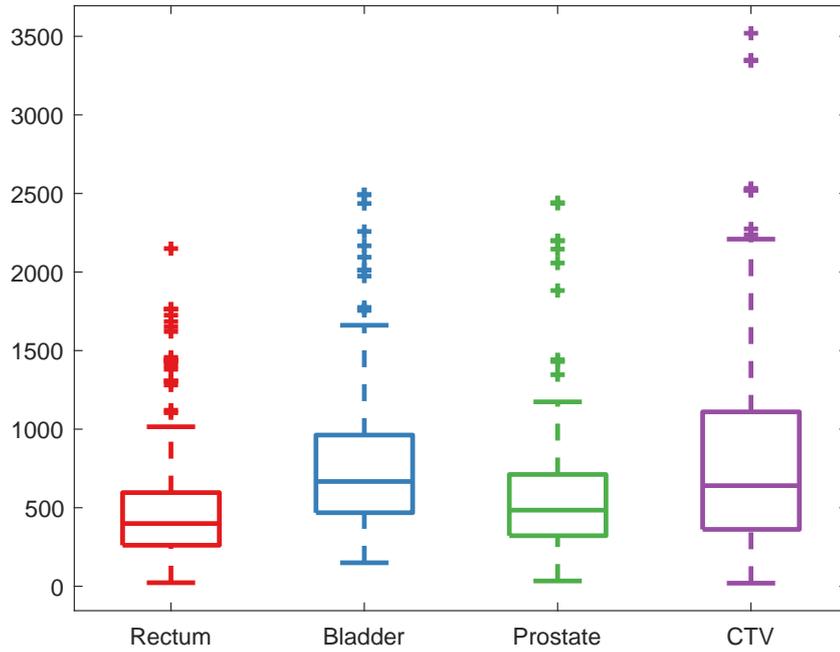


Figure 4.25: Distribution plots highlighting the deviation of each cohort element to the mean shape of the patient.

still require complex plotted data representations.

	Rectum	Bladder	Prostate	CTV
Mean μ	480.63	764.32	567.41	822.07
SD σ	335.79	421.08	424.40	662.90
Min	22.79	149.72	33.96	19.85
Max	2149.20	2492.10	2441.70	3519.60

Table 4.1: Distribution characteristics describing the aggregated shape differences for each organ measured in their respective shape spaces.

The estimation of *dispersion measures* like the standard deviation or the interquartile distance [Man11], reduces arbitrarily large collections of data points to a few numbers. The descriptiveness of these figures, largely depends on the properties of the underlying distribution. For example, the ideal use case of the standard deviation σ would be the description of a normal distribution, because its value would describe the percent of elements contained: $\mu \pm 2\sigma = 95\%$ [RFGD08]. These statistical dispersion measures are also only defined for numerical values. Some works suggest that the extension to categorical values could be made by measuring the number of values differing from the mode [SZ12]. As mentioned before, the output of the calculations only contains a few values, that could, for example, be visualized using color coding or glyphs. They do not

offer detailed knowledge about the shape of the underlying distribution. Instead they can provide a rough estimation of it. This enables the user to more easily compare larger numbers of aggregated elements to each other, as only these measures are inspected. An example of a couple of those figures is shown in Table 4.1.

	Rectum	Bladder	Prostate	CTV
KS-Test (Normal)	✓	✓	✓	✗
p-Value	0.12	0.20	0.19	0.04
KS-Test (Uniform)	✗	✗	✗	✗
p-Value	0.00	0.00	0.00	0.00

Table 4.2: Results of statistical tests performed on the shape difference values. Hereby we tested if the data for each organ is normally or uniformly distributed. We found slight indications that the data follows a normal distribution.

Lastly, when looking at the use of *statistical tests* we see that they also come with several different underlying requirements that need to be checked before they can be used, to ensure that the test result is valid. This could include checking if the data is normally distributed using the Kolmogorov-Smirnov test [Lil67] or if the variance is homogeneous in groups of independent samples using the Levene test [BF74]. Furthermore, the user needs to define beforehand a threshold at which a pattern in the distribution is considered significant. After its application, the result is a single value that determines if a statistically significant effect exists or not. This value does not encode the actual magnitude of the distribution or the number of elements in any way. It can, nevertheless, be used to detect significant patterns across the whole cohort. Table 4.2 encodes the exemplary results of two distribution tests on our shape difference data.

We perform two separate aggregation steps in our tool, once across time to group the first five days of treatment and the rest, and once to group patients with similar shape variations or types. As the former only aggregates between 5 and 8 individual sample points for a single patient, we decided that using a simple dispersion measure like the standard deviation to show the average deviation from the mean is enough to provide information about the distribution, as the number of elements is constant. The opposite can be said about the aggregation of patients in groups. While the average values are already encoded in the grid cells, users are more interested in understanding the number of elements contained in every single group, which is why we chose to employ a distribution plot. The goal is to convey if a certain group is common or only represents an extreme case.

Patient Distribution Visualization

Visualizing the distribution of values for a group of patients, we want to offer insights into the overall spread of values, allowing the user to understand which groups contain most of the data and which can be considered as outliers. Being able to reflect upon

the previous aggregation results, is, therefore, a task that we need to support, as we mentioned before, distribution plots can allow us to achieve just that. As with most visualization problems, there are multiple solutions one could use to tackle this problem. For this part of the work we look into box plots, histograms, violin plots, and scatter plots.

Box plots are one of the more common distribution visualization approaches and are created by measuring the median, as well as the quantiles to split the data into four equally sized portions. Histograms are bar charts that are created if the data is binned into groups and the number of elements in each group is counted and drawn as one bar [RFGD08]. Violin plots are created by using kernel density estimation, where at each point of the data a filter kernel is placed and summed up to create a final distribution of the overall density [HN98]. Scatter plot based approaches simply plot a dot for each data point. Although they are normally only used for 2D data, they can also be applied for 1D data [HTA⁺18].

We evaluate these approaches based on how well they can convey the *overall data distribution*, for example, how easy it is for a user to identify the region with most elements. This also extends to the ability of the approach to allow for the *identification of outliers* inside groups. We also look at normally contrasting features: *precision* and *intuitiveness*. The former feature describes the ability of the chosen method to allow the user to precisely estimate important key properties, like the center of all data points. The latter feature focuses more on how understandable the plot is for users who have not worked with this kind of representation before. Finally, we also want to look at the *computational requirements*, as well as the scalability of the approach for large amounts of data.

The interquartile regions shown in a *boxplot*, in combination with the median, encodes the range where 50% of the data are present, it's size shows how spread out the data is, e.g. a small box indicates a large group of points. The whiskers extending from the box indicate the distribution of most of the remaining data [WPK89]. This plot is shown in Figure 4.25. Most box plots also explicitly encode outliers, as points outside the whisker range. As this plot is based on statistical characteristics, it allows for precise analysis of these properties. Certain works suggest that this kind of plot often simplifies the underlying data too much, as many differently distributed sets of points can result in the same boxplot [MF17]. New users also need to have a certain knowledge of statistics, as well as an explanation of the general encoding to be able to correctly interpret this kind of representation. The computation of the plot requires an additional sorting step of the data, before the characteristics necessary for the plot can be calculated, which can take some time for large arrays [RFGD08]. The final plot is mostly independent of the number of elements, except for the drawn outliers.

Histograms are a bit more flexible in terms of showing the distribution of the underlying data. While box plots only highlight the region with most elements, histograms can show more complex distributions that might even have multiple regions with large numbers of elements, as seen in Figure 4.26. The number of bins largely impacts the precision of the

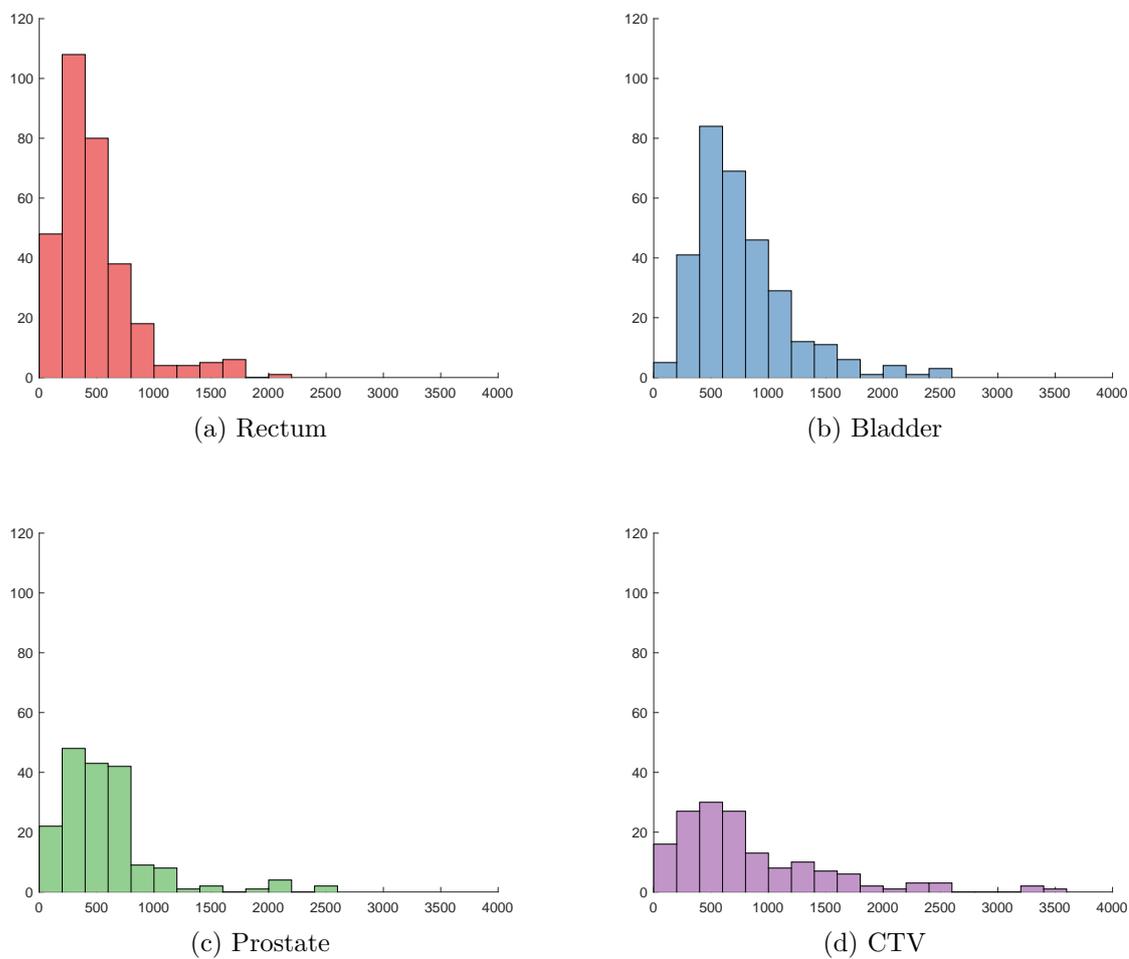


Figure 4.26: Histograms showing the distribution of the shape difference values inside the cohort on a per-organ basis. The x-axis shows the value range and the y-axis the number of element that fall into each of the bins. The higher the bar the more elements have a certain value.

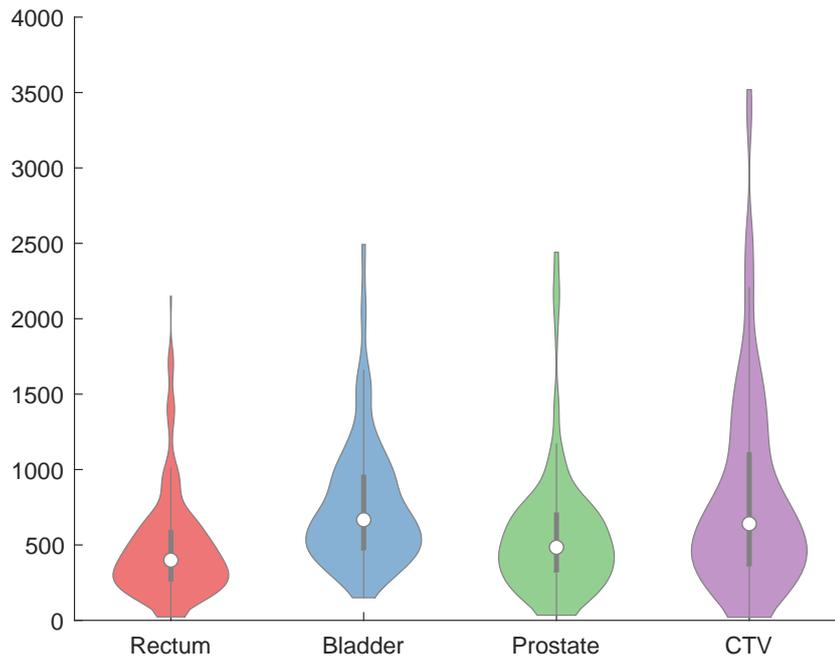


Figure 4.27: Violin plots showing the distribution of the shape difference values inside the cohort on a per-organ basis. The gray box in the center is a simplified box plot, including elements like the median, quantiles, and whiskers. The colored lobes around the box plot indicate the individual value distributions.

resulting plot [LM17]. Outliers can also be identified if certain parts of the histogram, contrast with the overall distribution, which might be a hint to create another group from these data points. Histograms precisely encode the number of elements contained in each region, allowing users to properly judge their occurrences. They do not offer any other statistical information, like the mean value, which can only be visually estimated. As bar charts are a common representation form in many tools, most users are familiar with and can directly interpret them. Choosing the ideal bin size offers some challenges. The calculation process itself is simply done by iterating over the data and adding the elements to the appropriate bins. The number of bars is only influenced by the number of bins defined and not by the aggregated elements. Furthermore, the plot can be visually reduced in size while still being understandable.

Violin plots contain all the information provided by a box plot, but also add a more detailed representation of the underlying distribution of data points in the form of a density plot. This allows users to not only judge where the largest amount of data is located but also to determine if there are several similarly big regions in the group [HN98]. Figure 4.27 shows examples of such plots. Outliers are not explicitly encoded in this kind of plot, they are connected to the main body of the created density plot. This may look for some users as if there were more values in-between than there actually are [HTA⁺18]. Regarding the precision with which statistical attributes can be explored, the

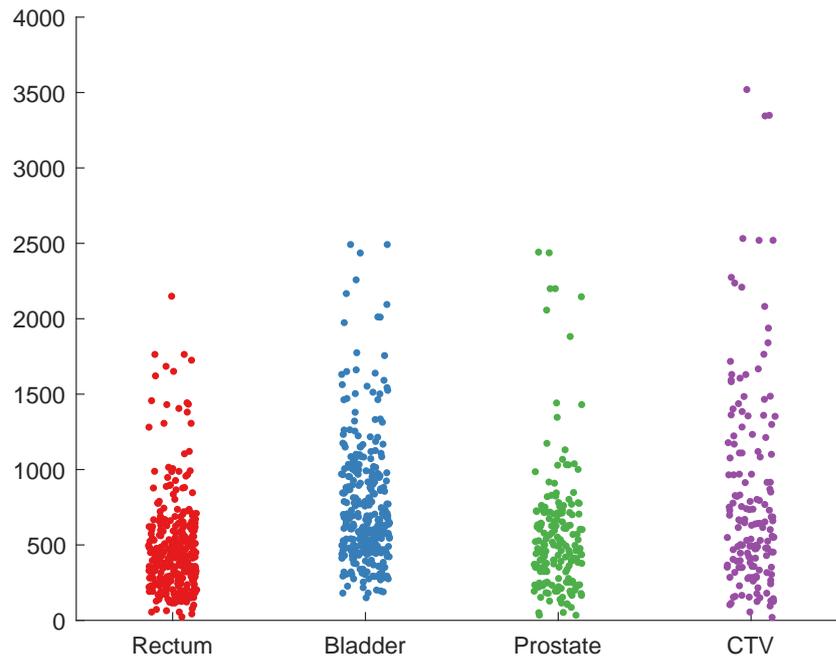


Figure 4.28: Scatter plot showing the distribution of the shape difference values inside the cohort on a per-organ basis. The actual values along the y-axis are plotted together with small random noise across the x-axis for each organ. The latter is necessary to prevent the points from overlapping too much and has no physical meaning.

violin plot contains a box plot and is, therefore, able to also precisely show the median and interquartile ranges. This also means that, similar to the box plot, a user needs an initial explanation, as to what all the parts of the plot mean, especially since violin plots are not that common [APW⁺18]. The application of a filter kernel at each data point is computationally quite expensive, which can lead to problems for large datasets. Many recent works look therefore into options for optimization and parallelization. The visual result of this process is independent of the underlying counting of elements.

Scatter plots and their related form i.e. beeswarm plots, encode the distribution of elements by plotting them directly, as seen in Figure 4.28. To avoid overlap for one-dimensional distributions they use strategies like random offsets in the orthogonal direction, or placing elements on the same height in a line [HTA⁺18]. This enables users to precisely see how many elements are at any given position in the range, allowing them not only to see regions where points are condensed but also to spot outliers. Precisely estimating the average value and other factors is not possible in this representation and can only be roughly derived from visual interpretation. Different from the other representation forms, this one does not assume prior knowledge, providing broader accessibility for new users [APW⁺18]. As there are no large underlying calculation processes involved, apart from standardizing the point range to fit into the drawn region, the data can be directly drawn. This results in a plot where the amount of drawn

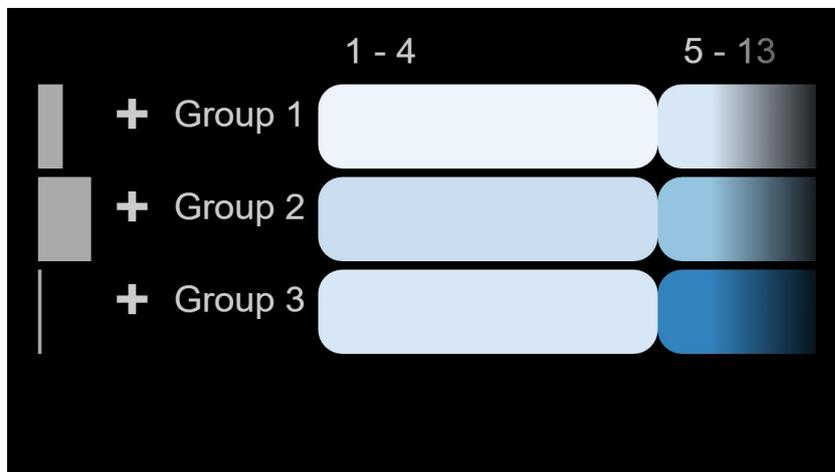
information depends on the number of elements. While this may work well for relatively small cohorts, more elements may lead to more overdraw or a lack of space for all the points to be drawn, generally reducing the understandability of the plot.

Based on this information, we decided to employ a histogram plot to encode the distribution of patients across the created classes. The reason for this was that this representation is able to precisely encode the number of patients in each group, giving direct knowledge about their individual prominence, while at the same time being a visualization that is known to most people. Its downside of relying on the appropriate bin size definition is not problematic in our case as the bin sizes are already predefined for each individual group and do not need to be modified as the group thresholds are currently predefined. If future extensions allow for a more flexible grouping, this representation could also be adapted to present the underlying distribution in more detail. We place the resulting histogram bars next to the corresponding labels for each row, resulting in a vertical bar chart. The space available for the bars is predefined and the individual bar heights are normalized. This ensures that the complete space would only be filled with a bar containing all elements. We also think that this representation scales better for smaller screen sizes as it only relies on the amount of space covered by the individual bars and not on intricate details. As the bars are also separate visual elements they can easily move with their corresponding labels if groups are collapsed and extended. All in all, we think that the simple histogram representation provides us with the most important information, in an easily understandable way, with manageable trade-offs. In Figure 4.29 we show a screenshot of this visualization in our tool.

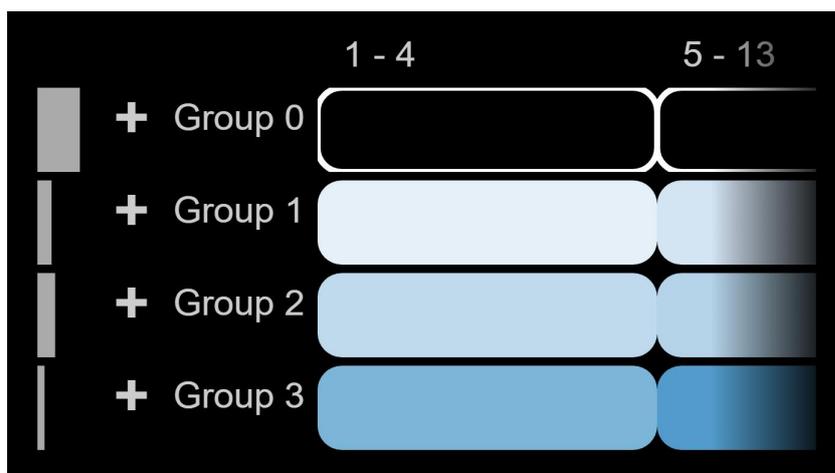
Shape Similarity Distribution Visualization

After solving the problem of visually encoding the value distribution of the aggregated groups in general, we also need to look at the aggregated shape variance cell in our tabular representation. As we previously deemed the calculation of the standard deviation as a dispersion measure to be the best indicator for the inner group variations, we now need to find a way to integrate this measure on a per-cell level. Up to now, we visually encode the shape difference between the mean shape of a patient and its current timestep for each cell using color-coding. This current approach is commonly referred to as a glyph, where some measure, in our case the distance value, is represented by a visually differentiable proxy. Further extensions have to keep this in mind and need to be integrated in a way that they provide additional information, without tampering with the previously presented one. Taking inspiration from Borgo et al. [BKC⁺13], we found four possible methods to encode the standard deviation and add it to our tabular view: size, texture, color, and blur [Mac86].

Size encoding was used to reduce the scale of the table cells based on their standard deviation, to have cells with low variance, i.e. high trustworthiness, appear at their full cell size and those with high standard deviation, i.e. low trustworthiness, respectively smaller [Mac86]. Another option we looked into was overlaying the cell with a textured pattern that conceals more and more of the cell the larger its dispersion is. The use



(a) Rectum



(b) CTV

Figure 4.29: Screenshots of the sorted and grouped cohort data. The images show two different distribution histograms for the groups, created on the basis of the average distribution of the specified organ. The black bar represents patients with missing CTV, the details of this encoding are discussed in Section 4.2.4.

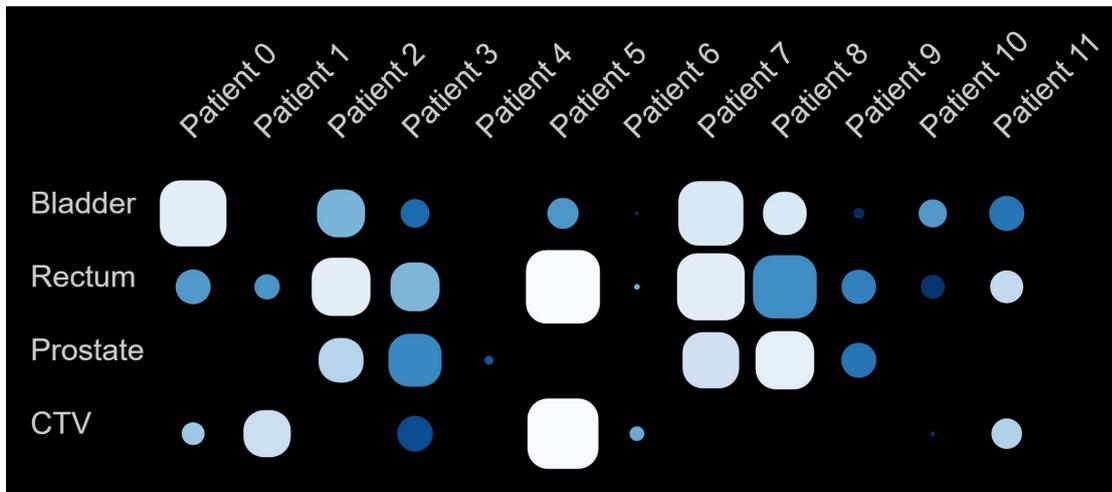


Figure 4.30: Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the **size** of the individual cells, with larger cells representing low standard deviation, i.e. high trustworthiness.

of color is similar to its previous application for encoding the shape difference, in that it simply employs a one-directional colormap to encode the deviation of the contained elements [RCG⁺17]. Finally, we also looked at the option of blurring each cell based on the standard deviation, so that the more the value grew, the more blurry the final cell would get [Mac92].

Each of these visual representations has different perceptual effects on the user, therefore we evaluate their applicability based on the following questions. First and foremost the approach needs to be able to properly *encode* a *numerical value* like the standard deviation, i.e. it should allow users to properly judge the scale of dispersion for individual cells. An extension from simply being able to know if a cell has a higher or lower inner group variation than another one would be the ability to tell the *precise difference* between to cells. As we generally aim at highlighting groups with low standard deviation, because their average value is more representative of the contained patients. Therefore, the used encoding has to *draw attention* to these elements. Furthermore, can the user selectively focus on elements of a certain value and detect them across the cohort, this property is also called *selectiveness*. Finally, the chosen approach should ideally be combinable with the shape difference encoding one to create a *combined glyph*. Ideally this should show both the average shape difference and standard deviation for an aggregated group, without creating unwanted side-effects.

Starting off with the use of *size* reduction to show higher variations, we can see that the property is one of the more common ones to encode numerical values as one can simply map the span between the maximal and minimal standard deviation value to the cell area. We show this encoding in Figure 4.30. To be able to interpret the difference between two elements, one must consider that scaling the side lengths of the table cell influences

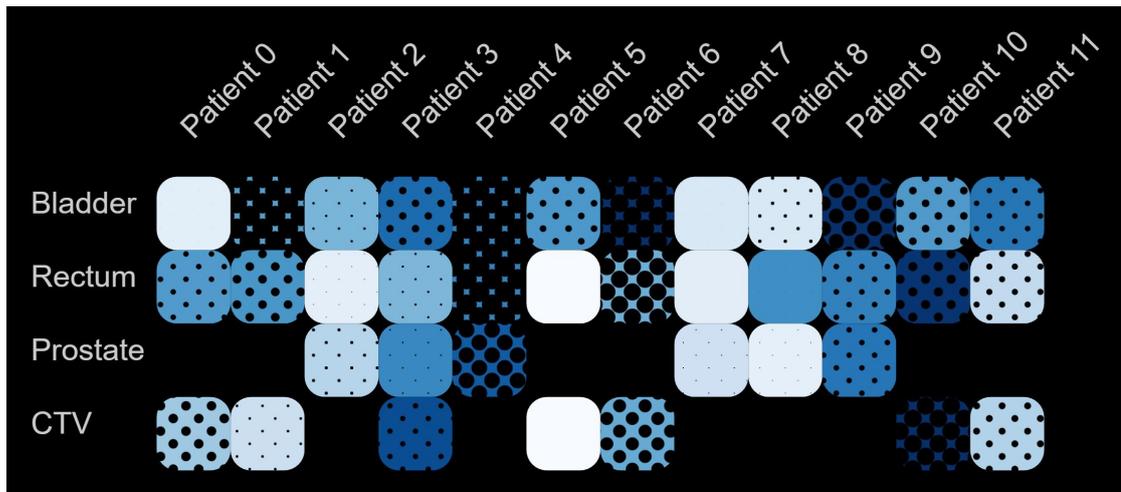


Figure 4.31: Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the **texture** of the individual cells, where cells with smaller dots represent low standard deviation, i.e. high trustworthiness.

the size of the result in a quadratic way. By keeping this property in mind we can allow the user to estimate the difference in dispersion [BKC⁺13]. Differently sized elements also have a perceptual imbalance, leading to larger elements gaining more attention than smaller ones. As works have shown, the user can also explicitly focus on a certain size of the element and search for these elements rather quickly. Combining this encoding with the already in use color channel would generally work. Furthermore, it has been shown that using both size and color works well if they are independent variables and the user only focuses on one of them at a time [RCG⁺17], which they are in our case. These effects for independent variables have been shown for choropleth maps, but they may not work if the color-coded cell itself is changed in size.

Textured patterns can be seen as a combination of a base color overlaid with a pattern of changeable density. In our case, the underlying color can be used for the initial shape difference value and the density of the texture, i.e. the amount of overlap, can be mapped to the value of standard deviation for the group, as seen in Figure 4.31. Generally, there exists no direct mathematical relationship between the mapped value and the resulting pattern density, which means while the user can tell that one element has a higher value than another one the concrete difference is not easily quantifiable [War12]. Due to us being quite adept in scanning for patterns, we are certainly able to detect similar elements in large collections. As these two encodings use different channels they can be applied to the same glyph or in our case table cell simultaneously, however, there might be some negative interference effects [HSIHP06]. For large standard deviation values, the pattern generally grows rather dense and it may alter the visual perception of the color-coded shape difference, e.g. a black dot pattern may lead to a darker perceived color of the cell and a wrongly interpreted shape difference value.

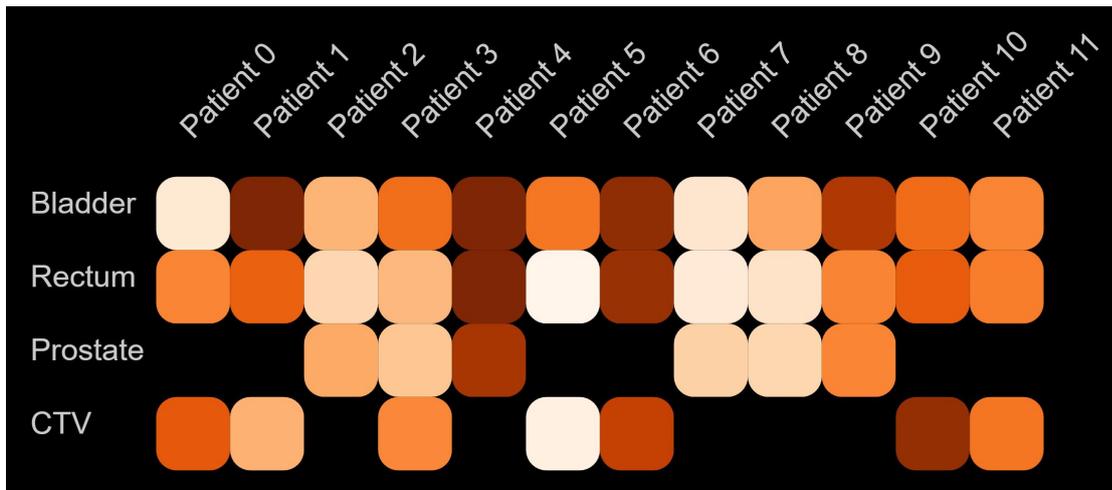


Figure 4.32: Table showing only the standard deviation. Hereby a different colormap is used to separate it from the shape difference encoding (low standard deviation - high trustworthiness  high standard deviation - low trustworthiness).

Color coding can come in many different forms depending on if hue, value, or saturation are varied. For our case, we will focus on sequential colormaps from Colorbrewer [HB03], which mainly vary in value, which is visible in Figure 4.32. To effectively encode the standard deviation, the chosen colormap must be perceptually linear, which is not the same as being linear in the space of the colors. The provided colormaps are already linearized beforehand and can be directly used with the normalized standard deviation. Modifying the color value results in representations which can be ordered based on their appearance, e.g. from light to dark. Still, their concrete value cannot be directly quantified by looking solely at them, which means that they are commonly combined with an explanatory colorbar [RCG⁺17]. Similar to size encoding, where larger elements are perceived more dominantly, in this encoding the bright colors appear more strongly, probably due to the black background. Still, users can selectively focus on elements of a specific value and search for them across the cohort. As we already use the color channel of the table cells for the encoding of shape differences, we cannot apply another coloring to the cells at the same time to show the standard deviation.

We can use fuzziness to *blur* the cells of the table in accordance with the amount of standard deviation in the group, such as in Figure 4.33. This effect is created by applying a Gauss filter to the elements. Increasing the standard deviation of the filter kernel increases the fuzziness of the result. A linear increase of the standard deviation does not lead to a linear increase of the blurring effect, requiring more complex mapping procedures. This encoding allows users to tell whether a group experiences variations. Although, using this approach is not as easy as with the other methods to order elements based on the amount of fuzziness [Kos01]. This also makes finding elements with similar encoded values more difficult. Elements without any blurring generally stand out due to

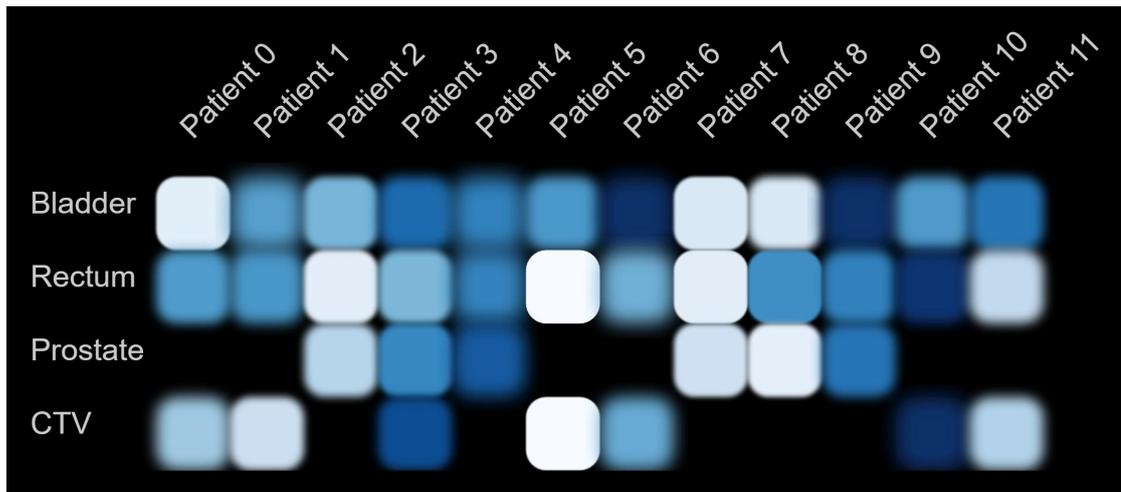


Figure 4.33: Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the **blur** of the individual cells, with sharper cells representing low standard deviation, i.e. high trustworthiness.

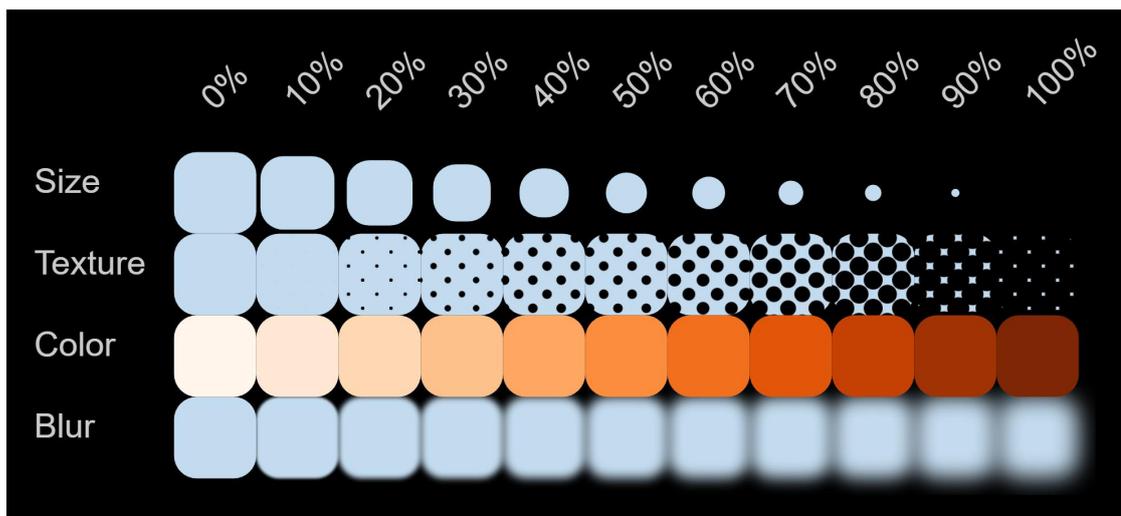


Figure 4.34: Direct comparison of each analyzed encoding for the standard deviation (minimal deviation: 0% and maximal deviation: 100%).

their sharp edges and can, therefore, be more readily identified [Mac92]. Combining this approach with the previously existing color coding works well in theory, but can lead, similar to the texture overlay, to a darkening of the perceived color if the blur value is large, due to a merging of cell color and background color.

Considering the properties of the discussed methods, all of which are compared to each other in Figure 4.34, we chose to use color-coding. Contrastingly to the other presented

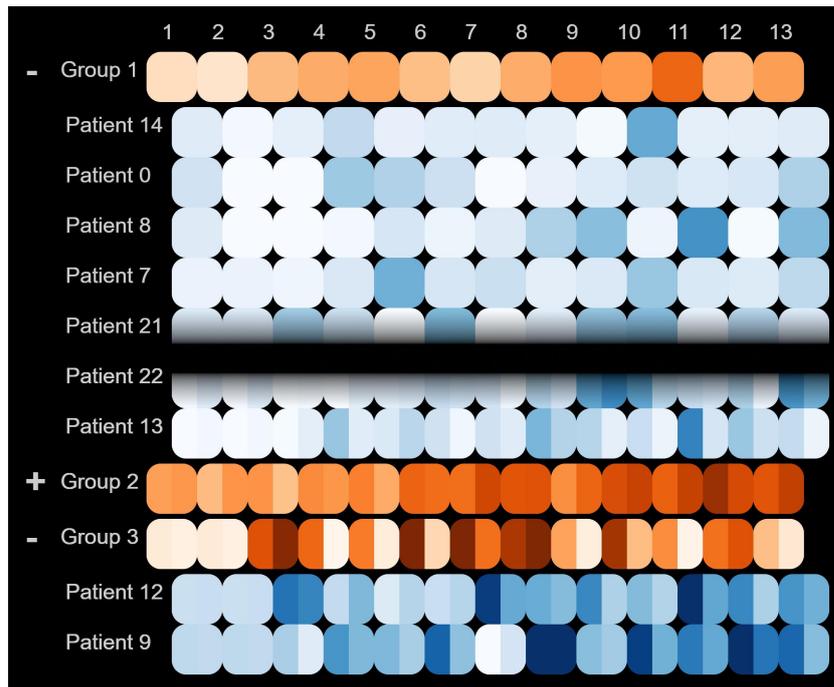
methods this approach allows the user to precisely judge the actual amount of standard deviation for each aggregated value, albeit with the use of a colorbar. Because this encoding uses the same visual channel as the aggregated shape differences, we have several benefits, but also disadvantages. As both types of encoding are similar, the user can directly employ its previously obtained knowledge to this one, e.g. while dark elements in the former representation stood for shape outliers, they stand in the new view for groups with high dispersion. A slight problem with this method is that both color encodings cannot be shown at the same time. Therefore, we give the user the option to manually switch between drawing the average shape difference values or the standard deviation values for aggregated groups. Atomic elements, e.g. single timesteps of a patient always show the shape difference values, as these elements are not created through aggregation. To help users differentiate between elements showing average values and those showing standard deviation values, we use different sequential colormaps. The former goes from white to blue while the newly introduced one goes from white to orange. This pair of colormaps was selected because it is also differentiable for people with colorblindness. We think that the biggest benefit of the separate representation of both values is that they do not interfere with each other to create distorted versions, as would texture patterns or fuzziness. We also think that this method can accurately represent linear increases in values as shown in Figure 4.34. In regards to the use of the table cell area as a dispersion encoding, we decided against it, because the cells themselves already appear rather small, especially if the individual timesteps are shown. Scaling them would make it unnecessarily difficult to interpret the values of the cells. All in all, we think that the chosen encoding, depicted in Figure 4.35, allows users to focus on either to analyze the overall shape differences or the standard deviation, i.e the reliability of the average value.

4.2.4 Missing Data on Demand

Another quality measure, which may affect the resulting group values, is the amount of missing data. The average value is only computed based on the items, which have a difference value for a specific organ. A large group where only a small portion has the difference values for an organ, would overstate the prominence of these values. Missing data occurs in the used shape cohort solely in the form of missing organ segmentations for individual patients, e.g. one patient might have a segmented prostate and no CTV, or the other way around.

Contrastingly to the distribution visualization, where we discussed the different approaches to represent distributions, we can skip such an analysis for the missing data, because the answer is, in this case, more readily available: Individual patients either have the organ data or do not have it and groups of patients have a certain percentage of patients with or without the organ data. Therefore we have to analyze not how to estimate the missing data, but instead, we can directly discuss how we visualize these cases.

For the encoding of missing data information, we considered both the unused visual presentations of Section 4.2.3, such as texture and size, but we also considered other encodings symbolizing emptiness or missing parts. The emptiness metaphor-based



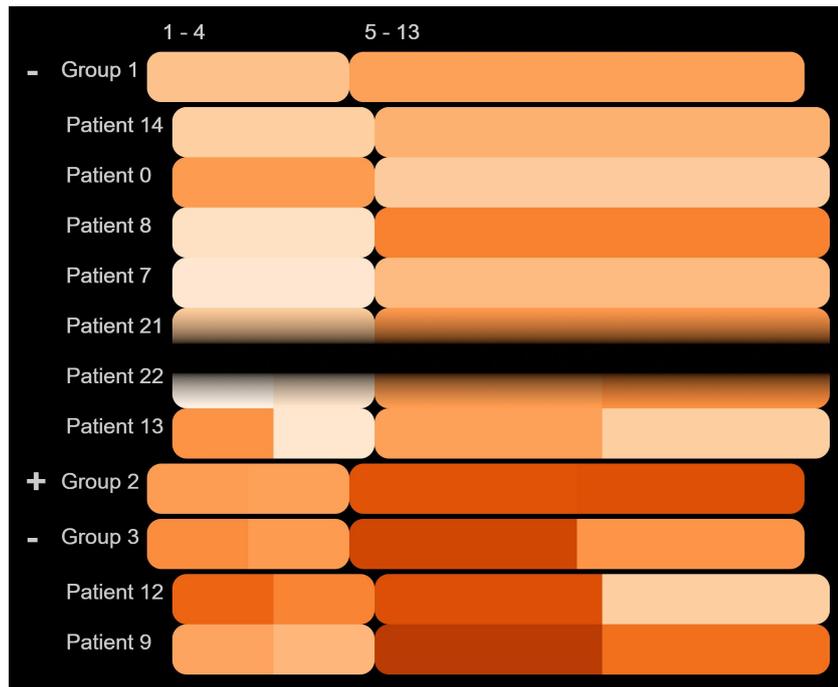
(a) Patient Grouping

Figure 4.35: Screenshots of the encoded disparity (increasing standard deviation values ) for the aggregated values (increasing difference values )

encoding is derived from the work of Blumenschein et al. [BBS⁺18]. In their work, they used two different overlay glyphs, first a random spot noise which grows in intensity the more data is missing, and secondly a triangular glyph increasingly covering the tabular cells.

We analyzed the perception of the encodings. Similar to before we want to know *precisely* how many elements are missing, without actually showing the numbers. Ideally, one could *combine* the chosen encoding with the previous encodings for shape similarity and shape similarity distribution. If possible the encoding should *highlight* elements with high data quality, i.e. with only a few or no missing values. In terms of *scalability*, the approach should work with the split cells for multiple organs and the aggregations we perform on timesteps.

Using *random spot noise* is quite similar to texture overlays. The concrete estimation of the underlying number of missing values is not easily doable, as this approach focuses on conveying the notion that data is missing. One can create a spot noise that covers a certain part of the table cell. The visual perception of this method is not linear and requires complex mappings to achieve linearity. Combining this approach with the previous color encodings might have the downside of altering the visual perception of the



(b) Timestep Grouping

Figure 4.35: Screenshots of the encoded disparity (increasing standard deviation values ) for the aggregated values (increasing difference values ) (*continued*).

table cell color, as it may appear darker due to the added spots. Regarding the extension to multiple organs, the general amount of noise is quick to perceive and one can derive the differences between neighboring organs rather well. Comparing over longer distances may be harder in this representation [RCG⁺17].

The usability of shape overlays, like the *triangular* glyph shown by Blumenschein et al. [BBS⁺18], largely depends on how the shape scales with increased values. In the used example the glyph scales linearly along the diagonal of the cell. Because the glyph has the same color as the background it appears that some part of the cell is missing. If more elements are missing in a group the cell gets covered more with the glyph. The problem with this approach is that the covered area is visually outweighing the covered distance along the diagonal, which makes this encoding more difficult to interpret [BKC⁺13]. This approach generally combines rather well with the previous color encodings as it only reduces the amount of shown color, but does make it appear darker, like the texture patterns or fuzziness. One problem of this encoding is that its look largely depends on the table cell it is supposed to occlude. For example, if the cell is split the glyph gets drastically reduced in size and if the timesteps are aggregated the glyph is stretched due to the cells being wider.

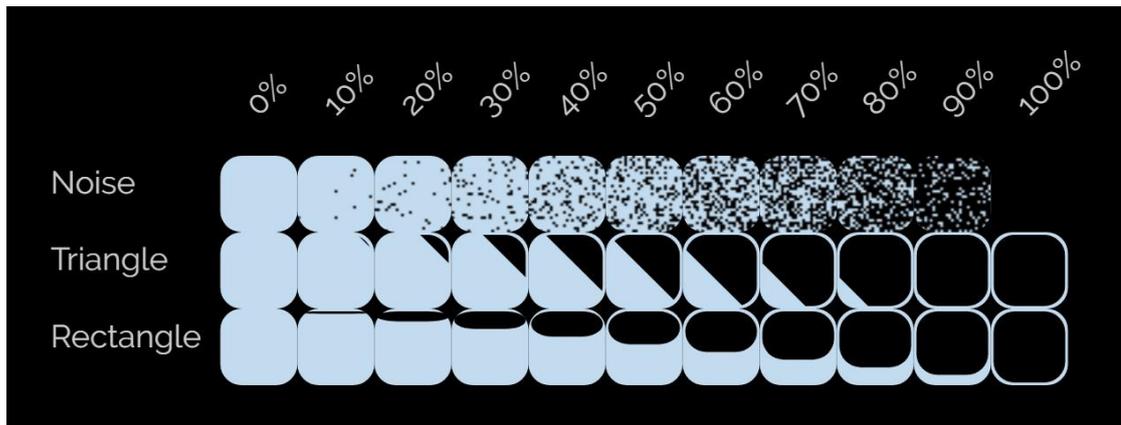
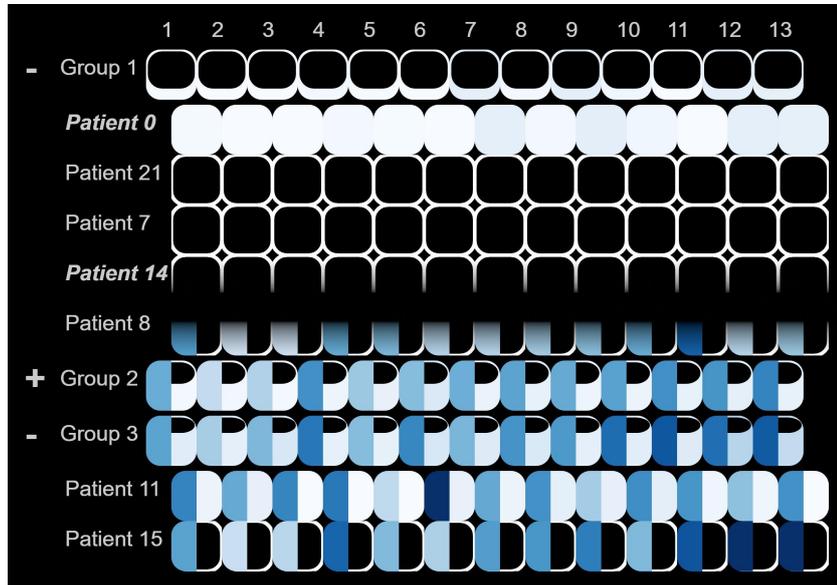


Figure 4.36: Direct comparison of each analyzed encoding for the missing amount of data (nothing is missing: 0% and everything is missing: 100%).

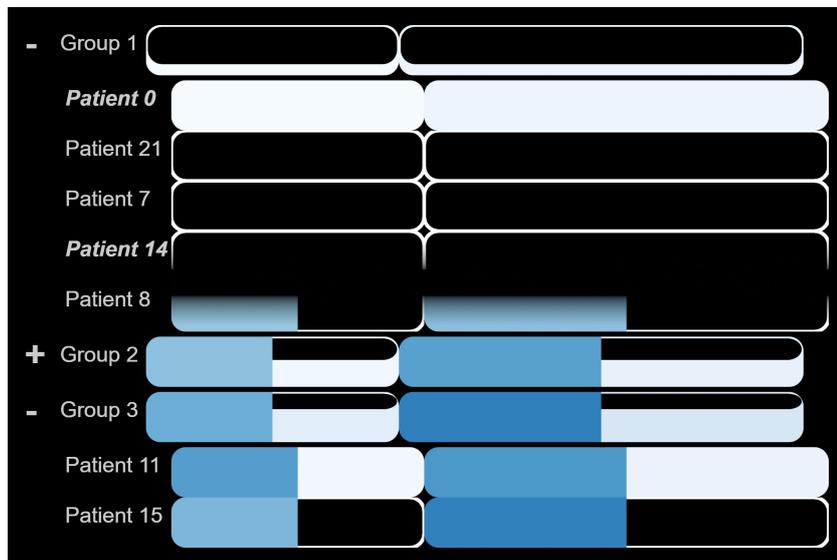
Considering these approaches, which are compared in Figure 4.36, we decided that none of them completely fulfilled our requirements to have both a representation that scales well and is precise in its interpretation. We, therefore, decided to take the glyph overlay and modify it. Instead of covering the cell along the diagonal, we cover it along its vertical axis. This creates a *rectangle* coverage glyph, which is also colored like the background. More simply it can be thought of a top-down bar chart. An example can be seen in Figure 4.36. In contrast to the triangular glyph, the rectangle glyph does not get too heavily distorted when table cells are split or aggregated. It still preserves the metaphorical emptiness, as the cell seems like an emptying glass the more data is missing. The benefit of this approach is that the users can easily determine how much of the data is missing and allows them quantitative comparisons between elements. The glyph also does not hide the color-coded values. Hereby cells with few missing values show more color and are therefore more prominent. Lastly, by only scaling the glyph from top to bottom we do not introduce heavy distortions and it even works well when comparing multiple organs next to each other. If data is completely missing, only a border remains to indicate the original cell, preventing holes in the tabular visualization. Figure 4.37 shows how we realized this part.

4.3 Shape Visualization

At the moment, we are able to analyze the extent of the overall shape changes over time and the predominant types of organ shapes in the data. However, up to now, we only looked at abstract values, like shape differences, and not at the actual shapes of the organs involved. Although this allowed for an effective overall analysis, users are additionally interested in understanding the concrete shapes of the involved organs. This task builds directly upon the previous cohort visualization and can be interpreted as providing the user with **Shape Information on Demand**. The main problem we tackle in this part



(a) Patient Grouping



(b) Timestep Grouping

Figure 4.37: Screenshots of the encoded missing data for the aggregated values.

involves finding appropriate representations to allow users to analyze both the overall shape of the organs and their variations for groups of elements. Additionally, we also want to provide the possibility to find the differences between several groups.

4.3.1 Statistical Analysis

Currently, the involved shapes are only present in the form of their dimensionality-reduced points placed in a shape space constructed using Principal Component Analysis. This low dimensional form allowed for easier high-level comparisons, like the ones employed during the cohort analysis in the previous section. The tabular view only allowed for an overview of generally similar shapes, but does not allow for the analysis of the actual shape differences. Through the use of Principal Component Analysis, we are not only able to create a reduced representation of 3D volumes, but also reverse this process and create volumes for arbitrary points in the shape space. Furthermore, recent works have shown that data aggregation using statistical measures in the reduced space also results in an appropriately aggregated shape in the reconstructed original space. For example, the mean of the reduced points also results in an averaged shape when reconstructed. As we aim at understanding not only the general shapes and variations found in larger groups of patients, but also the positional variance of the involved organs, we need to decide what measures are best suited for this analysis. We, therefore, look at the following measures of descriptive statistics: mean, median, mode, standard deviation, and interquartile range.

The first three measures deal with finding the center of a distribution, although they define it in different ways. For the estimation of the mean, we use its arithmetic definition, which is calculated by adding up all points and dividing them by their count. If the data was sorted, the median would represent the value, which would split the data into two equally sized parts. Finally, the mode, on the other hand, looks for the value that appears most of the time in the data. Instead of finding the center of the data the standard deviation and interquartile range try to estimate the spread of the distribution. The standard deviation is defined as the square root of the variance, which in turn is the mean squared distance of all data points from their mean. The standard deviation depends on the mean and the interquartile range is connected to the median. The median splits the data at the 50% mark, the interquartile range takes the 25% and 75% marks and estimates their distance.

Our decision, which of these measures should be used to visualize the shape changes of patients and groups of patients, is based upon a few key factors. As we have seen previously in the distribution plots of Section 4.2.3, the elements tend to align in a slightly *skewed normal distribution*. The chosen measure needs to handle this circumstance. Furthermore, an appropriate measure also needs to be resistant against a small number of *outliers*. They should not undergo large changes if only a small number of values is changed. In regards to the visualization we are mainly interested in presenting the users with both the *general shape* or position, as well as a region indicating the *upper* and *lower bounds* of the confidence interval. This interval is supposed to highlight the region in which 95% of all shapes or center points are contained. We highlight the properties of the presented

measures in two figures, Figure 4.38 shows the center measures and Figure 4.39 shows the disparity measures.

The ideal use case for the arithmetic *mean* is its application on normally distributed data, where it can precisely define the center of the data. Through the introduction of a skew in the distribution, the actual peak of the data elements and the mean start to diverge. Besides, the mean also tends to be strongly influenced by outliers. It can be shown that by changing only a single value in the data the mean can be altered arbitrarily. This also impacts its viability for calculating a good group representative, as single outliers can heavily alter the mean and lead to values not present in the data. Because the *standard deviation* largely depends on the value of the mean, it shares most of the mentioned properties, like being most accurate when describing data, which follows a normal distribution. The fulfillment of this requirement has the benefit of allowing one to precisely select regions that contain a certain percentage of all elements in the group. Visualizing this region would result in a confidence interval of the possible variations inside a group.

The *median*, in contrast, is able to better handle skewed distributions and outliers than the mean. The reason for this being that, while for the mean the change of one value is enough to affect the result, this would not affect the median. In the case of the median half of all values would need to be changed to affect the median. Implying that the statistical figure is neither drastically impacted by a long tail in a distribution, nor by only a few extreme outliers. The resulting median value of a group is guaranteed to be taken from the data. These mathematical properties also extend to the use of the *interquartile range* estimating the variations inside a group. The elements selected as the upper and lower quartile are also elements inside the cohort. The region in between them contains 50% of the overall data elements.

Usage of the *mode* leads to the most common value in the data, or in the case of a distribution the highest peak. The mode is not always unique, when multiple elements have an equal frequency, there would be multiple modes. Generally, the use of the mode is only recommended for nominal and ordinal data. Its application to sampled data can result in unexpected results, as it is often the case that all data points have slightly different values and not one of them appears more often than the others. In these cases, the mode would not be unique. One way to tackle this problem could be to discretize the data by rounding, for example. For a group of values, the resulting aggregate would either be a single element or a group of elements taken from the data.

Considering the presented facts, we decided against the use of the mode as it is generally too unstable for our kind of data and instead decided to employ a combination of mean, median and standard deviation. Starting with the analysis of shape variations, we decided to use the median to extract a concrete element inside the shape space as a general representation of the group. The reason for this is that we wanted a shape that is actually present in the cohort and is not easily influenced by outliers. This is especially important in cases where the bladder of a patient has a vastly different volume for only one treatment instance. In regards to the analysis of the general shape variations a patient undergoes, we

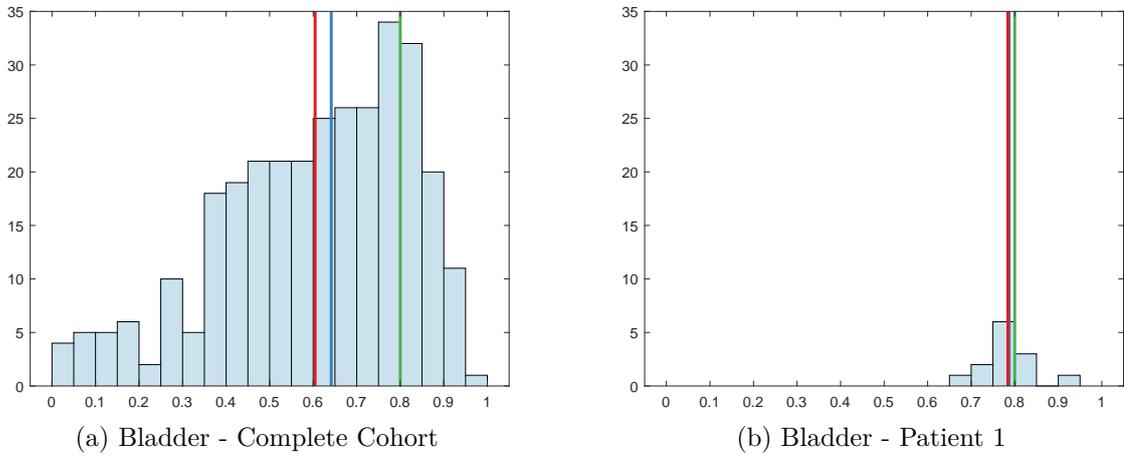


Figure 4.38: Examples of the distribution of the first principal component concerning the bladder shapes, once for the complete cohort and once for the first patient. We compare how descriptive the center measures ■ mean ■ median and ■ mode are for each of them. The median and mean are slightly apart for the skewed complete distribution. This effect vanishes if we focus on individual patients.

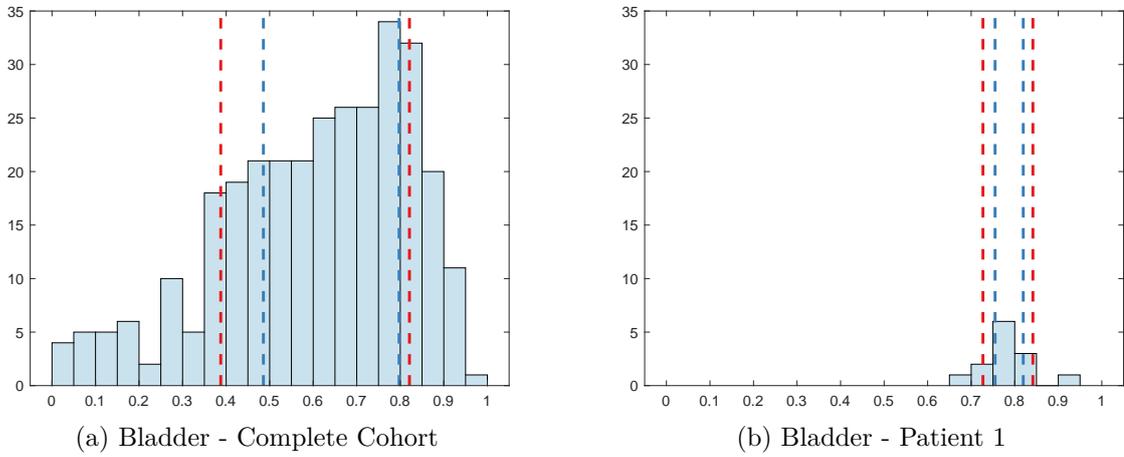


Figure 4.39: Examples of the distribution of the first principal component concerning the bladder shapes, once for the complete cohort and once for the first patient. We compare how descriptive the disparity measures ■ standard deviation and ■ interquartile range are for each of them.

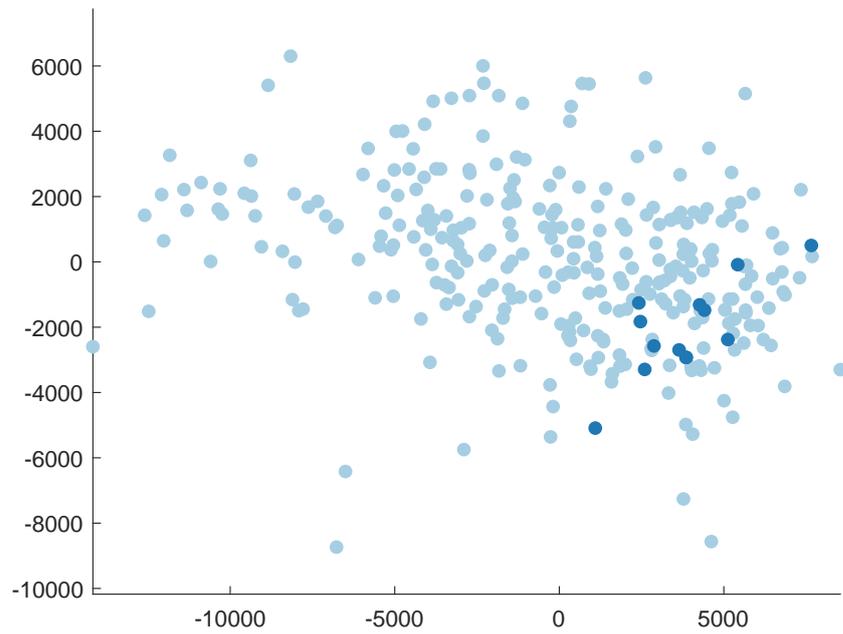
	Rectum	Bladder	Prostate	CTV
Shape	95.83	100.00	100.00	91.67
Position	100.00	100.00	92.86	94.44

Table 4.3: Percentage of all shapes and positions that follow a normal distribution if tested on a per-patient basis with a significance level of 5%.

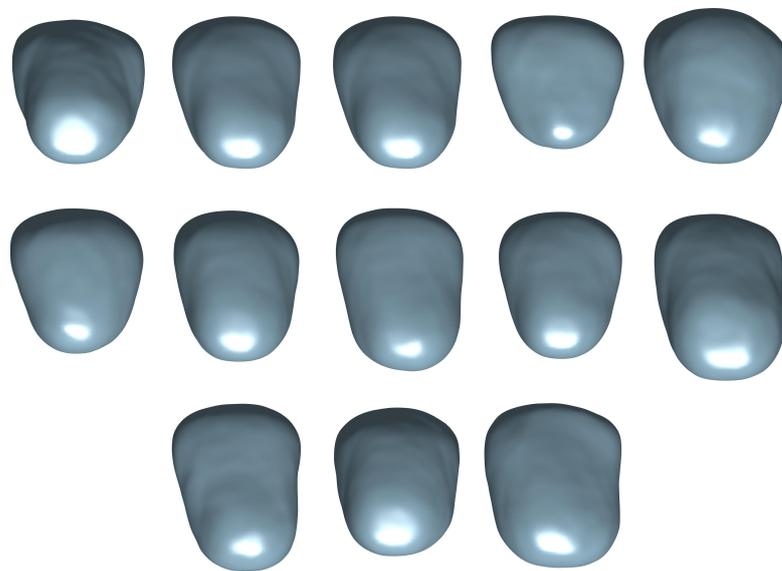
want to include even these outliers, as they can give information about which directions certain organs tend to grow if their volume changes. Effective usage of the standard deviation requires the data to be normally distributed. Although the shape space data as a whole does not follow this distribution, analyzing the distribution of the timesteps on a per-patient basis, we see that nearly all of them are normally distributed to some extent as seen in Table 4.3. Therefore we deem the use of mean and standard deviation as appropriate for the analysis of organ shapes. This combined approach, incorporating both the robust measure of the median as general shape representative and the standard deviation to calculate the upper and lower boundaries of the confidence interval, was already employed by previous works like the one of Ferstl et al. [FBW16, FKRW16]. The application of their method to our case is visualized in Figure 4.40. In regards to the analysis of the center point variations, which are supposed to indicate organ movement, the values themselves are also normally distributed for each individual patient and we, therefore, decided to use both mean and standard deviation to calculate the general center point and the main variation directions for groups of organs. This also in accordance with our registration method, where we also used the average center point for each patient to align its organs before the analysis.

4.3.2 Shape Display

Our application requires both a general understanding of multiple complex organ shapes in a scene, as well as a precise inspection of the overlaps and interactions between organs and different patient shapes. Therefore we decided to use a combined 2D/3D display, shown in Figure 4.41. We employ three slice-based contour views for each major axis direction through our 3D space as well as a 3D overview representation. Each of these views has the same size and they are arranged in a 2×2 raster. Hereby we also exploit previous knowledge of medical experts who often use tools that employ similar views, for things like MRI or CT data, where the volume is viewed in the sagittal, axial, and coronal planes, respectively. The idea to employ contour plots is that they allow users to properly judge distances and shapes at a given point in space. They lack any depth perception, therefore we augment them using a 3D overview. In this 3D view, we not only highlight the currently shown organ shapes, but also indicate the position of the slicing planes determining the contours in the 2D views. The user can either zoom in on details in the 2D views or move the slicing planes back and forth to scroll through the different shapes. Simultaneously, the plane indicators are updated in the overview representation.

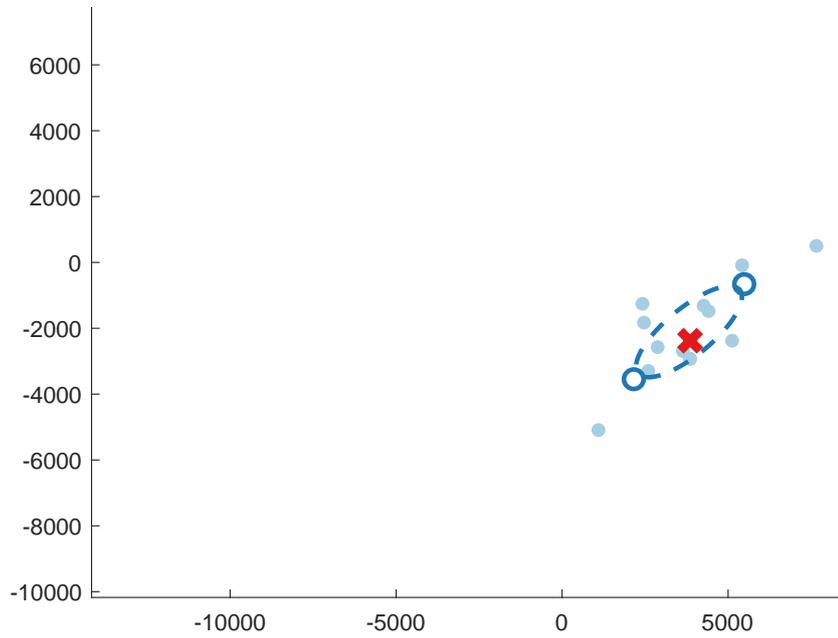


(a) Bladder shape distribution across the first two principal components. The bladder shapes of the first patient are highlighted.

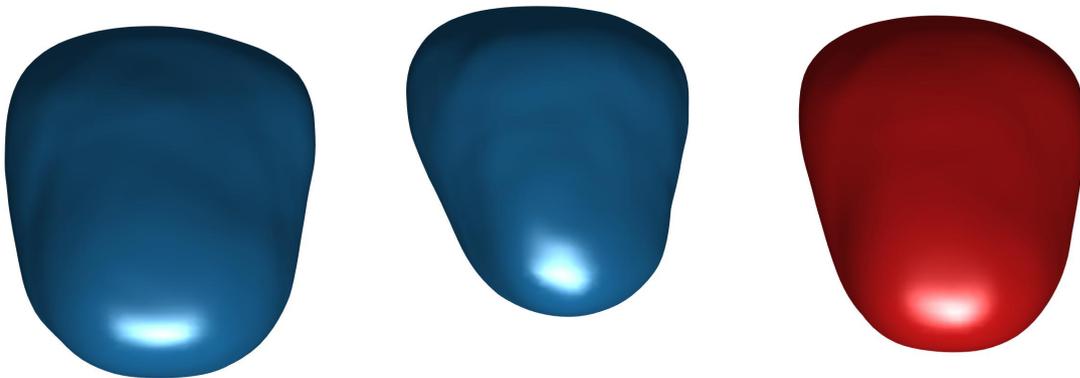


(b) Each bladder shape of the thirteen timesteps from the first patient.

Figure 4.40: Example for statistical shape analysis based on the used bladder shape data.



(c) Shape space positions of the first patient, with the used statistical measures: median and standard deviation.



(d) Lower Confidence Shape

(e) Upper Confidence Shape

(f) Median Shape

Figure 4.40: Example for statistical shape analysis based on the used bladder shape data (*continued*).

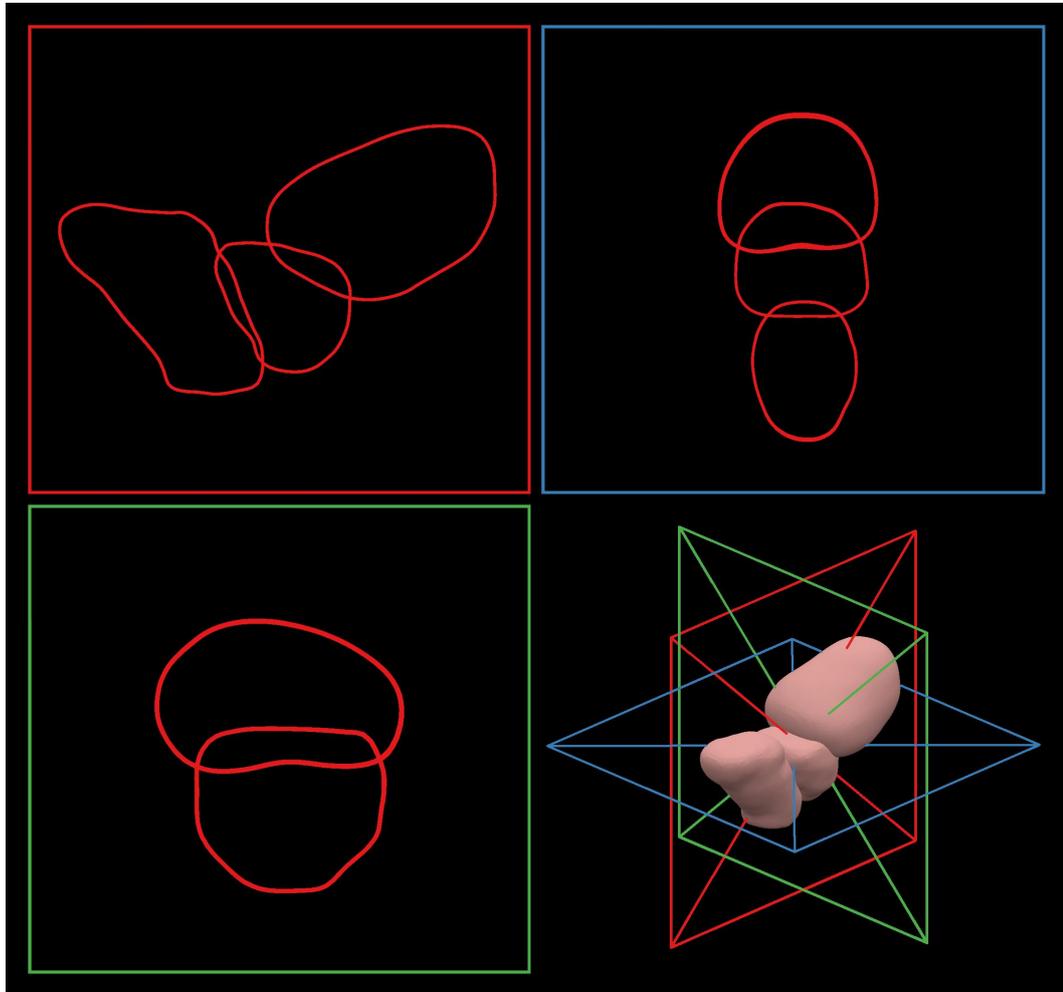


Figure 4.41: Screenshot of the final combined 2D and 3D representations. Each of the three 2D slices is based on the three medical planes (sagittal, axial, and coronal) and is assigned a specific color. The current slice position is shown by the colored rectangles in the 3D view.

In addition to this overview representation, the 2D views containing contours of the organ shapes are employed for further analysis of the distances and overlaps between multiple organs in the cohort. This is in accordance to what is usually employed in medical applications.

4.3.3 Overview Visualization for Shape Comparison

Having decided on the use of both 2D and 3D views to show the organ shapes, the next step is to decide on what visual encoding strategy should be employed in the respective views. For the 3D view, we have to find a way to provide users with a good overview of the currently shown organs, keeping in mind that often the organs of not only a single patient or group are analyzed, but instead multitudes of them are to be compared. Due to our statistical analysis, we obtained information about the general shape as well as its variations, in the form of the upper and lower confidence interval shapes. In this view, we mainly want to provide users with an understanding of the general organ placement and the respective margins between multiple organs. In addition, we aim at highlighting regions where large shape variations are present. This should allow the user to place the slicing planes of the linked 2D views in such a way that these regions of interest can be inspected in detail in their respective views. All of these tasks should also support the extension to not only one group but many. The involved challenges, be it the comparison of the confidence interval to the general organ shape or the comparison of multiple different organ shape types to one another, fall under the topic of comparative visualization. We will, therefore, look at three common encoding types in this field, juxtaposition, superposition, and explicit encoding and analyze them for our use.

Although we previously mentioned these comparative visualization types in Chapter 3, we briefly repeat their main focus and how it applies concretely to the use of 3D organ meshes. Juxtapositioning means placing the compared elements next to each other. In our current case, this could include placing all organs of one group next to each other or arranging median, upper, and lower confidence shapes side-by-side. Superpositioning places all shown elements in the same space, meaning that, for example, the same organs taken from multiple patients or groups are stacked in one place on top of each other. Often transparency is employed to show the inner structures and shapes of the organs and not only the outermost one. Finally, explicit encoding extracts the information we are interested in and renders it directly. In our case, this could include the extent of the shape variations at a given point on the surface of an organ. One way to encode this would be to employ color coding. Generally, animation could also be considered for our purposes. Yet, we deemed the estimation of the necessary intermediary steps as computationally not feasible and did not consider it further in our evaluation.

We use several principal factors to evaluate each of the possible visual encodings. As we aim at creating a 3D overview representation to support the other 2D views, the chosen encoding needs to provide users quickly with the main information about *scene composition*. To support the use of the slicing tool, how well does the selected encoding support users in detecting *regions of interest*, e.g. ones with high surface shape variations

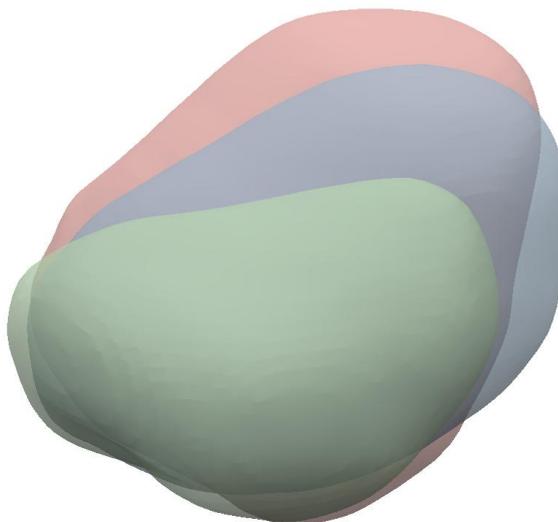


Figure 4.42: Example of superpositioned bladder shapes. Each shape has a distinctive color and is drawn transparently.

and hereby the precision of the process is of importance. Additionally, the encoding should *preserve* the original shape positions in such a way that it is still linked with the 2D slice views. As the current display layout only leaves a quarter of the available space for the 3D view, the encoding should also use the maximal amount of shown *information*. The opposite is true for the involved *interactions*, where any approach should aim at minimizing the amount necessary, due to us mainly aiming at providing a simple view, that should not distract from the more detailed 2D views. Each of these properties needs to be assessed in terms of their scalability for multiple organs and groups.

Placing multiple shapes on top of each other (*superposition*) in the same position, as shown in Figure 4.42, can allow users to quickly assess their general shape and especially size differences. Larger shapes tend to envelop and occlude smaller ones, which can be an effective way to determine regions in which a certain shape is larger than another one. Combined with transparency, users can also estimate the size and shape differences for smaller shapes completely occluded. This has complex rendering requirements as it can easily happen that an improper rendering order of transparent surfaces leads to artifacts. Depending on the chosen rendering method, the order in which the organs are drawn might also influence the result. Concrete shape difference estimations are not easily done in this view and often suffer from low precision. This is getting even more complicated if transparent structures in the background become visible through transparent surfaces. In general, this representation can either center the shapes on a per-group or per-organ basis depending on which of those two are compared. Both preserve the relative placement of

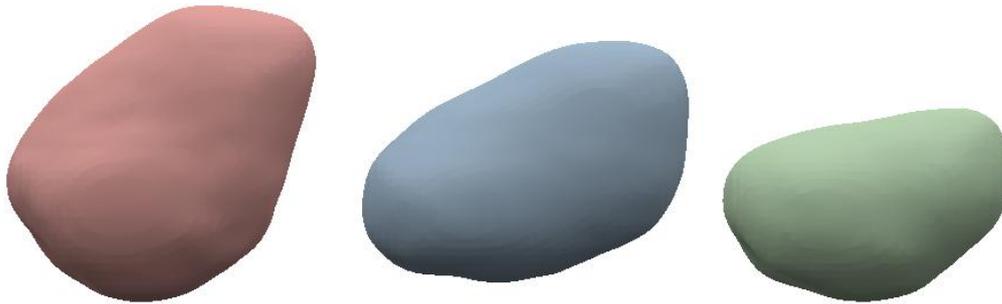


Figure 4.43: Example of juxtapositioning, where the three bladder shapes are drawn next to each other.

the involved shapes in such a way that the 2D slice views can be effectively employed to explore the shapes in more detail. In terms of used screen space, this approach can effectively render the organ shapes at their maximal possible size. To explore the view effectively, the most basic 3D interactions, i.e. zooming and rotating should be sufficient to gain an overview. Overall the approach works best if only a few of stacked elements are compared to each other. The more elements are centered at a given point the harder it gets to estimate differences between them.

Similar to the alignment used in the superposition encoding, the *juxtaposition* approach can either arrange the shapes next to each other on a per-group or per-organ basis, the former one is depicted in Figure 4.43. This placement provides users with a quick overview conveying both the general number of elements as well as distinct shape types. This type of comparison visually separates the compared elements, meaning that it may not be ideal to understand the upper and lower variation shapes of a single group. It is more focused on comparing multiple separate groups to each other. One problem of this approach is that while it allows users to gain a quick overview of the general shape variations, no precise shape comparison is possible. The comparison relies on the user's memory as he moves his gaze between the different elements. This encoding does not preserve the initial space completely, as it moves the elements in the direction of the viewing plane axes in such a way that they do not overlap. This also means that the elements in the 2D views need to be adapted, otherwise elements at the same depth would be moved apart and could not be directly compared in the 2D views anymore. With every element placed in such a way that it does not overlap with any others, the available screen-space becomes smaller and smaller the more elements are added. This also has implications on the necessary interactions. Apart from zooming and rotating, panning is needed if the user wants to focus his attention on one element.

Lastly, the *explicit* encoding shown in Figure 4.44 drastically reduces the number of shapes displayed, compared to the other aforementioned methods. This reduction comes

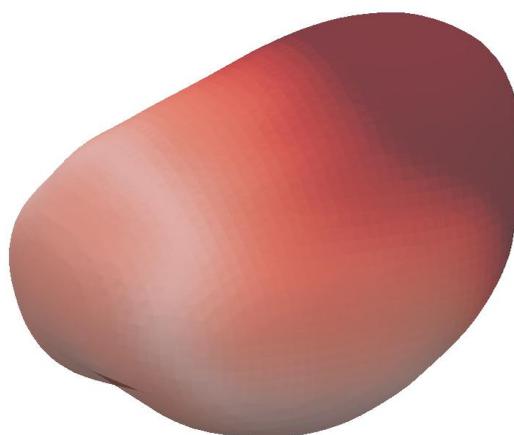


Figure 4.44: Example of explicit encoding, all the shapes are aggregated to a median and the deviations are encoded in color (low variations  high variations).

with the price of losing the concrete shape information. Only by close inspection of the explicitly encoded extent of the shape variations can the user understand the present shape differences. This may be too complex for a quick overview tool. The encoding allows for a precise analysis of local shape variations. This, in turn, can support users to quickly detect regions of interest for slicing planes in the 2D views. Because this method uses no displacements and only visually aggregates the differences, the guiding slices can still be used for both 3D and 2D views. As the estimated shape variation values only encode the amount of deviation at any point, the actual number of aggregated shapes and their respective shape variations is hidden. Similar to the superimposed view, the screen space used for the displayed shape is as large as possible and the required interactions are also kept at a rather basic level, including only zooming and rotating. One could also add the option to change the shape, which serves as a basis for the difference calculation and encoding, to better understand varying shape types. Overall this method's visual output is mainly independent of the number of input shapes and scales rather well in this regard. This also comes with a loss of providing contextual information about the actual shapes.

In the end, we decided to use both a superimposed 3D view and one explicitly encoding the shape variation amount. We assume that in general, the analysis of organ shape variations has the following steps. Firstly, multiple different groups are selected by either clicking on a cell or a row label in the cohort visualization. Afterward, the median shape and its variations are estimated. Each selection gets assigned a unique color from a

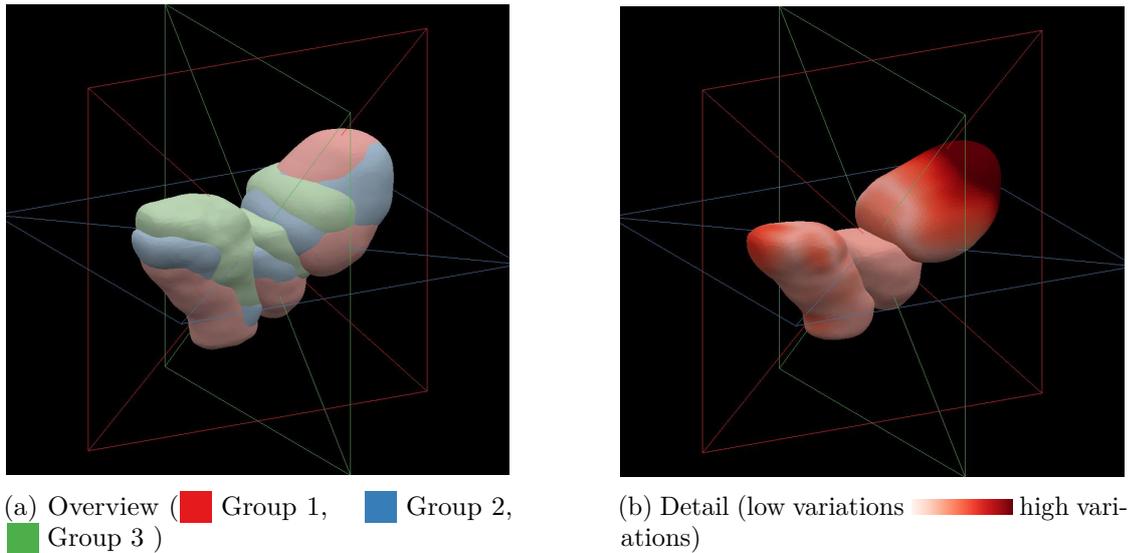


Figure 4.45: Screenshots of the employed 3D overview visualizations.

nominal coloring scheme taken from Colorbrewer [HB03]. The selected elements are indicated in the cohort view by a colored border or label. After this step shape analysis starts out with a general comparison. We, therefore, show the median shape of each selected group superpositioned in the 3D view. Hereby each organ of the same group is assigned the respective group color. The idea behind this arrangement is to present a simple guide to the general location of the involved organs in the scene. The lighting in the scene and the used surface material aims at highlighting the organ structure. We also did not employ transparency, as this view is only supposed to provide information about the general organ shape layout and the other 2D views should provide more detailed information. Finally, we also included a detailed explicit encoding of the surface variations, which is only visible if the user selects a single group, whose variations are to be explored in detail. In this case, the median shapes of other groups are hidden and the selected one's surface color is used to encode the amount of surface variation found using a sequential colormap based on the organ's group color. With this view, we aim at supporting users trying to find regions of interest in terms of shape changes for a selected group. In Figure 4.45 we show how these methods are used in our tool.

4.3.4 Details-on-Demand Visualization for Shape Comparison

After providing users with an overview of the 3D structure of the involved organ shapes, the final step in our intended shape analysis process is to provide them with a detailed inspection of the shape type and especially the underlying shape variation extent. For this, we already decided on employing three 2D views similar to the medical views commonly used for volumetric data, showing the contours of the shapes present at a given time. Our goals in regards to the organ shapes are to allow users to understand how

large the deviations for specific organs are in certain regions, as well as to highlight the various regions where organs overlap. Furthermore, we are concerned with the necessary encoding of the position variations inside an aggregated group of shapes, which indicate the respective organ centers. This allows users to understand in which directions organ tend to move during treatment. Both of these goals get more difficult the higher the number of involved groups is and especially if one aims at comparing the different groups to each other. Similar to the 3D case before, we need to employ some kind of comparative visualization technique to handle the amount of information present. We, therefore, look at three techniques with different methods for tackling the given problems: stacked contour plots exploded views, and contour variability plots. These techniques are already commonly employed in the analysis of weather forecast data and we explore their viability in our usage scenario.

Stacked contour plots are probably the most simple variant, which simply uses superpositioning to place the organ shapes on top of each other [KCK17]. To avoid confusion between elements of one group, like the median and confidence intervals, and the ones of other groups, color coding could be employed. Exploded views, generally fall into the category of juxtapositioning, where elements are displaced from their original position and placed next to each other [LACS08]. For the use in our case, this could include extruding specific groups or organs from their original position for a detailed analysis, while the rest of the shown contours remains in place. Contour variability plots are similar to boxplots in that they encode shape variations in the form of a confidence interval with the median shape at its center, aggregating the compared elements, and only showing their underlying statistical shape variation properties [WMK13, FBW16].

Our decision, to employ one of these methods is based on the following properties, determining how well suited they are for our tasks. First and foremost, we want to know the *precision* with which users determine the encoded shape variations between median and confidence shapes. The method should allow the user to focus on either understanding the shape variations and shape types for the *whole patient* or a *single organ type*. During the analysis of one organ or group, the other shapes are supposed to be preserved for *contextual information*. In some way, the *overlaps* between organs should be visible. For each organ the information about the changes in their *position* also needs to be included. Finally, we need to know which kind of *interaction* is necessary to explore the data with a given approach. As before, the most important aspect of all of these properties is their behavior if a *larger number of elements* is compared.

By simply *stacking* the contours of all organs and groups, such as in Figure 4.46, one can determine the amount of variation by looking at the distance between the median shapes. This gets more and more complicated the more contours are added and the more overlaps are present. This is a general problem of the representation. The more elements are compared the more visual clutter is present on the screen and the harder it gets for users to understand individual elements. In this view, the focus of the comparison largely depends on how the shapes are aligned. If they are aligned based on their original position they support the analysis of interactions between different organs. If they are

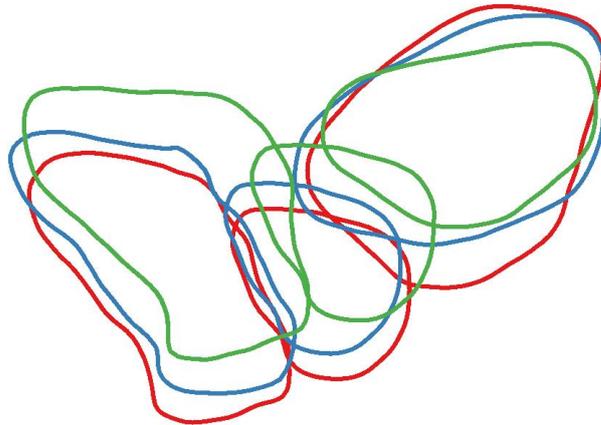


Figure 4.46: Example of an stacked contour plot for three individual patient groups, each showing bladder, rectum, and prostate.

aligned to a common center on a per-organ basis, it supports the analysis of general organ shape types. Generally, all contextual information regarding the relation to other organs and the involved shape variations are always visible in this view. It is possible to provide users with tools to hide certain contour lines to provide a more manageable view. This also leads directly to the necessary interaction, which includes on top of the visibility toggles functionalities like zooming, translation and changing the slice through the volume. Rotation is not necessary as we are dealing with 2D views. Overlaps are implicitly encoded as the regions between intersecting contour lines. The positions of the organs themselves could be represented as dots, since they are already placed in the same space.

Exploded views address the problem of the previous approach, where high numbers of elements lead to too many contour intersections, which make the interpretation difficult. With exploded views, parts of interest are manually extruded, as seen in Figure 4.47. This can allow users to focus on a selected group or organ and inspect their shape variations and types in detail. A problem of this method is that the contextual information provided through surrounding structures is missing, e.g. in the case of an extruded organ, the overlaps with the others are lost. One could, therefore, try to provide visual guides to the initial position, like ghosted through views. To inspect desired organ shapes, one needs a tool to select which group or organ should be extruded for a detailed analysis. Apart from that, the standard interactions are necessary. The scalability of this approach largely depends on the number of extruded elements, the more there are the less space each individual element has. As we are dealing with medical data, we need to convey the amount of distortion we introduce with this representation to the user.

In the *aggregated* contour box plot, visible in Figure 4.48, the concrete shape variations are shown directly as colored regions, allowing for a more direct understanding of their

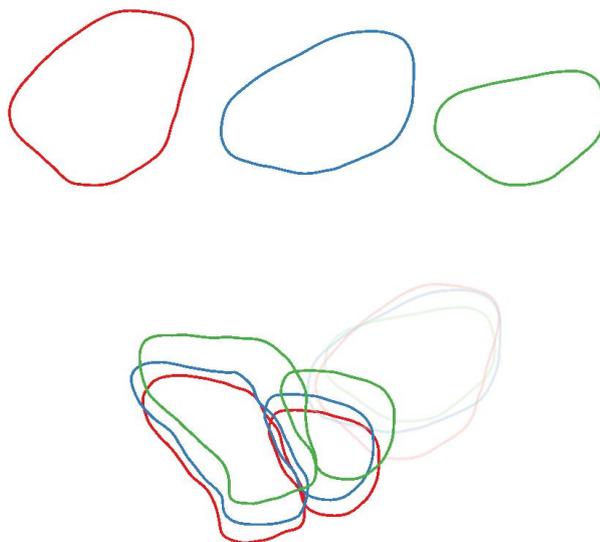


Figure 4.47: Example of an exploded view for three individual patient groups, each showing bladder, rectum and prostate. The bladder shapes are moved from their initial position and shown next to each other. A ghosted version of them is left behind.

extent, compared to contour line plots [MWK14, WMK13]. This aggregation gives a probabilistic view of the contained shape types. Assuming that this aggregation happens on a per-organ basis, the overlaps between organs can be indirectly encoded as the overlaps between the variability lobes. They can be highlighted by drawing the lobes slightly transparent. Positional variations can also be aggregated as lobes in 3D space, which stretch into the main variation directions. For this encoding, the same interaction methods suffice as for the other methods. Regarding its scalability, the aggregation process delivers always one lobe and one median independent of the number of aggregated elements. This could also mean that certain details, like outliers or smaller shape groups, are lost [WHLS18].

In the end, we decided to combine all three approaches. Firstly, we use the contour box plot approach to encode the shape variations for a selected group in the cohort. Similar to the 3D view, the basic color scheme of each group is also used to encode its shape variability lobe and median. If more than one group is selected, we position the lobes on top of each other and show mainly the median shape of each of them with their respective variability lobes drawn transparently. This aims at allowing users to compare general shape types. If desired the user can select one group, by simply clicking on its shape, after which its lobe is drawn more opaquely to highlight the shape variability. It is also drawn on top of all other stacked contours to provide an unobstructed view. The general strategy follows a focus and context approach, where one shape is inspected in detail and the remaining ones are still present in the background. This also affects the 3D

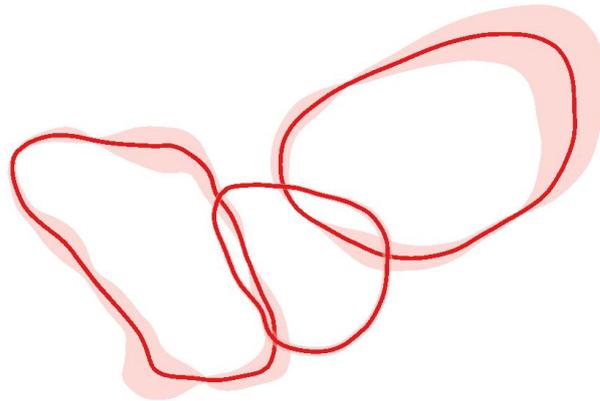


Figure 4.48: Example of an aggregated contour plot showing bladder, rectum and prostate. The individual shapes are aggregated into a shape median and a confidence interval.

view, where instead of showing all groups superimposed, only the currently selected one is depicted with its shape variability color-coded on the surface. This process mainly aims at allowing users to focus on exploring individual shape variations on a per-group basis. Examples of the actual implementation are depicted in Figure 4.49. If the user wants to explore these variations with a focus on a specific organ we provide the option to extrude it. In this exploded view, shown in Figure 4.50, the same organ type in all groups is taken and placed in such a way that it does not overlap with any other shape, while at the same time being centered at a common point [BVG10]. This allows users to compare the actual organ shapes to each other, without the influence of varying initial positions. To preserve parts of the initial context a guidance line connects the center of the extruded organ to its original position. The positions themselves and their variations are explicitly encoded by drawing ellipsoidal glyphs that deform in the direction of the highest positional variance, to indicate the main directions in which the organs move. The user can manually select both the shown organs and the visual encodings, like glyphs and contour variability plots, which can be seen in Figure 4.51.

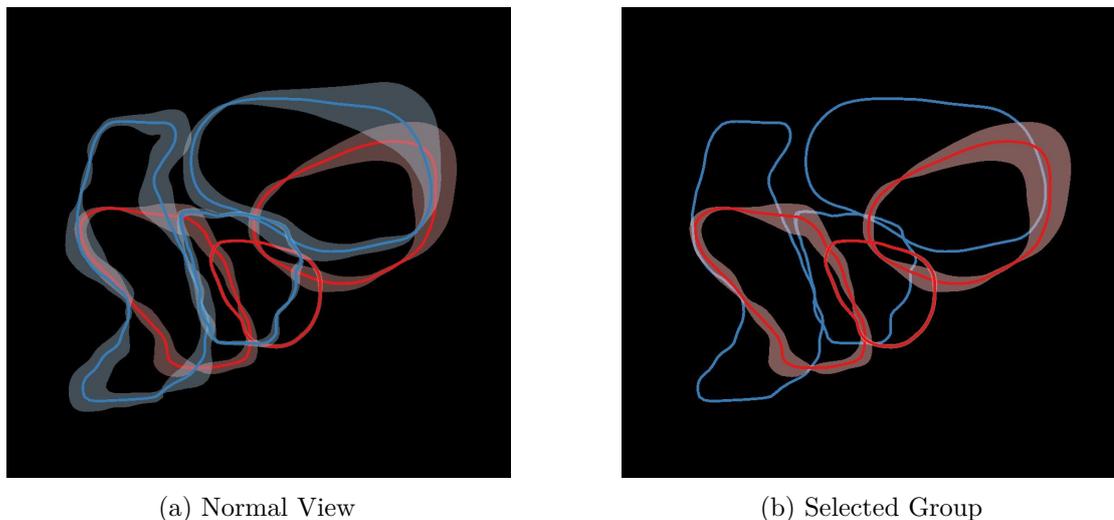


Figure 4.49: Screenshots of the 2D slice view. The shapes inside a group are drawn using contour variability plots, while multiple groups are stacked on top of each other. Selecting one group highlights their shape variations and hides the ones of the others.

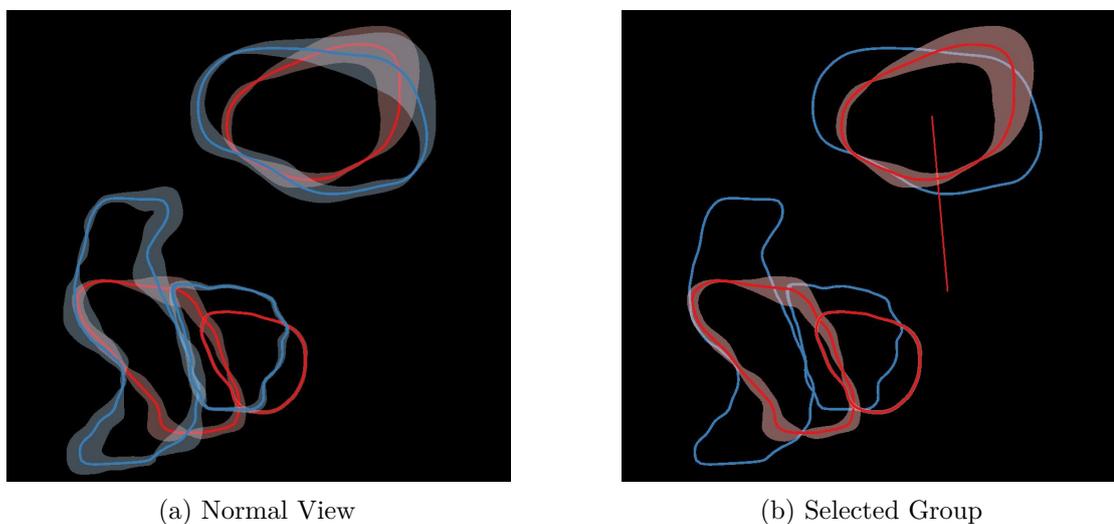


Figure 4.50: Screenshots of the 2D slice view, where a certain organ shape is extruded for an exploded view. All shapes of a certain type are moved away from the rest to prevent overlaps and are aligned based on their center points. If a shape is selected, its shape gets highlighted and a guidance line is drawn to its original position for context.

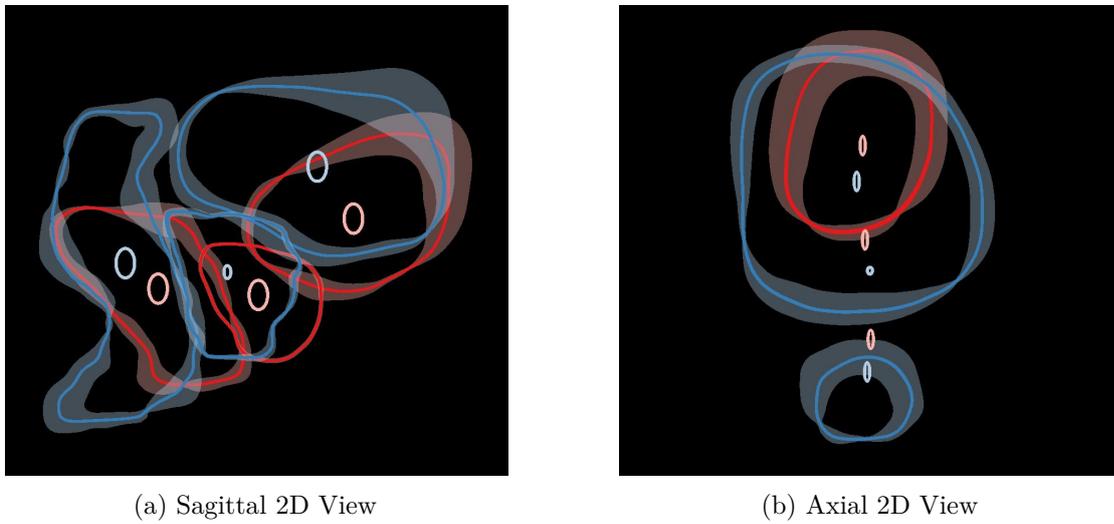


Figure 4.51: Screenshots of the 2D slice view, with the added position change glyphs, indicating the average movement of the organ centers. It is apparent that the organs generally move mostly in the directions of the sagittal plane.

Implementation

All the aforementioned techniques need to be implemented in a system that not only performs all the necessary tasks, but also fulfills the requirements we defined in Section 1.2.2 in regards to scalability, generalizability, and usability. These three requirements turned out to be hard to combine. It was difficult to create a system, which provides a responsive user interface and can handle large amounts of data. The responsive user interface is needed for the quick exploration of the cohort.

5.1 General Structure

As we found out that not a single platform on its own was able to handle all requirements, we decided to employ a four-component system. In the following, we want to discuss both the general system structure and the contained components. Relating to the Chapter 4, we will explain at which stage which parts got realized. Starting off with the highest level, we had three main parts to implement: shape analysis, cohort visualization, and shape visualization. The first one required the processing of large patient cohorts. The latter two are more focused on presentation and quick analysis of the shape variations. We decided to perform the shape analysis beforehand and visualize the resulting shape space results in real-time.

The flow of the data through our system, whose structure is shown in Figure 5.1, is as follows: Initially, the segmented data is transformed by MeVisLab into a format processable by MATLAB, which then analyses the organ shapes, after which it provides shape information on demand using a C# server. The server is able to supply our tool implemented in a web application with the shape difference information and statistical shapes.

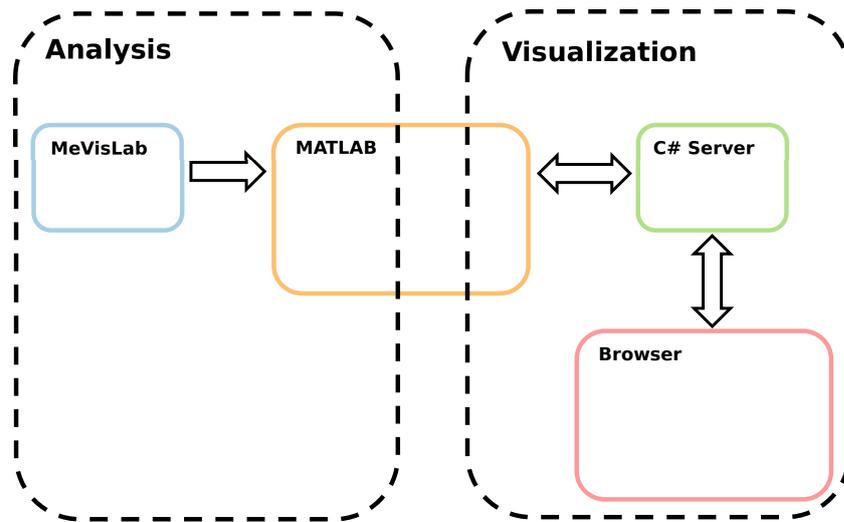


Figure 5.1: Overview of the general system structure, with the four main components.

5.2 MeVisLab

Collaboratively developed by MeVis Medical Solutions and Fraunhofer MEVIS, MeVisLab¹ is a framework that allows users to build rapid prototypes of medical visualization tools. In this regard, it offers a wide range of modules that can read, transform, and visualize medical images.

For our purpose, we employ it as an offline pre-processing tool preparing the data for our application. The organ segmentation data is organized in folders for patients and timesteps, each of them containing a file format used specifically in radiation therapy planning. Due to MATLAB not being able to process this kind of data, we used MeVisLab to iterate over all organ line ensembles and transform them into volumetric image data as described in Section 4.1.1. Due to recent updates in the tool a combination of MeVisLab 2.8 and 3.1 was necessary to perform this task. Only the former was able to read the radiation therapy data, while only in the most recent version the functionality to transform these segmentation lines into coverage masks was implemented. Therefore the process starts with MeVisLab 2.8 reading the radiation therapy data and transforming them into the commonly understood segmentation data format `.cso`, which is then read and transformed by MeVisLab 3.1 into a volumetric coverage mask format. As all contours across patients have vastly different names, we also need to provide a list of labels. The list is used to determine the contour corresponding for each organ.

¹<https://www.mevislab.de>

5.3 MATLAB

The next part of the shape analysis process is performed using MATLAB. This tool provides a wide variety of operations for the processing of matrix-based data, ranging from simple vector calculations up to complex machine learning support. It is not as efficient in its calculations as native C code, but its wide range of tools and their ease of use allowed us to quickly develop and test different approaches.

Taking the volume coverage masks from MeVisLab, MATLAB performs the next steps in the shape analysis process. The following procedure is rather memory intensive as the shape analysis with PCA using the covariance matrix, requires the whole cohort data to be present in the memory at once. To alleviate this burden we decided to split the creation of the shape space and the reduction of the cohort elements and perform it on a per-organ basis. Initially, the coverage masks are transformed into distance volumes and aligned based on the registration procedure described in Section 4.1.3. The organ volumes are then vectorized and arranged in a shared cohort matrix using the sampling procedure from Section 4.1.4. Following this, the cohort matrix of the organ is analyzed using PCA and its principal components are calculated. Hereby only that many components are used to ensure the preservation of 99% of the overall shape variability. With this knowledge, the shape volumes are reduced to a lower-dimensional space and the procedure is repeated for the remaining organs. After each iteration, the working memory is cleared to ensure that there is enough space for the next organ. While we performed this analysis on the complete cohort, after the initial shape space calculation the base matrix can be employed to reduce the organs of new patients and integrate them into our system. The whole process takes around two to three hours to perform all preprocessing steps and create the shape space with its elements. All of these operations were performed on a laptop with an Intel i7 processor and 16 GB of working memory. We save the information after the calculation and only load the data when the server is started and it is needed for the following visualization steps.

MATLAB is not only used as an analysis tool for the organ shapes, but also is involved in their visualization. Starting off with the cohort visualization, we employ MATLAB in estimating several different characteristics that we highlight in our tabular view, many of which are described in Section 4.2.1. Most prominently, we use MATLAB to estimate the shape difference values for the different organs across the cohort. By calculating the mean shape across a patient, the timesteps to be used for the calculation can be specified, e.g. we mostly use the first five. The machine learning capabilities of MATLAB are used to estimate similarly shaped clusters. For this it is possible to specify which organs shall be considered for the clustering. While often the possible shapes of a single organ are analyzed, users may also be interested to find similar patients. In any case, the calculation process results in several matrices, which are made available to our C# server using a COM interface. This interface is able to transmit information between an external program, like the used server, and the MATLAB engine. It can be used to start calculations and emit the results to external sources.

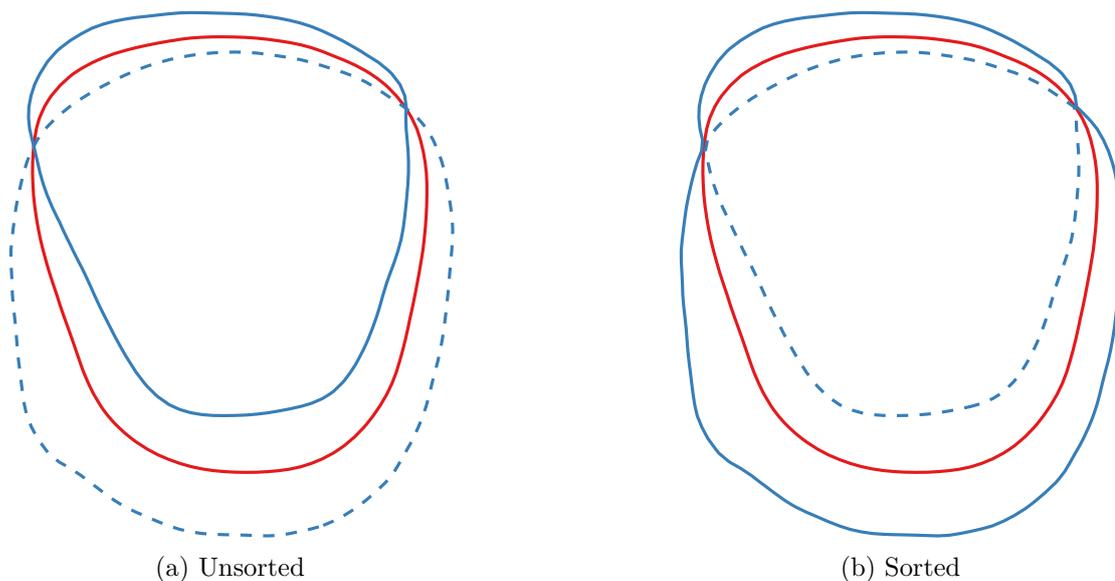


Figure 5.2: Comparison of the unsorted and sorted upper and lower confidence interval contours (— Median Contour - - - Lower Confidence Contour — Upper Confidence Contour).

Another use of MATLAB in the visualization process is the creation of statistical shapes. As mentioned in Section 4.3.1, the statistical analysis of the shape space can provide organ shapes that correspond to the upper and lower confidence intervals or the median of a group of shapes. As we use volumetric data as an input for the shape analysis, we also receive volumes as output.

At this point, an important correction step is needed. Organs like the rectum with large shape variations that wiggle through space can lead to the effect that the upper and lower confidence shapes move in and out of each other. This is unwanted as the following visualization procedures assume that the outer shape is a complete envelope of the lower one. To achieve this property, we make use of the fact that the volumetric representation with signed distance values to the surface of the object can be interpreted as a discretized indirect surface and simply calculate the union and intersection surfaces. The union surrounds the whole variability region, while the intersection surface represents the innermost region to which the shape variability extends. This process was also described in the work of Ferstl et al. [FKRW16]. Figure 5.2 shows how this process works on a simple 2D contour example.

As discussed in Section 4.1.1, the resulting volumetric data format is difficult to visualize and transmit from a server and we, therefore, extract a triangle mesh by using the isosurface extraction function from MATLAB. To reduce the size of the mesh we encode its vertices in the binary version of the `.stl` format. This format in its most basic version is not able to save color information for the individual vertices. As this is required

for the explicit variability encoding described in Section 4.3.3, we save this information in an additional matrix to be read by the server.

5.4 C# Server

We decided that the user interface of our tool is best implemented using a web-based application. This approach reduces the setup effort for possible users and makes it available for a wider audience. To achieve this we need a web server that is both able to host the website for the user interface as well as connect back to the MATLAB COM-interface. Because of this, we chose to implement a C# Server, or more precisely an ASP.NET core web application that is hosted on an IIS (Internet Information Server). Similarly to PHP, this setup allows us to develop complex web applications. The server not only provides and processes web pages but is also able to translate certain requests to function calls. These functions are implemented in controllers. Our final server has two of these controllers.

The first one is called `ValueController` and provides upon request several different shape characteristics by connecting to the MATLAB engine. Most importantly, it returns the shape difference values for a specified organ and center point. These two parameters are provided on request in the URL. The parameters are then passed to MATLAB and the function to estimate the difference values is called. After finishing the calculation, the server reads back the result into its own memory and translates it into the JSON file format to be passed back to the client. Similarly, the controller handles the clustering of certain organ shapes, where one or several organs are grouped into a specified number of clusters, whose individual ids are returned as a result. The server also provides the toxicity information for the individual patients, which is not calculated but known beforehand.

The second controller is the `MeshController`, which provides the client upon request with different reconstructed 3D meshes and their related information. Most basically, it returns the median, upper, and lower confidence interval shapes for a freely selectable group of patients and timesteps. The elements that serve as a basis are specified in a 2D array of patient and timestep information, which is provided as a parameter. Furthermore, the server is also able to return the position at which the reconstructed organ shape is centered, as well as the surface coloring required for the explicit encoding of the shape variability on the mesh.

5.5 User Interface

Lastly, we discuss the main component, with which the user will interact, the web interface. We aimed at providing the user with a simple website that provides the necessary shape variability information, realized in the discussed visualization approaches. These visualizations are implemented using JavaScript in combination with the libraries `D3.js` and `three.js`.

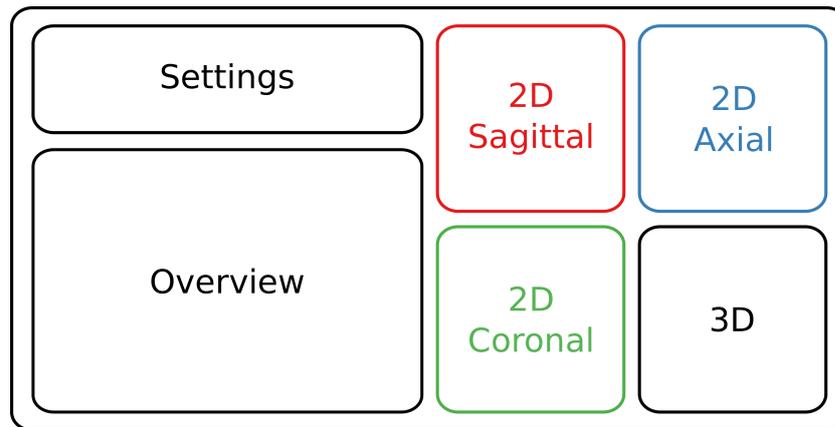


Figure 5.3: Illustration of the user interface structure.

The basic layout is shown in Figure 5.3 and defined through HTML. It includes the following elements: *Settings* Panel, 2D Cohort *Overview* Visualization and four different shape visualization windows, three of which are 2D slice viewers and one of which is a 3D view. One half of the screen is reserved for the settings and the cohort overview, while the other half is used to visualize the local shape variations.

The settings region itself is split in half similar to the general layout, the left half is based on Section 4.2.2. Here we provide the user with the possibility to sort, filter, and group the data on demand. For these operations the users needs to select the properties based on which the patients shall be arranged. The user can select which organ or property, like toxicity, the elements should be sorted and grouped, as well as defining which of the organs should be displayed in both the overview and the 2D and 3D views. The settings also allow the user to define the timesteps from which the mean shape of each patient is calculated. Following the "Details on Demand" approach we provide switches to show or hide information like the distribution (Section 4.2.3), missing (Section 4.2.4), and clusters (Section 4.2.1). The second half of the settings region concerns the shape variability visualization in the 2D and 3D views. Hereby, the user can select whether or not the median shapes, the variability lobes, and the position variation glyphs are displayed. Moreover, we can select one of the organs displayed and extrude them in an exploded view Section(4.3.4).

5.5.1 D3.js

The JavaScript library D3.js ² aims at supporting the development of dynamic visualizations in web applications using HTML, SVG, and CSS. Its general concept is based on the idea of simply providing the data to be displayed and transforming it into various, visual elements. It also offers a wide range of features and extensions, including the possibility to animate changes in the data and the incorporation of colormaps from Colorbrewer

²<https://d3js.org>

[HB03]. For our tool, we use the library to create an overview visualization of the whole cohort explained in Section 4.2. After downloading the shape difference information from the server, the data is displayed in a table with patients and timesteps. We first transform the data into an object-based format where each element is identified by the organ, patient, and timestep. This transformation allows us to process the data more easily with D3.js and JavaScript.

Contrastingly to the general approach behind D3.js we do not provide the rendering information directly to the library for visualization, but instead derive our own render objects from the data. These objects are created for every visible element in the cohort visualization and serve as an intermediate form before D3.js renders them. They encapsulate all the information necessary to create the visual elements, e.g. the colored rectangles for the tabular view. The render objects are created once after loading the data from the server and are dynamically updated if the shape difference values are recalculated or the groups change. D3.js is then provided with these objects and translates them into the colored rectangles, labels, and other glyph elements that make up the tabular overview visualization of the cohort. Most rendered objects, like the labels and rectangles, can be clicked by the user to show the detailed shape variations.

To facilitate the understanding of the user when interpreting the cohort visualization, a legend is provided. The color-coding is explained with two colorbars, i.e. one for the shape differences and one for the inner group deviations. A simple glyph representing the rectangle shape of the timesteps is shown where letters indicate which part of the rectangle corresponds to which organ if multiple organs are encoded at once.

5.5.2 three.js

For the implementation of the aggregated contour plots another JavaScript library, three.js ³ is used. Its basic functionality is to provide a simple wrapper for WebGL, allowing us to render 3D objects in the browser. It provides a variety of utility functions, for loading and transforming 3D meshes, but also provides built-in rendering options for different surface materials. The shape visualization is split into four equally sized windows, three 2D and one 3D window.

The 3D view is the simplest view of the four. It displays the median shape of the selected elements. The shapes themselves are requested from the server, downloaded, and displayed. A particularity of three.js is that the use of multiple views, four in our case, also requires all elements to be replicated, so that each view has its own separate resources. This leads to increased management requirements. Changes to the shapes, for example, in the case of an exploded view, require a correct propagation of updates. To achieve this, we created, similar to the 2D visualizations, a simple render object wrapper where all global properties are stored. It has handles to the copied versions in each view, to update them accordingly. Using the same selection as in the cohort overview, the user can define which organs are visible.

³<https://threejs.org>

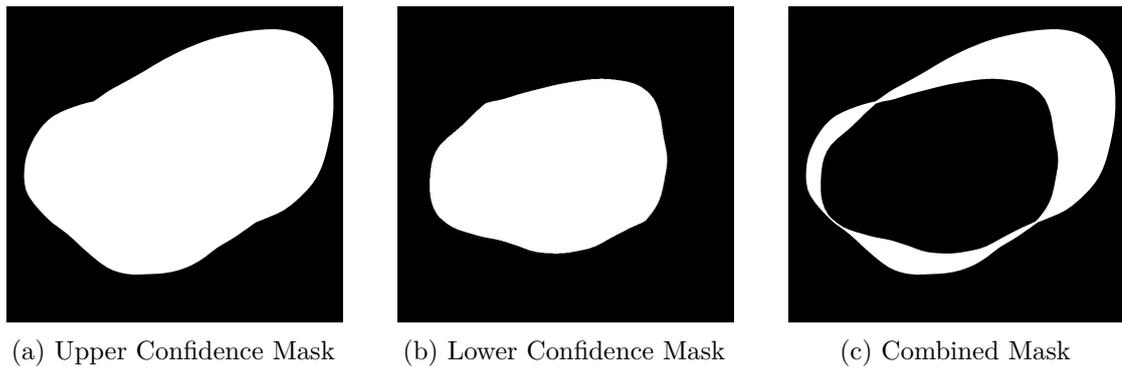


Figure 5.4: Masks created by the stencil buffer, to render the confidence interval, median and position glyph.

As described in Section 4.3.4, we use a slice-based 2D visualization that shows for a selected organ part how large the shape variations are based on the median shape and a plotted variability lobe. This kind of visualization was previously only created on the basis of 2D streamline data, where the dimension of the data coincides with the dimension of the visualization. However, as we receive 3D meshes, a transformation into the respective 2D space had to be found. Based on the assumption that the organ shapes involved are always watertight, i.e. they have no holes, we developed a simple approach that can render 2D variability plots from 3D meshes. Starting with the confidence interval, the goal is to render the region enclosed between the upper and lower confidence shape, for a selected plane position. We first cut open the shapes by using the slice plane as a clip plane. Afterward, we view the shapes parallel to the plane and start with rendering the upper confidence shape into the stencil buffer. Hereby we define that back faces increase the buffer and front faces decrease it. The idea behind this is to create a mask for regions that are not to be colored for the lobe. Normally a closed mesh would have as many front-facing as back-facing parts at any pixel. Because we cut parts of the mesh away, we expect more backfaces at the regions that represent the hole inside the confidence lobe. While this masks out the outer part, we also need to mask out parts inside the lobe. The process is therefore repeated for the lower confidence shape. In this case, the backfaces decrease the buffer and the front faces increase it. For all these calculations, the buffer must have wrapping enabled to prevent numerical caps at zero, which would introduce errors into the result. After the calculation, we have the finished lobe mask and can draw it to the screen, by rendering a screen aligned quad with the correct color and transparency. The complete process with the involved masks can be seen in Figure 5.4. By using blending, we can repeat this process for all drawn elements. The median is drawn with the same procedure. Allowing the users to directly click on the lobes in the 2D view, they can select them for closer inspection. To do this we render the lobes into an ID buffer and find out the selected element by reading the pixel value at the click position. Finally, to improve the overall performance of our tool, the 2D views are only updated if the slice planes are changed or new elements are added. The final

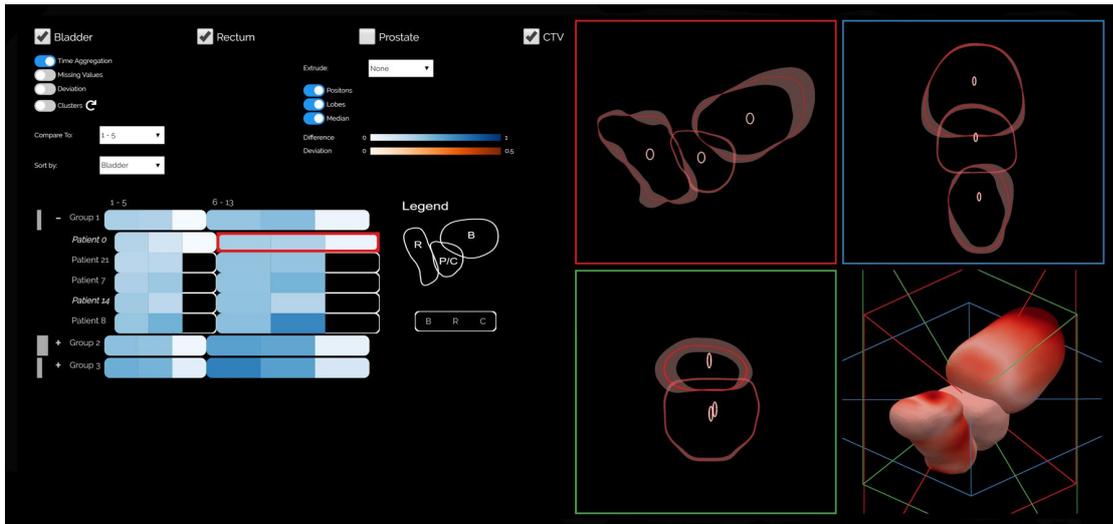


Figure 5.5: Overview of our complete application. One part of the cohort has been selected in the overview visualization on the left and the respective shape variations are shown in the anatomical shape visualization on the right.

complete user interface is shown in Figure 5.5, hereby all aforementioned visualizations are present.

Results & Evaluation

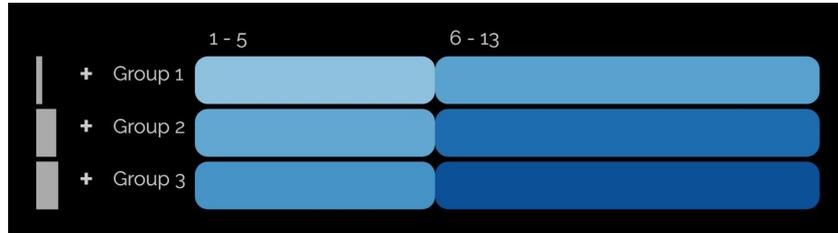
In this chapter, we will look at the finished tool and analyze how it is able to perform several different usage scenarios. These scenarios will be performed on single and multiple patients and organs. We will evaluate how well the proposed tasks **T1** and **T2** are fulfilled and if the asked research questions **Q1-Q7** are answered. Together with two domain experts, we looked at different usage scenarios. We recorded their feedback in an informal evaluation, to receive an indication of the strengths and weaknesses as well as possible future improvements of our approach.

6.1 Usage Scenarios

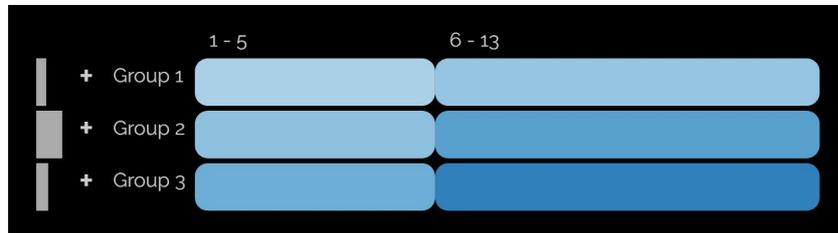
In these scenarios, we want to discuss possible use cases that can occur when analyzing the shape variability of a single or multiple organs, both for the whole cohort in general but also on a patient basis. We will present the ways our tool supports these cases and also which findings were made.

6.1.1 Cohort Exploration

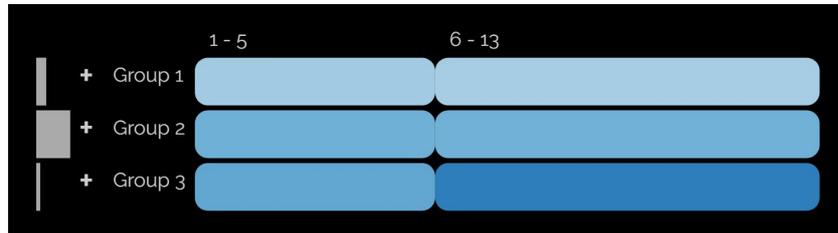
Initially, we start off with a bladder variability-based grouping. We see when looking at Figure 6.1b that Group 2 contains the most elements, based on the size of the corresponding histogram bar. This means that most bladders tend to have medium-sized variations. If instead of comparing each shape to the mean of the first five treatment days, we instead compare it to the first timestep only, we see in Figure 6.1a that the actual bladder shapes are way more different, as the second cell in each row is considerably darker than the first one. This is important because normally only the first timestep is used for treatment planning. When the differences to the average shape of all timesteps in Figure 6.1c is used, we see that the overall variability is at its lowest. One way of interpreting this would be that by performing the planning based on the several timesteps instead of only the first one could result in more precise models of the bladder shape.



(a) Bladder shape variations when compared to timestep 1



(b) Bladder shape variations when compared to the average shape of timesteps 1 - 5



(c) Bladder shape variations when compared to the average shape of timesteps 1 - 13

Figure 6.1: Comparison of the shape differences (low  high) between groups of patients. For each patient the average shape estimated based on the respective timesteps. The difference between the average shape of the patient and its remaining timesteps is estimated and aggregated for the groups. It can be seen that the more timesteps are considered for the average shape, the less the difference is to the remaining ones, i.e. the more descriptive the average shape is for the individual patient.

6.1.2 Single Organ Shape Exploration

Instead of looking only at the average shape differences in Figure 6.2a, we can also explore the precise shape variations. For this, we select each of the three groups, for low, medium, and high variations and look at their respective shape contour plots, as seen in Figure 6.2b. Looking at Figure 6.2c, we see that in general, all groups have a similar shape, shown by the relatively similar shapes of the medians. This can be due to the fact that the variability of the bladder is not connected to the specific shape types. The groups were created for patients with low, medium and high shape variations. We see that the amount of variability defined by the group can also be seen in the 2D views. For example, the group of low shape variability has small local shape variations shown by the extent of the lobes. The positional variations also correspond to the respective groups. The higher the general shape variability, the larger the positional variations are, leading to larger ellipsoids, as seen in Figure 6.2b. The ellipsoids are also the largest in the sagittal 2D view, indicating that most movement happens in this plane, which corresponds to the finding of other works in this field [MSD03].

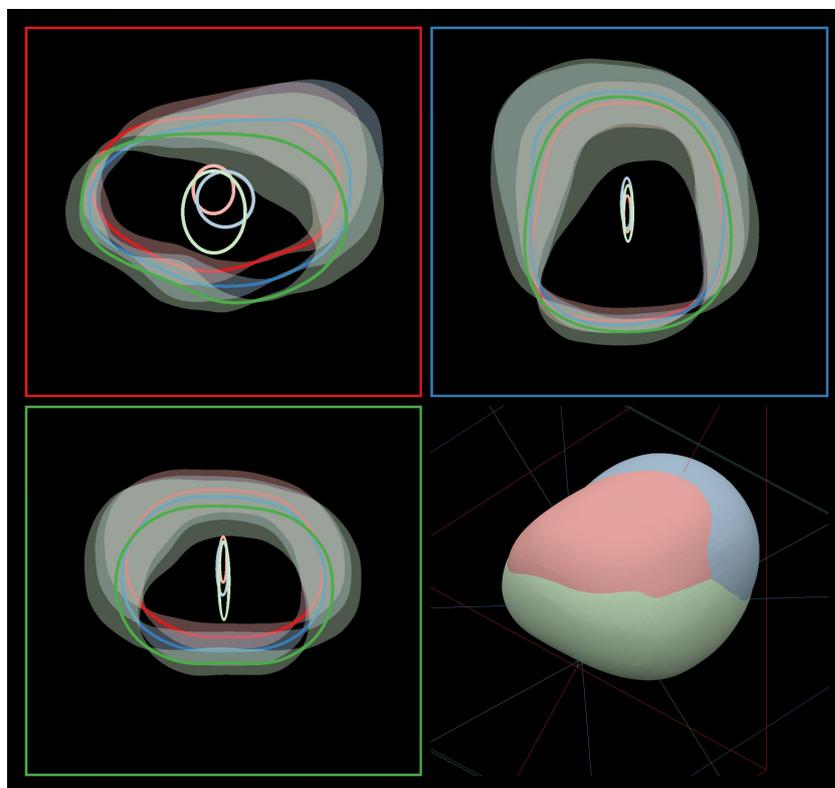
6.1.3 Multiple Organ Shape Exploration

The previous tasks can be repeated for any of the organs present in the data. To explore the connections between organs, the user can select multiple organs to be displayed simultaneously, like for example bladder, prostate, and rectum. A direct effect of this is that the cohort overview encodes the average values of the three organs side-by-side, see Figure 6.3a. We see that due to the sorting based on the amount of bladder shape change, the bladder glyphs are arranged in a gradient from Group 1 to Group 3. Looking at the rectum we see that the colors are roughly similar for each group. There seems to be little to no correlation between the change in bladder and rectum shape, which does appear logical due to the underlying biological functions of the two organs. The same holds true for the inverse, if sorting is based on the rectum shape changes. More interestingly, we see that the prostate does not undergo any large shape variations, as it has a near-white color for all groups. This is plausible since prostate and CTV delineations are often used as the target area for the radiation treatment. They are therefore not adapted in shape usually and are only moved in position, even if their physical counterparts change.

To see at a glance, how descriptive these group values are, we look at the deviation and missing value proportion in Figure 6.3b. The coloring shows that the values for the bladders are rather similar in each group, the ones of the rectum are strongly varying, and finally, the prostate seems to have similarly low values for all patients in the group. An interpretation of the deviation values, encoded in the orange coloring, could go along the following path: The grouped rectum values seem to vary largely in value, as they have rather dark orange cells. This indicates that there is no correlation to the bladder variations, based on which they are sorted. The black elements of the glyph also reveal that around a third up to half of all patients are missing a segmented prostate, which can also be directly seen if the associated patients are shown. By sorting the patients based on the prostate, we see in Figure 6.4, that the first group containing all patients with

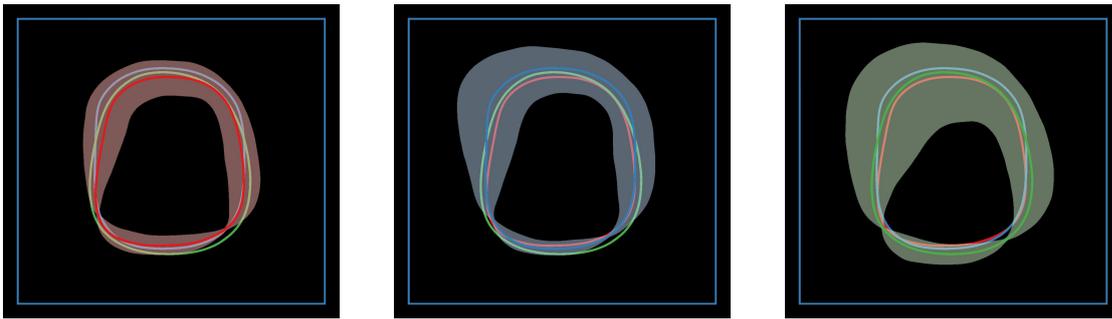


(a) Cohort Visualization of the shape differences (low  high) with the selected groups



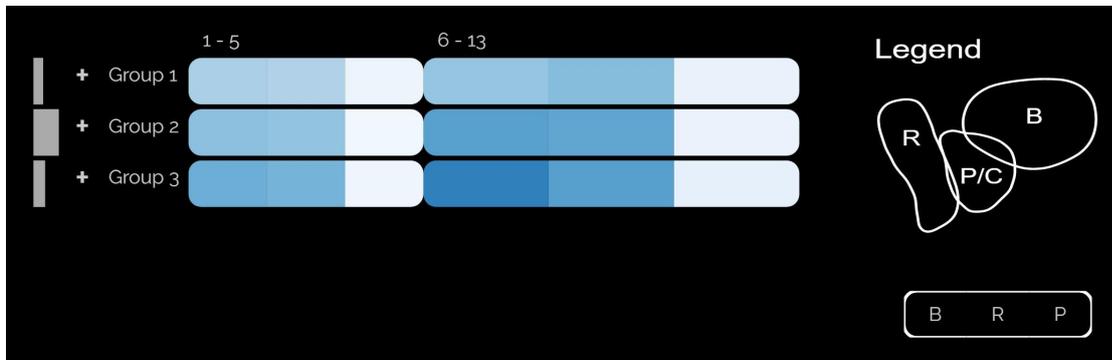
(b) Shape Visualization including position variation glyphs

Figure 6.2: Analysis of the local shape variabilities based on three groups (Median and Lobe Color:  Group 1,  Group 2,  Group 3) derived from the amount of bladder shape variations.

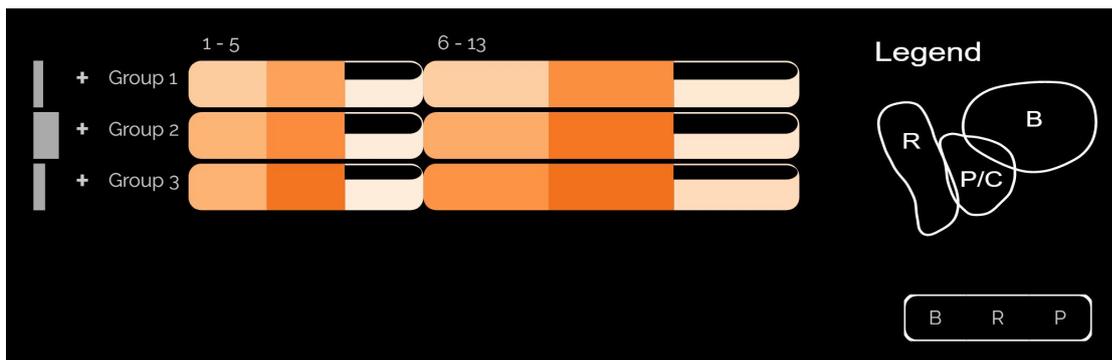


(c) Local Shape Variations - Created by selection

Figure 6.2: Analysis of the local shape variabilities based on three groups (Median and Lobe Color: ■ Group 1, ■ Group 2, ■ Group 3) derived from the amount of bladder shape variations (*continued*).



(a) Shape Differences (low high)



(b) Group Disparity (low high) and missing values

Figure 6.3: Cohort visualization of groups with increasing bladder shape variability.



Figure 6.4: Cohort visualization of groups with increasing prostate shape variability (low  high). The first group contains all patients with no segmented prostate. For all groups the prostate shape variations are extremely small.

missing prostate, has around half of all patients, with its histogram bar being roughly the size of all others combined. The interpretation of the deviation values, encoded in the orange coloring of the first figure, could go along the following path: The grouped rectum values seem to vary largely in value, as they have rather dark orange cells, indicating that there is no correlation to the bladder variations. We can assume that although most groups have missing shape values for the prostate, the patients with the shape are similar in value and also have similarly low shape variations. Both of these facts are indicated by the nearly white cells, shown in Figure 6.3b and Figure 6.4.

The next step would be to inspect the precise shape variations, to verify the made assumptions. Similarly to before we can do this by selecting a single patient and aggregate all its timesteps. Looking at the resulting contour variability plots in Figure 6.5, we see what we came to expect. Apart from the usual bladder variations described before, the prostate does not undergo any large shape changes during the treatment. This can be seen both by the consistently bright coloring of the prostate in the 3D view, but also in the small lobe around it in the 2D views. The rectum varies mostly above and below the prostate, extending towards the bladder based on its filling. This is indicated by the lobes further protruding from the median in these regions. We are also able to see that there exist slight overlaps between the prostate shape and the bladder, shown in the bright lobe color of regions where structures intersect. These may result from the fact that the prostate shape not only includes its biological counterpart but also a safety margin to ensure that all tumor cells are radiated. It is also notable that all organs seem to undergo the same positional changes, indicated by the similarly shaped ellipsoids. The prostate seems to move slightly less, as its ellipse is smaller than the others.

6.1.4 Shape Type Identification

Instead of looking at groups defined by properties like variation and toxicity, we can apply machine learning to investigate possible bladder shape types. In the initial grouping based

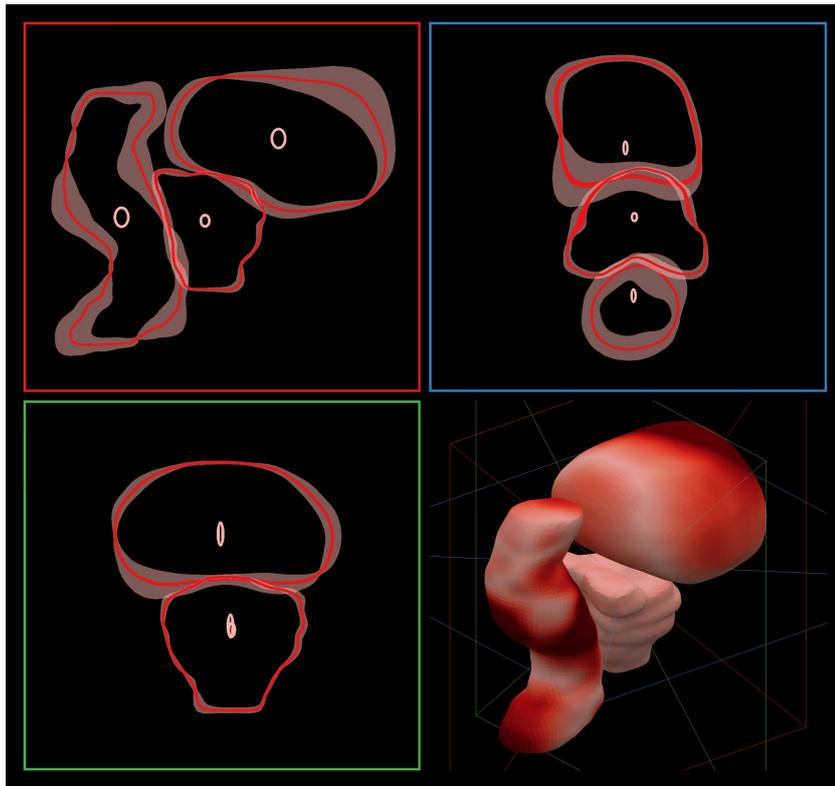
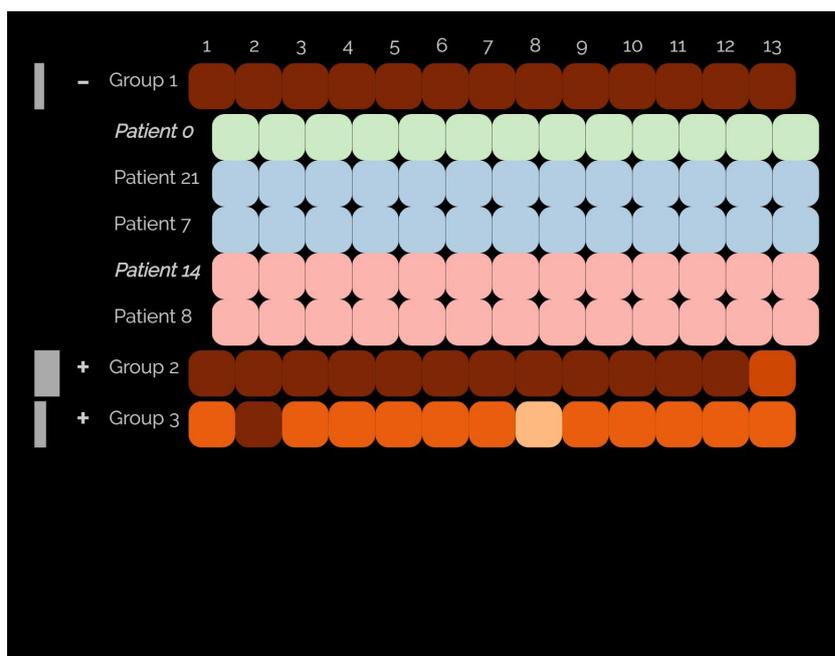


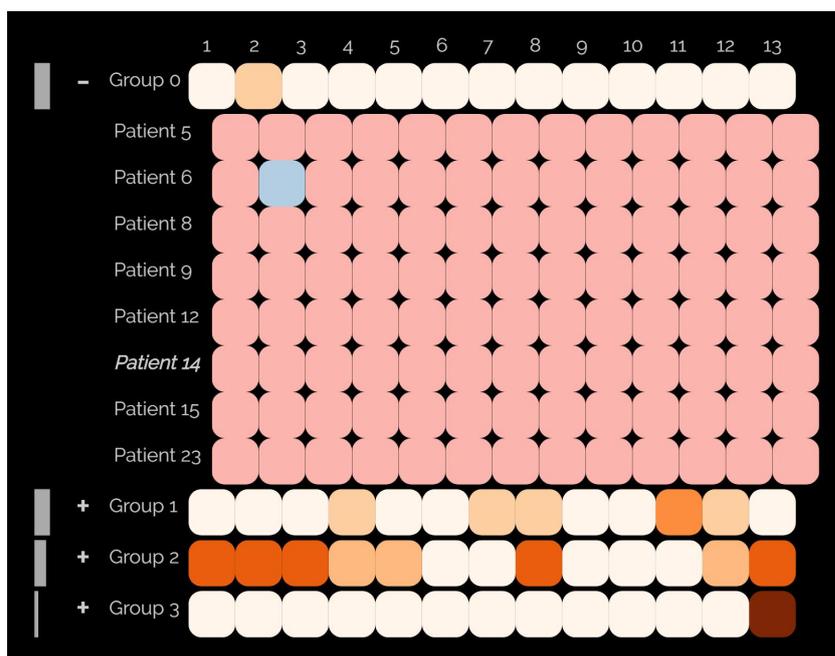
Figure 6.5: Shape Visualization of multiple organs for a single patient across time. The coloring in the 3D view encodes the amount of shape variation (low  high).

on the bladder shape variations in Figure 6.6a we see high deviations in the calculated bladder shape clusters. This is indicated by the dark orange cell colors. We can assume that the groups are mostly composed of mixed shape types and that the clustering does not reflect the amount of shape variation. Grouping the patients based on their most prominent shape type leads to more homogeneous groups with less deviation. This is visible in Figure 6.6b, where the cell colors are noticeably brighter than before. The first three groups contain most patients while the fourth one contains two. Interestingly, the third group as depicted in Figure 6.7 contains more than half of all patients with toxicity, indicated by the bold labels of the patients. This information is obtained by expanding the individual groups and inspecting the contained patients.

For a more detailed investigation, we select each group and inspect their median shapes, confidence lobes, and positions. In Figure 6.8 we see that the shapes produced by the clustering according to shape type appear more different than the ones produced by splitting the shapes based on their average shape variation, as seen before in Figure 6.2b. To better compare the bladder shapes, we can center them at a shared point and using an exploded view. In doing so we see that Group 3 and 4 are slightly bigger. Group 3, containing most patients with toxicity is rounder and protrudes further in the direction



(a) Grouping based on bladder shape differences



(b) Grouping based on bladder shape types

Figure 6.6: Deviation (low  high) of the aggregated timesteps for each group

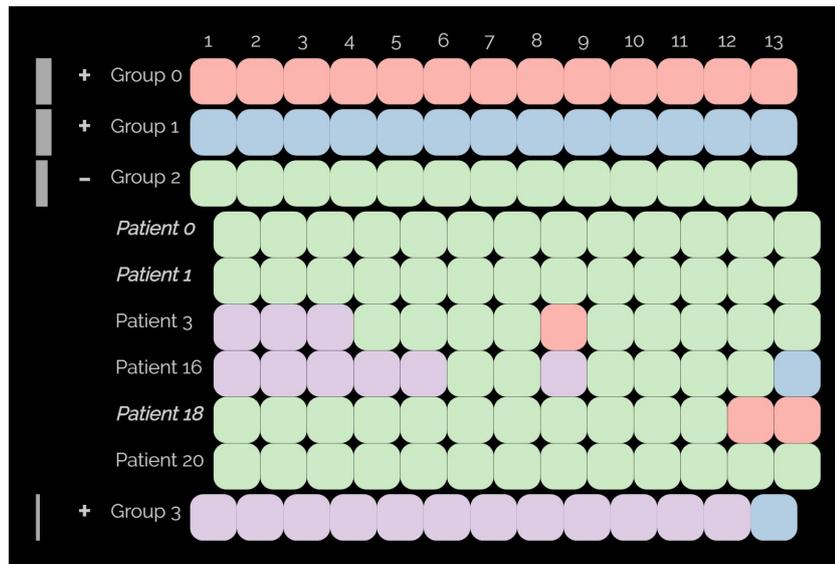


Figure 6.7: Cohort visualization of bladder shapes with their individual shape types (Group 1, Group 2, Group 3, Group 4), Group 3 has an increased number of patients with toxicity.

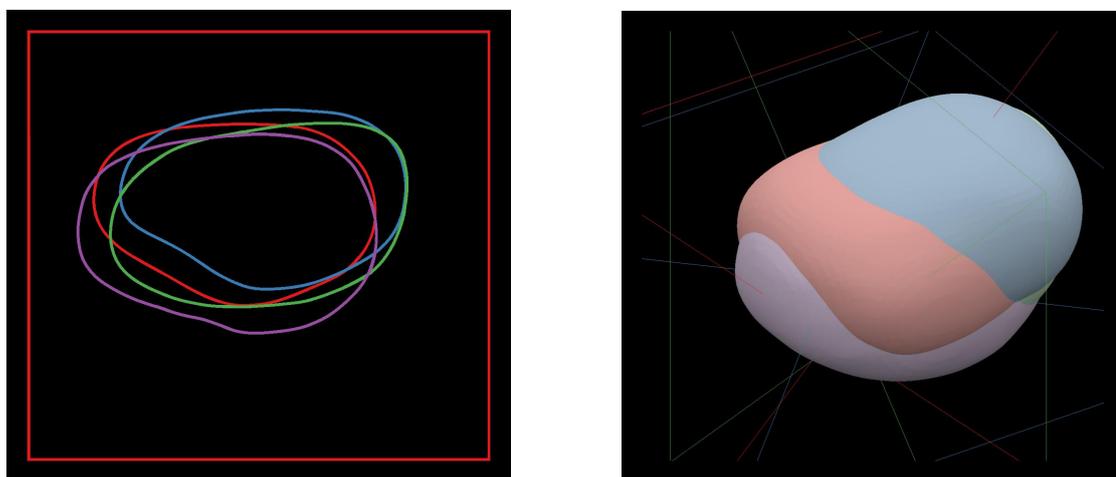
of the prostate. This is not only visible in the 2D views, but also in the superpositioned 3D view. Both Group 1 and 2 have a flatter interface towards the prostate.

Performing a more in-depth analysis of the shape variations of each group, we see in Figure 6.9 that all bladders tend to have the largest extent on the upper side. This makes sense as here the bladder has the fewest constraints in the body and can freely extend. At the opposite, the prostate is positioned. All bladder shapes have smaller lobes in this region, indicating that the bladder does not extend far into it. This is also anatomically plausible as growth in this direction is restricted due to the prostate and rectum.

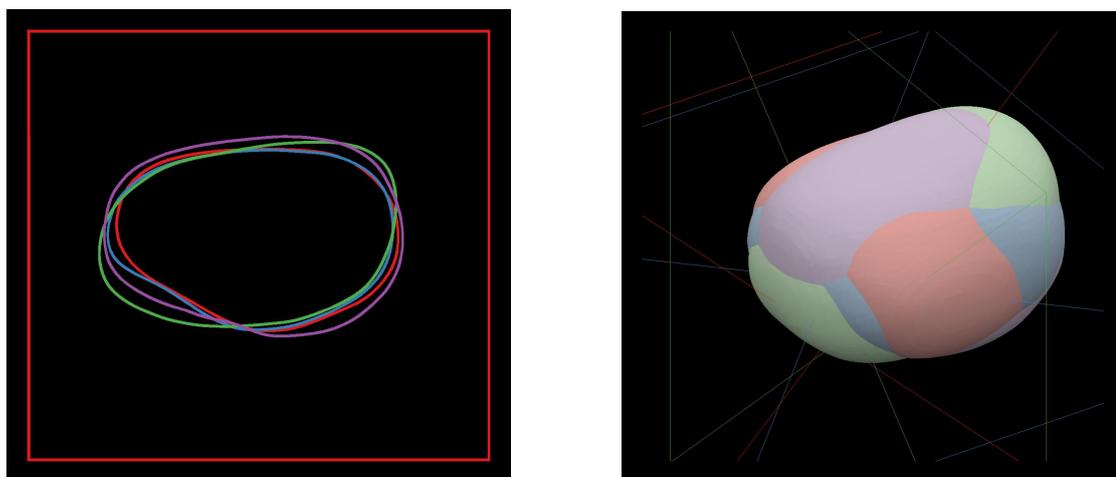
The position variations in Figure 6.10 are also different for each group, although all of the moves are predominantly in the sagittal plane. Its ellipsoids are larger than the ones in the other planes. For Group 2 and 3, no dominant movement direction is visible and the glyphs look like circles. The glyph of Group 3 is larger, indicating bigger movements in the bladder position. Both Group 1 and 4 move more in one direction than in the other one, leading to ellipsoidal glyphs. Recent works analyzing pelvic organ movement in radiotherapy, like the one of Pinkawa et al. [PAG⁺06], verify these findings.

6.1.5 Individual Patient Analysis

Up to now, we have mainly focused on the exploration of the complete cohort. However, we are also able to employ our tool on a per-patient basis. Still starting with a view of the whole cohort, as seen in Figure 6.11a, we can see that some patients seem to have higher deviation values than others. This indicates the presence of outliers, i.e. shapes that are



(a) Original position



(b) Centered position using the exploded view

Figure 6.8: Comparison of bladder shapes with different alignment strategies.

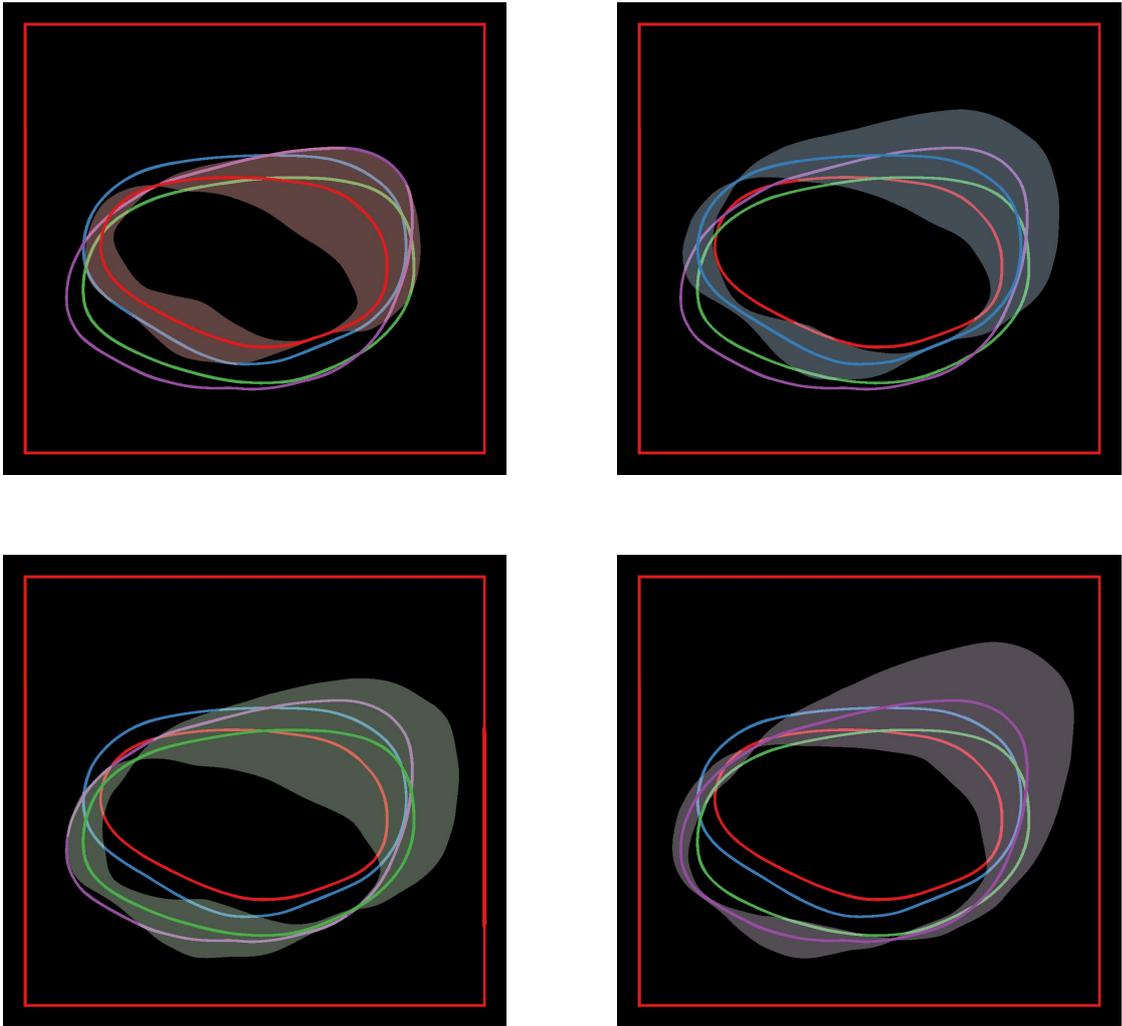
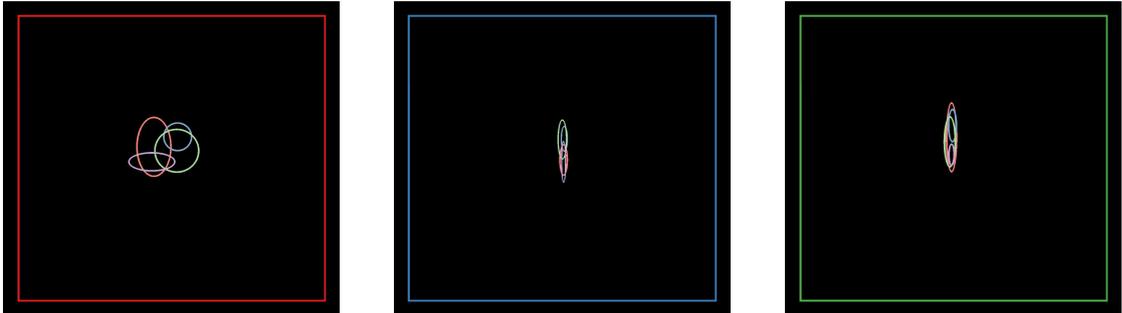


Figure 6.9: Local shape variations of the four different groups.



(a) Sagittal 2D View (b) Axial 2D View (c) Coronal 2D View

Figure 6.10: Position variations of the four different groups.

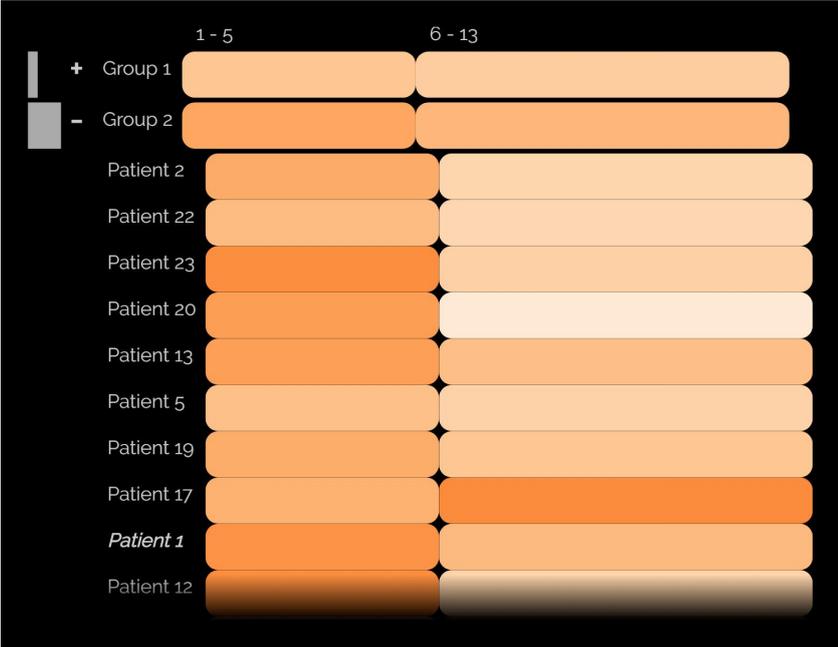
largely different from the rest. Visually, this can be detected by dark orange table cells, encoding high disparity values. Most patients tend to have more different shapes in the first five timesteps than the last eight ones, shown by the generally darker orange coloring of the first cells in the rows. Only Patient 17 stands out, whose second cell is noticeably darker than the first one. If we focus on Patient 17, we see upon closer inspection of the individual timesteps in Figure 6.11b, that Timestep 12 seems to be vastly different to the rest. This is indicated by the dark blue coloring. The most probable reason for this is the high disparity value for the patient.

We could either select all individual timesteps and plot them on top of each other, as seen in Figure 6.12a, or we could examine the aggregated median shape and its confidence interval by selecting the label of Patient 17 and the outlier, which is shown in Figure 6.12b. By inspecting the resulting plot we can directly see that the outlier seems to stem from the fact that the bladder is unusually small for the selected patient. This could be because the bladder had a different filling than at the other timesteps, e.g. empty instead of full. Upon further inspection, we see in Figure 6.13 that the bladder shape varies similarly to the previously described patterns. Its variability lobes tend to be larger on top than in the direction of the prostate. The axial 2D View also indicates that the bladder seems to grow more with a tilt to the left, a pattern that can also be seen in the works of Casares et al. [CMMH⁺17].

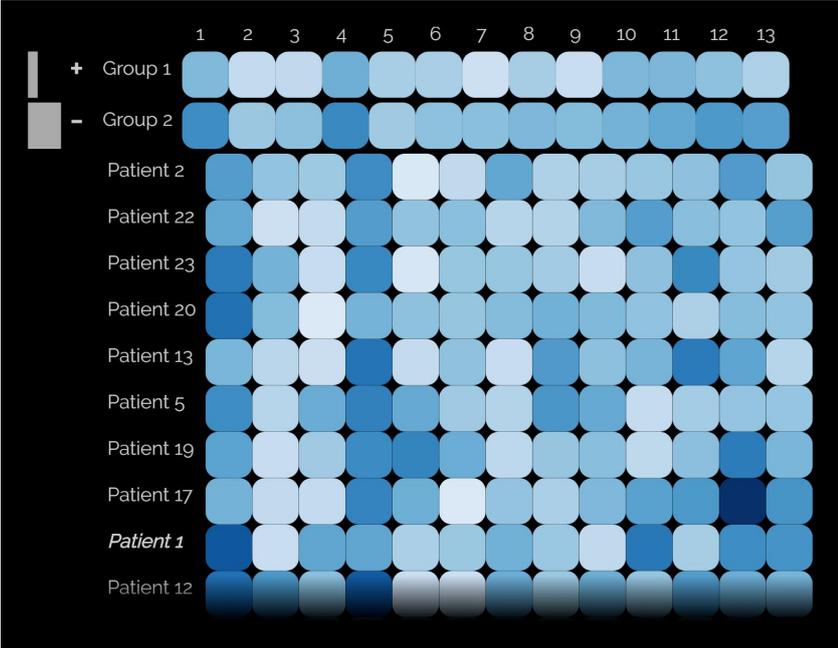
6.1.6 Toxicity Factor Analysis

To understand the effect of the bladder shape variations on toxicity, we can sort the elements based on this attribute, as seen in Figure 6.14. The resulting groups are of different sizes. More patients are in the group with no toxicity, which can be determined by comparing the histogram bars beside the group labels. Looking at the remaining elements in the group of patients with side-effects, we can detect several patterns. First see that there are both patients with high (1, 10, and 18) and low (0, 1) shape changes. The ones with low changes are identified by bright cells and the ones with high changes by the darker cells. Secondly, we also find patients whose average shape of the first five days is similar to the rest of the treatment (0, 1 and 14), but also those whose average shape is not (10 and 18), leading to higher variations. For this, we compare the color of the first cell in a row, representing the first treatment timesteps, with the second cell containing the following treatment timesteps. These patterns do not indicate a connection between bladder variability and induced toxicity, because they are also found across patients with no symptoms.

By selecting the groups of patients with and without toxicity, we can look for structural differences in the respective bladder shapes, which can be seen in Figure 6.15. In general, there seem to be no large differences in the median shapes themselves, although the one with toxicity seems to be slightly bigger. Both bladder and prostate shapes seem quite similar. One noticeable difference can be found when inspecting the position variations. While the bladder and rectum position variations look quite similar, the prostates with and without toxicity look vastly different. Looking at the sagittal 2D View, we see that



(a) Aggregated disparity visualization (low high)



(b) Individual shape differences (low high)

Figure 6.11: Outlier detection for individual patients.

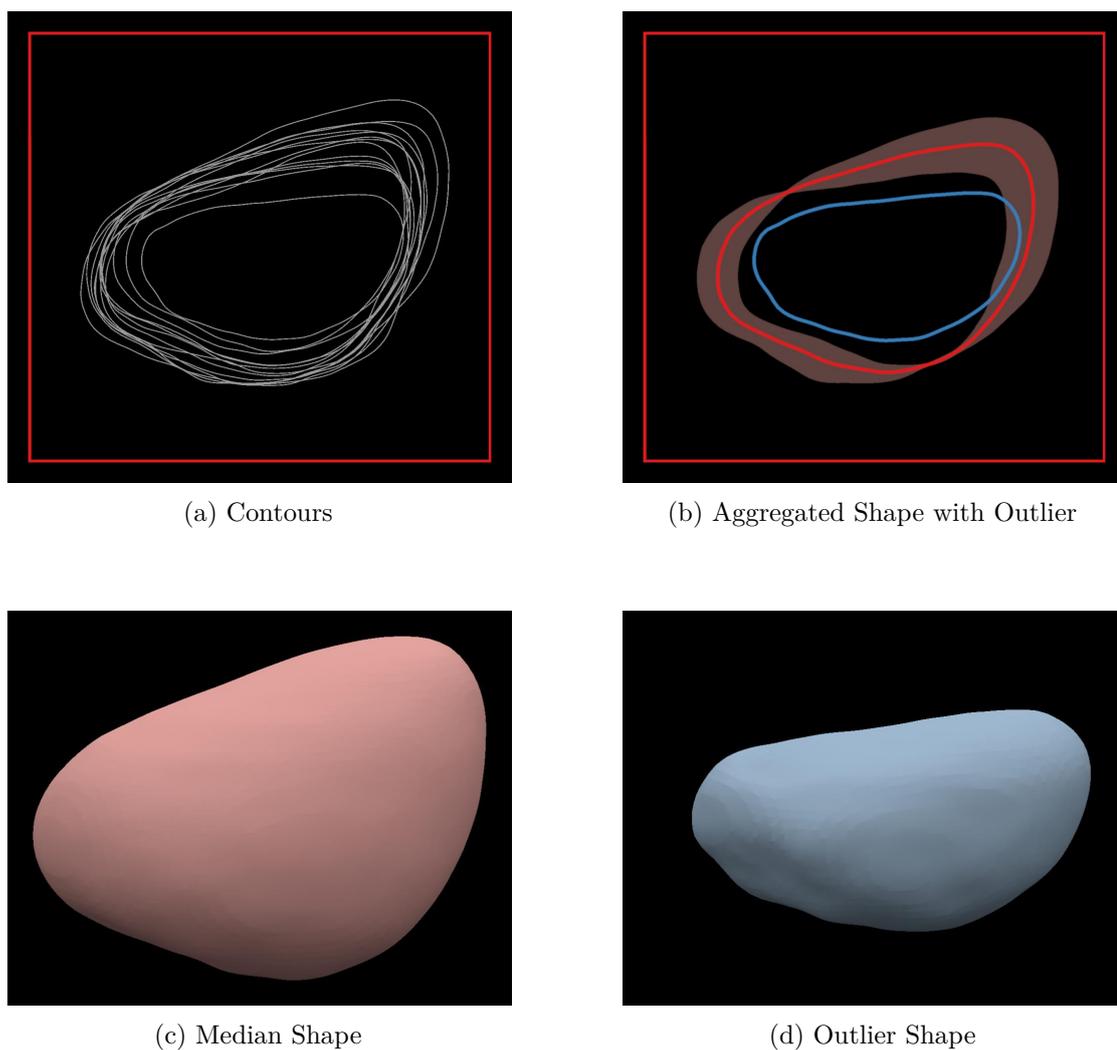


Figure 6.12: Bladder shape visualizations of an individual patient with a distinct outlier.

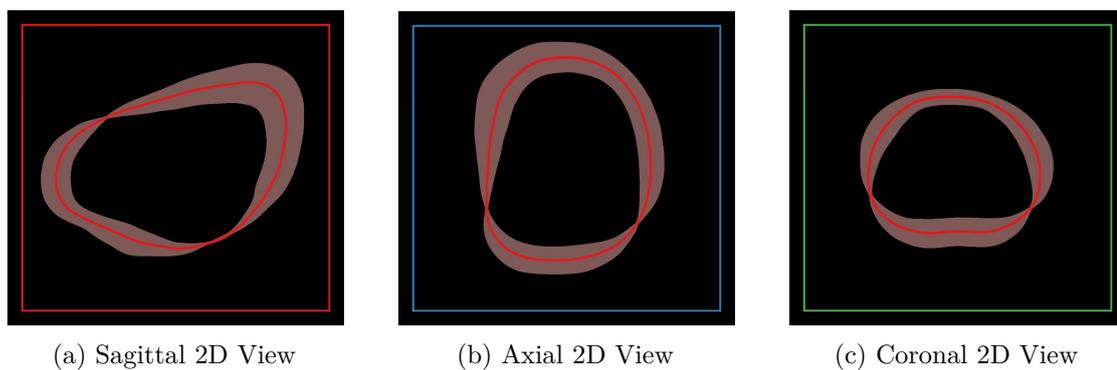


Figure 6.13: Local shape variations

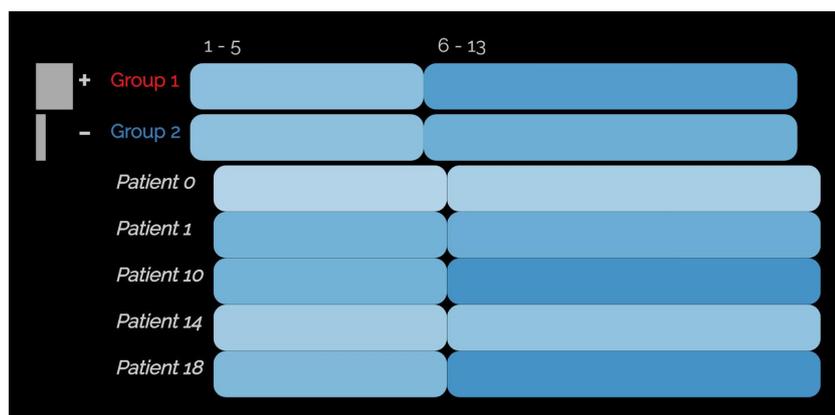


Figure 6.14: Bladder shape differences (low  high ) for patients  with and  without toxicity.

the prostate with toxicity seems to move more horizontally than the one without, as it extends much more from left to right. In this regard, the prostate from the group with toxicity has a larger overlap with the rectum than the other, as the intersecting region of the two shapes is bigger. A large overlap between the rectum and the treated region can lead to serious side-effects. However, it should be noted that the results for the group with toxicity are derived from only a small number of patients, and may, therefore, be susceptible to outliers.

6.2 Evaluation

We discussed our tool with two experts in an informal evaluation. The participants knew the data already from previous projects and are familiar with the exploration of patient cohorts. Initially, we presented them the tool, explained the underlying functionality and processes, and showed them the aforementioned usage scenarios. Throughout the whole conversation, they were able to ask questions and make suggestions concerning the interaction. Following this presentation, we asked them to comment on the shown functionality and give their thoughts about the strengths and weaknesses of our work. In the end, we also asked them for future improvements to our work and especially which further functionalities they would consider as necessary.

The overall perception of our work was rather positive. One of the experts described it as *"a promising and useful decision-making tool for radiation oncologists"*. Most notably the shape aggregation represents one of the more important features for them as it allows for a flexible and systematic exploration of their data in a new way. *"The tool offers a way of identifying the setup uncertainty of the entire treatment"*, which they have not been able to show precisely up to now. Although this use of our tool was not intentional, it could provide interesting future research aspects. Generally, they considered the 2D views to provide them with the most information. In the future they hope to be able

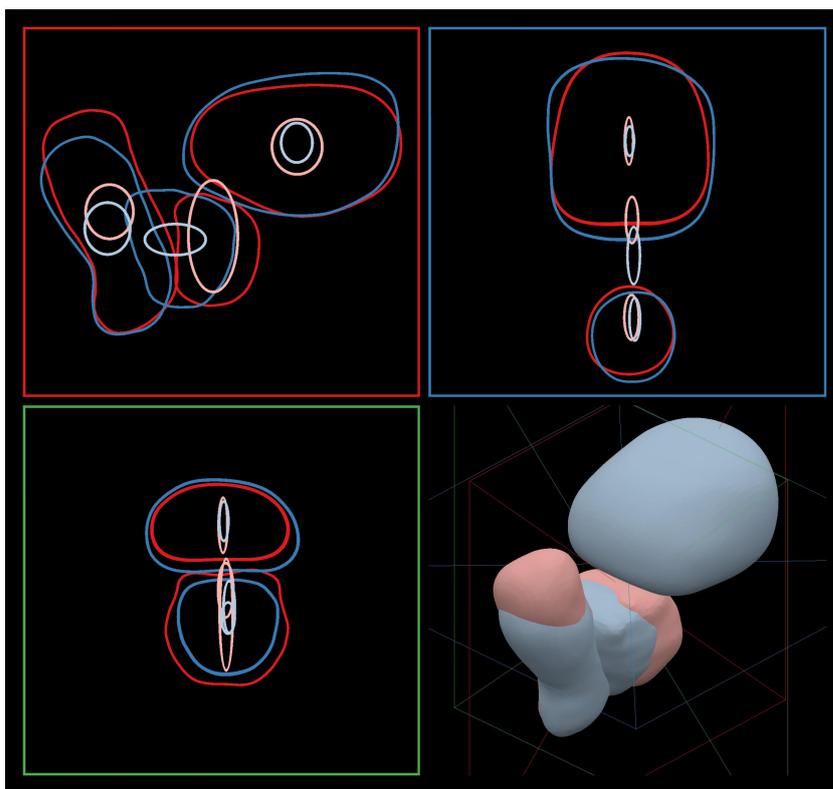


Figure 6.15: Local shape and position variations for patients ■ with and ■ without toxicity.

to explore their data into more detail, using our tool to search for new characteristics inside their data. The gained information can lead to improvements during the treatment planning, giving them *"indications of patients that will fail or that may develop toxicity at the beginning of the treatment"*. One way they imagined this could be achieved would be the derivation of thresholds for shape variations and postponing the treatment if the shape is too different from the initial one. Other future extensions they would consider helpful, include the addition of a probing tool along the median to show the extent of the variations in mm. A more in-depth evaluation of the capabilities should also be done according to the experts, where the results of our system are compared to other work like the Bladder Runner [RCMA⁺18].

Conclusions & Future Work

This chapter concludes the presented work and will mainly touch upon two topics: Firstly we will discuss how well we were able to answer the initial research questions and proposed tasks, after which we will look at possible areas of improvement that can build the foundation of future work.

7.1 Summary

The cornerstone of our work is summarized by a simple question: *How can we analyze and visualize the anatomical changes of organs to support medical experts in their research and treatment planning?* We split that question into seven individual parts that we all aimed at answering, as discussed in Section 1.2. They served as a basis for the analysis of related works in Chapter 3, where we analyzed recent scientific advances and concluded that none of them can satisfy all proposed question parts. In this section, we will go through each question and discuss how well our proposed solution performs and how it compares to previous works.

(Q1) Which patients or organs experience high amounts of variability inside the cohort?

The first three questions directly deal with the analysis on a global level, where the whole cohort should be analyzed. Several works estimated the amount of shape difference. Many of them did this for various non-medical datasets [BBS⁺18, SPA⁺14] or only analyzed a single organ [RCMA⁺18]. In this work, we estimate the amount of shape variation on a per-patient basis, by measuring the distance of the shape at each timestep to a defined center and visualize the scalar results in a tabular view. This allows users to identify the general amount of shape variation and outliers, where the shape looks largely different.

(Q2) What are common types of shapes and variations inside the cohort? Other works like the one of Reiter et al. [RBGR18] or Raidou et al. [RCMA⁺18]. The former work relied on shape descriptors, which are not suitable for our case, as they suffer from shape

specific weaknesses. Additionally, they are not able to synthesize intermediate shapes. The latter work by Raidou et al. [RCMA⁺18] only focused on one organ, the bladder. By clustering organs inside their respective shape spaces, we are able to assign similar shapes to the same group, for multiple different organs. Visualizing the assigned groups in combination with sorting and grouping of similar patients allows users to inspect prominent shape types.

(Q3) Which parts of the cohort are missing? As not all patients have all the organs segmented, users need to put the quality of the aggregated shape values into context. This problem is rarely tackled in other works. Blumenschein et al. [BBS⁺18] showed how this information can be provided for the analysis of non-medical datasets, which does not directly translate to our usage scenarios. Our work provides the possibility to highlight how many elements inside an aggregated group are missing, using a glyph-overlay. Previous encodings were often too hard to interpret and therefore we designed a representation that offers a more precise estimation of the proportion of missing data.

The next four research questions mainly focus on the detailed inspection of organ shapes, two of which can be answered together. *(Q4) How do organs vary in shape and position between patients or across treatment instances? and (Q5) Do certain parts of an organ experience especially high variations?* We start with the most basic search for general shape variations, they are shown by statistically analyzing the shape space and rendering a shape variability plot. This process was derived from the work of Ferstl et al. [FBW16, FKRW16]. Previous works analyzed position variations [CGL⁺10]. We presented, to the best of our knowledge, the first tool that integrates position variation information visually. We extended existing approaches for statistical shape analysis to work for pelvic organs and not only streamline data and introduced a method to derive shape variability plots from 3D meshes instead of 2D regions. The presented shape variability plot is not only able to aggregate shapes and positions of multiple patients, but also highlights specifically which parts undergo large changes. As for the position variations our use of ellipsoidal glyphs is able to show how much the centers of the individual organs vary.

(Q6) Are there overlaps between different organs inside the cohort? Similarly to the missing data visualization, this problem is not often approached in recent works. Tory et al. [TMA01] provided a tool closely related to our use case. Their analysis mainly focused on the aggregation of lesions, but their approach struggles the more elements are added, which does not concern us due to our statistical analysis. The amount of overlap between different organs of the same patient or the same organ of different patients or groups is encoded implicitly in our work. Whether the median shapes intersect or the semi-transparent variability lobes do, it can be assumed that the involved structures intersect each other.

(Q7) How do certain organs differ in shape between patients? Most other works either analyzed only one kind of shape [FBW16, RCMA⁺18] or only one way to analyze groups of shapes [RBGR18, BBP10]. Our method offers users the choice to either aggregate or superposition shape variability plots. This allows users to directly compare different

groups in a shared space without cluttering the view with too much information. We also provide the option to view an organ either in context with all other shapes, or to extrude all organs of a kind and view them aligned to each other in an exploded view.

We presented a visual analysis application for the exploration of segmented pelvic organs in multiple patients, across the whole radiation therapy treatment procedure. In this work, we focused on the global exploration and analysis of pelvic organ shape variability in an abstracted tabular view and on the local exploration and analysis in a combined 2D/3D anatomical view. We showcased the functionality of our tool in different usage scenarios and detailed how it is able to answer the previously proposed research questions, these results were also verified in a small evaluation conducted with two domain experts.

7.2 Limitations & Future Work

Looking at the presented results and the evaluation already gives a good impression of the capabilities of our proposed tool. For a comprehensive analysis, we also need to discuss where the limits of our work lie. Understanding them allows us to put the work into context with other solutions from the field and can also serve as an inspirational basis for future improvements and continued work.

Scalability was one of our three primary requirements that needed to be fulfilled for our final tool. We aimed to support the fast exploration of large cohorts of organ shape data from multiple patients and timesteps, while simultaneously allowing for easy addition of new patients and interactive exploration. In the end, we were only able to partially achieve this requirement. Although we were able to create a visualization that can analyze and aggregate groups of patients and organs, we are still facing two big challenges: First, the complex data format requires preprocessing and manual assignment of the shapes to be extracted. The possibility to add new patients to the dataset exists. However, this takes several minutes of computation and a separate tool is needed, before the data could be integrated into the main application. Secondly, the general performance of our tool during the exploration has some room for improvement. Namely, the time between the selection of a shape in the overview visualization and the display of its detailed shape variability plots needs to be reduced. Figure 7.1 depicts the necessary loading time. As can be seen from this plot the most time is consumed by the download of the 3D meshes from the server. This is since JavaScript works primarily single-threaded and is only able to process downloads sequentially. We implemented several improvements, like the interleaving of download requests to start as soon as the last one finishes, or the use of the `.stl` format, which has a memory-saving binary representation. To ensure an interactive exploration more improvements are necessary. One possible option could be to use web workers, to perform the resource loading in parallel in the background, which would require more complex resource management. A similar implementation already exists in `THREE.js` for `.obj` files.

Concerning the required usability, there is also room for improvements, which should be implemented mainly by closer cooperation with future users in the field of medicine.

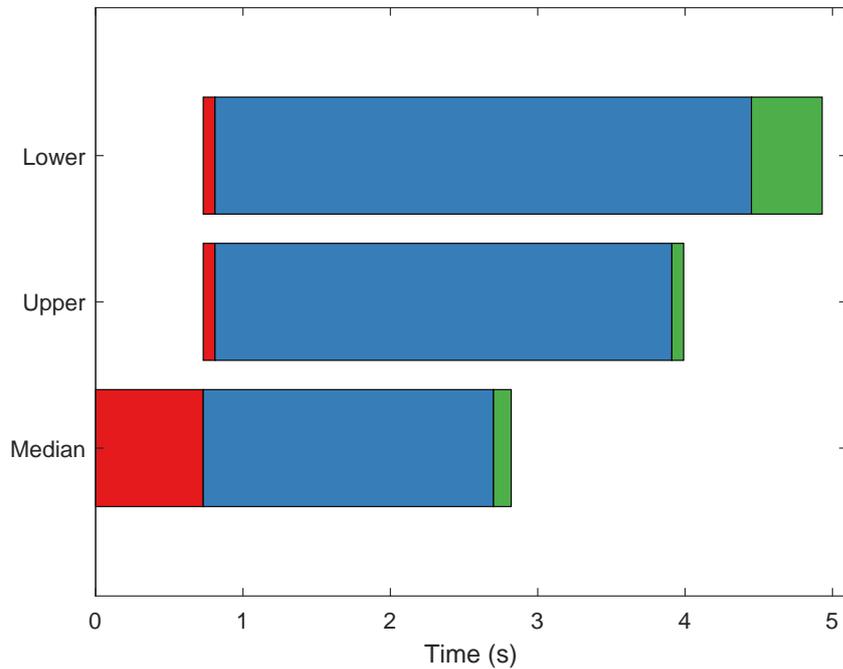


Figure 7.1: ■ Aggregation, ■ Download, ■ Processing Loading time for a shape variability plot of a bladder for an aggregated group of patients. The total time is split into the activities: shape aggregation and statistical analysis on the server, download from the server to the client, and client-side processing to create the displayed mesh.

Hereby, we look at two possible groups with different needs: researchers conducting cohort studies and doctors performing treatment planning.

For the former group, an apparent limitation that needs future improvements is the currently rather restricted way of *selecting and grouping elements*, both in the cohort view and the shape variability plot. Currently, patients are grouped, based on their average shape differences, into groups of low, middle and high values. Further improvements to this process could include allowing the user to manually add and remove patients from groups, for example, using drag-and-drop to provide more flexibility in the analysis. Another area where more flexibility could be needed is the selection of elements that are aggregated into a shape variability plot. For now, users can only select the whole patient or group, timesteps 1-5, 6-13, or single timesteps. Future versions should allow users to more precisely select the elements they are interested in aggregating, without the current restrictions.

The latter group of doctors, who perform treatment planning for individual patients, might benefit more from a *simplified tool* that focuses more on describing the organ shape variations of individual patients. Hereby, additional guidance functionality could benefit users in their work, like the automated suggestion of patients with similar organ shapes to provide background knowledge about common shape variations during the planning

process [CGM⁺16].

Before the use of our tool in medical practice a complete evaluation of the chosen *registration method* is necessary. Our results have shown that our visualization is susceptible to changes in the alignment of the analyzed organ shapes. We, therefore, developed a registration method based on the information from previous works and the results visible in the original data. This approach is able to return the expected results for organs like the bladder and rectum. However, we still see small variations in the prostate, where we were not expecting them [CMMH⁺17]. Future work should focus on understanding these variations and their sources, to allow researchers to safely use the application in medical research.

During the analysis of our results, we often detected large differences in the *position variations* of organs. While we are currently encoding this information in the detailed view of the organ shapes, we are not providing this information in the overview visualization. Their inclusion should be focused in future extensions, as this information may provide further clues to the underlying organ movements in different patients and possible ties to treatment side-effects. Another measure that should be included in future extensions is the amount of overlap between organs. Similarly to the position variations, this is currently only visible in the detailed views. Taking a look at the work of Weissenbck et al. [WFG⁺19], we could use a similar comparison approach between the shape volumes of two organs to detect overlapping regions. Our overall goal should be to provide a close connection between the cohort visualization and the detailed shape visualizations, to allow for a quick exploration, while at the same time maintaining an overview of all important factors.

Finally, a rework of the current *shape type classification* approach may be necessary. The created shape space can model the local shape variations rather well and is also able to detect noticeable shape differences in individual patients. It is hard to tell if it is able to properly differentiate organs based on their general shape type and to what extent it is influenced by the volume and position of the organs. To understand the created organ shape spaces better, future works should analyze the effects of its principal components, i.e. what parts of the shape are influenced by them. Another possible option could be to use a descriptor-based approach for the shape type grouping and perform the detailed shape analysis with our presented methods. These approaches may be less influenced by the respective organ volumes. Nonetheless, they may struggle with varying shapes, e.g. the elongated rectum [RCMA⁺18, RBGR18]. Future works should, therefore, analyze what the optimal way would be to provide a good shape type grouping for a cohort of shapes.

All these future extension options show that there is still a long way to go, but the tool presented in this work can be seen as another stepping stone in the integration of visual analytics in the investigation of medical data. In this work we presented one of the first tools analyzing shape variability in patient cohorts for multiple organs. It is aimed at aiding clinical researchers working on the development of robust treatment

7. CONCLUSIONS & FUTURE WORK

planning strategies, and demonstrates the necessity of including organ shape variability into radiotherapy treatments.

List of Figures

1.1	Overview of our general approach.	5
2.1	Pelvic Organs of the Male Body (Image adapted from Cancer Research UK [UK19])	8
2.2	Example of a simulated radiation therapy treatment plan applied to a patient with a full and an empty bladder. The left part shows the spatial distribution of the radiation applied to the prostate, while the right part shows the dose-volume-histogram explaining how much each surrounding organ is radiated. It also shows that by having a different bladder filling than the one at the time of the treatment planning the empty bladder receives a higher radiation dose. (Image from <i>Nejad-Davarani et al.</i> [NDSM ⁺ 19])	11
2.3	Current radiation therapy workflow as described in the work of <i>Schlachter et al.</i> [SRM ⁺ 19]. While the current process includes a per treatment adaptation, in terms of position and alignment, future techniques aim at having an even preciser radiated area by re-evaluating a larger part of the workflow for each treatment [SSB ⁺ 05].	12
2.4	Overview of the volume definitions used in radiation therapy planning.	14
2.5	General structure of the cohort data used in this work. The segmentation data of 24 patients is used, with each of them having 13 recorded timesteps throughout their treatment. For each point in the cohort multiple different organs are segmented, including bladder, rectum and most importantly the radiated volume.	16
2.6	Examples of the contours present in the provided data	17
3.1	Overview of the general topics touched upon in this thesis and how they interact with each other. Starting with an overview visualization of the cohort, interesting shapes are selected and analyzed. Groups of shapes are then treated as an ensemble and need to be visualized jointly. Finally one has to be able to compare multiple different ensembles to each other.	20
3.2	Examples of the classic approaches presented (Images from Casares et al. [CMMH ⁺ 17]).	22
3.3	Examples of the descriptor-based shape variability approaches presented.	24
3.4	Examples of the statistical shape variability approaches presented.	27
3.4	Examples of the statistical shape variability approaches presented (<i>continued</i>).	28

3.5	Ensemble visualization workflow, adapted from Wang et al. [WHLS18]. The dotted lines show that the respective pipeline parts do not have to be present in all works.	28
3.6	Examples of the ensemble visualization approaches presented.	30
3.6	Examples of the ensemble visualization approaches presented (<i>continued</i>).	31
3.7	Comparison between the fundamental techniques of comparative visualization, showcasing their scalability from two elements to n elements [KCK17].	31
3.8	Examples of the comparative visualization approaches presented.	33
3.8	Examples of the comparative visualization approaches presented (<i>continued</i>).	34
3.9	Examples of the cohort analysis approaches presented.	37
3.9	Examples of the cohort analysis approaches presented (<i>continued</i>).	38
4.1	Three different representation methods of shape data, shown on the example of a segmented bladder.	41
4.2	 Rectum  Bladder  Prostate  CTV Comparison of two different shape spaces, one created directly from volumetric (Volume) data and one derived from shape descriptors (Descriptor). The former was created from a $128 \times 128 \times 128$ volume sampled $32 \times 32 \times 32$ times and the latter is based on 25 shape descriptors calculated from the volume coverage mask, these descriptors include volume and orientation.	45
4.3	Comparison of four different registration methods on a simple 2D example, with the goal being to align the two boxes as good to each other as possible and center them at the origin.	49
4.4	Comparison of four different registration methods on the example of two bladders from a single patient at two different points in time. Looking at 4.4d shows the benefits of our method as it neither overly centers the shapes as in 4.4b and does not change the initial volume as in 4.4c.	50
4.5	A complete iteration includes the following steps: The initial element 4.5a is shrunk to half its size, after which it is replicated four times and placed at the positions and rotations indicated in 4.5b. These initially separated elements are then connected leading to the curve in 4.5c. This element can then undergo the same process to generate the next level of the Hilbert curve [Sag12].	51
4.6	General curve patterns for a row-wise scanline curve and the third iteration of a Hilbert curve. Both of which sample the underlying space in an 8×8 pattern. The color encodes the distance to the nearest point on the curve.	52
4.7	The same density field is sampled and mapped from 2D to 1D, once using a row-major Scanline curve and once using the Hilbert curve. The resulting lines show that the locality preservation of the Hilbert curve leads to fewer jumps and to a closer placement of neighboring parts.	53

4.8 Basic structure of an Autoencoder network, consisting of the encoder, which transforms the input image down to a previously defined number of output values, and the decoder which takes said output values as input and tries to recreate the original image as good as possible. Each element consists of the same components: The weight matrix w , the bias b , and an activation function. 55

4.9 Two examples illustrating how PCA works and showing how data variance affects the results. As PCA determines the principal components based on the dimension encoding the most variance, the data in the two groups in the first example can be easily separated as the difference between them is larger than the one inside each group. This assumption can lead to the problem shown in the second example, where the elements inside the group experience far larger variations and PCA fails to separate the groups appropriately. . 56

4.10 An example illustrating how LDA works and how it does not suffer from the restrictions as PCA. Contrastingly to PCA, LDA does not solely rely on the variance present in the data but also has access to the correct group labels. Therefore it can solve cases, which PCA cannot, namely: **Intergroup Variance: Low Intragroup Variance: High** 57

4.11 By training the Autoencoder network in Figure 4.8 to analyze images of cross-sections of our organ shape cohort, we analyzed how these networks work in principle by looking at their weight matrices in their centermost layer. The two shown images indicate which regions are deemed important. We can see the combined shape of rectum and bladder in the left image and a rough indication of a prostate shape in the right image. 58

4.12 Two examples of how t-SNE (Iterations: 1000, Perplexity: 20) transforms an input set of random points. In the first case the algorithm is able to clearly separate the two groups   from each other, creating two clusters of points. The second example shows a similar result as it also indicates two groups in the data, although the initial data just contains random points from a single distribution , highlighting the possibility of false positive results. 60

4.13  Rectum  Bladder  Prostate  CTV Comparison of four different analysis methods applied to the volumetric vectors of all organs of all cohort elements, with the idea of showing how well these methods are able to detect similar organs. Both PCA and the Autoencoder network show similar results, where the organs are spread across the whole space and similar organs are close to each other. LDA and t-SNE are able to show more pronounced groups of organs, hereby it is important to know that LDA was provided with the organ labels initially. In most cases the outlier CTVs, resulting from extremely large treatment volumes filling the whole bowel, are clearly visible. 61

4.14 Scatter plot of all bladder shapes in a 2D shape space. The similarity of certain shapes is implicitly encoded by their respective proximity. 63

4.15	Scatter plot of all bladder shapes in a 2D shape space. The similarity of certain shapes is encoded in the distance to the mean shape in the center. The coloring  and rings represent the distance to this point, where dark blue denotes increasing distance.	64
4.16	Scatter plot of all bladder shapes in a 2D shape space. The similarity is encoded by the cluster each shape belongs to, indicated by the color. The four clusters were found by using k-means, the crosses indicate the respective cluster centers.	65
4.17	A scatter plot matrix encodes the respective shape difference values for all organs and the patient and timestep ids. Each cell takes two of these features and creates a 2D scatter plot. The idea behind this is to uncover relationships between certain features. In the diagonal, histograms of the individual features show their value distributions.	68
4.18	The parallel coordinate plot (PCP) has a separate vertical axis for each feature, including the shape difference values for all organs, as well as the patient and timestep ids. For each observation the respective points on the axes are connected using a line. Correlations are encoded by polyline patterns between neighboring axes.	69
4.19	In this star glyph plot each patient's organ variability is encoded in the length of the four lines protruding from the center of each star. The longer the lines are, the more the respective organ varies. Starting from the top and going clock-wise the encoded organs are bladder, rectum, prostate, and CTV.	70
4.20	In the tabular plots the individual shape difference values are either aggregated based on patient or timestep ids. The average values are encoded as the cell color (low  high). Correlations are seen in increasing or decreasing gradients.	71
4.21	Screenshots of the tabular layout used in our tool to represent the similarity of each organ shape to the mean shape of each individual patient. The average shape difference values are encoded by the cell color (low  high).	73
4.22	Examples of the three different techniques presented to explore the tabular data containing numerical elements.	75
4.23	Example illustrating why the use of the mean on categorical or ordinal data leads to false results. If, for example, the categorical elements are saved as numbers $A = 1$, $B = 2$ and $C = 3$, which they commonly are, using the mean can lead to false results where the aggregated value depicts something that is not present in the data. In this case the use of the mode, which returns the most common value, is much more indicative of the contained elements.	75
4.24	Screenshots of the sorted and grouped cohort data.	77
4.24	Screenshots of the sorted and grouped cohort data (<i>continued</i>).	78
4.25	Distribution plots highlighting the deviation of each cohort element to the mean shape of the patient.	79

4.26	Histograms showing the distribution of the shape difference values inside the cohort on a per-organ basis. The x-axis shows the value range and the y-axis the number of element that fall into each of the bins. The higher the bar the more elements have a certain value.	82
4.27	Violin plots showing the distribution of the shape difference values inside the cohort on a per-organ basis. The gray box in the center is a simplified box plot, including elements like the median, quantiles, and whiskers. The colored lobes around the box plot indicate the individual value distributions. . . .	83
4.28	Scatter plot showing the distribution of the shape difference values inside the cohort on a per-organ basis. The actual values along the y-axis are plotted together with small random noise across the x-axis for each organ. The latter is necessary to prevent the points from overlapping too much and has no physical meaning.	84
4.29	Screenshots of the sorted and grouped cohort data. The images show two different distribution histograms for the groups, created on the basis of the average distribution of the specified organ. The black bar represents patients with missing CTV, the details of this encoding are discusses in Section 4.2.4.	86
4.30	Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the size of the individual cells, with larger cells representing low standard deviation, i.e. high trustworthiness.	87
4.31	Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the texture of the individual cells, where cells with smaller dots represent low standard deviation, i.e. high trustworthiness.	88
4.32	Table showing only the standard deviation. Hereby a different colormap is used to separate it from the shape difference encoding (low standard deviation - high trustworthiness  high standard deviation - low trustworthiness).	89
4.33	Table showing both the average shape difference values, as well as standard deviation. The latter one is encoded as the blur of the individual cells, with sharper cells representing low standard deviation, i.e. high trustworthiness.	90
4.34	Direct comparison of each analyzed encoding for the standard deviation (minimal deviation: 0% and maximal deviation: 100%).	90
4.35	Screenshots of the encoded disparity (increasing standard deviation values ) for the aggregated values (increasing difference values ).	92
4.35	Screenshots of the encoded disparity (increasing standard deviation values ) for the aggregated values (increasing difference values  (<i>continued</i>)).	93
4.36	Direct comparison of each analyzed encoding for the missing amount of data (nothing is missing: 0% and everything is missing: 100%).	94
4.37	Screenshots of the encoded missing data for the aggregated values.	95

4.38	Examples of the distribution of the first principal component concerning the bladder shapes, once for the complete cohort and once for the first patient. We compare how descriptive the center measures  mean  median and  mode are for each of them. The median and mean are slightly apart for the skewed complete distribution. This effect vanishes if we focus on individual patients.	98
4.39	Examples of the distribution of the first principal component concerning the bladder shapes, once for the complete cohort and once for the first patient. We compare how descriptive the disparity measures  standard deviation and  interquartile range are for each of them.	98
4.40	Example for statistical shape analysis based on the used bladder shape data.	100
4.40	Example for statistical shape analysis based on the used bladder shape data (<i>continued</i>).	101
4.41	Screenshot of the final combined 2D and 3D representations. Each of the three 2D slices is based on the three medical planes (sagittal, axial, and coronal) and is assigned a specific color. The current slice position is shown by the colored rectangles in the 3D view.	102
4.42	Example of superpositioned bladder shapes. Each shape has a distinctive color and is drawn transparently.	104
4.43	Example of juxtapositioning, where the three bladder shapes are drawn next to each other.	105
4.44	Example of explicit encoding, all the shapes are aggregated to a median and the deviations are encoded in color (low variations  high variations).	106
4.45	Screenshots of the employed 3D overview visualizations.	107
4.46	Example of an stacked contour plot for three individual patient groups, each showing bladder, rectum, and prostate.	109
4.47	Example of an exploded view for three individual patient groups, each showing bladder, rectum and prostate. The bladder shapes are moved from their initial position and shown next to each other. A ghosted version of them is left behind.	110
4.48	Example of an aggregated contour plot showing bladder, rectum and prostate. The individual shapes are aggregated into a shape median and a confidence interval.	111
4.49	Screenshots of the 2D slice view. The shapes inside a group are drawn using contour variability plots, while multiple groups are stacked on top of each other. Selecting one group highlights their shape variations and hides the ones of the others.	112
4.50	Screenshots of the 2D slice view, where a certain organ shape is extruded for an exploded view. All shapes of a certain type are moved away from the rest to prevent overlaps and are aligned based on their center points. If a shape is selected, its shape gets highlighted and a guidance line is drawn to its original position for context.	112

4.51	Screenshots of the 2D slice view, with the added position change glyphs, indicating the average movement of the organ centers. It is apparent that the organs generally move mostly in the directions of the sagittal plane. . . .	113
5.1	Overview of the general system structure, with the four main components.	116
5.2	Comparison of the unsorted and sorted upper and lower confidence interval contours ( Median Contour  Lower Confidence Contour  Upper Confidence Contour).	118
5.3	Illustration of the user interface structure.	120
5.4	Masks created by the stencil buffer, to render the confidence interval, median and position glyph.	122
5.5	Overview of our complete application. One part of the cohort has been selected in the overview visualization on the left and the respective shape variations are shown in the anatomical shape visualization on the right. .	123
6.1	Comparison of the shape differences (low  high) between groups of patients. For each patient the average shape estimated based on the respective timesteps. The difference between the average shape of the patient and its remaining timesteps is estimated and aggregated for the groups. It can be seen that the more timesteps are considered for the average shape, the less the difference is to the remaining ones, i.e. the more descriptive the average shape is for the individual patient.	126
6.2	Analysis of the local shape variabilities based on three groups (Median and Lobe Color:  Group 1,  Group 2,  Group 3) derived from the amount of bladder shape variations.	128
6.2	Analysis of the local shape variabilities based on three groups (Median and Lobe Color:  Group 1,  Group 2,  Group 3) derived from the amount of bladder shape variations (<i>continued</i>).	129
6.3	Cohort visualization of groups with increasing bladder shape variability. .	129
6.4	Cohort visualization of groups with increasing prostate shape variability (low  high). The first group contains all patients with no segmented prostate. For all groups the prostate shape variations are extremely small.	130
6.5	Shape Visualization of multiple organs for a single patient across time. The coloring in the 3D view encodes the amount of shape variation (low  high). .	131
6.6	Deviation (low  high) of the aggregated timesteps for each group	132
6.7	Cohort visualization of bladder shapes with their individual shape types ( Group 1,  Group 2,  Group 3,  Group 4), Group 3 has an increased number of patients with toxicity.	133
6.8	Comparison of bladder shapes with different alignment strategies.	134
6.9	Local shape variations of the four different groups.	135
6.10	Position variations of the four different groups.	135
6.11	Outlier detection for individual patients.	137
		153

6.12	Bladder shape visualizations of an individual patient with a distinct outlier.	138
6.13	Local shape variations	138
6.14	Bladder shape differences (low  high) for patients  with and  without toxicity.	139
6.15	Local shape and position variations for patients  with and  without toxicity.	140
7.1	 Aggregation,  Download,  Processing Loading time for a shape variability plot of a bladder for an aggregated group of patients. The total time is split into the activities: shape aggregation and statistical analysis on the server, download from the server to the client, and client-side processing to create the displayed mesh.	144

List of Tables

2.1	Cancer incidence rate in Austrian men in 2016. Prostate cancer is the leading type of cancer [AUS19a].	8
2.2	Number of segmented parts present in the data per patient ($n = 29$). All patients have a segmented bladder, rectum and femur heads. 26 Patients have either a CTV or a segmented prostate with seminal vesicles (in two cases the prostate and seminal vesicles are separately segmented). Although each patient has some sort of PTV. The type of PTV, defined by the radiation intensity, strongly varies from patient to patient. Each one of them has a unique therapy.	18
3.1	Overview of all related works and how their individual contributions can be used to solve our proposed research questions. (✗ = No, (✓) = Partially, ✓ = Yes). As can be seen no work offers a complete solution to the presented problems.	36
4.1	Distribution characteristics describing the aggregated shape differences for each organ measured in their respective shape spaces.	79
4.2	Results of statistical tests performed on the shape difference values. Hereby we tested if the data for each organ is normally or uniformly distributed. We found slight indications that the data follows a normal distribution. . . .	80
4.3	Percentage of all shapes and positions that follow a normal distribution if tested on a per-patient basis with a significance level of 5%.	99

Bibliography

- [AF07] John Ashburner and Karl J Friston. Rigid body registration. *Statistical parametric mapping: The analysis of functional brain images*, pages 49–62, 2007.
- [AGL⁺13] Anet Aselmaa, Richard H M Goossens, Anne Laprie, Sakakibara Ken, Tamara Fechter, Anjana Ramkumar, and Adinda Freudenthal. Workflow analysis report. *Delft University of Technology*, 2013.
- [AHK01] Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory*, pages 420–434. Springer, 2001.
- [AMHH18] Tomas Akenine-Moller, Eric Haines, and Naty Hoffman. *Real-time rendering*. AK Peters/CRC Press, 2018.
- [APW⁺18] Micah Allen, Davide Poggiali, Kirstie Whitaker, Tom R Marshall, and Rogier Kievit. Raincloud plots: a multi-platform tool for robust data visualization. *PeerJ Preprints*, 6, 2018.
- [AUS19a] STATISTIK AUSTRIA. Carcinoma in situ (CIS)-Fälle (Neuerkrankungen pro Jahr) nach Lokalisation und Geschlecht, Österreich seit 2006. https://www.statistik.at/web_de/statistiken/menschen_und_gesellschaft/gesundheit/krebserkrankungen/krebs_im_ueberblick/index.html, 2019. Accessed: 2019-04-01.
- [AUS19b] STATISTIK AUSTRIA. Prostata (C61) - Krebsmortalität (Sterbefälle pro Jahr), Österreich ab 1983. https://www.statistik.at/web_de/statistiken/menschen_und_gesellschaft/gesundheit/krebserkrankungen/prostata/index.html, 2019. Accessed: 2019-04-04.
- [AW10] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

- [BBF⁺11] Stef Busking, Charl P Botha, Luca Ferrarini, Julien Milles, and Frits H Post. Image-based rendering of intersecting surfaces for dynamic comparative visualization. *The visual computer*, 27(5):347–363, 2011.
- [BBP10] Stef Busking, Charl P Botha, and Frits H Post. Dynamic multi-view exploration of shape spaces. In *Computer Graphics Forum*, volume 29, pages 973–982. Wiley Online Library, 2010.
- [BBS⁺18] Michael Blumenschein, Michael Behrisch, Stefanie Schmid, Simon Butscher, Deborah R Wahl, Karoline Villinger, Britta Renner, Harald Reiterer, and Daniel A Keim. Smartexplore: Simplifying high-dimensional data analysis through a table-based visual analytics approach. In *IEEE Conference on Visual Analytics Science and Technology (VAST) 2018*, 2018.
- [BF74] Morton B Brown and Alan B Forsythe. Robust tests for the equality of variances. *Journal of the American Statistical Association*, 69(346):364–367, 1974.
- [BKC⁺13] Rita Borgo, Johannes Kehrner, David HS Chung, Eamonn Maguire, Robert S Laramee, Helwig Hauser, Matthew Ward, and Min Chen. Glyph-based visualization: Foundations, design guidelines, techniques and applications. In *Eurographics (STARs)*, pages 39–63, 2013.
- [But69] Arthur R Butz. Convergence with Hilbert’s space filling curve. *Journal of Computer and System Sciences*, 3(2):128–146, 1969.
- [BVG10] Jean-Paul Balabanian, Ivan Viola, and Eduard Gröller. Interactive illustrative visualization of hierarchical volume data. In *Proceedings of Graphics Interface 2010*, pages 137–144. Canadian Information Processing Society, 2010.
- [CGL⁺10] Pierre Castadot, Xavier Geets, John Aldo Lee, Nicolas Christian, and Vincent Grégoire. Assessment by a deformable registration method of the volumetric and positional changes of target volumes and organs at risk in pharyngo-laryngeal tumors treated with concomitant chemo-radiation. *Radiotherapy and Oncology*, 95(2):209–217, 2010.
- [CGM⁺16] Davide Ceneda, Theresia Gschwandtner, Thorsten May, Silvia Miksch, Hans-Jörg Schulz, Marc Streit, and Christian Tominski. Characterizing guidance in visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):111–120, 2016.
- [CLMW11] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

- [CM02] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):603–619, 2002.
- [CMMH⁺17] Oscar Casares-Magaz, Vitali Moiseenko, Austin Hopper, Niclas Johan Pettersson, Maria Thor, Rick Knopp, Joseph O Deasy, Ludvig Paul Muren, and John Einck. Associations between volume changes and spatial dose metrics for the urinary bladder during local versus pelvic irradiation for prostate cancer. *Acta Oncologica*, 56(6):884–890, 2017.
- [DCH88] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *ACM Siggraph Computer Graphics*, volume 22, pages 65–74. ACM, 1988.
- [DD09] Michel Marie Deza and Elena Deza. Encyclopedia of distances. In *Encyclopedia of distances*, pages 1–583. Springer, 2009.
- [DHS12] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.
- [DKN⁺99] David P Dearnaley, Vincent S Khoo, Andrew R Norman, Lesley Meyer, Alan Nahum, Diana Tait, John Yarnold, and Alan Horwich. Comparison of radiation side-effects of conformal and conventional radiotherapy in prostate cancer: a randomised trial. *The Lancet*, 353(9149):267–272, 1999.
- [DLMB04] Andreas Daffertshofer, Claudine JC Lamoth, Onno G Meijer, and Peter J Beek. PCA in studying coordination and variability: a tutorial. *Clinical biomechanics*, 19(4):415–428, 2004.
- [DS08] Leila DeFloriani and Michela Spagnuolo. *Shape analysis and structuring*. Springer, 2008.
- [DSTA95] Jérôme Declerck, Gérard Subsol, Jean-Philippe Thirion, and Nicholas Ayache. Automatic retrieval of anatomical structures in 3D medical images. In *International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine*, pages 153–162. Springer, 1995.
- [EBC⁺10] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 11(Feb):625–660, 2010.
- [EF09] Niklas Elmqvist and Jean-Daniel Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2009.

- [EST09] Laura Esserman, Yiwey Shieh, and Ian Thompson. Rethinking screening for breast cancer and prostate cancer. *Jama*, 302(15):1685–1692, 2009.
- [FA96] Jacques Feldmar and Nicholas Ayache. Rigid, affine and locally affine registration of free-form surfaces. *International journal of computer vision*, 18(2):99–119, 1996.
- [FBW16] Florian Ferstl, Kai Bürger, and Rüdiger Westermann. Streamline variability plots for characterizing the uncertainty in vector field ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):767–776, 2016.
- [FGS⁺17] Katarina Furmanova, Samuel Gratzl, Holger Stitz, Thomas Zichner, Miroslava Jaresova, Martin Ennemoser, Alexander Lex, and Marc Streit. Taggle: Scalable visualization of tabular data through aggregation. *arXiv preprint arXiv:1712.05944*, 2017.
- [FGW02] Usama Fayyad, Georges G Grinstein, and Andreas Wierse. *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 2002.
- [FKRW16] Florian Ferstl, Mathias Kanzler, Marc Rautenhaus, and Rüdiger Westermann. Visual analysis of spatial variability and global correlations in ensembles of iso-contours. In *Computer Graphics Forum*, volume 35, pages 221–230. Wiley Online Library, 2016.
- [FKRW17] Florian Ferstl, Mathias Kanzler, Marc Rautenhaus, and Rüdiger Westermann. Time-hierarchical clustering and visualization of weather forecast ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):831–840, 2017.
- [FVRS09] Claudio Fiorino, Riccardo Valdagni, Tiziana Rancati, and Giuseppe Sanguineti. Dose–volume effects for normal tissues in external radiotherapy: pelvis. *Radiotherapy and Oncology*, 93(2):153–167, 2009.
- [GAHGGPT15] Diego Alonso Gómez-Aguilar, Ángel Hernández-García, Francisco J García-Peñalvo, and Roberto Therón. Tap into visual analysis of customization of grouping of activities in elearning. *Computers in Human Behavior*, 47:60–67, 2015.
- [GKGR18] Nicolas Grossmann, Thomas Köppel, Eduard Gröller, and Renata G Raidou. Visualflatter: visual analysis of distortions in the projection of biomedical structures. In *Proceedings of the Eurographics Workshop on Visual Computing for Biology and Medicine*, pages 167–177. Eurographics Association, 2018.

- [GRM15] Youness Aliyari Ghassabeh, Frank Rudzicz, and Hamid Abrishami Moghaddam. Fast incremental LDA feature extraction. *Pattern Recognition*, 48(6):1999–2012, 2015.
- [HAB⁺08] Axel Heidenreich, Gunnar Aus, Michel Bolla, Steven Joniau, Vsevolod B Matveev, Hans Peter Schmid, and Filliberto Zattoni. EAU guidelines on prostate cancer. *European Urology*, 53(1):68–80, 2008.
- [HB03] Mark Harrower and Cynthia A Brewer. Colorbrewer.org: an online tool for selecting colour schemes for maps. *The Cartographic Journal*, 40(1):27–37, 2003.
- [Hil35] David Hilbert. Über die stetige Abbildung einer Linie auf ein Flächenstück. In *Dritter Band: Analysis Grundlagen der Mathematik Physik Verschiedenes*, pages 1–2. Springer, 1935.
- [HK15] Max Hermann and Reinhard Klein. A visual analytics perspective on shape analysis: state of the art and future prospects. *Computers & Graphics*, 53:63–71, 2015.
- [HN98] Jerry L Hintze and Ray D Nelson. Violin plots: a box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.
- [HSIHP06] Haleh Hagh-Shenas, Victoria Interrante, and Cheong Hee-Park. A closer look at texture metrics for visualization. In *Human Vision and Electronic Imaging XI*, volume 6057, page 60570W. International Society for Optics and Photonics, 2006.
- [HSK11] Max Hermann, Anja C Schunke, and Reinhard Klein. Semantically steered visual analysis of highly detailed morphometric shape spaces. In *2011 IEEE Symposium on Biological Data Visualization (BioVis)*, pages 151–158. IEEE, 2011.
- [HSSK14] Max Hermann, Anja C Schunke, Thomas Schultz, and Reinhard Klein. A visual analytics approach to study anatomic covariation. In *2014 IEEE Pacific Visualization Symposium*, pages 161–168. IEEE, 2014.
- [HTA⁺18] Joses Ho, Tayfun Tumkaya, Sameer Aryal, Hyungwon Choi, and Adam Claridge-Chang. Moving beyond p values: Everyday data analysis with estimation plots. *BioRxiv*, 2018.
- [HW13] Julian Heinrich and Daniel Weiskopf. State of the art of parallel coordinates. In *Eurographics (STARs)*, pages 95–116, 2013.
- [HWS⁺83] Hedvig Hricak, Richard D Williams, David B Spring, Kik L Moon Jr, Marcus W Hedgcock, Richard A Watson, and Lawrence E Crooks. Anatomy and pathology of the male pelvis by magnetic resonance imaging. *American Journal of Roentgenology*, 141(6):1101–1110, 1983.

- [ID90] Alfred Inselberg and Bernard Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proceedings of the 1st conference on Visualization'90*, pages 361–378. IEEE Computer Society Press, 1990.
- [Jai10] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.
- [JFC14] Jimmy Johansson, Camilla Forsell, and Matthew Cooper. On the usability of three-dimensional display in parallel coordinates: Evaluating the efficiency of identifying two-dimensional relationships. *Information Visualization*, 13(1):29–41, 2014.
- [JYW99] David A Jaffray, Dongsheng Yan, and John W Wong. Managing geometric uncertainty in conformal intensity-modulated radiation therapy. In *Seminars in Radiation Oncology*, volume 9, pages 4–19. Elsevier, 1999.
- [KARC15] Rassadarie Kanjanabose, Alfie Abdul-Rahman, and Min Chen. A multi-task comparative study on scatter plots and parallel coordinates plots. In *Computer Graphics Forum*, volume 34, pages 261–270. Wiley Online Library, 2015.
- [KCK17] Kyungyoon Kim, John V Carlis, and Daniel F Keefe. Comparison techniques utilized in spatial 3D and 4D data visualizations: A survey and future directions. *Computers & Graphics*, 67:138–147, 2017.
- [KEB99] Sei-ichiro Kamata, Richard O Eason, and Yukihiro Bandou. A new algorithm for n-dimensional Hilbert scanning. *IEEE Transactions on Image Processing*, 8(7):964–973, 1999.
- [Kei02] Daniel A Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.
- [Ken77] David G Kendall. The diffusion of shape. *Advances in applied probability*, 9(3):428–430, 1977.
- [KERC09] Daniel Keefe, Marcus Ewert, William Ribarsky, and Remco Chang. Interactive coordinated multiple-view visualization of biomechanical motion data. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1383–1390, 2009.
- [KFR03] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation invariant spherical harmonic representation of 3 d shape descriptors. In *Symposium on geometry processing*, volume 6, pages 156–164, 2003.
- [KHD⁺10] Daniel A Keim, Ming C Hao, Umeshwar Dayal, Halldor Janetzko, and Peter Bak. Generalized scatter plots. *Information Visualization*, 9(4):301–311, 2010.

- [KLR⁺13] Paul Klemm, Kai Lawonn, Marko Rak, Bernhard Preim, Klaus D Tönnies, Katrin Hegenscheid, Henry Völzke, and Steffen Oeltze. Visualization and analysis of lumbar spine canal variability in cohort study data. In *Vision, Modeling & Visualization*, pages 121–128, 2013.
- [KOJL⁺14] Paul Klemm, Steffen Oeltze-Jafra, Kai Lawonn, Katrin Hegenscheid, Henry Völzke, and Bernhard Preim. Interactive visual analysis of image-centric cohort study data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1673–1682, 2014.
- [Kos01] Robert Kosara. *Semantic Depth of Field - Using Blur for Focus + Context Visualization*. PhD thesis, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, 2001.
- [KPB14] Josua Krause, Adam Perer, and Enrico Bertini. Infuse: interactive feature selection for predictive modeling of high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1614–1623, 2014.
- [LACS08] Wilmot Li, Maneesh Agrawala, Brian Curless, and David Salesin. Automated generation of interactive 3D exploded view diagrams. In *ACM Transactions on Graphics (TOG)*, volume 27, pages 101–108. ACM, 2008.
- [LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436, 2015.
- [LC87] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM siggraph computer graphics*, volume 21, pages 163–169. ACM, 1987.
- [Lil67] Hubert W Lilliefors. On the Kolmogorov-Smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62(318):399–402, 1967.
- [LL12] Gaëtan Lehmann and David Legland. Efficient n-dimensional surface estimation using crofton formula and run-length encoding. *The Insight Journal*, 2012.
- [LM17] Aran Lunzer and Amelia McNamara. Exploring histograms, an essay by Aran Lunzer and Amelia McNamara. <https://tinlizzie.org/histograms/#>, 2017. Accessed: 2019-05-28.
- [Lun16] Frank Luna. *Introduction to 3D game programming with DirectX 12*. Stylus Publishing, LLC, 2016.

- [Mac86] Jock Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions On Graphics (TOG)*, 5(2):110–141, 1986.
- [Mac92] Alan M MacEachren. Visualizing uncertain information. *Cartographic Perspectives*, (13):10–19, 1992.
- [Man11] S Manikandan. Measures of dispersion. *Journal of Pharmacology and Pharmacotherapeutics*, 2(4):315, 2011.
- [MBL⁺02] Steven C Mitchell, Johan G Bosch, Boudewijn PF Lelieveldt, Rob J Van der Geest, Johan HC Reiber, and Milan Sonka. 3-D active appearance models: segmentation of cardiac MR and ultrasound images. *IEEE Transactions on Medical Imaging*, 21(9):1167–1178, 2002.
- [McL04] Geoffrey McLachlan. *Discriminant analysis and statistical pattern recognition*, volume 544. John Wiley & Sons, 2004.
- [MF17] Justin Matejka and George Fitzmaurice. Same stats, different graphs: Generating datasets with varied appearance and identical statistics through simulated annealing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 1290–1294. ACM, 2017.
- [MH08] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.
- [MJFS01] Bongki Moon, Hosagrahar V Jagadish, Christos Faloutsos, and Joel H Saltz. Analysis of the clustering properties of the Hilbert space-filling curve. *IEEE Transactions on Knowledge and Data Engineering*, 13(1):124–141, 2001.
- [MK01] Aleix M Martínez and Avinash C Kak. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233, 2001.
- [MMNG15] Haichao Miao, Gabriel Mistelbauer, Christian Nasel, and Eduard Gröller. Cowradar: Visual quantification of the circle of willis in stroke patients. In *Proceedings of the Eurographics Workshop on Visual Computing for Biology and Medicine*, pages 1–10, 2015.
- [moy] Screening for prostate cancer: US Preventive Services Task Force recommendation statement, author=Moyer, Virginia A, journal=Annals of Internal Medicine, volume=157, number=2, pages=120–134, year=2012, publisher=Am Coll Physicians.
- [MSD03] Ludvig Paul Muren, Rune Smaaland, and Olav Dahl. Organ motion, set-up variation and treatment margins in radical radiotherapy of urinary bladder cancer. *Radiotherapy and Oncology*, 69(3):291–304, 2003.

- [MSJ⁺15] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.
- [MW95] Allen R Martin and Matthew O Ward. High dimensional brushing for interactive exploration of multivariate data. In *Proceedings of the 6th Conference on Visualization'95*, page 271. IEEE Computer Society, 1995.
- [MWK14] Mahsa Mirzargar, Ross T Whitaker, and Robert M Kirby. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2654–2663, 2014.
- [NDSM⁺19] Siamak P Nejad-Davarani, Parag Sevak, Michael Moncion, Kimberly Garbarino, Steffen Weiss, Joshua Kim, Lonni Schultz, Mohamed A Elshaikh, Steffen Renisch, and Carri Glide-Hurst. Geometric and dosimetric impact of anatomical changes for mr-only radiation therapy for the prostate. *Journal of Applied Clinical Medical Physics*, 2019.
- [PAG⁺06] Michael Pinkawa, Branka Asadpour, Bernd Gagel, Marc D Piroth, Richard Holy, and Michael J Eble. Prostate position variability and dose–volume histograms in radiotherapy for prostate cancer with full and empty bladder. *International Journal of Radiation Oncology* Biology* Physics*, 64(3):856–861, 2006.
- [PBTM06] Alain Pitiot, Eric Bardinet, Paul M Thompson, and Grégoire Malandain. Piecewise affine registration of biological images for volume reconstruction. *Medical image analysis*, 10(3):465–483, 2006.
- [PF03] Josien PW Pluim and J Michael Fitzpatrick. Image registration. *IEEE Transactions on Medical Imaging*, 22(11):1341–1343, 2003.
- [PHL⁺16] Nicola Pezzotti, Thomas Höllt, B Lelieveldt, Elmar Eisemann, and Anna Vilanova. Hierarchical stochastic neighbor embedding. In *Computer Graphics Forum*, volume 35, pages 21–30. Wiley Online Library, 2016.
- [PI97] Markus Peura and Jukka Iivarinen. Efficiency of simple shape descriptors. *Aspects of visual form*, pages 443–451, 1997.
- [PLV⁺07] David Pasquier, Thomas Lacornerie, Maximilien Vermandel, Jean Rousseau, Eric Lartigau, and Nacim Betrouni. Automatic segmentation of pelvic structures from magnetic resonance images for prostate cancer radiotherapy. *International Journal of Radiation Oncology* Biology* Physics*, 68(2):592–600, 2007.
- [PW13] Tobias Pfaffelmoser and Rüdiger Westermann. Visualizing contour distributions in 2D ensemble data. *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers*, 71(72):133–138, 2013.

- [QLL⁺12] Enzhuo M Quan, Xiaoqiang Li, Yupeng Li, Xiaochun Wang, Rajat J Kudchadker, Jennifer L Johnson, Deborah A Kuban, Andrew K Lee, and Xiaodong Zhang. A comprehensive comparison of IMRT and VMAT plan quality for prostate cancer treatment. *International Journal of Radiation Oncology* Biology* Physics*, 83(4):1169–1178, 2012.
- [RBGR18] Oliver Reiter, Marcel Breeuwer, Eduard Gröller, and Renata G Raidou. Comparative visual analysis of pelvic organ segmentations. In *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers*, pages 37–41. Eurographics Association, 2018.
- [RBL⁺09] Mauricio Reyes, Miguel A Gonzalez Ballester, Zhixi Li, Nina Kozic, Ronald M Summers, and Marius George Lingurar. Anatomical variability of organs via principal factor analysis from the construction of an abdominal probabilistic atlas. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 682–685. IEEE, 2009.
- [RCG⁺17] Douglas Richardson, Noel Castree, Michael F Goodchild, Audrey Lynn Kobayashi, Weidong Liu, and Richard A Marston. *The International Encyclopedia of Geography: AB*, volume 1. John Wiley & Sons, 2017.
- [RCMA⁺18] Renata Georgia Raidou, Oscar Casares-Magaz, Aleksandr Amirkhanov, Vitali Moiseenko, Ludvig P Muren, John P Einck, Anna Vilanova, and Eduard Gröller. Bladder runner: Visual analytics for the exploration of rt-induced bladder toxicity in a cohort study. In *Computer Graphics Forum*, volume 37, pages 205–216. Wiley Online Library, 2018.
- [RF00] Marilina Rivero and Francisco R Feito. Boolean operations on general planar polygons. *Computers & Graphics*, 24(6):881–896, 2000.
- [RFGD08] Clemens Reimann, Peter Filzmoser, Robert G Garrett, and Rudolf Dutter. Statistical data analysis explained. *Applied environmental statistics with R. England: Wiley & Sons Ltd*, 2008.
- [RHC99] Yong Rui, Thomas S Huang, and Shih-Fu Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, 1999.
- [Row98] Sam T Roweis. EM algorithms for PCA and SPCA. In *Advances in neural information processing systems*, pages 626–632, 1998.
- [RPSC99] Harvey Ray, Hanspeter Pfister, Deborah Silver, and Todd A Cook. Ray casting architectures for volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 5(3):210–223, 1999.

- [Sag12] Hans Sagan. *Space-filling curves*. Springer Science & Business Media, 2012.
- [Shl14] Jonathon Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014.
- [Shn03] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization*, pages 364–371. Elsevier, 2003.
- [Sig06] Christian Sigg. *Representation and rendering of implicit surfaces*. PhD thesis, ETH Zurich, 2006.
- [SKB18] Vera Skorkovská, Ivana Kolingerová, and Bedrich Benes. A simple and robust approach to computation of meshes intersection. In *VISIGRAPP (1: GRAPP)*, pages 175–182, 2018.
- [SMB⁺10] Martijn D Steenwijk, Julien Milles, M Buchem, J Reiber, and Charl P Botha. Integrated visual analysis for heterogeneous datasets in cohort studies. In *IEEE VisWeek Workshop on Visual Analytics in Health Care*, volume 3, page 3, 2010.
- [SPA⁺14] Johanna Schmidt, Reinhold Preiner, Thomas Auzinger, Michael Wimmer, Eduard Gröller, and Stefan Bruckner. YMCA—your mesh comparison application. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 153–162. IEEE, 2014.
- [Spe01] Robert Spence. *Information visualization*, volume 1. Springer, 2001.
- [SPP00] Andrea Schenk, Guido Prause, and Heinz-Otto Peitgen. Efficient semiautomatic segmentation of 3D objects in medical images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 186–195. Springer, 2000.
- [SRM⁺19] Matthias Schlachter, Renata G Raidou, Ludvig P Muren, Bernhard Preim, and Katja Bühler. State-of-the-art report: Visual computing in radiation therapy planning. In *Computer Graphics Forum*, volume 38, pages 753–779. Wiley Online Library, 2019.
- [SSB⁺05] William Song, Bryan Schaly, Glenn Bauman, Jerry Battista, and Jake Van Dyk. Image-guided adaptive radiation therapy (igart): Radiobiological and dose escalation considerations for localized carcinoma of the prostate. *Medical Physics*, 32(7Part1):2193–2203, 2005.
- [SWD08] Anton Stangelberger, Matthias Waldert, and Bob Djavan. Prostate cancer in elderly men. *Reviews in Urology*, 10(2):111, 2008.

- [SZ12] Douglas S Shafer and Zhiyi Zhang. Beginning statistics. *Phylis-Barnidge publisher*, 304:47–48, 2012.
- [TBK03] Jeyarajan Thiyagalingam, Olav Beckmann, and Paul HJ Kelly. An exhaustive evaluation of row-major, column-major and morton layouts for large two-dimensional arrays. In *Performance Engineering: 19th Annual UK Performance Engineering Workshop*, pages 340–351, 2003.
- [THGE99] Christian Teitzel, Matthias Hopf, Roberto Grosso, and Thomas Ertl. Volume visualization on sparse grids. *Computing and Visualization in Science*, 2(1):47–59, 1999.
- [THLM⁺13] Juliette Thariat, Jean-Michel Hannoun-Levi, Arthur Sun Myint, Te Vuong, and Jean-Pierre Gérard. Past, present, and future of radiotherapy for the benefit of patients. *Nature Reviews Clinical Oncology*, 10(1):52, 2013.
- [TMA01] Melanie K Tory, Torsten Möller, and M Stella Atkins. Visualization of time-varying mri data for ms lesion analysis. In *Medical Imaging 2001: Visualization, Display, and Image-Guided Procedures*, volume 4319, pages 590–599. International Society for Optics and Photonics, 2001.
- [UK19] Cancer Research UK. Pelvic organs, 2019. [Online; accessed 20.07.2019].
- [VDM14] Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *The Journal of Machine Learning Research*, 15(1):3221–3245, 2014.
- [VLBK⁺13] Tatiana Von Landesberger, Sebastian Bremm, Matthias Kirschner, Stefan Wesarg, and Arjan Kuijper. Visual analytics for model-based medical image segmentation: Opportunities and challenges. *Expert Systems with Applications*, 40(12):4934–4943, 2013.
- [VLBM08] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.
- [VS01] Dejan Vranic and Dietmar Saupe. 3D shape descriptor based on 3D Fourier transform. In *EURASIP*, pages 271–274, 2001.
- [VYM⁺10] Akila N Viswanathan, Ellen D Yorke, Lawrence B Marks, Patricia J Eifel, and William U Shipley. Radiation dose–volume effects of the urinary bladder. *International Journal of Radiation Oncology* Biology* Physics*, 76(3):S116–S122, 2010.

- [War02] Matthew O Ward. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, 1(3-4):194–210, 2002.
- [War12] Colin Ware. *Information visualization: perception for design*. Elsevier, 2012.
- [WFG⁺19] Johannes Weissenböck, Bernhard Fröhler, Eduard Gröller, Johann Kastner, and Christoph Heinzl. Dynamic volume lines: Visual comparison of 3D volumes through space-filling curves. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1040–1049, 2019.
- [WHLS18] Junpeng Wang, Subhashis Hazarika, Cheng Li, and Han-Wei Shen. Visualization and visual analysis of ensemble data: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 2018.
- [WL15] Charles M Washington and Dennis T Leaver. *Principles and practice of radiation therapy*. Elsevier Health Sciences, 2015.
- [WMK13] Ross T Whitaker, Mahsa Mirzargar, and Robert M Kirby. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, 2013.
- [Wor89] Brian J Worton. Kernel methods for estimating the utilization distribution in home-range studies. *Ecology*, 70(1):164–168, 1989.
- [WPK89] David F Williamson, Robert A Parker, and Juliette S Kendrick. The box plot: a simple visual method to interpret data. *Annals of Internal Medicine*, 110(11):916–921, 1989.
- [WSM⁺04] Hongyan Wu, Leon Sun, Judd W Moul, Hongyu Wu, DAVID G McLEOD, Christopher Amling, Raymond Lance, LEO Kusuda, Timothy Donahue, John Foley, et al. Watchful waiting and factors predictive of secondary treatment of localized prostate cancer. *The Journal of Urology*, 171(3):1111–1116, 2004.
- [WTK⁺06] Stefan Wachter, Sandra Tomek, Amir Kurtaran, Natascha Wachter-Gerstner, Bob Djavan, Alexander Becherer, Markus Mitterhauser, Georg Dobrozemsky, Shuren Li, Richard Potter, et al. 11c-acetate positron emission tomography imaging and image fusion with computed tomography and magnetic resonance imaging in patients with recurrent prostate cancer. *Journal of Clinical Oncology*, 24(16):2513–2519, 2006.
- [WVJ16] Martin Wattenberg, Fernanda Viégas, and Ian Johnson. How to use t-sne effectively. *Distill*, 1(10):e2, 2016.

- [XW05] Rui Xu and Donald C Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, 2005.
- [YKSJ08] Ji Soo Yi, Youn-ah Kang, John T Stasko, and Julie A Jacko. Understanding and characterizing insights: how do people gain insights using information visualization? In *Proceedings of the 2008 Workshop on Beyond time and errors: novel evaluation methods for Information Visualization*, pages 4:1–4:6. ACM, 2008.
- [ZSK12] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(5):363–387, 2012.
- [ZYF⁺08] Michael J Zelefsky, Yoshiya Yamada, Zvi Fuks, Zhigang Zhang, Margie Hunt, Oren Cahlon, Jessica Park, and Alison Shippy. Long-term results of conformal radiotherapy for prostate cancer: impact of dose escalation on biochemical tumor control and distant metastases-free survival outcomes. *International Journal of Radiation Oncology* Biology* Physics*, 71(4):1028–1033, 2008.