An Efficient Inflation Method for Segmentation of Medical 3D Images

Eine Effiziente Inflationsmethode zur Segmentierung von Medizinischen 3D Bildern

Vom Department Elektrotechnik und Informatik
der Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen

zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von

Dženan Zukić

1. Gutachter:	Prof. Dr. Andreas Kolb
2. Gutachter:	Prof. Dr. Gerik Scheuermann
Vorsitzender:	Prof. Dr. Otmar Loffeld
Mitglied:	Prof. Dr. Marcin Grzegorzek

Tag der mündlichen Prüfung: 8. September 2014

Gedruckt auf alterungsbeständigem holz- und säurefreiem Papier.

Abstract

The diagnosis of certain spine pathologies, such as scoliosis, spondylolisthesis and vertebral fractures, are part of the daily clinical routine. Very frequently, MRI data are used to diagnose these kinds of pathologies in order to avoid exposing patients to harmful radiation, like X-ray.

Developing a segmentation system for an array of vertebrae is complex, so the method was first tested on brain tumors of types glioblastoma multiforme and pituitary adenoma. A small triangular surface mesh at the approximate center of the tumor is inflated towards the boundary using balloon force, keeping it approximately star-shaped. The boundary is implicitly binarized by the inflation rules, based on the minimum and maximum intensity from the initialization step. After the segmentation is finished, the tumor volume is calculated.

The spine segmentation system uses a bottom-up approach for detecting vertebral bodies based on just one manual initialization. A subdivision surface hierarchy is introduced as an efficient global-to-local smoothness constraint, which can be thought of as an internal force. Together with intensities, low-high (LH) values were initially used to ease boundary finding, but the boundary estimation evolved into a multi-feature combiner.

The final system utilizes a Viola-Jones detector to determine centers and approximate sizes of vertebral bodies. This gives the user a chance to manually correct detections, enables parallel feature calculation and segmentation, and is a basis for reliable diagnosis established at the end.

The system was evaluated on 26 lumbar datasets containing 234 reference vertebrae. Vertebra detection has 7.1% false negatives and 1.3% false positives. The average Dice coefficient to manual reference is 79.3% and mean distance error is 1.77 mm. No severe case of the three addressed illnesses was missed, and false alarms occurred rarely – 0% for scoliosis, 3.9% for spondylolisthesis and 2.6% for vertebral fractures.

The main advantages of this system are high speed, robust handling of a large variety of routine clinical images, and simple and minimal user interaction.

Zusammenfassung

Die Diagnose von bestimmten Wirbelsäulenerkrankungen, wie z.B. Skoliose, Spondylolisthesis oder Wirbelbrüche, sind Teil des Klinikalltags. Häufig werden zur Diagnose dieser Art von Erkrankungen MRT-Daten benutzt, um zu vermeiden, dass Patienten schädlicher Strahlung, wie z.B. Röntgenstrahlung, ausgesetzt werden.

Die Entwicklung eines Segmentierungssystems für eine Reihe von Wirbeln ist komplex. Deshalb wurde die Methode zuerst für zwei Typen von Gehirntumoren, Glioblastoma multiforme und Hypophysenadenom, getestet. Ein kleines Dreiecksnetz wird am ungefähren Zentrum des Tumors durch Ballon-Forces expandiert, wobei seine Struktur näherungsweise sternförmig gehalten wird. Der Datensatz wird durch diese Kräfte basierend auf den Minimum- und Maximumintensitäten beim Initialisierungsschritt implizit in ein inneres und ein äußeres Segment unterteilt. Nachdem die Segmentierung abgeschlossen ist, wird das Volumen des Tumors berechnet.

Das Segmentierungssystem für die Wirbelsäule benutzt einen „Bottom-up“-Ansatz zur Erkennung der Wirbel, der auf nur einer manuellen Initialisierung basiert. Als effiziente global-zu-lokal Glättungsbedingung wurde eine Oberflächenunterteilungshierarchie eingeführt, die man sich als interne Kraft vorstellen kann. Zu Beginn wurden Intensitätswerte zusammen mit „low-high“-Werten verwendet um die Ermittlung von Kanten zu erleichtern. Aber der Kantenschätzer entwickelte sich hin zu einem Multimerkmalsansatz.

Das endgültige System benutzt einen Viola-Jones-Detektor um das Zentrum und die ungefähre Größe von Wirbeln zu bestimmen. Dieser Ansatz gibt dem Nutzer die Möglichkeit die Erkennung manuell zu korrigieren und ermöglicht eine parallele Berechnung der Merkmale und Segmentierung und stellt eine Basis für eine zuverlässige Diagnose dar.

Das System wurde an 26 lumbalen Datensätzen evaluiert, welche 234 Referenzwirbel beinhalteten. Die Wirbelerkennung hat 7.1% „false positives“ und 1.3% „false negatives“. Der durchschnittliche Dice-Koeffizient im Vergleich zur Handsegmentierung ist 79.3% und der mittlere Abstandsfehler beträgt 1.77mm. Alle schlimmere Fälle der drei Erkrankungen wurde korrekt erkannt und Fehlalarme traten selten auf – 0% bei Skoliose, 3.9% bei Spondylolisthesis und 2.6% bei Wirbelfrakturen.

Die Hauptvorteile dieses Systems sind die hohe Geschwindigkeit, die robuste Handhabung von alltäglichen klinischen Aufnahmen und die einfache als auch minimale Benutzerinteraktion.

Contents

Chapter 1

Introduction

In the past few decades magnetic resonance imaging (MRI) has become an important diagnostic tool in the clinical routine. The reason for this is MRI's ability to capture a wide range of properties from living tissue [Bla08]. MRI is of great value for various diagnosing tasks, but for most illnesses it is not the first choice because of its high price. MRI is frequently used for neurological diagnosis because of the detail it provides for brain tissue, spinal cord and other tissues of the spine and around the spine.

1.1 Problem Statement

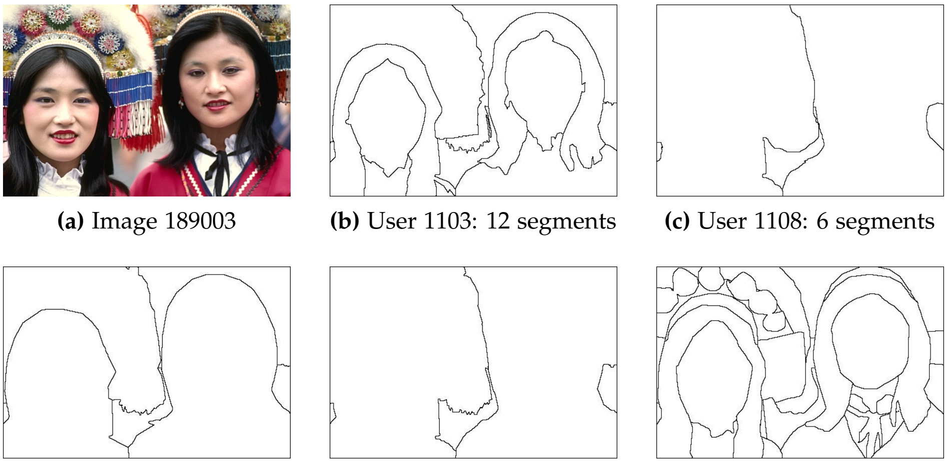
Image segmentation is the process of splitting an image into its constituent regions. Frequently, the image is divided into two parts: the object of interest and background to be ignored [SS01, 279]. The usual goal of segmentation is a simplified representation or a more meaningful or easier analysis.

Segmentation is useful for many things: face detection in images, pedestrian detection in surveillance videos, fingerprint recognition, etc. This thesis is, however, mostly concerned with medical applications, i.e.:

1. separating tumors from other parts of the brain and head
2. separating vertebral bodies from the rest of the acquired anatomy.

The main reason for segmenting tumors is estimating their volume. The reason for segmenting vertebral bodies is aiding diagnosis of certain diseases.

Segmentation is inherently ambiguous. This is clearly depicted in Figure 1.1, where five humans disagree on how the image should be segmented.



(a) Image 189003 **(b)** User 1103: 12 segments **(c)** User 1108: 6 segments

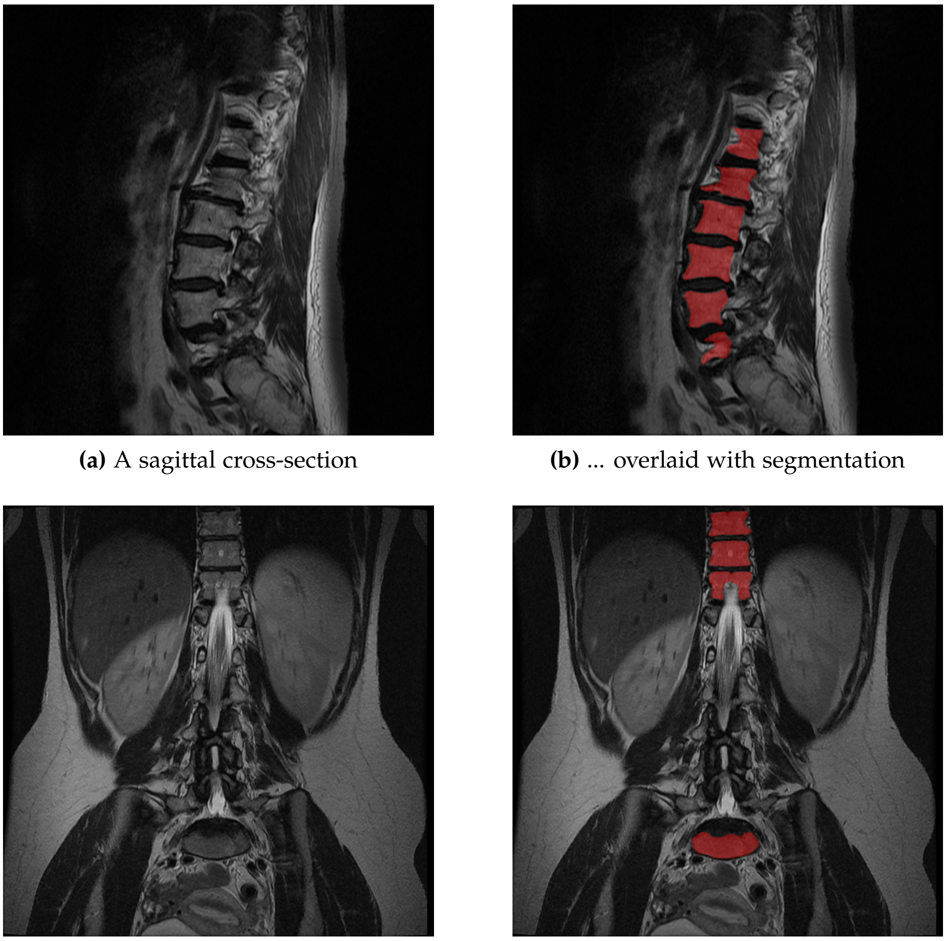
(d) User 1109: 8 segments **(e)** User 1112: 5 segments **(f)** User 1130: 41 segments

Figure 1.1: An image and segmentations from 5 different users. The image and its segmentations are taken from The Berkeley Segmentation Dataset and Benchmark [MFTM01].

It is very hard to segment images of an unknown scene with unknown lighting, but if the set of possible images is restricted the problem becomes easier (Figure 1.2 on the next page). Just restricting ourselves to radiology images completely eliminates the lighting variability. Radiology images also usually imply human subjects, which is another restriction.

However, radiological images also introduce new problems, such as little to no distinction between different tissues in computed tomography (CT) and inhomogeneous intensity due to inhomogeneous magnetic field in magnetic resonance (MR).

Noise in the images hampers and complicates all attempts to achieve a proper segmentation.



(a) A sagittal cross-section

(b) ... overlaid with segmentation

(c) A coronal cross-section

(d) ... overlaid with segmentation

Figure 1.2: MR images of the lumbar spine of two different subjects

1.2 Coarse System Overview

The 3D image is acquired by a medical scanner and stored into a DICOM format [MEM02] file. The image file is preprocessed to reduce noise and prepare it for usage by the segmentation algorithm. After segmentation is finished, analysis can be conducted. The analysis can be as simple as calculating the volume of the segmented tumor, or more complicated as calculation of the degree of scoliosis (see Fig. 1.3).

In this thesis the emphasis is on the segmentation and associated preprocessing, and less effort is devoted to the analysis (diagnosis).

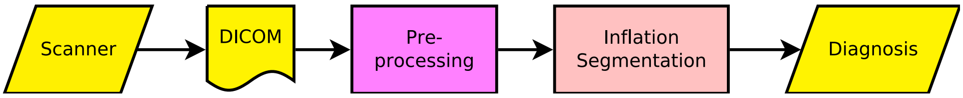


Figure 1.3: System overview

1.3 Contribution

This thesis deals with fast segmentation of medical magnetic resonance images (MRI). Towards that goal, the main contributions are:

- very fast and versatile basic segmentation algorithm for single star-shaped object [ZEB+10, EZB+10, ZEB+11, EZF+13]
- constrained subdivision hierarchy for smoothness control in deformable meshes [ZVD+12]
- multiple feature boundary estimation producing a smooth edge-probability image thus enabling gradient descent approach to edge finding [ZVD+12]
- vertebral column segmentation framework which handles diverse set of MR images. There are two variants, utilizing a:
  - bottom-up approach to vertebra detection [ZVD+12]
  - top-down approach to vertebra detection [ZVE+14]
- diagnosis of scoliosis, spondylolisthesis and vertebral collapse

1.4 Structure of Thesis

The next chapter gives a an overview of how 3D magnetic resonance images are created, their properties, a short description of several different approaches to their segmentation and review of published work related to brain tumors and vertebra segmentation.

Before attempting a segmentation of the vertebral column, the inflation approach was tested on brain tumors. Chapter 3 on page 39 introduces this single-object segmentation system.

Chapter 4 on page 55 builds upon the previous algorithm and extends it to multiple object segmentation, namely vertebral bodies. Based on the pose of the initial or previous vertebral body, the center of the adjacent vertebral body is predicted. Thus only one user-initialization is required. A constrained subdivision hierarchy is used to control smoothness.

Improvements to boundary estimation are the core of chapter 5 on page 75.

The final system is presented in chapter 6 on page 83. It uses automatic detection of vertebral bodies before the main segmentation step. This gives the user a chance to rectify initial detection errors, as correct detection is crucial to diagnostic analysis. Diagnosis of scoliosis, spondylolisthesis and vertebral collapse is implemented. Initial vertebra detection also enables parallel implementation of feature calculation and segmentation. Few other improvements are also given, such as parameter optimization.

Finally, chapter 7 on page 99 presents results for the three vertebral body segmentation systems presented in chapters 4, 5 and 6.

Chapter 2

Technical Foundation

Raster images, arising from observations of the real world, are arrays of points with different intensities, called pixels (*pi*cture *el*ements). Common example of two-dimensional (2D) images are photographs. Images don't have to be two-dimensional, they can be three-dimensional in various ways. If the image's pixel intensities do not represent incident light intensities, but rather distances from the camera to the points in the scene, those images are called range images or depth images. If the third dimension is time (2D+t), they are usually called video.

Images can also have three spatial dimensions – x, y, and z. In that case they are called volumetric or three-dimensional (3D) images. Elements of such images can still be called pixels, but are usually called voxels (*vo*lume *el*ements) to distinguish them from 2D pixels. Such images are produced by 3D computer simulations, X-ray CT, MRI, confocal microscopy etc. Even though special volumetric displays exist, they are usually displayed on conventional 2D displays using volume rendering and slicing. Slicing displays a single 2D slice of a 3D image at a time.

Images can have higher dimensions too. Common 4D images are 3D images with multiple time samples (3D+t) and multispectral 3D images (3D+λ), e.g. coming from confocal Raman microscopy.

The normal photographs have three color channels – red, green and blue. Multispectral images have more channels, sometimes in the range of hundreds [Lab13]. This thesis, however, deals with grayscale 3D images – images with only one channel.

2.1 Principle of Operation of MRI Scanner

Biologic tissues, including the human body, contain a large percentage of water and therefore a significant abundance of hydrogen atoms [Jäg11]. The nuclei of the ^{1}H atoms are protons, which have quantum spin of $\frac{1}{2}$. If protons are brought into a magnetic field, their spins tend to align parallel or anti-parallel to the field. The exact distribution of spins is governed by Boltzmann statistics, but somewhat more of the protons align parallel to the field $\vec{B_0}$ yielding a net magnetization $\vec{M}$.

When $\vec{M}$ is not parallel to $\vec{B_0}$, torque is exerted on it causing it to precess around $\vec{B_0}$, see Figure 2.1. The angular velocity of this precession is given by $\omega_0 = \gamma\|\vec{B_0}\|$. γ is called gyromagnetic constant and for ^{1}H nucleus it is 42.6 MHz/T (megahertz per Tesla), see [Jäg11].

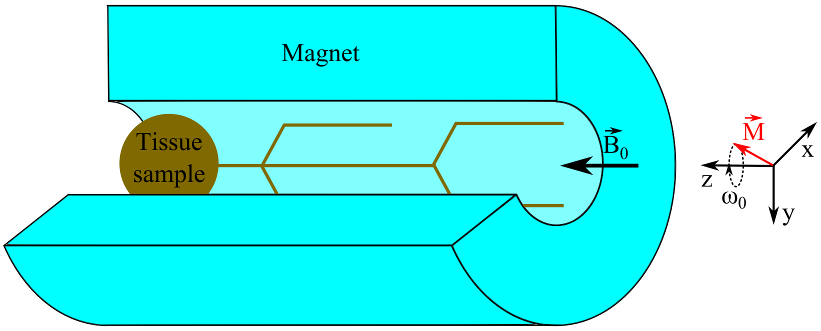


Figure 2.1: Protons with magnetization $\vec{M}$ precess around the magnetic field $\vec{B_0}$

The magnetization vector $\vec{M}$ can be flipped sideways if energy is introduced to the tissue sample using a radio-frequency (RF) pulse. This RF pulse has to match the resonant frequency of the sample, ω_0. When the RF pulse is turned off, the transverse component of the magnetization vector produces an oscillating magnetic field which can be measured.

The magnetization induced falls off exponentially. Longitudinal magnetization M_z (magnetization along $\vec{B_0}$) is restored with a time constant T_1 and transverse magnetization M_{xy} (magnetization perpendicular to $\vec{B_0}$) falls off with a time constant T_2:

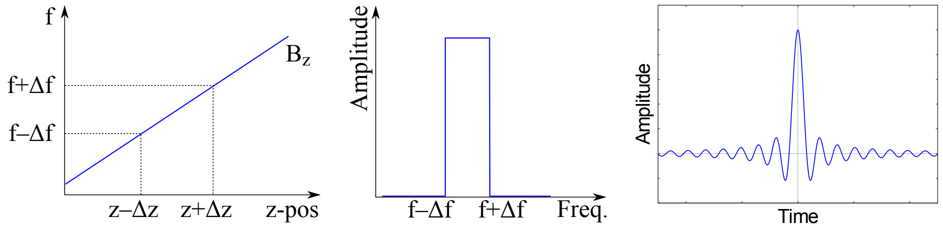
$$M_z(t) = (1 - e^{\frac{-t}{T_1}})M_0$$

$$M_{xy}(t) = e^{\frac{-t}{T_2}}M_0$$

The relaxation time constants T_1 and T_2 are tissue dependent, and typical values are around 1 second for T_1 and a few dozen milliseconds for T_2. These constants cannot be measured independently. Since T_1 is usually much longer than T_2, transverse magnetization falls off much quicker. If we measure the magnetization short time ($\approx T_2$) after the flip pulse, the signal will depend much more on T_2 because $M_z(t) \approx 0$ and $M_{xy} \neq 0$. This is called T_2 weighting. Similarly if we measure the signal long time ($\approx T_1$) after the flip pulse, it will be T_1 weighted.

2.1.1 Spatial Encoding

The theory so far gives us the way to measure aggregate signal of the whole tissue sample (usually the human body). To be of practical use, a way to measure the signal at the different spatial locations is required.



(a) Correspondence between z-position and frequency **(b)** Frequency range **(c)** Time domain sinc signal

Figure 2.2: The required z-position and the slice thickness $2\Delta z$ dictate the frequency, which through Fourier transform defines the RF pulse used.

The spin precession frequency $f = \frac{\omega}{2\pi}$ can be made position dependent if we make the total field strength $\|\vec{B}\|$ position dependent. This is usually accomplished by applying a magnetic gradient field G_z in z-direction to the main magnetic field $\vec{B}_0$, thus making the total field strength z-dependent: $B_z(z) = \|\vec{B}_0\| + G_z z$.

Now we can only flip (excite) a part of the sample, by using a pulse which contains a range of frequencies $(f - \Delta f, f + \Delta f)$. This way only a "slice" of the sample is excited, corresponding to a spatial range $(z - \Delta z, z + \Delta z)$, see Figure 2.2.

With only a single slice excited, meaning all spins flipped down and in phase, we can now focus on separating positions along x and y coordinates. A simple way to separate positions (pixels) along the x axis is to to apply a new gradient field G_x: $B_x(x) = \|\vec{B}_0\| + G_x x$. This way the precession

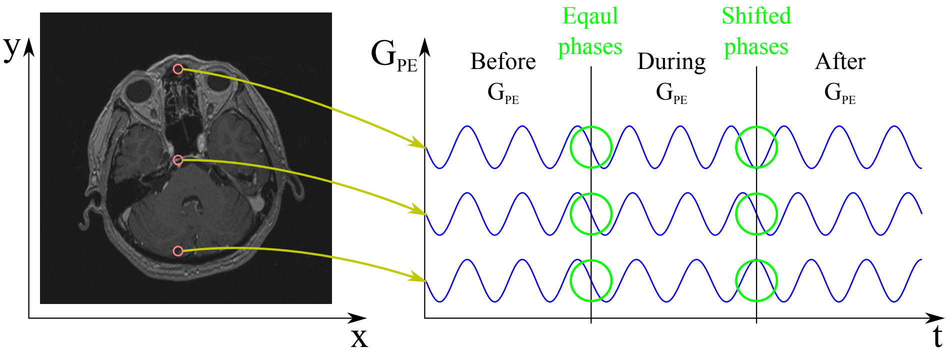


Figure 2.3: Before applying phase encoding gradient, all spins precess at the same frequency with the same phase. When G_{PE} is turned on, the spins start precessing at different frequencies. After G_{PE} is turned off, spins are no longer in phase, and the phase is linearly dependent on y position.

frequency along x axis varies linearly, and the difference in angular velocity with respect to the initial field strength can be calculated as: $\delta\omega = \gamma G_x x$. The gradient G_x is applied during readout, so x axis is also called readout direction or frequency encoding (FE) direction.

With frequency encoding taken care of, the signal which is read out is an integral over the y direction at a fixed x and z position. To differentiate pixels along the y axis, an additional gradient field G_y is applied for a defined duration before the readout. This alters the precession frequency. When G_y is turned off, the precession will continue at the original frequency, but different positions along the y axis will have different phases, see Figure 2.3. G_y is therefore usually called "phase encoding gradient" and symbolized G_{PE}.

2.1.2 A Pulse Sequence

In Figure 2.4 on the next page a complete timing diagram of a basic spin echo pulse sequence is shown.

After the pulse sequence is executed once, one obtains measurements of a single row of pixels of the selected slice. To obtain pixels of the whole slice, the pulse sequence is repeated with a different phase-encoding gradient. These repetitions are done after some waiting time, to allow longitudinal magnetization to recover. Then the pulse sequence is executed n_{PE} times for each of the adjacent slices until the 3D volume is acquired.

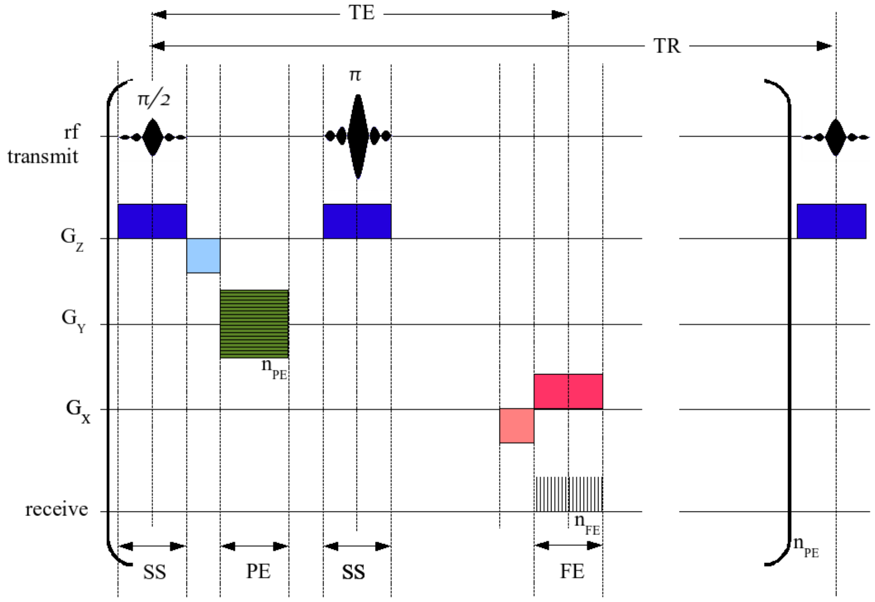


Figure 2.4: Simplified timing diagram for two-dimensional-Fourier-transform (2DFT) Spin Echo (SE) pulse sequence.

The horizontal axis represents time. The vertical axis represents: (top row) amplitude of radio frequency pulses; (middle rows) amplitudes of the three orthogonal magnetic field gradient pulses; and (bottom row) receiver analog-to-digital converter (ADC).

SS - slice selection. PE - phase encoding. FE - frequency encoding. n_{FE} - number of signal samples recorded, usually 128-512. n_{PE} - number of repetitions, usually 128-512. TE - echo time, time between sample excitation and spin echo caused by the refocusing pulse (second SS). TR - repetition time.

Reproduced from Wikipedia [Wik13].

2.1.3 Image Reconstruction

The signals measured for each slice form a two-dimensional matrix, the so-called k-space (Figure 2.5). Its entries are complex numbers formed by measurements of signal power coming from frequency encoding and phase encoding coils. To get a spatial image out of it, an inverse two-dimensional Fourier transform of the sampled data is performed. For human-perceivable image either the magnitude or phase of the Fourier transform can be taken, the magnitude being far more common.

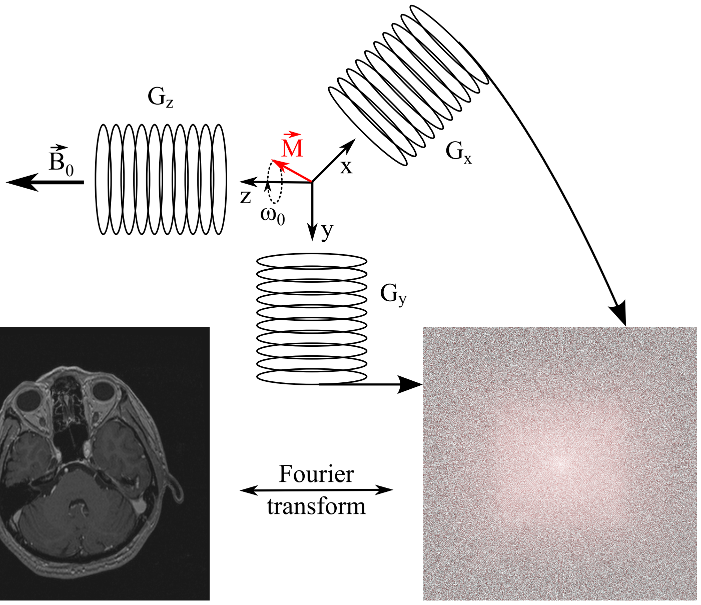


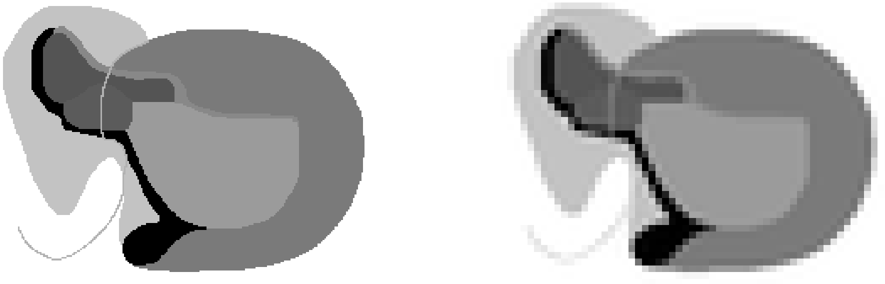
Figure 2.5: Measured signals produce a complex image in k-space. Inverse 2D Fourier transform is applied to get the normal (magnitude) MR image.

2.2 Properties and Artifacts of MRI Data

As all other sensor types, magnetic resonance imaging has artifacts. In addition to that MRI has some other inconvenient properties, such as a lack of unified physical unit. Some artifacts are always present and have to be taken into account in any case, e.g. partial volume effect. Others, such as magnetic susceptibility, occur rarely.

Partial Volume Effect As imaging sensors have a finite spatial resolution, a single voxel will frequently correspond to a volume which contains more than one tissue type. For a combination of two tissue types a and b, which occurs on tissue boundaries, the sampled intensity $i_{measured}$ will not exactly correspond to either of the involved tissue types but will instead be somewhere in between. The resulting intensity i is some kind of average weighted by proportions w of the given tissues a and b in the volume corresponding to the measured voxel:

$$i_{measured} = i_a \times w_a + i_b \times w_b, \text{ where } w_a + w_b = 1 \text{ and } w_a, w_b > 0.$$



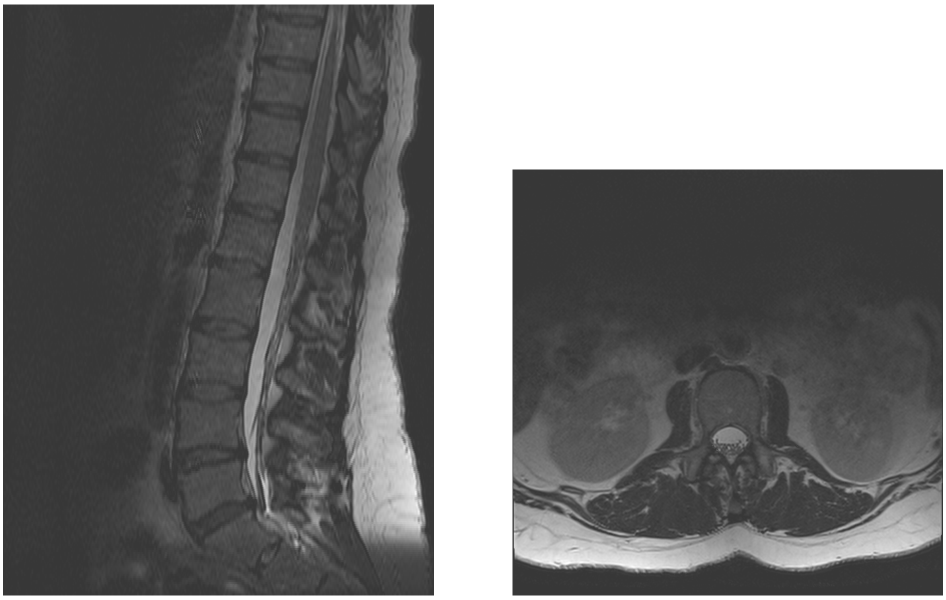
(a) A true state with 9 distinct intensities **(b)** An image with 208 distinct intensities

Figure 2.6: Partial volume effect. In (b) it is hard to even recognize the existence of some thin structures clearly visible in the ground truth image (a).

This in-between intensity usually corresponds to some other tissue type, because biological tissues have a complex structure and usually different parts of organs consist of different tissue types. Structures which are smaller than the image's voxel size and do not cover a complete voxel (such as small blood vessels) will get lost through partial volume effects. This is illustrated in Figure 2.6. The partial volume effect is more severely pronounced in images with anisotropic resolution, because elongated voxels may contain more tissue types.

A common way to deal with this is to use "soft segmentations" which allow a single voxel to partly belong to more than one tissue type. This requires a segmentation mask for each segmented object, and all object-belonging proportions should sum up to 1 (=100%) for each voxel. Segmentation representations which are not voxel-based, such as polygonal surfaces, can be converted into either binary segmentation masks or soft segmentation masks.

Intensity Non-Uniformity One artifact specific to magnetic resonance imaging is intensity inhomogeneity. Because the magnetic field strength is not perfectly homogenous in the entire acquisition volume, the same tissue may have different intensities in different parts of the image. The most frequent manifestation are reduced intensity and contrast away from the center of the acquisition volume (Figure 2.7).



(a) Sagittal

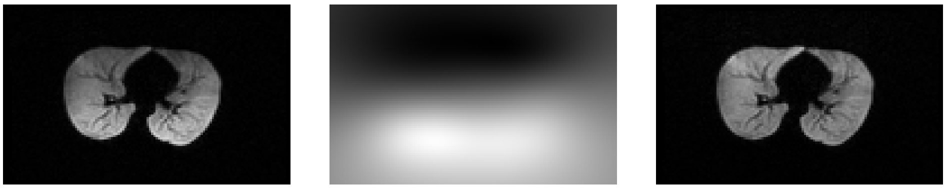
(b) Axial

Figure 2.7: Cross-sections of an MR image with pronounced intensity inhomogeneity. The effect can be most easily noticed on the subcutaneous fat (bright white) in both cross-sections.

This artifact is quite detrimental to automation of segmentation procedures for MRI. This effect needs to be either eliminated before the main segmentation step, or taken into account by the segmentation algorithm. Most approaches for elimination of this artifact model the inhomogeneity as an additive field, so it is also known as a bias field (Figure 2.8 on the next page).

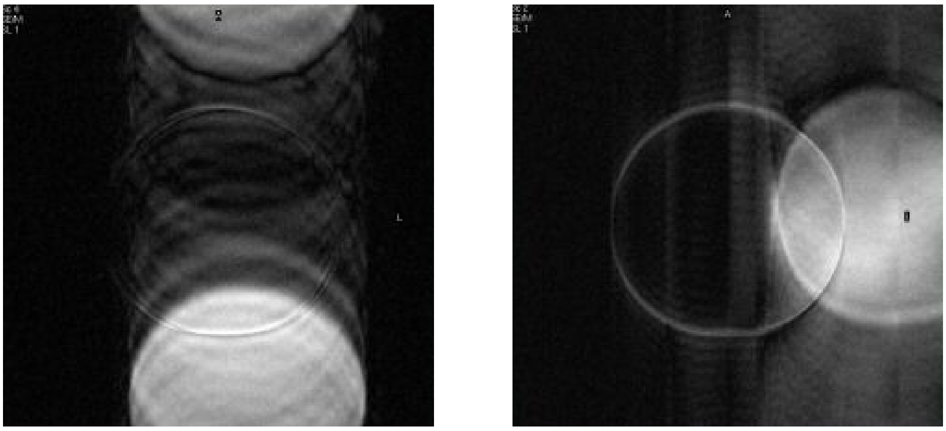
Motion Artifacts One of the assumptions utilized in acquiring MR images is that there is no patient movement during the whole scan. Any violations lead to blurring and anatomical structures overlapping each other.

Sampling along the frequency encoding direction is done in milliseconds and sampling along the phase encoding direction in seconds (by repeating the pulse sequence). Because of this, motion of the patient affects



(a) Uncorrected image **(b)** Estimated bias field **(c)** Corrected image

Figure 2.8: Axial cross-section of the lungs.



(a) Motion in the phase encoding (up-down) direction

(b) Motion in the frequency encoding (left-right) direction

Figure 2.9: Motion artifact of a bottle phantom. Reproduced from [Hig13] with permission.

phase encoding direction much more, regardless of the direction of motion (Fig. 2.9).

Wrap-Around The Wrap-around artifact, also known as aliasing, occurs when tissues outside of the field of view (FOV) are excited. They get interpreted as being on the opposite side of the image, "wrapping around" (Fig. 2.10 on the next page).

Chemical Shift Artifacts Protons in fat resonate at a slightly lower frequency than protons in water [Rad13a]. This causes a positional shift in the frequency encoding direction (Fig. 2.11 on the following page) for spin echo sequences. Slice selection direction is affected as well, but since routine

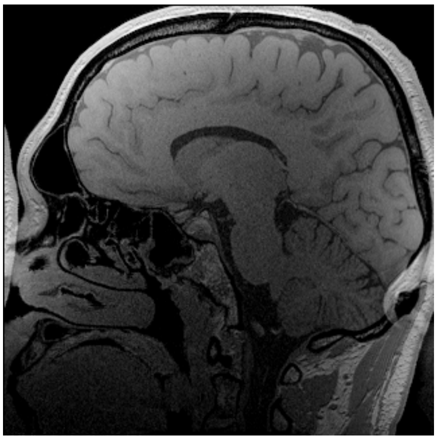
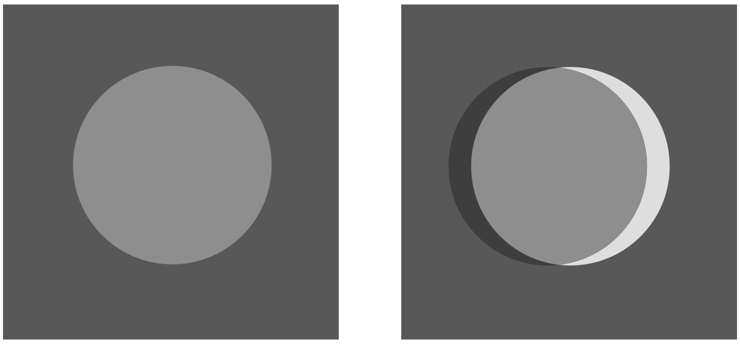


Figure 2.10: Example of a wrap-around artifact. The nose and back of the head were outside of the field of view.



(a) Ground truth position of a fatty object

(b) A fatty object appears shifted in the frequency encoding direction

Figure 2.11: Chemical shift artifact in spin-echo images

images are acquired with quite thick slices this is much less noticeable. In gradient echo sequences black lines appear around fat tissue (Fig. 2.12).

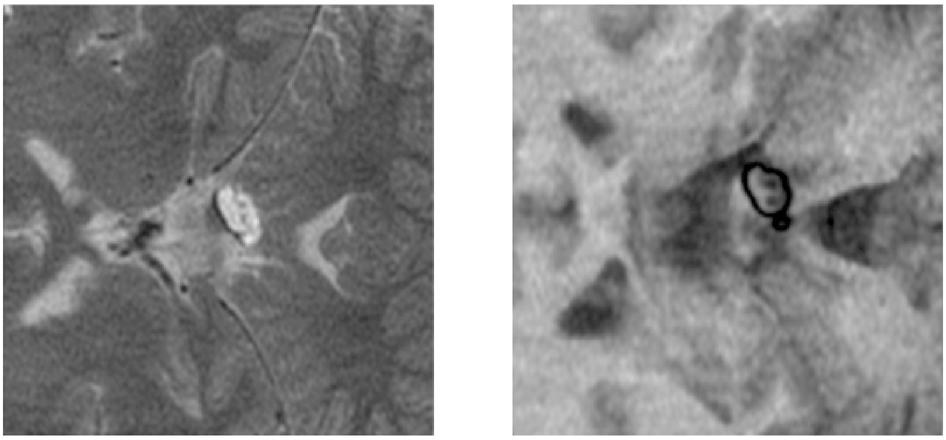

(a) A T_2-weighted spin-echo (b) A gradient echo

Figure 2.12: Chemical shift in a real image (slices through the brain). The lipoma is near the center. Images from Radiopaedia [Rad13b].

Other Imaging Artifacts There are other artifacts arising in the MR imaging process. Every imperfection in the imaging and reconstruction process causes some kind of artifact.

Radio-frequency signals coming from outside the scanner room when the door is opened can cause a zipper artifact (bands of electronic noise appear in images perpendicular to frequency encoding direction).

Fourier transform causes artifacts around sharp edges, due to finite number of frequencies used.

In Fig. 2.13 on the following page a magnetic susceptibility artifact is shown.

Other Properties Noise is caused by various hardware imperfections and limitations, as well as by image reconstruction software approximations. Truncation to finite duration of the exciting sinc signal (Fig. 2.2 on page 9), random fluctuations in the receiving coil electronics [LLT03], Brownian motion of spins, and many other effects contribute to the total noise in the image.

Radiologists look at 2D slices to establish a diagnosis and 3D information is used for overview of the anatomy, usually for radiotherapy or surgical

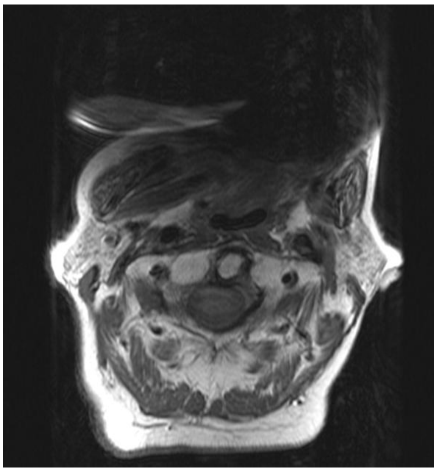


Figure 2.13: A magnetic susceptibility artifact caused by an implanted magnetic dental retention system. From Radiopaedia.

intervention planning. Lower number of slices means faster acquisition times and lower noise[1]. Coupled with lower storage requirements, this has led to usage of anisotropic images in routine clinical practice.

Anisotropy means that voxel's size is not equal along all three axes, usually the voxels a larger along the inter-slice axis. In other words, voxels are longer along inter-slice (Z) axis than along frequency encoding (X) axis and phase encoding (Y) axis. The voxel's dimensions are also called voxel spacings, meaning distances between voxel centers. For routine clinical acquisitions, the typical in-slice (also called in-plane) voxel dimensions range from 0.5 millimeters to 1.2 mm. The typical inter-slice distance ranges between 3 and 4.4 mm, giving voxels an anisotropy factor of about 5-6.

Tissue relaxation times T_1 and T_2 cannot be measured independently. There is also a great number of MRI pulse sequences, serving a multitude of purposes: basic spin-echo, inversion recovery, gradient echo, fat suppression, etc. Each sequence gives a different appearance (intensity) to tissues. This gives magnetic resonance imaging a large versatility, however it has prevented adoption of physical units.

In contrast to unit-less MRI, X-ray computed tomography (CT) has a well-defined unit, called Hounsfield Unit (HU). HU is a linear scale defined by two points: radio-density of air is -1000 HU and radio-density of distilled

[1] Signal to noise ratio increases with the increase of individual voxel's volume, because there are more protons in each voxel which give off measurable signals.

water is 0 HU at standard temperature and pressure. Most tissues are in the range of -100 HU to 100 HU, and bone 700 HU to 3000 HU.

Even though it would be logical to use milliseconds as units at least for T_1-weighted and T_2-weighted spin-echo images, this was not standardized. The absence of units in MRI makes it harder to automate analysis.

2.3 Segmentation Approaches

Image segmentation methods can be classified based on different criteria: supervised or unsupervised; automatic, semi-automatic or interactive; working in image space or physical space; parallel or single threaded; based on a model of a structure of interest; using multi-resolution optimization to reduce execution time; etc. Whichever criteria is chosen, there are usually some methods which are not clearly cut into one of the categories. It would go beyond the scope of this thesis to discuss all existing segmentation methods. The reader is referred to [SS01] or [PXP00] for a more detailed overview. Only the most common segmentation methods and the ones related to this thesis are going to be discussed.

2D images existed before 3D images, and it was natural that 2D measures and segmentation methods were applied to 3D images. The simplest one is picking a single best slice for the task, such as the mid-sagittal slice for spine health assessment or a slice going through the middle of a brain tumor.

Going from 2D to 3D images also spurred slice by slice processing known as 2.5D. It is is simpler than full 3D but does not utilize all the available spatial information.

Most approaches can be implemented both using voxel-based (e.g. binary masks or level sets [OS88]) and parametric (e.g. polygonal surface) representations of organs.

Thresholding is a very basic method. The simplest form is to declare all voxels with intensity above a certain threshold to belong to the object of interest. Thresholding works better for CT images, e.g. to extract bone. Thresholding by itself is usually not sufficient, as shown in chapter 3 on page 39. One popular thresholding method is that of Otsu [Ots79], where the threshold is selected using gray-level histogram to maximize separability between the resulting classes.

Region Growing starts with seeds and adds other pixels to these regions according to some rules – usually how similar is the pixel to the pixels already in the region. There are many variants of this approach, one of the oldest is from Adams and Bischof [AB94]. Hoad and Martel [HM02] use thresholded region growing for segmentation of vertebrae in MR images.

Watersheds derive regions by considering gradient magnitude as a heightmap, each catchment basin corresponding to a region. This method frequently suffers from over-segmentation (producing too many small segments), thus additional processing is needed. The method was first introduced by Beucher and Lantuéjoul [BL79]. Grau et al. [GMA$^+$04] is an improved version applied to medical images.

Graph-Cut family of methods represent voxels as nodes of a graph. It assigns links between nodes based on voxel adjacency, weight of the links corresponding to similarity of connected voxels. Minimum cut optimization is employed to split an image into a background and a foreground. This class of methods gained popularity after Boykov and Kolmogorov [BK04] introduced a polynomial time algorithm for minimum cut. Many authors applied this to medical images, one example being Egger et al. [EBK$^+$10].

Deformable Models are model-based and physically motivated methods that use 2D curves or 3D surfaces which are deformed by internal and external forces. Internal forces maintain smoothness and model-derived properties of the surface. External forces pull the surface towards image features, such as edges. Kass et al. [KWT88] is an early example. The method described in this thesis uses a deformable model with parametric surface representation.

Atlas-Based approach, also known as **segmentation by registration** was first introduced by Broit [Bro81]. This method revolves around an atlas image. It is an image with all voxels in it given an organ label, usually manually. The atlas image is registered with the image under observation. Then, organ labels from the atlas image are carried over to the image under observation, making it fully labeled as well.

This works well if the anatomy of the subjects is similar. To account for greater variability of subjects, multiple atlas images may be employed. Then the image under observation is registered with the most similar image.

Alternatively it is registered to all the atlas images and labels are carried over with different weight depending on registration difficulty.

The atlas-based methods are very resistant to noise. However, they lack the flexibility in locally tuning the segmentation boundary. Additionally, as they are heavily dependent on registration, all drawbacks of the registration method used are present in the overall method, commonly including long running times. Davatzikos et al. [DLSH02] use this approach.

Shape-Based methods learn shape or appearance of the organ of interest from co-registered examples. The learned shape model is imposed as a constraint during segmentation. Cootes et al.'s active shape models [CTCG95] allow deformation of the segmentation curve or surface, but only in ways which are characteristic for the class of shapes they represent. Same group introduced active appearance models [CET01], which add learning of textures to the models. Neubert et al. [NFE+12] applied this approach to vertebral column segmentation.

Interactive methods approach the problem from the side of manual segmentation, keeping the general interaction but trying to cut down segmentation time. They are usually built around quickly-running example driven segmentation algorithms. The user provides some positive and negative samples, the algorithm segments the image and the user reviews the result. In areas where the resulting boundaries are wrong, the user gives new samples which are added to the initialization set and the algorithm is run again. This loop is repeated until satisfactory results are obtained. Boykov and Jolly [BJ01] apply the interactive approach to N-dimensional images, and also show examples of medical CT and MRI images.

2.4 Measuring Precision

Dice similarity coefficient (DSC) [TG09, ZWB+04] and surface distance error [CRS98,NFE+12] are the most commonly used measures for correctness of segmentation in medical imaging.

The Dice similarity coefficient is the relative volume overlap between A and R, where A and R are the binary masks from the automatic (A) and the reference (R) segmentation. V(•) is the volume (in cm^3 or mm^3) of voxels inside the binary mask, by means of counting the number of voxels, then multiplying with the voxel size:

$$DSC = \frac{2 \cdot V(A \cap R)}{V(A) + V(R)}$$

Surface distance error is determined by considering points on the segmentation surface A and calculating minimum distance to the reference surface R. The mean μ of all those distances is usually reported (e.g. [NFE+12]) as the most useful measure, but other measures such as root mean square (RMS) and symmetric maximum distance (Hausdorff distance) also exist.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} d(p_i, R), \text{ where } d(p_i, R) = \min d(p_i, p_k) \text{ for } \forall p_k \in R$$

For detection accuracy, sensitivity and specificity can be used, as exemplified by Kelm et al. [KWZ+13]. Sensitivity, or true positive rate (TPR) and specificity, or true negative rate (TNR) are defined in relation to true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

$$\text{Sensitivity} = TPR = \frac{TP}{P} = \frac{TP}{TP + FP}$$

$$\text{Specificity} = TNR = \frac{TN}{N} = \frac{TN}{TN + FN}$$

2.5 Low-High Values

3D images are frequently visualized using volume rendering [DCH88] (Fig. 2.14 on the next page). However, a transfer function needs to be specified. 1D transfer function maps image values into visual properties (color and opacity). 2D transfer function maps a combination of image value and a derived attribute into a visual property. Higher dimensional transfer functions are possible, but unwieldy to specify.

Most volume rendering software uses 1D, and some 2D transfer functions. Gradient magnitude is usually used for the second dimension (the derived attribute) of the transfer function. The reason for this choice is that high gradient magnitude signifies a boundary of some kind, and boundaries are significant for proper content understanding in volume rendering.

If the image contains a small number of tissues, the 2D histogram on which the transfer function is specified has an arch structure (Fig. 2.15 on the facing page). Arches form between spatially adjacent tissues. Arches can overlap even if the tissues are not spatially adjacent (tissues F_1 and

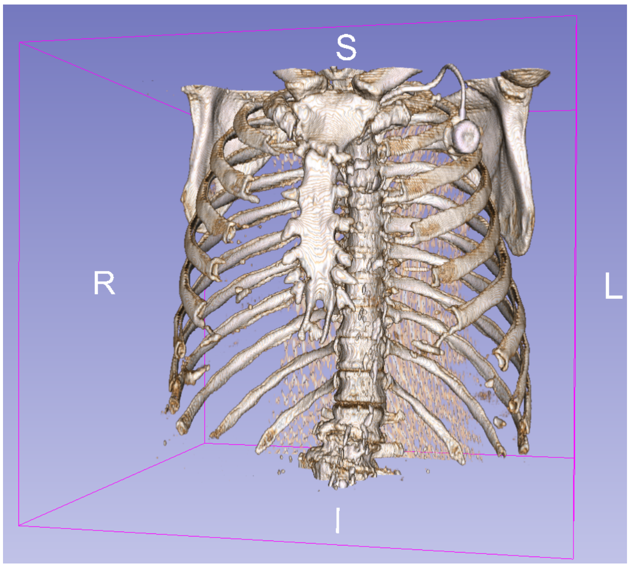


Figure 2.14: Volume rendered CT chest. Transfer function is aimed at showing bones.

F_2 in Fig. 2.15). It is therefore impossible to visually separate them if the standard intensity-gradient 2D transfer function is used.

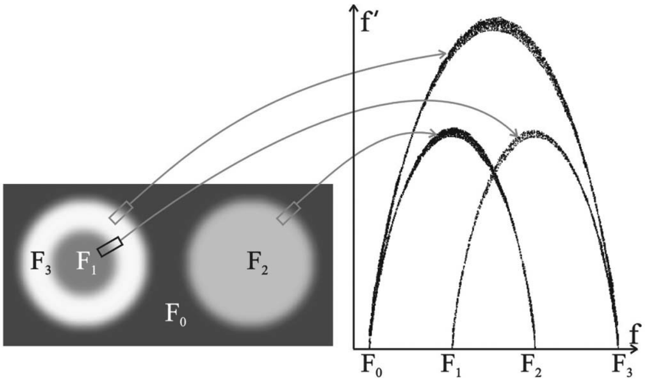


Figure 2.15: An artificial image containing 4 "tissues" with intensities F_i on the left, and the corresponding 2D histogram on the right. X axis is image intensity, Y axis is gradient magnitude. © 2006 IEEE. Reused with permission.

In order to combat that, Šereda et al. [ŠBSG06] introduced LH (low-high) values. Difference between low and high values indicates proximity to a boundary. In order to calculate them, vector gradient field is required. For a given starting position X_S, gradient field is traced to a local maximum in order to find the high value, and to a local minimum for the low value (Fig. 2.16).

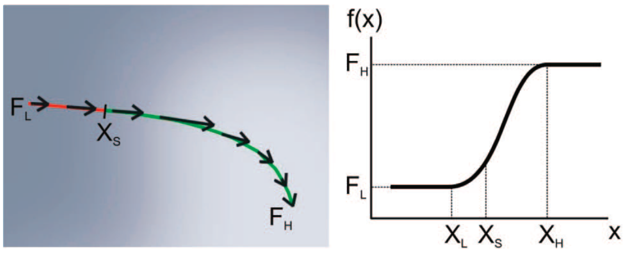


Figure 2.16: Calculation of L and H values for a voxel at starting position S. On the left, a path across a smooth boundary following gradient field. On the right, its intensity profile. X–position, F–image intensity. © 2006 IEEE. Reused with permission.

Figure 2.16 depicts a path which ends in large areas of constant intensity on both ends. In realistic images, that is rarely the case. All combinations of cases from Fig. 2.17 occur. In order to keep the computational complexity reasonable, a local extremum is declared to be found once the gradient magnitude of the examined point is below some threshold ϵ.

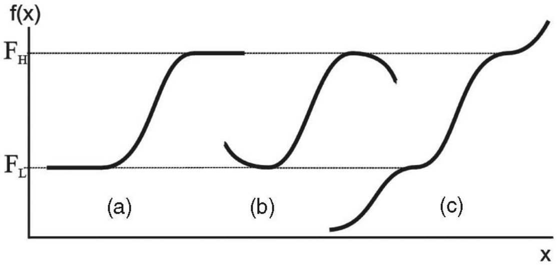


Figure 2.17: Different types of intensity profiles. Gradient tracing can end up in (a) large constant areas, (b) local extrema and (c) inflexion points. © 2006 IEEE. Reused with permission.

Šereda et al. [ŠBSG06] used LH values primarily for visualization, but they also showed its usefulness with region-growing segmentation. In this thesis, LH values help estimate boundaries between vertebral body and its surroundings (Fig. 2.18 on the next page).

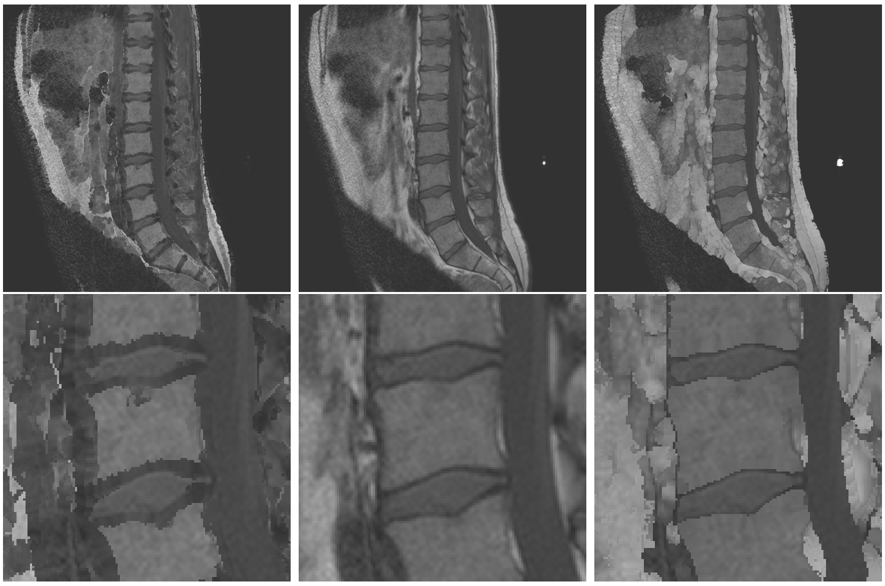


Figure 2.18: Left: low image. Middle: original slice. Right: high image. Bottom row: closeups. Cross-sections from a healthy volunteer (dataset DzZ_T1 from Tab. 7.1 on page 102).

2.6 Viola-Jones Object Detector

In 2001, Paul Viola and Micheal Jones introduced an efficient method for detection of objects in images [VJ01]. Their use-case was detection of human faces in natural images. The method uses a classifier cascade trained on positive examples (objects of interest, e.g. faces or vertebral bodies) and negative examples (background). The classifier cascade is evaluated for all the rectangles of various scales in an image.

Evaluation is done using a sliding window starting at top-left corner, which is moved right by one or more pixels in each step. Once the right edge of the image is reached, the sliding window is moved to the left edge and one or more pixels down. The sliding is finished once the bottom-right corner is reached. The window is evaluated once per step. Fig. 2.19 on the following page shows a single scale pass. Once a single scale detection is finished, the scale (size of the sliding window) is increased by some factor e.g. 1.25 (25%). After detection at all the scales is finished, the overlapping detections are grouped and the largest detection is kept.

The method is efficient because the classifier cascade is designed to quickly reject background windows (rectangles), and focus processing power only on promising candidate rectangles. The first classifier is very

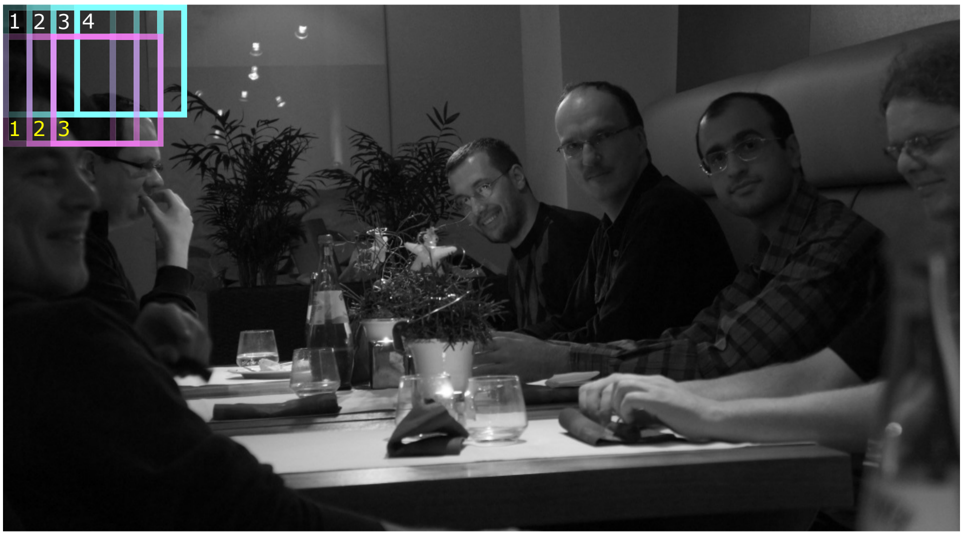


Figure 2.19: Sliding window is used to evaluate the cascade classifier at all positions in the test image.

simple and thus quick to evaluate, but rejects a large number of candidate (e.g. 50%) rectangles from further processing. The second classifier in the cascade only processes the candidate rectangles which the first classifier did not reject (Fig. 2.20), and is also trained only on the rectangles which the first classifier did not reject. When the last classifier in the cascade does not reject a rectangle, that rectangle is considered a positive detection (an object of interest). Each classifier must have a false negative rate close to zero, i.e. it must not reject a rectangle if it contains an object of interest.

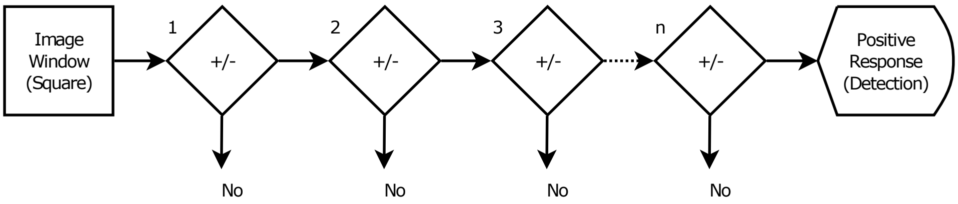
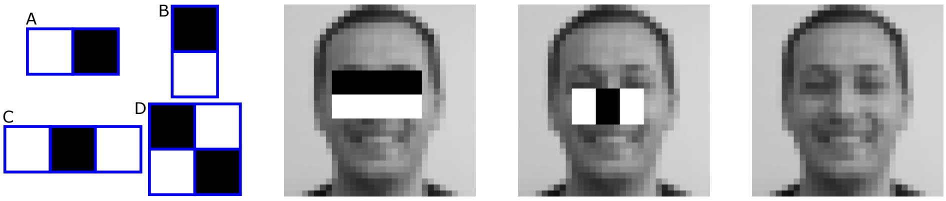


Figure 2.20: A cascade of simple classifiers is used to detect whether the rectangle being evaluated contains a face.

In order to build classifiers for the detection cascade, Viola-Jones detector uses combinations of simple features 2.21 on the facing page. The detectors which are in the later stages of the cascade progressively contain more features, and are thus more costly to evaluate.

Each feature is efficient to evaluate at any scale, because feature response is the sum of pixels under the black part of the feature minus the sum of



(a) Feature examples **(b)** Face with B **(c)** Face with C **(d)** Face 32×32

Figure 2.21: Basic features used in classifiers by the Viola-Jones algorithm (a) and their typical correspondence with features (b, c) of an upright frontal face (d).

pixels under the white part of the feature. Those sums can be calculated efficiently using summed area table. Summed area table has the same number of cells as the image has pixels. Each cell (i,j) contains the sum of pixels in the rectangle (0,0)-(i,j). Cell 1 contains the sum of all pixels in rectangle A (Fig. 2.22), cell 2 the sum of all pixels in rectangles A and B, cell 3 contains the sum of all pixels in rectangles A and C. Cell 4 contains the sum of all 4 rectangles. Thus the sum of pixels in rectangle D is 1+4-(2+3). Summed area table can be calculated from the image in a single pass.

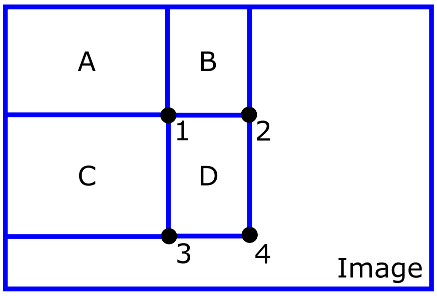


Figure 2.22: Summed area table enables calculation of the sum of any rectangle within the image with 4 array references and simple operations.

2.7 Medical Background for Brain Tumors

Gliomas are the most common primary brain tumors, whereof 70% are among the group of malignant gliomas (anaplastic astrocytoma World Health Organization (WHO) grade III, glioblastoma multiforme WHO grade IV) [KLS+02]. The glioblastoma multiforme (WHO IV) is one of the highest malignant human neoplasms. Due to the biological behavior, gliomas of WHO grades II to IV can not be cured with surgery alone. The multimodal therapeutical concept involves maximum safe resection followed by radiation and chemotherapy, depending on the patient's Karnofsky scale[2]. The survival rate still only accounts approximately 15 months [KJW$^+$03], despite new technical and medical accomplishments such as multimodal navigation during microsurgery, stereotactic radiation or the implementation of alkylating substances. Although there is still a lack of Class I evidence, literature today favors a maximum extent of tumor removal in both low grade and high grade gliomas as a positive predictor for longer patient survival [LASF$^+$01].

Approximately 10-15% of all intracranial neoplasms are sellar tumors. The most common sellar lesion is the pituitary adenoma [AE98, WZP$^+$07]. The lesions can be classified according to the size or the hormone secretion (hormone-active and hormone-inactive). Microadenomas are less than 1 cm in diameter, whereas macroadenomas measure more than 1 cm. The rare giant-adenomas have more than 4 cm in diameter.

Secreted hormones can be cortisol (Cushing's disease), human growth hormone (hGH; acromegaly), follicle stimulating hormone (FSH), luteinising hormone (LH), thyroid-stimulating hormone (TSH), prolactine, or a combination of these. Only for the prolactine-expressing tumors, a pharmacological treatment is the initial treatment of choice in form of dopamine agonists. Treatment is most commonly followed by a decrease of prolactine levels and tumor volume. For acromegaly and Cushing's disease, surgery remains the first-line treatment, although somatostatin receptor analogues or combined dopamine/somatostatin receptor analogues are a useful second-line therapeutical option for hGH-expressing tumors. Current medical therapies for Cushing's disease primarily focus on the adrenal blockade of cortisol production, although pasireotide and cabergoline show promise as pituitary-directed medical therapy for Cushing's disease.

Thus, not only for the most hormone-active, but also for hormone-inactive macroadenomas with mass-effect, surgery is the treatment of choice, most possibly via a transsphenoidal approach [BS09]. For hormone-inactive mircroadenomas (<1cm) there is no need for immediate surgical resection. The follow-up contains endocrine and ophthalmological evaluation as well as magnetic resonance imaging (MRI). In case of a continuous tumor volume progress, microsurgical excision is the treatment of choice. Thus, the tumor volume should be tracked over the time of the follow-up. Volumetric assessment of a tumor with manual segmentation of its outlines is a time-

[2] Karnofsky scale measures health on a 0-100% scale. 0 is death, 100 is "perfect" health, with standard 10-point steps in-between

consuming process that can be overcome with the help of computer-assisted segmentation methods.

The requirements for tumor segmentation can be roughly split into two differing groups: treatment planning and progress tracking. Treatment with radiation therapy requires precise tumor boundaries. The same usually holds for surgery planning, because blood vessels need to be avoided. In both these cases more time and attention can be devoted to the segmentation process in order to ensure high precision.

Tracking the tumor's progress over time is usually conducted in order to determine whether it is growing, stagnating, or perhaps receding. This information, combined with tumor volume, is used to decide whether to treat the tumor or wait. For this purpose, ease and speed of the segmentation process overshadow the segmentation precision.

2.8 Related Work for Tumors in MRI

3D segmentation of brain tumors has a long history, and a large number of published work deals with it. The arsenal of tools is quite diverse: from crude geometric models to complex computational methods.

Geometric models use one or several user-defined diameters – which can be manually achieved very quickly – to calculate the tumor volume. Briefly, according to the spherical model, the volume is defined as $1/6\ \pi \mathrm{d}^3$ (d is the diameter of the maximum cross-sectional area) and the ellipsoid model defines volume as $1/6\ \pi$ abc (a, b, c represent diameters in the three axes of the tumor) [ISKF+09].

Gibbs et al. [GBBH96] presented a combination of region growing and morphological edge detection for segmentation of enhancing tumors in T_1 weighted MRI data. Starting with a manually provided first sample of tumor signal and surrounding tissue, an initial segmentation using pixel thresholding, morphological opening and closing and fitting to an edge map is performed. Gibbs et al. have evaluated their procedure with one phantom dataset and ten clinical datasets. The mean segmentation time for a tumor was about ten minutes, and they did not exactly classify the tumors they used for their evaluation.

An interactive method for segmenting full-enhancing, ring-enhancing and non-enhancing tumors has been presented by Letteboer et al. [LOD+04]. They evaluated their approach with twenty clinical cases. Based on a manual tracing of an initial slice, morphological filter operations are applied to the MRI volume to divide the data in homogenous regions.

A deformable model depending on intensity-based pixel probabilities for tumoral tissue has been introduced by Droske et al. [DMRS05]. They used a level set formulation, in order to split the MRI data into regions of similar image properties for tumor segmentation. The method was then performed on image data of twelve patients.

Clark et al. [CHG$^+$98] proposed a knowledge-based automated segmentation on multispectral data to partition glioblastomas. After a training phase with fuzzy C-means classification, clustering analysis and a brain mask computation, an initial tumor segmentation derived from vectorial histogram thresholding is post-processed to eliminate non-tumor pixels. The introduced system has been trained on three MRI volume datasets and has been tested on thirteen unseen volume datasets.

A segmentation based on outlier detection in T_2 weighted MRI data has been developed by Prastawa et al. [PBHG04]. In order to detect abnormal tumor regions, the image data is registered on a normal brain atlas. Then, tumor and edema are isolated by statistical clustering of the differing voxels and a deformable model. However, they have applied the method only to three real datasets. For each case, the required time for automatic segmentation was about 90 minutes.

Sieg et al. [SHP01] have proposed an approach for segmenting contrast-enhanced, intra-cranial tumors and anatomical structures of registered multisprectral MRI data. Multilayer feed-forward neural networks with back-propagation are trained and a pixel-oriented classification is applied. The approach has been tested on 22 datasets, but the authors did not provide any computational time.

Egger et al. [EBK$^+$10, EBK$^+$11c, EBK$^+$11b] have introduced a segmentation scheme for spherical objects that creates a directed 3D graph by sending rays through the surface points of a polyhedron and sampling the graph's nodes along every ray. Thereafter, the minimal cost closed set on the graph is computed via a polynomial time s-t cut [BK04], creating a segmentation of the tumor non-iteratively. The center of the polyhedron is defined by the user and located inside the tumor.

Neubauer et al. [NWF$^+$05] and Wolfsberger et al. [WNB$^+$06] introduce STEPS, a virtual endoscopy system designed to aid surgery of pituitary tumors relying on both CT and MR images. STEPS uses a semi-automatic segmentation method that is based on the so-called watershed-from-markers technique. The watershed-from-markers technique uses manually defined markers in the object of interest and the background. A memory efficient and fast implementation of the watershed-from-markers algorithm – also extended to 3D – has been developed by Felkel et al. [FBW01].

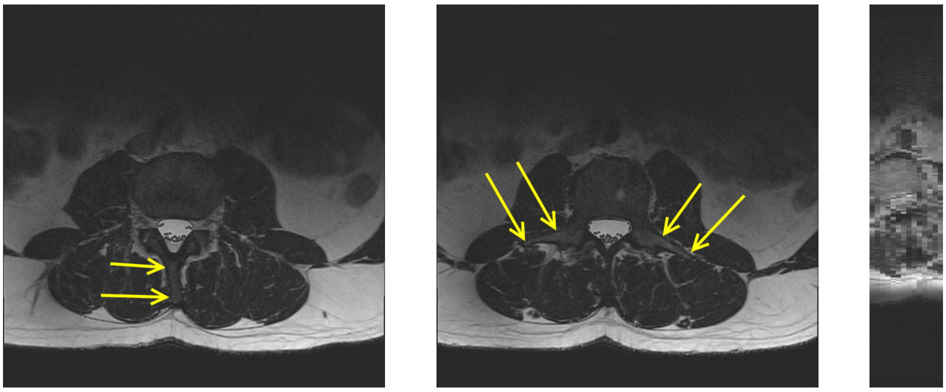
For a comprehensive overview of some deterministic and statistical approaches see the review of Angelini et al. [ACM+07].

2.9 Motivation for Spine Segmentation

Lower back pain for adults is rather common and its prevalence is rising [FHA+09]. The most common causes involve spinal cord, such as narrowing (stenosis) of the spinal canal, and that has been well investigated [HRSE+00]. For diseases involving the vertebrae, 2D X-ray is frequently used for screening, but the effective diagnosis is made based on Computed Tomography (CT) or Magnetic Resonance Image (MRI). The cancer risk from radiation exposure in CT imaging makes MRI preferable in the clinical routine [RGMB10]. Furthermore, 2D X-ray and CT cannot reveal all pathologies [APR+85] due to low sensitivity of soft tissues. Therefore, the usage of MRI is sometimes diagnostically required.

Most 3D segmentation approaches focus on CT datasets only, such as [KOE+09, ML13]. However, these methods rarely transfer to MRI because of additional challenges. Routine MRI has a lower and strongly anisotropic resolution. Unlike CT's Hounsfield unit (HU), MRI does not have standardized units of measurement. In spine CT, bone edges are the only high-intensity edges. In MRI there are strong edges between many tissue types including edges within vertebral bodies, e.g. Fig. 7.6 on page 115. MRI also has a non-homogeneous intensity across the image, e.g. the central region has higher intensity and better contrast than the marginal areas. Lastly, MRI has many different parameters which can be changed to emphasize different tissues, resulting in many different scanning sequences which sometimes produce radically different image intensities. All these facts are detrimental to the automation of segmentation procedures for MRI datasets.

Spinous processes are poorly seen on MR images of low inter-slice resolution, which are the routine in clinical practice. Also, transverse processes are usually not seen at all because they are outside of the routine acquisition volume (Fig. 2.23 on the next page). Therefore, focus is on the segmentation of vertebral bodies instead of the whole vertebrae.



(a) Case_2 slice 100

(b) Case_2 slice 113

(c) Ble s. 200

Figure 2.23: Axial cross-sections of L3 vertebra from a non-routine specialized scan and a routine sagittal scan (see Tab 7.2 on page 103 for details). Spinous process (a) and transverse processes (b) can easily be seen in this non-standard dataset. In the routine dataset (c) the transverse processes are largely outside of acquisition volume. Moreover it is hard to discern anything except the vertebral body.

In this thesis focus is on a segmentation system of practical clinical usefulness. This kind of system is required to work on a large variety of routine clinical datasets containing pathologies. It has to be reasonably fast and should not require cumbersome initialization or other manual intervention. Finally, it must support diagnosis of diseases of interest.

There were plenty of segmentation methods for vertebrae in CT images, that is easier to do due to high intensity boundaries between cortical bone and surrounding soft tissue. At the time when I started this research only two methods for segmenting vertebrae in MRI were published, which provided an incentive to work on this problem.

2.10 Spine Related Work

Much research has already been done on spine segmentation. There is a large number of 2D methods due to its relative simplicity and low computational requirements [MCP+09, SSQW07, HCLN09, PZWhL05, CGBM04, EKD+12]. 2D segmentation approaches are mostly applied to manually identified, best suitable cross-sections.

As 2D approaches process individual slices they can miss important information, such as a curvature or a positional shift in the anatomical left-right direction, so 3D approaches are preferable.

Most 3D approaches focus exclusively on computed tomography (CT) datasets. Only some of the CT-focused prior work is mentioned here.

Besides 2D/3D distinction, i.e. methods which work on only single slices vs. the whole volume, we can also classify methods according to the type of influence a segmentation of one vertebra has on the other vertebrae:

- Independent segmentation of each vertebra or vertebral body. Used by [ŠVPL11,APM+12,WKL+01,HM02,CGBM04,MCP+09]
- Top-down approach, where general spine shape and position is determined first, and then individual vertebrae are segmented. Used by [DLSH02,NFS+11,SSQW07,PZWhL05,YOS06,GS04,KWL+08] and the version presented in chapter 6 on page 83
- Bottom-up approach, where the global spine shape is built up from segmented vertebrae. Besides segmentation system versions presented in chapters 4 on page 55 and 5 on page 75 only [HCLN09], describing a 2D method, partially fits into this category.

2.10.1 Computed Tomography

Weese et al. [WKL+01] use a polygonal vertebra model and manual initialization. They use an iterative deformable surface model with explicit internal and external energy. Internal energy reflects statistical shape and distribution of mesh vertices, and external energy relies on image gradients. In each iteration method performs a surface detection step and a mesh reconfiguration step. The authors report 0.93 mm as the mean segmentation error and 30 seconds execution time.

Yao et al. [YOS06] focus on routine chest and abdominal CT images. The spinal canal is extracted using a watershed algorithm and directed acyclic graph search. The vertebrae are segmented by using a four-part vertebra model. The spinal column was correctly partitioned in 67 out of 69 cases. No execution time is reported.

Ghebreab and Smeulders [GS04] use manual initialization for first vertebra and global spine shape. They use B-spline surfaces with 12x12 control points for surface representation. Statistical spine shape is used for initializing segmentation of an adjacent vertebra. The mean shape of four different lumber vertebrae are independently constructed. The method was tested on 6 CT images, but execution time and precision were not given.

Klinder et al. [KWL$^+$08] initialize the global spine position by an automated rib cage segmentation method. A statistical constellation model for vertebrae is applied on a global scale to obtain an approximate position of individual vertebrae. Local adaptations of each vertebra are similar to [WKL$^+$01]. The method was evaluated on 10 thoracic CT datasets. The segmentation error was 1.0±0.3 ($\mu \pm \sigma$) mm, but execution time was not given.

Klinder et al. further improve their system in [KOE$^+$09]. They used generalized Hough transform (GHT) to bypass a rib cage segmentation and get the vertebral foramen points directly. They also provide vertebra labels. Profound prior knowledge is applied through the use of various kinds of models covering shape, gradient, and appearance information. They tested this method on 64 CT images, most of them including pathologies. In 56 cases the vertebrae were labeled correctly resulting in a final mean point-to-surface segmentation error of 1.12±1.04 mm. They aimed for maximum reliability, so their overall computation time was about half an hour.

Ma and Lu's method [ML13] is based on learned bone-structure edge detectors and a coarse-to-fine deformable surface model in order to segment and identify vertebrae in 3D CT thoracic images. They employ learned statistical shape models for 12 thoracic vertebrae. This algorithm performs successfully with reliable mean point-to-surface errors 0.95±0.91 mm on 40 volumes. They also applied vertebra labeling with 73% success rate. They do not state execution time.

3D approaches mostly rely on user initialization. However, to extract the approximate spine position, Yao et al. [YOS06] use Hounsfield values and Klinder et al. [KOE$^+$09] have a completely automatic method. Ma and Lu [ML13] can use either completely manual or completely automatic method.

2.10.2 Magnetic Resonance Imaging

At the time of writing this thesis only five prior fully 3D segmentation methods are available for MRI: Hoad and Martel [HM02], Davatzikos et al. [DLSH02], Štern et al. [ŠLPV11], Ayed et al. [APM$^+$12], and Neubert et al. [NFS$^+$11, NFE$^+$12]. Some further methods are concerned with detection only [RET13, KWZ$^+$13, SMB11] and could be plugged into the system described in chapter 6 on page 83 replacing the initial vertebra detection.

Kelm et al. [KWZ+13] optionally do segmentation of intervertebral disks, but report neither precision nor running time.

Hoad and Martel [HM02] have developed a segmentation algorithm that combines thresholded region-growing with morphological filtering and masking using predefined shapes. Their algorithm is manually initialized by one or two ellipses per vertebra, and can be split into three steps: segmentation of the vertebral bodies, segmentation of the posterior structures, and lastly manual corrections. Their datasets have isotropic voxels ($1 \times 1 \times 1$ mm^3), and they combined two images of a double echo FISP acquisition sequence. Overall, it is a method suited to assist spine surgery, using images quite different from routinely acquired ones. They tested their method on 30 vertebrae. The surface registration error for vertebral bodies was 1.25mm±0.28mm. They also calculated average percentage of "good" points to be 79.4%, with 3 rejected registrations out of 30 experiments. Running time of their algorithmic part was 5-10 minutes (not counting the manual steps).

Davatzikos et al. [DLSH02], with a long line of research in hierarchical deformable models [SHD01], were mainly interested in the registration of different spine images to a manually segmented template image. An isosurface was extracted from this template image resulting in 837 vertices (vertebral bodies L1 to S1 and a corresponding portion of the spinal canal). This deformable model was trained on 13 additional images. In order to determine the transformation which registers a test image with the template image, the surface model is manually placed in the test image overlapping the true position of the spine segment in the test image, and hierarchically deformed to conform to the edges of the test image. The transformation between the stereotaxic space of the template image and space of the test image is used to determine the correlation between the patient symptoms and lesions visible in the image. The evaluation was done using the leave-one-out method on routine images ($0.93 \times 0.93 \times 3$ mm^3) of healthy volunteers. The average overlap was 81.5%±3.6%. They require only one initialization per dataset but do not state the execution time.

Štern et al. [ŠLPV11] perform the segmentation by optimizing 29 parameters of a 3D deterministic model of the vertebral body. They maximized dissimilarity between inside and outside intensities, and steered their method by image gradients. The method is initialized with one point per vertebra and an accompanying size, depending on the anatomical position, i.e. upper thoracic, lower thoracic or lumbar. They evaluated their method on 75 vertebral bodies from nine T_2-weighted images. Three of their images were of a routine type ($0.4 \times 0.4 \times 3$ mm^3), the others were isotropic ($1 \times 1 \times 1$ mm^3). The mean radial Euclidean distance between the segmentation surface and ground truth points was 1.85mm±0.47mm. Performing the algorithm takes 1-15 minutes per vertebra.

Ayed et al. [APM$^+$12] formulate segmentation as a distribution-matching problem. They split it into a series of sub-problems, each of which can be solved via a convex relaxation and the augmented Lagrangian method. This results in a parallel method, and they implement CPU and GPU variants. They used T_2-weighted MR images of the lumbar spine. They evaluated their results only on 2D mid-sagittal slices, where they reach 85% DSC – a precision which corresponds to 78% for the 3D case. The GPU variant runs for 3 minutes, and the CPU variant for 75 minutes.

Neubert et al. [NFS$^+$11, NFE$^+$12] localize 3D spine curve and extract approximate positions of vertebral bodies using active rectangles. This serves as initialization for segmentation using active shape models with shape deformation using gray level models. Vertebral bodies and intervertebral discs are segmented. They use a high resolution SPACE MRI sequence with in-plane pixel size of 0.34×0.34 mm^2 and slice thickness of $1 - 1.2$ mm. They used 14 healthy volunteers with 132 vertebral bodies for quantitative evaluation. They achieve 91% DSC, 0.67mm mean absolute shape distance and 4.08mm Hausdorff distance. They also classified intervertebral discs into degenerate and healthy classes. Their method takes an average of 35 minutes per vertebra [NFE$^+$12], not counting some 3 minutes spent on intensity normalization per dataset. The method is completely automatic. This means that the user has no chance to correct mis-detections, and this is especially important if running time (5h) is taken into account.

Kelm et al. [KWZ$^+$13] present a method for detection and labeling of inter-vertebral disks. They combine marginal space learning (MSL) with a generative anatomical network which incorporates relative pose information for the detection of multiple objects. They present an efficient iterative MSL which enables examination of many candidates by the learned anatomical prior enabling detection of position, orientation and scale with high sensitivity. They trained their method on 42 MR and 30 CT images. Whole spine MR images are processed in 11.5 seconds on a dual-core laptop. Their detection achieves a 98.6% sensitivity.

A representative general segmentation method (not specific to MRI or vertebral column) with open source code is useful for comparison purposes. Coupre et al. [CGNT09] present one such method. They extend a framework for seeded image segmentation, which includes graph cuts, random walker and shortest path optimization. They represent an image as a weighted graph between pixels, and unify the three algorithms with a common energy function. This allows them to include optimal spanning forest for watersheds. They propose a new family of segmentation algorithms which they call power watersheds.

2.11 Fine-Grained System Overview

All four segmentation systems, i.e. the brain tumor segmentation and the three versions of spine segmentation, can roughly fit into the generalized processing workflow shown in Fig. 1.3 on page 4.

From Fig. 3.1 on page 39 it is visible that the tumor segmentation workflow matches the generalized workflow well, with pre-processing replaced by manual initialization and diagnosis by tumor volume calculation. The corresponding chapter 3 on page 39 centers on the introduction of the basic polyhedron inflation algorithm. The algorithm starts with a small triangular polyhedron and expands it by moving the vertices outwards until a boundary is reached.

Segmentation of a series of spinal vertebrae is principally more complicated - the vertebral bodies are treated as a chain of individual objects in chapter 4 on page 55. Segmentation of individual vertebrae is an improved version of the segmentation algorithm from chapter 3. The main improvements come from the usage of Low-High images and constrained subdivision hierarchies.

In order to predict position of the next vertebral body in the chain, vertebra's main axis is used. Face merging is used to reduce vertebral body to a disk-like shape from which "flat" sides are detected thus establishing its "axis". The axis points towards adjacent vertebral bodies. Reliance on this axis frees the user from additional manual initializations.

Chapter 5 on page 75 retains the overall structure. The major improvement is the usage of the multi-feature boundary estimation approach, instead of the binarized boundary which is employed in chapter 4. The minor improvement is the usage of the iterative closest points (ICP) algorithm for axis determination.

The final version of the segmentation system is described in chapter 6 on page 83. The major improvement is the detection of all vertebrae using the Viola-Jones algorithm before the actual segmentation is started, which enables parallel feature calculation and vertebral body segmentation. Finally, the detection of all vertebrae makes the diagnostic analysis at the end reliable.

Chapter 3

Segmentation of Brain Tumors

As the spine consists of multiple vertebrae (a constellation of objects), it is not easy to segment it. Therefore the single-object segmentation system was first realized and evaluated for the case of brain tumors.

This chapter introduces a simple segmentation system using balloon inflation. It is tested on two types of brain tumors. The results have been published in two stand-alone papers [ZEB$^+$10, ZEB$^+$11] and two comparison papers [EZB$^+$10, EZF$^+$13]. The system introduced in this chapter is expanded and improved upon by chapter 4 on page 55.

The system is initialized by an approximate outline of the tumor. The boundary is implicitly binarized by the inflation rules, based on the minimum and maximum intensity from the initialization step. The segmentation boundary is represented as a triangle mesh and stored using a quad-edge data structure [GS83]. Star-shape and smoothness constraints are maintained during iterative inflation process. After the segmentation is finished, the tumor volume is calculated.

3.1 Outline Initialization

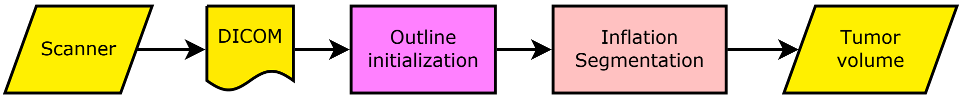


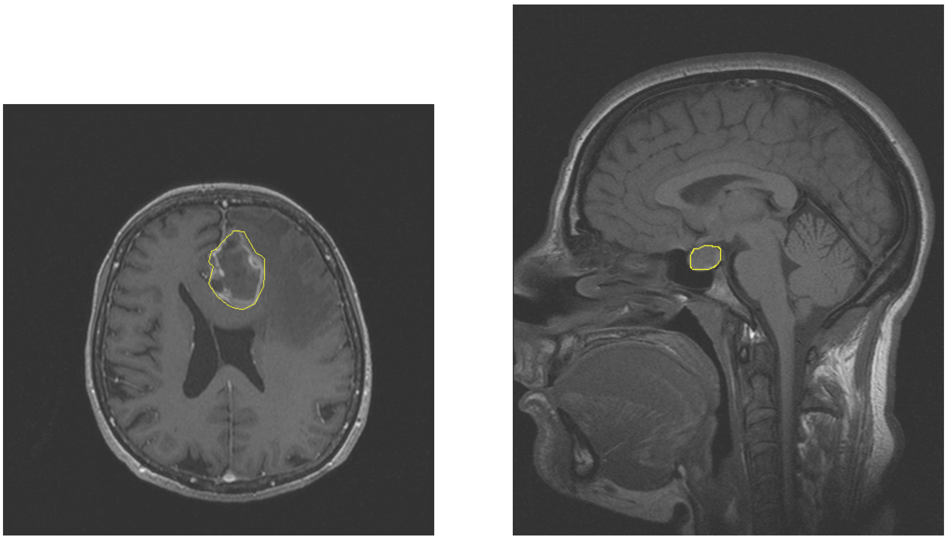
Figure 3.1: Overview of the tumor segmentation system.

The overall system relies on user initialization (Fig. 3.1 on the previous page). The user draws an approximate outline within vertebral body on a slice that is approximately located central to the tumor (Fig. 3.2). From this initialization, the following information is extracted:

1. Estimated center of the tumor. Two coordinates are extracted from the center of the outlined object (center of area), and the third coordinate is the index of the selected slice.

2. By analyzing pixel intensities in the selected slice, the minimum and maximum intensities of voxels of interest is determined. Here, the few highest and lowest percent are ignored in order to account for noise.

3. The average distance from the center to the boundary (radius) is determined. The "radius" is a dimensionally invariant measure – it is the same in 2D (slice) and 3D (whole volume), e.g. the radius of a circle is equal to the radius of the sphere made by rotating that circle around its diameter. Some complications of image anisotropy are avoided by usage of this radius.



(a) Dataset m1_ls (glioma), slice 94

(b) Dataset Ame (adenoma), slice 6

Figure 3.2: User initialization of a tumor boundary (yellow contour).

3.2 Constraint-Enforcing Inflation

The main idea is to start with a small triangular surface mesh in the shape of a convex polyhedron at the approximate center of the tumor. Balloon inflation forces [Coh91] are used to expand this mesh, keeping it approximately star-shaped. The aim is not to inflate beyond the tumor boundary. The overview of the algorithm is shown in Fig. 3.3.

To resolve ambiguities in the data, some constraints are needed. The first one is smoothness of the tumor surface. Soft biological tissues, including tumors, do not have jagged edges.

The second constraint is absence of self-intersections of the tumor surface, because they make no sense physically and are computationally inefficient. This constraint is fulfilled by approximately maintaining an even stricter constraint: star-shape (Fig. 3.8 on page 44). Star-shape is computationally efficient to maintain and it prevents self-intersections, but it also prevents segmentation of all imaginable shapes of tumors. As tumors rarely, if ever, have an exotic topology or geometry (they are mostly shaped similar to a potato) this is not a severe restriction.

Additionally, vertex movability rule is application specific, i.e. different for glioma and adenoma.

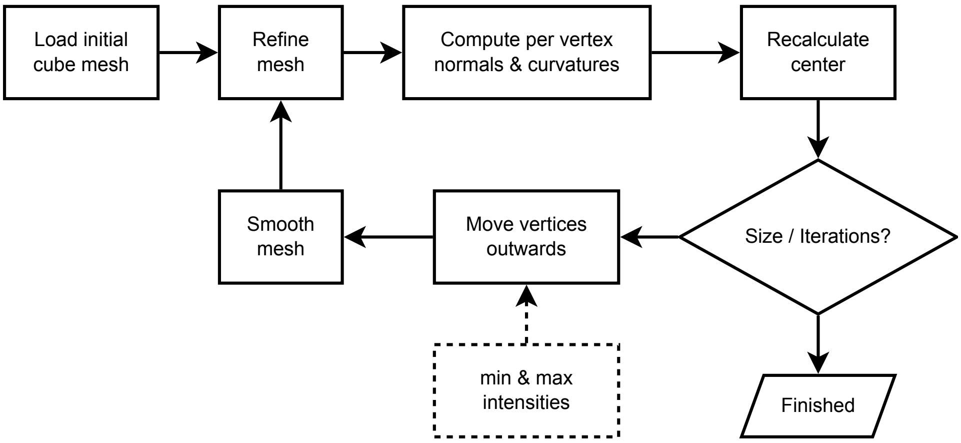
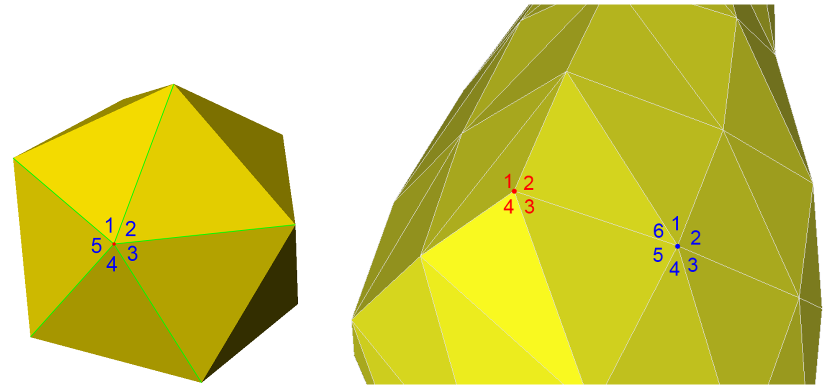


Figure 3.3: Overview of the inflation loop. The dotted block represents image intensity information collected from the user's outline initialization. Cube was chosen as initial mesh due to relative regularity of it subdivision compared to tetrahedron.

Mesh refinement is needed to keep the edge length comparable to the voxel size and to prevent long, thin triangles. This is ensured by splitting the edges which are 3 times longer than the average voxel spacing (geometrical mean of spacing in X, Y and Z direction).

The computation of vertex normals $\vec{n}$ and curvature estimates G (Fig. 3.4) is required before the vertex repositioning step. Immediately afterwards the center of the polyhedron is recalculated (Fig. 3.5). Vertex normals are averages of incident triangle normals weighted by the incident angle: $\vec{n} = \sum_i \vec{n}_i * \alpha_i$



(a) Icosahedron **(b)** Arbitrary shape

Figure 3.4: Angle deficit approximation of Gaussian curvature $G = 2\pi - \sum \alpha_i$. The sum of all incident angles of a vertex is close to 360° in low curvature regions. The angle deficit (sum<360°) signals a protruding vertex.

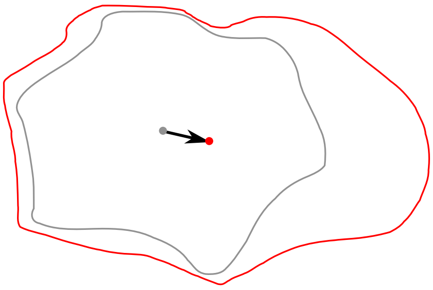
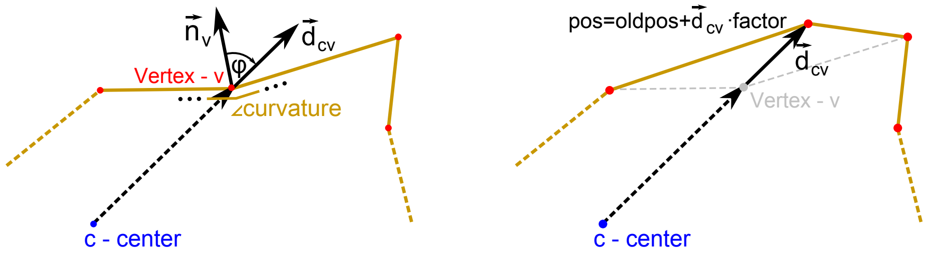


Figure 3.5: Because vertices are usually not evenly inflated, the polyhedron center needs to be recalculated after each iteration.

The segmentation is finished when the segmentation converges to some position through slowdown and the maximum number of iterations is reached or the center-surface distance (radius) reaches the user-initialized "radius".



(a) Before movement

(b) After movement

Figure 3.6: Vertex movement.

The vertex repositioning (inflation) consists of doing the following 3 steps for each vertex:

1. Calculate the cosine of the angle φ between center-vertex vector $\vec{d_{cv}}$ and surface normal vector $\vec{n_v}$ at the given vertex (Fig. 3.6). The greater the angle, the lower the inflation speed (Fig. 3.7 on the following page). While the surface is still nearly spherical, i.e. at the start of the inflation process, the inflation speed is higher and thus the surface reaches the boundary area faster.

2. Calculate the move-speed factor. The higher the curvature, the lower the inflation speed, thus the inflation speed is slowed down for vertices on ridges, valleys and peaks. This approach implicitly stimulates smoothness by taking into account the curvature estimate of the vertices.

3. If a vertex can be moved (depending on image intensity) it is moved in the direction of center-vertex vector, thus maintaining star shape (Fig. 3.8 on the next page). The rules about which vertices can be moved need to be adopted to the specific segmentation task. The vertex movability is described in section 3.2.1 on the following page for the glioma and in section 3.2.2 on page 45 for the adenoma.

   The displacement amount is adjusted by inflation speed factors (Fig. 3.6): $|\vec{d_{cv}}| = S_{xyz} \cdot \cos\varphi \cdot \frac{1}{\max(1,|G|)}$, G - Gaussian curvature and S_{xyz} - average spacing along x,y and z axes.

Smoothing the surface of the polyhedron slightly is required to overcome noisy voxels, which would otherwise prevent inflation of the mesh beyond them, even if they are in the middle of the tumor. We also know that the surface of the tumor is smooth and not jaggy.

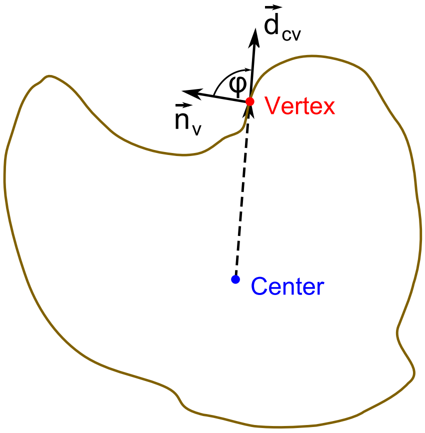


Figure 3.7: If the angle (φ) between the mesh normal at the given vertex ($\vec{n_v}$) and the radial vector ($\vec{d_{cv}}$) is close to 90° it makes little sense to move that vertex. Such a movement would not inflate the mesh, it would only disturb the surface density of the vertices.

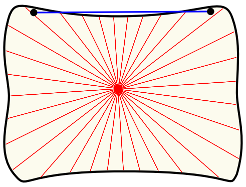


Figure 3.8: Concave but star-shaped (star-convex): there exists a point inside the shape (center) to which all the other points of the shape can be connected by straight line segments lying entirely within the shape.

Convergence condition is approximated by slowed inflation speed combined with a limited number of iterations. Additional stopping safeguard is that the "radius" is smaller than 150% of user initialized radius.

3.2.1 Glioma Movability Rule

Gliomas are usually imaged with contrast agent applied, which enhances the boundary (metabolically active tumor areas are bright). This is exploited by the vertex movability rule used in step 3 of vertex repositioning.

The vertex can be moved if the target position (current position + displacement outwards) has an intensity value within the initialized range of interest (range of values within the user initialized boundary). Additionally, the target intensity has to be higher than 80% of the maximum intensity this vertex has encountered so far. This additional condition prevents inflation once the high-intensity boundary region has been passed.

The "life", i.e. the temporal evolution of a single vertex through the iterations of a glioma segmentation is shown in Fig. 3.9. The vertex is moved outwards by a small amount in every iteration until iteration 9 when it hits a low-intensity voxel. In iterations 10-14 this vertex is pulled forward by neighboring vertices through smoothing. As the vertex is pulled forward, its target position for inflation is also moved, and due to trilinear interpolation the target position's intensity is also increased. Eventually in iteration 15 the vertex continues inflation normally.

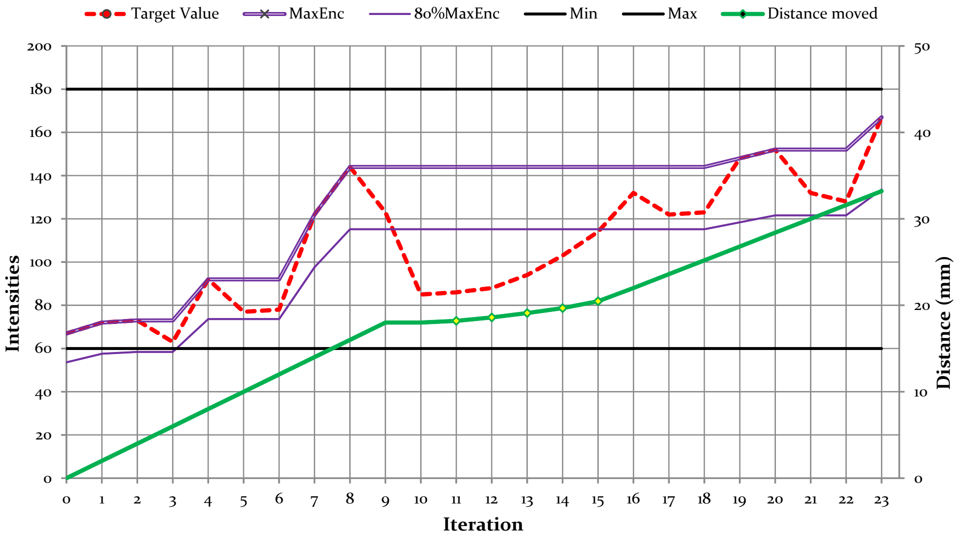


Figure 3.9: Life of a single vertex through the iterations of a glioma segmentation.

3.2.2 Adenoma Movability Rule

Pituitary adenomas are usually imaged without a contrast agent, which means that their boundary is not enhanced. However, their tissue intensity is more homogenous so tracking of the maximum encountered intensity is not required.

The vertex can be moved only if the destination voxel has an intensity value inside the range of interest (contents of the user-drawn boundary). This rule favors boundaries with lower intensity surrounding tissues, which is more common than higher intensity surrounding tissue.

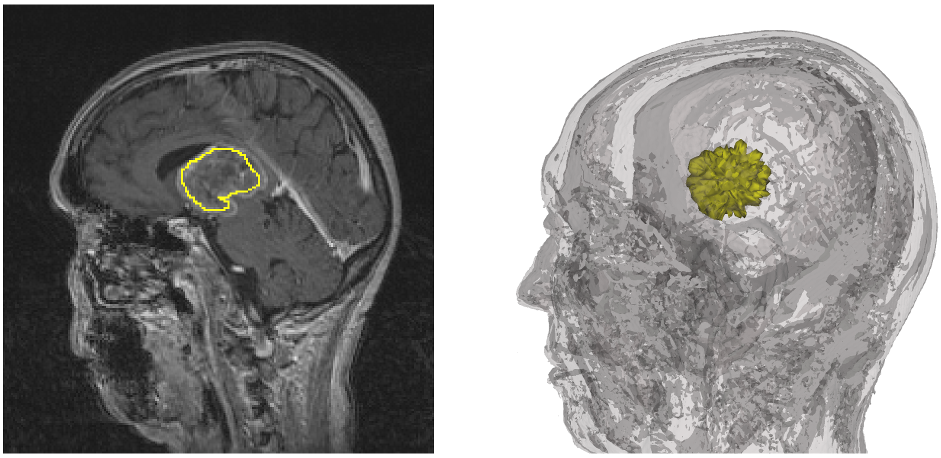
3.3 Results

The presented system was realized in C++ and the automatic segmentation took about one second per dataset. It takes about 30 seconds including the time it takes to locate the file, draw an outline, execute segmentation and extract a marching cubes isosurface. These measurements were taken on an Intel Core i7-920 CPU (2.66 GHz) with a GeForce 8800GTX graphics card on Windows7 x64. Note that the increase of the image region size unrelated to the tumor, e.g. when scanning the entire head instead of only the segment that contains the tumor, does not affect the segmentation speed.

3.3.1 Glioma

To evaluate the system, neurological surgeons with several years of experience in the resection of tumors performed manual slice-by-slice segmentation of 27 WHO grade IV gliomas. The tumor outlines for the segmentation were displayed by the contrast-enhancing areas in T_1 weighted MR images. Afterwards, the segmentation results were compared with the segmentation results of the proposed system via the Dice Similarity Coefficient (DSC) [SWM+06, ZWB+04].

The average DSC for all datasets was 80.8% (see Tab. 3.1 on the facing page).



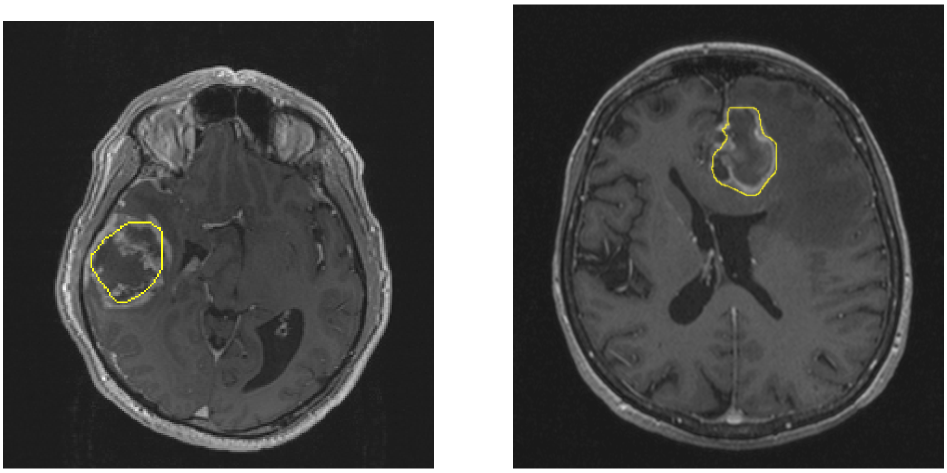
(a) Sagittal cross-section

(b) 3D view

Figure 3.10: Automatic segmentation result, dataset m1_hm.

Dataset	Tumor volume (mm^3)		DSC (%)	Time (min)
	Manual	Automatic		
m1_ag	1489.0	771.4	63.7	4
m1_cpa	3435.1	2441.8	73.6	5
m1_ek	10871.2	7956.3	80.2	11
m1_gkl	2164.5	2111.0	78.6	7
m1_hm	39598.7	36638.8	80.1	9
m1_ls	29513.7	26365.4	89.2	19
m1_mg	43507.7	33460.6	83.9	6
m1_mj	73452.5	73316.6	91.3	16
m1_ns	1631.3	1811.7	81.5	3
m1_sm	3226.7	2043.5	75.6	4
m1_td	9221.9	9888.4	77.7	10
m1_vm	1526.0	2465.0	73.8	3
m2_dg	31264.8	36891.6	88.3	14
m2_gg	57839.6	57318.9	88.0	6
m2_lg	785.6	1465.2	85.4	4
m2_mk	22985.1	32095.3	76.5	3
m2_sk	24326.0	25772.6	89.2	7
m3_vd	49886.1	40481.3	85.8	10
m3_war	36390.0	45843.0	94.0	7
m3_wg	29715.8	28882.2	68.6	7
m3_wi	23276.0	16183.4	79.3	6
m3_wj	23080.5	16176.9	81.1	5
m3_wn	8405.4	4844.5	80.7	3
m3_wpe	7832.1	4656.6	82.5	4
m3_wr	16259.7	16585.6	89.5	4
m3_zg	22607.0	14767.0	73.2	5
m3_zn	10036.4	5398.2	70.0	5
Mean	**21641.8**	**20245.7**	**80.8**	**6.9**
Std. deviation	**19156.1**	**19275.0**	**7.5**	**4.1**
Minimum	**785.6**	**771.4**	**63.7**	**3**
Maximum	**73452.5**	**73316.6**	**94.0**	**19**

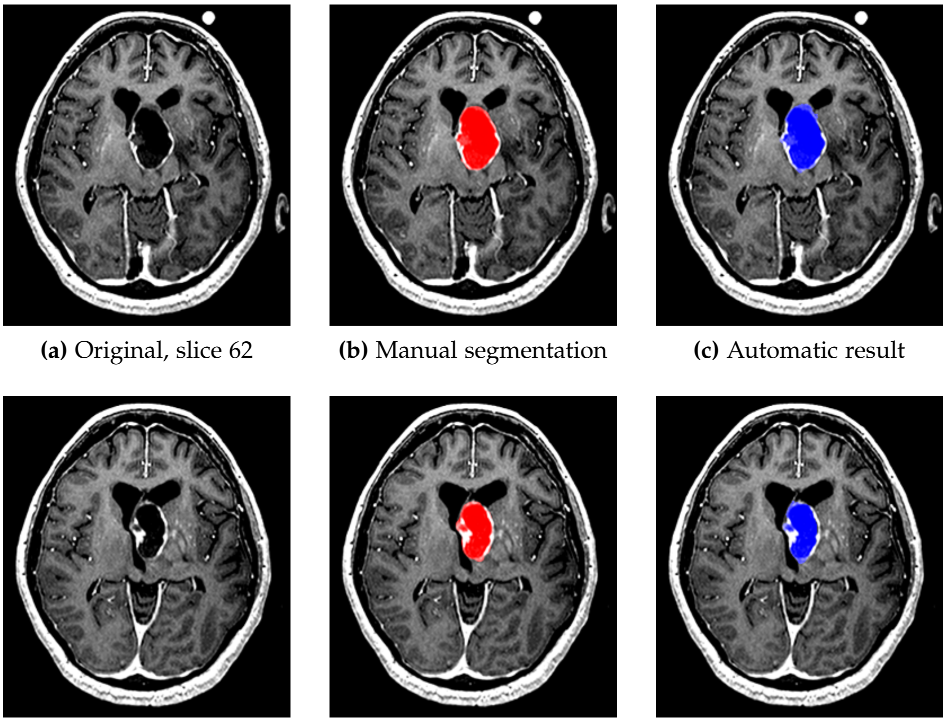
Table 3.1: Segmentation accuracy and manual segmentation times (in minutes) for 27 gliomas.



(a) Dataset m1_mj, slice 66

(b) Dataset m1_ls, slice 88

Figure 3.11: Automatic segmentation result for two glioma datasets.



(a) Original, slice 62

(b) Manual segmentation

(c) Automatic result

(d) Original, slice 68

(e) Manual segmentation

(f) Automatic result

Figure 3.12: Comparative results for glioma dataset m3_wr.

Figures 3.10 on page 46 and 3.11 on the facing page show segmentation results of the presented system. Fig. 3.12 on the preceding page presents a direct comparison of two manual and automatically segmented MRI slices.

Table 3.2 and Fig. 3.13 on the next page present the segmentation results for different user initializations for a selected dataset (m3_wr). The values belong to a tumor that was located in a MRI dataset between slice number 42 and slice number 73 (center slice for the tumor: 57/58). As can be seen, the Dice Similarity Coefficient decreases if the user draws the initial contour on a slice that is located near the border of the tumor. However, if the user selects one of the slices around the tumor center (55-60), the resulting DSC is over 80%. So, there is a certain robustness of the presented system if the user does not select a slice that is located too far away from the tumor center. Even when the user initialization was varied on one slice (slice 64), the following robust DSC results were obtained: 82.03%, 85.76%, 80.65%, 84.68% and 82.47%.

Slice	**Tumor volume (mm^3)**		**DSC (%)**
	Manual	**Algorithm**	
45	16259.7	2532.9	21.7
50	16259.7	9568.3	62.7
55	16259.7	15636.4	82.8
56	16259.7	16625.9	82.9
57	16259.7	15768.7	84.3
58	16259.7	18457.6	84.9
59	16259.7	19005.9	85.4
60	16259.7	18441.8	86.7
61	16259.7	20612.1	82.7
62	16259.7	21294.2	82.4
63	16259.7	18321.6	76.8
64	16259.7	18236.4	80.7
65	16259.7	20758.3	76.7
70	16259.7	6714.6	35.4

Table 3.2: Glioma segmentation results for different user initializations on dataset m3_wr.

Comparing with other methods is not straightforward. Gibbs et al. [GBBH96] do not report DSC or distance errors, they only report calculated tumor volume.

The automatic method of Prastawa et al. [PBHG04] also reaches overlap of around 70-80%, but they modeled tumor and edema (swelling) separately.

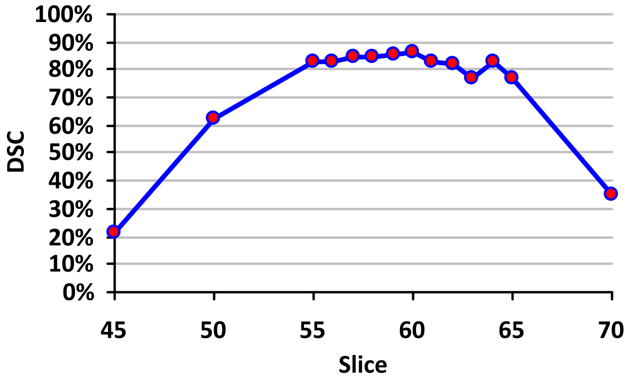


Figure 3.13: Dice Similarity Coefficient results for different user initializations on glioma dataset m3_wr. The lower tumor boundary for this dataset was slice number 42 and the upper tumor boundary was slice number 73.

Therefore, their method cannot be directly compared to the one presented here.

Clark et al. [CHG+98] did not report DSC, but it can be calculated from their "Table V" as:

$$DSC = \frac{2 \times TP}{2 \times TP + FP + FN}$$

They have an average DSC of 77.2%, which is slightly worse than obtained here (80.8%).

Egger's method [EZB+10] has a 2% better DSC on the same datasets, but also takes five times longer to execute. Given that the execution time is still only a few seconds, it can be said that his method is quite similar to the one presented here in terms of precision and performance.

3.3.2 Adenoma

The pituitary adenoma case was evaluated on ten cases manually segmented by neurosurgeons (Tab. 3.3 on the facing page). The average DSC for all datasets was 75.92%±7.24% (minimum 63.74 and maximum 86.08, see Tab. 3.4 on the next page for details).

Fig. 3.14 on page 52 shows the user initialization (yellow outline) that have been superimposed onto a slice of the magnetic resonance image and the corresponding segmentation result. The segmentation results for the remaining 9 pituitary adenoma datasets are presented in Fig. 3.15,

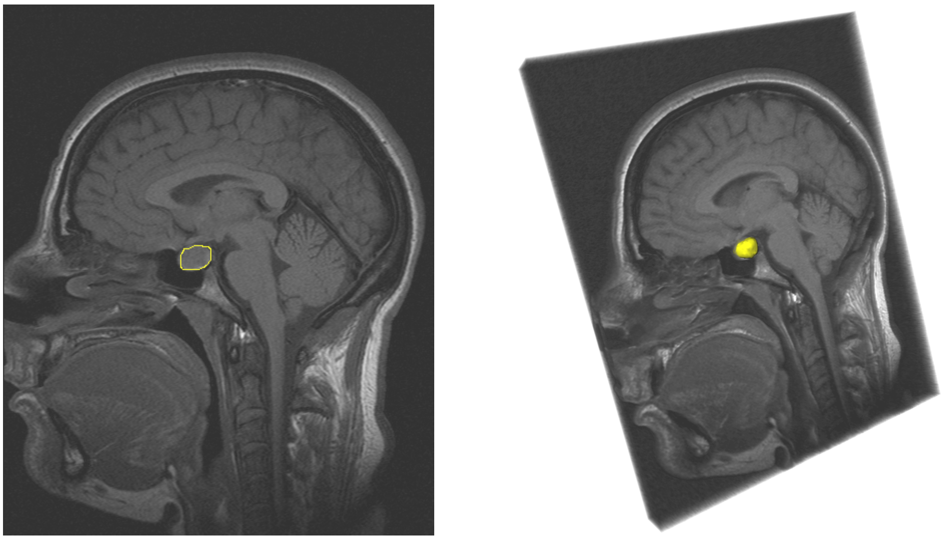
Dataset	Image size			Voxel size (mm)			Sequence	View
	X	Y	Z	X	Y	Z		
Wei	448	512	25	0.45	0.45	6	T_1-TRA SE	axial
Zon	256	256	160	0.98	0.98	1	T_1-TRA SE	axial
Aff	512	512	80	0.59	0.59	2	T_2-TSE	axial
Zin	416	512	25	0.45	0.45	6	T_1-TRA SE	axial
Ame	416	512	13	0.53	0.53	3.3	T_1-SE	sagittal
Lip	416	512	13	0.53	0.53	3.3	T_1-SE	sagittal
Men	384	512	13	0.49	0.49	3.3	T_1-SE	coronal
Tzi	512	512	19	0.45	0.45	3	T_2-TSE	coronal
Akd	416	512	13	0.53	0.53	3.3	T_1-SE	sagittal
Lam	416	512	13	0.53	0.53	3.3	T_1-SE	sagittal

Table 3.3: Information about pituitary adenoma datasets.

Dataset	Volume of tumor (mm3)		DSC (%)	Time
	Manual	Automatic		
Wei	6712.99	3484.46	63.74	4
Zon	4183.5	3282.92	80.9	3
Aff	7023.36	5620.09	77.76	5
Zin	5608.68	4316.91	71.6	3
Ame	2801.45	1825.35	75.55	4
Lip	9874.37	8149.63	86.08	4
Men	5573.5	3111.54	65.72	4
Tzi	4811.49	3430.79	78.77	4
Akd	15567	13049.8	83.74	4
Lam	843.383	597.16	75.35	4
Mean	**6300.0**	**4686.9**	**75.9**	**3.9**
Std Dev	**4069.7**	**3579.7**	**7.2**	**0.6**
Min	**843.4**	**597.2**	**63.7**	**3.0**
Max	**15567.0**	**13049.8**	**86.1**	**5.0**

Table 3.4: Segmentation accuracy and manual segmentation times (in minutes) for 10 adenomas.

by visualizing them as a three-dimensional closed surface model (yellow) faded into the corresponding MRI dataset.

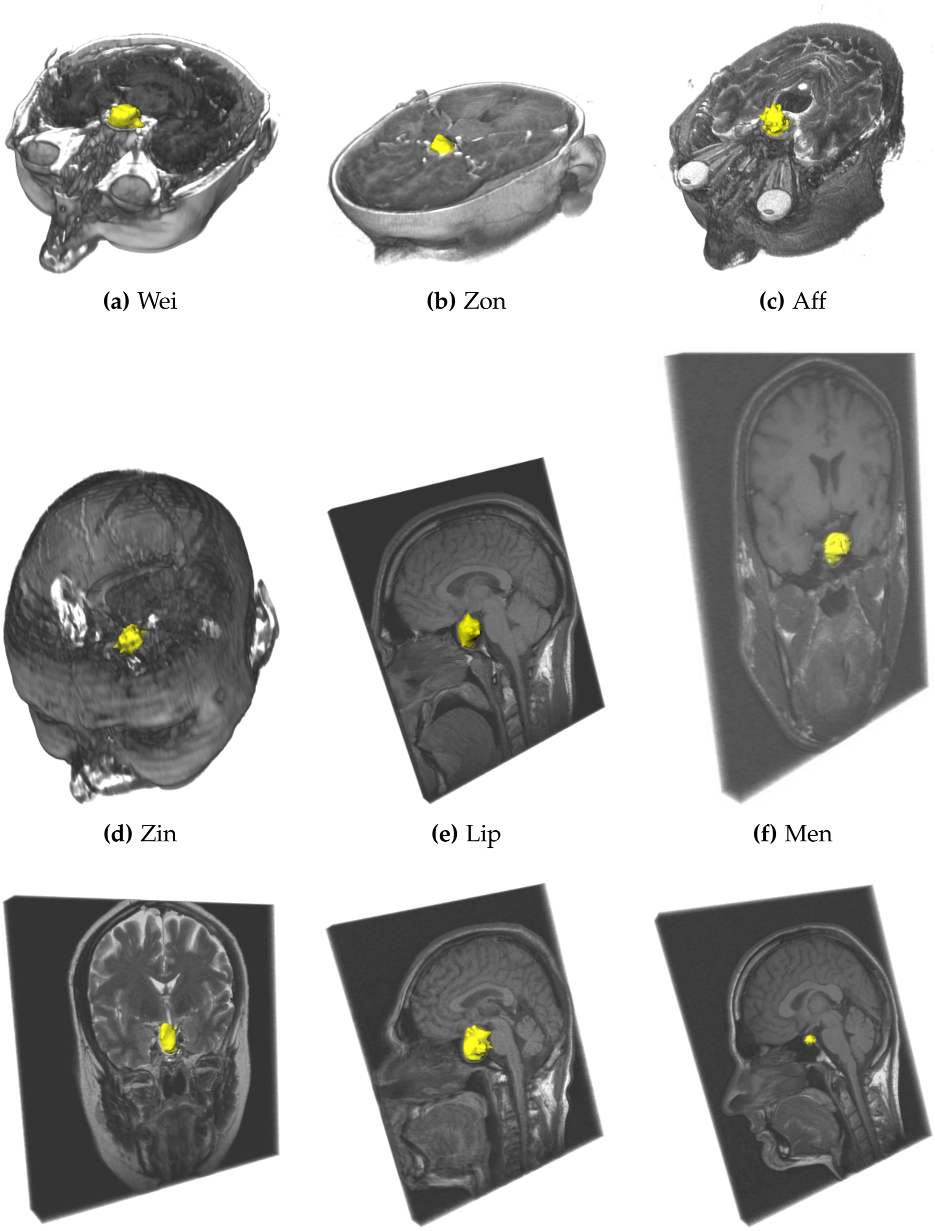


(a) Initialization contour

(b) Segmentation result

Figure 3.14: Initialization contour and automatic segmentation result for dataset Ame.

Comparing with Egger's method [EZF+13], 2% lower DSC can be justified by higher execution speed, but both differences can be considered negligible.



(a) Wei

(b) Zon

(c) Aff

(d) Zin

(e) Lip

(f) Men

(g) Tzi

(h) Akd

(i) Lam

Figure 3.15: Automatic segmentation results for 9 adenoma datasets.

3.4 Conclusion

For an accurate estimate of tumor volume it is necessary to develop systems – like the one introduced in this chapter – that use all slices to calculate the tumor boundaries. Simpler methods like geometric models provide only a rough approximation of the tumor volume and should not be used, as an accurate determination of the size is of paramount importance in order to draw safe conclusions in oncology.

In conclusion, the proposed system can be used to augment the manual segmentation for the purpose of tumor progress tracking. It imposes no special pre-processing requirements, executes very quickly on modern hardware, its initialization is easy, intuitive and robust, and it provides decent results.

But, of course, there are drawbacks. For this system gaps in contrast enhanced boundary and very inhomogeneous tumor interior are problematic, leading to inferior DSC results. Thus, the system is not suitable for treatment planning. For the case of brain tumors, better segmentation methods exist, such as that of Prastawa, even though they require longer processing time.

There are also areas of possible improvement. For example, some parameter specifications of the proposed system can be reduced in order to exchange the contour initialization with a single user-defined seed point that is placed in the object of interest. Moreover true convergence does not exist – it is approximated by a limited number of steps coupled with slow inflation speed after the boundary is reached.

Chapter 4

Basic Segmentation of Vertebral Bodies

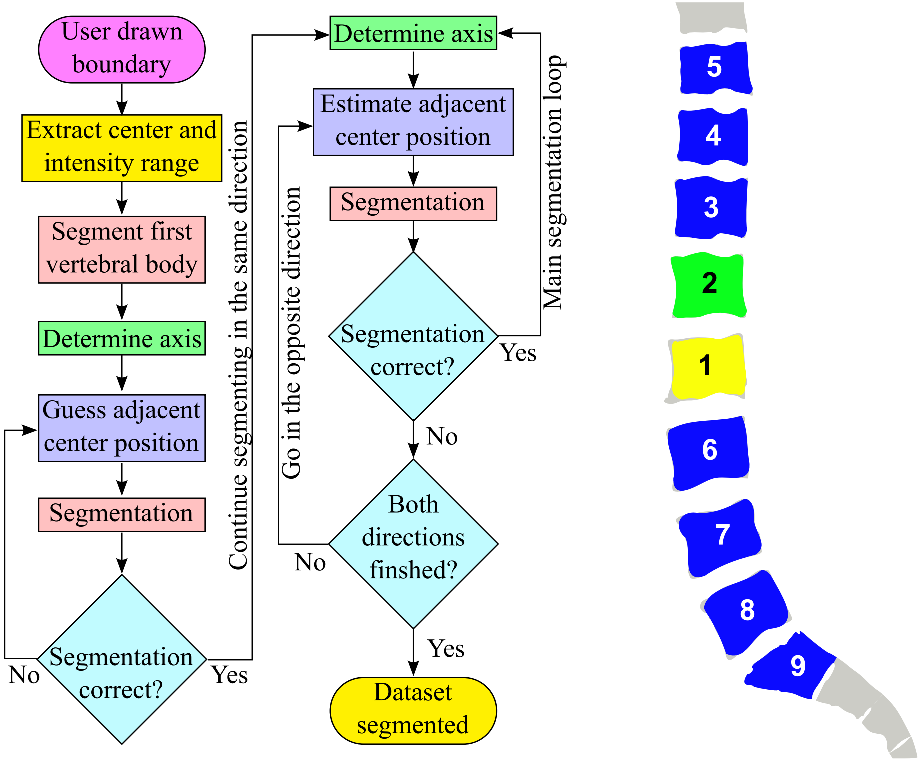
In this chapter a 3D bottom-up system in detecting vertebral bodies is presented, which relies on the segmented vertebra shape to predict the global spine shape. A subdivision surface hierarchy is introduced as an efficient global-to-local smoothness constraint which can be thought of as an internal force. Low-High (LH) values are also used to ease boundary classification. This system is very fast (3-4 seconds) and achieves around 70%-75% Dice similarity coefficient (DSC) for thoracic and lumbar datasets.

This chapter builds upon the system presented in the previous chapter. Compared to that algorithm:

- Outline initialization is the same
- The implicit binary boundary and inflation are improved by usage of LH images
- The inflation process can optionally use a constrained subdivision hierarchy
- New algorithms are added to manage an array of segmented objects, in order to free the human operator from initializing every vertebra separately

The overall structure introduced in this chapter is also kept in the next chapter, with improvements to individual building blocks such as boundary estimation and inflation. This chapter, together with the next one, is published in one paper [ZVD^{+}12].

4.1 Overview



(a) High-level block diagram of the system. Yellow - user interaction and initialization. Red - segmentation by inflation. Green - axis estimation. Blue - center distance calculation. Cyan - correctness and termination checks.

(b) Possible order of segmentation of vertebral bodies in a dataset. Yellow - initialized vertebra. Green - first adjacent vertebra whose center distance to the initialized center needs to be guessed. Blue - other vertebrae, whose center distance can be estimated well.

Figure 4.1: Overview of the segmentation steps.

In many clinical applications, a fast and accurate method for spine segmentation in routine magnetic resonance images is required. Thus, the goal of the system presented here is to provide a fast 3D technique for semi-automatic vertebral body segmentation, which is applicable to a wide variety of MRI datasets. In general, the MRI spine datasets are acquired as sagittal slices (see Fig. 4.2). Further details are given in section 7.2 on page 101.

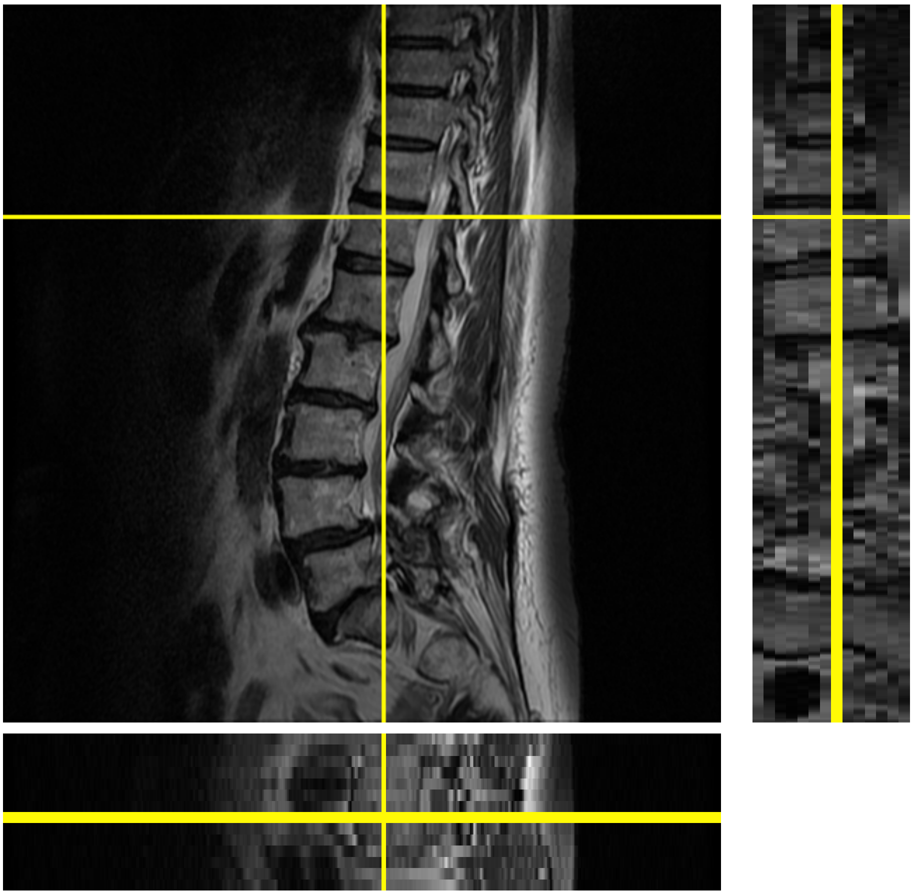


Figure 4.2: Cross-sections of a typical dataset. Top left: sagittal (XY). Top right: coronal (YZ). Bottom: axial (XZ). Yellow lines indicate positions of other the two cross-sections. The voxel size of this image is 0.625mm×0.625mm×4.4mm (X×Y×Z), i.e. the lateral resolution is 7 times lower than vertical and horizontal resolution.

The block diagram of the overall system is shown in Fig. 4.1a on the preceding page. The major steps are:

1. Use the user provided initialization to find the minimum and maximum intensities of the vertebral body tissue and determine the approximate center.

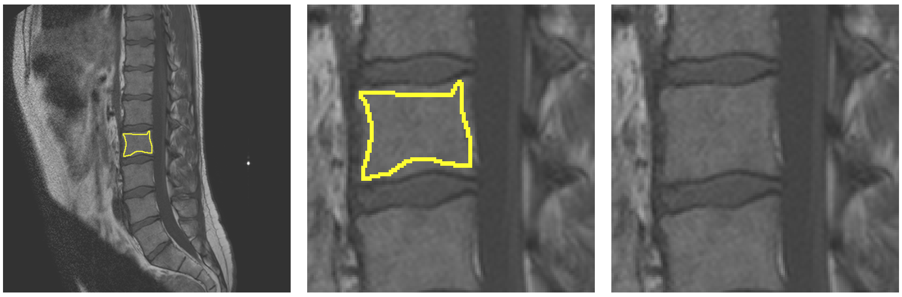
2. Segment the first vertebral body.

3. Determine the spinal axis deduced from the shape of the first vertebral body (which is roughly of cylindrical shape) and estimate the center position of an adjacent vertebral body.
4. Segment the second vertebral body. If this segmentation fails the plausibility check, go to step 3 with modified estimation parameters.
5. With two (or more) segmented vertebral bodies, use the axis of the last vertebra and both centers to estimate the position of the next center.
6. Continue the segmentation process in this direction by segmenting, determining the spinal axes and estimating the next centers.
7. When no more vertebral bodies are found in the initial search direction, go in the opposite direction, using the second and first vertebra to predict the adjacent vertebra's center position. Continue in this direction until no more vertebrae remain. See Fig. 4.1b on page 56 for an example.

4.2 Initialization

The initialization consists of a user-drawn approximate outline of a vertebral body. It should be on a slice which is approximately central to the spine. From this outline the algorithm calculates the center: X and Y from the center of the area of the bounded planar shape and Z from the selected slice's index. Furthermore, the minimum and maximum intensities of all bounded pixels of the selected slice are calculated, i.e. the voxel intensity range of interest. Intensities with low frequency of occurrence (0.1%) are considered as noise and are discarded. The drawn curve has to be completely inside the vertebral body, but close to the actual edge (see Fig. 4.3 on the next page). This is required in order to achieve a good first center and intensity-range estimate. Basically, the initialization is almost the same as in chapter 3 on page 39.

The user also indicates the vertebra that has been initialized. This is not used in the segmentation process, but for naming of result files to enable easier calculation of results and comparisons.



(a) Middle slice of a dataset with initialization outline in yellow **(b)** Initialization closeup **(c)** Closeup of the original slice

Figure 4.3: Example initialization, dataset DzZ_T1.

4.3 Segmenting the Vertebral Body

Each vertebral body is segmented using an iterative inflation algorithm. The algorithm starts with a small triangular surface mesh (see Fig. 4.8 on page 65) at the approximate center of the vertebral body. This mesh is enlarged using balloon inflation forces [Coh91], enforcing smoothness and a star-shaped geometry. The inflation ends when convergence is detected.

Star-shape is enforced by allowing the inflation only along center-vertex direction - this was kept from the previous chapter. In addition to curvature and angle checks smoothness can be enforced by using a subdivision surface scheme.

[ŠBSG06] introduced LH (low-high) values (Fig. 2.18 on page 25). The interesting feature of LH values is that their difference indicates proximity to a boundary. This information is used to speed up the inflation in the internal region. The area of an approximately constant H value is the vertebral body with its boundary region. Thus, we never want to inflate the segmentation surface beyond it.

Also, the vertices are allowed to move only into regions with an intensity within the range of interest (contents of the user-drawn boundary). They are not allowed to go into regions with significantly lower intensity, which surround the vertebral bodies (cortical bone, periosteum). See figures 4.2 on page 57 and 4.3. Some vertices may still end up slightly outside of the detected vertebral body boundary due to smoothing induced by the subdivision hierarchy normalization.

The following steps are performed iteratively:

1. Inflate the mesh (move the vertices outwards)
2. Normalize the subdivision hierarchy: Adjust the positions of vertices so that they comply with subdivision rules. Also minimize the amount of movement that needs to be applied to the vertices to obey the subdivision rules.
3. If the average edge length of the polygonal segmentation surface is greater than two times the average image spacing, another level to the subdivision surface hierarchy is added. This is needed to better adapt to small features.
4. Smooth the surface mesh slightly. This is required to overcome noisy voxels, which would otherwise prevent inflation of the mesh. The heuristic hierarchy normalization already includes smoothing as a side-effect, so this explicit smoothing step is not needed.
5. Update per vertex normals and curvature estimates and recalculate the polyhedron center. Only the first vertebra has a very good initial center because it is user-provided. All the other vertebral bodies must calculate their center from the polyhedron shape to account for different inflation speed in different directions.
6. Stop when convergence is detected.

4.3.1 Inflating the Mesh

The inflation is done by examining each vertex separately. If it is in the interior region (L and H values are equal), the vertex is moved using the largest step size. Otherwise the position where it should be placed is calculated and a boundary check conducted. If the destination position is "inside", the vertex is moved, otherwise it is not moved. The vertex will not be moved if the destination voxel:

- has an intensity outside of the range of interest
- has an H value lower than the previous voxel's H value (with 10% tolerance).
- has an intensity higher than the previous voxel's intensity (with 5% tolerance)

If a vertex is to be moved, it is moved in the direction of the center-vertex vector, thus maintaining the star-shape. In order to calculate the move distance the cosine of the angle between the center-vertex vector and the surface normal vector is taken into consideration. The greater the angle, the lower the inflation speed, i.e. the magnitude of the move vector is lower:

$$moveVectorMagnitude = cos\varphi \times stepSize$$

This slows down the inflation when the mesh starts adapting to the shape of the boundary. Also taken into account is the surface curvature: The higher the curvature, the lower the inflation speed. The move vector magnitude is also scaled by the curvature factor in the range [0.15,1.0]:

$$c_{factor} = min(1, \frac{1}{2 \times gaussCurvature})$$

Thus the inflation speed is lower for vertices near feature points such as ridges, valleys, peaks and dents.

4.3.2 Degenerate Vertex Distribution

If the initial center for the segmentation is too close to some boundary, many vertices will get stuck in that boundary and the surface will inflate much more in the opposite direction. This will result in a highly uneven distribution of vertices over the surface: the surface part which inflates the most will have the lowest vertex density (Fig. 4.4 on the next page), as the subdivision hierarchy does not allow inserting individual vertices as the mesh-based approach in chapter 3.

There are two options to counteract this. First one is to introduce some kind of regularization, to prevent uneven distribution from occurring. The other option is to detect if it happens, and fix it.

The prevention option was implemented as a change to inflation rules to also move the vertices tangentially in an attempt to have a more uniform distribution. The target position for inflation is shifted tangentially, minimizing distances to the neighboring vertices. However it takes a lot of time for this to propagate across the entire mesh, and the non-uniformity is introduced more quickly by the differing speeds of inflation.

For detect and fix option, the average and standard deviation of edge lengths are calculated in each iteration of the inflation. If a degenerate vertex distribution is detected (σ>0.8) during inflation, the segmentation is restarted using the current mesh-center (center of mass assuming uniform density) as the initial center. The current center is farther away from the boundary and closer to the true center. Thus, it greatly decreases the chance of an uneven vertex distribution occurring again. Restarting segmentation

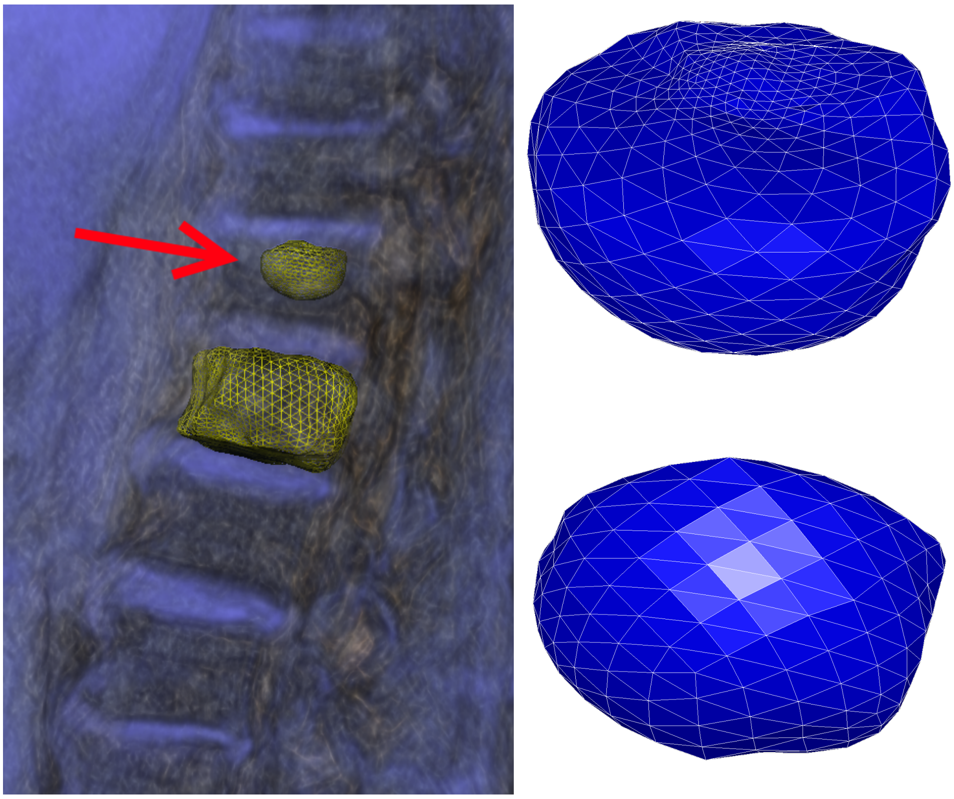


Figure 4.4: Uneven vertex density occurs with center estimate too close to a boundary (left image). A view from top (top right) reveals a high vertex density, while a view from bottom (bottom right) shows a low vertex density.

is also more computationally efficient than applying mesh optimization, and was therefore chosen.

An uneven vertex distribution occurs most frequently when segmenting the first vertebral body adjacent to the initial user-initialized vertebra. At this point in time, the axis of vertebral column can not be estimated reliably. This sometimes leads to an initial center farther away from the true center. One restart is usually sufficient to resolve the problem.

4.3.3 Convergence Check

The segmentation is finished when any of the following three conditions is met:

- The increase in average center-surface distance in the current iteration is smaller than 0.2×voxel spacing. This is the standard termination condition.
- The center-surface distance is 50% greater compared to the last vertebra (compared to user boundary for the first vertebra). This is a sanity check indicating a "bleeding" beyond the boundary, because the sizes of vertebral bodies change gradually as we go up or down the spine.
- The maximum number of iterations is exceeded (usually 200). This is just a precaution, in order to avoid a fluctuating behavior. In most cases the segmentation is finished in about 50 iterations.

4.4 Constrained Subdivision Hierarchy

The butterfly algorithm [DLG90] is the simplest interpolating subdivision scheme working on triangle meshes. Interpolating means that the vertex's position, once calculated, will be part of all finer mesh levels and consequently the coarser shape information is preserved. This is different from approximating subdivision schemes, where the new vertex positions are calculated by averaging the old vertices. "Modified butterfly" was presented in [ZSS96], and it avoids problems with irregular vertices (vertices with valence≠ 6) by using somewhat different rules for those cases.

Both versions of the butterfly algorithm subdivide the mesh by splitting each triangle's edges, and connecting them thus turning each triangle in the input mesh into 4 triangles in the refined output mesh (Figs. 4.5 and 4.6 on the following page). Position of each of these new points is independently calculated according to subdivision rules based on positions of "old" input points (Fig. 4.7 on the next page).

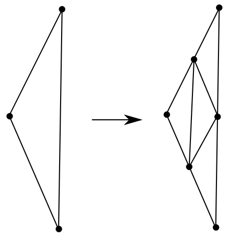


Figure 4.5: Mesh is subdivided by splitting edges.

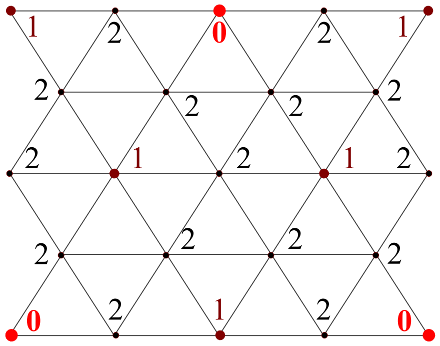


Figure 4.6: Surface patch showing the vertex hierarchy with two refined levels (0–base level).

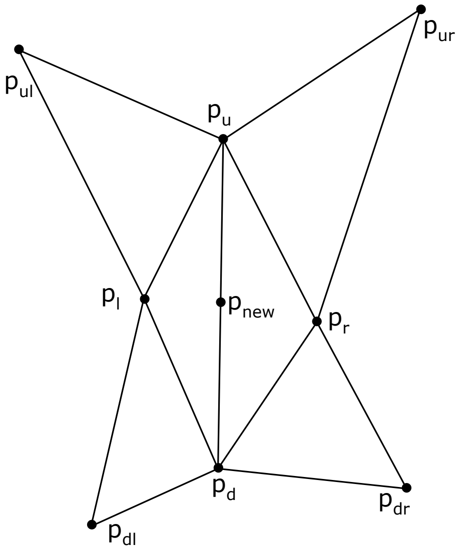
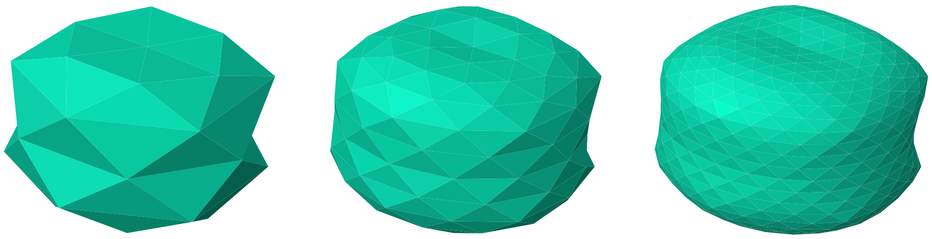


Figure 4.7: Butterfly subdivision rule for the regular case. Position of point p_{new} is based on positions of points around it: $p_{new} = \frac{1}{2}(p_u + p_d) + \frac{1}{8}(p_l + p_r) - \frac{1}{16}(p_{ul} + p_{ur} + p_{dl} + p_{dr})$.

A modified butterfly subdivision hierarchy scheme is used here to stimulate the smoothness of the segmentation surface. A closed triangular polyhedron (32 vertices, 60 triangles) is used as a base mesh for subdivision (see Fig. 4.8 on the facing page). New mesh levels are created through the subdivision rules until the average edge length l is comparable to the voxel size $s : s <= l < 2s$, where $s = \sqrt[3]{s_x s_y s_z}$ is the geometric mean of the voxel spacings. As the segmentation mesh grows, additional levels are added to maintain $s <= l < 2s$. Up to four levels were needed for representing final-stage vertebral body meshes in highest resolution images.



(a) Mesh with level 0 vertices only **(b)** Mesh with vertices from levels 0+1 **(c)** Mesh with vertices from levels 0, 1 and 2

Figure 4.8: The polyhedron used as an initial segmentation surface.

At the beginning, only the base level (level 0) is “free”: the vertices from this hierarchy level can be positioned freely during the inflation, but all vertices of finer levels have their positions calculated according to the subdivision rules. When the inflation starts converging, level 1 is also set free, so both level 0 and level 1 have free vertex positioning. The inflation continues until the next slowdown or iteration limit exhaustion, then another level is set free. This continues until all levels are free. Once additional levels (besides level 0) are started being freed additional iteration limits are also set, which get shorter as the algorithm approaches the complete removal of the subdivision rules. The iteration limit is given exponentially: $maxIterations = 2^{numberOfUnfreeLevels}$. When only one “unfree” hierarchy level remains, it has a limit of 2^1 iterations, and when all levels are free, the inflation is limited to only $2^0 = 1$ iteration. This favors global smoothness but still allows segmentation of some fine features. This iteration limit is needed because there is no true equilibrium condition.

The actual inflation step repositions the vertices with no regard to the subdivision rules. Thus, the enforcement of the subdivision hierarchy for all the unfree levels afterwards is needed. Two alternatives have been developed to enforce the subdivision rules:

- a local heuristic rule
- a global optimization

Both ways adjust the positions of vertices in free levels, trying to minimize the amount of movement of all vertices in the mesh. This movement is mostly influenced by the subsequent application of subdivision rules in the dependent levels. Once the positions of the vertices in free levels are known, positions of vertices in the dependent levels are recalculated using subdivision rules. This has a side-effect of strong smoothing in case of the heuristic rule and negligible smoothing in case of the global least squares fitting.

The heuristic hierarchy normalization is based on a local rule, which adjusts the positions of the vertices in the coarser levels until the base level has been reached. This adjustment is based on a position update which has been applied to each vertex during the inflation step. Basically, coarser level update vectors are averages of finer level update vectors, influenced by the modified butterfly subdivision rules. The averaging is formulated in a way to approximate the (non-existing) inverse of the subdivision rule.

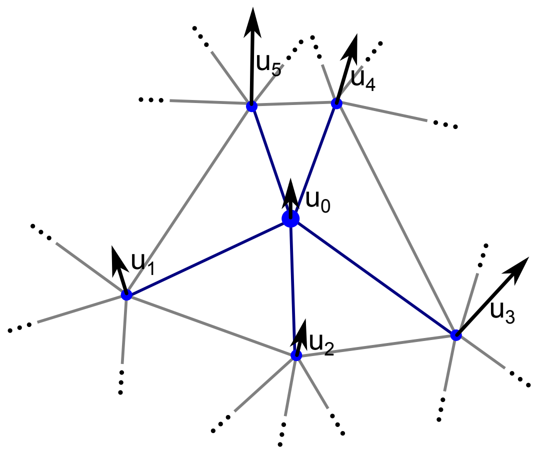


Figure 4.9: Heuristic hierarchy normalization adjusts positions of coarser level vertices, and then calculates the positions of dependent vertices using subdivision rules. u_i – position updates from inflation step. u_0 – (coarser) control vertex. u_1-u_5 – directly dependent vertices on the next finer level.

The position update vectors are propagated from finer to the coarser levels using the following formula (also see Fig. 4.9):

$$u_{adjusted} = w \times u_0 + \frac{1-w}{n} \sum_{i=1}^{n} u_i$$

$$w = \frac{1}{2}\frac{1}{3^{\lambda}}, w = (\frac{1}{6}, \frac{1}{18}, \frac{1}{54}, ...)$$

$$\lambda = level\,above\,finest$$

w is a weight related to the number of vertices which have contributed to the adjustment of directly dependent vertices. It decreases exponentially. 3 is chosen as the base base because in the regular case (6 neighbors), the number of vertices increases 3-fold with each level, i.e. each vertex in the base mesh contributes a half of 6 new vertices in the subdivided mesh (Fig. 4.6 on page 64).

A globally optimal normalization algorithm using a least squares fitting (Jacobi SVD) was investigated. This approach minimizes the sum of the squared distances between the inflated positions and the subdivision positions after optimization. This results in a system of linear equations where the number of unknowns is equal to the number of free vertices. The coefficient matrix of the system is constant, since it depends only on the polyhedron topology, the number of free levels and the total number of levels.

However the least squares fitting is less computationally efficient than the proposed heuristic because more calculations are involved. The initial calculation of the matrix data structures and their caching are needed. Also, if there are many vertices in the base mesh, overshooting effects start to appear, which is frequent in interpolating schemes. This can be avoided by using a low or moderate number of vertices in the base mesh, or by applying slight smoothing after each iteration, thus fighting noise and small ambiguities in the data. Explicit smoothing is not required for heuristic normalization because heuristic normalization includes smoothing as a side effect. Therefore, the heuristic approach was used more frequently.

Whether subdivision surface hierarchy was used, and whether it employed local rule or global optimization gave rise to three different variants of the segmentation system which were tested in chapter 7 on page 99:

- no subdivision hierarchy, i.e. free-form mesh (bbFF)
- subdivision hierarchy with local heuristic Normalization (bbH)
- subdivision hierarchy with optimal global normalization (bbO)

The version presented in this chapter was retrospectively named "binary boundary" and abbreviated **bb**.

4.5 Finding Neighboring Vertebrae

The main segmentation loop consists of predicting the position of the adjacent vertebra's center, where the next segmentation is initiated. This frees the user from further initializations.

Prediction consists of analyzing the shape of the previously segmented vertebral body. The main vertebra axis is extracted, which is then used together with the center positions of the last two vertebrae to predict the position of an adjacent vertebra's center. Then the segmentation is continued in that position. The required initial intensity statistics are taken from the previously segmented vertebra. Thus, the system adopts the gradual intensity change within the dataset, which is a common MRI artifact. See Fig. 4.1b on page 56.

4.5.1 Pose Estimation Using Face-Merging

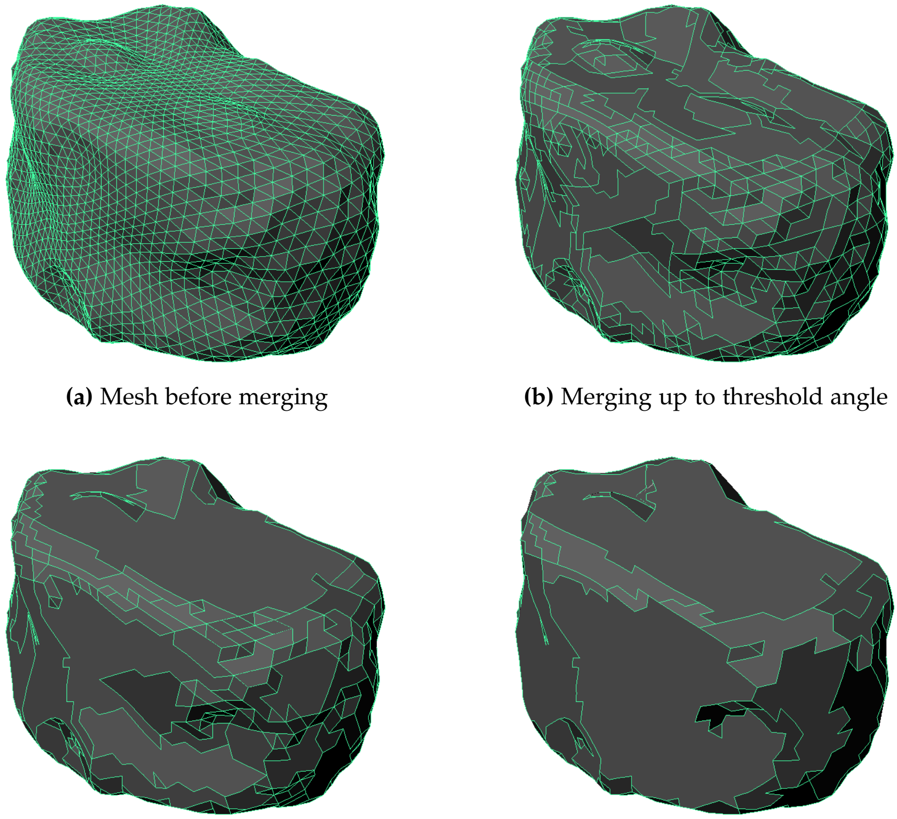
Vertebral bodies are geometrically similar to short and thick cylinders. This fact can be exploited in order to find the disk axis by merging mesh faces in order to identify a cylinder base. The face merging algorithm starts with the final mesh resulting from the vertebral body segmentation. The face normals are adjusted in such a way that their length is equal to the face's surface area, which is a half of the length of the vector product of two triangle edges. Then all pairs of adjacent faces in the mesh are examined. All pairs of normals which form an angle of 0.1 radians or smaller (threshold angle) are merged. The normal of the merged face is calculated as the sum of the normals of the old faces. This keeps the normal length approximately proportional to the face's surface area.

When no further merging is possible, the threshold angle is increased by another 0.1 radians and the merging process is repeated. Increasing the threshold angle by 0.1 radians is continued until it reaches 0.3 radians (17.2°). See Fig. 4.10 on the next page.

When all pairs of adjacent faces with an angle between their normals of 0.3 radians or less have merged, the vertebra has a shape which looks somewhat similar to a disk (cylinder). The two largest faces usually correspond to the vertebral body endplates, which is the part of the vertebral body which presses against the intervertebral disk. The cylinder sides are represented by several faces.

For the first vertebral body the normals of the four largest faces are kept, which are then used for estimation of adjacent vertebra center.

For all other vertebral bodies the vector between the centers of the last two segmented vertebral bodies is known. This gives a preferred axis direction and makes pose estimation easier. All normals are weighted by how parallel they are to this vector, using the cosine of the angle between them. This weighting process can increase the importance of some faces



(a) Mesh before merging

(b) Merging up to threshold angle

(c) Merging up to 2× threshold angle

(d) Merging up to 3× threshold angle

Figure 4.10: Merging faces in order to determine the main axis.

up to a factor of four. This is a kind of implicit model usage, and is useful for cases of imperfect segmentation where the top and bottom face do not appear flat.

4.5.2 Estimation of the Position of the Second Vertebral Body

Once the four axis candidates are available, the algorithm can use them to guess the center position of an adjacent vertebral body. The guessed center is calculated as the translation of the center of the first vertebral body along the axis, and the translation distance is the average distance between the center and the surface multiplied by a factor (see Fig. 4.11 on the following page).

Four factors (2.0, 1.8, 2.2, 1.6) for the normal of the largest face are tried first, then the same four factors for the normal of the second largest face.

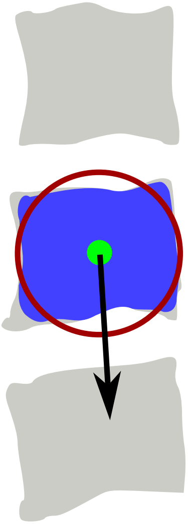


Figure 4.11: 2D schematic illustrating the process of estimating the position of the center of the first adjacent vertebral body. Grey: vertebral bodies. Blue: first segmented vertebral body. Green: center of the first vertebral body. Red: circle with radius equal to average center-surface distance of the first segmented vertebral body. Black: guess vector using the factor 2.0.

This continues until all four kept normals are exhausted, or until a vertebral body is successfully segmented based on one of these guesses. These four factors represent approximate anatomic ratio of vertebral body height and disk thickness to diameter, but are experimentally determined for usage in this algorithm.

The very first guess (the largest face and factor 2.0) is usually successful, but occasionally the largest face and some other multiplication factor get used. This is mostly due to anatomical differences. Using normals other than the one corresponding to the largest face is rare. It usually occurs when the supplied image has low contrast between vertebrae and surrounding tissues. Sometimes it occurs for images of the cervical spine, because the vertebral bodies of the cervical spine are small and less disk-shaped.

4.5.3 Estimating Positions of Other Vertebral Bodies

Once the algorithm has found the center positions of two vertebral bodies, it can make a good estimate of the center position of the next vertebral body (Fig. 4.12 on the next page). The adjacency vector is calculated as an average of the last vertebra's normalized axis and the normalized center-center vector from the last two vertebral bodies. This is more robust than using just one of these two, and robustness is important because one wrongly segmented vertebra breaks the chain and the algorithm stops. The length of this adjacency vector is the length of the center-center vector.

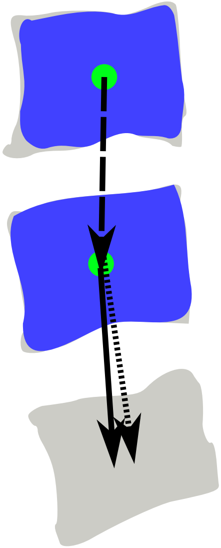


Figure 4.12: 2D schematic illustrating the estimation process. Grey: vertebral bodies. Blue: segmented vertebral bodies. Green: centers of the vertebral bodies. Dashed arrow: center-center vector. Dotted arrow: vertebral axis with length of the center-center vector. Solid arrow: average of the other two.

When more than two segmented vertebral bodies have already been segmented, just the last two are used. When starting to go into the opposite direction, the first and the second vertebrae are used, but the direction vector is also reversed so it points from the second to the first vertebral body.

4.6 Segmentation Correctness Measure

The main condition for assessing correctness of the segmentation is that the difference between the average center-surface distance of the current vertebral body is within 30% of the average center-surface distance of the previous vertebral body.

The segmentation correctness assessment has additional sanity checks using experimentally determined values:

- The volume of the current vertebral body is at most 2 times larger or smaller than the volume of the previous vertebral body. This roughly corresponds to 25% increase or decrease in vertebral body diameter.
- The volume of the current vertebral body is at most 12.5 times larger or smaller than the volume of the first segmented vertebral body. This roughly corresponds to 2.3× increase or decrease in vertebral body diameter. The difference between first and last vertebral body can be much bigger than between two adjacent ones, but it is desirable to limit this difference to prevent some obvious segmentation failures.
- The difference between the average center-surface distance of the current vertebral body is between $0.2 \times d_1$ and $1.8 \times d_1$, where d_1 is the average center-surface distance of the first segmented vertebral body.
- The distance between the centers of the current and the previous vertebral bodies is smaller than the average center-surface distance of the current vertebral body. This is a simple heuristic check for detecting some conceivable types of overlap.

4.7 Conclusion

A novel system for segmentation of vertebral bodies was presented [1]. This system relies on the segmented vertebral body shape to predict the global spine shape, so it can be called a bottom-up approach.

A subdivision surface hierarchy was introduced as an efficient global-to-local smoothness constraint. Since it is not tightly integrated into the presented system, it could be used independently, e.g. for other segmentation tasks. Low-High (LH) values are used to ease boundary finding. This can also be used in other application areas.

This system achieves 50%-72% detection rate for thoracic and lumbar datasets. It scores 70%-75% on Dice similarity coefficient. These results are not bad considering the segmentation speed and the dataset variety.

[1] Results are in chapter 7 on page 99. Taking the results sections out of vertebra segmentation chapters and putting them into a separate chapter enables better comparison between different versions of the vertebra segmentation system i.e. it better shows improvements from increased system complexity.

Finally, the greatest quality of this system is high speed. It processes a complete dataset in a few seconds, while other 3D spine segmentation methods take minutes.

The deficiency of detection (only ~72%) comes from the imperfect vertebral body segmentation. A wrong center estimate is usually caused by an incorrect vertebral body shape. And that is most influenced by the boundary detection imperfections. The next chapter therefore focuses on improving the boundary detection approach.

Chapter 5

Robust Soft-Boundary Segmentation

This chapter refines the system introduced in chapter 4, without significantly changing its structure. This chapter, together with the previous one, is published as one paper [ZVD$^+$12]. The next chapter is a complete reorganization of building blocks introduced in this chapter and chapters 3 and 4, with addition of many new elements as well.

In this chapter multiple image features are calculated in order to improve the boundary estimation. The multiple features could be directly integrated into the existing system, but the independent handling of various features would significantly complicate the algorithm. An easier and more efficient approach is to combine the multiple features into a single boundary probability estimate.

The main improvement introduced in this chapter over the previous one is combining of multiple features to arrive at an explicit soft boundary estimate instead of an implicit binary boundary. There are other important improvements:

- Faster-to-create single-point initializations are now possible.
- Pose estimation no longer uses face merging, instead it uses ICP. This is more reliable so averaging of vertebral axis with center-center vector is no longer needed. Vertebral axis alone is used to predict adjacent vertebra's center.
- Vertices can now be deflated, i.e. moved inwards, not just outwards from the center. This enables establishment of a real equilibrium, not just inflation slowdown.

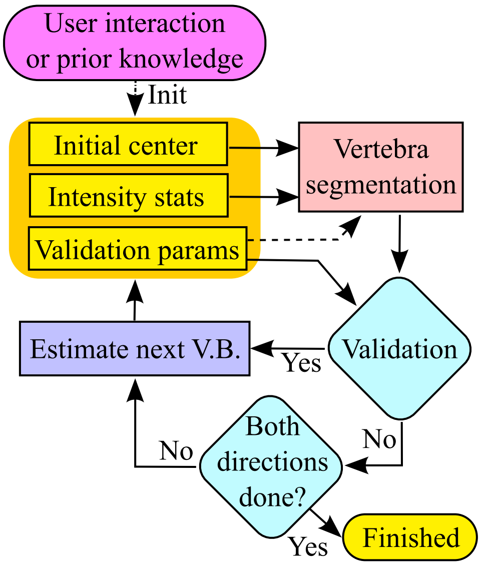


Figure 5.1: High-level diagram of the segmentation system

5.1 Spine Segmentation System

The main segmentation loop has improved initialization and pose estimation. Prediction of adjacent vertebra position and segmentation correctness measure are the same.

Initialization Two different types of initialization are implemented: center picking and freehand outline which is same as before. Center picking makes initialization even easier than before.

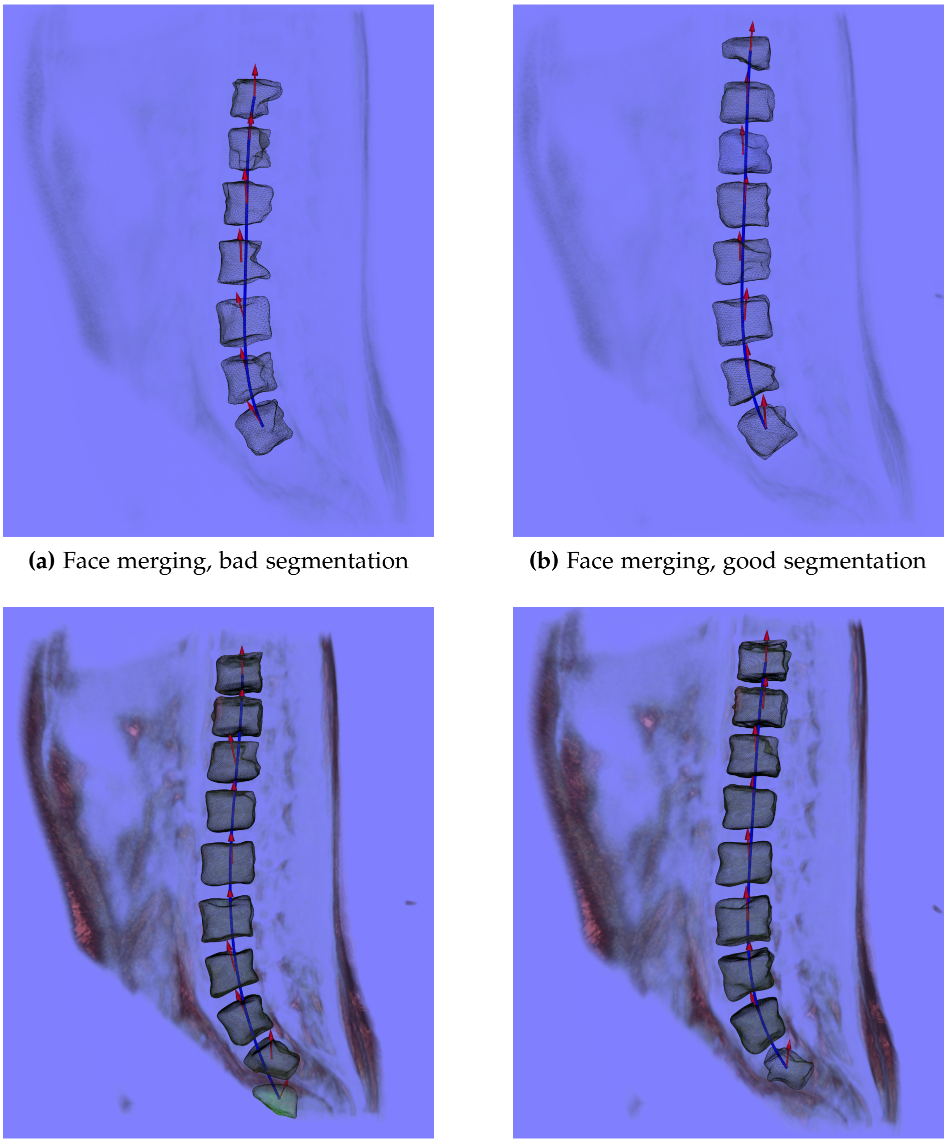
From either initialization the algorithm extracts the center coordinates and takes a sample of intensities. In the case of center picking, intensities are taken from a circle with 2.8 mm radius. This radius needs to be large enough to be easily click-able even on images with low resolution where this circle is just a few pixels in diameter. Choice of 2.8 mm was determined experimentally.

The outline initialization additionally gives an approximate size, otherwise the average human vertebra size is taken [MSM$^+$08].

Pose Estimation Using Iterative Closest Points The orientation of the vertebral body is determined by fitting to it a hand-crafted vertebral body shape (the same one which is used initially for inflation, Fig. 4.8 on page 65) using the iterative closest point (ICP) algorithm.

The initial position for ICP is determined by translating the average shape center to the segmented vertebral body center. The initial orientation of the average vertebral body shape is taken from the previous vertebra's fitted orientation (except the initial vertebra, where the anatomical upright orientation is used). The initial scaling of the average shape is determined from the radius of the inflated mesh.

The initialization brings the average shape close to the segmented one, which makes ICP robust. ICP is then allowed to optimize position, rotation and scaling to find the best fit. Vertebra's main axis (head-tail axis) is extracted from the orientation of the fitted shape. Overall, it is a noticeable improvement over face merging (Fig. 5.2 on the following page).



(a) Face merging, bad segmentation

(b) Face merging, good segmentation

(c) ICP, bad segmentation

(d) ICP, good segmentation

Figure 5.2: Visual impression of poses (red arrows) estimated using face merging and iterative closest points. "Good" and "bad" segmentations had different initializations. Dataset DzZ_T1 from Tab. 7.1 on page 102

.

5.2 Multi-Feature Boundary Classification

The vertebral body boundaries are estimated using multiple features, which are classified into probabilities of the voxel v being at a boundary, and combined to a final probability $p(v)$ using the weighted average (see Fig. 5.3). Using multiple classifiers and then combining them is known to improve results and robustness [KHDM98]. Three edge-based and two intensity-based features are used. As intensity-based features require local, per-vertebra intensity statistics, a small 2D neighborhood is examined around each of the detected vertebral body centers. After removing outlier values, minimum and maximum are used for further processing.

Due to the per-vertebra statistics, the boundary probability map needs to be calculated for each vertebra separately. This is done inside a bounding box which is twice as large as the detected diameter of the vertebra.

The edge features are based on LH (low-high) values [ŠBSG06], Canny edges, and thresholded gradient magnitudes. The difference between low and high values, L, H indicates proximity to a boundary. The boundary probability is deduced from these values and the current voxel v's intensity I by $p_{LH}(v) = ((H - I) - (I - L))/(I_{max} - I_{min})$, where I_{max} and I_{min} are the maximum and minimum intensity in the dataset (see Fig. 5.3 and Fig. 2.18 on page 25). Note that intensities are shifted so they start at 0 if it is not the case, before any other processing is done.

Gradient magnitudes and Canny edges are multiplicatively enhanced using the structure tensor as described by Fernández and Li [FL03] to improve the detection of 2-manifold edges and suppress one-dimensional features. Structure tensor is Jacobian of image gradients, and its eigen

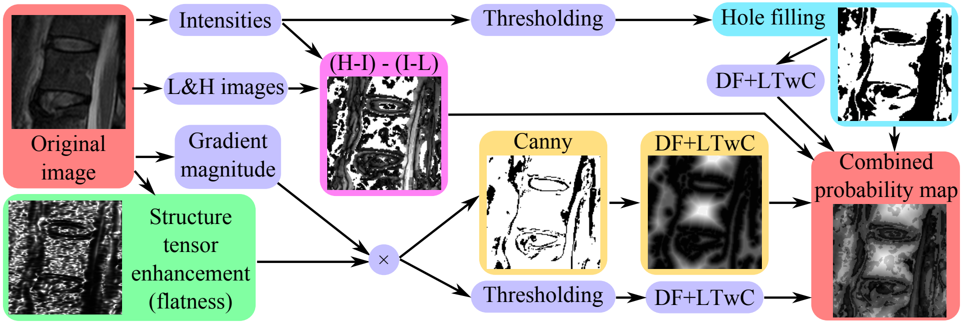


Figure 5.3: Features explained on T12 vertebra from F02 dataset. In all images except original black indicates edges, whereas white represents more homogenous regions like a vertebral body's interior.

analysis allows characterization of the local structure in the image. For eigenvalues μ_i, relationship $\mu_1 \gg \mu_2 \approx \mu_3$ is indicative of a 2-manifold edge.

For both gradient magnitudes and Canny edges, the respective probability is derived by applying a distance field (DF) and a linear transform with clipping (LTwC), see Fig. 5.3 on the preceding page.

The two intensity-based features work directly on the thresholded MRI intensities. The thresholds are taken from the intensity statistics collected in the proximity of the detected vertebral body centers, i.e. the minimum and maximum values after outlier removal. Holes are filled by morphological closing using a 3-voxel-diameter ball structuring element.

The binarized intensities (each voxel either 0% or 100%) are used as one feature. The second feature is the distance field constructed on this binary image, effectively treating 0→1 transition as another edge feature (therefore using DF+LTwC transformation). Thus both sharp edges (binary 0/1) and a smooth edge approach (normalized distance field) are incorporated for mesh inflation. It can be determined by visual inspection that after combining the probabilities, true edges usually end up having a boundary probability of around 90%.

The distance field constructed on binarized intensities is the most informative feature. After parameter optimization (Sec. 6.5 on page 96) it gets a weight of around 30%-35% while the weights of the other four features are around 10%-20%. Some other features, such as σ-weighted deviation from mean initialized intensity, have been investigated and discarded due to small benefit.

5.3 Soft-Boundary Inflation

Mesh inflation process is similar to the previous chapter. The big changes are how the inflation step repositions vertices and establishment of true equilibrium.

The following steps are performed iteratively:

1. Move the vertices (outwards) towards the boundary with no regard to subdivision rules.

2. Normalize the subdivision hierarchy by moving vertices so that they comply with subdivision rules.

3. Recalculate the polyhedron center from the polyhedron shape to account for different inflation speeds in different directions.

4. Stop when convergence is detected.

Constrained Subdivision Hierarchy is normalized using a global least squares optimization. Only the base mesh vertices are independent, i.e. the base mesh is the control mesh. Both of these are changes are done in order to make behavior of individual vertebra segmentation more predictable and overall segmentation system more robust.

The inflation examines each vertex separately. Depending on the boundary probability, the vertex is either inflated or deflated along the radial (center-vertex) direction. The step size is equal to the minimum voxel spacing, i.e. 0.5–1.2 mm for the examined datasets.

Initially, a vertex v is in the interior and will inflate as long as the boundary probability $p(v) < 0.5$. This way noise inside the vertebral body is ignored. When $p(v) \geq 0.5$ the sign of the probability derivative $\frac{\delta p}{\delta \vec{r}}$ in inflation direction $\vec{r}$ decides on whether to inflate or to deflate, thereby moving the vertex towards the maximum boundary probability.

Convergence is achieved when the average center-surface distance ("radius") stops increasing, i.e. the radius from the current iteration is not larger than the radii from the previous two iterations. An additional check is whether the radius is 50% greater than the last vertebra. This saves computation time in case of mis-segmentation.

An uneven distribution of vertices over the surface is detected using the standard deviation of the edge lengths ($\sigma > 1\,\mathrm{mm}$). In this case, the segmentation is restarted at the current center. Experiments show that one to two restarts with a new center are sufficient to resolve the problem. This mostly happens with the second vertebra, because intervertebral distance at that time is only an estimate.

5.4 Conclusion

In this chapter a multiple-features boundary estimator was introduced along with many other minor improvements. A true equilibrium conditions were established instead of relying on stalled inflation.

Results are improved, both the detection rate (63%-75% up from 50%-72%) and Dice similarity coefficient (77%-78% up from 70%-75%). It now takes about half a minute to segment the whole dataset. This is an order of magnitude longer than the previous version (from chapter 4). This just indicates that the law of diminishing returns is at play when human precision levels (86%-96% DSC) [EBK$^+$11a] are approached.

A drawback from the previous chapter remains: one wrongly segmented vertebral body can foil subsequent segmentations, thus detection rate (75%) is far from 100%.

Chapter 6

Initialized Parallelized Segmentation

This chapter reorganizes the building blocks introduced previously, and adds many more. It has been published as [ZVE$^+$14], and is the pinnacle of this thesis.

The main problem of the system described in the previous chapter is the dependency of the detection ratio on the accuracy of previously segmented vertebral bodies. One bad segmentation breaks the detection chain, leading to low (75%) vertebra detection ratio.

Detecting all vertebral bodies before segmentation is the main improvement in this chapter, but there are other improvements as well:

- Automatic detection of vertebral body centers and sizes based on a Viola-Jones [VJ01] method with novel candidate filtering, which enables
  - a global prediction of all vertebra centers, which improves the overall segmentation and diagnosis process,
  - parallelized computation of per-vertebra features, classified image merging and constrained inflation, and
  - a novel size-goal force, which guides the segmentation to detected vertebral body size, thus improving segmentation accuracy.
- Optimization of system parameters through training on manually segmented datasets.

- Center+mesh representation of segmentation allows easy diagnostic measurements of the vertebral shape in order to detect scoliosis, spondylolisthesis and crushed vertebrae.

6.1 Overview

This section gives an overview of the three major steps depicted in Fig. 6.1.

Vertebra Detection: In this stage, the vertebral centers are detected using a Viola-Jones method [VJ01]. The initially detected centers are filtered in order to remove false positive and false negative detections and thus to minimize the required user intervention. Finally, the sizes of the user-picked vertebral bodies are estimated (see Sec. 6.2 on the next page).

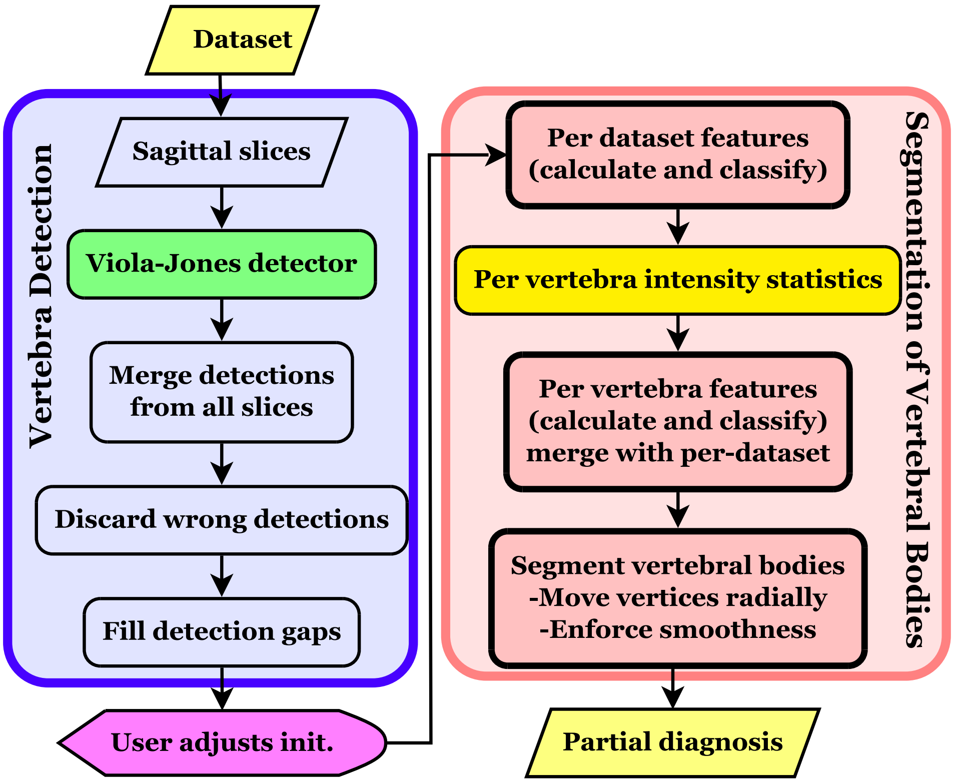

Dataset
Vertebra Detection
Sagittal slices
Viola-Jones detector
Merge detections from all slices
Discard wrong detections
Fill detection gaps
User adjusts init.
Segmentation of Vertebral Bodies
Per dataset features (calculate and classify)
Per vertebra intensity statistics
Per vertebra features (calculate and classify) merge with per-dataset
Segment vertebral bodies -Move vertices radially -Enforce smoothness
Partial diagnosis

Figure 6.1: High-level diagram of the full system. Boxes with thick black outlines are computationally intensive. Green and red boxes employ parallel computation on the CPU.

Segmentation of Vertebral Bodies: The multi-feature boundary classification is from previous chapter is slightly improved. The size parameter estimated in the detection stage is utilized in the form of size-goal force to increase robustness (see Sec. 6.3 on page 90). Segmentation and calculation of per-vertebra features is parallelized.

Diagnosis: Based on the segmentation result, position and volume of the vertebral bodies are used to deduce disease-specific parameters which drive the final diagnosis (see Sec. 6.4 on page 92).

Please note, that in contrast to the previous chapter, all vertebral centers are detected in a first step. This allows the parallel execution of several downstream processing steps (Fig. 6.1 on the facing page).

Furthermore, a standard technique for *parameter optimization* was applied in order to set various algorithmic parameters like thresholds; see Sec. 6.5 on page 96.

6.2 Initial Vertebra Detection

This section describes the detection of all vertebral body centers, used as initialization for the segmentation step. After applying the Viola-Jones method to detect candidate vertebrae (Sec. 6.2.1), the candidates are filtered (Sec. 6.2.2 on page 88) and minimal human intervention is needed to complete this step (see Sec. 6.2.3 on page 90). The Viola-Jones method was chosen for its speed (a few seconds), because other approaches like the Generalized Hough Transform or that of Vrtovec et al. [VLP05] are relatively slow (minutes to hours [ŠLPV10]).

This section uses the coordinate system induced by the acquisition of sagittal MR images, i.e. with respect to the human body, x is front-to-back, y is top-to-bottom and z left-to-right.

6.2.1 Viola-Jones Detector

The training data for the (single) Viola-Jones detector is derived from the reference segmentations utilizing all training datasets jointly. A bounding box is determined for each vertebral body on each slice of all datasets used for training. Expanding this bounding box to a square serves as a positive sample for detector training. All positive samples are resized to

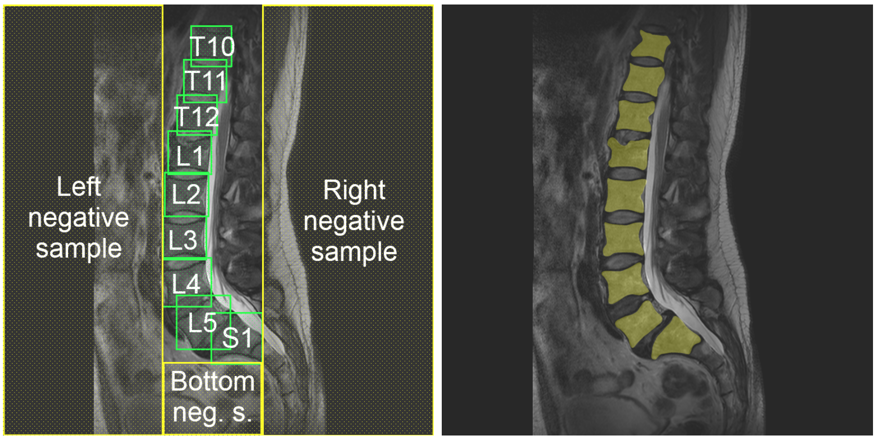


Figure 6.2: Left: training samples for Viola-Jones detector. Green boxes are positive samples, yellow boxes are negative samples. Right: ground truth overlaid over the original slice (dataset F02, slice 9).

16x16 pixels and have 10 additional, slightly rotated versions (±15°). The Viola-Jones detector has a built-in scaling, and the square size (20-50 mm) was targeted to adults.

Left, right and bottom rectangles of each slice, which are not covered by positive samples, are added to negative samples (Fig. 6.2), along with all slices which contained no part of reference vertebrae. Additionally some hand-chosen rectangles very near the vertebral bodies are added to the negative samples, otherwise some nearby structures such as aorta or spinous processes get mistaken for vertebral bodies.

A boosted cascade with 40 stages, each stage with a minimum hit rate of 0.998 and target maximum false alarm rate of 0.5 (limited by the maximum stage size of 100 tree stumps) has been applied. The number of positive and negative samples used was 7500 each. The OpenCV implementation of the Viola-Jones detector [VJ01, LM02] was used.

The initial vertebra detection based on the Viola-Jones detector [VJ01] is done on all sagittal slices independently. If a dataset is not acquired as sagittal slices, it is reformatted into sagittal slices. The candidates' in-slice positions provide the (x, y) coordinates and approximate size (vertebral body diameter) of the vertebral body candidates, the $z-$coordinate of the center is derived from the weighted average of the slice-adjacent candidate centers.

The variety of datasets the Viola-Jones detector can handle depends on its training set. The training set used has a lot of variety – T_1, T_2 and TIRM

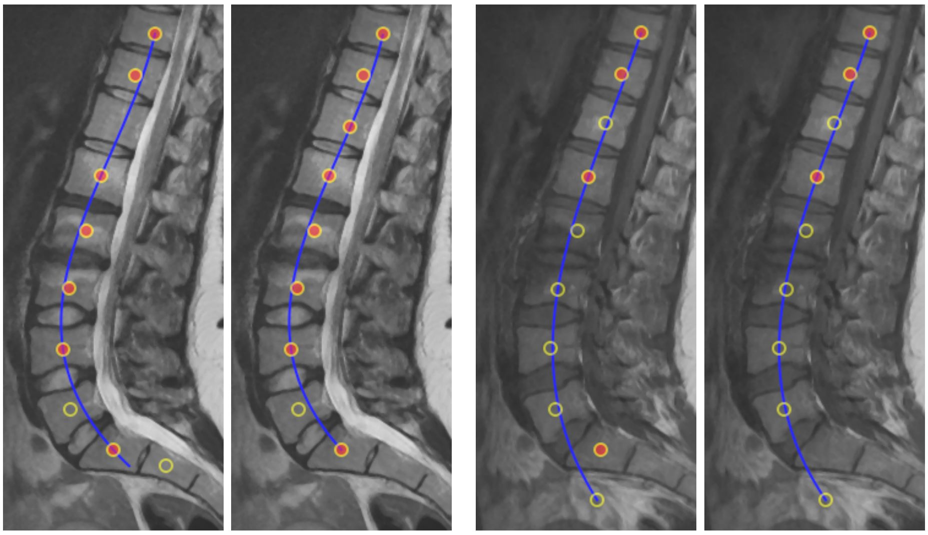


Figure 6.3: Vertebrae detection (red-filled points are candidates on the current slice, empty points are lying on other slices, the fitting curve is shown in blue; see Tab. 7.2 on page 103 for information about the datasets): Left, the unfiltered center candidates of dataset AKs5 and, middle left, a fully successful filtering (the T12 gap was filled and the unwanted S2 vertebra detection was removed). Unfiltered center candidates of dataset AKs6 (middle right) with wrong result – 1 false negative and 1 false positive after filtering (right). In 38% (10/26) of datasets no user intervention is required.

sequences, and a range of TE, TR and other parameters. If an untrained MRI sequence is among the unseen test data, the method does not break down, there is only an increase of detection errors. When AKa3, AKs7 and AKs8 datasets (see Tab. 7.1 on page 102) were removed from the training set (leaving only 6 T_2 datasets for training), the number of detection errors increased by 24%, from 21 to 26 out of 240 vertebrae. However if new MRI sequences need to be handled with low amount of errors, the detector needs to be re-trained with the expanded training set.

Note, that the result of the Viola-Jones detector is imperfect, containing spurious detections (false positives) and missing vertebral bodies (false negatives); see Fig. 6.3.

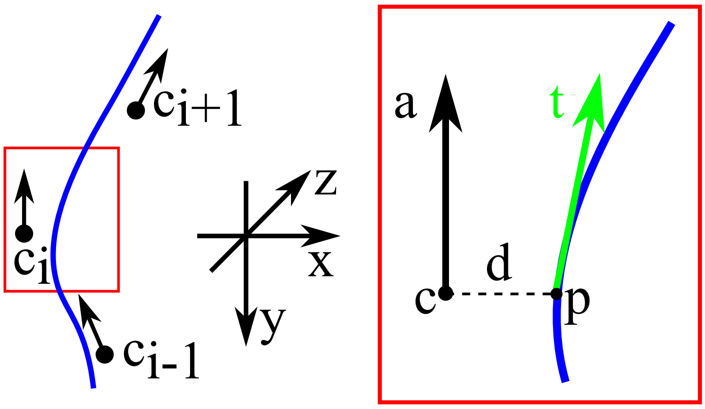


Figure 6.4: Spine center curve fitting. Left: vertebra centers, axes, and the fitting curve. Inset on the right: vertebra i shown in detail (index i omitted). Y axis is used as abscissa and the X and Z polynomials are dependent on it. The distances d_i between vertebra candidate centers c_i and their projections onto the abscissa p_i are minimized using the L_1-norm.

6.2.2 Detection Filtering

Filtering along z axis is done as part of Viola-Jones algorithms' rectangle filtering and merging. Detected rectangles from all slices are put through this step together, and the largest of the many overlapping rectangles is kept as a vertebra candidate. The $z-$coordinate of this vertebra candidate is the average of all the overlapping rectangles from different slices.

In order to remove wrong detections and to fill missed vertebrae, a spine center curve is fitted to all detected vertebra candidates in 3D (see Fig. 6.4). The S-shaped lower spine has two distinct curvatures and therefore a third order polynomial is an appropriate curve model. At this stage misdetections along the x-axis are more important than precise fitting along the z-axis. Since robustness to false positives is needed, precise fitting to the scoliotic-shaped spine (which requires a fourth order polynomial) was sacrificed.

The distances of all vertebra candidate centers to the spine center curve are calculated and the one with the largest distance is removed, if it is above the approximate radius of that vertebral body candidate. Then a new spine center curve is fitted and the procedure is repeated for as long as there are candidates to be removed. This outlier removal is done in sagittal plane (using 2D planar distances).

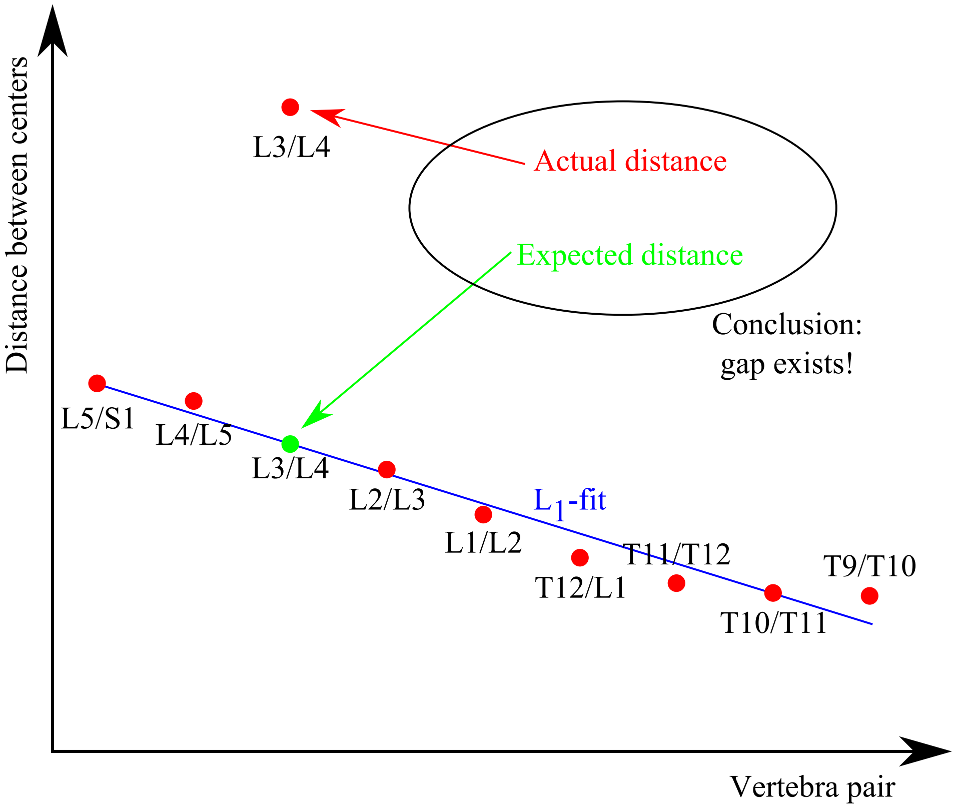


Figure 6.5: Detection gap analysis uses distances between centers of adjacent vertebra pairs. The change in distances along the spine is exaggerated here.

The topmost and bottommost vertebra detections are frequently fitted well by the polynomial, even if they are unwanted false positives. Therefore, they are removed in case they are located far from their adjacent vertebra.

Furthermore, the frequency of candidates along the fitted spine center curve is examined. The Theil-Sen linear estimator [The92] is used to determine robustly the linear function of expected distance (Fig. 6.5) between vertebral body centers – the distances increase towards the lower end of the vertebral column. If a particular distance is below 75% of the expectation from this fitting, meaning that an extraneous vertebra is detected there, the corresponding candidate is removed. Similarly, gaps larger than 150% of expected distance are filled with the best-fitting number of evenly distributed new centers.

Still missing vertebrae, e.g. the topmost or bottommost, need to be added manually in the next step.

6.2.3 Manual Correction and Labeling

Since a correct setting of the initial vertebral body centers is crucial for the following segmentation and diagnosis, the result of the filtered detections is presented to the user for verification. The user can reposition (mouse drag), add (left click) and remove points (right click). This is done on sagittal cross-sections, even if the dataset was acquired along some other body axis (e.g. coronal). The user chooses cross-section with a slider, and the middle slice of the detected vertebral body centers is preselected by the program.

New vertebral body centers should be placed on a slice which is approximately central to the vertebral body. The size of these new vertebral bodies is linearly interpolated from the sizes of neighboring vertebral bodies. The user also chooses a label for the bottommost vertebra (usually S1) from which other vertebra labels are calculated, in order to have a correct human-readable diagnosis result.

6.3 Segmentation of Vertebral Bodies

Inflation is similar to the previous chapter. However, a size-goal force is added, which drives the segmentation towards the estimated size detected by the Viola-Jones method.

The following steps are performed iteratively:

1. Move the mesh vertices radially towards the boundary with no regard to subdivision rules. Vertices are inflated or deflated, depending on the merged probability image.
2. Normalize the subdivision hierarchy (Sec. 4.4 on page 63) by moving vertices so that they comply with subdivision rules.
3. Recalculate the polyhedron center from the polyhedron shape to account for different inflation speed in different directions.
4. Stop when convergence is detected.

6.3.1 Inflating the Mesh

Vertebral body detection gives us not just the center but also an approximate vertebral ”radius”, and this can be used to steer the segmentation process towards that size by modifying the equilibrium conditions.

Depending on the boundary probability and the size-goal force, the vertex is either inflated or deflated along the radial (center-vertex) direction. The step size is equal to the minimum voxel spacing, i.e. 0.5–1.1 mm for the examined datasets.

Initially, a vertex v is in the interior and will inflate as long as the boundary probability $p(v) < 0.5 + f_{sg}$, where f_{sg} is the size-goal force. When $p(v) \geq 0.5 + f_{sg}$, the sign of the probability derivative (adjusted by the size-goal force) $\frac{\delta(p+f_{sg})}{\delta\vec{r}}$ in inflation direction $\vec{r}$ decides on whether to inflate or to deflate, thereby moving the vertex towards the maximum boundary probability. By using the boundary probability gradient only when the probability is high (vertex near the boundary), noise inside the vertebral body does not block the inflation. Convergence is achieved when the average center-surface distance ("radius") stops increasing.

$$f_{sg} = \kappa \cdot \tanh\left(\frac{r_{detected} - r_i}{r_{detected}}\right) \cdot \frac{s_{min} - (r_i - r_{i-1})}{s_{min}}$$

is the size-goal force designed to overcome local minima which sometimes occur at smaller sizes than anatomically possible. $r_{detected}$ is the approximate radius detected by the Viola-Jones algorithm – the goal radius, r_i is the radius in the current iteration and s_{min} is the the minimum voxel spacing. κ ($\approx$ 0.1) is a parameter steering the force influence and is optimized for segmentation accuracy (see Sec. 6.5 on page 96). f_{sg} only gives preference to a certain size, it does not guarantee it.

When $r_i < r_{detected}$, f_{sg} increases inflation pressure, and when $r_i > r_{detected}$, f_{sg} creates deflation pressure. Out of precaution this factor is limited to (-1,1) by the *tanh* function, which is almost linear around zero and gradually turns into a clipping function from about 2 onwards.

The last factor is there to suppress the size-goal force during normal inflation. When the radius growth between the last two iterations is large ($r_i - r_{i-1}$ close to s_{min}), this factor lowers the influence of f_{sg}. This is done in order to respect the edge probabilities, and to give influence to f_{sg} only when the inflation enters a local minimum (small $r_i - r_{i-1}$) which is far from the desired size. This factor is always positive, because $r_i - r_{i-1}$ cannot be bigger than s_{min} which is the maximum possible step size.

If the initial center estimate for the segmentation is very close to a boundary, the surface will inflate much more in the opposite direction. This will result in a highly uneven distribution of vertices over the surface, which is detected using the standard deviation of the edge lengths ($\sigma > 1\,\text{mm}$). In this case, the segmentation is restarted at the current center; this sometimes occurs with S1 vertebra or with some thoracic vertebrae with low contrast, when the segmentation leaks outside of the true vertebral body.

6.3.2 Constrained Subdivision Hierarchy

The global least squares optimization is used, minimizing not only the vertex position correction as before, but also an edge length deviation. The latter stimulates an even vertex distribution.

Another change is that the number of control levels (number of levels, the vertices of which are independent) is defined by how many levels exist overall: $n_{control} = \lfloor \frac{n_{total-1}}{2} \rfloor$.

If the control levels contain many vertices, overshooting effects start to appear, which is frequent in interpolating schemes. Thus, the least squares fitting scheme is combined with the heuristic hierarchy normalization which implicitly smooths the mesh. The influence of the heuristic scheme is controlled by a weight parameter (smoothFactor) which is part of the optimization (see Sec. 6.5 on page 96).

6.4 Diagnosis

The focus is on three diseases which can be diagnosed from segmentation of vertebral bodies alone, i.e. on scoliosis, spondylolisthesis and vertebral fractures.

Spinal canal stenosis and slipped disc, which were present in some datasets, cannot be diagnosed from vertebral body segmentations. Hemangioma, in spite of distinct appearance in MR images and being a significant complicating factor for vertebral body segmentation, has little to none clinical relevance.

6.4.1 Scoliosis

Normally, the vertebral column has curves in front-back (sagittal) plane, having a shape of two concatenated letters “S”. In the left-right direction (coronal plane), healthy spine is generally straight with only minor asymmetry – similar to facial asymmetries. If there is, however, a curve in the coronal projection of the spine, that condition is known as scoliosis.

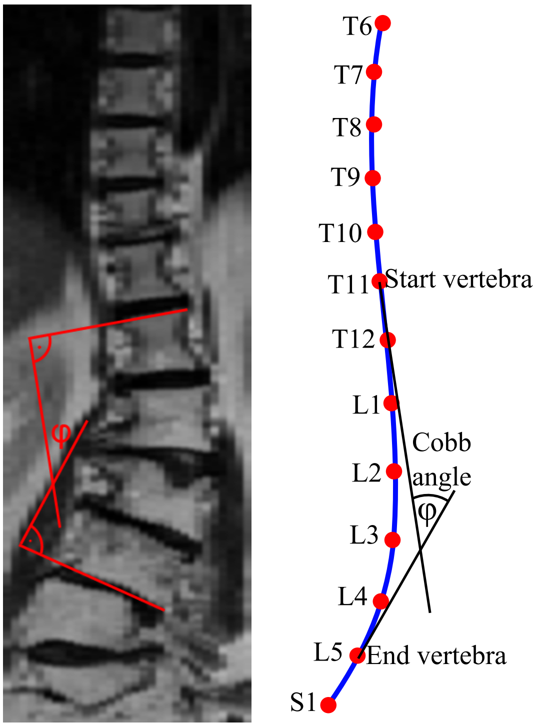


Figure 6.6: Scoliosis is visible on coronal slices, looking at a person from front or back. On the left is a real image (dataset C002) with manual measurement lines. On the right is how scoliosis is calculated using the fitted centerline and its tangents at each vertebral body center. In this case: $\varphi = 32.6°$.

According to standard clinical practice the severity is determined by spinal curvature measured in degrees, called Cobb angle [Aeb05]. Fig. 6.6 (left) depicts how the Cobb angle φ is measured. If the Cobb angle is above 20° tracking is warranted, and above 30° treatment is considered.

For determining the degree of scoliosis, the spine center curve (fourth-order polynomial) is fitted to segmented vertebral body centers (see Fig. 6.4 on page 88). The Cobb angle [Cob48] is evaluated between all possible pairs of vertebrae, using tangents of the fitted curve taken at vertebra center positions. The biggest Cobb angle found is reported along with the vertebrae between which it is occurring.

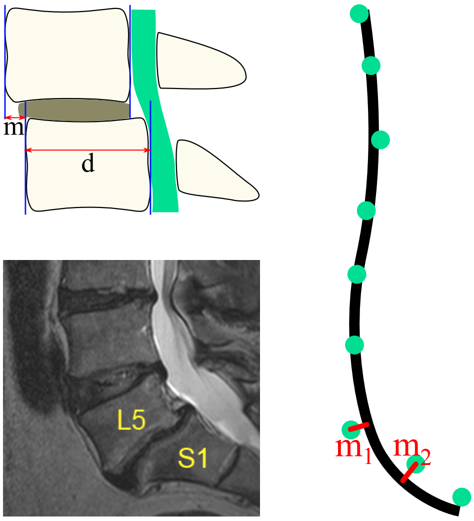


Figure 6.7: Spondylolisthesis schematic on a sagittal slice. The usual medical calculation method (top left) is not always easy to apply, especially for L5/S1 pair (bottom left, dataset S01). Right: the way misalignment is calculated ($m_1 + m_2$) using the fitted centerline and vertebral body centers (L5/S1 misalignment=14%).

6.4.2 Spondylolisthesis

Spondylolisthesis is misalignment of vertebrae which results in change of posture and can include pain. It is usually divided into low-grade which does not usually require treatment, and high-grade (above 25% [Sar87]).

Measurement of the degree of spondylolisthesis in clinical practice $S_{\%} = \frac{m}{d}$ is adapted for manual measurement on a single sagittal slice or 2D X-ray (Fig. 6.7 top-left) [WW83] .

Here, the degree of spondylolisthesis is determined by calculating the distance of vertebral body centers to the fitted spine center curve and dividing it by vertebral body diameter: $S_{\%} = \frac{m_1+m_2}{d}$ (Fig. 6.7 right). This measure is more robust and stable with respect to the segmentation imperfections and does not require the analysis of the shape of each vertebral body so it is also computationally efficient.

6.4.3 Vertebral Fracture

Crushed vertebral bodies are usually due to mechanical stress applied to a weakened bone (typically a result of osteoporosis). In clinical practice

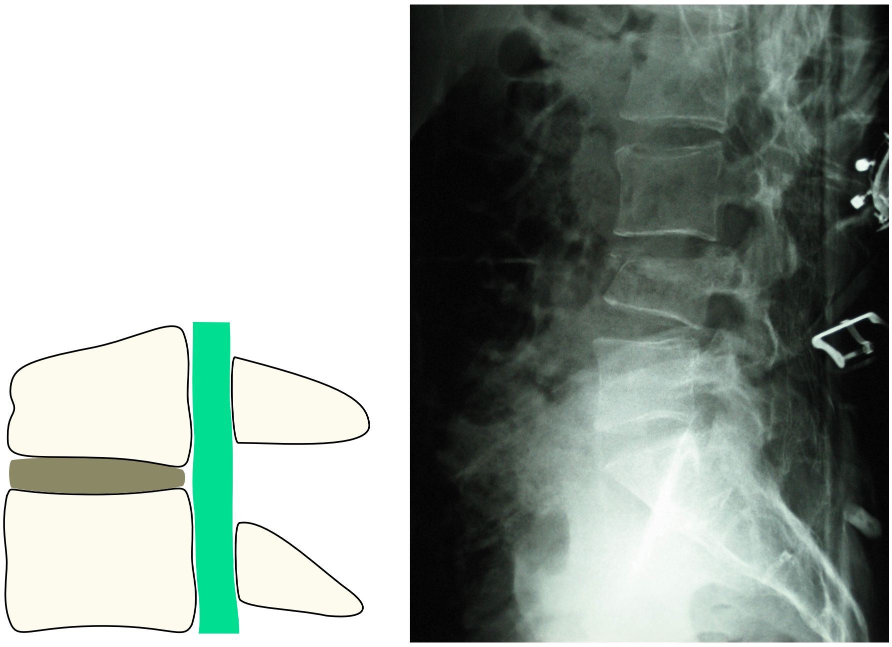


Figure 6.8: Crushed vertebral body. Left: schematic view. Right: X-ray image, reproduced from Wikipedia.

diagnosis is usually established based on qualitative impression [LRDG04]. Several different approaches have been used in medical research [LRDG04], but they are tailored for manual measurements based on 2D sagittal X-rays or single sagittal slices [WLJG95], see Fig. 6.8.

To diagnose crushed vertebrae a third-order polynomial is fitted to vertebral body volumes using the L_1-norm. That way outliers do not influence the fitting, so the detection is robust. Besides the actual volume of each vertebral body we now also have a volume expectation $V(i) = P(i)$, by evaluating the fitted polynomial at each vertebra index (similar to analysis of candidate frequency along the center-line curve, see Fig. 6.9 on the following page). Analyzing the difference between them results in diagnosis. If the actual segmented volume is below 80% of the polynomial-derived volume expectation, the vertebra is indicated as crushed.

80% of the expected volume parameter (20% crushedness) was obtained empirically. The crush-measure routinely reaches 10% in non-crushed (fracture-less) datasets. Average crushedness over all the datasets is 2.46% with standard deviation σ =3.87%. 5σ ≈20% was chosen as the threshold.

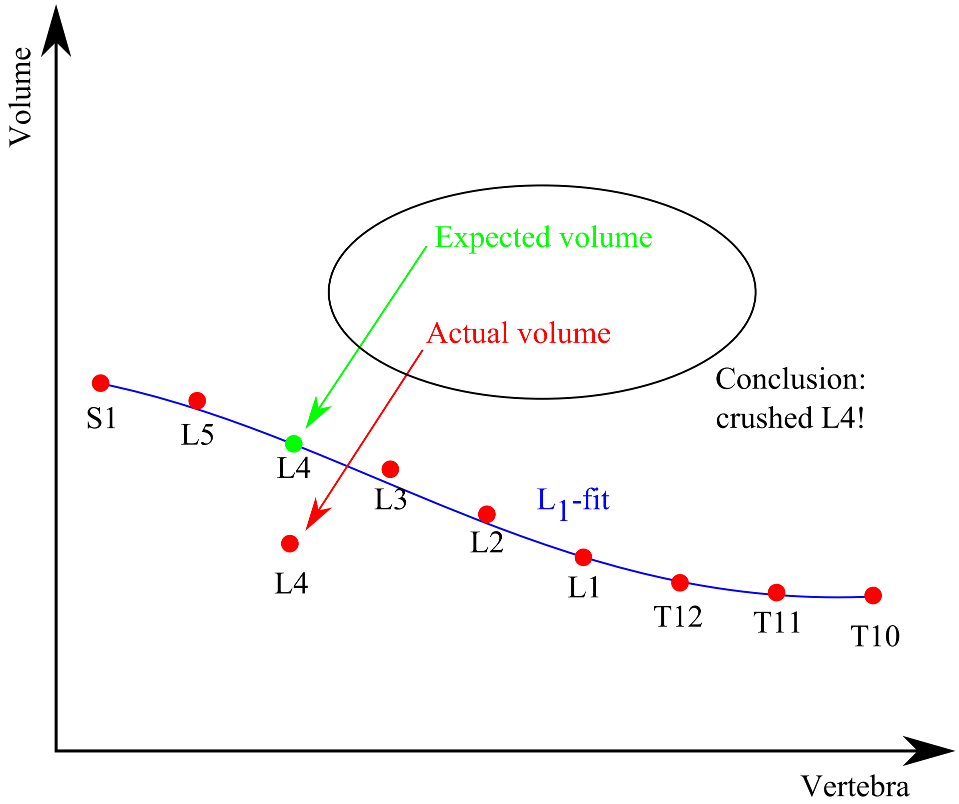


Figure 6.9: Crushed vertebra analysis uses volumes of segmented vertebrae. The change in volumes along the spine is exaggerated here.

6.5 Parameter Optimization

The Viola-Jones detector clearly relies on training data. The rest of the segmentation system is not dependent on any kind of training. However, since the whole system has some $N = 16$ parameters p_i, such as thresholds or weights (see Tab. 6.1 on the next page), which should not be exposed to the user, their values are determined via an optimization process.

The objective function is defined via the Dice Similarity Coefficient (DSC):

$$f(p_1, ..., p_N) = \sum_{x \in T} DSC(x; p_1, ..., p_N)$$

where T is the set of datasets used for training.

A single evaluation of the function consists of segmenting all the training datasets with the given parameter values and returning the sum of all DSC values. A local, derivative-free COBYLA algorithm [Pow94] is used to

Parameter name	Used in/for	Initial	Optimized
bilateralDomainSigma	Feature calc.	10	4.6
bilateralRangeSigma	Feature calc.	3.0%	1.0%
structureTensorSigma	Feature calc.	1	1.7
cannyVariance	Feature calc.	2	1.23
lhEps	Feature calc.	1.0%	1.5%
multBinaryThreshold	Feature calc.	10.0%	27.6%
cannyLowerThreshold	Feature calc.	5.0%	4.3%
cannyUpperThreshold	Feature calc.	15.0%	24.0%
DFonThreshFeatureW	Boundary est.	20.0%	33.3%
DFonCannyEdgesFeatW	Boundary est.	20.0%	16.3%
DFonEnhancedGradFW	Boundary est.	20.0%	14.0%
LowHighDiffFromImage	Boundary est.	20.0%	15.6%
MinMaxThresholdedImg*	=1-$\sum w$	20.0%	20.8%
percentMoreInflated	Inflation	5.0%	7.8%
smoothFactor	Inflation	30.0%	44.4%
edgeConstraintImportance	Inflation	3	8.21
sizeGoalForceKappa	Inflation	10.0%	10.2%

Table 6.1: Optimized parameters. MinMaxThresholdedImg is optimized implicitly, because the sum of feature weights has to be 1.

maximize the function f. This optimization increases the average DSC by ≈1% and reduces distance error by ≈5%. 9 T_2 weighted datasets get a reduction of DSC, and 10 get an increase. The average of T_2 datasets increases from 78.4 to 78.5, but the average of non-T_2 datasets increases from 76.7 to 81.3. In other words, the majority of the gain is for non-T_2 datasets. This can be explained by the fact that initial parameters were more suited for T_2 datasets, because of higher availability for testing during algorithm development.

6.6 Conclusion

A rounded system which goes beyond the state of the art was presented in this chapter.

This system addresses the drawbacks (complex initialization, long running time, special MRI sequences) of other systems in literature and allows for comparably fast and robust segmentation of pathological spine and vertebra shapes on a large variety of datasets (both T_1 and T_2 weighted).

This vertebral body segmentation system is inflation-based and incorporates a novel constrained subdivision surface approach for smoothness control. A multiple features boundary estimator along with size-goal force provide robustness. Automatic vertebra center detection reduces initialization time and allows parallel segmentation. The segmentation takes about 70 seconds for the whole dataset, which is significantly better than the other spine MRI 3D methods (which were described in Sec. 2.10 on page 32).

This system was tested on a larger set of 234 vertebral bodies (199 from pathological, 35 from healthy datasets) than prior work (Hoad and Martel: 30 vertebrae, Štern et al.: 75 vertebral bodies; Ayed et al.: 75 vertebral bodies, Neubert et al.: 132 vertebral bodies from healthy volunteers, Davatzikos et al.: 84 vertebral bodies from healthy volunteers). Most importantly this approach works on a large variety of datasets, whereas others restrict themselves to just one type of datasets: Štern et al. – T_2 weighted, Davatzikos et al. – T_1 weighted, Ayed et al. – T_2 weighted, Neubert et al. [NFE$^+$12] – T_2 SPACE sequence, and Hoad and Martel – T_1 FISP sequence.

Chapter 7

Spine Segmentation Results

Rather than having detection and precision results spread over three chapters, which would force the reader to leaf the pages (or scroll if viewed on a computer) in order to compare them, all the results are merged into this chapter.

Test setup and the data used for evaluation are described first. Results are then given for:

- Vertebra detection accuracy – how many of the vertebral bodies visible in the dataset have been detected?
- Segmentation accuracy – how good is the agreement between automatic segmentation and manual reference?
- Diagnosis accuracy – have the diseases been correctly diagnosed?

7.1 Test Setup

All tests were executed on a machine with Windows7 64-bit operating system, Intel Core i7-920 (4 cores @ 2.67GHz) processor and GeForce GTX480 graphics card. This system also had 16GB of RAM, but that was important only for the two datasets with 120 million voxels. This system was not developed with memory conservation in mind, because the routine datasets are of relatively small size (5-10 million voxels) compared to today PC's RAM capacities (4-16 gigabytes).

Besides comparing to methods developed specifically for spine MRIs, all available datasets were subjected to the power watersheds of Couprie et al. [CGNT09] as one representative of a general segmentation

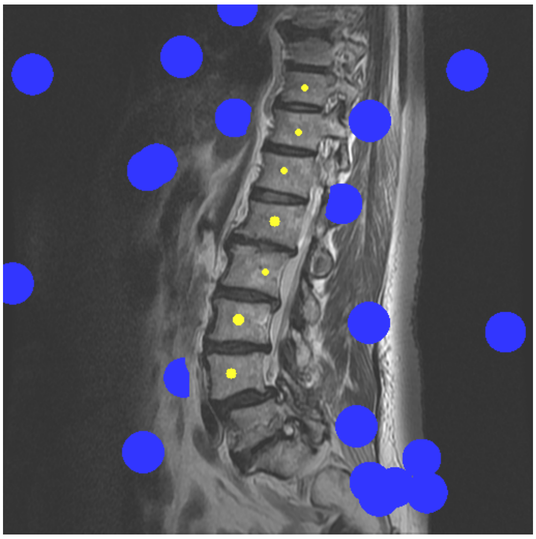


Figure 7.1: Power Watersheds initialization. One cross-section of seeds overlaid on image (dataset Ble). Blue are background seeds, yellow are seeds of vertebral bodies. It is noticeable on V.B. seeds that they have different Z-positions (they appear to be of different sizes on this cross-section).

method. The source code for this method is publicly available at `http://powerwatershed.sourceforge.net/`, so it could be run on the same datasets as the systems described here. The execution time of power watershed is more dependent on image size than the systems presented here, because unimportant parts are not mostly ignored like in the systems introduced in this thesis. The execution time of power watersheds on a typical dataset (512x512x16) is about 20 seconds.

The graph cuts method (and its extension power watersheds) is well suited for interactive segmentation, which is a mode where additional seeds are placed until a satisfactory segmentation is achieved. It is not suitable for semi-automatic segmentation, because it requires a lot of seeds and can lead to implausible results which requires additional seed placement. The manual initialization of this method in 3D is quite painstaking so I opted for automatic seed creation, which is derived from ground truth data, for the purpose of a thorough comparison.

One seed was placed into each vertebral body on a random position near the center, and twice as many seeds into the background (Fig. 7.1). Since the background seeds were large (20 voxel radius), they were clipped by a safety region of interest around the vertebral bodies. This safety region was created by morphologically dilating ground truth vertebral body masks with a 20 voxel radius spherical structuring element (Fig. 7.1 on the facing page). There was one such initialization for each vertebra in the dataset.

The surface distance errors were measured using the Metro mesh comparison tool [CRS98]. The distance error is measured by iterating through vertices and sample points on polygons of the segmentation surface and measuring the distance to the closest point of the reference surface.

To get a relative distance error, the absolute distance error (measured in millimeters) is divided by voxel size (also measured in mm). In case of anisotropic voxels, the distance error is divided by the edge of isotropic voxel which is equivalent by volume, $A_{iso} = \sqrt[3]{s_x s_y s_z}$ where s_* are voxel spacings along the image axes. The average A_{iso} for 26 datasets used in the evaluation is 1.19mm±0.27mm.

This, for example, transforms 1.77mm distance into 1.60 relative distance error. The division is done for each dataset independently and then the average is calculated. Just dividing the average distance by the average voxel size does not yield the correct result.

Power watersheds crashed on a few datasets (with $\geq$10 million voxels) due to hitting 2GB user memory limit of the available Win32-implementation. The soft boundary version also crashed on Neubert's case_* datasets [NFE+12], because it was developed with only small routine datasets in mind. No attention was paid to memory consumption, and at that time I had no access to the case_* datasets.

7.2 Data

The system was tested on 22 pathological datasets and 4 datasets from healthy volunteers for a total of 234 vertebrae. 13 are female and 13 male patients. The source code and 17 of these datasets are publicly released, aiming at the full reproducibility of research.

Data came from 7 different hospitals and 9 scanning stations (2 hospitals had 2 each). Reference segmentations were produced by manually tracing the vertebral body edges in the primary acquisition plane (23 sagittal, 2 axial, 1 coronal). 10 of the reference segmentations were done by neurosurgeons, 14 by an experienced user (myself) and 2 by an experienced user under neurosurgeon supervision (Aleš Neubert). The important property of these datasets is the high anisotropy of voxel size between 2.7× and 8.2×

Dataset	**MRI Seq.**	**Sex**	**Age**	**Path.**	**SB**	**Station**
AKa2	T_2 frFSE	F	21	+	EU	Sarajevo
AKa3 (*)	T_1 _FSE_	_F_	_21_	+	_EU_	_Sarajevo_
AKa4	TIRM	F	21	+	EU	Sarajevo
AKs3	T_2 frFSE	F	22	+	EU	Sarajevo
AKs5 (*)	T_2 _frFSE_	_F_	_22_	+	_EU_	_Sarajevo_
AKs6	T_1 FSE	F	22	+	EU	Sarajevo
AKs7 (*)	_TIRM_	_F_	_22_	+	_EU_	_Sarajevo_
AKs8 (*)	T_1 _FSE_	_F_	_22_	+	_EU_	_Sarajevo_
Ble	T_2 frFSE	F	64	+	NS	Marburg2
_C002 (*)_	T_2 _TSE_	_F_	_74_	_Sco_	_NS_	_Prague1_
case_2	T_2 SPACE	M	40	None	AN	Brisbane
case_10	T_2 SPACE	F	47	None	AN	Brisbane
DzZ_T1	T_1 TSE	M	27	None	EU	Siegen
DzZ_T2	T_2 TSE	M	27	None	EU	Siegen
_F02 (*)_	T_2 _SE_	_M_	_51_	_VF,SL_	_NS_	_Prague2_
F03	T_2 TSE	M	72	VF	NS	Prague1
F04	T_2 TSE	F	69	VF+	EU	Prague1
Geh	T_2 frFSE	M	25	+	NS	Schwabach
Hoe	T_2 frFSE	M	58	+	EU	Marburg2
Lan	T_2 TSE	M	79	+	EU	Bad Wildungen
LanII	T_1 TSE	M	79	+	EU	Bad Wildungen
LC	T_2 SE	M	47	+	NS	Marburg1
_S01 (*)_	T_2 _SE_	_M_	_65_	_SL_	_NS_	_Prague2_
_S02 (*)_	T_2 _SE_	_F_	_55_	_SL_	_NS_	_Prague2_
Sch	T_2 frFSE	M	42	SL+	NS	Marburg
_St1 (*)_	T_2 _SE_	_M_	_71_	_St_	_NS_	_Prague2_
Average			**45**			
StdDev			**22**			

Table 7.1: Medical information about the datasets used for the quantitative evaluation. Seq. – sequence, Path – pathologies (Sco–scoliosis, VF–vertebra fracture, SL–spondylolisthesis, +–other pathologies not diagnosable from vertebra segmentation). N$_{VB}$ – number of vertebral bodies in a dataset. SB – segmented by (NS–neurosurgeon, EU–experienced user).

Dataset	**Sx,y**	**Sz**	**xyR**	**zR**	**AF**	**N$_{sVB}$**	**MVox**	**A$_{iso}$**
AKa2	0.7	4	512	15	5.69	9	3.93	1.26
AKa3 (*)	*0.7*	*4*	*512*	*15*	*5.69*	*9*	*3.93*	*1.26*
AKa4	0.7	4	512	15	5.69	9	3.93	1.26
AKs3	0.7	4	512	25	5.69	10	6.55	1.26
AKs5 (*)	*0.7*	*4*	*512*	*15*	*5.69*	*9*	*3.93*	*1.26*
AKs6	0.7	4	512	15	5.69	9	3.93	1.26
AKs7 (*)	*0.7*	*4*	*512*	*15*	*5.69*	*9*	*3.93*	*1.26*
AKs8 (*)	*0.7*	*4*	*512*	*15*	*5.69*	*9*	*3.93*	*1.26*
Ble	0.63	4.4	512	14	7.04	10	3.67	1.2
C002 (*)	*1.12*	*3.3*	*448*	*31*	*2.96*	*13*	*6.22*	*1.6*
case_2	0.34	1.2	641	296	3.49	9	121.62	0.52
case_10	0.34	1	636	299	2.91	8	120.94	0.49
DzZ_T1	0.68	4.4	512	12	6.44	9	3.15	1.27
DzZ_T2	0.55	4.4	640	12	8.05	9	4.92	1.1
F02 (*)	*0.5*	*3.85*	*768*	*18*	*7.7*	*9*	*10.62*	*0.99*
F03	1.19	3.3	320	25	2.77	7	2.56	1.67
F04	1.12	3	448	23	2.69	17	4.62	1.55
Geh	0.63	4.4	512	10	7.04	7	2.62	1.2
Hoe	0.63	4.4	512	14	7.04	8	3.67	1.2
Lan	0.78	4.4	384	13	5.63	9	1.92	1.39
LanII	0.78	4.4	384	13	5.63	9	1.92	1.39
LC	0.73	4.4	384	14	6.03	7	2.06	1.33
S01 (*)	*0.47*	*3.85*	*640*	*16*	*8.19*	*7*	*6.55*	*0.95*
S02 (*)	*0.47*	*3.85*	*640*	*16*	*8.19*	*8*	*6.55*	*0.95*
Sch	0.63	4.4	512	16	7.04	7	4.19	1.2
St1 (*)	*0.5*	*3.85*	*704*	*20*	*7.7*	*8*	*9.91*	*0.99*
Average	**0.68**	**3.8**	**527**	**38**	**5.85**	**9.0**	**13.53**	**1.19**
StdDev	**0.21**	**0.88**	**104**	**76.5**	**1.69**	**2.1**	**31.79**	**0.27**

Table 7.2: Imaging information about the datasets used for the quantitative evaluation. S$_{x,y}$ – voxel spacing in $x-$ and $y-$directions (millimeters), S$_z$ – spacing along $z-$axis. xyR – resolution of image along $x-$ and $y-$axes, zR - $z-$resolution. AF – anisotropy factor $\frac{S_z}{S_{xy}}$. N$_{sVB}$ – number of segmented vertebral bodies in a dataset. MVox – millions of voxels. A$_{iso}$ – spacing of isotropic image with equivalent voxel volume.

(see Tab. 7.2 on the previous page and Fig. 7.6 on page 115). The manual segmentation time was 3-6 minutes per vertebra.

9 segmentations were used for training the Viola-Jones detector [VJ01] and later to optimize the parameters in section 6.5 on page 96. They were chosen from the datasets that could be publicly released. The goal was to have in both, the training set and the test set, various MRI sequences, anisotropy factors and at least one each of the three diagnosable pathologies. Other than that, the choice was arbitrary. From Tabs. 7.1 on page 102 and 7.2 on the previous page and Figs. 7.2 on page 108, 7.3 on page 108, 7.10 on page 117 and 7.9 on page 117 (AvgTrain and AvgTest) it can be seen that both of these sets have very similar characteristics and results. With hindsight it can be claimed that the training set is a bit more challenging.

To check segmentation expertise of the author, one dataset (F02) was manually segmented both by the author and a neurosurgeon. Dice coefficient is 91%, which is on par with segmentation correlations between neurosurgeons [EBK^{+}11a] (86-96% DSC).

7.3 Vertebra Detection Accuracy

The following abbreviations are used for different systems and their variants 7.3.

Abbr.	Initials	Chapt.	Variant
PW	Power Watersheds		Couprie et al.
bbFF	Binary Boundary – FreeForm	4	No subdivision
bbH	Binary Boundary – Heuristic	4	Heuristic subdivision hierarchy normalization
bbO	Binary Boundary – Optimal	4	Least squares optimal subdivision hierarchy normalization
mfO	Multi-Feature – Outline	5	Soft-boundary segmentation, outline initialization
mfSP	Multi-Feature – Single Point	5	Soft-boundary segmentation, single point initialization
AI	Automated Initialization	6	Initialized parallelized segmentation

Table 7.3: System abbreviations.

Dice Similarity Coefficient (DSC) is expressed in percentages, and is calculated with respect to the manual reference segmentation. Judging only by segmentation accuracy measures (e.g. DSCs from Tab. 7.9 on page 112), one would easily get the impression that the improvement was not great. But since the **bb** and **mf** systems derive detection from correctness of segmentation, they end up only accepting uniformly sized segmentations. Because of that, their detection accuracy suffers. Both the **bb** and **mf** systems had unsatisfactory detection rates because a single wrongly segmented vertebral body prevents further segmentations in that direction (up or down).

For the Viola-Jones detector (**AI** system) detection of upper thoracic vertebrae was deficient, because only one of the datasets used for detector training had vertebrae above T10 (dataset C002 also had vertebrae T6-T9).

Most false positives occurred for the S1 vertebra and the upper thoracic vertebrae, which is influenced by the low contrast for the upper thoracic region. The S1 vertebra simply has a significantly different geometry than lumbar and thoracic vertebrae, which is not taken into account by this system.

All variants had low number of false positives (detection as vertebra of something which is not a vertebra). Therefore the main improvement was the reduction of false negatives (failing to detect a vertebra which exists in the image). This can be most clearly seen in Tab. 7.4 on the following page.

Kelm et al. [KWZ$^+$13] established a state of the art regarding vertebra detection in magnetic resonance images. They report sensitivity of 98.6%, but do not report specificity. For MRI they used healthy individuals, but ascribe increase of errors in CT partly to pathologies.

The **AI** system presented in chapter 6 has 93.3% sensitivity, 98.6% specificity and 95.8% accuracy. These numbers can be calculated from Tabs. 7.4 on the next page and 7.5 on page 107. Number of manual corrections of detections depends on accuracy.

Kelm's detection results seem somewhat better, but they come at a cost of an order of magnitude longer processing time: for 6 million voxel images Kelm's method takes an average 11.5 seconds, while the detector presented in Sec. 6.2 on page 85 takes about 1 second. Even if dual-core/quad-core difference is accounted for, Kelm's method is still about 5 times slower. If we take into account that Kelm et al. used 38 datasets for training their detector, it can be concluded that the detector presented in Sec. 6.2 on page 85 is comparable to that of Kelm.

Dataset	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	56%	44%	44%	56%	56%	0%
AKa3 (*)	*89%*	*78%*	*89%*	*78%*	*67%*	*0%*
AKa4	78%	11%	11%	78%	56%	11%
AKs3	90%	70%	80%	60%	60%	20%
AKs5 (*)	*44%*	*11%*	*44%*	*56%*	*56%*	*0%*
AKs6	67%	22%	56%	89%	56%	11%
AKs7 (*)	*44%*	*44%*	*44%*	*44%*	*33%*	*0%*
AKs8 (*)	*67%*	*67%*	*78%*	*78%*	*56%*	*11%*
Ble	50%	10%	70%	50%	40%	10%
C002 (*)	*62%*	*54%*	*54%*	*46%*	*92%*	*15%*
case_2	89%	89%	89%			0%
case_10	78%	67%	78%			11%
DzZ_T1	11%	0%	0%	11%	0%	0%
DzZ_T2	33%	0%	0%	11%	0%	11%
F02 (*)	*67%*	*56%*	*67%*	*11%*	*89%*	*11%*
F03	38%	38%	38%	38%	38%	0%
F04	82%	82%	82%	53%	35%	12%
Geh	0%	0%	0%	0%	0%	0%
Hoe	38%	25%	25%	0%	0%	0%
Lan	11%	11%	33%	22%	11%	11%
LanII	89%	0%	11%	89%	22%	11%
LC	0%	0%	0%	0%	0%	0%
S01 (*)	*38%*	*13%*	*63%*	*13%*	*13%*	*13%*
S02 (*)	*40%*	*60%*	*60%*	*20%*	*20%*	*0%*
Sch	25%	0%	0%	0%	0%	13%
St1 (*)	*63%*	*50%*	*75%*	*25%*	*0%*	*0%*
AvgTest	**49.0%**	**27.6%**	**36.3%**	**37.1%**	**24.9%**	**7.1%**
AvgTrain (*)	*57.0%*	*48.0%*	*63.7%*	*41.1%*	*47.2%*	*5.6%*
AvgAll	51.8%	34.7%	45.8%	38.6%	33.2%	6.6%
StdDevAll	27.8%	30.0%	31.2%	30.1%	29.4%	6.5%
StdDevTest	**32.4%**	**31.5%**	**33.6%**	**33.1%**	**24.6%**	**6.5%**

Table 7.4: Vertebra detection, false negatives.

Dataset	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	0%	0%	0%	0%	0%	0%
AKa3 (*)	_0%_	_11%_	_11%_	_33%_	_0%_	_11%_
AKa4	0%	0%	0%	0%	0%	11%
AKs3	50%	0%	0%	0%	0%	0%
AKs5 (*)	_0%_	_0%_	_0%_	_0%_	_0%_	_0%_
AKs6	0%	11%	11%	0%	0%	11%
AKs7 (*)	_0%_	_0%_	_0%_	_11%_	_0%_	_0%_
AKs8 (*)	_0%_	_0%_	_0%_	_0%_	_0%_	_0%_
Ble	0%	0%	0%	0%	0%	0%
_C002 (*)_	_0%_	_0%_	_0%_	_0%_	_0%_	_0%_
case_2	0%	11%	0%			0%
case_10	0%	0%	11%			0%
DzZ_T1	0%	0%	0%	0%	0%	0%
DzZ_T2	0%	0%	0%	0%	0%	0%
_F02 (*)_	_0%_	_0%_	_0%_	_0%_	_0%_	_0%_
F03	0%	0%	0%	0%	0%	0%
F04	0%	0%	0%	0%	0%	0%
Geh	0%	0%	0%	14%	0%	0%
Hoe	0%	0%	0%	0%	0%	0%
Lan	0%	0%	0%	0%	0%	0%
LanII	0%	0%	0%	0%	11%	0%
LC	0%	0%	0%	0%	0%	0%
_S01 (*)_	_0%_	_0%_	_0%_	_0%_	_0%_	_0%_
_S02 (*)_	_0%_	_10%_	_0%_	_10%_	_0%_	_10%_
Sch	0%	0%	0%	0%	0%	0%
_St1 (*)_	_0%_	_0%_	_0%_	_0%_	_13%_	_0%_
AvgTest	**2.9%**	**1.3%**	**1.3%**	**1.0%**	**0.7%**	**1.3%**
AvgTrain (*)	_0.0%_	_2.3%_	_1.2%_	_6.0%_	_1.4%_	_2.3%_
AvgAll	1.9%	1.7%	1.3%	2.9%	1.0%	1.7%
StdDevAll	9.8%	4.0%	3.6%	7.6%	3.3%	4.0%
StdDevTest	**12.1%**	**3.7%**	**3.7%**	**3.7%**	**2.9%**	**3.7%**

Table 7.5: Vertebra detection, false positives.

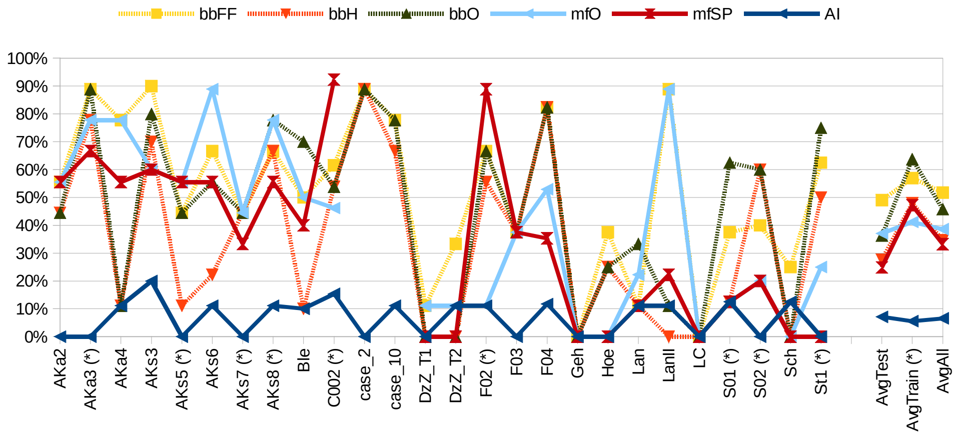


Figure 7.2: Vertebra detection, false negatives. Lower is better.

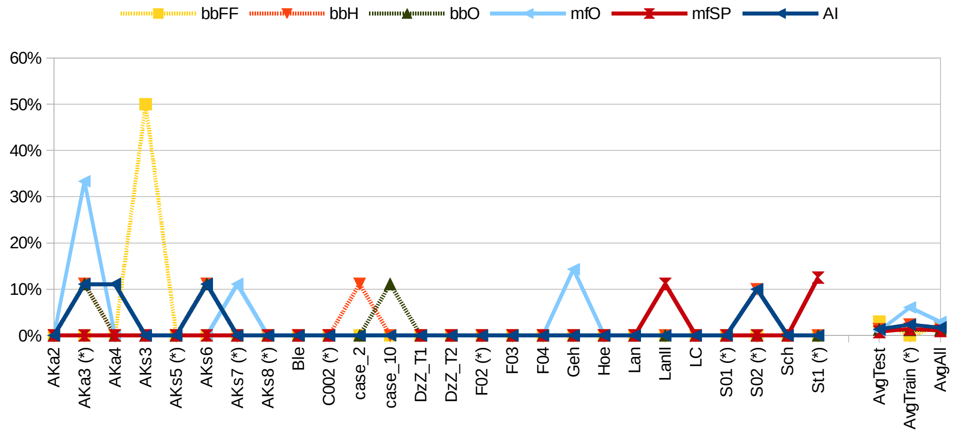


Figure 7.3: Vertebra detection, false positives. Lower is better.

7.4 Segmentation Accuracy

The majority of discrepancies between manual and automatic segmentations result from lateral slices and upper thoracic vertebrae (see Fig. 7.6 on page 115). Lateral slices make it harder to algorithmically discern a boundary due to significant partial volume effects. Power watersheds mostly fail on lateral edges of vertebral bodies, too (Figs. 7.4 on page 113 and 7.11 on page 118).

The averages presented in this chapter only take into account the segmented vertebral bodies, ignoring the missed vertebrae (all systems except PW and AI) and datasets which caused a crash (PW and MF*). Therefore the results seem somewhat better than they really are (for all systems

Dataset	**PW**	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	36.28	21.86	22.21	23.34	10.27	12.2	10.82
AKa3 (*)	*45.54*	*16.17*	*40.58*	*39.28*	*51.91*	*12.03*	*10.79*
AKa4	26.80	23	12.7	12.4	23.05	28.1	9.29
AKs3	60.10	108.17	13.28	11.37	11.04	13.04	17.67
AKs5 (*)	*32.73*	*20.84*	*17.59*	*19.29*	*6.74*	*8.07*	*10.8*
AKs6	57.24	9.71	20.7	31.85	10.94	13.18	10.58
AKs7 (*)	*22.44*	*16*	*10.12*	*12.13*	*42.94*	*24.86*	*11.28*
AKs8 (*)	*23.69*	*16.34*	*18.71*	*18.54*	*23.81*	*11.51*	*9.11*
Ble	22.31	16.46	16.82	17.01	9.01	14.94	14.43
C002 (*)	*33.88*	*18.86*	*9.38*	*9.48*	*12.83*	*25.12*	*11.81*
case_2		32.27	18.95	19.28			16.59
case_10		37.16	16.72	44.76			11.29
DzZ_T1	45.46	17.55	10.43	10.42	10.14	11.35	12.01
DzZ_T2	63.00	15.71	11.37	13.52	12.32	11.81	13.59
F02 (*)		*22.88*	*25.12*	*26.04*	*15.12*	*17.78*	*15.65*
F03	42.23	16.22	10.98	11.25	13.85	15.78	14.14
F04	49.56	13.69	12.82	13.9	10.14	8.71	11.4
Geh	86.09	17.43	17.47	19.77	11.68	15.23	7.54
Hoe	65.47	18.42	16.24	18.37	11.79	9.71	8.91
Lan	43.26	16.1	15	12.12	14.4	9.83	10.47
LanII	29.88	10.35	14	12.43	17.41	14.28	11.06
LC	46.92	13.75	12.06	12.2	13.85	12.94	10.51
S01 (*)	*37.28*	*22.74*	*21.75*	*21.08*	*12.79*	*14.11*	*19.56*
S02 (*)	*24.33*	*15.26*	*14.82*	*16.05*	*14.9*	*9.91*	*9.73*
Sch	67.67	10.68	14.66	16.74	11.99	12.94	11.76
St1 (*)		*22.33*	*24.91*	*13.02*	*26.2*	*17.58*	*17.22*
AvgTest	**49.48**	**23.44**	**15.08**	**17.69**	**12.79**	**13.60**	**11.89**
AvgTrain (*)	*31.41*	*19.05*	*20.33*	*19.43*	*23.03*	*15.66*	*12.88*
AvgAll	43.73	21.92	16.90	18.29	16.63	14.38	12.23
StdDevAll	17.18	18.62	6.54	8.76	10.65	5.17	3.01
StdDevTest	**17.34**	**23.01**	**3.42**	**8.86**	**3.54**	**4.51**	**2.65**

Table 7.6: Average Hausdorff (symmetric maximum) distance between segmented and reference vertebral body meshes, in millimeters.

Dataset	**PW**	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	8.10	3.55	2.98	4.08	1.04	1.17	1.72
AKa3 (*)	*11.82*	*2.61*	*22.25*	*22.49*	*22*	*1.51*	*1.63*
AKa4	5.99	5.44	1.35	1.63	6.64	9.47	1.56
AKs3	15.25	76.41	2.67	2.72	1.51	1.64	2.04
AKs5 (*)	*8.62*	*2.87*	*2.51*	*3*	*0.99*	*1.17*	*1.37*
AKs6	15.40	1.98	4.51	8.95	1.12	1.73	1.85
AKs7 (*)	*5.74*	*2.63*	*1.43*	*1.7*	*19.55*	*7.38*	*1.96*
AKs8 (*)	*4.49*	*2.78*	*2.73*	*2.8*	*3.87*	*1.51*	*1.46*
Ble	3.74	2.31	2.61	2.88	1.1	1.73	2.07
C002 (*)	*7.62*	*3.84*	*1.43*	*1.67*	*1.93*	*9.32*	*1.84*
case_2		7.49	2.11	2.3			2.29
case_10		13.09	1.94	14.79			1.09
DzZ_T1	11.25	3.17	1.14	1.42	1.41	1.54	1.76
DzZ_T2	14.61	2.89	1.3	1.71	1.61	1.75	2.04
F02 (*)		*2.44*	*2.99*	*3*	*1.77*	*2.9*	*2*
F03	11.93	4.11	1.53	1.91	2	2.1	2.18
F04	14.50	3.05	3.11	3.71	1.61	1.51	2.18
Geh	24.00	2.03	2.04	2.55	1.5	2.2	1.08
Hoe	14.58	2.34	2.18	2.55	1.37	1.32	1.23
Lan	9.05	2.63	1.72	1.8	1.7	1.47	1.74
LanII	7.33	1.65	2.13	1.88	3.19	3.92	1.98
LC	9.97	2.4	1.45	1.66	1.96	2.02	1.76
S01 (*)	*6.78*	*2.31*	*2.7*	*3.72*	*1.35*	*1.45*	*2.12*
S02 (*)	*5.79*	*1.97*	*1.73*	*2.15*	*3.54*	*1.2*	*1.38*
Sch	18.67	1.99	1.5	1.95	1.39	1.52	1.41
St1 (*)		*2.57*	*2.93*	*1.95*	*2.37*	*2.01*	*1.95*
AvgTest	**12.29**	**8.03**	**2.13**	**3.44**	**1.94**	**2.34**	**1.77**
AvgTrain (*)	*7.27*	*2.67*	*4.52*	*4.72*	*6.37*	*3.16*	*1.75*
AvgAll	10.69	6.17	2.96	3.88	3.61	2.65	1.76
StdDevAll	5.07	14.51	4.01	4.71	5.44	2.44	0.34
StdDevTest	**5.24**	**17.84**	**0.85**	**3.41**	**1.40**	**2.07**	**0.38**

Table 7.7: Mean distance of points in the automatic segmentation mesh from the reference mesh, in millimeters.

Dataset	**PW**	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	6.45	2.82	2.37	3.25	0.83	0.93	1.37
AKa3 (*)	*9.41*	*2.08*	*17.73*	*17.92*	*17.53*	*1.2*	*1.3*
AKa4	4.77	4.33	1.08	1.3	5.29	7.55	1.24
AKs3	12.15	60.88	2.13	2.17	1.2	1.31	1.62
AKs5 (*)	*6.87*	*2.29*	*2*	*2.39*	*0.79*	*0.93*	*1.09*
AKs6	12.27	1.58	3.6	7.13	0.89	1.38	1.47
AKs7 (*)	*4.57*	*2.1*	*1.14*	*1.35*	*15.58*	*5.88*	*1.56*
AKs8 (*)	*3.57*	*2.21*	*2.17*	*2.23*	*3.08*	*1.2*	*1.16*
Ble	3.12	1.93	2.18	2.41	0.92	1.45	1.73
C002 (*)	*4.76*	*2.4*	*0.9*	*1.04*	*1.21*	*5.82*	*1.15*
case_2		14.36	4.05	4.42			4.4
case_10		26.68	3.96	30.13			2.23
DzZ_T1	8.85	2.49	0.89	1.11	1.11	1.21	1.39
DzZ_T2	13.33	2.63	1.19	1.56	1.47	1.59	1.87
F02 (*)		*2.47*	*3.03*	*3.04*	*1.79*	*2.94*	*2.03*
F03	7.14	2.46	0.91	1.15	1.2	1.25	1.31
F04	9.35	1.96	2.01	2.39	1.04	0.98	1.4
Geh	20.04	1.7	1.71	2.13	1.25	1.84	0.9
Hoe	12.18	1.95	1.82	2.13	1.15	1.1	1.03
Lan	6.51	1.89	1.24	1.3	1.23	1.06	1.26
LanII	5.27	1.19	1.53	1.35	2.29	2.82	1.42
LC	7.50	1.8	1.09	1.25	1.48	1.52	1.33
S01 (*)	*7.16*	*2.44*	*2.85*	*3.93*	*1.43*	*1.53*	*2.24*
S02 (*)	*6.11*	*2.08*	*1.82*	*2.27*	*3.73*	*1.27*	*1.46*
Sch	15.58	1.66	1.25	1.63	1.16	1.27	1.18
St1 (*)		*2.6*	*2.97*	*1.98*	*2.4*	*2.04*	*1.97*
AvgTest	**9.63**	**7.78**	**1.94**	**3.93**	**1.50**	**1.82**	**1.60**
AvgTrain (*)	*6.07*	*2.30*	*3.85*	*4.02*	*5.28*	*2.54*	*1.55*
AvgAll	8.50	5.88	2.60	3.96	2.92	2.09	1.58
StdDevAll	4.24	12.40	3.22	6.29	4.34	1.77	0.67
StdDevTest	**4.58**	**15.14**	**1.03**	**6.92**	**1.10**	**1.65**	**0.79**

Table 7.8: Relative surface distance errors (mean divided by A_{iso}).

Dataset	**PW**	**bbFF**	**bbH**	**bbO**	**mfO**	**mfSP**	**AI**
AKa2	74.0%	56.3%	61.3%	44.8%	87.3%	86.5%	82.3%
AKa3 (*)	*63.6%*	*79.3%*	*40.8%*	*40.4%*	*15.0%*	*81.1%*	*81.6%*
AKa4	73.1%	75.9%	84.1%	82.0%	41.6%	44.0%	83.2%
AKs3	64.3%	0.0%	73.6%	72.5%	82.8%	81.9%	73.8%
AKs5 (*)	*70.7%*	*65.1%*	*70.6%*	*62.9%*	*87.2%*	*85.5%*	*84.2%*
AKs6	60.9%	81.7%	51.7%	45.1%	86.1%	79.7%	80.6%
AKs7 (*)	*74.5%*	*82.9%*	*85.9%*	*82.8%*	*31.8%*	*48.9%*	*78.0%*
AKs8 (*)	*76.2%*	*65.4%*	*63.9%*	*57.0%*	*61.2%*	*83.4%*	*83.7%*
Ble	78.9%	72.9%	65.2%	65.5%	86.1%	74.4%	72.5%
C002 (*)	*64.9%*	*70.2%*	*79.6%*	*77.0%*	*73.4%*	*0.8%*	*74.5%*
case_2		75.9%	76.1%	73.2%			69.2%
case_10		70.5%	73.0%	38.6%			80.6%
DzZ_T1	63.1%	86.9%	87.7%	89.4%	87.3%	86.9%	84.3%
DzZ_T2	65.4%	84.9%	85.3%	82.2%	82.6%	81.5%	78.9%
F02 (*)		*65.4%*	*60.7%*	*60.1%*	*77.7%*	*62.3%*	*74.2%*
F03	61.2%	76.5%	84.9%	82.1%	79.4%	79.0%	76.1%
F04	58.2%	62.2%	61.5%	54.7%	77.6%	79.9%	72.6%
Geh	56.8%	77.5%	79.3%	73.7%	82.2%	74.5%	88.4%
Hoe	56.0%	70.2%	74.8%	66.9%	79.5%	81.0%	83.1%
Lan	60.6%	72.7%	76.9%	80.1%	78.6%	82.1%	80.0%
LanII	64.0%	84.7%	76.4%	79.8%	68.6%	69.8%	77.9%
LC	58.8%	81.0%	84.2%	82.3%	79.2%	78.9%	81.7%
S01 (*)	*70.4%*	*71.3%*	*73.2%*	*62.4%*	*82.6%*	*82.3%*	*76.7%*
S02 (*)	*71.3%*	*74.3%*	*77.2%*	*70.9%*	*69.9%*	*82.3%*	*79.7%*
Sch	56.4%	82.4%	80.1%	73.4%	81.4%	81.2%	82.4%
St1 (*)		*75.8%*	*61.0%*	*80.3%*	*67.9%*	*79.7%*	*79.9%*
AvgTest	**63.5%**	**71.3%**	**75.1%**	**69.8%**	**78.7%**	**77.4%**	**79.3%**
AvgTrain (*)	*70.2%*	*72.2%*	*68.1%*	*66.0%*	*63.0%*	*67.4%*	*79.2%*
AvgAll	65.6%	71.6%	72.7%	68.4%	72.8%	73.7%	79.2%
StdDevAll	6.9%	16.5%	11.5%	14.5%	18.4%	18.9%	4.5%
StdDevTest	**6.9%**	**20.1%**	**10.0%**	**15.2%**	**11.3%**	**10.2%**	**5.0%**

Table 7.9: Dice similarity coefficient between automatic and reference segmentation masks.

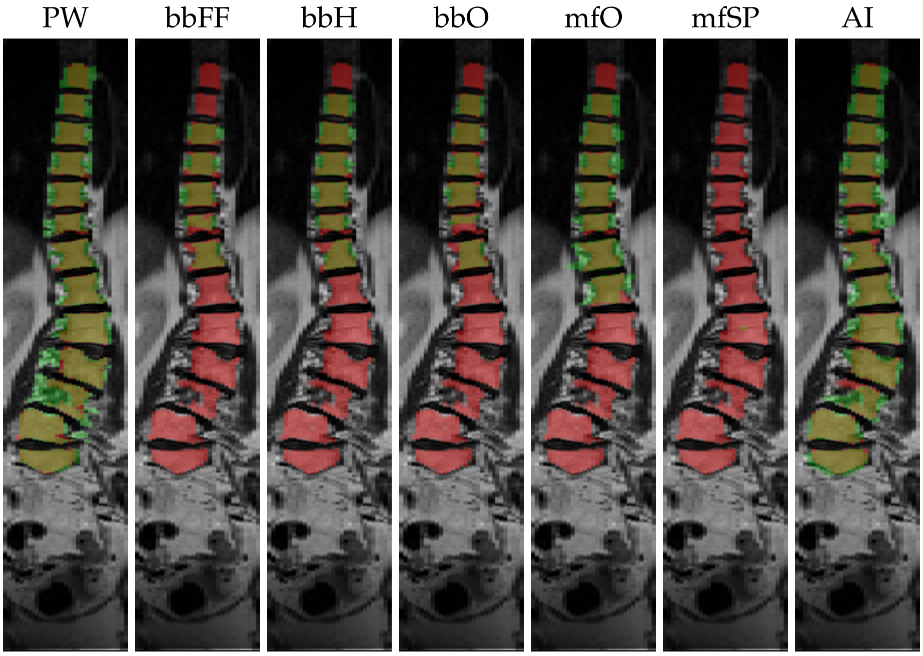


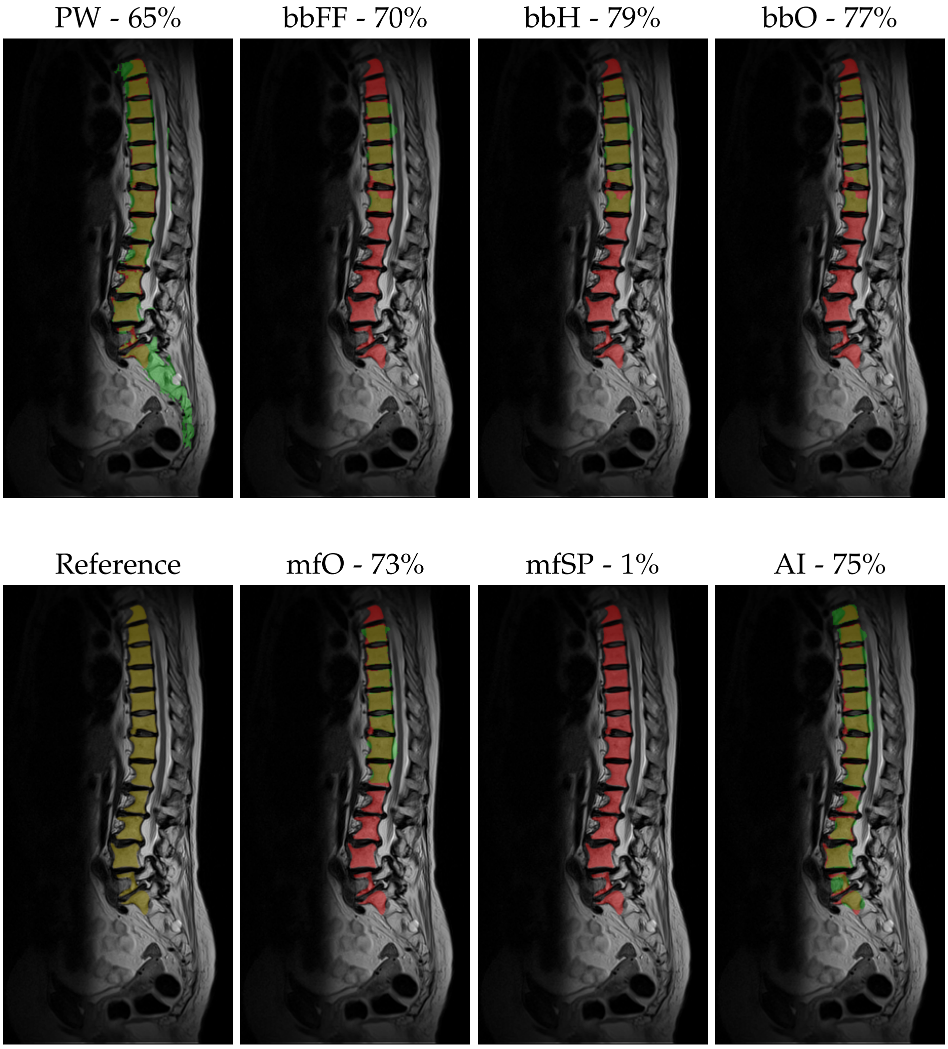
Figure 7.4: Overlay of segmentation results (green) and manual reference (red). Intersection is yellow. Dataset C002, coronal slice 242. It is fairly obvious that the majority of segmentation errors comes from lateral (left and right) edges of the vertebral bodies. The reason are thick slices (coarse voxel spacing along the inter-slice axis).

except AI). Since vertebra detection accuracy was discussed in the previous section, this section focuses only on segmentation precision decoupled from detection accuracy.

In spite of this, a progress (**bb**→**mf**→**AI**) can be easily seen in the tables and charts in this section. For comparison to methods of other authors, the best (AI) system will be taken into account.

For the AI system the mean distance of segmentation from the reference surface is 1.77 mm ± 0.36 mm. When converted into relative distance it is 1.60 ± 0.79. This relative distance should be used for comparing to results obtained on isotropic 1mm^3 datasets.

The systems presented here are vastly superior to power watersheds for the purpose of vertebral body segmentation in MRI. The execution time is similar, but **AI** has higher DSC (79% vs 67%), and significantly lower Hausdorff distance (12mm vs 49mm) and relative distance error (1.59 vs. 9.63).



PW - 65%

bbFF - 70%

bbH - 79%

bbO - 77%

Reference

mfO - 73%

mfSP - 1%

AI - 75%

Figure 7.5: Overlay of segmentation results (green) and manual reference (red). Intersection is yellow. Dataset C002, mid-sagittal slice (slice 14). For each system only the segmented vertebrae were used to calculate the DSC. It can be seen that single point initialization variant of the soft boundary system has failed.

Power watersheds have lower DSC than all the systems presented here in spite of fairly rich initialization (Fig. 7.1 on page 100). Power watersheds frequently produce some protruding spikes (Fig. 7.11 on page 118). These spikes are thin, so they do not hurt DSC measure so much, but surface distance errors are significantly higher (12.29mm, 9.63 relative).

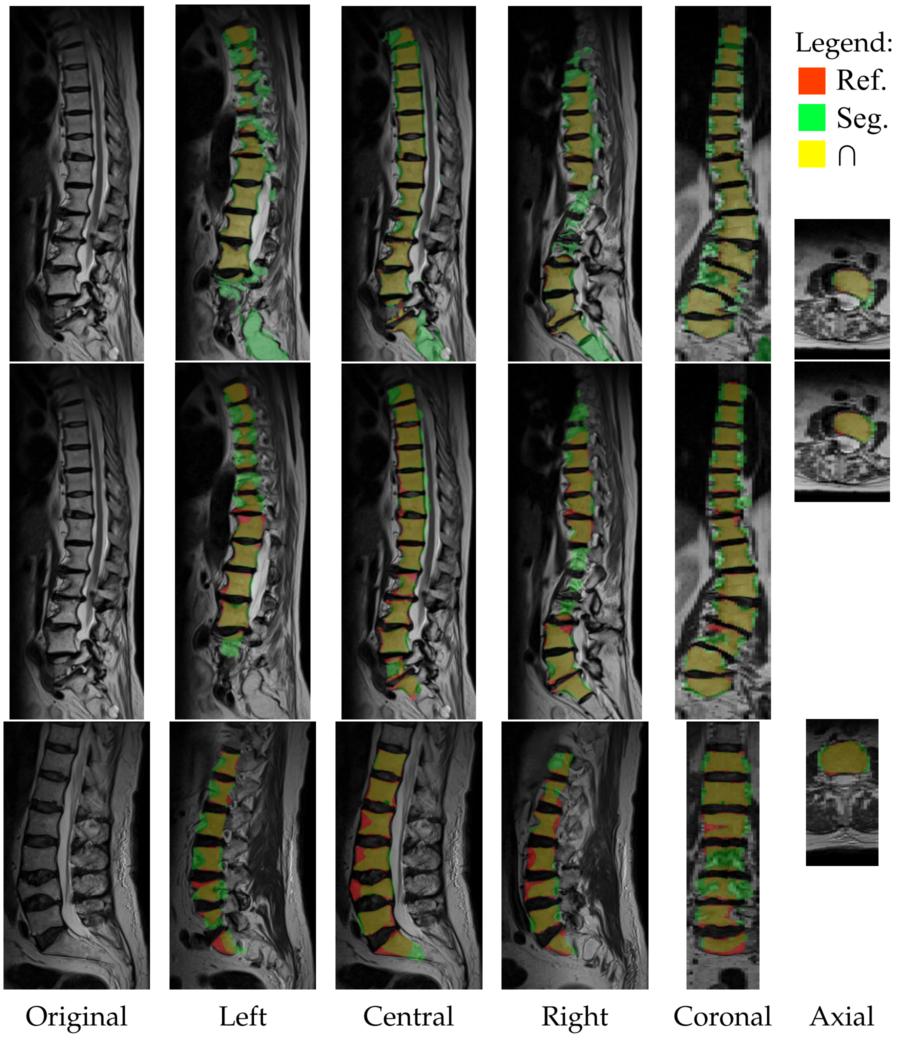


Figure 7.6: Overlay with reference segmentation showing several representative slices. Power watersheds (**PW**, top row) and initialized parallelized segmentation (**AI**, bottom two rows). Images were cropped to save space (unimportant parts were cut off). See Tab. 7.2 on page 103 for dataset properties. Top and middle: dataset C002. Bottom: dataset F03. Dice coefficients are 65%, 75% and 76%.

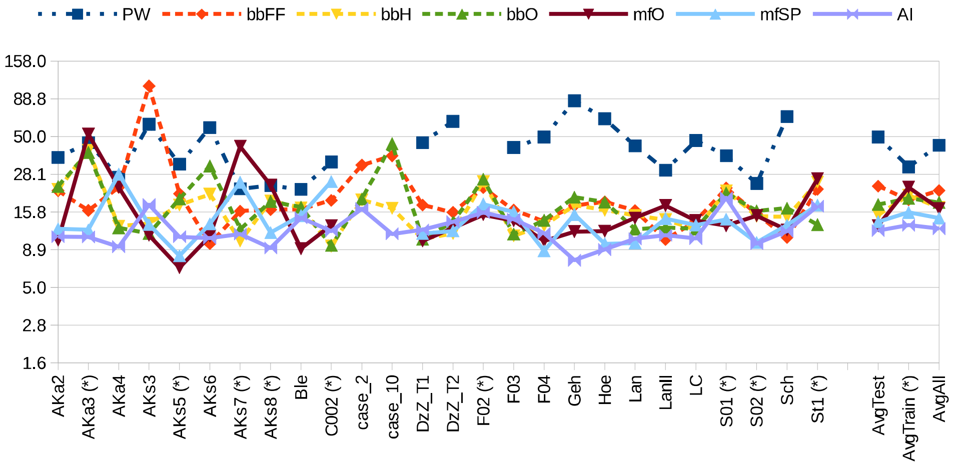


Figure 7.7: Average Hausdorff distance between segmented and reference vertebral body meshes. Scale is logarithmic, in millimeters.

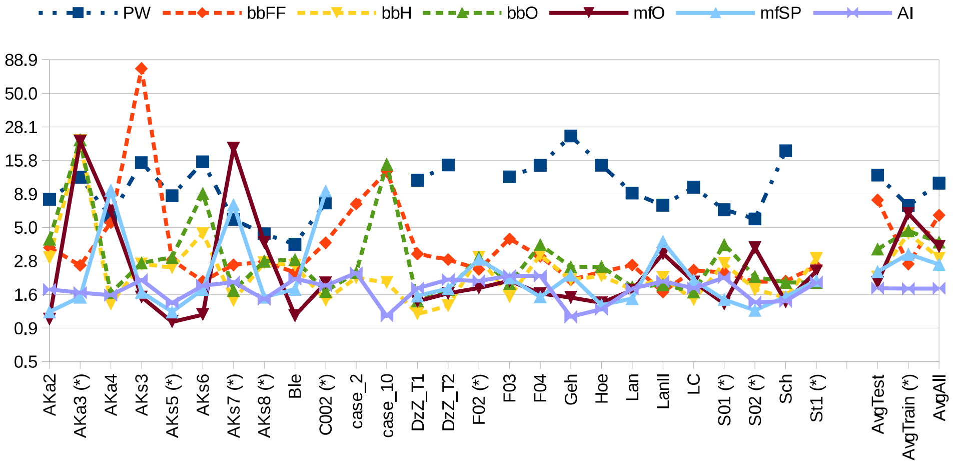


Figure 7.8: Mean distance of points in the automatic segmentation mesh from the reference mesh. Scale is logarithmic, in millimeters.

The average DSC of 79.3% is close to Davatzikos et al. with 81%, but they used healthy individuals. Healthy volunteer datasets are easier to segment and achieve better DSC. The datasets with more severe pathologies (e.g. crushed vertebra) have lower DSC than less noticeable pathologies (e.g. stenosis). Pathologies obviously and naturally reduce the detection rate. Using the relative distances error of 1.60±0.79, my system can be compared to Štern et al., who report an error of 1.85mm±0.47mm on a 1 mm^3 grid. When taking into account only the mid-sagittal slice, my system achieves ≈86% DSC, which is similar to Ayed et al. at 85%. Hoad

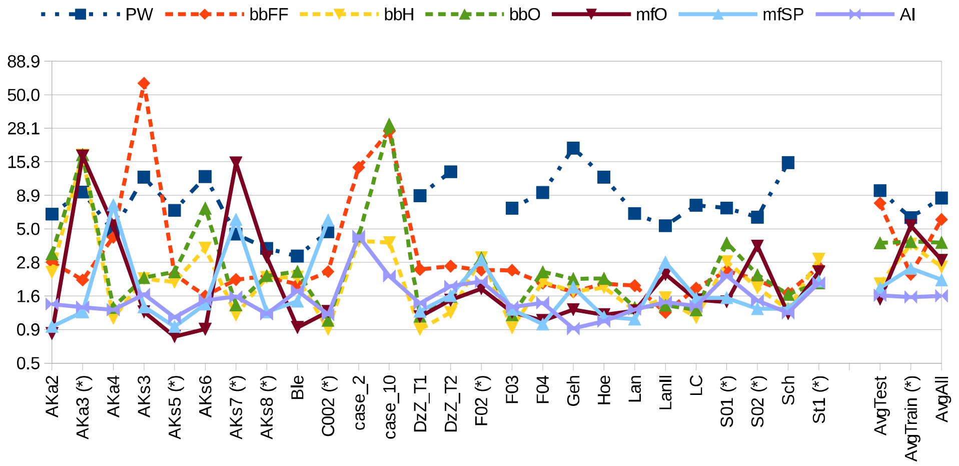


Figure 7.9: Relative surface distance errors. Scale is logarithmic.

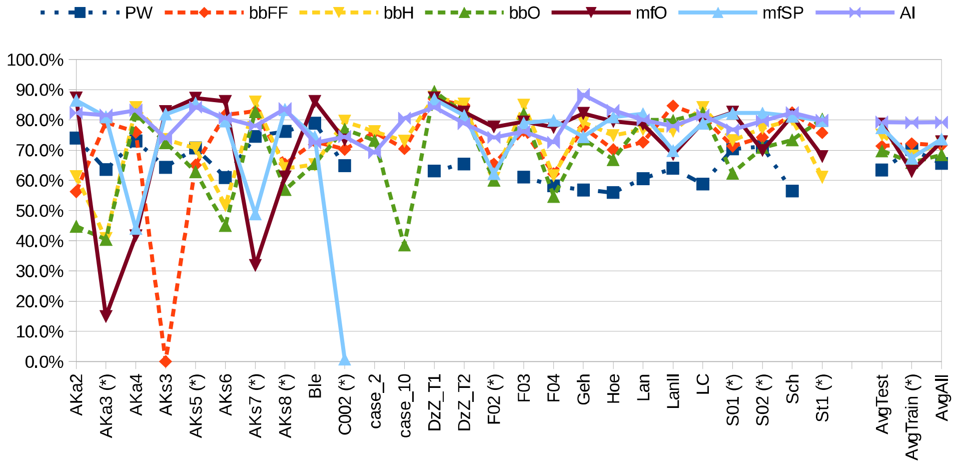


Figure 7.10: Dice similarity coefficient between automatic and reference segmentation masks.

and Martel's surgery-oriented method with thorough initialization and manual correction of segmentation at 1.25mm±0.28mm mean distance error remains the most precise.

Neubert kindly provided two datasets with manual segmentations (case_2 and case_10), so I could directly compare the system from chapter 6 on page 83 to theirs [NFS+11, NFE+12].

In order to have a fair comparison to Neubert et al. [NFE+12] their distance results should be converted into relative distances. A_{iso} of 0.5mm transforms distance error of 0.67mm into a relative distance error of 1.34.

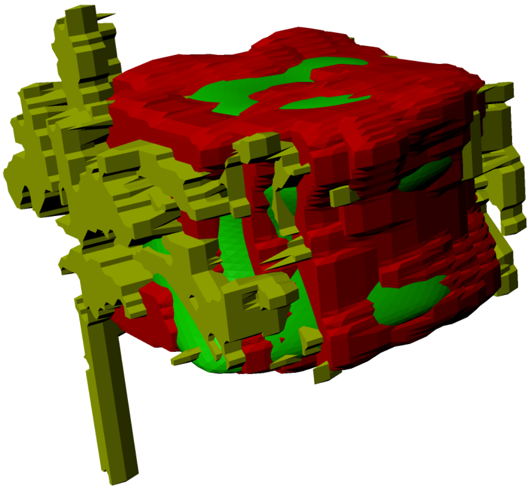


Figure 7.11: Surface overlay of soft boundary segmentation (**mfSP**, green) and power watersheds (**PW**, yellow). Red is the reference segmentation. Reference and PW surfaces were derived from binary masks using marching cubes. L2 vertebral body from DzZ_T1 dataset. DSCs: PW–85%, mfSP–88%.

This is better than **AI** relative distance error (1.6). Neubert et al. also have higher DSC (91% vs my 78%). However this high quality comes at a cost of average execution time per dataset of 5.5 hours.

When they ran their method on datasets of higher anisotropy, their DSC was two percentage points lower: 83% vs 85% (see [NFS+11], dataset group I vs II). If less time is allotted to the iterative optimization, DSC is lower: tenfold reduction in execution time (5.5h→35min) lowers DSC from 91% [NFE+12] to 85% [NFS+11].

As my system was tailored for lower resolution routine datasets, it had low precision on Neubert's high-resolution datasets, case_2: 2.29mm and case_10: 1.09mm. However, the method worked without crashing, albeit with a large memory consumption. Unfortunately, Neubert did not evaluate his method on the 17 publicly available datasets for full cross-validation.

If case_2 and case_10 datasets are not taken into account, relative distance error decreases considerably for **AI** system, and both relative and absolute distance errors decrease for **bb** system. **PW** and **mf** systems crashed on these datasets, so there is no difference for them (Tab. 7.10 on the next page).

Restricting the test set to the routine datasets, for which the systems presented in this thesis were designed, brings the relative distance error of

Measure	**system**	**17DS**	**15DS**	**Relative improvement**
Hausdorff	PW	49.48	49.48	0.00%
	bbFF	23.44	21.94	6.41%
	bbH	15.08	14.71	2.43%
	bbO	17.69	15.78	10.80%
	mfO	12.79	12.79	0.00%
	mfSP	13.60	13.60	0.00%
	AI	11.89	11.61	2.31%
MeanSD	PW	12.29	12.29	0.00%
	bbFF	8.03	7.73	3.76%
	bbH	2.13	2.15	−0.67%
	bbO	3.44	2.76	19.78%
	mfO	1.94	1.94	0.00%
	mfSP	2.34	2.34	0.00%
	AI	1.77	1.77	−0.53%
RelativeSD	PW	9.63	9.63	0.00%
	bbFF	7.78	6.09	21.82%
	bbH	1.94	1.67	14.15%
	bbO	3.93	2.15	45.29%
	mfO	1.50	1.50	0.00%
	mfSP	1.82	1.82	0.00%
	AI	**1.60**	**1.37**	**14.35%**
DSC	PW	63.5%	63.5%	0.00%
	bbFF	71.3%	71.0%	−0.35%
	bbH	75.1%	75.1%	0.09%
	bbO	69.8%	71.6%	2.58%
	mfO	78.7%	78.7%	0.00%
	mfSP	77.4%	77.4%	0.00%
	AI	79.3%	79.9%	0.73%

Table 7.10: Averages of results on entire test set (17 datasets) and without taking case_2 and case_10 datasets into account (15 datasets). The most important improvement is emphasized.

AI system down to 1.37±0.25. This is very close to Neubert et al. at 1.34 and comparable to Hoad and Martel at 1.25±0.28.

It is of interest to note that this does not improve the Dice Coefficients much. The reason is that the voxels in Neubert's datasets have about 15 times smaller volume than average. Due to that about 15 times more voxels make up an object (vertebral body in this case), and a relative difference of a single voxel has about 15 times smaller influence on the DSC.

7.5 Diagnosis Accuracy

Diagnosis is highly dependent on correctness of segmentation. It does not make much sense to try to establish a diagnosis based only on the segmentations of half of the vertebrae present in the image. In addition, correct vertebra labels are needed to have a somewhat automated evaluation. Only the latest system (AI) provides this, so diagnosis accuracy is examined only for this case.

The used datasets had no examples of high-grade vertebral fracture. Two instances of high-grade spondylolisthesis were classified correctly, as was the only high-grade scoliosis. There were 6 false alarms for severe crushed vertebra and 9 for severe spondylolisthesis, or 2.6% and 3.9% respectively. There were no false alarms for scoliosis.

Overall, the system errs on the side of caution, falsely drawing attention to non-diseased vertebrae (5.5%) and never missing a diseased one, leading to a conservative and robust behavior.

The diagnostic measurement errors for crushing percent, spondylolisthesis percent and Cobb angle are 4.3 percentage points, 4.6 percentage points and 3°, respectively. Relative errors (55%, 52% and 31%) are quite large because the ground truth measurements are fairly low (2.5%, 4% and 9.2°).

The averages for automatic results were 4.9% crushing, 7.9% spondylolisthesis and 9.8° Cobb angle. The difference to ground truth should mostly be ascribed to imperfect segmentation.

This proves once again that high reliability of diagnosis is not easy to achieve. The diagnosis results established using the systems presented here are a valuable help to the medical professionals, but cannot replace their diagnosis.

	Time	**Crushed**		**Spondyl.**		**Cobb**		**Diag.**		**N$_{VB}$**
Dataset	**[sec]**	**PP**	**Rel.**	**PP**	**Rel.**	**AD**	**Rel.**	**FP**	**FN**	
AKa2	32	2.4	45%	6.5	68%	3.3	21%	0	0	9
AKa3 (*)	_17_	_4.5_	_58%_	_5.0_	_48%_	_4.3_	_28%_	_1_	_0_	_9_
AKa4	29	5.7	60%	1.6	29%	1.9	14%	1	0	9
AKs3	29	7.0	55%	11.0	67%	5.5	44%	4	0	10
AKs5 (*)	_36_	_2.0_	_50%_	_4.0_	_38%_	_2.0_	_28%_	_1_	_0_	_9_
AKs6	38	5.2	52%	5.5	56%	2.8	27%	1	0	9
AKs7 (*)	_27_	_3.2_	_47%_	_2.6_	_34%_	_1.9_	_26%_	_2_	_0_	_9_
AKs8 (*)	_20_	_1.7_	_22%_	_3.4_	_47%_	_1.8_	_14%_	_2_	_0_	_9_
Ble	50	5.2	56%	8.4	58%	7.6	34%	3	0	10
_C002 (*)_	_26_	_2.5_	_48%_	_3.7_	_32%_	_1.8_	_6%_	_1_	_0_	_13_
case_2	396	8.1	52%	5.0	72%	3.4	59%		0	9
case_10	317	6.3	74%	2.1	54%	0.4	5%		0	8
DzZ_T1	18	2.9	59%	2.1	56%	0.4	11%	2	0	9
DzZ_T2	62	4.1	68%	3.6	64%	2.2	61%	0	0	9
_F02 (*)_	_114_	_8.4_	_45%_	_4.9_	_50%_	_7.4_	_36%_	_2_	_1_	_9_
F03	14	2.6	28%	7.6	49%	7.6	58%	0	1	7
F04	23	10.0	80%	9.6	53%	0.5	6%	3	2	17
Geh	27	1.4	66%	0.5	26%	0.6	15%	0	0	7
Hoe	33	1.8	60%	1.8	45%	4.3	40%	0	0	8
Lan	12	2.6	43%	4.0	54%	3.4	40%	2	0	9
LanII	13	2.1	39%	4.1	61%	5.1	59%	1	0	9
LC	24	3.3	46%	3.1	38%	2.7	21%	1	0	7
_S01 (*)_	_78_	_3.7_	_50%_	_2.4_	_31%_	_1.2_	_19%_	_0_	_0_	_7_
_S02 (*)_	_59_	_3.8_	_51%_	_4.6_	_66%_	_1.3_	_50%_	_0_	_1_	_8_
Sch	37	2.7	50%	1.9	40%	0.8	10%	0	0	7
_St1 (*)_	_91_	_4.4_	_49%_	_2.9_	_60%_	_4.5_	_34%_	_1_	_0_	_8_
AvgTest	**80**	**4.4**	**53%**	**4.2**	**51%**	**3.2**	**30%**	**1.3**	**0.1**	**8.9**
AvgTrain (*)	_30_	_3.6_	_50%_	_4.5_	_47%_	_2.7_	_29%_	_1.0_	_0.4_	_9.2_
AvgAll	62	4.1	52%	4.3	50%	3.0	29%	1.2	0.2	9.0
StdDevAll	91	2.3	12%	2.6	13%	2.2	18%	1.1	0.5	2.1
StdDevTest	**109**	**2.1**	**10%**	**2.6**	**15%**	**2.2**	**18%**	**1.2**	**0.2**	**1.4**

Table 7.11: Quantitative evaluation of diagnosis. PP – difference in percentage points. AD – angle difference in degrees. FP – false positives. FN – false negatives. N$_{VB}$ – number of vertebral bodies in dataset. Execution time is for the entire system (except vertebra detection), not just the diagnosis.

Chapter 8

Conclusion

A vertebral body segmentation system was developed with a variety of routine MR images in mind. The system is therefore general and applicable also to specialized images, such as those of 3D SPACE sequence. Significant usage of working memory was a technical side-effect. More important was the failure of this system to fully utilize the high resolution offered by 3D SPACE sequence. This conclusion just once again emphasizes that there is no "magic bullet" algorithm which works with all kinds of data and exploits each dataset type to the maximum.

Increasing algorithmic complexity improves the results but also increases computational burden. In chapter 4 it is 3-4 seconds, in chapter 5 about 30 seconds and in chapter 6 around 70 seconds.

The diagnosis established at the end relies on all steps done before, so every imperfection makes diagnosis harder. Especially hard was diagnosing crushed vertebral bodies. A sufficient diagnosis for that would require a more intricate shape analysis which in turn requires higher segmentation precision. Overall, the established diagnosis can only aid the physician, not replace his judgment.

The levels of the butterfly scheme have large steps so it is sometimes impossible to apply just the right number of control vertices. Perhaps that would be somewhat alleviated if the $\sqrt{3}$-scheme [Kob00] was used.

Open science

The program code is made publicly available under an open source MIT license, including 17 anonymized datasets with corresponding segmentations: `http://www.cg.informatik.uni-siegen.de/en/spine-segmentation-and-analysis` A written consent was collected for every MRI study. Please cite [ZVE$^+$14] or this thesis if you use any of these in your work.

The project is configured using Cross-platform Make (CMake), and all the libraries used are multi-platform too. The datasets are in single-file DICOM format, readable by Insight Toolkit [ISNC05] and many user programs, such as MeVisLab and 3DSlicer. Segmentations are provided as image masks in MHA format, MeVisLab's contour segmentation objects and polygonal meshes with vertex positions given in DICOM user space.

Acknowledgments

Images taken from Radiopaedia are licenced under Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported licence.

I would like to thank my adviser Prof. Andreas Kolb, my main collaborator Dr. Jan Egger, neurosurgeon Aleš Vlasák, my other collaborators, and all my colleagues from Computer Graphics and Multimedia Systems Group of University of Siegen. Without them, this thesis would not be possible.

Bibliography

[AB94] R. Adams and L. Bischof. Seeded region growing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(6):641–647, 1994. 20

[ACM$^+$07] E. D. Angelini, O. Clatz, E. Mandonnet, E. Konukoglu, L. Capelle, and H. Duffau. Glioma dynamics and computational models: A review of segmentation, registration, and in silico growth algorithms and their clinical applications. *Current Medical Imaging Reviews*, 3(4):262–276, 2007. 31

[AE98] S. L. Asa and S. Ezzat. The cytogenesis and pathogenesis of pituitary adenomas. *Endocrine Reviews*, 19(6):798–827, 1998. 28

[Aeb05] M. Aebi. The adult scoliosis. *European Spine Journal*, 14(10):925–948, 2005. 93

[APM$^+$12] I. B. Ayed, K. Punithakumar, R. Minhas, R. Joshi, and G. J. Garvin. Vertebral body segmentation in MRI via convex relaxation and distribution matching. In *Medical Image Computing and Computer Assisted Intervention*, pages 520–527. Springer, 2012. 33, 34, 36

[APR$^+$85] F. Aichner, W. Poewe, W. Rogalsky, K. Wallnöfer, J. Willeit, and F. Gerstenbrand. Magnetic resonance imaging in the diagnosis of spinal cord diseases. *Journal of Neurology, Neurosurgery & Psychiatry*, 48(12):1220–1229, 1985. 31

[BJ01] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112. IEEE, 2001. 21

[BK04] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004. 20, 30

[BL79] S. Beucher and C. Lantuéjoul. Use of watersheds in contour detection. In *International workshop on image processing, real-time edge and motion detection*, 1979. 20

[Bla08] A. M. Blamire. The technology of MRI — the next 10 years? *British Journal of Radiology*, 81(968):601–617, 2008. 1

[Bro81] C. Broit. *Optimal registration of deformed images*. PhD thesis, University of Pennsylvania, 1981. 20

[BS09] M. Buchfelder and S. Schlaffer. Surgical treatment of pituitary tumours. *Best Practice & Research Clinical Endocrinology & Metabolism*, 23(5):677 – 692, 2009. 28

[CET01] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence*, 23(6):681–685, 2001. 21

[CGBM04] J. Carballido-Gamio, S. Belongie, and S. Majumdar. Normalized cuts in 3-D for spinal MRI segmentation. *IEEE Transactions on Medical Imaging*, 23(1):36–44, January 2004. 32, 33

[CGNT09] C. Couprie, L. J. Grady, L. Najman, and H. Talbot. Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest. In *ICCV'09*, pages 731–738, 2009. http://powerwatershed.sourceforge.net/. 36, 99

[CHG^{+}98] M. Clark, L. Hall, D. Goldgof, R. Velthuizen, F. Murtagh, and M. Silbiger. Automatic tumor segmentation using knowledge-based techniques. *IEEE Transactions on Medical Imaging*, 17(2):187–201, 1998. 30, 50

[Cob48] J. Cobb. Outline for the study of scoliosis. *Instructional Course Lectures*, 5:261–275, 1948. 93

[Coh91] L. D. Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, 1991. 41, 59

[CRS98] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, 1998. 21, 101

[CTCG95] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61(1):38 – 59, 1995. 21

[DCH88] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *SIGGRAPH Comput. Graph.*, 22(4):65–74, June 1988. 22

[DLG90] N. Dyn, D. Levine, and J. A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9(2):160–169, April 1990. 63

[DLSH02] C. Davatzikos, D. Liu, D. Shen, and E. H. Herskovits. Spatial normalization of spine MR images for statistical correlation of lesions with clinical symptoms. *Radiology*, 224(3):919–926, 2002. 21, 33, 34, 35

[DMRS05] M. Droske, B. Meyer, M. Rumpf, and C. Schaller. An adaptive level set method for interactive segmentation of intracranial tumors. *Neurological Research*, 27(4):363–370, June 2005. 30

[EBK$^+$10] J. Egger, M. Bauer, D. Kuhnt, B. Carl, C. Kappus, B. Freisleben, and C. Nimsky. Nugget-cut: A segmentation scheme for spherically- and elliptically-shaped 3d objects. In M. Goesele, S. Roth, A. Kuijper, B. Schiele, and K. Schindler, editors, *Pattern Recognition*, volume 6376 of *Lecture Notes in Computer Science*, pages 373–382. Springer Berlin Heidelberg, 2010. 20, 30

[EBK$^+$11a] J. Egger, M. H. A. Bauer, D. Kuhnt, B. Carl, C. Kappus, B. Freisleben, and C. Nimsky. Evaluation of a novel approach for automatic volume determination of glioblastomas based on several manual expert segmentations. *Computing Research Repository*, abs/1103.1474, 2011. 82, 104

[EBK$^+$11b] J. Egger, M. H. A. Bauer, D. Kuhnt, B. Freisleben, and C. Nimsky. Pituitary adenoma segmentation. *Computing Research Repository*, abs/1103.1778, 2011. 30

[EBK$^+$11c] J. Egger, M. H. A. Bauer, D. Kuhnt, C. Kappus, B. Carl, B. Freisleben, and C. Nimsky. A flexible semi-automatic approach for glioblastoma multiforme segmentation. *Computing Research Repository*, abs/1103.1777, 2011. 30

[EKD$^+$12] J. Egger, T. Kapur, T. Dukatz, M. Kolodziej, Dž. Zukić, B. Freisleben, and C. Nimsky. Square-Cut: A Segmentation

Algorithm on the Basis of a Rectangle Shape. *PLoS ONE*, 7(2):e31064, 02 2012. 32

[EZB+10] J. Egger, Dž. Zukić, M. H. A. Bauer, D. Kuhnt, B. Carl, B. Freisleben, A. Kolb, and C. Nimsky. A Comparison of Two Human Brain Tumor Segmentation Methods for MRI Data. In A. Alexandrov and et al., editors, *6th Russian-Bavarian Conference on Bio-Medical Engineering*, pages 9–13. Bauman Moscow State Technical University (BMSTU), November 2010. 4, 39, 50

[EZF+13] J. Egger, Dž. Zukić, B. Freisleben, A. Kolb, and C. Nimsky. Segmentation of pituitary adenoma: Graph-based vs. balloon inflation. *Computer Methods and Programs in Biomedicine*, 110(3):268–278, June 2013. 4, 39, 52

[FBW01] P. Felkel, M. Bruckschwaiger, and R. Wegenkittl. Implementation and complexity of the watershed-from-markers algorithm computed as a minimal cost forest. *Computer Graphics Forum*, 20(3):26–35, 2001. 31

[FHA+09] J. Freburger, G. Holmes, R. Agans, A. Jackman, J. Darter, A. Wallace, L. Castel, W. Kalsbeek, and T. Carey. The rising prevalence of chronic low back pain. *Archives of Internal Medicine*, 169(3):251–258, February 2009. 31

[FL03] J.-J. Fernández and S. Li. An improved algorithm for anisotropic nonlinear diffusion for denoising cryo-tomograms. *Journal of Structural Biology*, 144(1-2):152–161, 2003. 79

[GBBH96] P. Gibbs, D. L. Buckley, S. J. Blackband, and A. Horsman. Tumour volume determination from mr images by morphological segmentation. *Physics in Medicine and Biology*, 41(11):2437, 1996. 29, 49

[GMA+04] V. Grau, A. Mewes, M. Alcaniz, R. Kikinis, and S. K. Warfield. Improved watershed transform for medical image segmentation using prior information. *Medical Imaging, IEEE Transactions on*, 23(4):447–458, 2004. 20

[GS83] L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of voronoi diagrams. In *ACM symposium on Theory of computing*, pages 221–234, New York, NY, USA, 1983. ACM. 39

[GS04] S. Ghebreab and A. Smeulders. Combining strings and necklaces for interactive three-dimensional segmentation of spinal images using an integral deformable spine model. *IEEE Transactions on Biomedical Engineering*, 51(10):1821–1829, October 2004. 33

[HCLN09] S.-H. Huang, Y.-H. Chu, S.-H. Lai, and C. Novak. Learning-based vertebra detection and iterative normalized-cut segmentation for spinal MRI. *IEEE Transactions on Medical Imaging*, 28(10):1595–1605, October 2009. 32, 33

[Hig13] D. Higgins. Motion artefact, 2013. [Online; accessed 2013-10-11] `http://www.revisemri.com/questions/artefacts/motion_phase`. 15

[HM02] C. L. Hoad and A. L. Martel. Segmentation of MR images for computer-assisted surgery of the lumbar spine. *Physics in Medicine and Biology*, 47(19):3503, 2002. 20, 33, 34, 35

[HRSE+00] P. Hastreiter, C. Rezk-Salama, K. Eberhardt, B. Tomandl, and T. Ertl. Functional analysis of the vertebral column based on MR and direct volume rendering. *Medical Image Computing and Computer Assisted Intervention*, 2000. 31

[ISKF+09] G. Iliadis, P. Selviaridis, A. Kalogera-Fountzila, A. Fragkoulidi, D. Baltas, N. Tselis, A. Chatzisotiriou, D. Misailidou, N. Zamboglou, and G. Fountzilas. The importance of tumor volume in the prognosis of patients with glioblastoma. *Strahlentherapie und Onkologie*, 185(11):743–750, 2009. 29

[ISNC05] L. Ibáñez, W. Schroeder, L. Ng, and J. Cates. *The ITK Software Guide*. Kitware, Inc., second edition, 2005. 124

[Jäg11] F. Jäger. *Normalization of Magnetic Resonance Images and Its Application to the Diagnosis of the Scoliotic Spine*, volume 34 of *Studien zur Mustererkennung*. Logos Verlag Berlin GmbH, 2011. 8

[KHDM98] J. Kittler, M. Hatef, R. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226 –239, March 1998. 79

[KJW+03] R.-D. Kortmann, B. Jeremic, M. Weller, L. Plasswilm, and M. Bamberg. Radiochemotherapy of malignant glioma in adults. *Strahlentherapie und Onkologie*, 179(4):219–232, 2003. 28

[KLS+02] P. Kleihues, D. N. Louis, B. W. Scheithauer, L. B. Rorke, G. Reifenberger, P. C. Burger, and W. K. Cavenee. The who classification of tumors of the nervous system. *Journal of Neuropathology & Experimental Neurology*, 61(3):215–225, March 2002. 27

[Kob00] L. Kobbelt. $\sqrt{3}$ subdivision. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 103–112, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 123

[KOE$^+$09] T. Klinder, J. Ostermann, M. Ehm, A. Franz, R. Kneser, and C. Lorenz. Automated model-based vertebra detection, identification, and segmentation in CT images. *Medical Image Analysis*, 13(3):471 – 482, 2009. 31, 34

[KWL$^+$08] T. Klinder, R. Wolz, C. Lorenz, A. Franz, and J. Ostermann. Spine segmentation using articulated shape models. In D. Metaxas, L. Axel, G. Fichtinger, and G. Székely, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI 2008)*, volume 5241 of *Lecture Notes in Computer Science*, pages 227–234. Springer Berlin / Heidelberg, 2008. 33, 34

[KWT88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988. 20

[KWZ$^+$13] B. M. Kelm, M. Wels, S. K. Zhou, S. Seifert, M. Suehling, Y. Zheng, and D. Comaniciu. Spine detection in CT and MR using iterated marginal space learning. *Medical Image Analysis*, 17(8):1283 – 1292, 2013. 22, 34, 35, 36, 105

[Lab13] B. Labitzke. *Visualization and Analysis of Multispectral Image Data*. PhD thesis, Naturwissenschaftlich-Technischen Fakultät, Universität Siegen, October 2013. 7

[LASF$^+$01] M. Lacroix, D. Abi-Said, D. R. Fourney, Z. L. Gokaslan, W. Shi, F. DeMonte, F. F. Lang, I. E. McCutcheon, S. J. Hassenbusch, E. Holland, K. Hess, C. Michael, D. Miller, and R. Sawaya. A multivariate analysis of 416 patients with glioblastoma multiforme: prognosis, extent of resection, and survival. *Journal of Neurosurgery*, 95(2):190–198, 2001. PMID: 11780887. 28

[LLT03] M. Lysaker, A. Lundervold, and X.-C. Tai. Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *IEEE Transactions on Image Processing*, 12(12):1579–1590, 2003. 17

[LM02] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *IEEE International Conference on Image Processing*, volume 1, pages 900–903, 2002. 86

[LOD$^+$04] M. M. Letteboer, O. F. Olsen, E. B. Dam, P. W. Willems, M. A. Viergever, and W. J. Niessen. Segmentation of tumors in magnetic resonance brain images using an interactive multiscale watershed algorithm. *Academic Radiology*, 11(10):1125 – 1138, 2004. 29

[LRDG04] L. Lenchik, L. F. Rogers, P. D. Delmas, and H. K. Genant. Diagnosis of osteoporotic vertebral fractures: importance of recognition and description by radiologists. *American Journal of Roentgenology*, 183(4):949–958, 2004. 95

[MCP$^+$09] S. Michopoulou, L. Costaridou, E. Panagiotopoulos, R. Speller, G. Panayiotakis, and A. Todd-Pokropek. Atlas-based segmentation of degenerated lumbar intervertebral discs from MR images of the spine. *IEEE Transactions on Biomedical Engineering*, 56(9):2225–2231, September 2009. 32, 33

[MEM02] P. Mildenberger, M. Eichelberg, and E. Martin. Introduction to the DICOM standard. *European Radiology*, 12(4):920–927, 2002. 4

[MFTM01] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision*, volume 2, pages 416–423, July 2001. 2

[ML13] J. Ma and L. Lu. Hierarchical segmentation and identification of thoracic vertebra using learning-based edge detection and coarse-to-fine deformable model. *Computer Vision and Image Understanding*, 117(9):1072 – 1083, 2013. 31, 34

[MSM$^+$08] Y. Masharawi, K. Salame, Y. Mirovsky, S. Peleg, G. Dar, N. Steinberg, and I. Hershkovitz. Vertebral body shape variation in the thoracic and lumbar spine: Characterization of its asymmetry and wedging. *Clinical Anatomy*, 21(1):46–54, 2008. 76

[NFE$^+$12] A. Neubert, J. Fripp, C. Engstrom, R. Schwarz, L. Lauer, O. Salvado, and S. Crozier. Automated detection, 3D segmentation and analysis of high resolution spine MR images using statistical shape models. *Physics in Medicine and Biology*, 9:8357–8376, 2012. 21, 22, 34, 36, 98, 101, 117, 118

[NFS$^+$11] A. Neubert, J. Fripp, K. Shen, O. Salvado, R. Schwarz, L. Lauer, C. Engstrom, and S. Crozier. Automated 3D segmentation of vertebral bodies and intervertebral discs from MRI. In *International Conference on Digital Image Computing Techniques and Applications*, pages 19–24, 2011. 33, 34, 36, 117, 118

[NWF$^+$05] A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl, and K. Bühler. Advanced virtual endoscopic pituitary surgery. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):497–507, 2005. 30

[OS88] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988. 19

[Ots79] N. Otsu. A threshold selection method from gray-level histograms. *Systems, Man and Cybernetics, IEEE Transactions on*, 9(1):62–66, Jan 1979. 19

[PBHG04] M. Prastawa, E. Bullitt, S. Ho, and G. Gerig. A brain tumor segmentation framework based on outlier detection. *Medical Image Analysis*, 8(3):275 – 283, 2004. 30, 49

[Pow94] M. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*, volume 275 of *Mathematics and Its Applications*, pages 51–67. Springer Netherlands, 1994. 96

[PXP00] D. L. Pham, C. Xu, and J. L. Prince. Current methods in medical image segmentation. *Annual review of biomedical engineering*, 2(1):315–337, 2000. 19

[PZWhL05] Z. Peng, J. Zhong, W. Wee, and J. huei Lee. Automated vertebra detection and segmentation from the whole spine MR images. In *27th Annual International Conference of the Engineering in Medicine and Biology Society*, pages 2527–2530, January 2005. 32, 33

[Rad13a] Radiopaedia. Chemical shift artifact, 2013. [Online; accessed 2013-10-14] `http://radiopaedia.org/articles/chemical_shift_artifact`. 15

[Rad13b] Radiopaedia. Quadrigeminal plate lipoma, 2013. [Online; accessed 2013-10-14] `http://radiopaedia.org/cases/quadrigeminal-plate-lipoma`. 17

[RET13] M. Rak, K. Engel, and K.-D. Tönnies. Closed-form hierarchical finite element models for part-based object detection. In *Vision, Modeling and Visualization*, pages 137–144. Eurographics, 2013. 34

[RGMB10] P. J. Richards, J. George, M. Metelko, and M. Brown. Spine computed tomography doses and cancer induction. *Spine*, 35(4):430–433, 2010. 31

[Sar87] H. Saraste. Long-term clinical and radiological follow-up of spondylolysis and spondylolisthesis. *Journal of Pediatric Orthopaedics*, 7(6):631–638, 1987. 94

[ŠBSG06] P. Šereda, A. Bartrolì, I. Serlie, and F. Gerritsen. Visualization of boundaries in volumetric data sets using LH histograms. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):208–218, March 2006. 23, 24, 59, 79

[SHD01] D. Shen, E. H. Herskovits, and C. Davatzikos. An adaptive-focus statistical shape model for segmentation and shape modeling of 3-D brain structures. *IEEE Transactions on Medical Imaging*, 20(4):257–270, 2001. 35

[SHP01] C. Sieg, H. Handels, and S. Pöppl. Automatic segmentation of contrast-enhanced brain tumors in multispectral mr-images with backpropagation-networks. *Bildverarbeitung für die Medizin*, pages 347–351, 2001. 30

[ŠLPV10] D. Štern, B. Likar, F. Pernuš, and T. Vrtovec. Automated detection of spinal centrelines, vertebral bodies and intervertebral discs in CT and MR images of lumbar spine. *Physics in Medicine and Biology*, 55(1):247, 2010. 85

[ŠLPV11] D. Štern, B. Likar, F. Pernuš, and T. Vrtovec. Parametric modelling and segmentation of vertebral bodies in 3D CT and MR spine images. *Physics in Medicine and Biology*, 56(23):7505, 2011. 34, 35

[SMB11] F. Schulze, D. Major, and K. Bühler. Fast and memory efficient feature detection using multiresolution probabilistic boosting trees. *19th International Conference in Central Europe on Computer*

Graphics, Visualization and Computer Vision, 19(1):pp. 33 –40, 2011. 34

[SS01] L. G. Shapiro and G. C. Stockman. *Computer Vision*. Prentice Hall, 2001. 1, 19

[SSQW07] R. Shi, D. Sun, Z. Qiu, and K. Weiss. An efficient method for segmentation of MRI spine images. In *IEEE/ICME International Conference on Complex Medical Engineering*, pages 713–717, May 2007. 32, 33

[ŠVPL11] D. Štern, T. Vrtovec, F. Pernuš, and B. Likar. Segmentation of vertebral bodies in CT and MR images based on 3D deterministic models. In R. M. Summers and B. van Ginneken, editors, *SPIE Medical Imaging*, volume 7963. SPIE, February 2011. 33

[SWM$^+$06] M. Sampat, Z. Wang, M. Markey, G. Whitman, T. Stephens, and A. Bovik. Measuring intra- and inter-observer agreement in identifying and localizing structures in medical images. In *IEEE International Conference on Image Processing*, pages 81–84, 2006. 46

[TG09] N. Tustison and J. Gee. Introducing Dice, Jaccard, and Other Label Overlap Measures To ITK. *The Insight Journal*, 12 2009. 21

[The92] H. Theil. A rank-invariant method of linear and polynomial regression analysis. In *Henri Theil's Contributions to Economics and Econometrics*, pages 345–381. Springer, 1992. 89

[VJ01] P. Viola and M. J. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Vision & Pattern Recognition*, volume 1, pages 511–518, 2001. 25, 83, 84, 86, 104

[VLP05] T. Vrtovec, B. Likar, and F. Pernuš. Automated curved planar reformation of 3D spine images. *Physics in Medicine and Biology*, 50(19):4527, 2005. 85

[Wik13] Wikipedia. Physics of magnetic resonance imaging, 2013. [Online; accessed 2013-10-02] `http://en.wikipedia.org/w/index.php?title=Physics_of_magnetic_resonance_imaging&oldid=563987513`. 11

[WKL$^+$01] J. Weese, M. Kaus, C. Lorenz, S. Lobregt, R. Truyen, and V. Pekar. Shape constrained deformable models for 3D medical image segmentation. In M. Insana and R. Leahy, editors, *Information Processing in Medical Imaging*, volume 2082 of *Lecture Notes in Computer Science*, pages 380–387. Springer Berlin / Heidelberg, 2001. 33, 34

[WLJG95] C. Wu, J. Li, M. Jergas, and H. Genant. Comparison of semiquantitative and quantitative techniques for the assessment of prevalent and incident vertebral fractures. *Osteoporosis International*, 5(5):354–370, 1995. 95

[WNB$^+$06] S. Wolfsberger, A. Neubauer, K. Bühler, R. Wegenkittl, T. Czech, S. Gentzsch, H.-G. Bocher-Schwarz, and E. Knosp. Advanced virtual endoscopy for endoscopic transsphenoidal pituitary surgery. *Neurosurgery*, 59(5):1001–1010, November 2006. 30

[WW83] L. L. Wiltse and R. B. Winter. Terminology and measurement of spondylolisthesis. *The Journal of Bone and Joint Surgery*, 65(6):768–772, July 1983. 94

[WZP$^+$07] M.-A. Weber, S. Zoubaa, M. Politi, I. Grunwald, and M. Essig. Tumoren der sellaregion. *Der Radiologe*, 47(6):492–500, 2007. 28

[YOS06] J. Yao, S. O'Connor, and R. Summers. Automated spinal column extraction and partitioning. In *IEEE International Symposium on Biomedical Imaging: Nano to Macro*, pages 390–393, April 2006. 33, 34

[ZEB$^+$10] Dž. Zukić, J. Egger, M. H. A. Bauer, D. Kuhnt, B. Carl, B. Freisleben, A. Kolb, and C. Nimsky. Glioblastoma Multiforme Segmentation in MRI Data with a Balloon Inflation Approach. In A. Alexandrov and et. al., editors, *6th Russian-Bavarian Conference on Bio-Medical Engineering*, pages 40–44. Bauman Moscow State Technical University (BMSTU), November 2010. 4, 39

[ZEB$^+$11] Dž. Zukić, J. Egger, M. H. A. Bauer, D. Kuhnt, B. Carl, B. Freisleben, A. Kolb, and C. Nimsky. Preoperative Volume Determination for Pituitary Adenoma. In R. M. Summers and B. van Ginneken, editors, *SPIE Medical Imaging*, volume 7963, page 79632T. SPIE, February 2011. 4, 39

[ZSS96] D. Zorin, P. Schröder, and W. Sweldens. Interpolating subdivision for meshes with arbitrary topology. In *ACM SIGGRAPH*, pages 189–192, New York, NY, USA, 1996. ACM. 63

[ZVD$^+$12] Dž. Zukić, A. Vlasák, T. Dukatz, J. Egger, D. Hořínek, C. Nimsky, and A. Kolb. Segmentation of Vertebral Bodies in MR

Images. In T. Grosch, H. Theisel, K. Toennies, and B. Preim, editors, *Vision, Modeling and Visualization*, pages 135–142. Eurographics, Eurographics Digital Library, November 2012. 4, 55, 75

[ZVE+14] Dž. Zukić, A. Vlasák, J. Egger, D. Hořínek, C. Nimsky, and A. Kolb. Robust Detection and Segmentation for Diagnosis of Vertebral Diseases using Routine MR Images. *Computer Graphics Forum*, 33(6):190–204, 2014. 4, 83, 124

[ZWB+04] K. H. Zou, S. K. Warfield, A. Bharatha, C. M. Tempany, M. R. Kaus, S. J. Haker, W. M. W. III, F. A. Jolesz, and R. Kikinis. Statistical validation of image segmentation quality based on a spatial overlap index. *Academic Radiology*, 11(2):178 – 189, 2004. 21, 46