DISSERTATION
submitted
to the
Combined Faculties for the Natural Sciences and for Mathematics
of the
Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

Put forward by

Diplom-Inf. Dzmitry Hlindzich
Born in Minsk, Belarus

# Medical image analysis methods for anatomical surface reconstruction using tracked 3D ultrasound

Advisor: Prof. Dr. Reinhard Männer

# Abstract

The thesis focuses on a study of techniques for acquisition and reconstruction of surface data from anatomical objects by means of tracked 3D ultrasound. In the context of the work two experimental scanning systems are developed and tested on both artificial objects and biological tissues. The first system is based on the freehand ultrasound principle and utilizes a conventional 2D ultrasound transducer coupled with an electromechanical 3D position tracker. The main properties and the basic features of this system are discussed. A number of experiments show that its accuracy in the close to ideal conditions reaches 1.2 mm RMS.

The second proposed system implements the sequential triggered scanning approach. The system consists of an ultrasound machine, a workstation and a scanning body (a moving tank filled with liquid and a transducer fixation block) that performs transducer positioning and tracking functions. The system is tested on artificial and real bones. The performed experiments illustrate that it provides significantly better accuracy than the freehand ultrasound (about 0.2 mm RMS) and allows acquiring regular data with a good precision. This makes such a system a promising tool for orthopaedic and trauma surgeons during contactless X-ray-free examinations of injured extremities.

The second major subject of the thesis concerns development of medical image analysis methods for 3D surface reconstruction and 2D object detection. We introduce a method based on mesh-growing surface reconstruction that is designed for noisy and sparse data received from 3D tracked ultrasound scanners. A series of experiments on synthetic and ultrasound data show an appropriate reconstruction accuracy. The reconstruction error is measured as the averaged distance between the faces of the mesh and the points from the cloud. Dependently on the initial settings of the method the error varies in range 0.04 - 0.2% for artificial data and 0.3 - 0.7 mm for ultrasound bone data. The reconstructed surfaces correctly interpolate the original point clouds and demonstrate proper smoothness.

The next significant problem considered in the work is 2D object detection. Although medical object detection is not integrated into the developed scanning systems, it can be used as a possible further extension of the systems for automatic detection of specific anatomical structures. We analyse the existent object detection methods and introduce a modification of the one based on the popular Generalized Hough Transform (GHT). Unlike the original GHT, the developed method is invariant to rotation and uniform scaling, and uses an intuitive two-point parametrization. We propose several implementations of the feature-to-vote conversion function with the corresponding vote analysis principles. Special attention is devoted to a study of the hierarchical vote analysis and its probabilistic properties. We introduce a parameter space subdivision strategy that reduces the probability of vote peak omission, and show that it can be efficiently implemented in practice using the Gumbel probability distribution.

# Zusammenfassung

Diese Arbeit konzentriert sich auf die Studie von Techniken zur Erfassung und Rekonstruktion der Oberflächendaten anatomischer Objekte mittels Tracked 3D Ultrasound. Im Rahmen dieser Arbeit werden zwei experimentelle Scansysteme entwickelt und mit Hilfe sowohl von künstlichen Objekten und als auch biologischen Geweben verifiziert. Das erste System basiert auf dem Freihandultraschallprinzip und verwendet einen herkömmlichen 2D Ultraschallwandler gekoppelt mit einem elektromechanischen 3D Positionsverfolger. Die wichtigsten Fähigkeiten und die grundlegenden Eigenschaften dieses Systems werden diskutiert. Die durchgeführten Experimente zeigen, dass die Genauigkeit dieses Systems unter idealen Bedingungen 1.2 mm RMS erreichen kann.

Das zweite vorgeschlagene System basiert auf dem sequentiellen Scanansatz. Das System besteht aus einem Ultraschallgerät, einer Workstation und einem Scannerkörper (ein beweglicher flüssigkeitsgefüllter Behälter mit einer Befestigungsvorrichtung für den Schallkopf). Im Gegensatz zum Freihandultraschall wird der Schallkopf mechanisch positioniert. Dieses System wird anhand künstlicher und echter Knochendatensätze getestet. Die durchgeführten Experimente veranschaulichen, dass der sequentielle Scanansatz eine deutlich höhere Genauigkeit als der Freihandultraschall hat (etwa 0.2 mm RMS) und eine Erfassung der regelmäßigen Daten mit guter Präzision ermöglicht. Dies macht das System zu einem vielversprechenden Werkzeug für Orthopäden und Unfallchirurgen bei röntgenlosen Untersuchungen verletzter Extremitäten.

Das zweite große Thema der Arbeit betrifft die Entwicklung von medizinischen Bildanalysemethoden für die 3D Oberflächenrekonstruktionen und 2D Objekterkennungen. Wir stellen eine Methode basierend auf der Mesh-growing-Oberflächenrekonstruktion vor, die für verrauschte und spärliche Daten geeignet ist. Eine Reihe von Experimenten, basierend auf Ultraschall- und synthetischen Daten, zeigen eine gute Rekonstruktionsgenauigkeit. Der Rekonstruktionsfehler wird als der gemittelte Abstand zwischen den Flächen eines 3D-Gitters und einer Punktwolke berechnet. In Abhängigkeit von den Grundeinstellungen der Methode variiert der Fehler für künstliche Daten von 0.04 bis 0.2%, bei realen Knochen von 0.3 bis 0.7 mm. Die rekonstruierten Oberflächen interpolieren die ursprünglichen Punktwolken korrekt und zeigen die richtige Glätte.

Das nächste wichtige Problem, das in der Arbeit betrachtet wird, ist die 2D Objekterkennung. Obwohl die medizinische Objekterkennung nicht in die entwickelte Scansysteme integriert wird, kann sie als eine mögliche Erweiterung der Systeme zur automatischen Erkennung der spezifischen anatomischen Strukturen verwendet werden. Wir analysieren die vorhandenen Objekterkennungsmethoden und entwickeln eine Modifikation, die auf der bekannten Generalized Hough-Transformation (GHT) basiert. Im Gegensatz zur originalen GHT, ist die entwickelte Methode invariant bezüglich Rotation und gleichmäßiger Skalierung, und verwendet eine intuitive Zwei-Punkt-Parametrisierung. Dazu stellen wir mehrere Implementierungen der Feature-to-Vote Konvertierungsfunktion mit den entsprechenden Vote Analyse Prinzip vor.

Besondere Aufmerksamkeit wird dabei auf die Studie des hierarchischen Vote Analyse Prinzips und ihren probabilistischen Eigenschaften gelegt. Wir schlagen eine Parameterraumunterteilungsstrategie vor, die die Wahrscheinlichkeit der fehlenden Vote Maxima reduziert, und zeigen, dass diese Strategie mittels der Gumbel Wahrscheinlichkeitsverteilung effizient realisiert werden kann.

# Acknowledgments

This thesis presents the results of the work, which I have done at the Institute for Computer Engineering, Heidelberg University, Germany. First of all I would like to thank Prof. Dr. Reinhard Männer for his guidance and thorough support during all this time. Particular thanks to Prof. Dr. Jürgen Hesser for important remarks and a lot of advice that have significantly improved the quality of the thesis.

This work could not be possible without the exceptional contribution of Dr. Aleh Kryvanos. I would like to thank him for the invaluable scientific support, great ideas and productive discussions during our collaboration in numerous projects.

I am also grateful to Prof. Dr. med. Markus Schwarz (Department of Experimental Orthopedic and Trauma Surgery, University Medical Center Mannheim), Dr. med. Frederic Bludau (Orthopaedic and Trauma Surgery Center, University Medical Center Mannheim) and Dr. med. Christina Dösch (First Medical Clinic of the University Hospital Mannheim) for the pleasant and productive cooperation.

Dzmitry Hlindzich
Mannheim, January 2014

# Notation

The following abbreviations are used in this thesis:

| | |
|---|---|
| **CT** | Computed Tomography |
| **DPGHT** | Dual-Point Generalized Hough Transform |
| **ECG** | Electrocardiography |
| **GHT** | Generalized Hough Transform |
| **GPU** | Graphics Processing Unit |
| **HT** | Hough Transform |
| **ICP** | Iterative Closest Point |
| **MRF** | Markov Random Field |
| **MRI** | Magnetic Resonance Imaging |
| **PACS** | Picture Archiving and Communication System |
| **PCA** | Principle Component Analysis |
| **PET** | Positron Emission Tomography |
| **RMS** | Root Mean Square |
| **RT** | Radon Transform |
| **US** | Ultrasound |

# Contents

# 1. Introduction

Medical imaging has brought medicine to a new stage. After the invention of X-rays and acquisition of first images of the human body at the end of 19-th century, medical imaging has been progressively developing. Today it provides a variety of new possibilities for modern clinical diagnostics, treatment and anatomical study. Supporting physicians with visual information it allows significantly improving the quality of medical care. However, the recent advance would be hardly possible without the corresponding development of the methods of medical image analysis. Owing to this mutual effort, the data acquired from scanners can be appropriately reconstructed, analysed and prepared for visualization.

In spite of the apparent progress modern medical imaging modalities are still quite far from ideal. Use of ionized radiation, imperfect image quality and high costs are the typical problems that can be met here. Moreover, majority of the applied today methods of image analysis also need further development and improvements in quality and speed. Therefore, further research in these both fields of science is highly essential today.

## 1.1. Motivation and objectives

Ultrasound is a medical imaging modality that has a high potential as it is harmless for patients, relatively cheap and mobile. A combination of these features makes it unique among the other imaging techniques. However, the problems of efficient 3D data acquisition, reconstruction and analysis (including the specific artefacts interpretation) are quite acute for ultrasound. This, in turn, makes this modality inferior to e.g. magnetic resonance or computed tomography, where high-quality 3D imaging has become a standard. In this work we aim at a possible solution of the afore mentioned problems. We study the 3D surface reconstruction of anatomical objects acquired by conventional 2D ultrasound transducers, and perform its evaluation. Along with this we consider a number of accompanying image analysis problems that occurred during the project. The obtained results can be important for the development of a clinical 3D-tracked ultrasound scanning system as well as for all those who are doing research in this field. We also provide a detailed study of object recognition based on the Generalized Hough Transform, that covers a substantial part of this thesis. Although it is not directly used as a part of our surface reconstruction system, it can be applied in a number of additional medical image analysis tasks for automatic detection of specific anatomical structures.

Summarizing the above, in this thesis we are focusing on the following main *objectives*. The first one is to assemble an ultrasound-based scanning system suitable for 3D reconstruction of anatomical surfaces of a) viscera and b) the long bones of the human

extremities (engineering part). The second one is to develop the corresponding medical image processing algorithms for a) 3D reconstruction of surfaces from the acquired ultrasound data, and b) rotation-scale invariant object detection (theoretical part).

## 1.2. Structure of the work

The work is divided into two parts. The first part concerns the problems of medical imaging and scanner engineering, and consists of two chapters. In Chapter 2 we provide a review of the existent medical imaging modalities and discuss their basic features. Chapter 3 is devoted to development of two experimental scanning systems based on the tracked 3D ultrasound. The first system (Section 3.3) realizes the freehand tracked ultrasound paradigm, while the second one (Section 3.4) is designed to provide an automated scanning process. The both systems are tested on artificial objects and some biological tissues.

The second part of the work is dedicated to medical image analysis. In Chapter 4 we present a 3D surface reconstruction method based on growing meshes that is utilized in the mentioned scanning systems. Chapters 5 and 6 are devoted to image segmentation and object detection problems. In Chapter 5 an overview of the existent medical image segmentation methods is provided. In Chapter 6 we discuss modern object detection methods, study their properties and develop an alternative rotation-scale invariant 2D object detection method based on the Generalized Hough Transform.

# Part I.

# Medical imaging techniques

# 2. Existent medical imaging techniques

Medical imaging, in its general form, is defined as a medical diagnosis subdiscipline involving (noninvasive) scientific study of the human organism aimed at the acquisition of human body images. From the technical point of view, that is the main objective of this work, medical imaging is considered as a noninvasive process used to create images of the inner anatomical structures.

Modern medical imaging is mainly presented by a number of modalities that utilize such physical phenomena as X-rays, nuclear magnetic resonance, ultrasound reflection and electrical activity in the human body. Regardless of the selected modality the modern medical image acquisition holds on the typical scheme that consists of the following sequence of steps: body scanning and image reconstruction (i.e. medical imaging tasks, Chapter 3), image processing and analysis (see Chapters 4 - 6), with optional storage of the images in a picture archiving and communication system (PACS)[89].

In the next sections we will shortly discuss the most important existent medical imaging techniques and present two experimental scanning systems for reconstruction of 3D surfaces of human organs based on tracked 3D ultrasound.

## 2.1. Radiography

X-ray is a form of ionizing electromagnetic radiation that was discovered by Wilhelm Röntgen in 1895 while he was experimenting with cathode tubes. A cathode tube consists of a vacuum tube with a cathode and an anode. The electrons released at the cathode by thermal excitation are accelerated toward the anode by a voltage between the cathode and the anode. These electrons collide with the anode and release their energy in the form of X-rays and characteristic radiation.

Due to their ionizing nature, X-rays exert a harmful influence upon biological tissues. They can break molecular bonds, especially those of DNA and hence bare a potential risk of radiation induced cancer.

In medical imaging the most significant feature of X-ray beams is associated with their attenuation while passing through biological tissues that is described by the Beer–Lambert law [185]. Using this property of X-rays and special detectors (films, storage phosphor screens, photo-conductors and digital matrices) the beams can be captured and converted to images (radiographs). The resolution of the radiographs mostly depends on the focus size of the X-ray tube, the spatial resolution of the detector and the amount of X-ray scattering inside the body. Modern radiography uses digital detectors with scatter grids that are able to filter scatter by absorbing photons with a large incidence angle and passing photons with a small one. Such scanning systems allow

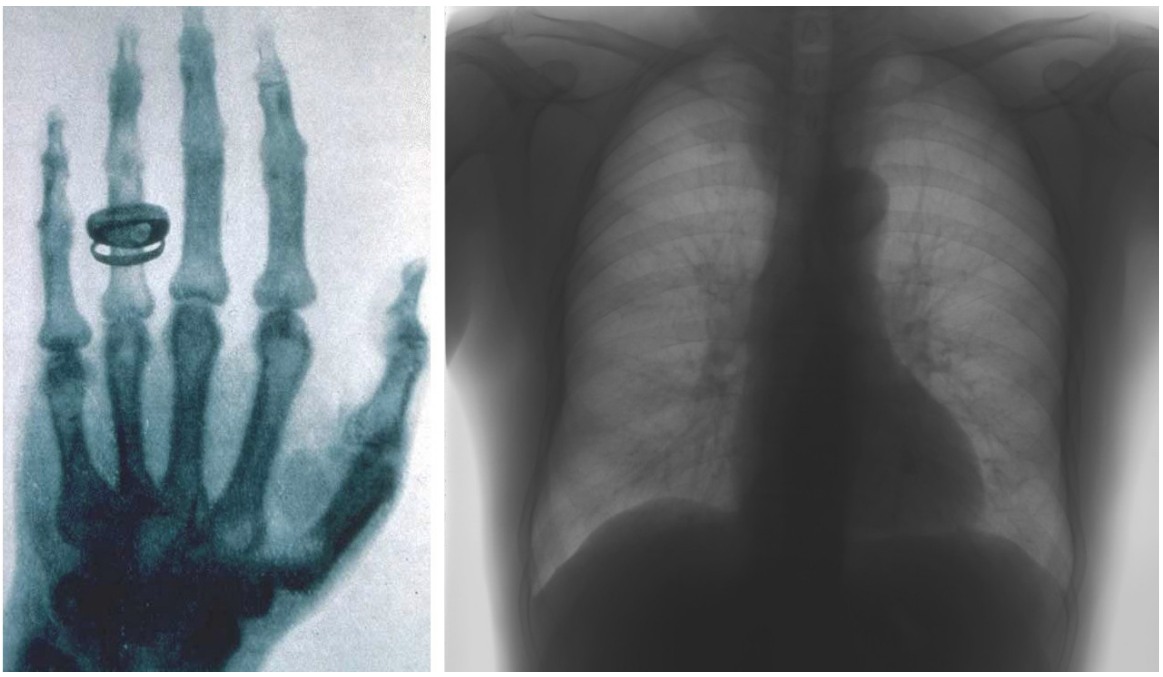

Figure 2.1.: Historical scan of Albert von Köllikers hand made by Wilhelm Röntgen in 1896 (left). Modern chest X-ray, a 56 year old male with a suspected hamartoma (right), JSRT free digital image database [179].

producing image resolution up to 30 μm per pixel, that is sufficient for the most medical applications.

Today X-rays are widely used for skeletal and chest imaging, mammography, dental imaging, fluoroscopy and angiography. However, it should be taken into account that even small doses of X-ray radiation are capable to damage cells, that can lead to cancer or genetic changes. Decreasing the radiation dose, in turn, has negative effect at the signal to noise ratio of radiographs as the noise amplitude is proportional to the square root of the signal amplitude [185]. In spite of the fact that the modern medical radiography provides good quality of images and much less suffers from artefacts than other modalities, it gradually gives way to other medical imaging techniques.

## 2.2. Computed Tomography

Computed tomography (CT), or X-ray computed tomography, is a medical imaging modality that utilizes computer-processed X-rays to produce cross-sectional images of the body. The first CT scanner was developed in 1972 by Godfrey Hounsfield (engineer at EMI Ltd.), and was based on the mathematical theory developed by Allen Cormack. The basic architecture of a CT scanner is represented by a X-ray source that rotates around the scanned object, placed in the rotation center, and X-ray sensors that are positioned on the diametrically opposite side from the X-ray source relative to the patient body. The X-ray source produces a geometrically narrow X-ray beam of intensity $I_0$ that traverses the scanned body along the line $L(\rho, \theta)$ and is detected by the sensors. If the

X-ray source-sensors system is rotated parallel to $XY$ plane, then $L(\rho, \theta)$ defines the line that makes an angle $\theta$ with the Y-axis at distance $\rho$ from the origin.

Accordingly to the Beer–Lambert law, the intensity profile $I(\rho, \theta)$ of the outgoing beam can be calculated by the formula:

$$I(\rho, \theta) = I_0 \; e^{-\int_{L(\rho,\theta)} \mu(x,y) ds},$$

where $\mu(x, y)$ is the unknown attenuation function. Using this relation the projection of $\mu(x, y)$ along the angle $\theta$ is calculated as:

$$p(\rho, \theta) = -ln \frac{I(\rho, \theta)}{I_0} = \int_{L(\rho,\theta)} \mu(\rho \cos\theta - s \sin\theta, \rho \sin\theta + s \cos\theta) ds, \qquad (2.1)$$

that is nothing else as the Radon Transform [161] of function $\mu(x, y)$.

The set of all projections $p(\rho, \theta)$ for some discretion of the parameters $\Delta\rho$ and $\Delta\theta$ represents a 2D dataset called a sinogram. The main computation problem of CT is to reconstruct the unknown function $\mu(x, y)$ from the given set $p(\rho, \theta)$, i.e. to calculate the *inverse Radon Transform* of the sinogram. This problem can be solved using different reconstruction algorithms [88] including simple backprojection, filtered backprojection (convolution), series expansion, and direct Fourier reconstruction using the Projection-slice theorem [185]. The numbers provided by the reconstruction algorithm are gathered into a 2D image, each pixel of which represents a CT number (CTN, measured in Hounsfield units) that is related to the linear attenuation coefficients $\mu$ as $CTN = 1000(\mu - \mu_w)/\mu_w$, where $\mu_w$ is the linear attenuation coefficient of water. The scanning and reconstruction procedures are repeated with sequential shifting of the scanned body along the Z-axis. This produces a stack of 2D images that can be combined into a single 3D volume. This scanning principle (also known as circular CT) roughly describes how the first CT scanners worked.

Since the invention of CT in 1972, a series of scanner generations has been developed. Each generation added new features into the scanner architecture, improved the reconstruction method, the image quality and the speed of the scanning process. A number of modifications of the basic CT scanning approach were proposed [88, 185], including helical (spiral), multi-slice and ultrafast (cardiovascular). The majority of the modern CT scanners have an increased number of detectors (as well as X-ray tubes) and are able performing data acquisition using cone-beams and reconstruction images directly in three dimensions (circular or helical reconstruction). They are based on the 3D inverse Radon transform [79] that is commonly computed using some 3D modifications of the filtered backprojection algorithm [185].

The image quality of CT mostly depends on the scanner architecture, the properties of the X-ray sources and the detectors, as well as the features of the reconstruction algorithm. Modern clinical scanners can provide fine in-plane image resolution of less than 0.5 mm per pixel with the effective slice thickness of about 0.5 mm. On the other hand, CT often suffers from electronic noise (normally distributed), quantum noise (Poisson statistics) and a number of artefacts such as scatter, undersampling, streak, windmill, stairstep, motion etc. However, the main disadvantage of CT is concealed in extremely high radiation dose that is 10 to 100 times higher than in radiography and hence increases the radiation induced cancer risk. The relationship between image

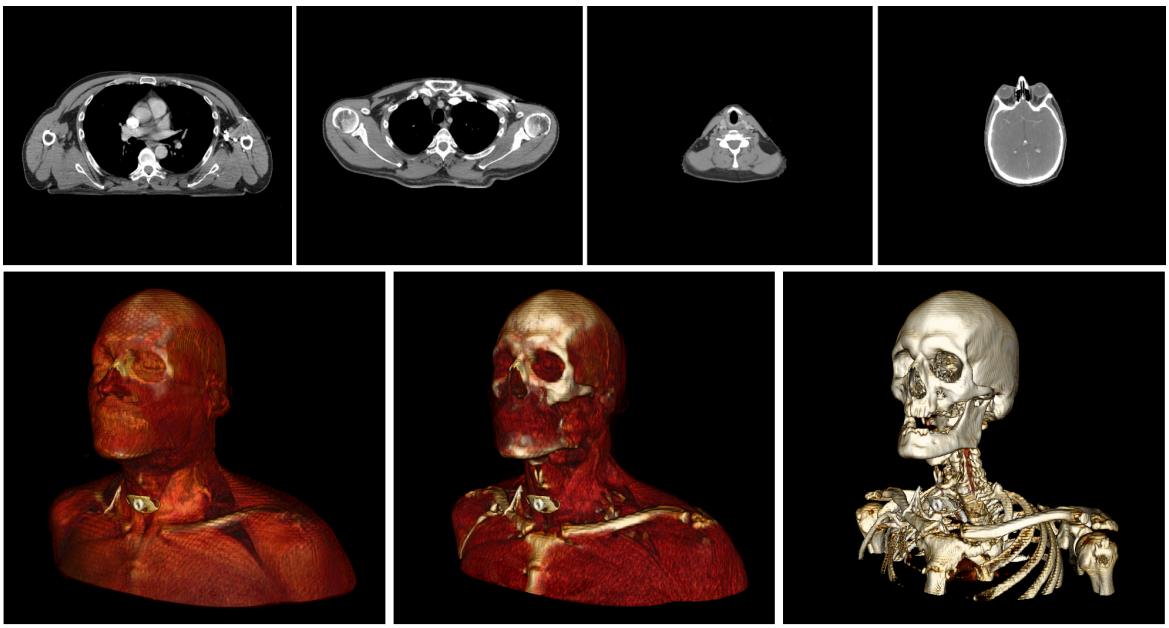

Figure 2.2.: Head-neck CT data consisted of 130 slices (512 x 512), TCIA free digital image database, National Cancer Institute (top). The corresponding 3D texture-based visualisation with different tissue transparency (bottom) using MITO, open-source medical imaging toolkit.

quality and dose $D$ can be approximated as $D \propto s^2/(e^3 b)$, where $s$ is the signal to noise ratio, $e$ is the image spatial resolution, and $b$ is the slice thickness [88].

In spite of this, CT is an important modality for organ visualization and is widely used in clinical practice for the variety of purposes including investigations of head, thorax, abdomen, urogenital tract and musculoskeleton system. Together with ECG synchronization CT is utilized in cardiovascular investigations for 4D (3D + time) heart image acquisition.

## 2.3. Magnetic Resonance Imaging

Magnetic resonance imaging (MRI) is a medical imaging technique that is based on the nuclear magnetic resonance (NMR) phenomenon of nuclei inside the body [88]. The NMR is explained within the theory of quantum electrodynamics as resonant absorption and re-emission of electromagnetic energy by atomic nuclei with nonzero spin in an external magnetic field at a specific resonance frequency. This phenomenon was first described and measured in molecular beams by Isidor Rabi in 1938 [160]. After this a number of attempts to apply NMR in the field of medical imaging were made by several groups of scientists [47]. However, the most successive approach was proposed in 1973 by Paul Lauterbur [121], who introduced gradients in the magnetic field. Several years later Peter Mansfield developed a mathematical technique that allowed this modality to produce fast and relatively clear cross-sectional images of the body.

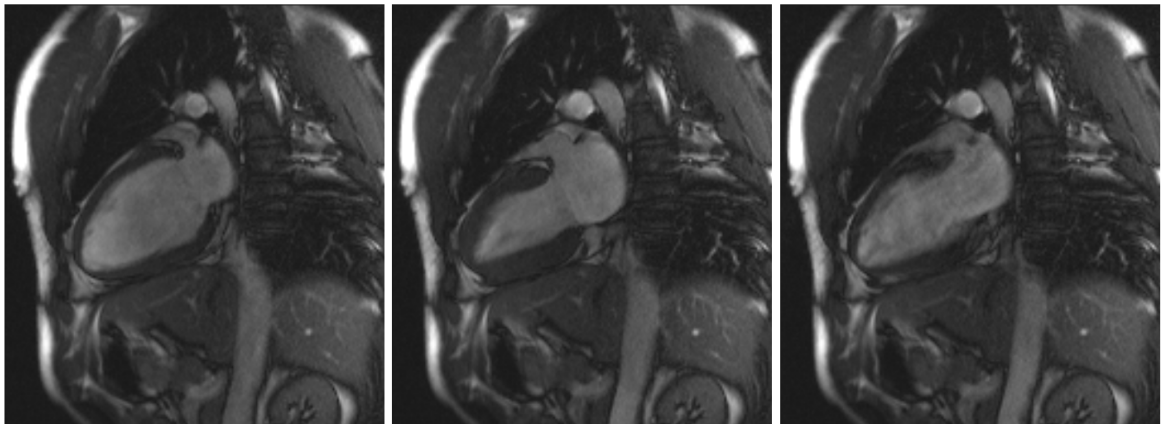

Figure 2.3.: Images of the heart cycle events (sagittal view) obtained from T2-weighted cardiac MRI (1.5 Tesla, 1.56 mm/pixel), University Medical Centre Mannheim, Institute for Clinical Radiology and Nuclear Medicine.

The quality of MRI depends on the physical properties of the scanner, the scanning procedure and the image reconstruction method. Modern MRI scanners can provide image resolution less than 1 mm per pixel. But it should be noted that MRI highly suffers from thermal noise in the scanned tissue and the noise in the electronic signal receivers. Several image artefacts are possible due to deviations of the main magnetic field, non-linear magnetic field gradients and discretization inaccuracies during image reconstruction (wrap-around, ringing) [88, 185].

Because of utilization of nonionizing radio waves, MRI is considered to be a safe medical imaging modality, that is the principle advantage of this technique. The main limitation of MRI consists in complete prohibition of ferromagnetic objects (including cardiac pacemakers and implants) in the MR examination room due to the strong static magnetic field.

In spite of extremely high costs of MRI equipment (US$1 - 2.5 million for a scanner), it is actively developed and applied in medical imaging of all parts of the human body that contain hydrogen (i.e. with exception of bones, lungs and gastrointestinal tract). Together with the convenient MRI a number of specialized MRI techniques (functional, diffusion, real-time MRI) have been invented since the last time. The current development of MRI is focusing on improvement of the image quality and reduction of the acquisition time by application of higher magnetic fields (up to 9.4 Tesla) and alternative reconstruction algorithms.

## 2.4. Nuclear medical imaging

Nuclear medical imaging is a medical imaging modality that is based on detection of photons emitted from a radiopharmaceutical (tracer) introduced into the patient body. This modality is mainly represented by two medical imaging techniques, single-photon emission computed tomography (SPECT) and positron emission tomography (PET).

In SPECT, tracer molecules carrying gamma-emitting radionuclides are injected into the bloodstream of a patient. Spreading with the blood in the body the tracer molecules

are involved in metabolic processes. The bound radionuclides emit $\gamma$-rays that can be detected by gamma-cameras [88, 185]. This allows estimating the concentration and location of the tracer molecule in the body and therefore allows measuring metabolism. First SPECT scanners were developed in early 1960s and have been incessantly improved till today. A common SPECT scanner receives multiple projections of the areas containing tracer molecules and reconstructs them into 3D images similarly to CT. Due to very small number of photons that are acquired by the gamma-cameras, the detecting system of SPECT must be much more sensitive than the one of CT, that results in relatively low signal to noise ratio and poor image quality of SPECT imaging.

A more efficient nuclear medical imaging technique is PET. It is based on detection of photon pairs produced in the body after positron-electron annihilation. Injected into the bloodstream tracer molecules carry radioisotopes that emit positrons. Each emitted positron travels through the tissue and interacts with an electron producing a pair of photons moving in opposite directions. This pair of photons can be detected by scintillation cameras that form a ring around the patient. So, each time two photons are detected by any pair of oppositely located cameras simultaneously (e.g., within $10^{-9}$ seconds) this event is registered and the corresponding tracer molecule is assumed to be located within a line segment between these cameras. The energy of the detected pair of photons is used to estimate linear attenuation of the tissue along the line. The registered lines that belong to the same slice form a sinogram, that can be reconstructed into a 2D image. The reconstruction is performed similarly with CT (although it is more complicated due to the low number of registered photons) using filtered back projection or Bayesian approaches (maximum-likelihood or maximum-a-posteriori) [185]. The reconstructed 2D images can be grouped slice-by-slice into 3D volumes. However, modern PET scanners allow reconstructing 3D images directly by detection of the pairs of photons both in the transverse plane and the oblique planes. Some improvement to PET has been recently introduced by so-called time-of-flight technique. It allows measuring time delay between detection of photons in a pair and therefore estimating the region of photon emission. This information is used to improve the image quality.

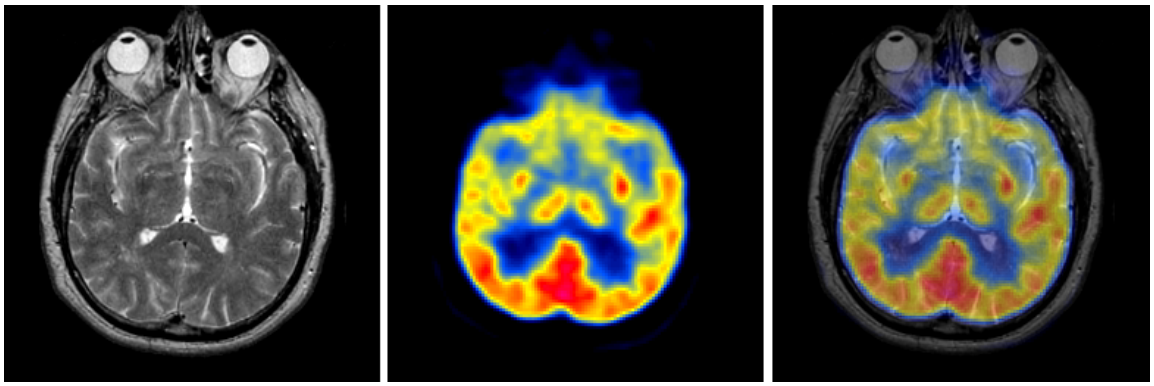

Figure 2.4.: Hybrid MRI-PET imaging of brain (The Whole Brain Atlas, Harvard Medical School). T2-weighted transaxial view of head (left), the corresponding PET using fluorodeoxyglucose tracer (center), the result of registration MRI-PET (right).

The image resolution of SPECT and PET is significantly poorer than in the other medical imaging modalities. In PET imaging two points are distinguished if the spatial distance between them is not less than 5 mm. For SPECT this value is about 1 cm. Image quality of PET and SPECT depends on the number of detectors, their resolution, and the properties of the utilized tracer. Nuclear medical imaging significantly suffers from noise and artefacts caused by false attenuation correction and scatter. Another problem of this modality consists in handling and high costs of radiopharmaceutical that normally have a short half-life and with a few exceptions must be produced by a cyclotron in the hospital.

Nuclear imaging involves significant exposure to ionizing radiation that is comparable to CT and can seriously harm patients. Furthermore, the injected radiopharmaceutical remains in the body for hours or even days after the clinical examination is finished. This is a limiting factor for all-round application of this modality in clinical practice. In spite of these disadvantages, nuclear imaging remains the most accessible approach to measure metabolic processes in the patient body. Nowadays it is the only practical alternative in studies of bone metabolism, myocardial perfusion and viability, lung embolism, tumors, and thyroid function [185]. As an alternative to more modern and expensive MRI, PET can be utilized for functional neuroimaging (Brain PET, Figure 2.4). Together with this nuclear imaging is recently used as a part of so-called hybrid systems, when PET and SPECT are combined with CT or MRI in one imaging system for complex investigation of tissue structure and metabolic processed inside it.

## 2.5. Ultrasound

Ultrasound imaging (echography) is based on interactions of propagating longitudinal compression waves (normally in MHz ultrasonic frequency range). The ultrasonic waves are typically generated by piezoelectric crystals that deform under the influence of an electric field. The deformation of the piezoelectric crystals causes variable acoustic pressure (regions of high and low particle density) in the medium that propagates as a wave parallel to the deformation vector. After a wave generation the piezoelectric crystal switches into the receiver mode and transforms the mechanical energy of the reflected waves back into an electric signal. Considering this property (conversion of one form of energy into another form) the devices based on piezoelectric crystals are called transducers. The received signals are measured as a function of time, that makes it possible to calculate the distance from the transducer to the reflecting or scattering objects if the the velocity of the ultrasonic wave in the medium is known.

### 2.5.1. Conventional 2D ultrasound

Ultrasound imaging was first developed in early 1900 for military and industrial applications after the work of Pierre Curie (1880) and Lord Rayleigh (1877) [166]. In the medical field it was first applied in 1942 [59] and was intensively developed during the next two decades. During this period such techniques as 2D gray scale and Doppler ultrasound appeared. The first 2D gray scale real-time ultrasound imaging was developed in 1965 by Siemens. Ultrasound imaging was steadily improved during the next three

decades and has become one of the most widely used medical image modalities.

From the point of view of ultrasound imaging each homogeneous tissue can be characterized by its acoustic impedance $Z$ that is the ratio of the acoustic pressure $p$ to the particle velocity response $v$. From another hand, it can be shown that $Z$ depends on the mass density $\rho$ and acoustic wave velocity $c$ in the medium: $Z = \rho c$, [185]. If the acoustic pressure of the wave is small enough, the wave propagation in a homogeneous media is described by the linear wave equation:

$$\nabla^2 p = \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2}. \tag{2.2}$$

The wave propagation in a homogeneous tissue described by (2.2) is accompanied by such physical phenomena as wave *diffraction*, *attenuation*, *reflection* and *refraction* [88].

Wave diffraction is caused by complex interference of multiple coherent waves originating from different sources and can be constructive (the waves amplify) or destructive (the waves vanish). Diffraction of ultrasonic waves produces a complex interference pattern that is defined by the properties of tissue, the amplitude, phase and frequency of the waves; and their spatial origin.

Attenuation is the decrease of the acoustic energy of the wave (wave amplitude) during propagation. It depends on the wave frequency and is described by the exponential law [185]:

$$H(f, x) = e^{-\alpha_0 f x}, \tag{2.3}$$

where $f$ is the frequency, $x$ is the distance propagated through the tissue and $\alpha_0$ is the attenuation constant of the particular tissue. From this equation follows that high-frequency waves are more liable to attenuation that the low-frequency ones.

Wave reflection and refraction appear when an acoustic wave crosses the interface between tissues with different densities and sound velocities. In this case a part of the acoustic energy is reflected from the interface while another part is transmitted further. This process can be described by the Snell's law for pure specular reflections:

$$\frac{\sin\theta_i}{c_1} = \frac{\sin\theta_r}{c_1} = \frac{\sin\theta_t}{c_2}, \tag{2.4}$$

where $\theta_i$, $\theta_r$ and $\theta_t$ are the angles of incidence, reflection, and transmission (refraction) correspondingly. Values $c_1$ and $c_2$ stand for the sound velocities in the incoming and the outgoing tissues respectively. The frequency of the refracted and reflected waves stay the same as the incidence one, but their amplitudes are changed accordingly to the transmission ($T$) and reflection ($R$) coefficients [185]:

$$T = \frac{A_t}{A_i} = \frac{2Z_2\cos\theta_i}{Z_2\cos\theta_i + Z_1\cos\theta_t}, \quad R = \frac{A_r}{A_i} = \frac{Z_2\cos\theta_i - Z_1\cos\theta_t}{Z_2\cos\theta_i + Z_1\cos\theta_t},$$

where $A_i$, $A_r$, and $A_t$ are the incident, reflected and transmitted amplitudes and $Z_1$ and $Z_2$ are the acoustic impedances of the tissues.

With increase of wave amplitude the linear propagation law (2.2) is no longer valid. In this case such physical phenomena as *distortion* appears. Distortion is associated with change of the waveform during propagation and generation of higher harmonics

of the original frequency. The effect of non-linear distortion increases with propagation distance.

The biological tissues are inhomogeneous, therefore the pure specular reflections described by (2.4) are contributed by scatter reflections. Such reflections can be modelled by a set of small inhomogeneities in the tissue that retransmit the incident wave in all possible directions (Huygens principle). This results in backscatter (waves directed back to the transducer) and wave diffraction. The backscatter in ultrasound imaging has a complex nature that can be described by two principle components: *Rayleigh- and Mie scattering*. Rayleigh scattering is produced by the tissue inhomogeneities that are much smaller than the wavelength, while the Mie scattering is caused by particles with the sizes comparable to the wavelength. Intensity of Rayleigh scattering is proportional to the fourth degree of the wave frequency and is identical in the forward and reverse directions, while the intensity of Mie scattering is roughly independent of the wave frequency and is larger in the forward direction. In both cases backscatter can be considered as a special type of noise that provides additional information about the structure of the tissue that is often used in analysis of ultrasound images.

Modern ultrasound scanners are able to reconstruct detected ultrasonic signals using two approaches, gray scale imaging and Doppler imaging. Gray scale imaging is represented by A-mode, B-mode, C-mode and M-mode.

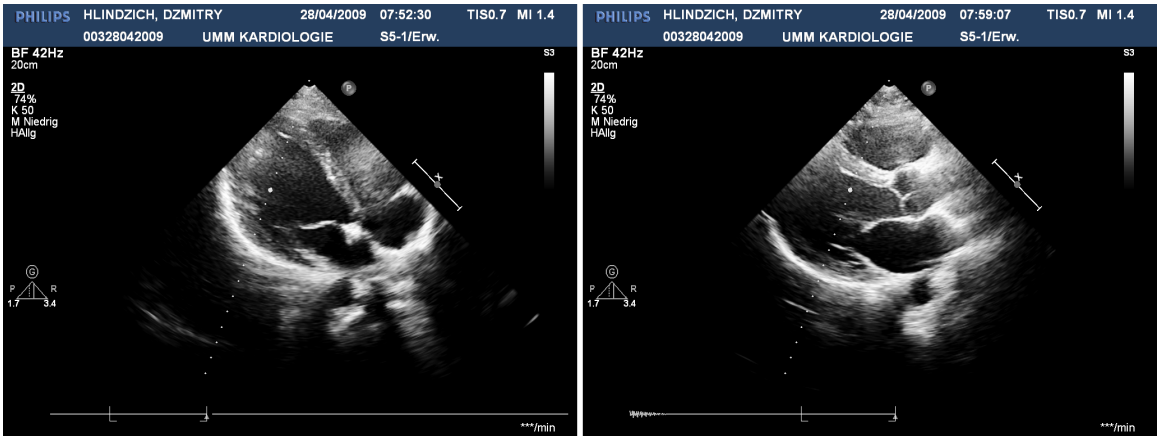


Figure 2.5.: B-mode ultrasound images of the heart, apical four-chamber view (left) and parasternal long axis view with the mitral and aortic valves (right). University Medical Centre Mannheim, Department of Medicine I, Cardiology.

*A-mode* (amplitude) is the basic scanning mode that utilizes the principle described above. After the pulse transmission the transducer is used as a receiver and records the specular and scattered reflections as a function of time. Assuming that sound velocity in the tissue is known, the time can be translated into the distance and the structure of the reflectors can be visualised along the wave propagation line using the amplitudes of the received signal.

In *B-mode* (brightness) imaging the A-mode is repeated by translating or tilting the transmitter and the receiver in space. This produces a 2D gray scale image of the tissue defined by the used transmitter movement (Figure 2.5). Modern ultrasound transducers contain an array of piezoelectric crystals that allows moving the ultrasonic

beam by electronically switching the crystals sequentially (linear-array transducers) or in a phased manner (phased-array transducers).

In *M-mode* (movement) imaging the A-mode is repeated for a static transducer and a moving object. As the result a 2D image of the object movement within the scanned line can be obtained.

The *C-mode* allows obtaining images of planes parallel to the surface of the transducer. It can be achieved by rotating or wobbling a transducer in B-mode to acquire images sequentially from different scan planes. In order to avoid mechanical movement of the transducer, electronic switching can be applied by utilization of 2D phased arrays (see 3D ultrasound, Section 2.5.2).

A series of standard steps is used for image reconstruction in all modes of gray scaled imaging. These steps include filtering, envelope detection, attenuation correction, log-compression, and scan conversion [185].

Doppler imaging allows visualizing velocities of moving tissues. This technique is represented by three modes: continuous wave (CW), pulsed wave (PW), and color flow (CF) Doppler (Figure 2.6, 2.7). For tissue velocity visualization CW Doppler uses the Doppler effect, while PW and CF Doppler imaging are based on pulse-echo principle and calculate the velocity using time delay or phase shift between subsequent received pulses. Considering image dimensionality PW Doppler is equivalent to the M-mode, while CF Doppler corresponds to the B-mode. Due to the higher number of ultrasonic pulses needed for PW and CF Doppler, these techniques are commonly several times slower than the gray scaled imaging.

Spatial resolution of echography is relatively high and depends on the used wave frequency. Due to its anisotropic property, the spatial resolution consists of the axial, lateral and elevation terms. The axial resolution is commonly better than the lateral and the elevation ones. The lateral and the elevation resolutions highly depend on the size, the shape and focusing of a transducer. The range of wave frequencies used in medical ultrasound (commonly between 1 and 20 MHz) provides the axial resolution of up to 0.1 - 0.5 mm per pixel, that is often better than the resolution of the other medical imaging modalities except radiography. On the one side, the range of the used ultrasonic frequencies depends on the resonance frequency of a transducer, that, in turn, is defined by the thickness of the piezoelectric crystals. On the other side, it is a trade-off between the spatial resolution and the strength of the wave attenuation effect (2.3) that results in the maximal imaging depth (field of view, FOV).

Temporal resolution of ultrasound imaging is among the highest of medical image modalities. It is mainly defined by the number of ultrasonic pulses used to acquire one image and the FOV. In dependence with the selected imaging mode echography can provide tens of images per second (normally 10 - 60 images/sec). Unfortunately, ultrasound imaging suffers from a number of specific artefacts such as side lobes caused by imperfect shape of the ultrasound beam, and reverberations associated with recurring reflections of the waves from the transducer. Acoustic shadowing and enhancement, multiple pathway and refraction artefacts [88] can also significantly reduce the image quality. Due to scatter reflections ultrasound images often look more noisy than images from the other modalities. However, unlike many other medical imaging modalities, the scatter noise here carries additional information about the tissue structure.

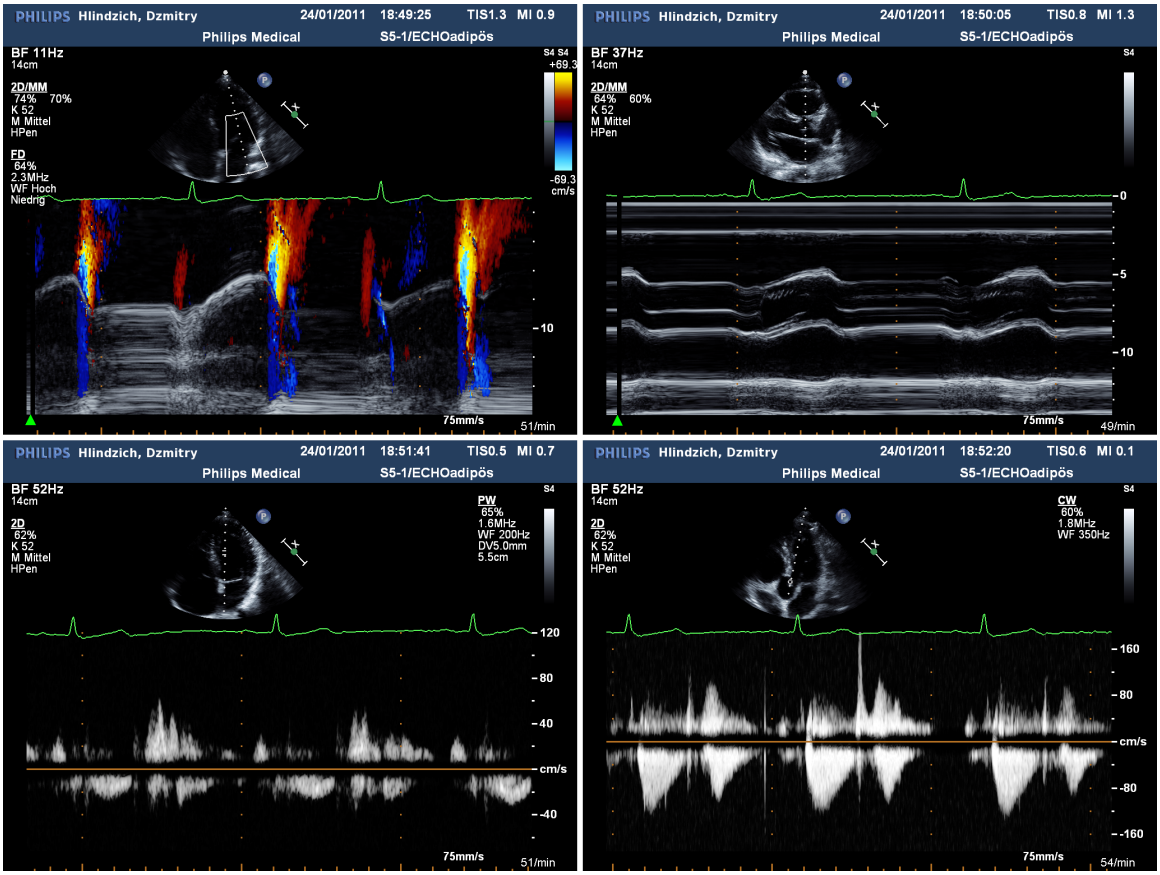


Figure 2.6.: Doppler ultrasound images of the heart. From left to right: M-mode + Doppler and M-mode (top); PW and CW Doppler (bottom). University Medical Centre Mannheim, Department of Medicine I, Cardiology.

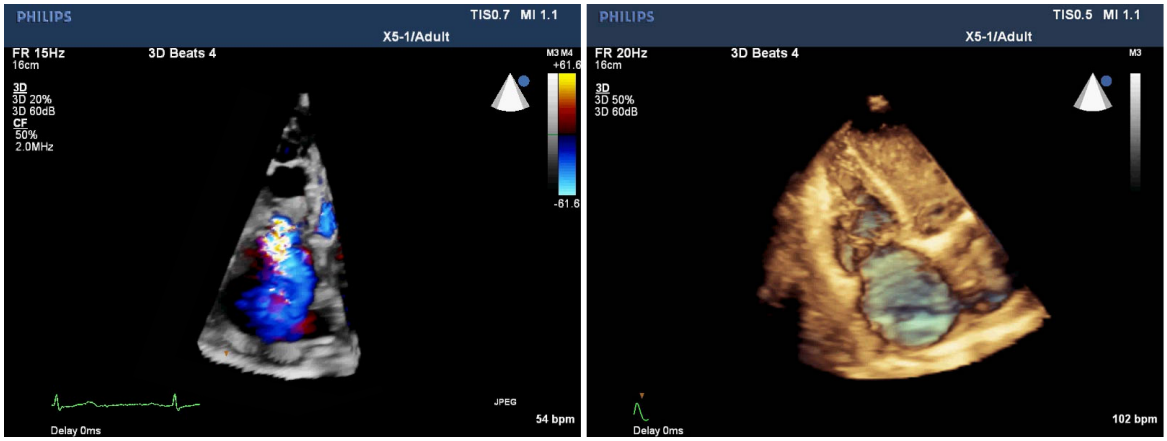


Figure 2.7.: 3D cardiac ultrasound images acquired using the novel Philips iE33 xMATRIX Echocardiography System. Apical view of the mitral valve, 3D + CF Doppler (left); right ventrical and tricuspid valve (right). The images are taken from http://www.healthcare.philips.com.

Ultrasound imaging is a safe and relatively cheap imaging modality. Moreover, it is highly transportable and does not require any special infrastructure. The only side-effects that can potentially harm patient are associated with tissue heating and cavitation. But, fortunately, these effects can be fully controlled by the settings of an ultrasound scanning system. Due to the considered advantages, nowadays echography is one of the most popular medical image modalities. It is applied in a wide range of investigations of almost all parts of the body.

### 2.5.2. 3D ultrasound

The technique, based on 2D phased array transducers, is also known as 3D ultrasound for its ability to reconstruct 3D volumes of the scanned area. It was first described already in 1956 [98], but active development in this field started only after the patent of von Ramm and Smith in 1987 [196]. The first commercial 3D ultrasound machine was produced in 1989 by Kretztechnik AG (Zipf, Austria) [166]. Unfortunately, due to the large number of piezoelectric elements 2D phased arrays transducers are still quite bulky and the image reconstruction time is relatively big. In spite of this, commercial 3D ultrasound imaging system have become recently available. Today they are mostly used for 3D/4D (3D + time) visualization of fetus and the heart [100] (Figure 2.7).

3D ultrasound imaging systems based on 2D phased arrays are often called *fast acquisition systems*, because they are able to scan a selected 3D local region of an object "at once" in a very short period of time [197]. Along with 2D phased arrays the fast acquisition systems can utilize fast rotating 1D phased array transducers [54]. Such systems are the most convenient solution for fast 3D imaging of local object regions (such as the cardiac valves), though in many cases they are not able to cover the whole organ of an investigated patient.

The most recent research in the field of 3D ultrasound imaging is related to development of sequential triggered and freehand *tracked 3D ultrasound* systems (see Section 3.1). The main idea of tracked 3D ultrasound consists in utilization of position tracker devices for estimation of the transducer's spatial location and orientation during scanning. Using the known pose of the transducer in the global coordinate space, all received ultrasound data can be reconstructed into a single 3D volume that covers the whole scanning area. Position tracking can be used both with 2D and 3D ultrasound for extended 3D imaging, improved data representation and navigation. First the principle of tracked 3D ultrasound was described in early 1980s [28], however development of real-time tracked 3D ultrasound systems is actual until now.

Along with tracked 3D ultrasound, several authors are working on *sensorless freehand 3D ultrasound* methods [72, 73, 97]. Here the relative positions of slices in 3D are not measured by a tracking system but are defined using some features within the images themselves or by the assumption that the transducer is moved along a straight path with constant orientation and speed. In spite of the fact that this technique cannot yet provide accurate image reconstruction and requires high operator skills, it has been implemented in some commercial ultrasound systems (e.g. Siemens ACUSON X300, Philips CX50).

Since the last two decades the field of research in ultrasound medical imaging has been gradually shifting from 2D to 3D [63]. Often in literature 3D ultrasound is related

to all ultrasound imaging techniques that are able to acquire 3D images (not only those ones based on 2D phased array transducers, but also tracked and sensorless freehand ultrasound and all fast acquisition systems). 3D ultrasound have shown a number of advantages over conventional 2D ultrasound [166]. It provides more natural representation of surfaces of biological structures (that is actively used today e.g. in fetal examinations) and allows more accurate measurement of organ volume. 3D ultrasound imaging is able to visualize objects in 2D cross-sectional planes that are physically inaccessible with the conventional 2D techniques (e.g. parallel to skin), to provide more data for accurate diagnostics and geometry reconstruction of patients organs [189]. Significant contribution of 3D ultrasound has been introduced to echocardiographic examinations (Figure 2.7), allowing direct assessment of ventricular volume and mass, examination of specific cardiac abnormalities, myocardial performance and contraction [100, 201].

Despite all the listed above advantages, such problems as poor image quality and insufficient spatial coverage are still acute for 3D ultrasound imaging systems and require further investigation. The latest reports in this field show significant improvement of real-time imaging quality resulting from application of parallel computing (e.g. replacing existing FPGA- and CPU-based processing functions by CUDA in new Siemens ACUSON SC2000). Today, the opportunity to extend the spacial coverage of echography by developing mechanical and freehand tracked 3D ultrasound systems excites a big interest for future investigations. In the next chapter we will discuss tracked 3D ultrasound and introduce two experimental imaging systems based on this idea.

# 3. Two prototypes of tracked 3D ultrasound scanning systems

## 3.1. State of the art

Tracked 3D ultrasound is an imaging technique based on the advantages of both the convenient ultrasound imaging and tracking (positioning) systems. Utilization of the known spatial position of the ultrasound transducer allows treating the acquired ultrasound images as interrelated slices similarly to the basic principles of tomography and, therefore, to reconstruct them into a single 3D volume or (after segmentation) into a point cloud.

The idea of tracking the position of a transducer and gathering 2D ultrasound images into a single 3D volume is far from novel. The first experimental systems based on this principle were proposed in late 1970s - 1980s and used magnetic [159], acoustic [28], mechanical [182], optical [24, 143], inertial motion (accelerometer) tracking systems or their combinations [77]. However, these systems were quite complicated, slow and did not provide high reconstruction accuracy. During the last decade a lot of publications and research work was dedicated to the problem of engineering and improvement of the systems for tracked 3D/4D ultrasound data acquisition, interpretation and analysis [166, 19, 190, 191, 77, 46, 158, 186]. Along with the engineering problems these developments considered such questions as volumetric reconstruction of inner tissue structures, volume interpolation, device calibration and parallel acceleration. In spite of the fact that *volumetric* reconstruction is becoming increasingly popular today, visualization of anatomical *surfaces* has an important role in orthopaedics (diagnosis of bone fractures), cardiac and abdominal diagnostics. Reasoning from this, here we concern the problems of development and evaluation of tracked 3D ultrasound imaging systems for *3D reconstruction of surfaces* of anatomical objects and the corresponding medical image analysis.

Similarly to tomography, tracked 3D ultrasound needs an explicit step that transforms ultrasound data from a set of 2D images into a 3D space. A number of notable algorithms has been proposed for solution of this problem [181, 107, 158]. Nevertheless, the main attention there is focused on the problem of reconstruction of ultrasound data into a regular grid, that can be represented similarly to CT or MRI as a 3D volume. Such an approach is a suited solution for reconstruction of dense 3D ultrasound data where the inner structure of scanned organs and the corresponding intensities of voxels are important. However, in the case of surface reconstruction, processing of large 3D volumes can be successfully avoided. The ultrasonic data acquired for 3D surface reconstruction is presented by an irregular set of binary contours and therefore is relatively sparse. This allows using the point cloud model directly and avoiding additional

memory consumption and computations needed for regular grid reconstruction and interpolation. Furthermore, the increasing power of modern computers along with the highly-optimized programming libraries [162, 168] are making it possible to operate with point clouds efficiently.

Nowadays, tracked 3D ultrasound imaging is presented by two principle branches: **freehand-** [189] and **sequential triggered** [26] scanning.

Freehand scanning systems allow an operator to move the transducer manually in 3D space, while its coordinates are continuously computed by a tracking device. This makes freehand ultrasound a convenient and a powerful tool for 3D medical imaging. Generally, the freehand ultrasound scanning systems can be equipped with various position trackers that provide different accuracy [63, 92] and are based on one of the acoustical [28], electromagnetic [19, 159], electromechanical [74], optical [190] or fibre optical [48] principles. The optical and electromechanical position trackers (electromechanical arms) are of particular interest here as they provide the best cost-accuracy relation. In spite of the high accuracy and flexibility, optical tracking in some cases can be inconvenient for clinical usage as it requires a direct optical contact with the tracked tool (that is not always possible). From this point of view the electromechanical position trackers are more preferable.

One of the first freehand scanning systems equipped with an electromechanical arm was developed in 1982 by Geiser et. al. [74] and was used for calculation of left ventricular wall motion. The system was clinically tested and provided the accuracy of positioning a cross-section of the long axis of 6 mm. This result was promising and corresponded to the state-of-the art ultrasound and tracking technologies. However, for the past thirty years ultrasound imaging and positioning devices have been continuously developed and improved. Therefore, estimating the accuracy of a modern freehand 3D ultrasound system based on this principle is particularly interesting.

Sequential triggered scanning systems, in turn, make use of stepper motors for programmable 3D-positioning of the transducer and its coordinates calculation. They can be considered as fast and automated but spatially-restricted analogue of the freehand scanning systems. Today the sequential triggered scanning principle is most successfully presented in clinical mammography (automated breast volume scanners developed by Siemens and General Electric) for 3D volume reconstruction. These scanners are able to provide high-resolution ultrasound volumes up to 15cm x 17cm x 5.0cm. Unfortunately, similar systems are still not widely applied in orthopaedics, traumatology and cardiology because of their complexity, need for high experienced medical staff, still low level of automation and, as the result, prolongation of the diagnostic and surgical time. So, the problem of developing an effective automated ultrasound imaging system in the mentioned clinical branches is quite acute today.

Regardless of some specific features all tracked 3D ultrasound imaging systems consist of the common principle functional modules (Figure 3.1): transducer, ultrasound machine, tracking device and workstation.

The signal received from the transducer is reconstructed by the ultrasound machine into 2D(3D)-scans and is sent to the workstation. At the same time the workstation reads the information about the transducer position and orientation from the tracking device and realigns the acquired scans in 3D space correspondingly to the actual position of the transducer. Finally, the transformed ultrasound data is reconstructed into 3D volumes

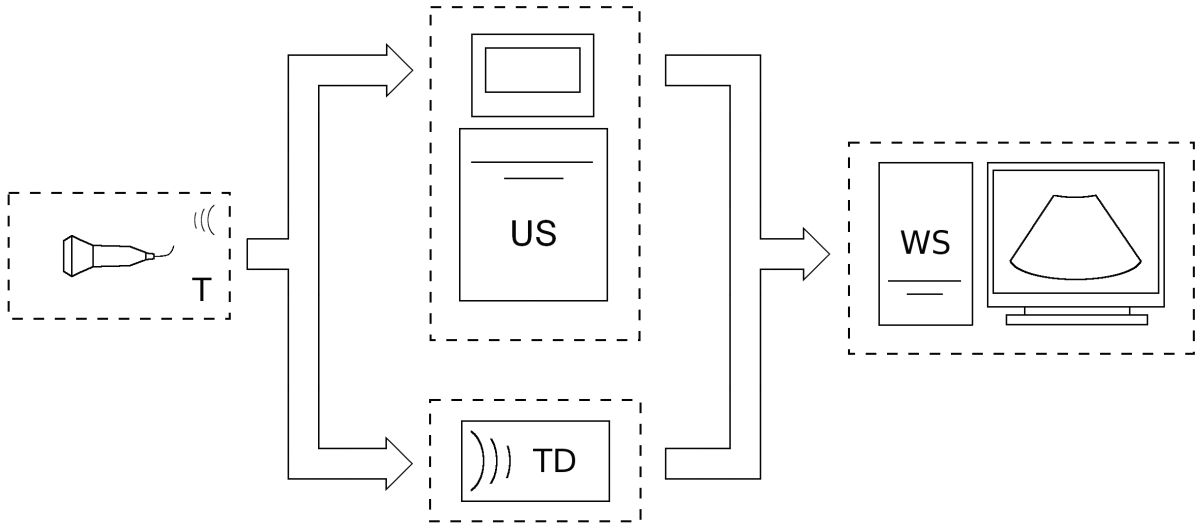


Figure 3.1.: Principle functional modules of a tracked 3D ultrasound imaging system (T - transducer, US - ultrasound machine, TD - tracking device, WS - workstation).

or surfaces and visualized. Additionally, the functional modules can be extended with a time-synchronization module (usually based on a ECG-device) for 4D image acquisition, and a server for data storage or communication with a PACS.

The choice between the freehand and the sequential triggered scanning often depends on its application. For example, compact tissue areas with relatively simple surface geometry and easy access (e.g. the long bones of the extremities) can be scanned automatically by use of sequential triggered systems. In the case of a more complicated tissue shape and difficult to access locations (e.g. inner organs of abdomen and thorax, neck) automated systems can be quite complicated and it is often easier to use freehand tracked 3D ultrasound. In the following sections we consider each of these two approaches, develop two experimental scanning systems based on the freehand- and the sequential triggered principles, evaluate the accuracy and discuss their possible practical applications.

## 3.2. Motivation and objectives

As it was mentioned in the previous chapter, all existent medical imaging modalities have a number of specific advantages as well as drawbacks. One of the most significant drawback is related to the harm to the patient health appearing during the scanning procedure. Such modalities as radiography, CT and nuclear medical imaging involve considerable exposure to ionizing radiation, that produces a risk of damage to DNA and subsequently can cause cancer. However, there are two modalities, MRI and ultrasound, that are considered to be relatively safe for patients. MRI is an excellent medical imaging technique providing 3D/4D images of high quality and resolution, but unfortunately today it suffers from extremely high costs for equipment, infrastructure and maintenance. Hospital departments have to spend millions of dollars for an average MRI scanner and a

specially equipped room, isolated from a strong magnetic field, external radio frequency signals and supporting safe utilization of superconducting electromagnets.

From this point of view ultrasound imaging is obviously a much more preferable technique, as it is tens (sometimes even hundreds) times cheaper, does not require special infrastructure and is easy in maintenance. However, the disadvantage of ultrasound imaging consists in its comparatively low image quality (though the spatial resolution is commonly high) caused by a number of specific acoustic artefacts and the nature of ultrasound itself, that is able to visualize only the scatter density and the tissue interfaces reflecting ultrasonic waves. 3D ultrasound, that has recently appeared in commercial scanners, can cover rather restricted 3D volumes (such as parts of organs) and cannot be applied for the whole body visualization. That is why the solutions that improve the imaging quality and extend the possibilities of the conventional ultrasound to acquire 3D/4D images of large spatial regions of the body are essential today. And one of such solutions is represented by the tracked 3D ultrasound.

Considering orthopaedics and trauma surgery, application of conventional projective radiographs, CT and MRI is also frequently restricted on practice. Namely, CT provides excellent 3D image quality but it's usage is limited because of the high radioactive dose. X-ray images successfully provide information about axial deformities of bones. But owing to their projective nature, measurements of lengths and detection of small fractures are often considered as inaccurate, while measurements of torsion angles are practically impossible. Through the fact that human bones contain relatively small amount of hydrogen atoms, MRI does not provide good representation of bone structures. 3D tracked ultrasound, in turn, can be used for diagnostics of bone fractures and deformities as well. It is able to provide riskless, non-invasive and very compact tools for trauma surgeons.

Our objectives in this chapter are to develop two prototypes of the scanning systems based on the tracked 3D ultrasound for reconstruction of anatomical surfaces of a) viscera and b) the long bones of the extremities; to measure their accuracy and to compare it with the accuracy of the existing medical imaging modalities. The developed systems must satisfy the following requirements:

- non-invasiveness,
- easy in implementation,
- compact size,
- low costs.

## 3.3. A freehand tracked 3D/4D ultrasound imaging system for tissue surface reconstruction

The developed scanning system is based on the conventional 2D ultrasound (B-mode) and is equipped with an universal electromechanical tracker (electromechanical arm) for digitizing 3D coordinates of the ultrasonic transducer. We consider the architecture of the system and its calibration using a plane phantom, carry out basic validation of the setup on a porcine heart specimen and discuss its basic functionality and features.

## 3.3.1. Material and methods

### System architecture

The system consists of the standard functional blocks (Figure 3.1) including a 2D ultrasound machine with several transducers, a tracking system, a workstation and (optionally) an ECG device for temporal synchronization (Figure 3.2).

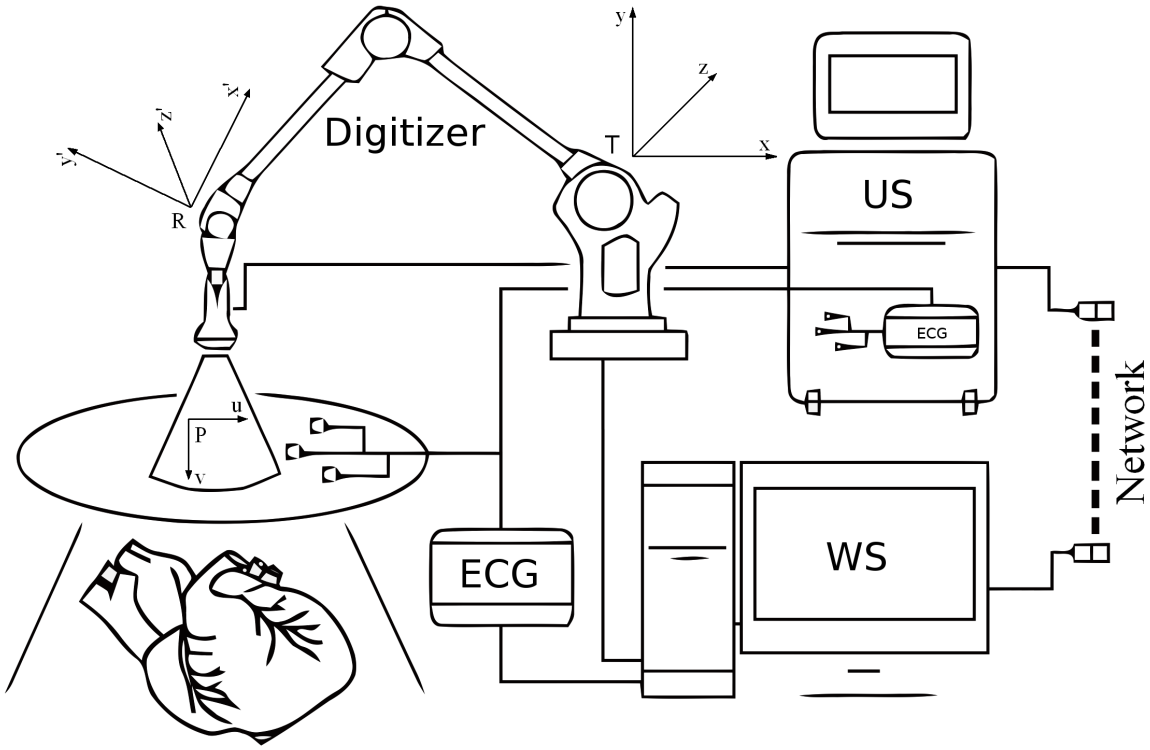


Figure 3.2.: Principle functional modules of the tracked 3D/4D ultrasound imaging system based on *Immersion MicroScribe* digitizer.

The ultrasound data are acquired using a linear transducer (linear array HL7 5-8 MHz 40 mm, convex array C3.5 2-5 MHz 60 mm or phased array 1.5-3 MHz). Position tracking is performed using the electromechanical arm *Immersion MicroScribe*.

The decision to use an electromechanical arm was made after comparison of this approach with several alternative techniques. We tested the system with an optical stereo-camera and a multiple-camera tracking device. Other types of trackers have not been considered due to either lower measurement accuracy or difficulties with their application within a portable system.

First experimental setups started with an optical position sensor *NDI Polaris Vicra* with the measurement accuracy of 0.35 mm root mean square (RMS). The ultrasound transducer was marked with an optical tool, consisted of 4-5 light reflexive spheres of 8 mm diameter forming an unique geometric pattern. During scanning the transducer with the optical tool was tracked by the position sensor and the coordinates of the transducer were reconstructed at the workstation. However, a number of clinical experiments with cardiac ultrasound demonstrated a significant drawback of this approach. The position

sensor required a direct optical contact with the optical tool fixed on the transducer. Keeping this condition during data acquisition was problematic due to the special clinical requirements to the positions of a patient and the locations of ultrasonic acquisition windows. Frequently the optical contact was broken by physicians. As the result, the scanning process became rather time-consuming and required special attention from the side of medical personnel.

Therefore, an alternative solution was proposed. The optical system was replaced by the electromechanical arm (digitizer) *Immersion MicroScribe* with the measurement accuracy of 0.21 mm RMS (Figure 3.2). A series of experiments showed that this approach is more robust to incautions during operation and introduces less inconvenience to the scanning procedure. Providing six degrees of freedom (6-DOF) the digitizer with the fixed transducer allow scanning patients from various directions within the operative volume of radius 0.5 m without changing the initial setup of the system. The only drawback of this approach consists in the requirement that the digitizer must be located at the same place during the whole acquisition procedure. Otherwise a rigid reference marker must be used to link the coordinate systems after each reposition of the device. However, in this case the operative volume can be considerably extended without serious modifications of the tracking system.

The workstation performs synchronization, image reconstruction and visualization tasks. Synchronization consists of linking of the ultrasound scans received from the ultrasound machine with the corresponding transducer coordinates obtained from the tracking device. The system can also use temporal information provided by an ECG device, that allows reconstructing 3D images of moving tissues. In this case the workstation performs temporal landmarking and rearrange the acquired scans in separate data stacks accordingly to their time events. If the ECG device is integrated into the ultrasound machine (that is common for commercial cardiac ultrasound scanners), temporal landmarking is performed by the ultrasound machine. In our clinical experiments we used *Philips iE33* echocardiography scanner, while in the labour a portable 2D ultrasound machine based on hardware from *Medison Transducers* and an external ECG module *EMI12* from *Corscience GmbH* were used.

Image reconstruction and visualization can be done in real-time or offline. Though the system has a big potential to perform these tasks in real-time, only offline reconstruction and visualization was implemented in this prototype.

### Calibration

The goal of calibration is to compute the unknown transformation from the coordinate space of 2D ultrasound images into the coordinate space of the digitizer tip. A comprehensive review of existing calibration techniques used for freehand 3D ultrasound calibration is provided in [99].

The idea of the used calibration method is based on combining a plane object with a cross-wire phantom (so-called plane phantom, Figure 3.3) [57]. The phantom geometry has been measured on *Siemens Emotion* CT-scanner with the resolution of $0.34 \times 0.34 \times 0.5$ mm / voxel. During the calibration procedure the phantom has been placed in a water basin with a water temperature about 35 degree Celsius. The temperature has been selected to provide the acoustic velocity compared with the mean speed of

ultrasound waves in the human body.

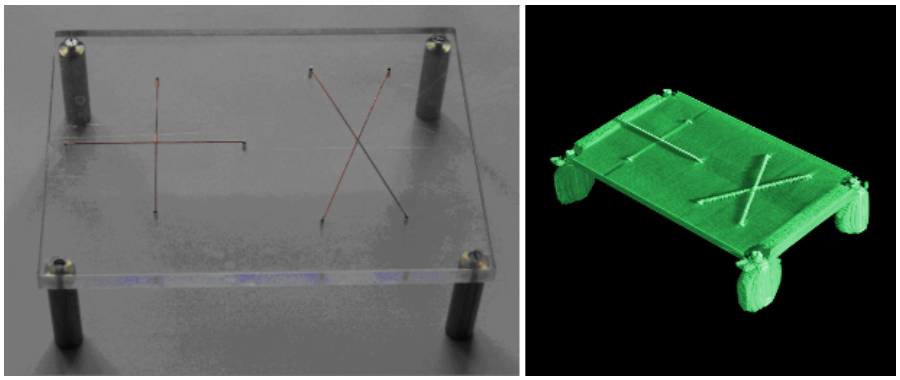

Figure 3.3.: The plane phantom used for calibration. Original view (left) and CT reconstruction (right).

The common practice for modelling the US calibration is to use four coordinate systems (see Figure 3.2): the coordinate system of a B-scan ($P$), the coordinate system ($R$) of the digitizer tip (or the optical tool in the case of optical tracking systems) tracked by the positioning device, the coordinate system of the tracking device ($T$) and the phantom coordinate system ($C$). The transformation from the system $P$ to the system $C$ is describes as:

$$M = T_T^C \cdot T_R^T \cdot T_P^R \cdot P, \tag{3.1}$$

where $T_X^Y$ denotes a transformation matrix from a coordinate system $X$ to a coordinate system $Y$, $M = (x, y, z, 1)^T$, $P = (S_x u, S_y v, 0, 1)^T$, $(x, y, z)$ are the coordinates in $C$ and $(u, v)$ are the coordinates on a B-scan. The values $Sx$ and $Sy$ are the scale factors of the US image along OX and OY axes respectively. Using these definition we can formulate the goal of calibration as to find the unknown transformation $T_P^R$ from the coordinate space $P$ to $R$. The transformation $T_R^T$ is provided by the tracking device for each position of the transducer, the matrix $T_T^C$ is acquired one time during the calibration of the tracking device and is known.

During calibration the plane phantom is scanned from different directions. For each scan we obtain three equations that correspond to the first 3 rows in the relation (3.1). After acquisition of $m$ scans (in our experiments we took $m \geq 10$) and detection of the wire cross coordinates $P_i$, $i = \overline{1, m}$ (automatic [57] or manual), the required transformation matrix $T_P^R$ is computed using the *Levenberg-Marquardt algorithm* [127] for the least squares curve fitting problem. The resulting calibration accuracy (Table 3.1) is calculated as the root mean square (RMS) of the residuals $Y_i = ||F_C - M_i||$:

$$Y_{RMS} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} Y_i^2}, \tag{3.2}$$

where the relative coordinates $F_C$ of the wire cross are known from the CT model of the phantom.

| Transducer | Frequency range, MHz | Accuracy RMS, mm |
|---|---|---|
| Linear array HL7 | 5-8 | 0.25 |
| Phased array Phillips iE33 | 1.5-3 | 0.5 |
| Convex array C3.5 | 2-5 | 0.65 |

Table 3.1.: System calibration accuracy using different types of ultrasound transducers.

### Data acquisition

After the calibration procedure the system is ready for data acquisition and reconstruction. For 3D reconstruction the ultrasound data must be synchronized with the position tracking device.

In our clinical experiments (Figure 3.4, 3.5, 3.6) we utilized manual synchronization consisted in simultaneous acquisition of a B-scan and the corresponding transducer coordinates by a signal generated manually by an assistant of the operating physician. The acquired B-scans were stored in a clinical network, while the corresponding coordinates were transformed into matrices $T_R^T$ and written directly into the database at the workstation. After the scanning procedure the ultrasound data were downloaded onto the workstation, merged with the position data and is then ready for 3D reconstruction. Such a complicated synchronization was explained by the restricted direct access to the clinical imaging hardware and software. Though, from the technical point of view, this process could be fully automatized. Temporal landmarking based on the embedded clinical ECG was performed directly by the ultrasound system, so the information about the corresponding cardiac event was accessible for each B-scan. The B-scans related to different cardiac events were grouped in different data arrays and reconstructed separately.

Within the labour experiments, however, the synchronization was automatized and the acquired B-scans with the corresponding matrices $T_R^T$ were accessible at the workstation in real-time without the merging step and ready for processing and 3D reconstruction.

The 3D reconstruction procedure consists of two steps. First, all acquired B-scans are pre-processed at the workstation and 2D contours of the scanned surfaces are segmented using the ultrasound processing algorithm *Rainfall* provided by *UltraOsteon GmbH* [71, 115, 134]. A review of the existent medical image segmentation and edge detection methods is given in Section 5.

Second, the points from the computed 2D contours are transformed into 3D space using the equation (3.1). The resulted 3D points are stored as a point cloud that can be visualized or transmitted for further 3D surface reconstruction. The description of the surface reconstruction method specially developed for this task is presented in Section 4.

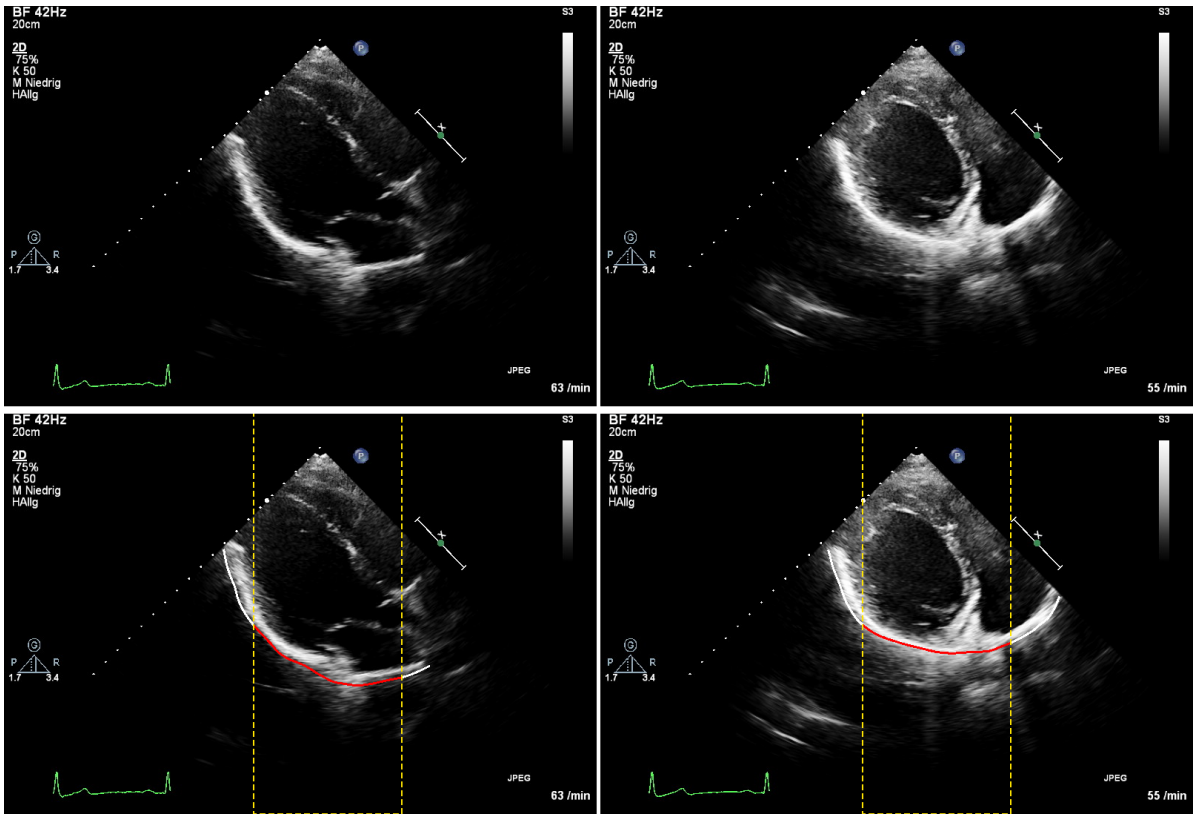


Figure 3.4.: ECG-synchronized cardiac ultrasound images acquired using Philips iE33 system (top); contour extraction principle, the left and the right parts of the contours are excluded due to the higher geometric distortions on the sides caused by ultrasonic artefacts (bottom).

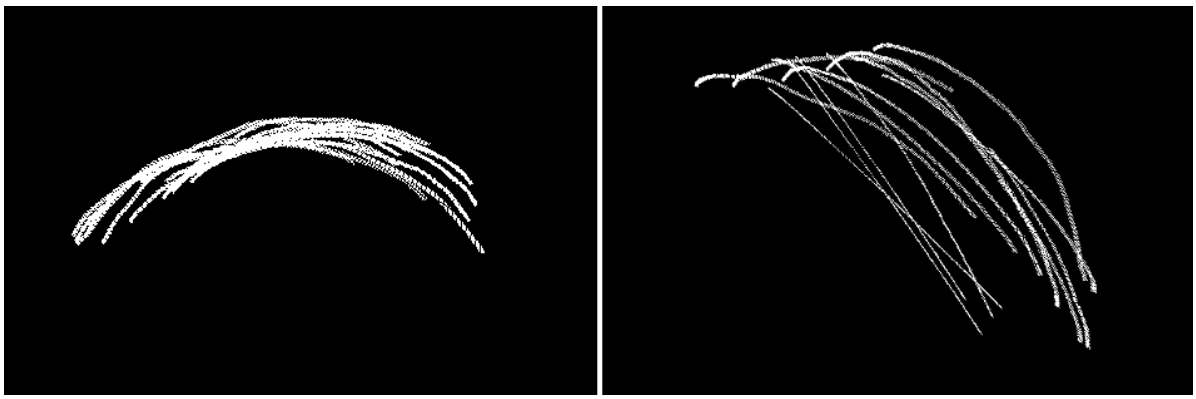

Figure 3.5.: The profile (left) and a projective view (right) of 14 extracted cardiac contours after 3D reconstruction. Contour extraction inaccuracies as well as the patient breathing and motion result in considerable geometric artefacts.

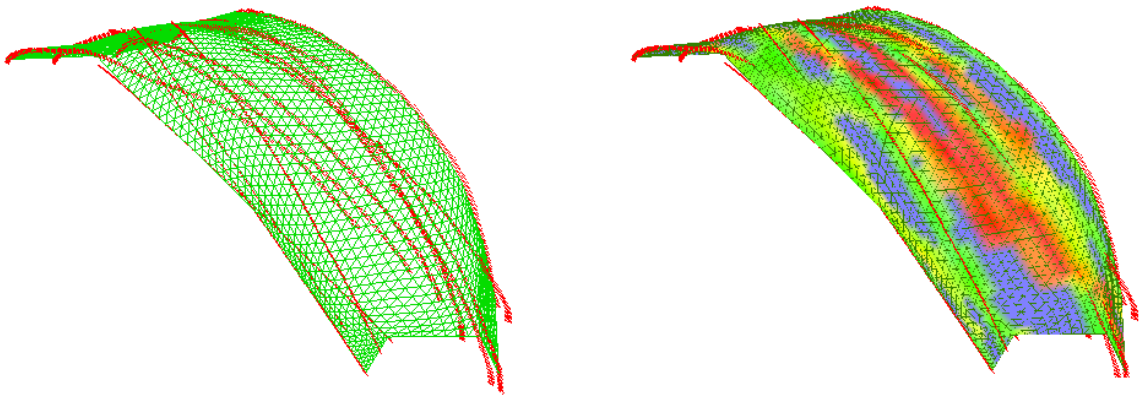

Figure 3.6.: A fragment of 3D surface reconstructed from 14 cardiac contours (left); "Temperature map" depicts closeness of the reconstructed mesh to the original data. Green to red color range corresponds to the distances from 0 to 5 mm, blue regions have no correspondence with the original points (right).

### Data visualization

A variety of methods have been recently developed for 3D ultrasound data reconstruction and rendering [63, 189]. Here we shortly describe the main visualization modes that were tested during our experiments.

A reconstructed 3D point cloud (or their sequence in the 4D mode) is given to the input of the surface reconstruction and visualization module and can be directly rendered. Although the main goal of this module is to reconstruct and to visualize the mesh surface of the input data, several alternative visualization modes (pseudo-3D, voxel-based, and model-based) are also partially supported.

*Surface rendering* is a common technique used in medical imaging for anatomical objects visualization that are represented as geometrical meshes (Figure 3.6, 3.7). This approach shows a significant advantage in rendering speed comparing with the volume rendering method, but often possesses such drawbacks as mesh constructions inaccuracy, ambiguity or object oversimplification. For reconstruction of 3D surfaces from the acquired point clouds we analysed the existent methods and developed an alternative one that is relatively fast and well suited for sparse and noisy point clouds (see Section 4).

In the *voxel-based* mode the acquired ultrasound data (3D point cloud or the original data from B-scans, that is more relevant in this case) are converted into 3D volumes by placing it in the regular Cartesian grid. A number of interpolation filters [107, 158] can be applied to this volume for hole-filling and smoothing. In our experiments we did not interpolate the resulting volume and visualized it as is. Volume rendering was implemented using 3D textures. Voxel-based visualization is a popular data representation technique in 3D ultrasound as it supports Multi-Planar-Reformatting (MRP). MRP allows extracting arbitrary planar cuts from 3D data that makes it possible to render 2D ultrasound slices even if they cannot be accessed physically by the ultrasound transducer during scanning. Orthogonal planes propose a convenient viewing and navigation interface that can be used for control of measurement markers (used for calculation of

distances and angles in 3D) and diagnostics.

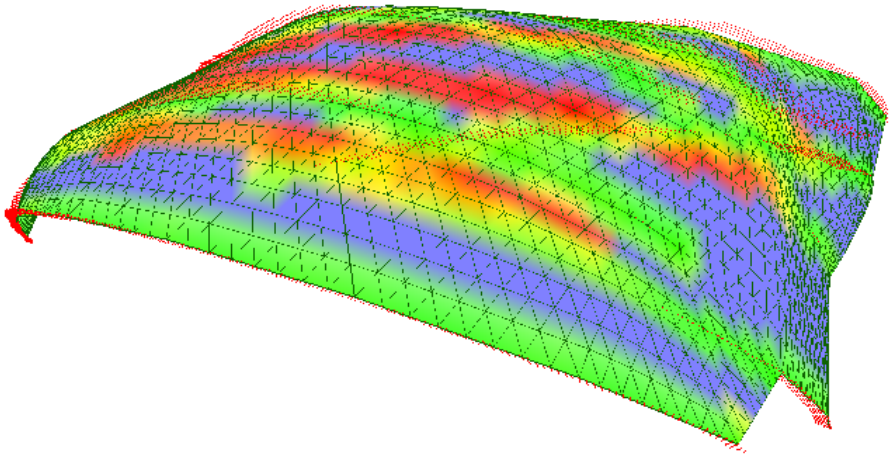

Figure 3.7.: A fragment of cardiac 3D surface. The contour data consist of 9832 points (reconstructed using Philips iE33, phased array 3 MHz). The mesh consists of 3328 faces, the mean distance to the original data 1.94 mm, RMS is 2.76 mm.

In the *pseudo-3D* visualization mode the ultrasound data are represented as a set of B-scans spatially-ordered (rotated and translated) in three dimensions (Figure 3.8) subject to the corresponding ultrasound transducer positions. The advantage of this method is the fast rendering and low system resources consumption comparing to the other visualization modes. The pseudo-3D rendering can be considered as a preferred real-time visualization and navigation mode.

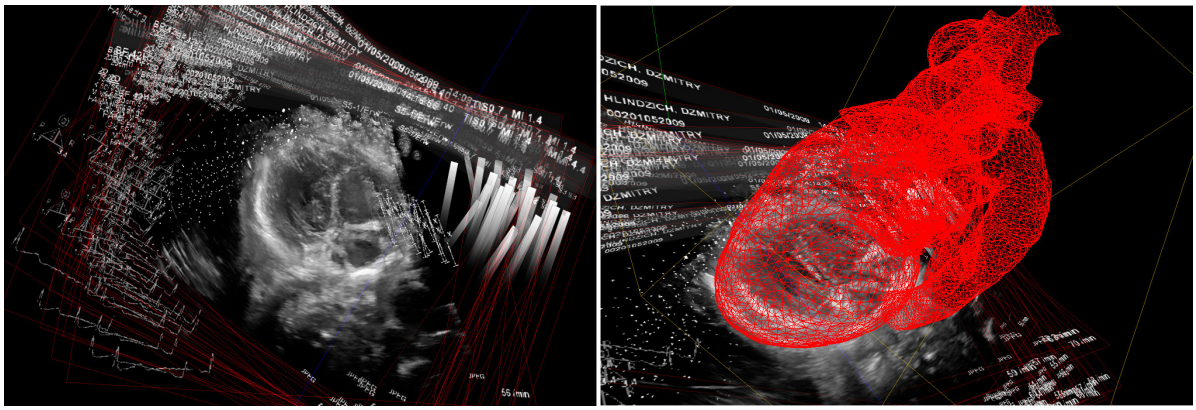

Figure 3.8.: Visualization of clinical cardiac data using Philips iE33 ultrasound system (phased array 1.5-3 MHz). Pseudo-3D (2.5D) visualization mode (left); Model-based visualization mode (right).

The *model-based* mode allows visualizing ultrasound data relatively to an organ model that is pre-loaded into the system and manually calibrated using a series of the

reference B-scans for appropriate initial scaling, rotation and translation setting. For our examination on cardiac data, a human heart model was created using cryosection data from the "Visible Human" Project, National Library of Medicine and University of Michigan [2] and a 3D content creation suite (Blender). (Figure 3.8). This mode can be used for ultrasound navigation during scanning and for training. Together with the dynamic 4D mesh-to-ultrasound morphing the model-based mode can introduce the elements of augmented reality into the scanning process, that is a perspective field of further investigations.

### 3.3.2. System validation

Validation of the system has been performed on a porcine heart specimen that have been prepared (sutured up and embalmed) and placed in a specially constructed scanning box filled with water (Figure 3.9, 3.11).

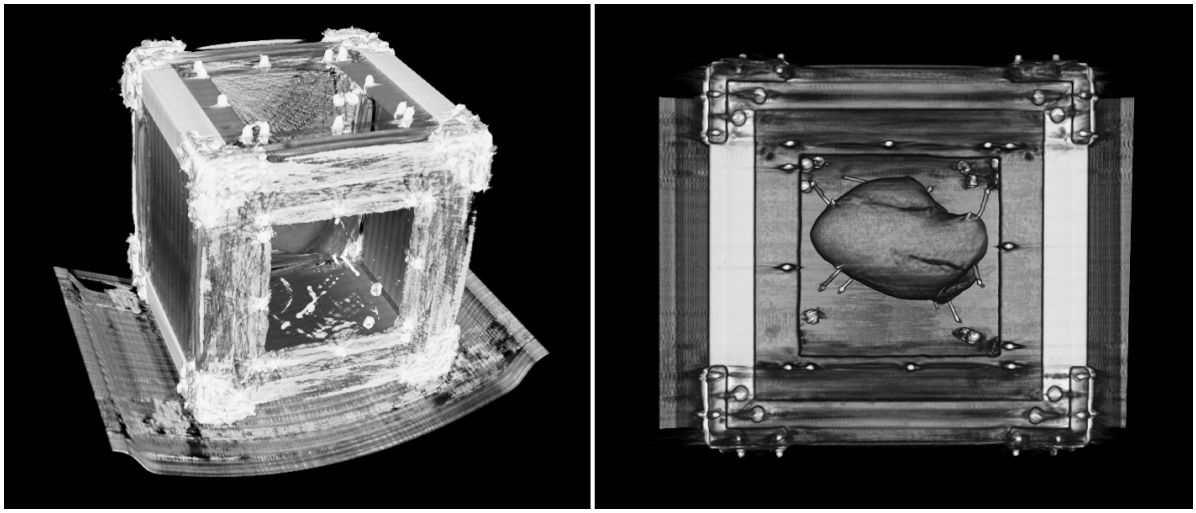

Figure 3.9.: CT image of the scanning box (25×25×25 cm), the porcine heart specimen is accessible from two membrane windows (MITO, 3D texture-based visualization).

The box is composed of 3 mm aluminum frame for high resiliency and 5 mm Plexiglas sides with two scanning windows covered with a latex-silicone membrane transparent for ultrasound. The heart specimen has been stretched out using hooks and nylon fibers strained with steel springs in order to fix the organ quickly and to avoid deformations during storage and transportation.

After the described preparation a CT volume of the heart model was acquired on *Siemens Somatom Emotion* scanner with the spatial resolution of about 0.85 mm / voxel (Figure 3.10). Next, the specimen was scanned several times using our experimental tracked 3D ultrasound system Figure (3.12).

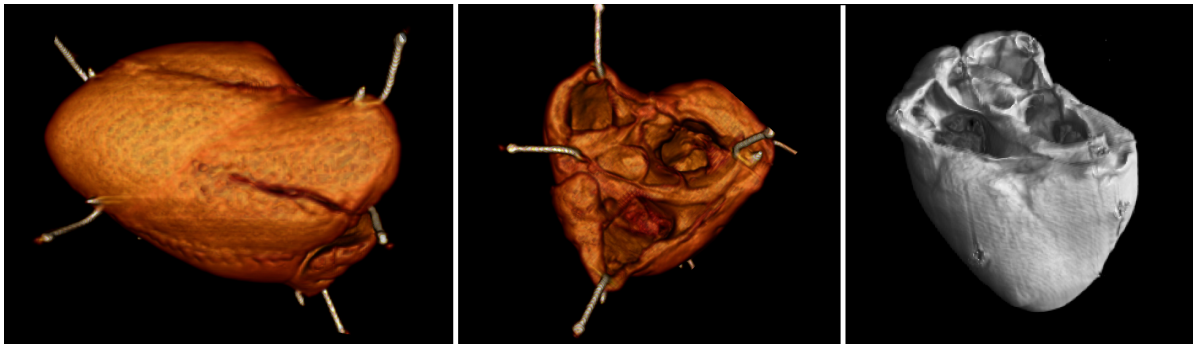

Figure 3.10.: CT image of the porcine heart specimen with the fastening (left, middle) and after manual pre-segmentation (right) (MITO, 3D texture-based visualization).

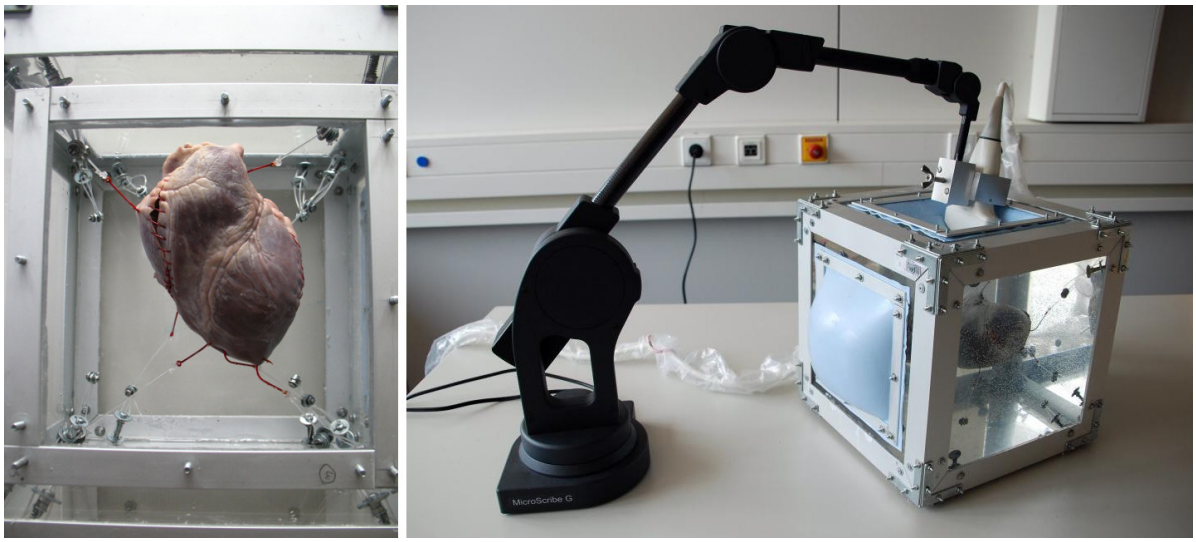


Figure 3.11.: Scanning the fixed heart specimen using the freehand tracked 3D ultrasound system.

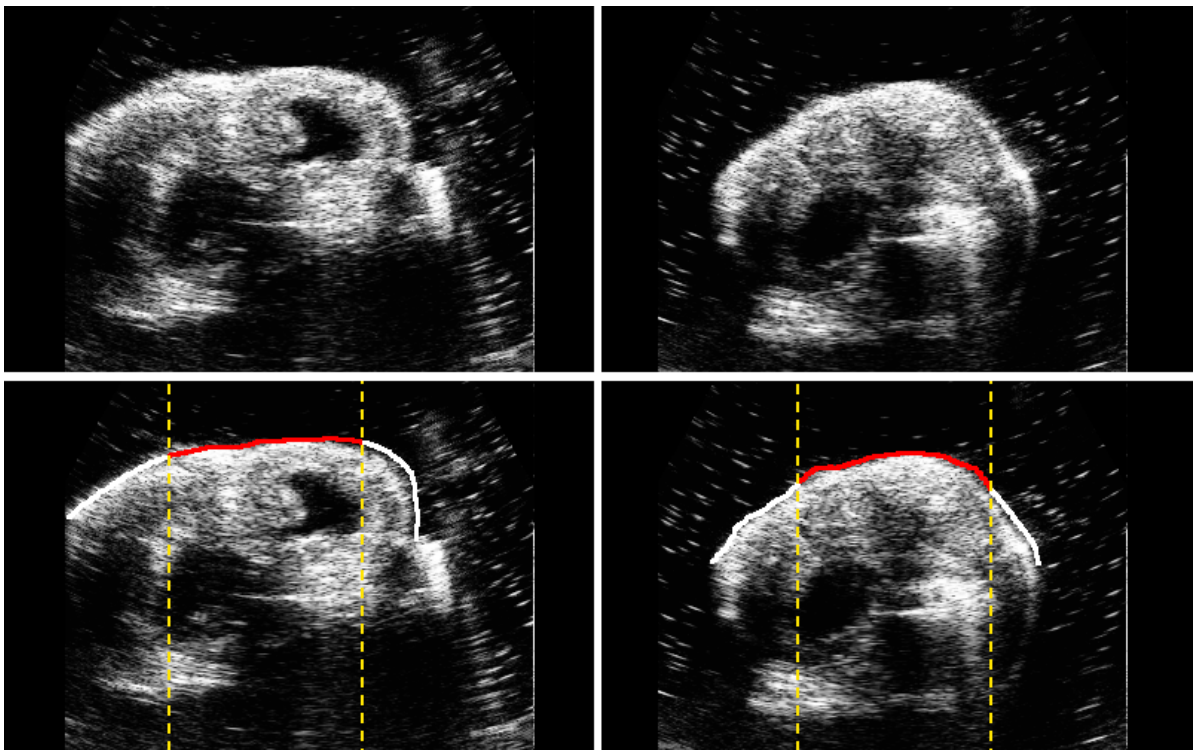

Figure 3.12.: Ultrasound images of the heart specimen acquired using the linear array HL7 transducer (top); automatic contour extraction using UltraOsteon *Rainfall* algorithm, with the following truncation of the contour ends (bottom).

The resulting models (CT and US 3D point clouds) have been registered using the Iterative Closest Point (ICP) algorithm with a manual initialization. The system scanning *accuracy* was estimated as the RMS of the distances between the closest points of the models and had value of 1.2 mm.

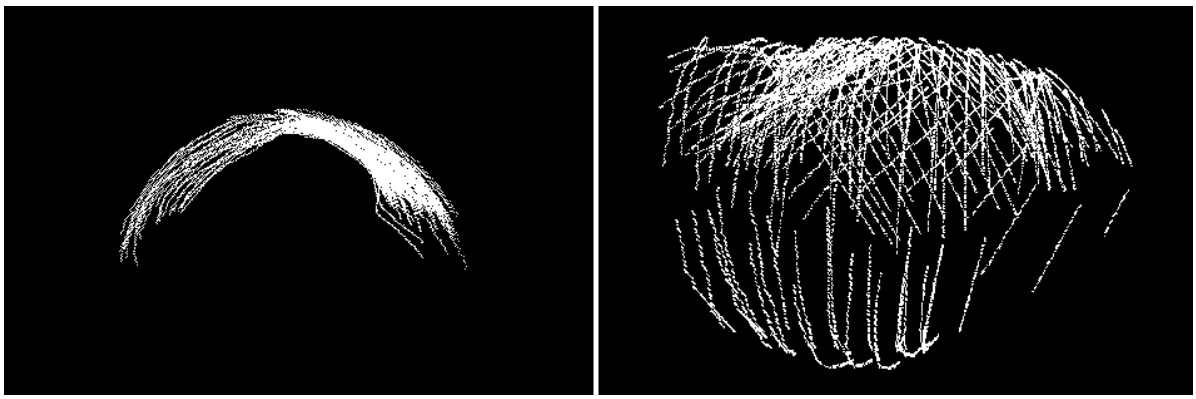

Figure 3.13.: The profile (left) and a projective view (right) of 147 extracted contours after 3D reconstruction.

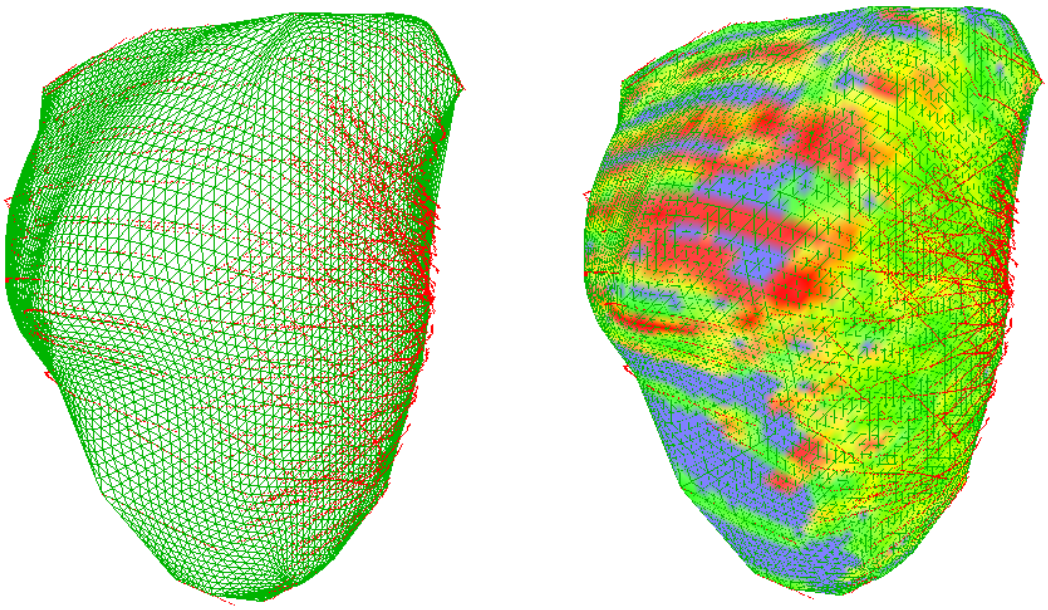

Figure 3.14.: Reconstructed 3D surface of the porcine heart specimen. The contour data consist of 16059 points (reconstructed using linear array HL7 5 MHz). The mesh consists of 7936 faces, the mean distance to the original data 1.15 mm, RMS is 1.59 mm. Green to red color range corresponds to the distances from 0 to 2 mm, while blue represents absence of correspondence. The distances were calculated using the algorithm described in Appendix A.

## 3.4. A 3D ultrasound scanning system for automated surface reconstruction of long bones

The second developed prototype is based on the sequential triggered scanning principle and is designed for 3D surface reconstruction of the long bones of the extremities[1]. Here we consider the architecture of the scanning system and its validation on artificial and biological models [25]. The scanning system is planned to support orthopaedic and trauma surgeons during contactless X-ray-free examinations of injured extremities, providing numerical data and 2D/3D visualization of bone surfaces for optimal decision making. It is designed to meet all the general requirements mentioned in Section 3.2 extended with additional two ones:

- high scanning speed,
- contactless (does not cause additional pain).

[1]The scanning body was assembled in the University Medical Centre Mannheim. The current version is motor-free, all control is performed manually using fine-mechanics screws.

## 3.4.1. Material and methods

### Scanner architecture

The general setup of the developed prototype consists of a scanning body (that performs transducer positioning and tracking functions), ultrasound machine and a workstation. The scanning body is a mechanical part of the system that includes a tank filled with water and a fixation block for ultrasound transducers (Figure 3.15a). The tank can be rotated over 360 degrees around the vertical axis, while the transducer block can be moved within 140 mm along this axis.

The system is equipped with one or two linear array transducers HL7 5-8 MHz. The transducers can be placed either in a vertical or a horizontal position on the fixation block in such a way that the beam emission is directed towards to the rotation axis of the tank. All rotations and translations in the first version of the prototype are carried out manually using fine-mechanics screws. The distance between the transducer and the rotation axis varies depending on size and shape of a scanned object. The transducers are connected to the beam formers controlled by the workstation. B-mode ultrasound images are acquired from different positions and orientations of the transducer. Bone contours are extracted automatically and reconstructed into 3D surfaces taking into account the known geometrical transformations provided by the mechanical tracking system.

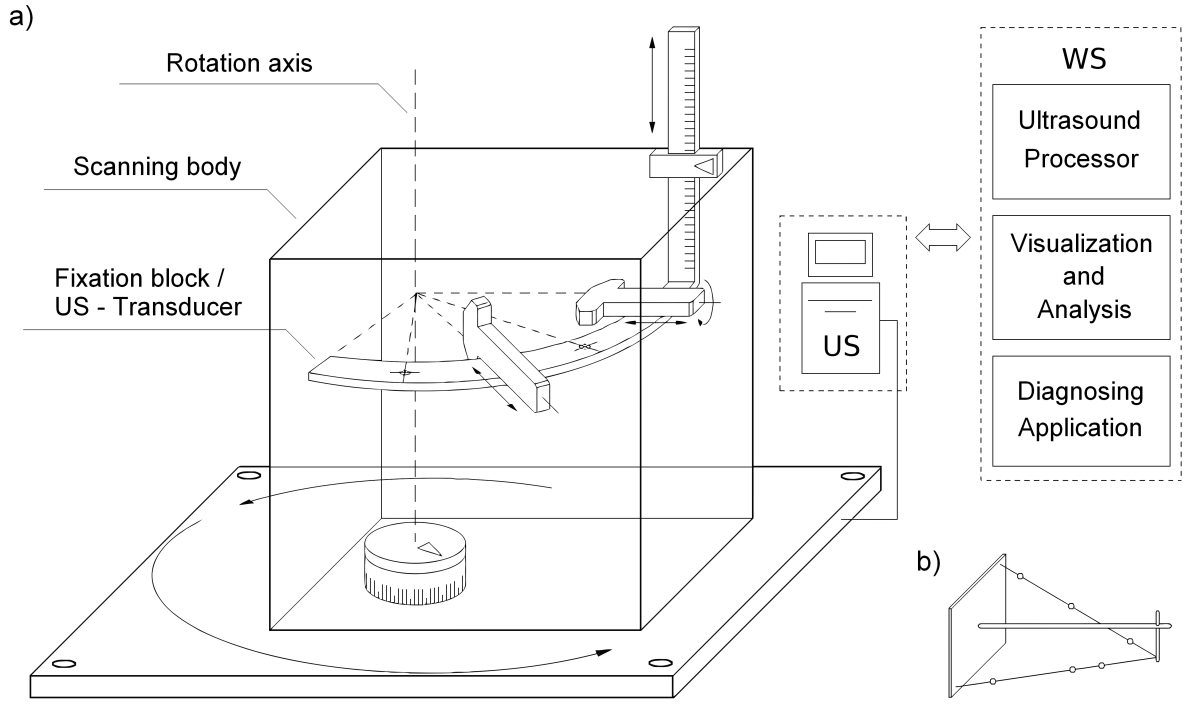


Figure 3.15.: a) General scheme of the experimental sequential triggered ultrasound scanning system (US - ultrasound machine, WS - workstation); b) multi-bead phantom used for system calibration.

### Calibration

The idea of the calibration procedure is similar to the one described in Section 3.3.1. Calibration of the system is carried out once after fixation of the ultrasound transducer. The proposed calibration method is based on a multi-bead phantom that consists of two nylon wires in 3D space (Figure 3.15b). The wires have diameter of 0.1 mm and carry glass beads with the diameter of 1 mm that are fixed on various altitudes and used as landmarks. The phantom geometry has been measured using the electromechanical tracking system *Immersion MicroScribe* with accuracy of 0.21 mm RMS. The phantom is placed into the scanning body filled with warm water and scanned by the installed transducer from various directions.

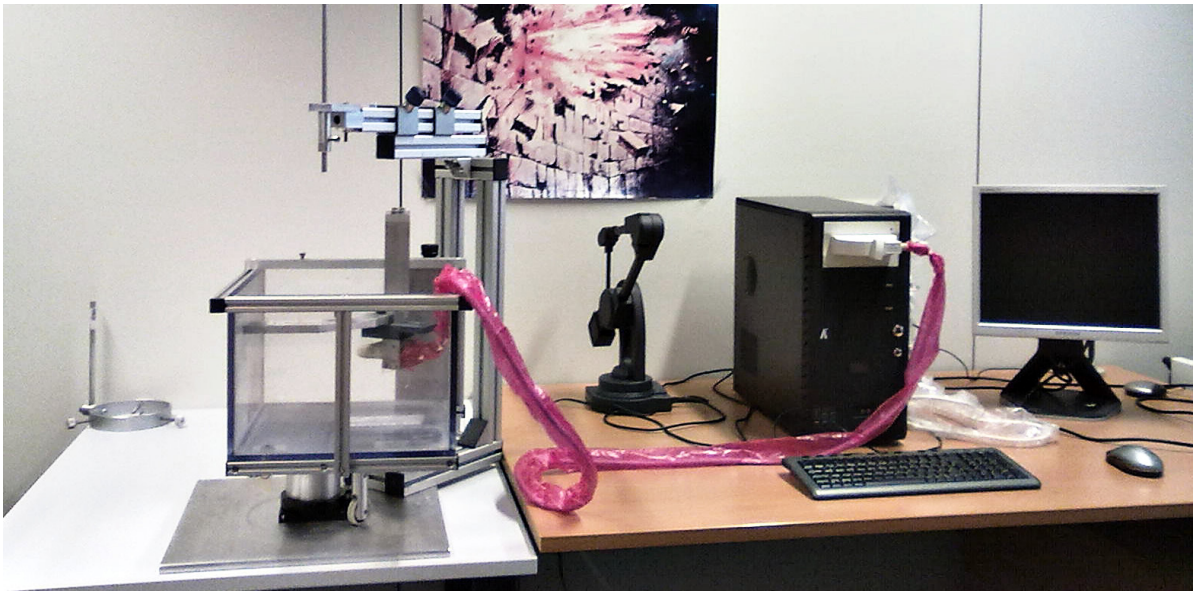

Figure 3.16.: The basic setup of the system.

Similarly with the calibration of the freehand ultrasound (Section 3.3.1), we use the following coordinate systems: the coordinate system of B-scans ($P$), the coordinate system of the transducer fixation block ($R$), the coordinate system of the scanning system itself ($T$) and the coordinate system of the phantom ($C$). The transformation from the coordinate system $P$ to $C$ is described by formula (3.1).

The matrix $T_R^T$ is defined by the tracking mechanism of the scanning system for each position of the transducer. The transformation $T_T^C$ is calculated using the *MicroScribe* digitizer once after the phantom has been installed. The goal of the calibration is to find the unknown calibration matrix $T_P^R$ that translates two-dimensional coordinates of points from B-scans into the coordinate space of the the transducer fixation block. The landmarks of the phantom with the known (pre-measured) relative coordinates $F_i, i = \overline{1, N}$ are scanned from different positions and the corresponding points $P_i, i = \overline{1, N}$ are manually detected on the acquired B-scans. In order to calculate the calibration matrix $T_P^R$ the following system of linear equations are solved using the Levenberg-Marquardt

algorithm [127]:

$$F_i = T_T^C \cdot T_{Ri}^T \cdot T_P^R \cdot P_i, i = \overline{1, N} \tag{3.3}$$

where $F_i = (x_i, y_i, z_i, 1)^T$ - coordinates of the landmarks in the phantom coordinate system, $P_i = (S_x u_i, S_y v_i, 0, 1)^T$ , $(u_i, v_i)$ - corresponding coordinates on B-scans. $S_x$ and $S_y$ are scale factors of B-scans along $X$ and $Y$ axes respectively and, finally, $T_{Ri}^T$ - transformations corresponding to the position of the transducer from which the *i-th* landmark has been scanned. In general the system of equations (3.3) has 6 unknowns (3 for rotation and 3 for translation), so, theoretically the number $N$ of landmarks must be not less than 2. But practical experiments show that a stable calibration is achieved at much bigger values of $N$ (in our experiments we used $10 \leq N \leq 14$). The accuracy of calibration was measured as RMS (3.2) of the residuals in the linear system (3.3), and was about 0.16 mm in our experiments.

### Data acquisition and representation

Similarly with the freehand system (Section 3.3.1), 3D reconstruction consists of two steps: bone contours extraction and 3D coordinates calculation. At the first step bone contours are extracted from the acquired B-scans automatically using the Rainfall algorithm [71, 115, 134]. Then, 2D coordinates of the contour points $P_j$ are transformed into the coordinate space of the calibration phantom using the known transducer calibration and the corresponding transformation matrices using formula (3.1). The scanned object is stored in the system as a *point cloud*. In this form the data is available for further processing, analysis (such as registration with other objects, surface reconstruction) and visualization.

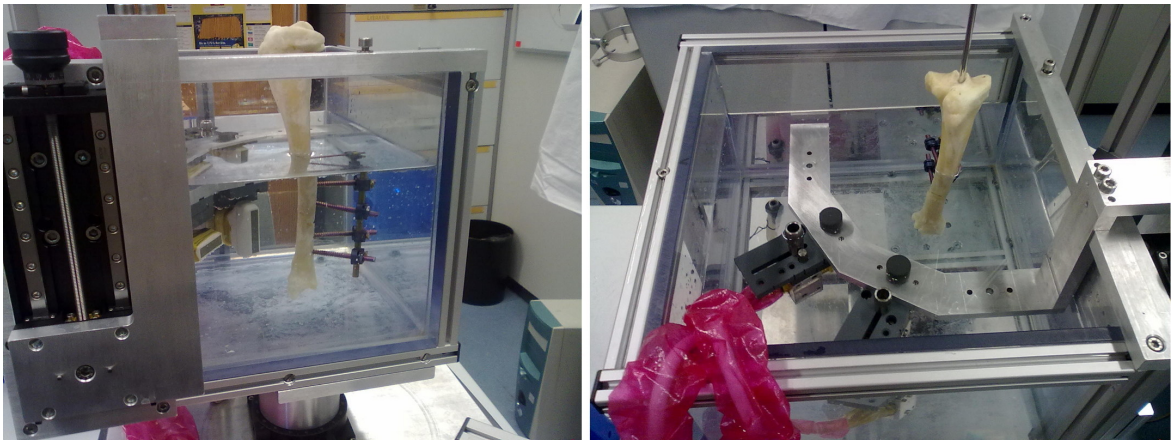

Figure 3.17.: Scanning a sheep bone with several fixed fractures using a two-transducer setup (a vertical and a horizontal transducers). Vertical displacement mechanism of the transducer block (left) and the rotational mechanism (right).

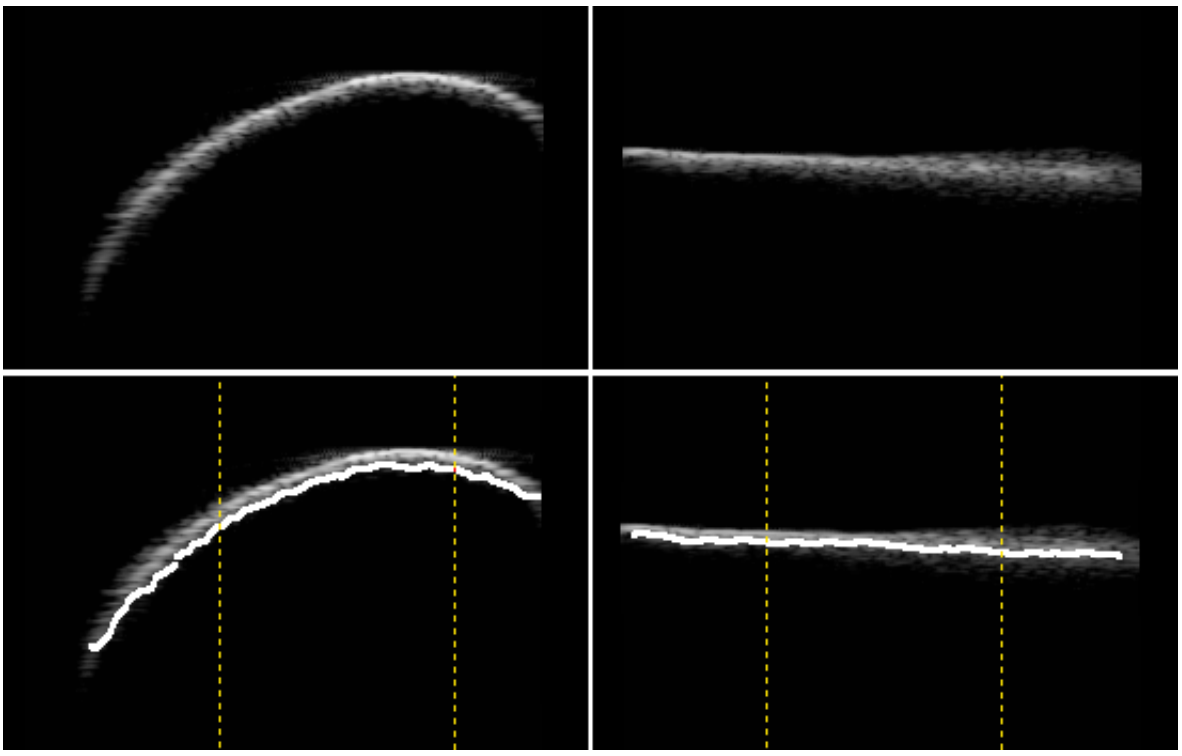

Figure 3.18.: Ultrasound images of the sheep bone specimen acquired using the linear array HL7 transducer. Top: typical examples of images from the horizontal (left) and the vertical (right) positioned transducers. Bottom: automatic contour extraction using the *Rainfall* algorithm.

### Data visualization

After the data acquisition and 3D-reconstruction the scanned object can be visualized optionally as a point cloud or a mesh surface.

The point cloud visualization mode provides the most adequate raw representation of bone ultrasonic data that can be used for diagnostics and analysis of the bone geometry.

The surface visualization mode is utilized for an alternative data representation and the standard export for external rendering applications. Like many anatomical structures, the long bones of the human extremities mainly have smooth surfaces, therefore sharp geometric details are inherently excluded from the considered point clouds. It makes the surface reconstruction process easier and more stable. Moreover, a series of experiments show that due to the inherently smooth geometry of long bones it is possible to emphasize bone areas with possible fractures by calculating local distances from the reconstructed surface (with specially selected parameters of the reconstruction algorithm) to the point cloud (Figure 3.20). However, this feature is not yet strictly proven and needs further investigation.

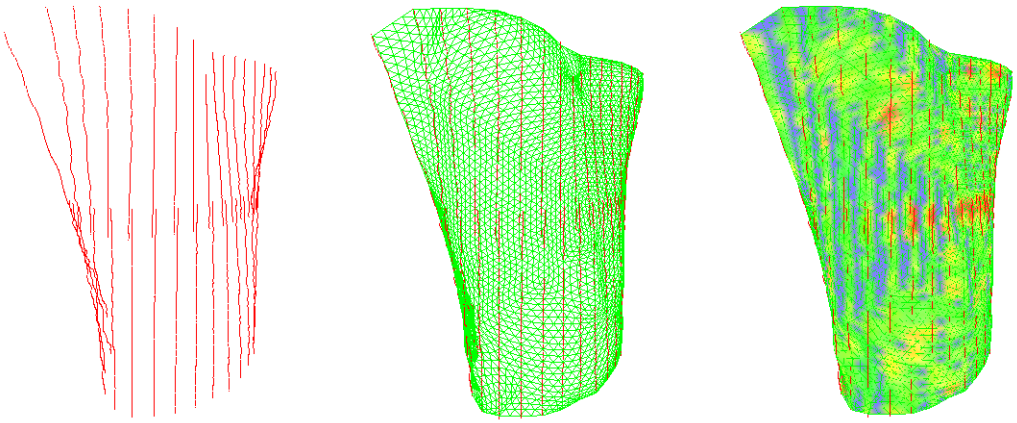

Figure 3.19.: A fragment of the sheep bone specimen scanned with the vertically positioned transducer. Left: the original contours reconstructed from 32 B-mode images (7168 points). Middle: surface mesh (5136 faces). Right: distances from the mesh to the original data (green to red color range corresponds to the distances from 0 to 1 mm respectively).

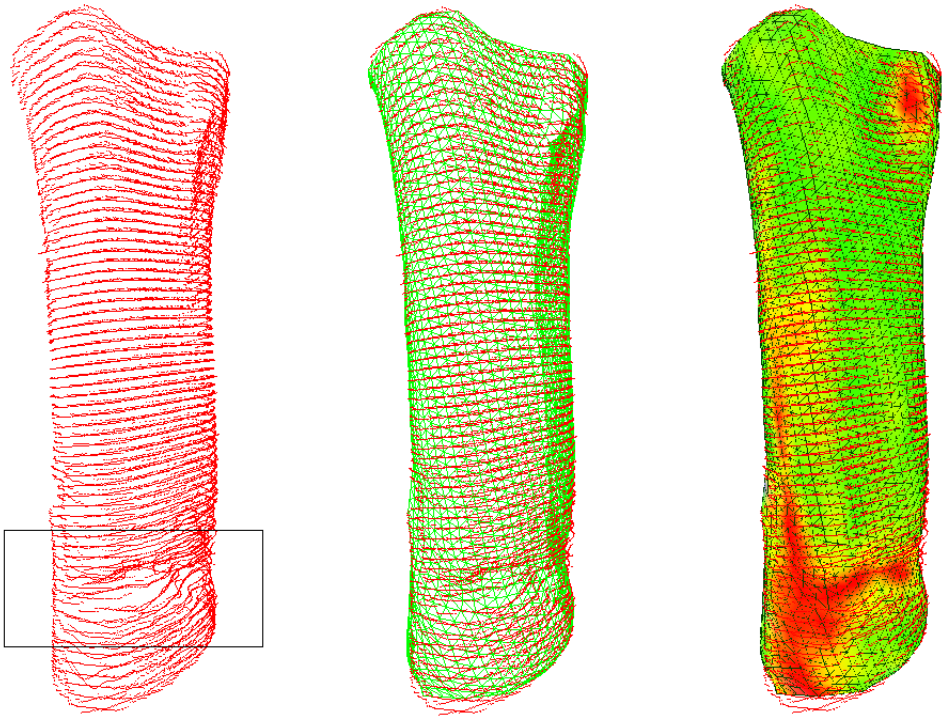

Figure 3.20.: A fragment of the sheep bone specimen with a fracture in the lower part (in a frame), scanned with the horizontally positioned transducer. Left: the original contours reconstructed from 264 B-mode images (40899 points). Middle: surface mesh (5136 faces). Right: distances from the mesh to the original data (green to red color range corresponds to the distances from 0 to 1 mm respectively).

In our experimental system we propose to use the following two surface reconstruction algorithms: mesh-growing from sparse data based on the Complex Propagation described in Chapter 4 and surface reconstruction from parallel slices (Section 4.6). The first one can be used in both vertical and horizontal transducer scanning modes, while the second one is suitable only for the horizontal mode, but provides a better performance.

### 3.4.2. System validation

A validation study has been carried out on a distal femur sawbone from *Synthes*. The main goal of the study was to test the accuracy and geometrical correctness of 3D reconstruction provided by the developed system. The geometrical correctness was estimated by 3D visualization of the acquired model and the following mesh surface reconstruction (Figure 3.21). The accuracy was measured as the distance error between the models reconstructed by our system and a preoperative reference model (Table 3.2) acquired from the same femur sawbone on *Philips Mx8000D* CT scanner with the resolution of 0,2 x 0,2 x 0,2 mm. The reconstructed set consisted of ten models of 24 takes (ultrasound contours) each.

| **Model** | **Number of takes** | **Mean, mm** | **Standard deviation, mm** | **RMS, mm** |
|---|---|---|---|---|
| 1 | 24 | 0.161132 | 0.119180 | 0.200418 |
| 2 | 24 | 0.167672 | 0.124697 | 0.208957 |
| 3 | 24 | 0.156245 | 0.117398 | 0.195435 |
| 4 | 24 | 0.156991 | 0.119513 | 0.197305 |
| 5 | 24 | 0.167658 | 0.128608 | 0.211304 |
| 6 | 24 | 0.167842 | 0.132509 | 0.213845 |
| 7 | 24 | 0.163644 | 0.125867 | 0.206451 |
| 8 | 24 | 0.162891 | 0.123839 | 0.204621 |
| 9 | 24 | 0.164278 | 0.123082 | 0.205272 |
| 10 | 24 | 0.164784 | 0.127320 | 0.208241 |

Table 3.2.: Distal femur sawbone *Synthes*. Distance errors between the model points reconstructed by the developed prototype and the points of the reference CT model after the registration procedure.

Additionally, the results of 3D reconstruction were compared with two sets of alternative models acquired from the same sawbone. The first set of models was acquired using a touch probe with attached light reflective balls. Spatial coordinates of the probe were detected using an optical tracking system from *Polaris Vicra*. This method represented a standard intraoperative modelling approach. The set consisted of ten models of 24 takes (frames) each. The models were registered with the same preoperative reference CT-model and the corresponding distance errors were calculated (Table 3.3).

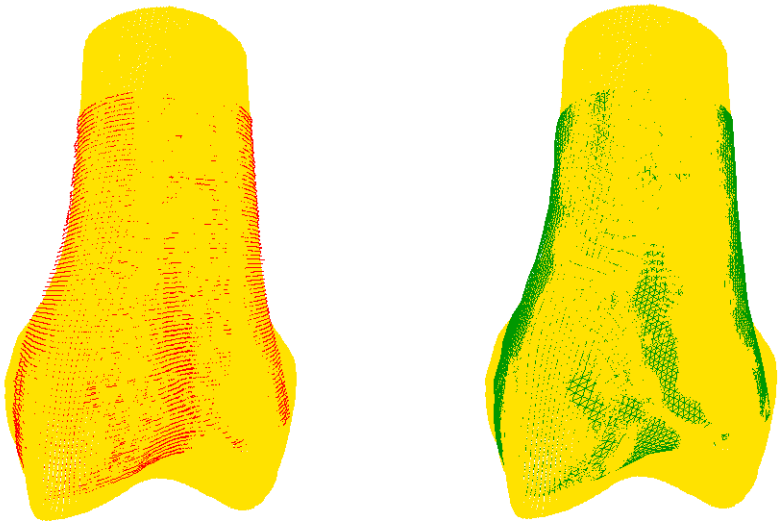

Figure 3.21.: Registration of the reference CT model (point cloud, 1.2M points) with the data acquired using the tested scanner prototype. Ultrasound contours as a point cloud (57338 points, 228 scans) (left) and the corresponding mesh surface (7600 faces) (right).

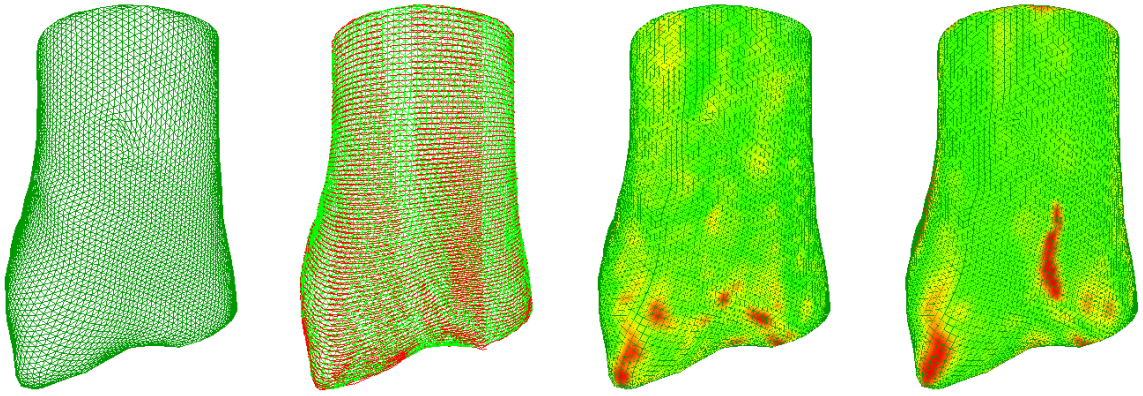

Figure 3.22.: Validation on the distal femur sawbone *Synthes*. From left to right: the reconstructed surface mesh (7600 faces); the mesh with the original ultrasound contours (228 scans); distances between the mesh and the ultrasound contours; distances between the mesh and the reference CT data (green to red color range corresponds to the distances from 0 to 1 mm respectively).

The second set of models was acquired using freehand tracked 3D ultrasound. A navigated (with attached optical markers) ultrasound probe was moved over the distal part of the femur. The position of the transducer in the operation space was determined using *Polaris Vicra*. For each scan, the orientation of the ultrasound probe was selected in such a way that the reflection of ultrasound echoes from the bone surface was maximal. The set consisted of ten models with 24 takes (ultrasound contours) in each. The models were registered with the same preoperative reference CT-model and the corresponding

distance errors were calculated (Table 3.4).

| Model | Number of takes | Mean, mm | Standard deviation, mm | RMS, mm |
|---|---|---|---|---|
| 1 | 24 | 0.108124 | 0.066710 | 0.127047 |
| 2 | 24 | 0.108773 | 0.041309 | 0.116353 |
| 3 | 24 | 0.124175 | 0.071173 | 0.143126 |
| 4 | 24 | 0.120516 | 0.050279 | 0.130584 |
| 5 | 24 | 0.089583 | 0.036897 | 0.096884 |
| 6 | 24 | 0.143007 | 0.074936 | 0.161451 |
| 7 | 24 | 0.104318 | 0.047622 | 0.114674 |
| 8 | 24 | 0.124703 | 0.055100 | 0.136333 |
| 9 | 24 | 0.100614 | 0.036452 | 0.107014 |
| 10 | 24 | 0.100524 | 0.037861 | 0.107418 |

Table 3.3.: Distal femur sawbone *Synthes*. Distance errors between the points of the *touch probe* models and the reference CT model after the registration procedure.

| Model | Number of takes | Mean, mm | Standard deviation, mm | RMS, mm |
|---|---|---|---|---|
| 1 | 24 | 0.185344 | 0.158972 | 0.244182 |
| 2 | 24 | 0.203526 | 0.196033 | 0.282581 |
| 3 | 24 | 0.186016 | 0.155833 | 0.242664 |
| 4 | 24 | 0.160486 | 0.128648 | 0.205684 |
| 5 | 24 | 0.162062 | 0.118180 | 0.200576 |
| 6 | 24 | 0.167543 | 0.117334 | 0.204544 |
| 7 | 24 | 0.203335 | 0.144615 | 0.249516 |
| 8 | 24 | 0.179706 | 0.122531 | 0.217504 |
| 9 | 24 | 0.211337 | 0.166038 | 0.268760 |
| 10 | 24 | 0.210310 | 0.191252 | 0.284267 |

Table 3.4.: Distal femur sawbone *Synthes*. Distance errors between the points of the *free-hand usltrasound* models and the reference CT model after the registration procedure.

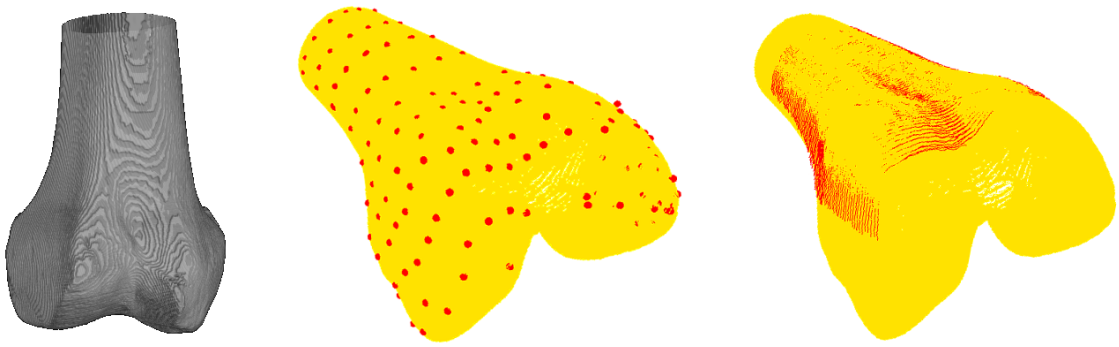

Figure 3.23.: Validation of the scanner prototype on the distal femur sawbone *Synthes*. Left: 3D view of the reference CT model. Middle: the result of registration of a touch probe model with the CT model (point clouds). Right: the result of registration of a model acquired using the developed prototype with the CT model (point clouds).

## 3.5. Results and discussion

Concerning the freehand ultrasound prototype, the estimated error provides only a first picture of how accurate such a scanning system can be. The real accuracy obtained on clinical data can considerably differ from this value due to ultrasonic artefacts appearing in the scanned tissues, patient movement and inaccurate scanning. However, the obtained results show the order of magnitude of the accuracy (1.2 mm RMS relatively to the corresponding CT model) that can be achieved by the proposed architecture in the close to ideal conditions. This is although still less accurate than modern radiography, CT and MRI, but significantly better than the first freehand ultrasound systems [74]. From this point of view the obtained results are quite promising, but from another side it is obvious that further investigation and development are needed before the system can be applied in practice. Though the system was validated only on a heart specimen, it can be used in the same way for 3D surface reconstruction of any viscera and bones that are accessible for the conventional ultrasound. It also should be noted, that acquisition of large amount of data using the freehand approach can be quite time consuming, therefore automatic techniques are more preferable for detailed surface reconstruction.

The validation study of the second scanning system showed feasibility of 3D reconstruction of long bones with an accuracy of about 0.2 mm RMS. This is comparable to the accuracy of modern radiography, CT and MRI. However, this result is highly depended on the accuracy of calibration and segmentation quality. The experiments proved that the shapes of the long bones can be successfully reconstructed from sets of 2D B-Mode images and can be used further for fracture analysis and deformity measurements. The problems of the reconstruction quality enhancement and the acquisition time reduction seem to be solvable by a robotised (motor-based) system integration.

The *accuracy* of the developed prototype is slightly lower than the accuracy of intraoperative touch-point modelling and is higher than the one of freehand ultrasound based on optical probe tracking. Mechanical tracking systems do not suffer from orientation

and illumination artefacts and provide regular data sets that can be processed in a more effective way. Due to minimization of the human factor in the scanning procedure, automatized sequential triggered ultrasound systems can provide better *precision* (the degree to which repeated scans under unchanged conditions show the same results) than the touch-point and freehand ultrasound systems. This was experimentally shown in the performed validation study (Tables 3.2, 3.3 and 3.4).

The reconstruction accuracy in the case of real bones was slightly higher than in the case of sawbones due to the higher density of real bones and therefore better reflection of echoes. Scanning with a transducer oriented parallel to the long axis of bones provides better reconstruction in comparison with the transverse orientation of the transducer. This result is explained by a higher amount of scatter reflections detected by the transducer in the last case. We have also found out that the developed prototype is able to reconstruct cross fractures if their thickness is bigger than 0.6-0.8 mm (depends on the length of ultrasonic waves).

Integration of a robotized system can provide full automation and acceleration of the scanning process without additional attention from an acquisition surgeon. The contactless scanning principle of the system allows a patient to avoid additional pain. The considered approach makes it possible to develop low-cost diagnostic systems for analysis of skeleton parts with a simple anatomy. However, we have to notice the following limitations of the proposed system: 1) analysis of joint areas is extremely difficult or even impractical, 2) open fractures and any damages of the skin are contraindicated for scanning with this system, 3) analysis of complex fractures has to be avoided, due to the high amount of ultrasonic artefacts.

# Part II.

# Medical image analysis. Investigated and developed special-purpose techniques

Modern medical imaging devices produce a great amount of data. In most cases these data are represented by 2D, 3D images or image sequences. In order to make it accessible for medical diagnostics and treatment the acquired data must be processed, analysed and prepared for visualization. These tasks are solved by a class of methods that form the field of study of medical image analysis. Along with active development of medical imaging technology it has become a highly essential and a wide field of scientific research. Being a sub-discipline of a more general subject field, image analysis, medical image analysis principally solves the same problems, but considering medical origin of the images. This specialization often requires new sophisticated computerized quantification and analysis algorithms and tools that can be unique for a concrete medical imaging modality.

The most common problems solved by the methods of medical image analysis are image segmentation, filtering, various types of 2D and 3D reconstruction, object detection and recognition. In the following chapters we will discuss *3D surface reconstruction*, *image segmentation* and *object detection*. We provide a review of the state of the art methods for each problem and describe several special-purpose techniques, that although have been initially developed for medical imaging applications, can be also used for solution of more general problems.

# 4. Mesh-growing 3D surface reconstruction from sparse ultrasound data

## 4.1. State of the art

Surface reconstruction from point clouds is a major topic of research in the field of computer vision, engineering, virtual reality and computational medicine. There is a wide range of applications that utilize surface reconstruction and give rise to a number of conjugated exploratory problems. Various types of mechanical, optical and ultrasonic sensors are involved in such applications to scan surfaces of 3D objects. The reconstructed surface models are widely used for further simulation, shape analysis, animation and production of various visual effects.

Since the last decade a great number of 3D surface reconstruction methods has been developed. They differ in the application field, characteristics of input data and properties of generated surfaces. Recent overviews of existing methods can be found in [170, 35, 142]. Most generally they can be divided into two main classes [95]: *implicit* and *explicit*. The implicit methods approximate an input point cloud by an implicit function and reconstruct the surface as a zero-level set of the evaluated function. The

recent research in this field is mainly presented by the methods of moving least squares [128, 129], Poisson surface [111], partition of unity and radial basis functions [187]. The explicit methods, in turn, are aimed at direct triangulation of a given point cloud. They are presented by Voronoi/Delaunay-based and mesh growing approaches. The first ones reconstruct a surface mesh by performing the 3D Delaunay triangulation [13] of the point cloud, e.g. Crust [5], Cocone [6] or their modifications [7]. These algorithms provide theoretical guarantees for mesh reconstruction under several conditions for well-sampled data [5] but encounter with difficulties in the case of noisy and under-sampled point clouds. Within the mesh growing approaches the surface reconstruction is started from a seed triangle and the mesh is iteratively propagated under some criteria. These algorithms utilize the techniques based on ball-pivoting [21], intrinsic property of point clouds [135], Gabriel 2 - simplex (G2S) criterion [53] and 2.5D active contours [56]. In spite of the fact that the mesh growing methods may produce artefacts in regions with high surface curvature and complex geometry, they work well on noisy and irregular data and implement an intuitive and flexible surface reconstruction paradigm.

Another important problem that often follows 3D surface reconstruction is enhancement of the built surface meshes. There are two main classes of the mesh enhancement methods presented in the literature: *mesh smoothing* (also called nodes relaxation) and *subdivision surface*. The first class [84] involves the methods of spatial modification of the mesh nodes preserving its topology. These methods produce relatively smooth surfaces without additional complication of the intra-mesh connections. But as it follows from the definition, the mesh smoothing methods are not preferable in the case when it is important to preserve the coordinates of the original nodes and to avoid high deviations from the initial mesh shape.

The methods from the second class solve the smoothing problem by introducing additional connections into the mesh topology. Namely, they recursively subdivide the mesh faces into sub-faces that better approximate a smooth surface. Considering their refinement schemes the subdivision surface methods can be separated into *approximating* and *interpolating* subclasses. Nowadays the approximating subdivision methods (e.g. Catmull-Clark [30], Doo-Sabin [55], Loop [137]) have become a powerful tool widely used in computer graphics and geometric modelling. They produce relatively fair surfaces (the limit surfaces possess up to $C^2$ continuity) but similarly to the smoothing methods only approximate the initial mesh acting as low-pass filters. Nevertheless, in the last years several works [36, 52, 51] presented approximating subdivision schemes for construction of meshes whose limit surface interpolates the nodes of the initial mesh. Such methods are usually more complicated and require non-trivial setup of supplementary parameters that have significant influence on the shape of the limit surface [52].

Alternatively, the classical interpolating subdivision methods (e.g. Butterfly scheme [60, 209], Kobbelt [114]) are relatively simple and produce rather smooth surfaces that interpolate the initial mesh nodes. Later works on mesh interpolation [203] have introduced face based and normal based subdivision schemes that better preserve sharp features of the control mesh but need more computational time. It should be noticed that the subdivision surface methods are much more sensitive to mesh irregularities and inferior to the smoothness of the approximating schemes (normally $C^1$ continuity). Nevertheless, these two disadvantages often play a secondary role in the selection of a fast and simple interpolating method. Moreover, excessive surface smoothness and high

deviations from the initial mesh shape may be undesirable in medical applications where local anatomical features have to remain after the smoothing process.

## 4.2. Objectives

In this chapter we refer to the problem of surface reconstruction from noisy sparse data received from tracked 3D ultrasound scanners [92, 91]. Here we consider mainly freehand ultrasound and sequential triggered scanning systems described in Chapter 3. We target at reconstruction of surfaces from anatomical structures such as long bones of the human extremities, myocardium and other viscera with relatively smooth surfaces and, therefore, do not consider the surfaces with complicated geometry such as joints and the small bones of the hand and foot. So, complex geometric features are inherently eliminated in the corresponding point clouds. However, due to the nature of the tracked 3D ultrasound and the architectural features of the mentioned scanning systems, the obtained data is mostly irregular, noisy and can be rather sparse.

Taking into account these properties, we aim at developing a fast surface reconstruction method that is able to process irregular point clouds. Particularly, due to its simplicity and tolerance to sparse data, we are focusing on the mesh-growing approach based on 2.5D active contours [56] and its adaptation for the obtained object clouds.

## 4.3. Material and methods

Let $P = \{P_i\}$, $P_i \in \boldsymbol{R}^3$ , $i = \overline{1, N}$ be a point cloud in 3D representing an anatomical surface $S \subset \boldsymbol{R}^3$ acquired using a freehand or a sequential triggered 3D ultrasound scanning system. Due to the properties of such scanning systems the points in $P$ are grouped into a set of planar contours forming in 3D space a sparse image of $S$. The main goal of the proposed algorithm is to reconstruct the original surface $S$ using the given point cloud $P$. At the initial stage the acquired points are stored in a $k$d-tree container and initially marked as unprocessed. We denote as $P_u$ and $P_v$ the sets of unprocessed and already processed points respectively. The proposed method realizes an interpolation growing model and consists of three steps: initializing, mesh growing and post-processing.

### 4.3.1. Mesh initialization

At the initialization step a seed triangle $S_0$ is created in accordance with the following rule:

a) the first vertex $P^1 \in P_u$ of the triangle is selected in a random way from the available unprocessed points;

b) the second vertex $P^2 \in P_u$ is chosen as the nearest unprocessed point to $P^1$ such that $\Delta_{min} < d(P^1, P^2) < \Delta_{max}$ , where $d(x, y)$ is the $L^2$-norm of the vector $x - y$, $\Delta_{min}$ and $\Delta_{max}$ are predefined parameters that respond for the scale of the mesh triangles;

c) the third vertex is defined by the following relation that forces the resulted triangle to be close to an equilateral one:

$$P^3 = arg \min_{\substack{P \in P_u \\ \Delta_{min} < d(P,P^1),\, d(P,P^2) < \Delta_{max}}} \rho(P, P^1, P^2) + \rho(P, P^2, P^1),$$

where $\rho(x, y, z) = |d(x, y) - d(y, z)|$. The edges of the seed triangle $S_0$ are marked as active and are considered as the initial deformable contour that will iteratively expand over the object surface. The vertices $P^1$ , $P^2$ and $P^3$ are marked as processed and moved from $P_u$ to $P_v$.

### 4.3.2. Mesh growing

At the growing step the mesh boundaries that include active edges propagate iteratively. In a general form the process of the boundary propagation is described by the following differential equation proposed in [56]:

$$\frac{\partial S(x,t)}{\partial t} = \vec{F}(x,t),\ S(x,t_0) = S_0(x),$$

where $S(x, t)$ is a parametric representation of the contour at time $t > t_0$, $x \in [0, 1)$. $S_0(x)$ is the initial contour and $\vec{F}(x, t)$ defines the propagation direction at the contour position $x$ and time $t$. For the calculation of the surface growth direction we propose to use the following approach, that we call *the Complex Propagation*:

$$\vec{F}(x,t) = (1-\alpha)\vec{\nu}(x,t) + \alpha\vec{\tau}(x,t). \tag{4.1}$$

Here $\vec{\nu}(x, t)$ is the inertial propagation vector defined by the preceding expansion of the reconstructed surface and $\vec{\tau}(x, t)$ is the tangential propagation vector that is defined by the local tangent plane of the point cloud at the position $x$ and the time $t$. The local tangent plane is calculated within the neighbourhood of $x$ with the radius $\varepsilon$:

$$N_\varepsilon(x,t) = \{P \in P, d(P, S(x,t)) < \varepsilon\}, \tag{4.2}$$

using the principle component analysis (PCA). The parameter $\alpha \in [0, 1]$ is the point cloud saturation factor that defines the rate of inertial and tangential growth within the surface propagation process in dependence with the local density of the point cloud. Introduction of the inertial term allows sustaining of the propagation process in local regions of the point clouds where the data are sparse and irregular (Figure 4.1b), while the tangential term provides robustness of the reconstruction process to the noisy data and closeness of the surface to the point cloud in regions with sufficient point density and strong geometric variations such as high curvature and discontinuities (Fig. 4.1c).

The saturation factor $\alpha$ can be defined as a constant value for uniform point clouds or calculated as the *adaptive saturation* for non-uniform sparse data:

$$\alpha = \alpha_\varepsilon(x,t) = \begin{cases} |N_\varepsilon(x,t)|/\lambda_\varepsilon\ ,\ |N_\varepsilon(x,t)| < \lambda_\varepsilon\ , \\ 1,\ otherwise \end{cases} \tag{4.3}$$

where $\lambda_\varepsilon$ is the predefined saturation threshold value. This intuitive rule allows the

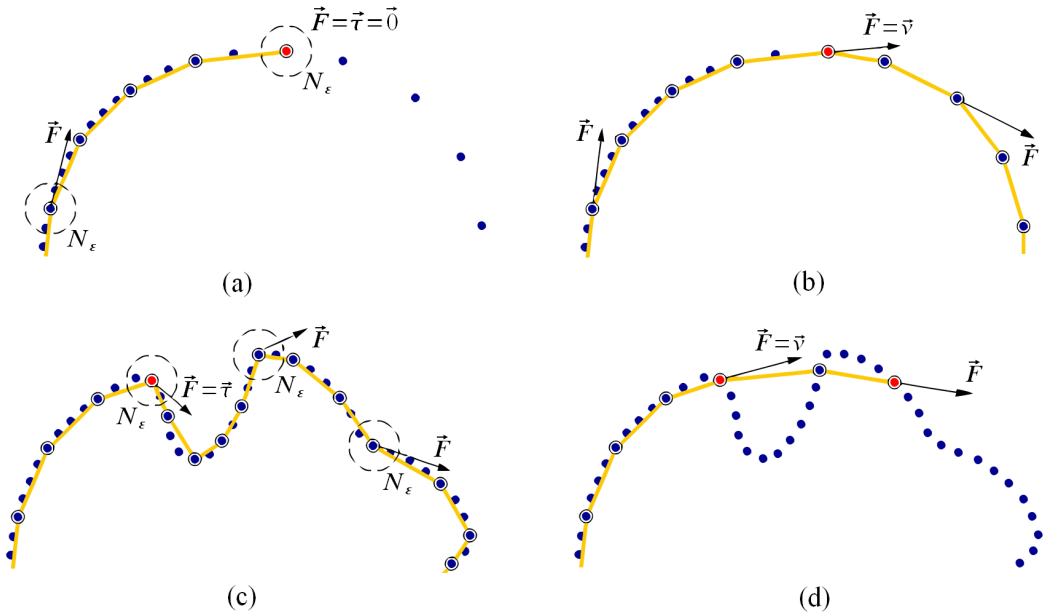


Figure 4.1.: 2D illustration of contour propagation. Examples of correct results: b) inertial propagation on sparse data, $\alpha = 0$ ; c) tangential propagation on dense and complex data, $\alpha = 1$ . Examples of invalid results: a) process interruption on sparse data with tangential propagation, $\alpha = 1$ ; d) coarse reconstruction on dense and complex data with inertial propagation, $\alpha = 0$.

growing procedure to rely on the inertial propagation in the regions with the low point presence and to utilize local object information in the areas with the high point density. It should be noticed that for constant saturation factor $\alpha = 0$ the propagation function (4.1) is similar to [56]. Considering time discretion the contour propagation process is approximated by the explicit iterative equation:

$$\begin{gathered} S(x,t_{i+1}) = S(x,t_i) + \vec{F}(x,t_i)\Delta_{t_i}, \quad S(x,t_0) = S_0(x), i \in \mathbb{N}, \\ \vec{F}(x,t_i) = (1-\alpha_\varepsilon(x,t_i))\vec{\nu}(x,t_i) + \alpha_\varepsilon(x,t_i)\vec{\tau}(x,t_i). \end{gathered} \tag{4.4}$$

In practise, at each iteration $t_i$ the contour $S(x,t_i)$ , $x \in [0,1)$ , is approximated by a sequence of linear segments $\{S_{x_k,t_i}\}_{k=1}^{N(t_i)}$ and the reconstructed object surface is represented by a triangle mesh. During the propagation an approximation of the contour expansion process (4.4) is performed. Each active contour segment $S_{x_k,t_i} = [P_{x_k,t_i}, P_{x_{k+1},t_i}]$ produces a triangle $T_{x_k,t_i} = (P_{x_k,t_i}, P_{x_{k+1},t_i}, P_{y,t_{i+1}})$ where:

$$P_{y,t_{i+1}} = arg \min_{\substack{d(P,R)<\Delta_p \\ P \in P_u^*(i)}} d(P,R),$$

$$R = \frac{1}{2}(P_{x_k,t_i} + P_{x_{k+1},t_i}) + \vec{F}(\frac{x_k + x_{k+1}}{2}, t_i).$$

Here $P_u^*(i) = P_u \cup \limits_{\substack{j=\overline{1,i} \\ k=\overline{1,N(j)}}} \{P_{x_k,t_j}\}$ is the set of all unprocessed points from $P$ and all nodes from the already built mesh at the iteration $t_i$. The parameter $\Delta_p$ defines the

maximal allowed distance between the propagated point $R$ and its projection on the point cloud.

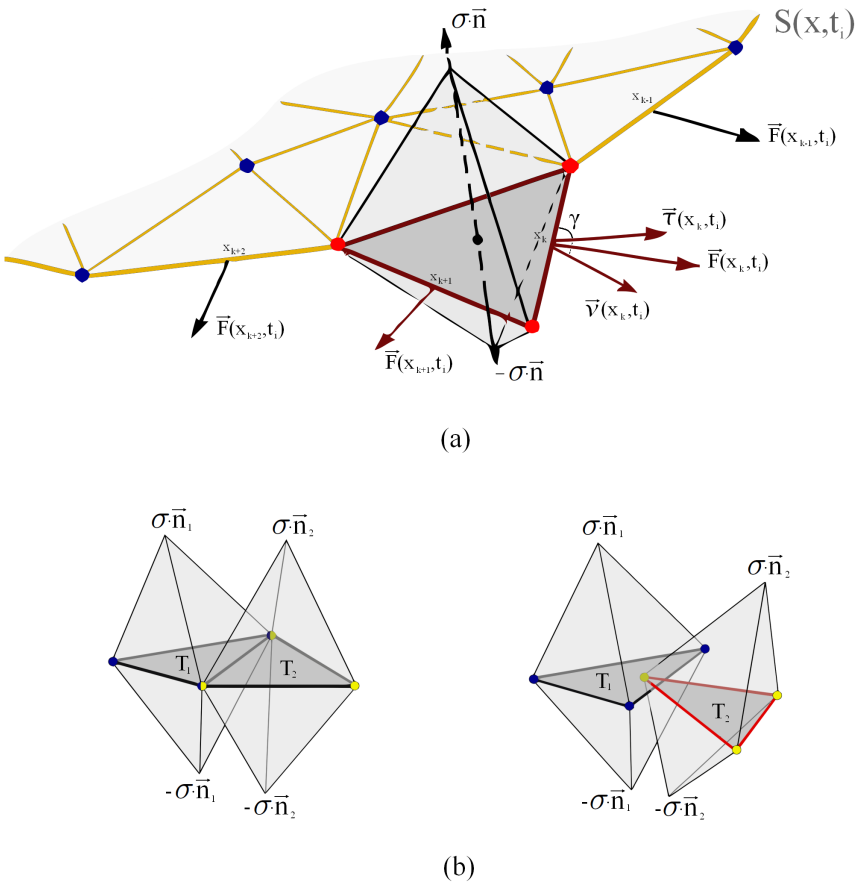


Figure 4.2.: a)Surface $S(x, t_i)$ propagation scheme at iteration $t_i$. The parameter $\sigma$ defines thickness of the surface and the height of the dipyramid-shaped neighbourhood; b) Dipyramid-shaped neighbourhoods of triangles $T_1$ and $T_2$ : no collisions (left), collision (right).

The vectors $\vec{\nu}$ and $\vec{\tau}$ are defined as follows. The vector $\vec{\nu}$ is parallel to the tangential plane of the parent triangle for the segment $S_{x_k,t_i}$ (the triangle that contains the edge $S_{x_k,t_i}$). The vector $\vec{\tau}$ is parallel to the local tangential plane of the point cloud. The angle $\gamma$ between these vectors and the corresponding contour segment $S_{x_k,t_i}$ (Figure 4.2a) can be set as the constant $\pi/2$ (the normal of $S_{x_k,t_i}$) or be equal to the angle between $S_{x_k,t_i}$ and the prolongation of the median line of the edge $S_{x_k,t_i}$ in its parent triangle. The lengths of the vectors can be either constant, that provides uniform propagation speed, or depend on the properties of the constructed contour (e.g. the curvature). In our experiments we used the constant vector lengths $\Delta_{min}$ and the angle $\gamma$ defined by the median lines to the corresponding segments. This allows the algorithm to construct relatively regular meshes with well balanced triangles avoiding expensive computations.

After finishing the process of the triangle creation, the edge $S_{x_k,t_i}$ becomes inactive

and a series of tests is performed. First, the new triangle $T_{x_k,t_i}$ is inspected for collisions with the current mesh. This is carried out by searching intersections of the triangle in some neighbourhood with the mesh triangles. If a collision is detected, the triangle $T_{x_k,t_i}$ is rejected. We propose to use the dipyramid-shaped neighbourhoods to detect the intersections (Figure 4.2b). The test of two dipyramids for collision is performed using the *Separating plane theorem* [62]. Second, the new contour edges are tested for possible occurrence of irregular acute-angled triangles. Such triangles can be eliminated by stitching adjacent contour segments and attaching new triangle vertices to the existent mesh nodes. This helps to avoid appearance of holes, mesh artefacts and collapsed triangles in the reconstructed surface mesh.

The points whose orthogonal projections onto the tangential plane of $T_{x_k,t_i}$ lie inside this triangle are marked as processed if their distance to the plane is less than the predefined value $\sigma$. After the accomplishment of these tests the triangle $T_{x_k,t_i}$ is inserted into the mesh and its edges $(P_{x_k,t_i}, P_{y,t_{i+1}})$ and $(P_{y,t_{i+1}}, P_{x_{k+1},t_i})$ are marked as active within the new contour.

The initialization and mesh-growing steps are repeated until all points from $P$ are processed.

### 4.3.3. Post-processing

The goal of the post-processing step is to improve the aspect ratio of the mesh triangles and to make the mesh more realistic by determining the amount of smoothing applied to the mesh. We demand that the mesh post-processing procedure satisfies three conditions:

a) smoothes the mesh;

b) keeps the minimal spatial deviation of the optimized mesh from the initial one (it is especially important for medical applications where local anatomical features have to be preserved after the mesh processing);

c) can be applied in real time (i.e. to be highly productive).

The aspect ratio is optimized by performing the edge swapping procedure. It purposes at turning the average valence of the mesh nodes to $\nu = 6$ by minimizing the term $\sum_{P \in \boldsymbol{P}_{mesh}} (valence(P) - \nu)^2$ over the set of all mesh nodes [56].

After this, a mesh enhancement procedure is performed. Since the goal of the post-processing step is precise fitting of the smoothed surface to the initial mesh nodes, high performance and simplicity, it was proposed to use interpolating subdivision based on the *modified Butterfly scheme* [209]. This method is computationally effective, supports irregular triangle meshes and produce sufficient smoothing effect.

## 4.4. Evaluation of the method

Evaluation of the method has been done by calculating the average Euclidean distance between the original point clouds and the corresponding reconstructed meshes. The distance calculation procedure is based on a consequent search of the closest mesh face (and the corresponding closest point on it) for each point from the given point cloud. This is performed using the method based on the kd-tree search and the maximal shape radius of the mesh faces, described in Appendix A. Due to the properties of the proposed surface reconstruction method, the resulted meshes are relatively regular and their maximal shape radius can be either defined within the reconstruction algorithm as an additional restriction for the propagation vector and the projection operation, or can be calculated directly.

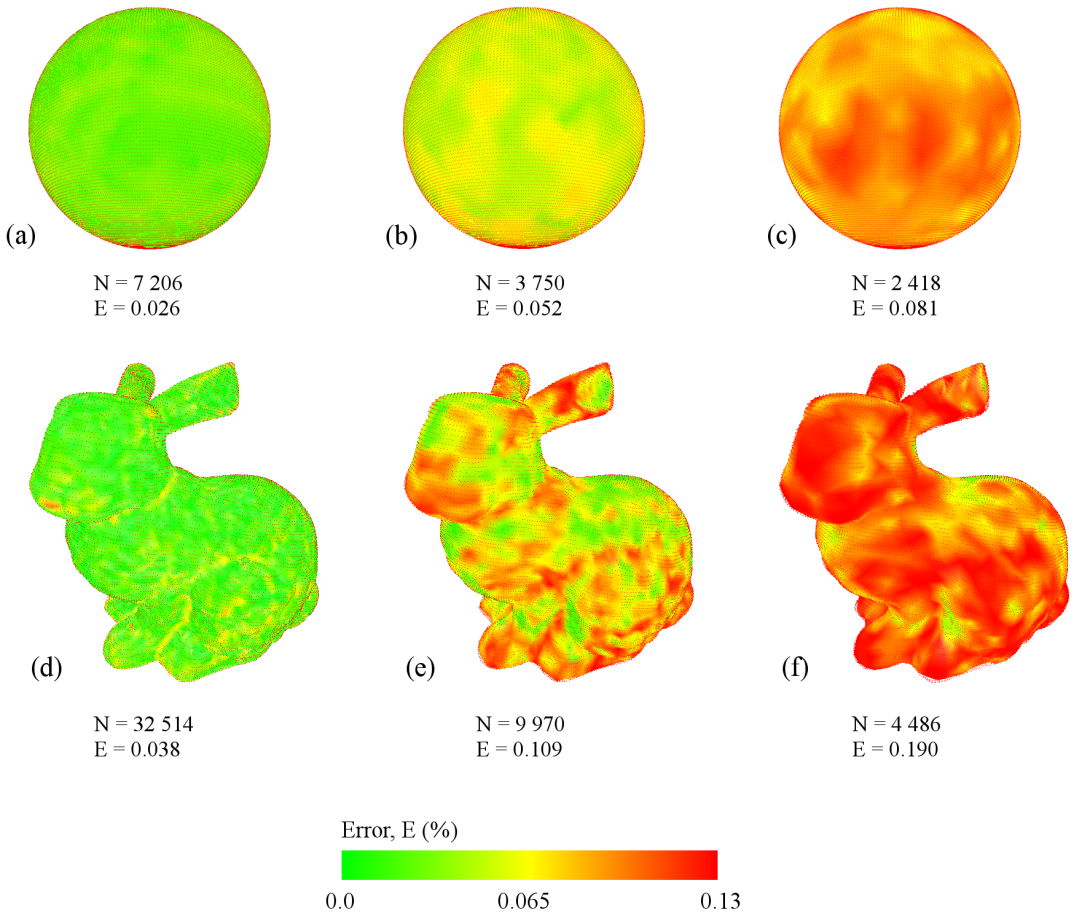


Figure 4.3.: Mesh reconstruction error (E, % of the model's shape diameter) depending on the number of faces (N). The sphere model (a)-(c): 65160 vertices, the Stanford bunny model (d)-(f): 35947 vertices. Reconstruction time (T, seconds): a) T = 8.8; b) T = 6.3; c) T = 6.2; d) T = 8.5; e) T = 3.2; f) T=3.1.

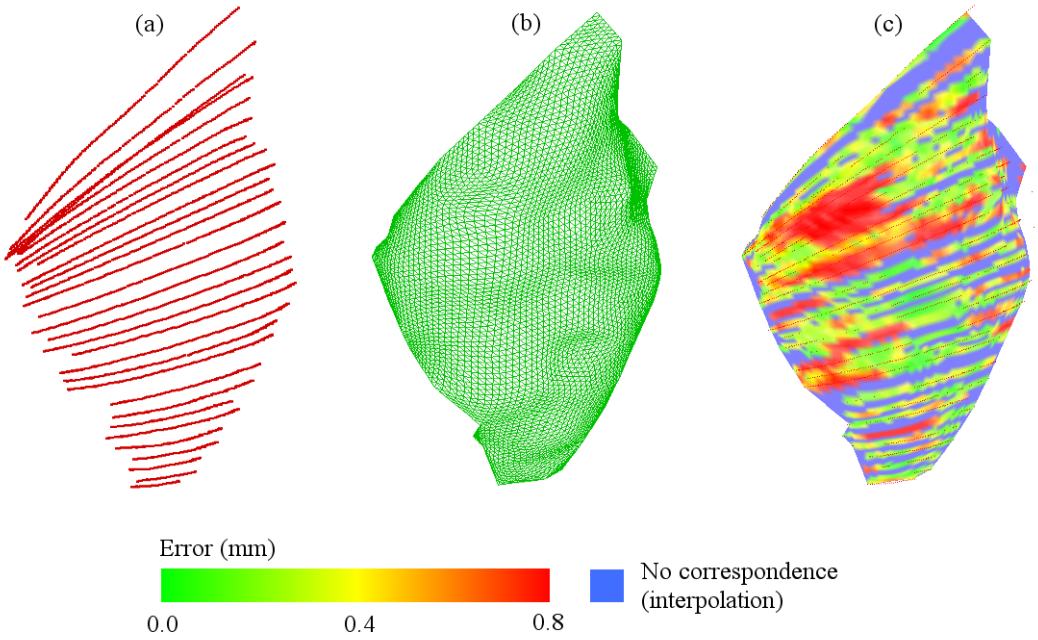


Figure 4.4.: Mesh reconstruction from 3D freehand ultrasound data. A fragment of a swine cadaver myocardium: a) region of the left ventricle, 3911 points; b) reconstructed mesh, N = 8128, T = 0.44s; c) reconstruction error map, E = 0.68mm.

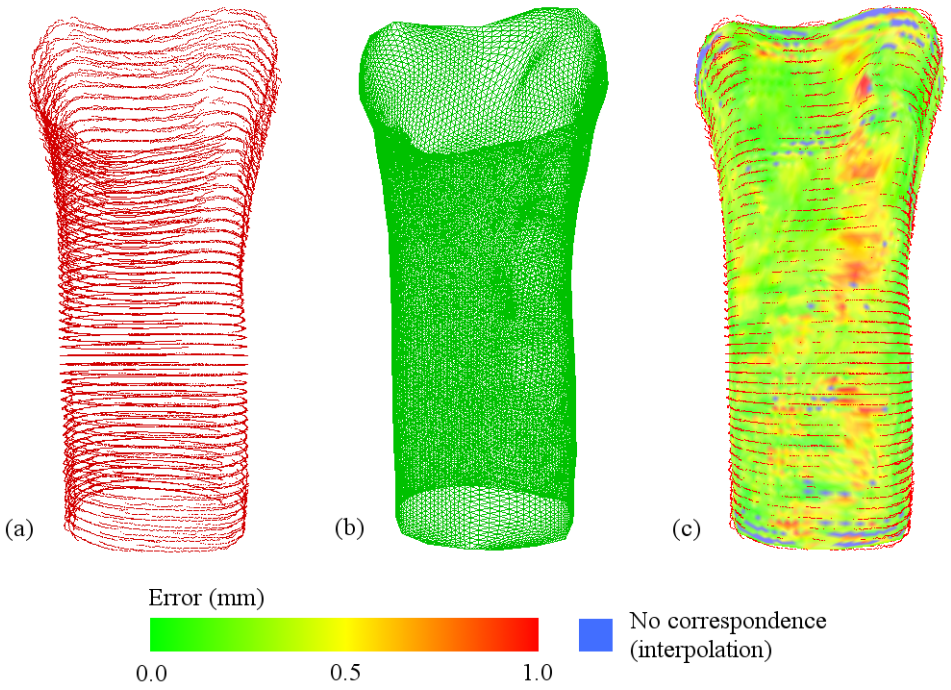


Figure 4.5.: Mesh reconstruction from 3D sequential triggered ultrasound data. A distal tibia of a sheep cadaver: a) ultrasound data, 53674 points; b) reconstructed mesh, N = 28992, T = 12.1s; c) reconstruction error map, E = 0.32mm.

In order to provide a quantitative estimation of the reconstruction accuracy for each local region of a mesh, a rating value is assigned for every face of the resulted mesh. These values are calculated as the average distance between the faces and their corresponding points from the original point cloud. The resulted rating values were mapped over a color space and used for visualization of the mesh reconstruction accuracy (Figure 4.3-4.5). The root mean square value ($E$) of the distances from the points to the mesh is used as a general reconstruction accuracy characteristic for the whole mesh model.

A set of tests was carried out on dense artificial data. The pictures of the reconstructed surfaces, the reconstruction time (in seconds) on a workstation Intel Core i7 2,80GHz 4Gb RAM and errors $E$ (in percentage terms of the original model's shape diameter) are given in Figure 4.3.

Further, the proposed method was tested on two sparse medical datasets. First, a segment of the left ventricle of a swine cadaver heart was scanned in a water basin using a freehand tracked 3D ultrasound system [92]. The cardiac ultrasound data (3911 points) consisted of 30 2D non-parallel images in 3D space were reconstructed on the same workstation. The quantity of mesh polygons, the reconstruction time and the root mean square value were estimated as 8128, 0.44 seconds and 0.68 mm respectively (Figure 4.4).

Second, a distal tibia of a sheep cadaver was scanned using the 3D sequential triggered scanner. The ultrasound data were stored in 612 2D images and 53674 points of bone surfaces were automatically extracted (Figure 4.5a). A surface mesh of 28992 triangle polygons was reconstructed on the same workstation in 12.1 seconds (Figure 4.5b,c). The root mean square value was measured as 0.32 mm (Figure 4.5c). In the performed experiments the mesh regularity value $r$ was between 0.5 and 0.7.

At the next step the dependencies of the surface reconstruction accuracy from the saturation factor $\alpha$ (4.3), from the average size of the mesh faces ($S$) and from the neighbourhood radius $\varepsilon$ (4.2) were investigated. A synthetic model with irregular surface point density and curvature was generated (Figure 4.6a) and used in a number of tests together with the earlier utilized medical models. As a result, the tangential-based reconstruction accuracy started to increase with growth of the neighbourhood radius $\varepsilon$ used for the local tangential planes calculation (4.2), but after some model-defined value $\varepsilon^*$ it tended to decrease (Figure 4.7a). For the optimal $\varepsilon^*$, the relation of the reconstruction error $E$ to the saturation factor $\alpha$ for the synthetic model is shown on Figure 4.7b.

The experiments shown that the Complex Propagation scheme provides more reconstruction accuracy than the pure tangential or inertial mesh growing (Figure 4.6b-c). The optimal saturation factor $\alpha^*$ that corresponds to the minimal reconstruction error (Figure 4.7b) can be experimentally found for a given class of models with similar properties (e.g. similar shape, point density, surface curvature and complexity). If $\alpha^*$ is however unknown the adaptive saturation scheme (4.3) can be used. In spite of the fact that this scheme does not provide maximal reconstruction accuracy it results in the reconstruction error of the same order as the average error on $\alpha \in [0, 1]$ (Figure 4.7c).

## 4.5. Results and discussion

The proposed surface reconstruction method adaptively combines the inertial and the tangential propagation directions (4.1) that makes the mesh-growing process more robust to irregular and sparse data. A series of experiments on synthetic and ultrasound data showed quite a good accuracy of surface reconstruction. The reconstruction error was measured as the averaged distance between the faces of the mesh and the points from the cloud. The reconstructed surfaces correctly interpolate the initial data and demonstrate an appropriate smoothness.

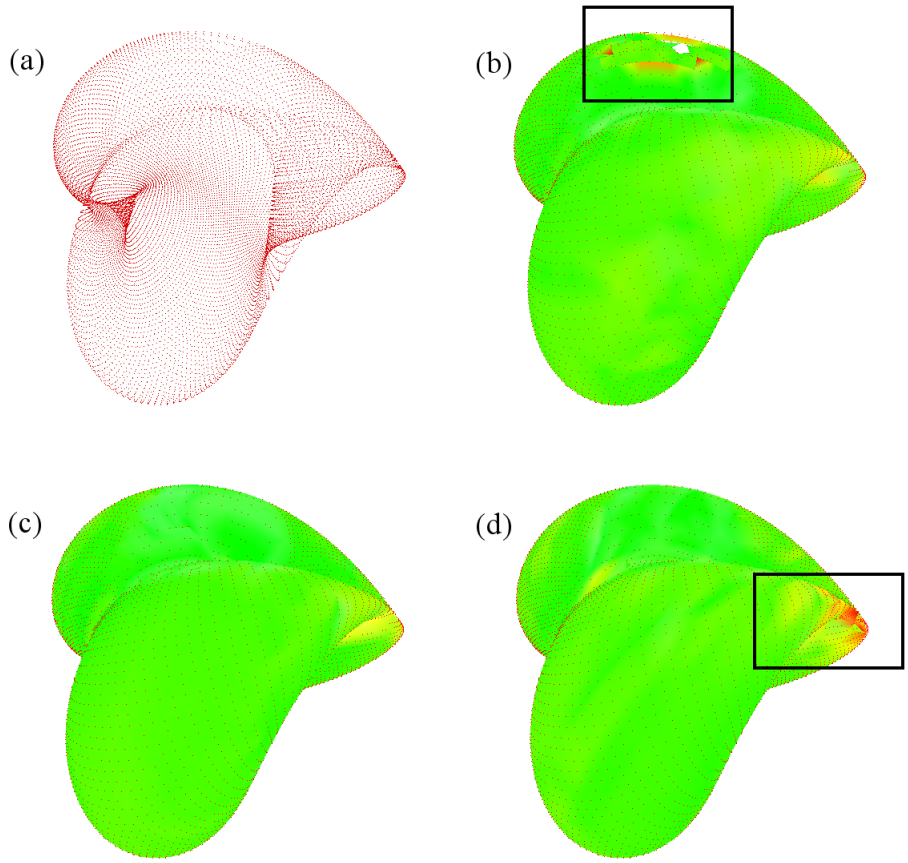


Figure 4.6.: a) Synthetic model: deformed sphere of non-constant radius $R(\alpha, \beta) = R(1 + 0.3cos(2\alpha + 2\beta))$ with low point density in the upper part ($\alpha, \beta \in [0°, 90°]$) and high curvature in the right part; b) tangential propagation-based surface reconstruction ($\alpha = 1$), $E = 0.5$ with an artefact (in the frame); c) adaptive saturation-based surface reconstruction, $E = 0.22$; d) inertial propagation-based surface reconstruction ($\alpha = 0$), $E = 0.32$ with an artefact (in the frame).

It was found that for some models the optimal linear combination of the propagation terms differs from the trivial ones ($\alpha = 0$ and $\alpha = 1$). The tangential surface growing ($\alpha = 1$) may produce artefacts within areas with insufficient point density (Figure 4.6b), while the inertial surface growing ($\alpha = 0$) is suboptimal on complex surface regions

(Figure 4.6d). It was found that a linear combination of the considered terms can overcome these problems and reduce the reconstruction error. Nevertheless, the optimal value of the linear parameter $\alpha^*$ has to be experimentally evaluated for a processed class of models that can be a limitation of the proposed approach. Alternatively the adaptive saturation scheme (4.3) can be used. It seems to provide the reconstruction accuracy comparable with the average accuracy on the set $\alpha \in [0, 1]$. It also should be noted that the accuracy of reconstruction highly depends on the size of the triangles used within the growing procedure. In some cases, due to suboptimal selection of the triangle size (as well as the parameters $\varepsilon$ and $\sigma$), the method can produce holes or mesh oversimplifications. That is another drawback of the method.

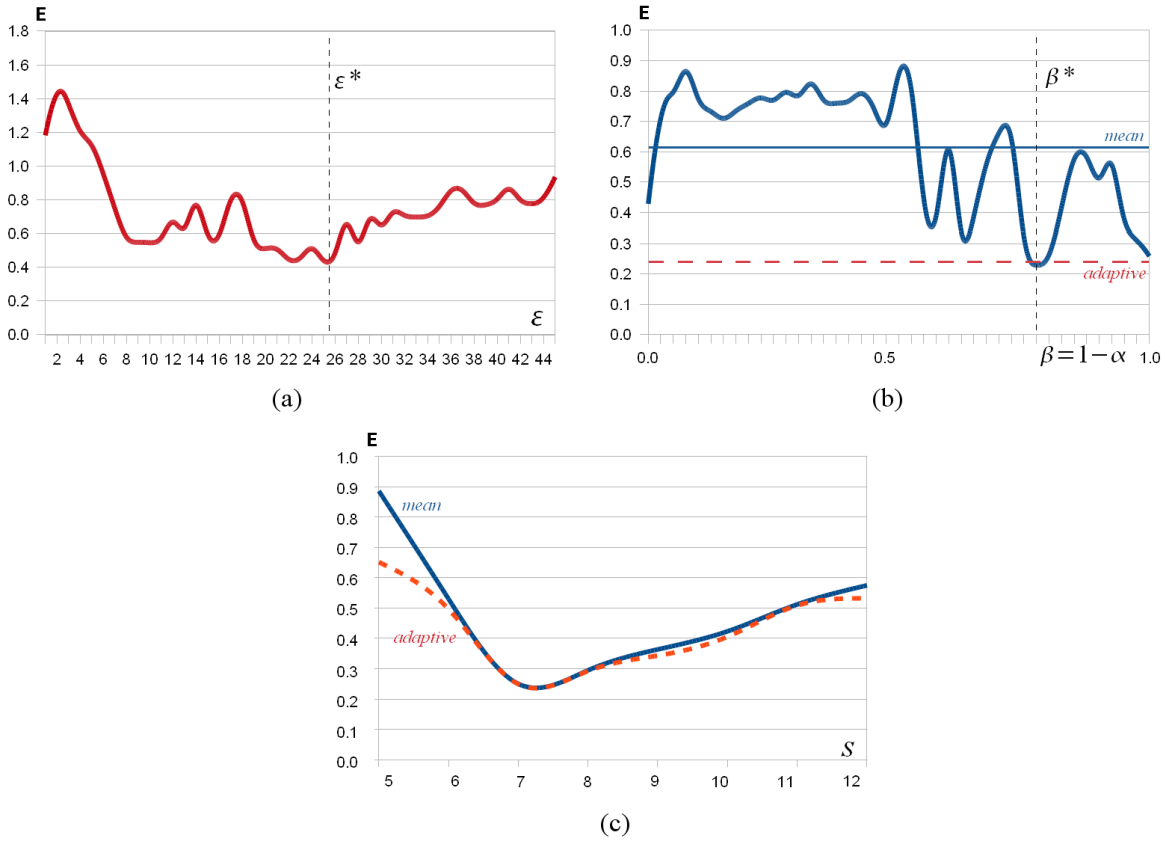


Figure 4.7.: Synthetic model. Relation of the reconstruction error $E$ to: a) the neighbourhood radius $\varepsilon$ for the tangential propagation ($\alpha = 1$); b) the saturation factor ($\beta = 1-\alpha$), the mean value level and adaptive propagation error level ($\lambda_\varepsilon = 1200$) are 0.61 and 0.22 correspondingly; c) the average size $S$ of the mesh faces (adaptive saturation error and the average error on $\alpha \in [0, 1]$).

An additional interpolating surface subdivision procedure is integrated into the post-processing step of the algorithm that guarantees smoother surface geometry. The implementation of the algorithm utilizes kd-trees that results in sufficient acceleration of the processing speed. The method is quite intuitive, provides good accuracy and can be successfully utilized in the applications aimed at reconstruction of surfaces from irregular and sparse data.

## 4.6. Outlook

As it can be seen in our experiments, the point clouds acquired by the sequential triggered ultrasound scanner with the horizontally oriented transducer (Section 3.4) have a regular structure. The points are located at parallel slices in 3D space and form point stacks with the high in-slice resolution and the low inter-slice one. This information was not explicitly used by the developed mesh-growing surface reconstruction method. Therefore, a more optimal solution for reconstruction of such type of data can be found.

As a possible solution we propose to use a technique based on a combination of a 2D case of the described in this chapter method with the existent algorithms for 3D surface reconstruction from parallel contours [180, 39]. So, the technique consists of two basic steps:

1) Reconstruction of a 2D contour within each B-scan separately using a 2D contour-growing analogue of the complex propagation scheme (4.1);

2) Tiling and stitching. The tiling procedure is applied for every pair of neighbouring parallel contours (we used the algorithm proposed by *Christiansen and Sederberg* [39]). The resulted surface patches are stitching together into one mesh.

After these two steps we obtain a surface approximating the original data (Figure 4.8). A number of experiments showed that this reconstruction is about three times faster that the general 3D mesh-growing method. However, it was found that the resulting mesh has a serious lack of smoothness. Indeed, the proposed two-step approach does not involve any inter-slice smoothness terms. Therefore, a third step must be performed in order to enhance the resulting surface.

3) Mesh smoothing that preserves closeness of the surface to the point cloud.

The standard subdivision surface methods described in Section 4.3.3 seem to be not very helpful in this case due to quite a strong disturbance of the surface smoothness. The standard node relaxation techniques could help, but unfortunately they produce smooth meshes that can significantly deviate from the original point cloud. Therefore, in order to obtain a smooth mesh that approximates the point cloud properly, we propose to compute it using the following energy minimization problem

$$E(X,P) = \gamma E_s(X,P) + (1-\gamma)E_d(X,P) \longrightarrow \min_{X\in\mathbb{R}^{3m}},$$

where vector $X$ is the spatial coordinates of $m$ mesh nodes, vector $P \in \mathbb{R}^{3n}$ is the coordinates of $n$ points from the cloud, $E_s(X,P)$ is the internal energy responsible for the mesh smoothness, and the term $E_d(X,P)$ is responsible for closeness of $X$ to $P$.

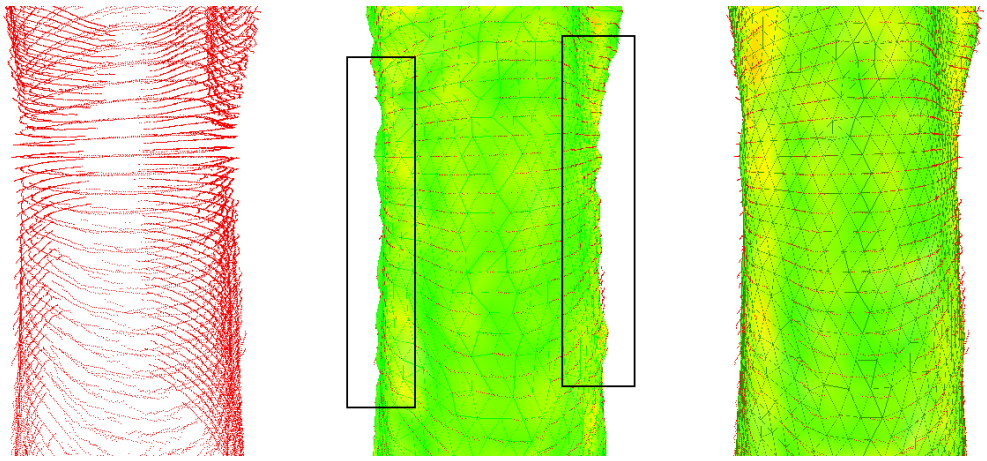

Figure 4.8.: The original bone ultrasound data acquired using the sequential triggered ultrasound scanner (left). The reconstructed surface after two (center) and three steps (right). Term $E_s$ is based on the Laplacian smoothing operator.

Accordingly with this, the first and second steps produce the initial rough mesh that interpolates the point cloud. At the third step the initial mesh is enhanced in such a way that the resulting surface represents a trade-off between smoothness and closeness to the original data defined by the external parameter $\gamma \in [0; 1]$.

Our preliminary tests showed that such an approach is quite an effective solution for surface reconstruction from noisy parallel slices that can outperform the general 3D mesh-growing approach in speed and quality. The future work will be targeted at improvement of this method and development of alternative ones that consider the specific properties of the input data.

# 5. Image segmentation

## 5.1. State of the art

The goal of image segmentation is to identify homogeneous regions on an image depending on pixel intensities, their mutual location and a number of derivative properties such as object shape, texture, distribution of intensities, etc. There are three principal approaches to the concerned problem: *manual*, *automatic* and *semi-automatic* image segmentation. The first approach has been the prevalent standard for many years in clinical research and is used for complex anatomical structures segmentation. Unfortunately, manual segmentation has a number of disadvantages such as human errors, high time consumption and limited adaptation to multi-dimensional (3D and 4D) data. On the other hand, automatic segmentation methods process multi-dimensional data in relatively short time, but usually have a number of problems with quality and, in the long run, need an expert intervention into the final results or into the initialization (that is called semi-automatic methods). In this section we discuss the existent automatic and semi-automatic image segmentation methods.

Automatic and semi-automatic medical image segmentation has been intensively studied by image analysis community since 1960's. As the result a wide variety of techniques has been developed till today. The existent image segmentation methods can be roughly divided into the next four classes: *clustering-based*, *region-based*, *deformable models*, and *active shape models*. Along with this, a certain part of them can be merged into the groups of methods that make use of the variational framework [174] and the probabilistic graphical models [9].

**Clustering-based methods**

Clustering [4] is the principal statistical data analysis technique of unsupervised learning that is widely used in image analysis and pattern recognition. The goal of clustering is partitioning a data set into subsets of clusters based on some distance measure. Clustering differs from data classification (that is also used in image analysis) in one basic feature: clustering does not use training sets for learning the algorithm of classification. Clustering-based algorithms utilized for image segmentation are represented by such well-known techniques as *k-means* [86] and *fuzzy c-means, FCM* [23]. In their standard form these algorithms are not widely applied in medical segmentation tasks, however they can be used in combinations with more advanced image segmentation techniques. As an example, k-means and FCM are used within a spatio-temporal model for segmentation of cardiac ECG-gated MR image sequences [67], or within the Gibbs random field model [112].

Recently a more effective clustering-based image segmentation technique has been

proposed, called *spectral clustering* [178, 11]. The method utilizes the similarity matrix of the data $S = (s_{ij})$, that represents a measure of the similarity between each pair of pixels $i$ and $j$ on the image. Spectral clustering makes use of the eigenvalues of the similarity matrix that allows reducing dimensionality of the clustering problem and therefore increase the speed and effectiveness of the algorithm.

### Region-based methods

Region-based class includes a number of methods that are well studied and widely used in medical image segmentation. The main feature of the region-based methods consists in utilization of global information from the image regions (e.g. statistics, textures, boundaries) for pixel labelling, region splitting or merging [176]. The most popular methods applied in medical image segmentation are based on *region growing* [42, 85], *Bayesian* and *Markov random field, (MRF)* based models [75, 133, 34].

Region growing algorithms start segmentation from the labelled seed pixels (voxels) that are selected in image $I$ manually or automatically inside the regions of interest. The goal of a region growing algorithm is to expand from a seed pixel and to label all homogeneous regions. In order to perform this task a pixel similarity measure $P(i, j)$, $i, j \in I$ and a threshold $T$ are selected. For already labelled pixels $i \in I$ the algorithm recursively marks new pixels $j \in N(i)$ with the same label if $P(i, j) > T$. Here $N(i)$ denotes the neighbouring set for the pixel $i \in I$. Nowadays the research in the region growing medical image segmentation is aimed at construction of new optimal growing criteria (the measure $P$ is becoming more and more sophisticated and depends on a variety of parameters) and at acceleration of the algorithm [155].

MRF based image segmentation [133] forms a big class of probabilistic methods that treat the image as a realization of a random field and utilize mutual dependence of the pixels (voxels) in neighbourhoods. This approach is widely used for medical image segmentation since 1980's and till today it has become a variety of modifications and improvements. MRF based methods are considered to be edge preserving and allow saving global boundary and shape properties of the segmented objects. However the process of calculation maximum a posterior (MAP) estimation of the image regions is computationally expensive. Several solutions for this problem were proposed. Geman and Geman [75] profited by the *Hammersley-Clifford theorem* and proposed to use the Gibbs distribution:

$$P\{X\} = \tfrac{1}{Z} e^{-U(X)/T},$$

for image restoration within Bayesian framework. Here,

$$Z = \sum_{z \in \mathbb{X}} e^{-U(z)/t},$$

denotes a normalizing factor, $\mathbb{X}$ is the set of all possible images with MRF property in 3D, where

$$U(X) = \sum_{C \in C} V_C(X),$$

models the energy function of the image $X$. The value $C$ is the set of all cliques defined on the image $X$ and "temperature" $T$ is responsible for degree of "peaking" of $P\{X\}$ [34].

Use of the Gibbs distribution within the random field models allows reducing the MAP estimation problem to the problem of minimization of an energy function. Nevertheless, the methods based on the described by Geman and Geman [75] model did not use any boundary information for qualitative objects segmentation and edge preservation. Therefore, Chen and Metaxas [34] proposed a new energy function for the Gibbs prior model that integrates objects boundary information and provides a better segmentation quality. They used the iterated conditional modes (ICM) algorithm [22] for energy minimization, but this approach seems to be rather time consuming and implements a greedy strategy. As a result, a lot of works were dedicated to the problem of optimal computation of the MRF MAP estimation, that is a NP-hard problem. A number of interesting solutions were introduced: modified metropolis dynamics [109], mean field annealing [207], modified simulated annealing [139, 113], methods based on the max-sum problem and duality theory of linear programming [198]. Several authors [192] use Markov chain Monte Carlo (MCMC) method [149] to solve the labelling problem. The idea of MCMC is based on constructing a Markov chain with the desired distribution as its equilibrium distribution. The desired distribution represents parameters of the model that is under consideration and can be obtained from the state of the chain after a large number of steps. The error of the approach is proportional to the number of the steps.

The original methods described by Geman and Geman [75], Chen and Metaxas [34] are mainly intended for processing of 2D data. However, later works in this field of research [104, 38] extended the MRF based image segmentation to 3D. Thus, now this approach is successfully applied in 2D/3D medical tasks [139, 164].

In our earlier paper [90] we proposed to combine the 3D MRF model with the high-order statistical gradients [153] for cardiac MRI image segmentation. The statistical gradients were introduced into the energy function $U(X)$ as an additional term in order to exploit the image regions with hidden boundaries which look like random textures. Although utilization of the high-order statistics allows improving segmentation of MRI brain images [153], our experiments on cardiac MRI and ultrasound data did not show significant improvement of the segmentation quality.

### Deformable models

The main idea of the deformable model methods [141] is to deform some elastic body (e.g. curve or surface) to the desired object considering the influence of physical forces and constraints. This class of methods can be divided into several subclasses such as active contour models (ACM) [108], dynamic deformable models (DDM) [140] and deformable templates [167].

Active contour models also known as "snakes" are a popular approach of image segmentation that was firstly proposed by Kass in 1987 [108]. A snake is a geometrical contour parametrically defined on image $I$,

$$s(r) = (x(r), y(r)) \subset I, r \in \mathbb{R}.$$

The shape of the contour $s(r)$ is defined by the energy function

$$E(s) = E_{\text{int}}(s) + E_{ext}(s).$$

The term $E_{\text{int}}(s)$ models the internal deformation energy of the snake while $E_{ext}(s)$ models the external energy. The goal of the method is to minimize the energy over all contours defined on the image. According to the calculus of variations the optimal contour can be found using the Euler-Lagrange equation [141].

There are a lot of various implementations of this idea in medical image segmentation, such as active contours without edges [31], stochastic active contours [154] and minimal path deformation contour [202]. But in most cases these methods require a user pre-initialization and the segmentation quality strongly depends on it. Nevertheless new fully automated deformable model techniques have been recently developed [110, 145, 152].

It should be also noticed that the deformable models utilize only generic a-priori information about the shape of the segmented organ. This can be suboptimal in the case when the shape of the object can be approximated by some geometrical rule. Therefore, a modification of the deformable models called deformable templates was introduced for medical image segmentation [167].

Spatio-temporal segmentation of some biological tissues such as the heart is an important matter for diseases diagnostics. Therefore the deformable models have been extended to 4D and produced the the class of dynamic deformable models [140, 105, 208]. The main idea of this approach is to unify the shape and motion in one process that can be modelled using the principles of the Lagrangian mechanics. In this multidimensional case the snake can be defined as a dynamic contour $s(r,t) = (x(r,t), y(r,t))$, $r \in \mathbb{R}$, $t \in [0,T]$, and can be calculated using the Lagrange partial differential equations.

### Active shape models

Active shape models (ASM) for locating structures in medical images were firstly proposed by Cootes et al. in 1993 [40]. ASM represent a statistical model of object shapes which can be iteratively deformed and merged with the desired object in order to segment it. The degree of the deformations is controlled by a statistical shape model (SSM) that allows ASM to vary with the constrained freedom defined by a training set of objects. The approach is rather similar to ACM with the additional global shape constraints. In order to define these constraints a point distribution model based on landmark points is constructed. Let $\mathbb{X} = \{X_1, X_2, ..., X_n\}$ be a set of $n$ aligned shapes in 3D. Each shape consists of $m$ points and is represented as a vector in $\mathbb{R}^{3m}$, $X_i = (x_{ij})_{j=1}^{3m}$, $i = \overline{1,n}$. The mean shape $\bar{X}$ of the set is defined by the expression

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$$

In order to define the modes of variations, a $3m \times 3m$ covariance matrix $\Sigma$ is calculated,

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^T(X_i - \bar{X}).$$

Let $\{v_1, v_2, ..., v_{3m}\}$ be the eigenvectors of $\Sigma$ with the corresponding eigenvalues $\{\lambda_1, \lambda_2, ..., \lambda_{3m}\}$. Then the variation of the statistical models in the process of segmentation is represented by the equation

$$X = \bar{X} + V^T b,$$

where $V = [v_1, ..., v_{3m}]$ is the matrix of $3m$ eigenvectors and $b = (b_1, ..., b_{3m})$ is the vector of eigenvector weights. Thus, varying the parameter $b$ during the segmentation procedure, new optimal models can be generated. These models are shape-constrained and possess basic shape properties similar to the ones of the models from the training set $\mathbb{X}$. The variations along the eigenvectors with the largest modules of the eigenvalues are of the biggest interest here, as they represent the most significant statistical shapes of the model set.

Due to its flexibility and the ability to inherit the shape of the training set, the ASM technique is well applicable for medical image segmentation tasks [41, 193]. Recently two modification of this technique have been developed: active appearance models (AAM) [69] and active feature models (AFM) [119]. The AAM is the extension of the ASM based on both statistical shapes and local texture variations. So, the AAM method (also known as "smart snakes") combines the properties of both ASM and ACM and can improve the quality of automatic image segmentation. The AFM methods are quite similar to AAM and are based on combination of statistical shapes models and local texture descriptors for texture representations. The AFM approach has advantages with complex image data (e.g. anatomical structures) with high texture variation. In this case it AFM can improve accuracy and speed of the AAM method. One more modification of the snakes model, called "diffusion snakes", was earlier proposed in the literature [45]. It introduces non-linear shape statistics into the Mumford-Shah functional [148] and provides additional robustness against noise, clutter and occlusions.

A popular approach to image segmentation is related to utilization of variational models, and consists in optimization of some functional that defines the required segmentation of an image. The recent research in this field is represented by such directions as convex approximations of the popular variational models (e.g. Mumford-Shah [173], Chan-Vese [65], based on the Gromov-Wasserstein distance [172]), total-variation-like regularizers [126], level sets [87, 44] and integration of statistical shape priors into the variational models [43, 27].

## 5.2. Segmentation of ultrasound medical images

Practically, all described above methods can be used for segmentation of medical ultrasound data. With varying degrees of success different implementations of deformable models (ACM) [116], region-growing [85] (including morphological- and watershed-based methods [132, 71]), active shape models (ASM) [94] and clustering [11] have been ap-

plied to this problem. A number of specialized techniques based on probabilistic models including Bayesian frameworks [103], Kalman filtering [1] and Monte-Carlo methods [10] have been also proposed. Some authors make use of grey level distributions and the well-known methods based on intensity derivatives and edge detectors (e.g. Sobel operator, Canny filter, etc.). However, it was shown in [147] that the local image phase is more robust for acoustic boundary detection than the intensity gradient. Therefore local phase-based methods, e.g. [83, 163], are more preferable for this task.

On the one side, specialized methods provide better segmentation quality than the standard edge detection and morphological techniques. On the other side, the computation time is an important aspect that must be considered during selection of an appropriate image segmentation algorithm. Majority of sophisticated algorithms can work only in offline mode due to the huge amount of computations. Therefore, a trade-off between the quality and the computation time must be found. In order to perform ultrasound contour extraction in real-time mode, in our applications we used the *Rainfall algorithm* [71, 115, 134]. It is based on a modified watershed segmentation technique and provides acceptable contour extraction accuracy and sufficient speed.

Regardless of the used mathematical model, an image segmentation method applied to the ultrasound data must consider a series of specific features of ultrasound modality [103]. The method must not completely rely on the image intensity but must filter the image in order to emphasize the needed features. It should be taken into account that the surface response can be quite thick (up to 2 - 4 mm). The method should involve such ultrasonic features as tissue shadows, reflections and incidence angle. In the case of bone ultrasound it should be considered that the bone surface is highly specular. On an image it is represented by a high intensity feature followed by a shadow feature. The high intensity feature looks like a curve with the shape closely (but not completely) resembling the bone surface and having the thickness up to several millimetres. It was shown in [103] that the actual surface on an image is close to the highest gradient and the highest intensity points. Since there are inherent uncertainties in resolving the surface location from a single B-mode image, it is preferable in tracked 3D ultrasound to capture the likelihood of the surface occurrence from each 2D image and to combine these 2D likelihoods into a superior 3D likelihood of the surface. All the mentioned above facts show that development of a correct segmentation method for ultrasound data is a challenging multivariable problem.

# 6. Rotation-scale invariant object detection based on the Dual-Point GHT

## 6.1. State of the art

The main task of *object detection* is to find instances of a specific object or a set of objects in a generic image, that can be represented by a 2D image, a video sequence, a 3D scene or a point cloud. Commonly this process is getting complicated by the image variability (illumination, intensity transformations, noise), object transformations (similarity, affine or non-rigid) and object occlusions. Object detection is closely related to *object recognition* that is aimed at identifying a specific object and its classification among objects of a general class in a particular region of the image [8]. Both object detection and object recognition present one part of a complex computer vision problem that is aimed at development of an artificial system similar to the biological system eye-brain [175, 165], so they often utilize the common feature extraction and learning algorithms. From the point of view of computer vision, object recognition follows object detection in the visual processing pipeline and is applied to already detected objects for further identification. From another point of view, the modern implementations of object recognition methods often integrate the detection step and are able to identify the objects from unprocessed images using a huge database of the object classes. Moreover, as it was mentioned in [125] both in computer vision and in human vision, *object recognition* and *image segmentation* are heavily intertwined processes. The knowledge from recognition is often used for guiding the segmentation process and, oppositely, the use of top-down segmentation usually improves the recognition results. Due to the close relation of these two processes the same techniques (e.g. deformable and active shape models) are often utilized in both segmentation and object recognition tasks.

Object detection methods were comprehensively reviewed by Amit in [8]. Accordingly to this work the object detection methods can be classified into three main classes: deformable-template models, searching correspondence space, and searching pose space.

**Deformable-template models**

The methods from this class [8] are commonly based on relaxation techniques for maximizing posterior distributions. They are well suited for detecting non-rigid objects (particularly in biological and medical images), but all suffer from a similar drawback: location and scale must be roughly known, that means that initialization provided by

the user is necessary. These methods are closely related to the ones utilized in image segmentation and usually make use of the variational models.

**Searching correspondence space**

The methods from this class [80] search for arrangements of some local features (SIFT [138], SURF [17], histogram descriptors [136], etc.) in the processed images consistent with the arrangements of the same features in the template model. For example, when the relationship between a pair of image features and a pair of template model features is considered, the method provides invariance of detection to object pose. For the correspondence search the probabilistic graphical models [171, 20] are often utilized.

**Searching pose space**

This class is presented by the variety of methods based on the Hough Transform (HT) [96] or pose clustering [183]. To this class we also attribute the methods based on the divide-and-conquer search in the pose space [117], and implicit shape models [124, 125] that combine local-feature codebooks and probabilistic Hough voting into one process performing image segmentation and object detection tasks. Having such advantages as noise resistance, tolerance to boundary gaps and robustness to objects overlapping, the methods from this class give rise to complications induced by high dimensionality and memory consumption. Particularly, it is essential for the problem of arbitrary shapes matching invariantly to some transform (e.g. affine, projective) in two or thee dimensions that widely occurs in medical and industrial applications.

### 6.1.1. Hough Transform based object detection

Originally the Hough Transform was proposed by Paul Hough in 1962 to extract straight lines in the particle tracks recognizing procedure [96]. It was popularized in 1972 image analysis after the work of Duda and Hart [58], who proposed to use the Hough Transform for more general curves fitting and introduced the common rho-theta parametrization for lines representation (Figure 6.1) that was already standard for the Radon Transform [161]. In 1981, the Hough Transform was extended by Ballard to the detection of quadratic curves and to the extraction of general shapes using the Generalized Hough Transform [15].

A great number of optimizations and modifications has been proposed in this area till today. A series of works in HT were dedicated to the problem of accumulator array size reduction, that was particularly important for the approaches worked with high-dimensional parametric spaces [156]. The Fast Hough Transform (FHT) introduced by Li [130] realized the coarse-to-fine strategy of the parameter space exploration, so that only the subsets of parameters which contain more than a specified number of votes were investigated in greater details. Another modification of the HT is the Adaptive Hough Transform (AHT) that implements an iterative multi-resolution peak search in the parameter space using a small fixed size accumulator and the parameter limits calculation procedure [101]. Proposed by Princen et al. [156] the Hierarchical Hough Transform (HHT) works with a pyramidal structure of sub-images extracting short line segments

Figure 6.1.: Example of 2D line detection using the Hough Transform. The original image (top left); pre-processed image after application of the Sobel operator (top right); the standard Hough transformation of the pre-processed image is the set of sinusoids in the rho-theta space (bottom left); lines corresponding to local maximums in the Hough space (bottom right).

and combining them within local neighbourhoods into longer lines. The Randomized Hough Transform (RHT) proposes an iterative stochastic approach to the problem of the HT-based object recognition [200]. The algorithm uses a set of points sampled randomly from the edge image and defines only one point from the parameter space, therefore only one corresponding cell of the accumulator array is incremented. The process continues until a specified threshold in the cell is reached or the number of iterations is exceeded.

Most of the first developed methods were designed to extract a very narrow class of analytic curves such as lines and ellipsoids that was often insufficient for their successful application in real scenes. Hence, a number of later papers [122, 150] concentrated on the development and improvement of techniques for general shape extraction based on the GHT. Originally the GHT invented by Ballard [15] is not invariant to geometric transformations, therefore many improvements and modifications [66, 3, 61] have been proposed to consider a more general shape transformation (e.g. similarity and affine). In [12] it was proposed to used such invariant image features as local Fourier-based descriptors, that made the transform also more robust to noise. A number of works [3, 144, 184, 37] realize so called $n-to-1$ mapping approach, accordingly to which the

image features are selected as $n$-combinations instead of single points. The advantage of this approach is that each $n$-combination of image points produces votes in the parameter space considering some type of geometric invariance (e.g. pairs of points support invariance up to the similarity transform, triplets of points support the affine transform). It is also possible to reduce the dimensionality of the Hough space by using subspaces corresponding to different parameters independently [3], and therefore decrease memory consumption of the algorithm. The drawback of the $n-to-1$ mapping approaches consists in increased computational complexity (exponentially in $n$), redundant mapping and spurious voting, that is the object of the further improvement in this field [37].

The recent approaches on Hough transform-based object recognition are dedicated to the problem of construction an optimal mapping of the image features into the votes in the Hough space, such as the implicit shape model (ISM) [124, 125], Hough forests [68] and the principled implicit shape model (PRISM / Fast PRISM) [123]. These methods operate in the high-dimensional Hough parameter spaces and are optimized for the features that are typical for digital photos, video sequences and real-world scenes. A number of the latest techniques learn the problem of vote processing and accumulation, including probabilistic models [16, 146] and optimal strategies for the Hough space analysis such as intrinsic and minimum-entropy Hough transforms [199]. The common trend here is to reduce the number of votes per feature that will decrease the "uncertainty mist" in the Hough space and produce better accuracy of the object recognition process. The cutting-edge minimum-entropy approach solves this problem by introducing vote weights and considering the Hough transform as a kernel density estimation problem. It performs minimization of the vote distribution entropy in the Hough space with respect to the vote weights using the iterated conditional modes (ICM), that provides the high level of accuracy and speed comparing to other methods [199].

### 6.1.2. Classification of the existent HT-based methods

Summarizing the variety of the existent Hough Transform based methods, we classify them using the following criteria:

a) dimensionality of the object space (2D or 3D) and the Hough space (from 2D to 8D);
b) type of the feature-to-vote conversion function;
c) vote analysis principle in the Hough space.

Since the object space is commonly one of 2D (photo, medical and industrial images) or 3D (real life scenes, medical volumes), the Hough space dimensionality varies from 2D in the primitive object recognition problems (e.g. the original Hough Transform for 2D line matching) to 8D in the most complex ones (3D object matching invariant to translation, scaling and rotation).

Concerning the second criteria, the majority of the methods utilize such image features as pixel/voxel intensities, boundaries gradients, different types of invariant descriptors and their combinations, applying special methods of image segmentation and analysis [124, 125, 68]. The extracted features are mapped into sets of parameters in the Hough space using some feature-to-vote conversion function. Such sets of parameters

receive so called "votes" that can be e.g. accumulated during the whole feature conversion procedure. The parameter with the maximal number of votes define the positions of the detected objects. The used feature-to-vote conversion functions, that can vary in their nature and complexity, define the principal quality of the HT-based methods.

Nevertheless, due to the high dimensionality of the parameter spaces, and as the result, a huge number of discrete elements in them, the direct analysis of the votes (such as vote accumulation and brute force search for maximum) is often impossible. Therefore, a number of optimizations and approximations of the vote analysis process were developed. They all can be divided into the following classes [199]: *standard*, *approximate*, *irregular*, *hierarchical* and *iterative optimization*.

**Standard** or grid approaches use multidimensional arrays for vote accumulation that cover the full Hough space. They were initially proposed in the first HT algorithms and were successfully used for low-dimensional object recognition due to their high speed. But with the growth of the HT dimensionality usage of the grid approaches has become ineffective due to their exponential memory consumption.

**Approximate** approaches use reduced-dimensional approximations of the full Hough space to find the vote peaks. It is done by dividing a high-dimensional Hough space into several subspaces of smaller dimensionality and then analysing these subspaces separately. As an example, a 6D (translation/rotation) space can be reduced into two 3D subspaces (one for translation and one for rotation) with the following peak search in these two subspaces [64]. Together with this, the peak search can be also performed over the translation subspace with the further computation of the average rotation for the found peaks [188]. Due to operation in low-dimensional subspaces these methods are relatively memory undemanding, but give only an approximate solution that can vary in quality.

**Irregular** approaches exploit sparsity of the Hough space by sampling it only in the regions where objects are likely to be detected. The Randomize Hough Transform [200] and the intrinsic Hough transform [199] fall into this category.

**Hierarchical** approaches, such as the Fast [130] and Adaptive [101] Hough Transforms, explore the full space in a coarse-to-fine manner, avoiding precise analysis of the areas with the low total number of votes. Though the complexity of such vote analysis is high, the coarse-to-fine technique allows significantly reducing the memory consumption.

**Iterative optimization** approaches find vote peaks in the full space through optimizing some functional. The mode-seeking approaches (mean shift [205], medoid shift [177], quick shift [195]) that fall into this group need a multiple initialization and are not guaranteed to find every peak. Nevertheless, due to their reduced computational complexity they are successfully applied to high-dimensional Hough spaces. The minimum-entropy approach [199] can overcome the drawback of the multiple initialization but it utilizes a "greedy" strategy in the iterative local optimization that does not in general produce an optimal solution.

As an example, according to the above classification the standard Hough Transform for 2D line detection can be classified in the following way: a) dimensionality of the object space is 2D, dimensionality of the parameter space is 2D; b) feature-to-vote function: object points are mapped into sinusoids in the rho-theta parameter space; c) vote analysis principle: standard (grid-based).

In the following sections we consider the features of the Generalized and the Fast Hough Transforms.

### 6.1.3. Generalized Hough Transform

First, we consider the formal definition of the Generalized Hough Transform and take a look on the relation of the GHT and the RT. This relation as well as elaboration of a conventional mathematical definition of the GHT was an object of discussions for many authors since the last years. A number of authors noted that the GHT and RT are very closely related to each other [49, 194], however several works mentioned disparity of these two transforms [157].

First of all we introduce the basic notations and variables. Let $\mathbb{X} \subset \mathbb{R}^N$ be a $N$-dimensional set of spatial coordinates and $\mathbb{I} = \{I(x) : \mathbb{X} \to \mathbb{R}\}$ is the space of all images defined on $\mathbb{X}$. We consider also $\mathbb{P} \subset \mathbb{R}^M$ as the $M$-dimensional parameter space and the constraint function:

$$C(x, p) : (\mathbb{X}, \mathbb{P}) \to \mathbb{R},$$

that defines a desired template. The template represents a parametric subset of points in the set of spatial coordinates that satisfy the equation $C(x, p) = 0$ for $p \in \mathbb{P}$. Each parameter value $p \in \mathbb{P}$ defines some geometric transformation of the template in space $\mathbb{X}$. The RT in a general form is a mapping from the image and parameter spaces into the set of real numbers:

$$RT_C : (\mathbb{I}, \mathbb{P}) \to \mathbb{R},$$

and may be defined by the formula:

$$RT_C(I, p) = \int_{\mathbb{X}} I(x)\delta(C(x, p))dx, \tag{6.1}$$

where $\delta(x)$ is the *Dirac delta function* and $\delta(C(x, p))$ is the transformation kernel. Originally the RT was defined as the integral of a function over hyperplanes specified by the parameter $p \in \mathbb{P}$.

In the context of the Radon Transform the GHT can be specified as its discretization in the case of binary images, i.e. when $\mathbb{I} = \{I(x) : \mathbb{X} \to \{0, 1\}\}$. However, such a theoretical interpretation first given by Deans (1981) does not explicitly consider the structure of the practical realization of the GHT and the idea of the voting process, proposing so called reading/writing paradigms [194] for better explanation of the difference in calculations of the RT and the GHT. The main feature of the GHT realization

in practice consists in utilizing *the voting mapping* that is a mapping from the set of spatial coordinates $\mathbb{X}$ to the space of all subsets of the parameter space[1]:

$$H_C : \mathbb{X} \to \mathbb{P}^*,\ \mathbb{P}^* = \{P^*, P^* \subseteq \mathbb{P}\}.$$

We define the voting mapping by the formula:

$$H_C(x) = \{p \in \mathbb{P}, C(x,p) = 0\}, x \in \mathbb{X}. \tag{6.2}$$

It means that for each point $x \in \mathbb{X}$ the voting mapping defines a set of parameters for which $x$ has voted. Using this definition the GHT may be written in the form:

$$GHT_C(I,p) = \int\limits_{\mathbb{X}} I(x)\delta(\rho(H_C(x),p))dx, \tag{6.3}$$

where $\rho(P^*,p) : \mathbb{P}^*, \mathbb{P} \to \mathbb{R}$ is a distance between parameter $p$ and the subset $P^*$ in the parameter space. In this context the equation (6.3) defines the same transformation as it was specified by (6.1), but this equation view allows us to emphasize the features of the GHT calculation in practice.

The algorithm that realizes the GHT practically works with discrete spaces $\mathbb{X}$ and $\mathbb{P}$. In this case the transformation (6.3) can be written in the form:

$$GHT_C(I,p) = \sum_{\substack{x \in \mathbb{X} \\ I(x)=1}} 1_{p \in H_C(x)}. \tag{6.4}$$

From the equation (6.4) follows that the values of $GHT_C(I,p)$ equal to the number of the votes that are given to the parameter $p \in \mathbb{P}$ by all object points (features) of the image, i.e. $x \in \mathbb{X}$, $I(x) = 1$. Therefore, for optimization of the calculation process we can allocate an accumulator array that stores all the votes of all the object points using only one traversal through the image (the grid voting principle). The further calculation of the value $GHT_C(I,p)$ for any $p \in \mathbb{P}$ is fulfilled in constant time by accessing to the corresponding element of the accumulator array. This exposes the main idea of the standard Hough formalism, that makes the GHT a computation effective, but on the other hand a memory consuming method. It also should be noticed that usage of the GHT is preferable for the tasks where the votes are needed to be calculated for large sets of the parameters (e.g. in the problem of the global maximum search). The discrete RT approach is more optimal in the cases when the vote number is needed to be found for a certain point in the parameter set.

In the above definition we considered the object points (a set of $N$-dimensional spatial coordinates $x \in \mathbb{X}$ with the property $I(x) = 1$) as the only possible features that produce votes in the Hough space. In a more general case the features can have a more complex view and consist of a number of additional characteristics (e.g. gradients, local

[1] It should be noted, that in the most general case $\mathbb{X}$ is considered as the space of image features. In this case the voting mapping is nothing else as the feature-to-vote conversion function mentioned above. For more clearness we suppose here that the image features are presented by the coordinates of the pixels that belong to objects.

descriptors). In this case the set $\mathbb{X}$ includes these additional dimensions and is interpreted as the set of features. The corresponding general form of the voting mapping (6.2) $H_C(x) : \mathbb{X} \to \mathbb{P}^*$, is called *feature-to-vote conversion function.*

The classical GHT algorithm proposed by Ballard [15] in 1981 uses one reference point parametrization for detection of non-parametric objects with arbitrary shapes in 2D. The parameter space in this case is two dimensional and coincides with the image spatial coordinates $\mathbb{X}$. The object recognition procedure is based on two consistent steps: a) conversion of the processed image features into votes, and b) the vote analysis.

**Standard GHT feature-to-vote conversion.** First, the template model encoding is performed. A reference point $p \in \mathbb{X}$ is selected within the template domain according to some rule (e.g. mass center of the template model) and the template model (the shape that must be detected) is represented in so called R-table format (Table 6.1). According to this format every boundary point of the template model $c \in \mathbb{X}$ is encoded as a pair $(r, \alpha)$, where $r$ is the length of the vector connecting $c$ with $p$, and $\alpha$ is the angle between this vector and $OX$ (Figure 6.2). The resulting pair is stored in the R-table in a row, that is defined by the angle $\phi$ between $OX$ and the boundary gradient calculated at the point $c$.

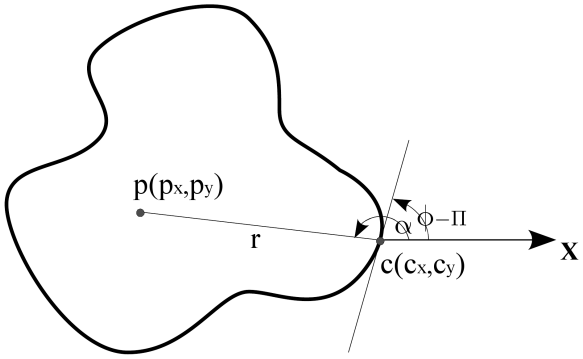


Figure 6.2.: Parameters used in the template model encoding of the GHT.

| Gradient angle | Encoding pairs |
|---|---|
| 0 | $(\alpha_{11}, r_{11})$ $(\alpha_{12}, r_{12})$ ... $(\alpha_{1n_1}, r_{1n_1})$ |
| $\Delta\phi$ | $(\alpha_{21}, r_{21})$ $(\alpha_{22}, r_{22})$ ... $(\alpha_{2n_2}, r_{2n_2})$ |
| $2\Delta\phi$ | $(\alpha_{31}, r_{31})$ $(\alpha_{32}, r_{32})$ ... $(\alpha_{3n_3}, r_{3n_3})$ |
| ... | . . . |
| $(m-1)\Delta\phi$ | $(\alpha_{m1}, r_{m1})$ $(\alpha_{m2}, r_{m2})$ ... $(\alpha_{mn_m}, r_{mn_m})$ |

Table 6.1.: The R-table example.

After the template model is encoded the object image (where the model detection is performed) is ready to be processed. At every object boundary point $b = (b_x, b_y)$ the

absolute gradient angle $\phi$ is calculated, and the pairs $(b, \phi)$ are considered as the image features. For each feature $(b, \phi)$ the parameters (reference points) that should receive votes are defined according to the relations:

$$\begin{cases} p_x = b_x - r\cos\alpha, \\ p_y = b_y - r\sin\alpha, \end{cases} \tag{6.5}$$

for every pair $(r, \alpha)$ from the row of the R-table specified by the value of $\phi$. This algorithm defines the standard GHT feature-to-vote conversion function $H_C(b, \phi)$ which maps an image feature $(b, \phi)$ into a set of reference points $\{(p_x, p_y)\}$ that receive votes. The main purpose of the following step is to find the unknown vote peaks in the parameter space, i.e. the coordinates of the reference points that received the high amount of votes.

**Standard GHT vote analysis principle.** In order to the find vote peaks all calculated votes using (6.5) are consistently accumulated in a preallocated two-dimensional array (grid). Upon completion of the feature-to-vote conversion for all object points, the accumulator elements with the high number of votes specify the coordinates of the desired reference points. For each such reference point the recognized template model position within the object image can be reconstructed using the R-table and a straightforward decoding procedure. This is the main idea of the standard GHT-based object recognition proposed by Ballard [15].

The main inconvenience of the GHT is that it was originally developed for detection of only one geometrically transformed instance of the template. Later, in order to overcome this limitation, the method was modified for detection of the affine transformed versions of the template by means of introduction into (6.5) external scale-rotation parameters and extension of the parameter space. However, this made the GHT too bulky as the voting process used the brute force search in the scale-rotation subspace. Namely, the whole GHT voting procedure was performed for each instance of the scaling and rotation parameters, that exponentially increased the computational complexity of the algorithm. In Section 6.3 we introduce an alternative and more intuitive way to expand the GHT-based object recognition up to the rotational and uniform scaling invariance.

### 6.1.4. Fast Hough Transform

An alternative approach to the problem of the vote analysis for the standard GHT, called the Fast Hough Transform (FHT), was proposed by Li [130, 131] in 1986. The FHT is based on the idea of the coarse-to-fine parameter space exploration. It reduces the complexity of the standard grid vote analysis by considering a hierarchically organized parameter space. The *FHT feature-to-vote conversion function* maps the image features into the parameters that form hyperplanes in the Hough space. Accordingly to the *FHT vote analysis principle*, the parameter space is iteratively subdivided and the number of votes that receives each subspace is defined by the number of hyperplanes intersecting this subspace (Figure 6.3). The voting and the subdivision procedures are recursively

performed for the subspaces with the high number of votes until the subspaces cannot be further subdivided. All subspaces at the last subdivision iteration form the desired set of the parameters that define the locations and the poses of the detected objects.

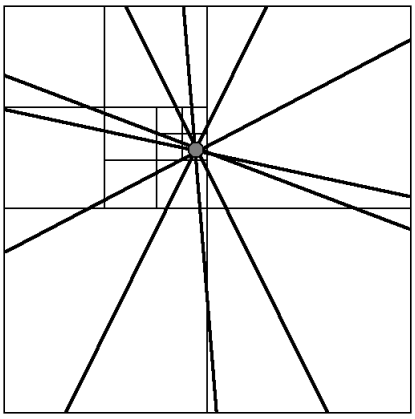

Figure 6.3.: An example of the hierarchical 2D space subdivision based on the number of intersections with 2D hyperplanes (thick lines) used in the FHT. The thin lines show the borders of 2D hypercubes for 4 iterations (on each iteration only one subspace is further subdivided.

The hierarchical vote analysis principle leads to significant reduction of memory consumption comparing to the standard one, as it uses "low-resolution" arrays for vote accumulation. The size of such accumulator arrays is defined by the number of the subspaces produced by a parent space at each iteration. For example on the Figure 6.3 the FHT algorithm uses a $2 \times 2$-array for vote analysis, while the GHT needs a $16 \times 16$ one. On the other hand the FHT performs the vote analysis procedure in four iterations (four calls of the feature-to-vote conversion routine), while the GHT needs only one. From this follows the main informal property of the hierarchical vote analysis principle: it "converts" the excess memory consumption into additional computational time and possible peak omission errors. This property is especially essential for the high-dimensional and high-resolution GHT-based transforms when the sizes of the standard accumulators can exceed hundreds of terabytes. For example, a grid accumulator for the 4D parameter space analysis (translation, uniform scaling and rotation) with the resolution of 1024 elements in each dimension needs about 4TB of memory, that is hardly implementable on a convenient workstation. The modern GHT-based algorithms [199] operate with the high-resolution parameter spaces up to 8D, that absolutely eliminates usage of the standard accumulators. On the other hand, the increase of the computational time that ensues from the hierarchical principle, is compensated by the growing power of the modern processors and GPU's providing the possibility to parallelize the algorithm effectively.

As we already mentioned, in spite of the advantage in low memory consumption the FHT method has several complications [102]. The subspaces for the recurrent subdivision are selected by the number of their votes that must exceed some threshold. In complex images this threshold may be difficult to determine. From this follows the main drawback of the method: if the threshold is set too low the efficiency of the method will

decrease as the algorithm will explore too many subspaces. On the other hand, if the threshold is set too high the hierarchical vote analysis will produce peak omission errors. It can happen when a subspace that contains a vote peak has a small total number of votes and is skipped by the subdivision algorithm. In this case the right object location and the pose will not be detected. In order to solve this problem an optimal subdivision policy should be used. Also, it should be mentioned that the original FHT method does not explain properly how all two-dimensional shapes can be naturally mapped into hyperplanes.

## 6.2. Motivation and objectives

In spite of its popularity the GHT has an imperfect feature: it does not implicitly gather the votes for the object scale and orientation. The common technique that allows using the GHT in rotation-scale invariant object detection is based on the exhaustive search within the predefined range of scales and angles. For each fixed pair (angle, scale) the standard search for the position is performed, that is a rather ineffective approach. One intuitive solution to improve this drawback is to introduce the second reference point into the GHT. This, on the one side, allows the transform to consider rotation-scale information from the objects in a more effective way, but on the other side, extends the parameter space from 2D to 4D. That results in a significant increase of the size of the accumulator array used in the standard GHT vote analysis.

The idea of two reference points utilization in the GHT-based object recognition was first proposed by Yip et al. [204]. This method contained a quite complicated voting procedure using supplementary invariant tables associated with the Ballard's R-table and two-dimensional arrays for accumulation of three- and four-dimensional parameters without further refinement in 4D, that produced only approximations of the exact solutions. Several two-point variations of the GHT were also introduced in the works of Chau and Siu [32], however these methods did not use two-reference points for object parametrization. They proposed an indexing scheme and a search algorithm based on the characteristic angle non-invariant to rotation and scaling. Later they improved their method and developed the adaptive approach that provides better shape encoding and reduces the number of false votes [33].

In this chapter we aim at developing a straightforward modification of the Generalized Hough Transform for 2D object recognition that is invariant to uniform scaling and rotation. Because of the extension of the parameter space from 2D to 4D, the standard vote accumulation used in the GHT is becoming too memory consumptive. Therefore, the second our objective is to develop the corresponding effective vote analysis method.

## 6.3. Material and methods

The proposed transform, called Dual-Point GHT (DPGHT), operates in a 4D parameter space and is based on the two-reference-point parametrization that is used for template encoding and the feature-to-vote function construction [93]. The features of template models are encoded relatively to the reference points and do not fastened to the global

image coordinate system. The two-reference-point parametrization is an intuitive and a flexible tool for 2D object recognition that allows extracting models invariantly to the similarity transformation and can be applied in various tasks for objects detection up to rotation and uniform scaling.

In this section we will introduce two alternative feature-to-vote functions and analyse their properties. The first function maps image features into sets of constrained four-dimensional lines, while the second one transforms the features into constrained hyperplanes in the parameter space. As the image features the both functions utilize object boundary location and tangent vectors. Utilization of the boundary tangents reduces the number of false evidences during the voting procedure and improves the quality of results. We will also introduce two different approaches to the vote analysis. The first approach can be classified as an approximate one, while the second one is based on an optimized hierarchical vote analysis strategy that minimizes the probability of the vote peaks omission. Depending on the combination of the proposed feature-to-vote conversion functions with the vote analysis techniques we distinguish three variations of the Dual-Point GHT shown in Table 6.2.

| **Dimensionality (Object space/ Hough space)** | **Feature-to-vote conversion function** | **Vote analysis principle** |
|---|---|---|
| 2D/2D(4D) | Feature (intensity + gradient) to Lines in 2D | Approximate (two 2D projections) |
| 2D/4D | Feature (intensity + gradient) to Lines in 4D | Optimized hierarchical |
| 2D/4D | Feature (intensity + gradient) to Hyperplanes in 4D | Optimized hierarchical |

Table 6.2.: Variations of the proposed Dual-Point Generalized Hough Transform.

## 6.3.1. Feature-to-vote conversion function

We distinguish two basic classes of feature-to-vote conversion functions that can be used in the GHT-like object recognition methods: *unconstrained* and *constrained*. The functions from the first class map image features into unconstrained sets in the parameter space such as lines, hyperplanes and curves, maximizing the number of both true and false evidences. The functions from the second class realize a more selective voting process by mapping image features into constrained sets or even single points in the parameter space minimizing the number of false evidences and trying to preserve the number of true ones. This class is presented by some modern methods [3, 16] that involve probabilistic models and local object features for suppression of the false evidences. The proposed further line-based and hyperplane-based feature-to-vote conversion functions belong to the *constrained* class.

### Line based feature-to-vote conversion

Similarly to the conventional Dual-Point GHT [204], two different reference points $P_1 = (P_{1x}, P_{1y}) \in \mathbb{X}$ and $P_2 = (P_{2x}, P_{2y}) \in \mathbb{X}$ are selected within the template do-

main according to some rule (e.g. inside the template model symmetrically to its center of mass). Analogously to the GHT the template model is first *encoded* and represented in so-called $\alpha$-*table* format as follows (Table 6.3). Every boundary point of the model $C \in \mathbb{X}$ is mapped into a pair $(\alpha, \beta)$, where $\alpha$ is the angle between vectors $P_1C$ and $P_1P_2$ and $\beta$ is the angle between $P_2C$ and $P_1P_2$ as it is shown on the Figure 6.4. The resulting pair is stored in the $\alpha$-table in the row that is defined by angle $\gamma$ between the vector $CP_1$ and the boundary tangent vector $T$ calculated at the point $C$.

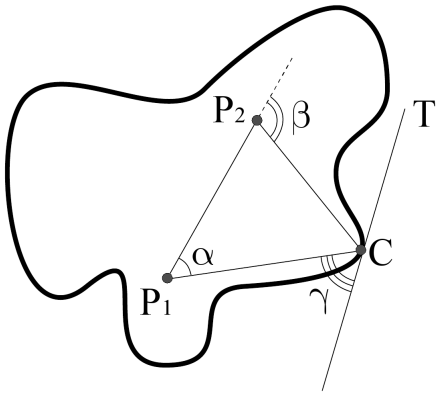


Figure 6.4.: Parameters used in the DPGHT template model encoding.

| **Tangential angle** | **Encoding angles** |
|---|---|
| 0 | $(\alpha_{11}, \beta_{11})$ $(\alpha_{12}, \beta_{12})$ ... $(\alpha_{1n_1}, \beta_{1n_1})$ |
| $\Delta\gamma$ | $(\alpha_{21}, \beta_{21})$ $(\alpha_{22}, \beta_{22})$ ... $(\alpha_{2n_2}, \beta_{2n_2})$ |
| $2\Delta\gamma$ | $(\alpha_{31}, \beta_{31})$ $(\alpha_{32}, \beta_{32})$ ... $(\alpha_{3n_3}, \beta_{3n_3})$ |
| ... | . . . |
| $(m-1)\Delta\gamma$ | $(\alpha_{m1}, \beta_{m1})$ $(\alpha_{m2}, \beta_{m2})$ ... $(\alpha_{mn_m}, \beta_{mn_m})$ |

Table 6.3.: The $\alpha$-table example.

After the template model is encoded the object image is ready for processing. At every boundary object point $b = (b_x, b_y)$ the unit boundary tangent vector $T = (T_x, T_y)^T$ is calculated and the pairs $(b, T)$ are considered as the image features. For each feature $(b, T)$ and every pair $(\alpha, \beta)$ with the corresponding $\gamma$ from the $\alpha$-table, the parameters that should receive votes are represented by a line in the 4D space:

$$L = L(\alpha, \beta, \gamma) = D \cdot t + B,\ t \in \mathbb{R}, \tag{6.6}$$

where $B = (b_x, b_y, b_x, b_y)^T$, and $D$ is the product of the $4 \times 2$ directional matrix of two rotational $2 \times 2$ blocks and the tangential vector $T$:

$$D = \begin{bmatrix} Rot(\gamma) \\ Rot(\gamma + \beta - \alpha) \cdot \sin\alpha / \sin\beta \end{bmatrix} \cdot T,\ Rot(x) = \begin{bmatrix} \cos x & -\sin x \\ \sin x & cosx \end{bmatrix}. \tag{6.7}$$

In other words, each image feature $(b, T)$ defines the set of lines in 4D parameter space passing through point $B$ parallel to their directional vectors $D$ specified by the pre-calculated $\alpha$-table. Each line defines a set of 2D point pairs (4D points), that reconstruct all scaled and rotated versions of the corresponding template segments $CP_1$ and $CP_2$ relatively to the current tangent $T$ and the point $b$ (Figure 6.5). The proportion of these segment lengths $\|CP_2\|/\|CP_1\| = \sin\alpha/\sin\beta$ (according to the law of sines) is constant and equals to the proportion from the template model.

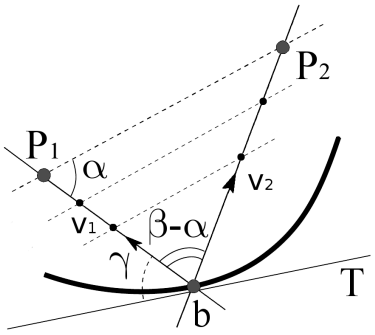


Figure 6.5.: Geometry used in the line-based DPGHT feature-to-vote conversion function.

Note, that each point that belongs to a 4D line votes for the scaling level (scaling on which the objects are detected) that is defined by the distance of this point to $B$. Namely, the point defined by some $t \in \mathbb{R}$ from (6.6) corresponds to the scaling level:

$$s(t) = \frac{\|(v_{1x} - v_{2x}, v_{1y} - v_{2y})\|}{\|P_1 - P_2\|}\,|t|, \tag{6.8}$$

where $P_1$ and $P_2$ are the initial reference points, and $(v_{1x}, v_{1y}, v_{2x}, v_{2y})^T = D$.

In order to avoid singular solutions of the object recognition problem that occur at the scaling level $s = 0$ (in this case each image point $b$ produces a peak in the parameter space in itself, i.e. the detected $P_1 = b$ and $P_2 = b$), we have to determine positive constrains for the scaling range: $0 < s_1 \leq s(t) \leq s_2$. This gives us the corresponding constraints $t_1$ and $t_2$ for the line parameter $t$ that are defined by the relation (6.8). Summarizing the above, each image feature $(b, T)$ generates a set of line segments in the parameter space defined by equations (6.6)-(6.7) with the constrains $t_1 \leq t \leq t_2$ defined by (6.8) for the given scaling interval $[s_1, s_2]$. The described algorithm represents the line based feature-to-vote conversion function $H_C : \{(b, T)\} \to \mathbb{P}^*$.

It should be mentioned that at the template model encoding step and during the feature-to-vote construction we did not consider the case when $\sin\beta = 0$ (that is also equivalent to $\beta = \alpha$). This corresponds to the boundary points that are located on the line passing through the both reference points $P_1$ and $P_2$, or to infinitely far points (that are not relevant in practical applications and therefore are not considered here). Thus, in this case, at the object encoding step we set $\alpha = 0$ and $\beta = P_1C \cdot P_1P_2 \;/\; \|P_1P_2\|^2$, and if during the feature-to-vote mapping we obtain a pair $(\alpha, \beta)$ with $\alpha = 0$, we calculate $D$ using the following formula instead of (6.7):

$$D = \begin{bmatrix} Rot(\gamma) \cdot \beta \\ Rot(\gamma) \cdot (\beta - 1) \end{bmatrix} \cdot T.$$

It should be also noted that due to inaccuracies that can occur in the tangent vector $T$ calculation, the angle $\gamma$ in (6.7) may be replaced by a set of angles from some intervals $[\gamma-\epsilon;\gamma+\epsilon]$, $\epsilon > 0$, that will give us a set of segments instead of one. It provides additional stability and improves the accuracy but, in turn, increases the complexity of the algorithm.

The described feature-to-vote conversion function belongs to the class of constrained functions as it maps each image feature into a bunch of four-dimensional segments calculated relatively to this feature. The function uses boundary tangent information of the analysed objects that, on the one side, provides the low level of false evidences generated by each object feature. But on the other side, it is highly dependable on the accuracy of the tangent vector calculation and in the case of noisy data and deformed object boundary it can miss true evidences. In order to mitigate this drawback and to increase the number of true evidences in the voting procedure, we will expand the bundle of four-dimensional lines into a hyperplane. After this we will introduce a number of constrains on the created hyperplane that will filter out possible false evidences.

### Hyperplane based feature-to-vote conversion function

This function realizes a more stable paradigm by operating with hyperplanes instead of bundles of lines in 4D [14]. Accordingly to the linear feature-to-vote conversion, for each image feature $(b, T)$ and every pair $(\alpha, \beta)$ with the corresponding $\gamma$ from the $\alpha$-table the parameters that receive votes belong to a line in the 4D parameter space, defined by the point $B = [b, b]^T$ it is passing through and the four-dimensional directional vector $D = (v_{1x}, v_{1y}, v_{2x}, v_{2y})^T$ (6.6, 6.7). The directional vector in the ordinary case consists of the elements of two 2D vectors $v_1 = Rot(\gamma)\ T$ and $v_2 = Rot(\gamma + \beta - \alpha)\ T \cdot \sin\alpha / \sin\beta$, such that the angle between them is $(\beta-\alpha)$, and the angle between $v_1$ and the tangential vector $T$ is $\gamma$ (Figure 6.7). Due to inaccuracies in the tangential vector calculation, image noise or minor deformations of the analysed objects, the consideration of only one line in the parameter space may be insufficient and can result in loss of true evidence and therefore suboptimal voting process. As it has been already mentioned, in order to overcome this problem a bundle of boundary tangential vectors $T$ has to be considered for each feature point $b$. In the most common case this bundle consists of all rotated versions of the original tangential vector $T$, that in turn produces a one-parametric family of directional vectors:

$$D(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} v_{1x} \\ v_{1y} \\ v_{2x} \\ v_{2y} \end{pmatrix},$$

$$D(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{pmatrix} \cdot D.$$

Notice, that the point set $L(\theta, t) = B + D(\theta) \cdot t$, $\theta \in [0; \pi]$, $t \in \mathbb{R}$, that define a bundle of lines passing through $B$, belongs to a *hyperplane* in 4D with the normal vector $n = (v_{2y}, v_{2x}, -v_{1y}, -v_{1x})^T$. Indeed, it is easy to show that the dot product $D(\theta) \cdot n = 0$, $\forall\theta \in \mathbb{R}$:

$$D(\theta) \cdot n = \begin{pmatrix} -n_4\cos\theta + n_3\sin\theta \\ -n_4\sin\theta - n_3\cos\theta \\ n_2\cos\theta - n_1\sin\theta \\ n_2\sin\theta + n_1\cos\theta \end{pmatrix} \cdot \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix} =$$

$$= -n_1n_4\cos\theta + n_1n_3\sin\theta - n_2n_4\sin\theta - n_2n_3\cos\theta +$$
$$+n_3n_2\cos\theta - n_3n_1\sin\theta + n_4n_2\sin\theta + n_4n_1\cos\theta = 0.$$

For convenience we will denote a hyperplane as a pair of its normal vector and a point on this plane. Thus, we showed that for a fixed image boundary point $b$, each entry from the $\alpha$-table corresponds to a subset on a hyperplane $H_b = (B, n)$ in the parameter space passing through the point $B$ with the normal vector:

$$n = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \cdot D. \tag{6.9}$$

Utilization of these hyperplanes in the vote generation process guarantees high presence of true evidences, but on the other hand, absence of additional restrictive conditions makes the feature-to-vote conversion function redundant. In order to avoid this and exclude false votes we introduce a number of *constraints* in the parameter space. The constrains are divided into two classes: the tangential (defined by the image tangent vectors) and the scaling ones (defined by the scaling range used within the current object detection task).

For each image feature $(b, T)$ and each $\alpha$-table entry we consider a class of two-dimensional half-planes $U(b, w) = \{x \in \mathbb{R}^2 \mid (x - b) \cdot w \geq 0\}$ of the following view (Figure 6.6):

$$U_{\theta,v} = U(b, Rot(\theta + \pi/2) \cdot v),\ U_{q,v} = U(b + q \cdot v, v),$$

and the function that describes the position of a 2D point $x$ relatively to a given half-plane $U(b, w)$:

$$z(x, U) = (x - b) \cdot w.$$

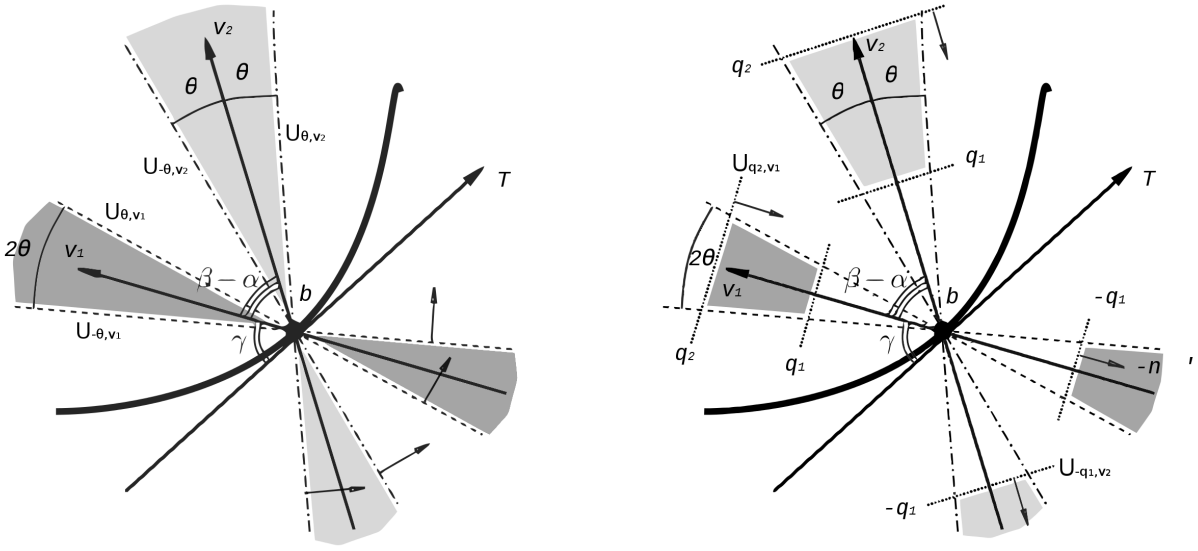


Figure 6.6.: 2D projection of the hyperplane produced by a local feature $(b, T)$ with the tangential- (left) and the both tangential and scaling constraints (right).

Using this we define the feature-to-vote mapping as follows. The image feature $(b, T)$ and the $\alpha$-table entry vote for the parameters $p = (p_{1x}, p_{1y}, p_{2x}, p_{2y}) \in \mathbb{P}$ that belong to the base hyperplane $H_b = (B, n)$ and the following constraints are satisfied:

*1) Tangential constraints:*

$$z(p_1, U_{\theta,v1}) \cdot z(p_1, U_{-\theta,v1}) \leq 0,$$
$$z(p_2, U_{\theta,v2}) \cdot z(p_2, U_{-\theta,v2}) \leq 0,$$

*2) Scaling constraints:*

$$z(p_1, U_{\pm q1,v1}) \cdot z(p_1, U_{\pm q2,v1}) \leq 0,$$
$$z(p_2, U_{\pm q1,v2}) \cdot z(p_2, U_{\pm q2,v2}) \leq 0.$$

Where $p_1 = (p_{1x}, p_{1y})^T$ and $p_2 = (p_{2x}, p_{2y})^T$. The value $\theta$ defines the maximal allowed tangential angle inaccuracy. The bigger is $\theta$, the more robust is the method to tangential errors, noisy and irregularly deformed objects, but on the other hand, the more false evidences are considered. The values $q_1$ and $q_2$ represent the scaling range in which the objects are detected, that is calculated similarly to $t_1$ and $t_2$ from (6.8).

So, we defined the function that maps features of arbitrary 2D objects into constrained hyperplanes in 4D. Further will discuss the problem of vote peaks estimation in the four-dimensional parameter space.

### 6.3.2. Vote analysis

In this section we consider two vote analysis approaches that can be used in the Dual-Point GHT. The first one falls into the approximate class and operates in 2D projections

of the 4D parameter space with the following refinement of the results in 4D. It performs estimation of the vote peaks in the parameter space using quadratically reduced memory arrays comparing to the standard grid approach used in the GHT and the HT. The second approach can be classified as a hierarchical one. It utilizes an optimized parameter space subdivision strategy that minimizes the probability of the vote peaks omission.

### Approximate DPGHT vote analysis

This approach is based on the specificities of the both GHT and the Radon Transform (RT) [161], but avoid extreme memory consumption of the standard GHT vote analysis, that is especially significant in 4D. We estimate the vote peaks in the four-dimensional parameter space by analysing its two-dimensional orthogonal projections and therefore using two-dimensional accumulators instead of four-dimensional ones. First, we find the areas of the most probable parameter location using 2D projection analysis of the full 4D parameter space and the standard HT formalism for votes accumulation. Second, in a similar manner like the brute force searching pose space (mentioned by Amit [8], Chapter 8), we calculate the precise position of the high-voted reference points within the found areas considering the RT-like form of the Dual-Point GHT that corresponds to the general form (6.4) with the *line-based feature-to-vote conversion function* $H_C(b,T)$:

$$F_C(I,P) = \sum_{(b,T)\in I} 1_{P\in H_C(b,T)}, \tag{6.10}$$

where $F_C(I,P) : \mathbb{P} \to \mathbb{N},\ P = (P_{1x}, P_{1y}, P_{2x}, P_{2y}) \in \mathbb{P}$. In other words this function defines how many votes were given to the parameter $P$ by all the features from the image $I$.

At the first step, similarly to the approximate vote analysis principles, we calculate the orthogonal cumulative projections of $F_C(I,P)$ onto two-dimensional subspaces $\mathbb{P}_1 = \{(P_{1x}, P_{1y})\}$ and $\mathbb{P}_2 = \{(P_{2x}, P_{2y})\}$:

$$F_{1,C}(I,P_1) = \sum_{(P_{2x},P_{2y})\in\mathbb{P}_2} F_C(I,P),$$
$$F_{2,C}(I,P_2) = \sum_{(P_{1x},P_{1y})\in\mathbb{P}_1} F_C(I,P).$$

These projections indicate the total number of the votes given to one reference point (either $P_1$ or $P_2$) independently from the second one. They can be calculated using 2D arrays and the standard grid vote analysis approach. Indeed, applying the function $H_C$ to each image feature $(b,T)$ we accumulate the values of $H_C(b,T)$ for only two of its dimensions ignoring the other ones. Practically, we project the four-dimensional lines $L$ (6.6) onto 2D arrays, performing the grid accumulation procedure similarly to the GHT. Each line $L$ produces two lines $l_1$ and $l_2$ in the spaces $P_1$ and $P_2$ correspondingly. These lines specify the sets of potential positions of reference points in 2D invariantly to rotation and uniform scaling of the initial template model (Figure 6.7). Once the vote accumulation process is performed for all image features, the areas of the 2D arrays with the high amount of votes (more than a predefined threshold $\tau$) can be used as an

initial estimation of the regions in 4D with the high probability of the vote peaks. We assign the corresponding sets in 2D as $\mathbb{Q}_1(\tau) \in \mathbb{P}_1$ and $\mathbb{Q}_2(\tau) \in \mathbb{P}_2$.

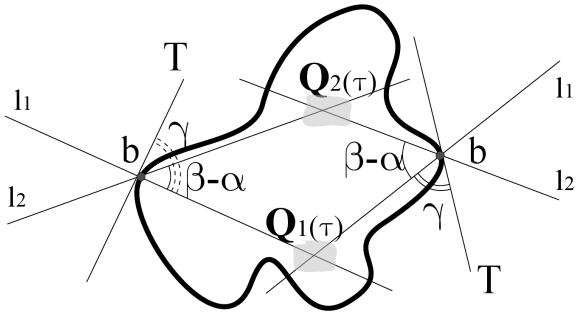


Figure 6.7.: Scheme of 2D projections reconstruction used in the approximate DPGHT vote analysis.

At the second step we analyse the parameters $P \in \mathbb{P}$ such that $(P_{1x}, P_{1y}) \in \mathbb{Q}_1(\tau)$ and $(P_{2x}, P_{2y}) \in \mathbb{Q}_2(\tau)$ in order to find the precise vote peak in four dimensions. This procedure is based on the RT-like (6.10) direct calculation of the function $F_C(I, P)$ for all parameters from the cartesian product $\mathbb{Q}_1(\tau) \times \mathbb{Q}_2(\tau)$. Such an approach is effective for high values $\tau$ due to its low memory consumption and the relatively small cardinality of the mentioned cartesian product.

Practically, we can even further accelerate the direct calculation of $F_C$ by considering it as a two-dimensional transform for every fixed $(P_{1x}, P_{1y}) \in \mathbb{Q}_1(\tau)$ and performing the grid vote analysis for the remaining free parameters $(P_{2x}, P_{2y}) \in \mathbb{Q}_2(\tau)$ as follows. For every fixed reference point $P_1 \in \mathbb{Q}_1(\tau)$ we accumulate votes for $P_2 \in \mathbb{Q}_2(\tau)$ using a 2D array. Namely, for every image feature $(b, T)$ we calculate the angle $\gamma$ between the vectors $bP_1$ and $T$. For every pair $(\alpha, \beta)$ that corresponds to $\gamma$ from the $\alpha$-table, the reference point $P_2$ that should receive a vote is defined as the intersection of two lines passing through the points $P_1$ and $b$ with the angles $\alpha$ and $(\beta\text{-}\alpha)$ to the vector $bP_1$ correspondingly as it shown on the Figure (6.8).

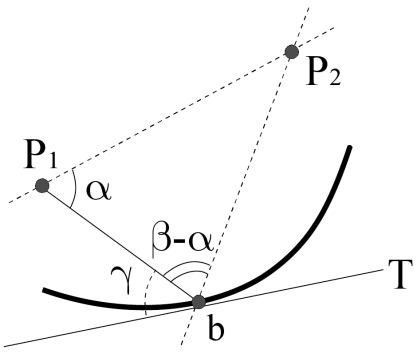


Figure 6.8.: Second step of the DPGHT vote analysis. Reconstruction of the reference point $P_2$ for a fixed $P_1$.

The parameters with the high vote number in the resulting array define the candidate

reference points. Once the procedure is performed for all $P_1 \in \mathbb{Q}_1(\tau)$, the best candidates (with the highest number of votes) $\{(P_1^*, P_2^*)\}$ define the desired reference points. Using these points $P_1^* = (P_{1x}, P_{1y})$, $P_2^* = (P_{2x}, P_{2y})$ and the $\alpha$-table we reconstruct all detected positions and scales of the template model. For each pair $(P_1^*, P_2^*)$ we *decode* the model points $R(\alpha, \beta)$ as the intersections of the lines passing through $P_1^*$ and $P_2^*$ and having the angles $\alpha$ and $\beta$ with the vector $P_1^*P_2^*$ correspondingly (Figure 6.4). The coordinates of these points are calculated accordingly to the formula that follows from the sine law:

$$R(\alpha, \beta) = P_1^* + \frac{\sin\beta}{\sin(\beta - \alpha)} Rot(\alpha) \cdot (P_2^* - P_1^*). \tag{6.11}$$

If a model point belongs to the line passing through $P_1$ and $P_2$, i.e. the corresponding $\alpha = 0$, its coordinates are calculated as:

$$R(\alpha, \beta) = P_1^* + \beta \cdot (P_2^* - P_1^*). \tag{6.12}$$

**Computational complexity.** The computational complexity of the first step of the approximate vote analysis is $O(N_1 N_2)$, where $N_1$ is the number of the features in the template model and $N_2$ is the number of the image features. The second step has the computational complexity $O(|\mathbb{Q}_1(\tau)|N_1N_2)$.

**Memory consumption.** Assuming that the parameter space has the same discretion as the analysed image $I$ and the reference points of the detected objects belong to $I$, the total memory needed for the approximate vote analysis is $O(K)$, where $K$ is the number of pixels in $I$.

Relatively low memory consumption and low computational complexity together with a highly-parallelizable paradigm make the DPGHT approximate vote analysis with the line-based feature-to-vote conversion function an effective combination for real-time object recognition applications (experiments and practical implementation are discussed in Section 6.4). However, the approach does not guarantee the global optimal solution, and because of its projection nature it may miss local vote peaks, that results in a sub-optimal object detection. This is the main drawback of this approach. In the next sections we will consider an alternative vote analysis principle and solve the problem of minimisation of vote peaks omission.

### Hierarchical DPGHT vote analysis

The exact solution of the peak search problem in the Hough space can be obtained using the standard grid vote analysis. However it is often awkward on practice due to extreme memory consumption of the grid arrays. To avoid this problem, that makes it difficult or even impossible to process high-resolution images and to operate in the multidimensional parameter space, we have modified the hierarchical principle used in the Fast Hough Transform (Section 6.1.4) and integrated it with *the line-based* and *the hyperplane-based feature-to-vote functions* discussed in Section 6.3.1.

**Definition 6.1.** *Let $T = (S, E)$ be an oriented perfect $K$-ary tree of height $M$, where $S$ is the set of nodes and $E$ is the set of edges directed by the increase of the node depths.*

**Definition 6.2.** *Let $c(s) = \{s' \in S \mid e(s, s') \in E\}$ be the set of all child nodes of node $s \in S$ in tree $T$.*

**Definition 6.3.** *Let $d(s)$ be the depth of node $s \in S$, and $c_q(s) = c^{q-d(s)}(s)$, $q > d(s)$, be the set of all its descendants of depth $q$.*

From Definition 6.1 follows that each node $s \in S$ (except the leaf ones) has exactly $K$ children, and all the leaf nodes are at the same depth $M$. We assume that each $s \in S$ bijectively corresponds to some hyperrectangle $h(s) \subset \mathbb{P}$ in the parameter space $\mathbb{P} \subset \mathbb{R}^4$, such that the following condition is performed:

$$\bigcup_{s' \in c(s)} h(s') = h(s).$$

We also demand that all hyperrectangles corresponding to the same node depth have the equal volumes and do not intersect each other: $h(s_1) \bigcap h(s_2) = \emptyset$, $\forall s_1, s_2 \in S$, $d(s_1) = d(s_2)$, $s_1 \neq s_2$.

Thus, each $s \in S$ with its child nodes $c(s)$ define a subdivision of hyperrectangle $h(s)$ into $K$ non-intersecting hyperrectangles. The hyperrectangle $h(s_0)$, that corresponds to the tree root $s_0$, envelops the whole region of interest within the parameter space $\mathbb{P}$, where the search for the vote peaks is performed. Therefore, $T$ represents a complete recursive subdivision of the region of interest into smaller hyperrectangles. The size of these hyperrectangles, their number and, therefore, the maximal possible accuracy of the vote analysis is defined by the tree height $M$.

The *objective* of the hierarchical vote analysis is to construct an oriented *full* $K$-ary tree $T_1 = (S_1, E_1)$ of height $M$ (see Figures 6.9 and 6.10), such that $T_1$ is a subtree of $T$ and its nodes $S_1$ satisfy some *subdivision condition*. This condition is based on the number of votes $v(s)$, that the hyperrectangles $h(s)$ receive in respect to the feature-to-vote conversion function. Analogously to $T$, $T_1$ represents a specific hierarchical subdivision of the region of interest, defined by the node votes $v(s)$ and the subdivision condition. The number of nodes in $S_1$ defines the complexity of the hierarchical vote analysis.

As an example, accordingly to the $FHT$ hyperrectangle $h(s)$, $s \in S$, will be subdivided (i.e. nodes $c(s)$ and the corresponding edges will be included into $T_1$), if $v(s)$ is more than some threshold value.

Thus, there are two main issues that have to be defined before the construction of $T_1$ (or, equally, the subdivision of the parameter space): a) vote calculation procedure and b) subdivision condition.

**Vote calculation procedure**. The number of votes $v(s)$ received by hyperrectangle $h(s)$, $s \in S$, is calculated accordingly to the selected feature-to-vote function. In the case of *the line-based feature-to-vote function*, $v(s)$ is proportional to the total length of all intersections of $h(s)$ with the four-dimensional segments produced by the analysed image features. If *the hyperplane-based feature-to-vote function* is used, $v(s)$ is proportional

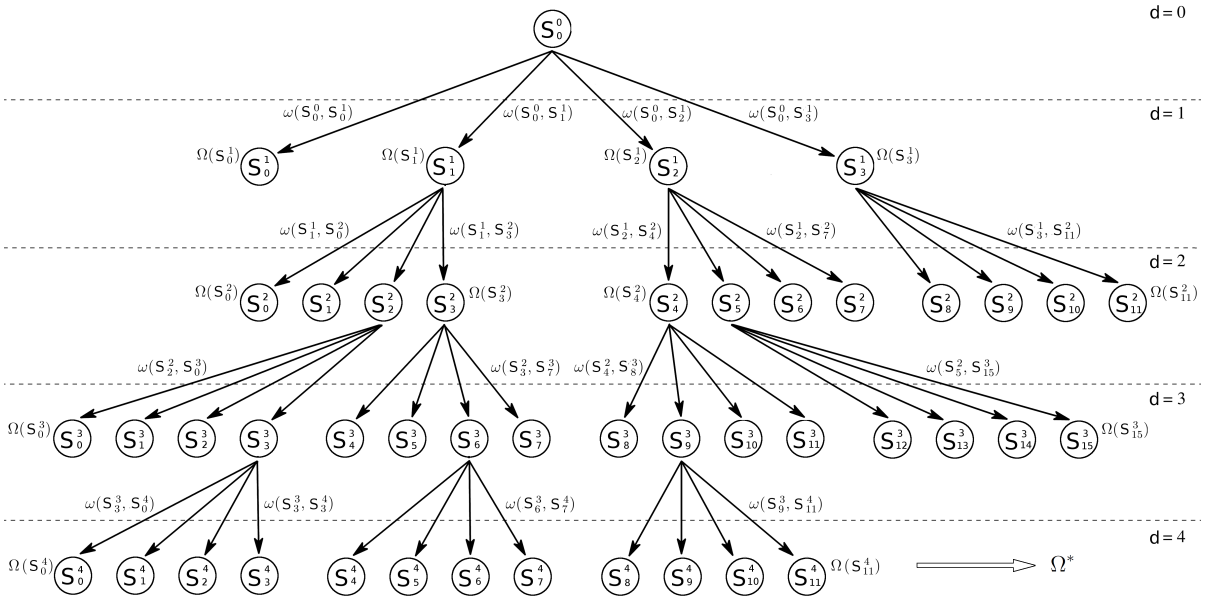


Figure 6.9.: An example of tree $T1$ for $\mathbb{P} \subset \mathbb{R}^2$, $K = 4$, $M = 4$, subdivision rule B.

to the total volume of all intersections of $h(s)$ with the four-dimensional constrained hyperplanes (Figure 6.6) produced by the analysed image features (see Appendix B for a detailed discussion of the volume of a hyperplane-hyperrectangle intersection).

**Subdivision condition.** $T_1$ construction is preformed iteratively starting from the tree root $s_0$. Let $L_1 \subseteq S_1$ be the set of all leaf nodes of $T_1$ at the current iteration, and $c(L_1) = \{s' \in S \mid (s, s') \in E, s \in L_1\}$ be the corresponding child nodes from $T$. For each child node $s \in c(L_1)$ the number of votes $v(s)$ is calculated. Then the best nodes from $c(L_1)$ are added into tree $T_1$, and a new iteration begins. The selection of the best nodes from $c(L_1)$ can be done using one of *two rules*:

***A)*** only one node $s \in c(L_1)$ with the maximal in $c(L_1)$ number of voices is added:

$$s = arg \max_{s \in c(L_1)} v(s);$$

***B)*** all nodes from $c(L_1)$ that in aggregate satisfy some condition are added.

As an implementation of such a condition we propose to use a probabilistic function that for a given integer $q \leq M$ and the set of nodes $C = c(L_1)$ defines its minimal subset $\widetilde{C} \subset C$, such that the probability of the event that the $q$-depth descendants of $\widetilde{C}$ will produce a higher vote peak than the $q$-depth descendants of $C \backslash \widetilde{C}$, is more than the pre-defined threshold $\Omega \in [0, 1]$:

$$P\left(\max_{s \in c_q(\widetilde{C})} v(s) > \max_{s \in c_q(C \backslash \widetilde{C})} v(s)\right) \geq \Omega \tag{6.13}$$

Having such a probabilistic function we will be able to define the optimal nodes for further subdivision considering their number of votes, depth and the depth $q$ at which

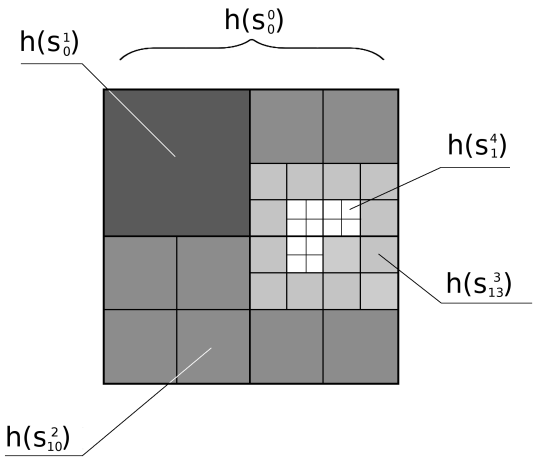


Figure 6.10.: The corresponding to Figure 6.9 example of parameter space $\mathbb{P} \subset \mathbb{R}^2$ subdivision. Intensity of colors in the rectangles depicts the density of votes $v(s)$.

the vote peaks are needed to be estimated. Section 6.3.3 is dedicated to the problem of constructing this function and investigation of the probabilistic properties of the hierarchical vote analysis. It also contains a more detailed description of the algorithm that realizes the subdivision rule B (Section 6.3.6). The hierarchical search defined by this rule can be considered as a probabilistic analogue of the branch and bound method [118, 206].

**Computational complexity.** The subdivision strategy influences directly the number of branches in the constructed tree and together with the tree height $M$ defines the computational complexity of the vote analysis. Notice, that the rule A causes the minimal computational complexity but the lowest accuracy. With the growth of the number of nodes selected for subdivision the accuracy and the computational complexity of the algorithm increase respectively (that corresponds to the rule B). Namely, the computational complexity is proportional to the number $Q$ of calls of the feature-to-vote conversion procedure performed in total at all iterations. That is equal to the total number of nodes in $T_1$ excluding the leafs and the root node.

Considering the minimal (rule A) and the maximal (rule B with subdivisions of all nodes) complexity cases, we can estimate the lower and the upper boundaries for the number of the feature-to-vote procedure calls $Q$:

$$K(M-1) \leq Q \leq \tfrac{K^M-1}{K-1} - 1.$$

The upper boundary of $Q$ is calculated for a perfect $K$-ary tree (i.e. in the case $T_1 = T$), while the lower one is for a minimal full $K$-ary tree of height $M$ (rule A). The value $Q$ also defines the ratio of the computational complexity of the hierarchical vote analysis to the complexity of the grid one. Summarizing the above, the complexity of the hierarchical vote analysis is $O(QN_1N_2)$, where $N_1$ is the number of the features in the template model and $N_2$ is the number of the image features. In spite of the

fact that, in general, the proposed hierarchical algorithm is computationally much more complex that the grid one, it is well intended for parallelization and needs significantly less memory.

**Memory consumption.** Utilization of the grid vote analysis for object detection invariantly to the similarity transformation on a 2D image of size $N \times N$ needs $O(N^4)$ bytes of memory. The considered hierarchical algorithm optimizes the memory consumption by operating at each level with a "low-resolution" accumulator arrays of $L$ elements in each dimension, that needs $O(L^4)$, $L \leq N$ bytes of memory. Using such accumulators the vote analysis procedure performs the peak search by construction a tree structure with $M \leq \lceil log_L^N \rceil$ resolution levels. For comparison, processing an image of size $512 \times 512$ in the case of the grid vote analysis needs an array of $512^4 \simeq 7 \cdot 10^{10}$ elements, that is 256GB of memory. The hierarchical DPGHT vote analysis with $L = 64$ reduces the accumulator array to $64^4 \simeq 2 \cdot 10^7$ elements (64MB of memory) and needs a tree structure with only two depth levels. So, at the expense of computational complexity we significantly reduced memory consumption of the method.

### 6.3.3. Probabilistic model of the hierarchical vote analysis

In this section we will analyse the probabilistic properties of the hierarchical vote analysis model in order to find out the optimal Hough space subdivision strategy. One of the important questions here concerns the accuracy of the hierarchical subdivision algorithm and the possible peak omission errors that are directly caused by the chosen subdivision strategy. The subdivision rule A (see Section 6.3.2) defines the most trivial subdivision strategy that results in a crude approximation of the peak search process. In this section we will consider the *rule B* with a probabilistic condition and answer the following question: how should one select the tree nodes for subdivision in order to avoid vote peak omission with the predefined probability. First we introduce a number of mathematical statements that will help us to elaborate the optimal subdivision strategy.

We will assign the set of all non-negative integers as $\mathbb{N}^0$, the set of all positive integers as $\mathbb{N}$ and consider the following problem. Let $V = \{v_1, v_2, ..., v_K\}$, $v_i \in \mathbb{N}^0$, be a set of natural numbers, and $S = \{S_1, S_2, ..., S_K\}$, $S_i = (s_{i1}, s_{i2}, ..., s_{in_i})$, be the corresponding set of series of non-negative integer elements $s_{ij}$ *uniformly distributed* on intervals $[0, v_i]$, such that $\sum_{j=1}^{n_i} s_{ij} = v_i$, $1 \leq i \leq K$. One has to find a minimal subset $\widetilde{S} \subseteq S$, such that for a predefined value $\Omega \in [0, 1]$ the following relation was true:

$$P\left(\max_{\widetilde{S}} s_{ij} > \max_{S \setminus \widetilde{S}} s_{ij}\right) \geq \Omega. \tag{6.14}$$

In order to solve this problem we provide the following statements.

**Definition 6.4.** *Denote $f_n^k$ the number of all n-permutations $(a_1, a_2, ..., a_n)$, $a_i \in \mathbb{N}^0$, $n \in \mathbb{N}$, with $\sum_{i=1}^{n} a_i = k$, $k \in \mathbb{N}^0$.*

**Lemma 6.1.** *For all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$, the value $f_n^k$ can be calculated using the formula:*

$$f_n^k = C_{k+n-1}^{n-1} = \frac{(k+n-1)!}{k!(n-1)!}. \tag{6.15}$$

*Proof.* Notice, that $f_n^k$ calculation is equivalent to the problem of calculating the number of all subdivisions of the segment $[0;k]$ into $n$ parts of arbitrary non-negative integer lengths. The length of each part $l_i \in \mathbb{N}^0$ in such an interpretation defines the value of the element $a_i$ in the corresponding permutation, $i \in \{1, 2, ...n\}$. The sum of the lengths is obviously equal to $k$. The last problem, in turn, is equivalent to calculation of all combinations with repetitions of collocation of $n-1$ points within the segment $[0;k]$. Each of these points defines a delimiter between the corresponding parts of the segment $[0;k]$. The number of such combinations with repetition is calculated using the standard formula: $\widetilde{C_{k+1}^{n-1}} = C_{(k+1)+(n-1)-1}^{n-1} = C_{k+n-1}^{n-1}$. The lemma is proven.

□

Further, we will consider $f_n^k$ as a function $f(k,n)$ and expand its domain over all $k, n \in \mathbb{Z}$, by assigning $f(0,0) = 1$, and $f(k,n) = 0$, for any other negative $k$ and non-positive $n$.

**Definition 6.5.** $f(k,n) = \begin{cases} f_n^k, & k \geq 0, n \geq 1, \\ 1, & k = 0, n = 0, \\ 0, & otherwise, \end{cases} \quad k, n \in \mathbb{Z}.$

Using this definition we introduce the following property of the number of $n$-permutations with the constant sum of elements.

**Lemma 6.2.** *For all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$, function $f(k,n)$ satisfies the following recurrent relation:*

$$f(k,n) = f(k,n-1) + f(k-1,n). \tag{6.16}$$

*Proof.* This relation follows from the Pascal's rule for the binomial coefficients: $C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$. In accordance with it $f_n^k = C_{k+n-1}^{n-1} = C_{k+n-2}^{n-2} + C_{k+n-2}^{n-1} = f_{n-1}^k + f_n^{k-1}$.

It should be noticed that this relation is valid only for those $k$ and $n$, where the both terms $f_{n-1}^k$ and $f_n^{k-1}$ in its right side are defined, i.e. $(k-1) \in \mathbb{N}^0$ and $(n-1) \in \mathbb{N}$, that is equivalent to $k \geq 1$, $n \geq 2$. For these parameter values function $f(k,n)$ by its definition coincides with $f_n^k$, so the relation (6.16) is true for $k \geq 1$, $n \geq 2$.

In order to prove (6.16) for all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$, we will show that this relation is additionally performed for *a)* $k = 0$, $n \geq 1$ and *b)* $k \geq 1$, $n = 1$. Indeed, in the case (a) expression (6.16) has the view:

$$f(0,n) = f(0,n-1) + f(-1,n) = f(0,n-1),$$
$$f(0,n) = f(0,n-1),\ n \geq 1.$$

Considering that $f(0,0) = 1$, we obtain $f(0,n) = 1$, $\forall n \geq 1$, that satisfies (6.15) by the definition of function $f$: $f(0,n) = f_n^0 = 1$. In the case (b) expression (6.16) has the view:

$$f(k,1) = f(k,0) + f(k-1,1) = f(k-1,1),$$
$$f(k,1) = f(k-1,1),\ k \geq 1.$$

Taking into account that $f(0,1) = f_1^0 = 1$, we obtain $f(k,1) = 1$, $\forall k \geq 0$, that also satisfies (6.15). The mentioned above means that the relation (6.16) is correct for all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$.

□

**Corollary 6.3.** *For all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$, function $f(k,n)$ satisfies the following recurrent relation:*

$$f(k,n) = \sum_{t=0}^{k} f(k-t, n-1). \tag{6.17}$$

*Proof.* This expression is obtained by recurrent application of the formula (6.16) to itself, namely to the second summand of its right side.

□

**Definition 6.6.** *Denote $f_n^k(x)$ the number of all $n$-permutations $(a_1, a_2, ..., a_n)$, $n \in \mathbb{N}$, with $a_i \in \{0, 1, 2, ...x\}$, $x \in \mathbb{N}^0$, and $\sum_{i=1}^{n} a_i = k$, $k \in \mathbb{N}^0$.*

Similarly to Definition (6.5) we introduce function $f(x,k,n)$ that expands the domain of $f_n^k(x)$ over all $k, n \in \mathbb{Z}$.

**Definition 6.7.** $f(x,k,n) = \begin{cases} f_n^k(x), & k \geq 0, n \geq 1, \\ 1, & k = 0, n = 0, \\ 0, & otherwise, \end{cases} \quad k, n \in \mathbb{Z},\ x \geq 0.$

**Lemma 6.4.** *For all $k, n \in \mathbb{Z}$ and $x \in \mathbb{N}^0$, such that $x \geq k$, the following relation is true:*

$$f(x,k,n) = f(k,n).$$

*Proof.* It follows from the fact that for all $x \geq k$, there is no more limitation above on the values of the non-negative elements of $n$-permutations with the constant sum $k$, therefore $f_n^k(x) = f_n^k$, and accordingly to Definitions (6.5 - 6.7), $f(x,k,n) = f(k,n)$.

□

**Lemma 6.5.** *For all $x, k \in \mathbb{N}^0$, $n \in \mathbb{N}$, function $f(x,k,n)$ satisfies the following recurrent relation:*

$$f(x,k,n) = \sum_{t=0}^{x} f(x, k-t, n-1). \tag{6.18}$$

*Proof.* Let us show that the function $f(x,k,n)$ satisfies the recurrent relation (6.18) for $k \geq 0$, $n \geq 1$. Notice, that for this parameter range the values of $f(x,k,n)$ equal by the definition to the number of all $n$-permutations with the constant sum of non-negative elements bounded above by value $x$. Suppose that the values of this function are known for $(n-1)$-permutations, i.e. all $f(x,i,n-1)$, $i \leq k$, are defined. In order to express

$f(x,k,n)$ recursively using the known $\{f(x,i,n-1)\}_{i\le k}$ we add one extra position $a_n$ to $(n-1)$-permutations, and fill it consequently with the values $t \in \{0,1,...,x\}$. Obviously, at each step $t$ the number of permutations $(a_1,a_2,...,a_{n-1},t)$ is defined by the number of $(n-1)$-permutations with the bounded by $x$ elements and the sum decreased by $t$ respectively, i.e $f(x,k-t,n-1)$. Taking sum of this expression for all values $t \in \{0,1,...,x\}$ we obtain $f(x,k,n) = \sum_{t=0}^{x} f(x,k-t,n-1)$.

It should be noticed that this formula is valid only if the values of all summands $f(x,k-t,n-1)$ in its right side have sense of $f_{n-1}^{k-t}(x)$, $t \in \{0,1,...,x\}$, i.e. $k \ge x$, $n \ge 2$. In order to prove (6.18) for all $k \in \mathbb{N}^0$, $n \in \mathbb{N}$, we will show that this relation is additionally performed for *a)* $k < x$, $n \ge 1$ and *b)* $k \ge x$, $n = 1$.

Notice, that in the case (a) accordingly to Lemma 6.4 function $f(x,k,n)$ equals to $f(k,n)$. Let us show that in this case the expression (6.18) has the same view as the recurrent formula (6.17). Indeed, performing decomposition and considering that by Definition (6.7) all $f(x,i,n-1)$ with negative $i$ are zeros, we obtain:

$$f(x,k,n) = \sum_{t=0}^{x} f(x,k-t,n-1) = \sum_{t=0}^{k} f(x,k-t,n-1) + \sum_{t=k+1}^{x} f(x,k-t,n-1),$$

$$f(x,k,n) = \sum_{t=0}^{x} f(x,k-t,n-1) = \sum_{t=0}^{k} f(x,k-t,n-1) + 0,$$

$$f(x,k,n) = \sum_{t=0}^{k} f(x,k-t,n-1),$$

$$f(k,n) = \sum_{t=0}^{k} f(k-t,n-1),$$

that considering Corollary (6.3) means that (6.18) in this case is true.

In the case (b) from the Definition (6.7) follows that for $x \in \mathbb{N}^0$, $k \ge x$, $n = 1$, $f(x,k,1) = f_1^k(x)$, that is the number of all 1-permutations (placements) of element $0 \le a_1 \le x$, such that $\sum_{i=1}^{1} a_i = k$, i.e. $0 \le a_1 = k \le x$. Obviously, from this follows that $f_1^k(x)$ is zero for $x < k$ and one for $x = k$. Show that expression (6.18) defines the same values. Indeed, in this case it has the view:

$$f(x,k,1) = \sum_{t=0}^{x} f(x,k-t,0),$$

that is in accordance with Definition (6.7) equivalent to:

$$f(x,k,1) = \begin{cases} 1, & x = k \\ 0, & x < k \end{cases},$$

that shows validity of (6.18) in this case as well.

All mentioned above means that the relation (6.18) is correct for all $x,k \in \mathbb{N}^0$, $n \in \mathbb{N}$. The lemma is proven.

□

Thus, the recurrent formula (6.18) defines a practical method of calculating the number of all permutations with the constant sum of non-negative elements bounded

above by some non-negative value. It should be also mentioned that the following statement is performed:

$$f(x,k,n) = \begin{cases} 0, & x < k/n, \\ 1, & x = k/n, \end{cases}$$

as the sum $k$ of $n$ elements strictly bounded above by value $k/n$ is never reached, and there is exactly one $n$-permutation with elements $k/n$ and sum $k$.

**Definition 6.8.** *Denote $p_n^k(x)$ the number of all $n$-permutations $(a_1, a_2, ..., a_n)$, $a_i \in \mathbb{N}^0$, $n \in \mathbb{N}$, with $\sum_{i=1}^{n} a_i = k$, $k \in \mathbb{N}^0$, and $\max\{a_i\}_{i=1}^{n} = x$, $x \in \mathbb{N}^0$.*

**Lemma 6.6.** *For all $x, k, n \in \mathbb{N}$, value $p_n^k(x)$ can be calculated using the following formula:*

$$p_n^k(x) = \sum_{t=1}^{\lambda} C_n^t f(x-1, k-tx, n-t), \quad \lambda = \max(\lfloor k/x \rfloor, n). \tag{6.19}$$

*Proof.* The number of all $n$-permutations with sum of the elements $k$ and the maximal element $x$ can be calculated as the number of all combinations with presence of at least one element $x$, multiplied by the corresponding number of all permutations with the values less than $x$ and the respectively decreased sum of the remaining elements. The maximal length of such combinations consisted of all elements $x$ is restricted by $k$ and $n$, and equals to $\lambda = \max(\lfloor k/x \rfloor, n)$, where $\lfloor \cdot \rfloor$ means the floor operator. The number of $n$-permutations that includes exactly $t$ $(1 \le t \le \lambda)$ elements $x$ is defined by the number of all $t$-combinations from $n$ elements, multiplied by the number of all $(n-t)$-permutations with the elements less than $x$ and the constant sum $k - tx$, that can be calculated as $f(x-1, k-tx, n-t)$. Therefore, the number of $n$-permutations that includes exactly $t$ elements $x$ is equal to $C_n^t f(x-1, k-tx, n-t)$. Taking sum of all such combinations from 1 to $\lambda$ we obtain the desired formula.

□

Notice, that (6.19) does not define the values of $p_n^k(x)$ for $x = 0$ and $k = 0$, but it can be done straightforward in the following way:

$$p_n^k(x) = \begin{cases} 1, & x = 0, k = 0, \\ 0, & x = 0, k \neq 0 \quad or \quad x \neq 0, k = 0. \end{cases}$$

**Corollary 6.7.** *For all $x, k, n \in \mathbb{N}$, the following recurrent relation is true:*

$$f(x,k,n) = \sum_{t=0}^{\lambda} C_n^t f(x-1, k-tx, n-t), \quad \lambda = \max(\lfloor k/x \rfloor, n). \tag{6.20}$$

*Proof.* Notice, that $p_n^k(x) = f(x,k,n) - f(x-1,k,n)$, therefore using relation (6.19) we obtain:

$$f(x,k,n) - f(x-1,k,n) = \sum_{t=1}^{\lambda} C_n^t f(x-1, k-tx, n-t),$$

$$f(x,k,n) = \sum_{t=1}^{\lambda} C_n^t f(x-1, k-tx, n-t) + f(x-1,k,n),$$

that is equivalent to the desired formula.

□

The relation (6.20) defines an alternative method of calculating the number of all permutations with the constant sum of non-negative elements bounded above by some non-negative value. This method is computationally more effective than the one defined by formula (6.18) as it involves deeper recursive steps.

Using the proposed statements, we will consider the problem of calculating the probability of the maximal element value in a random series with the known total number of votes.

**Theorem 6.8.** *Let* $(\xi_1, \xi_2, ..., \xi_n)$ *be a series of* $n \in \mathbb{N}$ *independent random variables uniformly distributed over the set* $\mathbb{D} = \{0, 1, 2, ..., m\}$, $m \in \mathbb{N}^0$. *Then the following statements are true:*

*a) the probability of the event that* $\max\{\xi_i\}_{i=1}^n = x$, $x \in \mathbb{D}$, *on conditions that* $\sum_{i=1}^{n} \xi_i = k$, $k \in \mathbb{N}^0$, *is defined by the formula:*

$$P_n^{k,m}(x) = \frac{p_n^k(x)}{f_n^k(m)}; \tag{6.21}$$

*b) the probability of the event that* $\max\{\xi_i\}_{i=1}^n \leq x$, $x \in \mathbb{D}$, *on conditions that* $\sum_{i=1}^{n} \xi_i = k$, $k \in \mathbb{N}^0$, *is defined as:*

$$F_n^{k,m}(x) = \frac{f_n^k(x)}{f_n^k(m)}. \tag{6.22}$$

*Proof.* Notice, that the probability mentioned in (a) equals to the ratio of the number of all $n$-permutations with the constant sum $k$ of non-negative elements and the maximal element $x$ to the total number of $n$-permutations with the constant sum $k$ of non-negative elements from the set $\mathbb{D}$. Taking into account Definitions (6.8) and (6.4) we obtain the desired expression. In the same manner using Definitions (6.6) and (6.4) we prove the statement (b).

□

In this context $F_n^{k,m}(x)$ has sense of the *cumulative distribution function* that corresponds to the *probability mass function* $P_n^{k,m}(x)$.

**Corollary 6.9.** *If in Theorem 6.8 the set* $\mathbb{D} = \mathbb{N}^0$, *the relations (6.21) and (6.22) has the following (unrestricted) view correspondingly:*

$$P_n^{k,\infty}(x) = P_n^k(x) = \frac{p_n^k(x)}{f_n^k}; \tag{6.23}$$

$$F_n^{k,\infty}(x) = F_n^k(x) = \frac{f_n^k(x)}{f_n^k}. \tag{6.24}$$

*And the cumulative distribution function* $F_n^{k,m}(x)$ *can be expressed using the unrestricted cumulative distribution function* $F_n^k(x)$ *as:*

$$F_n^{k,m}(x) = \frac{F_n^k(x)}{F_n^k(m)}. \tag{6.25}$$

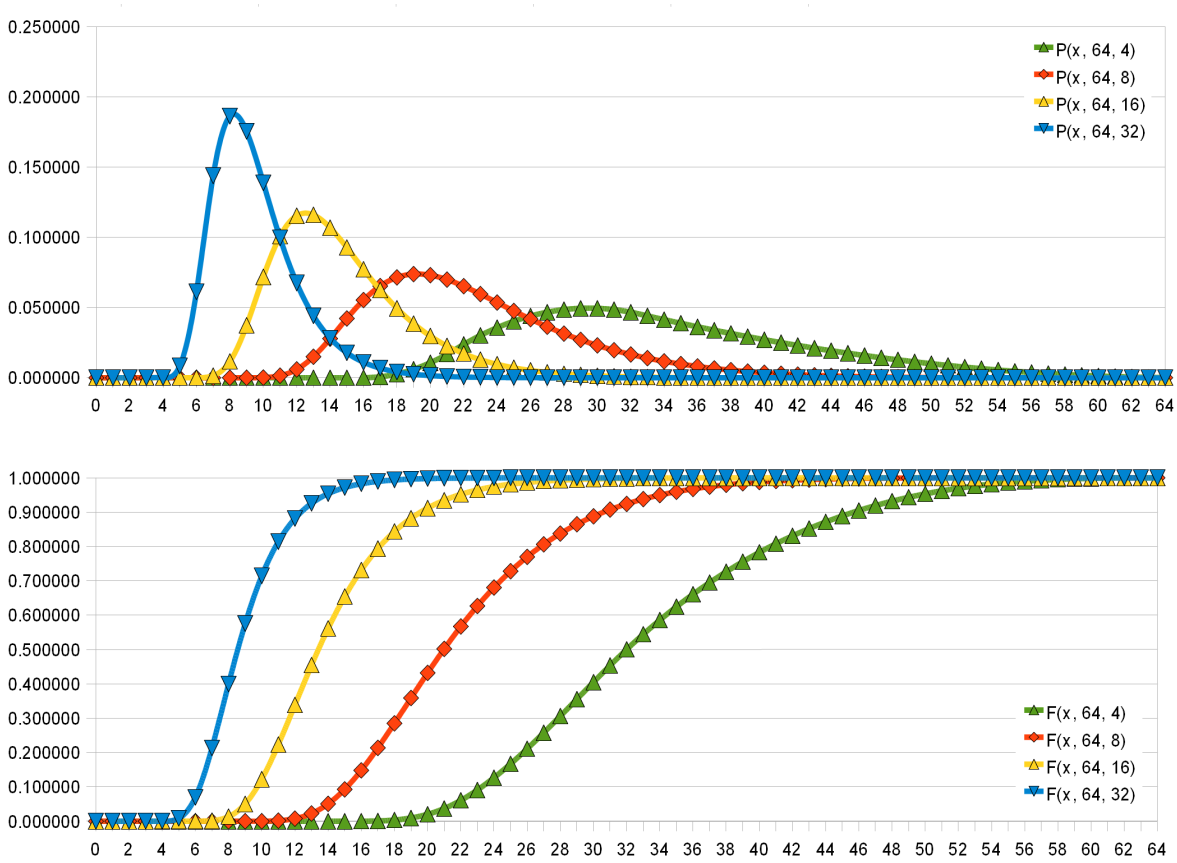


Figure 6.11.: Plots of the probability mass functions $P_n^k(x)$, $k = 64$, $n = 4, 8, 16, 32$ (top); and the corresponding cumulative distribution functions $F_n^k(x)$ (bottom).

**Corollary 6.10.** *For all* $x, k, n \in \mathbb{N}$*, such that* $k/2 < x \leq k$*, the relations (6.23) and (6.24) has the following non-recurrent forms respectively:*

$$P_n^k(x) = n\frac{f(k-x, n-1)}{f(k,n)}, \tag{6.26}$$

$$F_n^k(x) = 1 - n\sum_{t=x+1}^{k} \frac{f(k-t, n-1)}{f(k,n)}. \tag{6.27}$$

*Proof.* The first equation ensues from (6.23), Lemma 6.4 and Lemma 6.6. From the condition $k/2 < x \leq k$, follows that $\lambda = \lfloor k/x \rfloor = 1$, and accordingly to Lemma 6.6, $p_n^k(x) = \sum_{t=1}^{\lambda} C_n^t f(x-1, k-tx, n-t) = C_n^1 f(x-1, k-x, n-1) = nf(x-1, k-x, n-1)$.

Notice, that expression $x > k/2$ is equivalent to $x > k - x$, or $x - 1 \geq k - x$. So, using Lemma 6.4 we obtain $f(x-1, k-x, n-1) = f(k-x, n-1)$. Therefore, the fraction (6.23) has the desired form $P_n^k(x) = p_n^k(x)/f_n^k = nf(k-x, n-1)/f(k,n)$.

The second equation follows from the common formula for the cumulative distribution function, $F_n^k(x) = (f(k,n) - p(k,n) - p(k-1,n) - ... - p(x+1,n))/f(k,n) = 1 - \sum\limits_{t=k}^{x+1} P_n^k(t)$. □

## 6.3.4. Probability of the vote peak in a subset

Considering the proposed afore statements, we will return back to the problem (6.14) of finding the probability of the vote peak in a given subset of tree nodes. Here we will consider this problem in the context of tree $T_1$ construction. Each tree node $s \in S$ with the corresponding number of votes $v(s)$ can be considered as a series of its $q$-depth descendants $c_q(s)$ with the unknown numbers of votes. So, given at some iteration of $T_1$ construction procedure a set of leaf nodes $C = \{s_1, s_2, ..., s_K\}$, we can calculate the probability of the event that the number of votes of all $q$-depth descendants of $s_i$ is not greater than $x \in \mathbb{N}^0$ as $F_n^k(x)$, with $k = v(s_i)$, $n = |c_q(s_i)|$, using formula (6.24). It should be noted, that this statement is only true under the assumption of the uniform distribution of the votes.

**Definition 6.9.** *Let us denote* $P_{\widetilde{C},C} = P\left(\max\limits_{s \in c_q(\widetilde{C})} v(s) > \max\limits_{s \in c_q(C \setminus \widetilde{C})} v(s)\right)$, $\widetilde{C} \subseteq C$.

**Theorem 6.11.** *The probability* $P_{\widetilde{C},C}$ *can be calculated using the following formula:*

$$P_{\widetilde{C},C} = \sum_{t=1}^{V} (F_{\widetilde{C}}(t) - F_{\widetilde{C}}(t-1)) F_{C \setminus \widetilde{C}}(t-1), \tag{6.28}$$

*where for some set of nodes* $A$, $F_A(t) = \prod\limits_{s \in A} F_{|c_q(s)|}^{v(s)}(t)$, *and* $V = \max\limits_{s \in \widetilde{C}} v(s)$.

*Proof.* Notice that the event $\left\{\max\limits_{s \in c_q(\widetilde{C})} v(s) > \max\limits_{s \in c_q(C \setminus \widetilde{C})} v(s)\right\}$ in the current terminology is equivalent to the event

$$\left\{\bigcup_{t \in \mathbb{N}} \left(\max_{s \in c_q(\widetilde{C})} v(s) = t, \max_{s \in c_q(C \setminus \widetilde{C})} v(s) \leq t-1\right)\right\}.$$

Considering that the maximal values of the elements from $c_q(\widetilde{C})$ are limited above by the corresponding total number of votes $V$, the last expression is equivalent to

$$\left\{\bigcup_{1 \leq t \leq V} \left(\max_{s \in c_q(\widetilde{C})} v(s) = t, \max_{s \in c_q(C \setminus \widetilde{C})} v(s) \leq t-1\right)\right\}.$$

Therefore, the probability $P_{\widetilde{C},C}$ can be rewritten as

$$P_{\widetilde{C},C} = P\left(\bigcup_{1\le t\le V}\left(\max_{s\in c_q(\widetilde{C})} v(s) = t,\ \max_{s\in c_q(C\setminus\widetilde{C})} v(s) \le t-1\right)\right) =$$
$$= \sum_{t=1}^{V} P\left(\max_{s\in c_q(\widetilde{C})} v(s) = t,\ \max_{s\in c_q(C\setminus\widetilde{C})} v(s) \le t-1\right) =$$
$$= \sum_{t=1}^{V} P\left(\max_{s\in c_q(\widetilde{C})} v(s) = t\right)\cdot P\left(\max_{s\in c_q(C\setminus\widetilde{C})} v(s) \le t-1\right).$$

The above decomposition is valid as all the considered events are independent. From the independence of the events it also follows the next relation for the right-hand multiplier is performed:

$$P\left(\max_{s\in c_q(C\setminus\widetilde{C})} v(s) \le t-1\right) = \prod_{s_i\in C\setminus\widetilde{C}} P\left(\max_{s\in c_q(s_i)} v(s) \le t-1\right) = \prod_{s_i\in C\setminus\widetilde{C}} F^{v(s_i)}_{|c_q(s_i)|}(t-1) =$$
$$= F_{C\setminus\widetilde{C}}(t-1).$$

Similarly to this, we decompose the left-hand multiplier:

$$P\left(\max_{s\in c_q(\widetilde{C})} v(s) = t\right) = P\left(\max_{s\in c_q(\widetilde{C})} v(s) \le t\right) - P\left(\max_{s\in c_q(\widetilde{C})} v(s) \le t-1\right) =$$
$$= F_{\widetilde{C}(t)} - F_{\widetilde{C}}(t-1).$$

Substitution of these two decompositions into the equation for $P_{\widetilde{C},C}$ gives us the desired formula.

□

### 6.3.5. Approximation of the probabilistic model

In spite of the fact that the equations (6.19) and (6.20) together with Theorem 6.8 define a theoretical approach for calculating $F^k_n(x)$ and $P^k_n(x)$, their utilization for large $n$ and $k$ in real-time applications is hardly possible. This results from the big depth of the recursion that is involved into the calculation process, and therefore leads to the large stack sizes and long computational time. The values of the numerators and denominators in (6.24) can exceed the capacity of the integer types and require utilization of the floating-point or arbitrary-precision arithmetic.

These problems can be solved in two ways. The first one is to find a closed form solution of the partial recurrence equations of the variable order (6.19 - 6.20), and calculate $F^k_n(x)$ and $P^k_n(x)$ directly using the floating-point or arbitrary-precision arithmetic. The mentioned problem is non-trivial and it does not have a general solution scheme (with an exception of a number of special cases). The second way is to approximate $P^k_n(x)$ and $F^k_n(x)$ by some functions that can be calculated directly and fast. Such a solution is considered in the this section.

Using formulas (6.23 - 6.24), we analysed the probability mass function $P^k_n(x)$, and found that it can be interpolated with a high degree of accuracy (Table 6.4) by the probability density function (PDF) of the *Gumbel distribution* [82]:

$$g(x;\mu,\beta) = \frac{1}{\beta} e^{\frac{\mu-x}{\beta} - e^{\frac{\mu-x}{\beta}}}, \tag{6.29}$$

with some parameters ($\mu$, $\beta$), where $\mu \in \mathbb{R}$ is the location and $\beta \in \mathbb{R} > 0$ is the scale.

The Gumbel distribution (proposed by Emil Julius Gumbel in 1958 [82]) is used in probability theory and mathematical statistics to model the distribution of the maximal or minimal value of samples of normal or exponential distributions. It is one of the principle distributions applied in extreme value theory for modelling and predicting the probability of extreme natural disasters (e.g. earthquakes, floods and wildfires), and extreme values in several probabilistic models such as market risks and biological mutations.

| $P_n^k(x)$ | | $g(x;\mu,\beta)$ | | **Residuals** |
|---|---|---|---|---|
| $k$ | $n$ | $\mu$ | $\beta$ | **(RMS)** |
| 16 | 64 | 2.15 | 0.61 | 0.00367 |
| 32 | 64 | 3.36 | 0.83 | 0.00122 |
| 48 | 64 | 4.50 | 1.05 | 0.00015 |
| 64 | 64 | 5.60 | 1.26 | 0.00044 |
| 64 | 48 | 6.57 | 1.51 | 0.00066 |
| 64 | 32 | 8.31 | 1.96 | 0.00090 |
| 64 | 16 | 12.71 | 3.13 | 0.00103 |
| ... | ... | ... | ... | $<$0.01 |

Table 6.4.: Root mean square of the residuals after fitting $P_n^k(x)$ with the Gumbel probability density function $g(x;\mu,\beta)$. Sampled for several $(k,n)$ and the corresponding best fit parameters $(\mu,\beta)$.

Having the experimental results (Figure 6.12 - 6.13), we estimated the best fit parameters ($\mu$, $\beta$) for the test range of pairs $(k,n)$, $k \leq 128$, $n \leq 1024$, by solving the optimization problem:

$$[\mu(k,n),\beta(k,n)] = arg\min_{\mu,\beta} ||P_n^k(x) - g(x,\mu,\beta)||,$$

and obtained two sets of samples for the functions $\mu(k,n)$ and $\beta(k,n)$ (Figure 6.14 - 6.15). After analysing these sets, it was proposed to approximate $\mu(k,n)$ and $\beta(k,n)$ using the following class of functions:

$$c(k,n) = a \cdot kn^q + b, \tag{6.30}$$

with $a,b \in \mathbb{R}$, and $q \in \mathbb{R} < 0$.

In order to find the optimal approximation parameters, the following minimization problem over the mentioned above range of pairs $(k,n)$ was solved:

$$\sum_{k,n} ||g(x;a_1kn^{q_1}+b_1,a_2kn^{q_2}+b_2) - P_n^k(x)|| \xrightarrow[\substack{a_2,b_2,q_2}]{a_1,b_1,q_1} min.$$

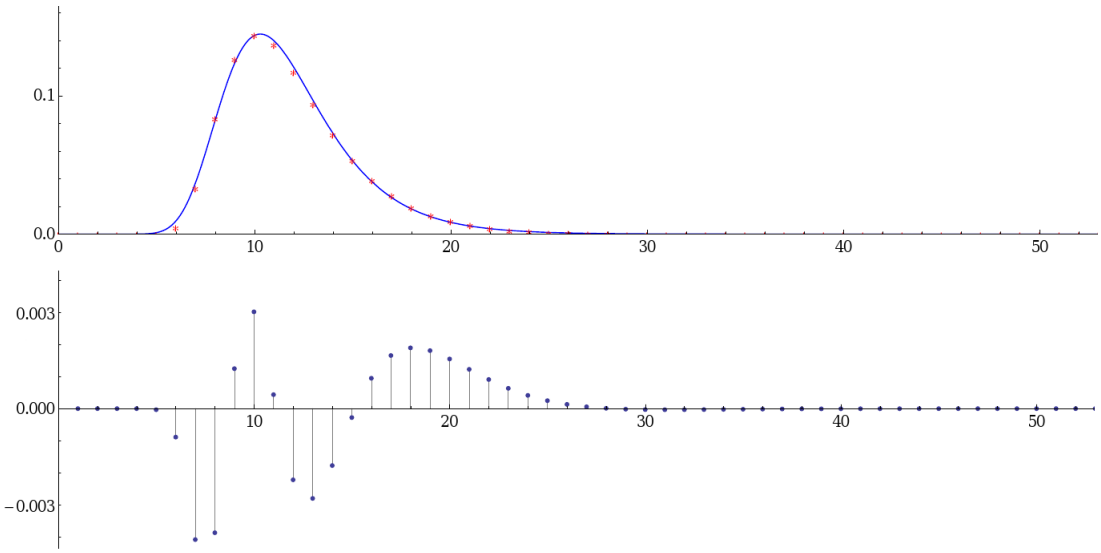


Figure 6.12.: An example of the residuals distribution after fitting $P_n^k(x)$, $k = 53$, $n = 17$ (marked by asterisks) with $g(x; \mu, \beta)$, $\mu = 10.29$, $\beta = 2.55$.

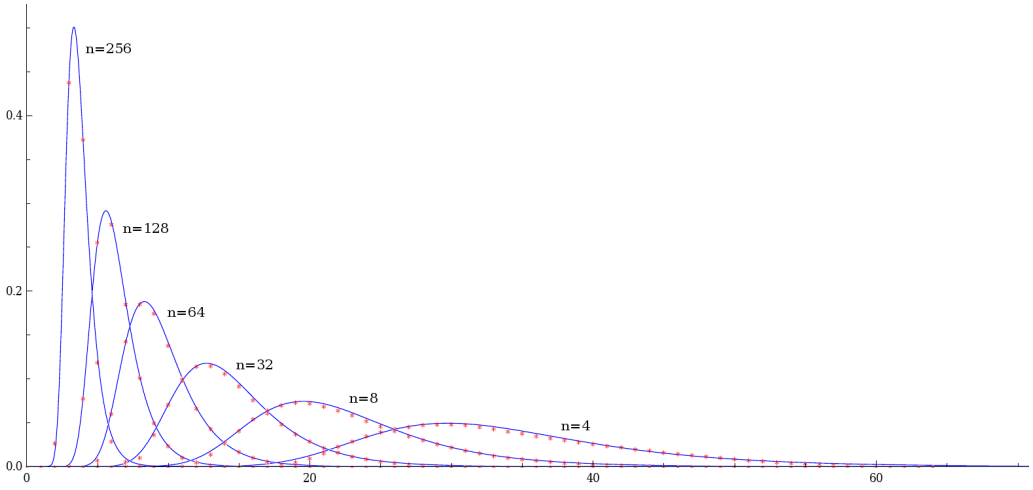


Figure 6.13.: A series of $P_n^k(x)$, $k = 64$, $n = 4, 8, 32, 64, 128, 256$ (marked by asterisks) and their corresponded Gumbel probability density functions.

So, the obtained optimal approximation parameters for the Gumbel PDF have the next view:

$$\begin{aligned} \mu(k,n) =& 1.16\ kn^{-2/3} + 1.00, \\ \beta(k,n) =& 0.40\ kn^{-4/5} + 0.32, \end{aligned} \tag{6.31}$$

that results in the average approximation error $E = \frac{1}{KN} \sum\limits_{k,n} \| P_n^k(x) - \widetilde{P}_n^k(x) \| = \frac{1}{KN} \sum\limits_{k,n} \sum\limits_{x=0}^{k} |P_n^k(x) - \widetilde{P}_n^k(x)| = 0.055$.

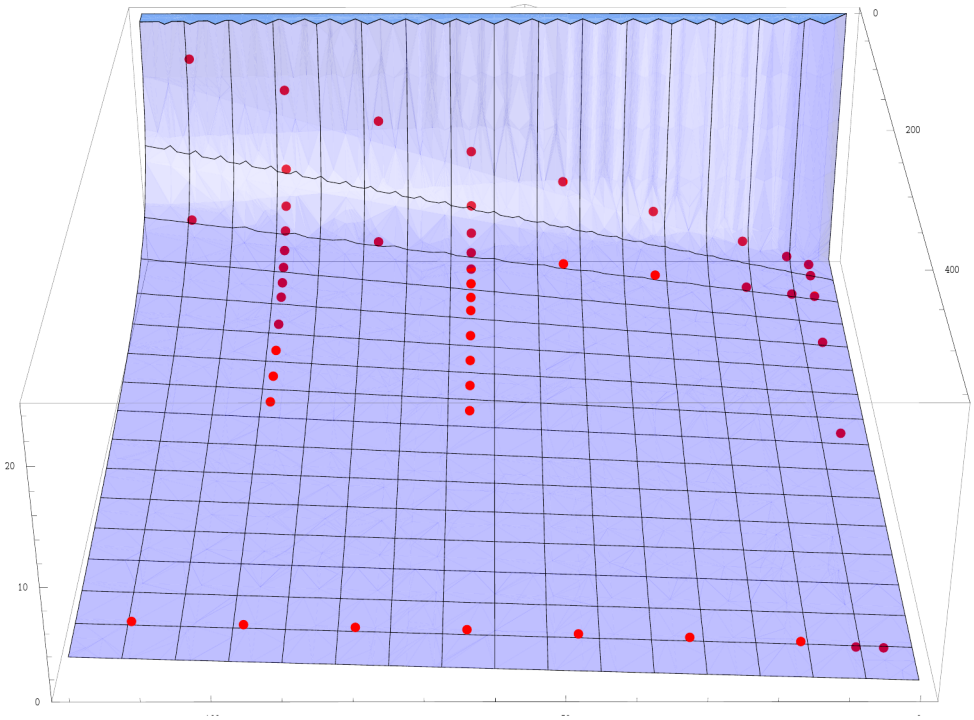


Figure 6.14.: The samples of the best fit Gumbel PDF parameter $\mu$ and the corresponding approximating surface $\mu(k,n) = 1.16\ kn^{-2/3} + 1.00$.

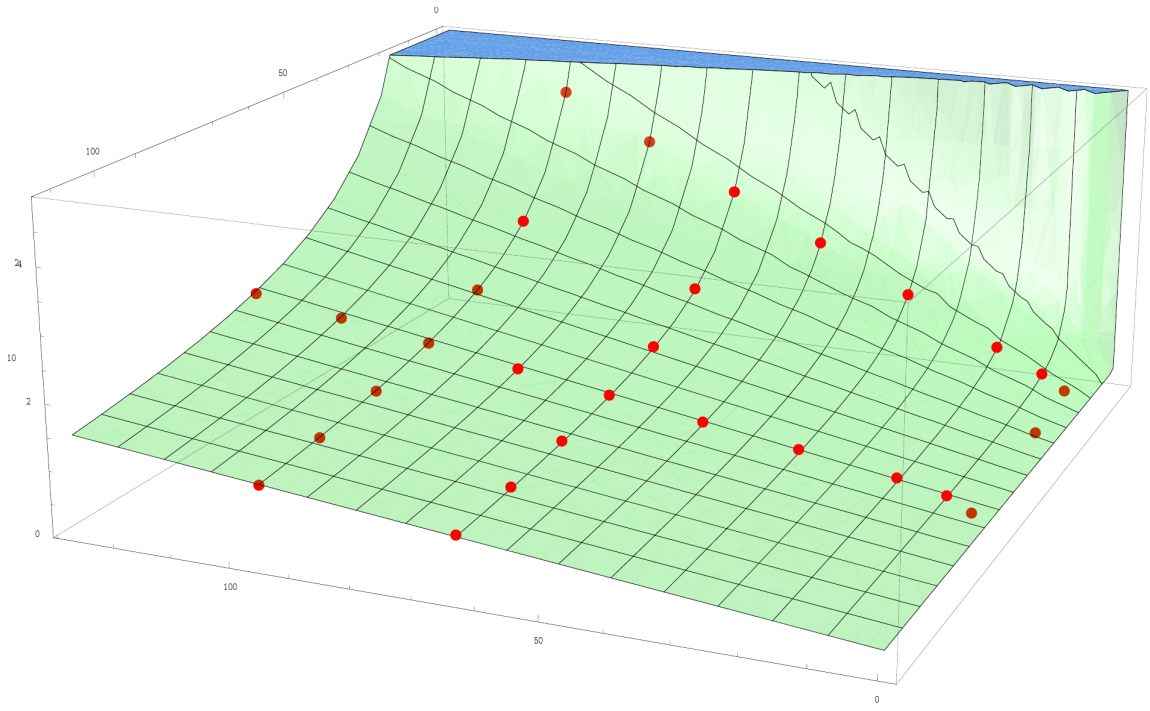


Figure 6.15.: The samples of the best fit Gumbel PDF parameter $\beta$ and the corresponding approximating surface $\beta(k,n) = 0.40\ kn^{-4/5} + 0.32$.

The distribution of the approximation error in terms of $(k, n)$ is shown on Figure (6.16). The experiments shown that the error decreases with growth of the parameters $k$, $n$ and the relation $k/n$.

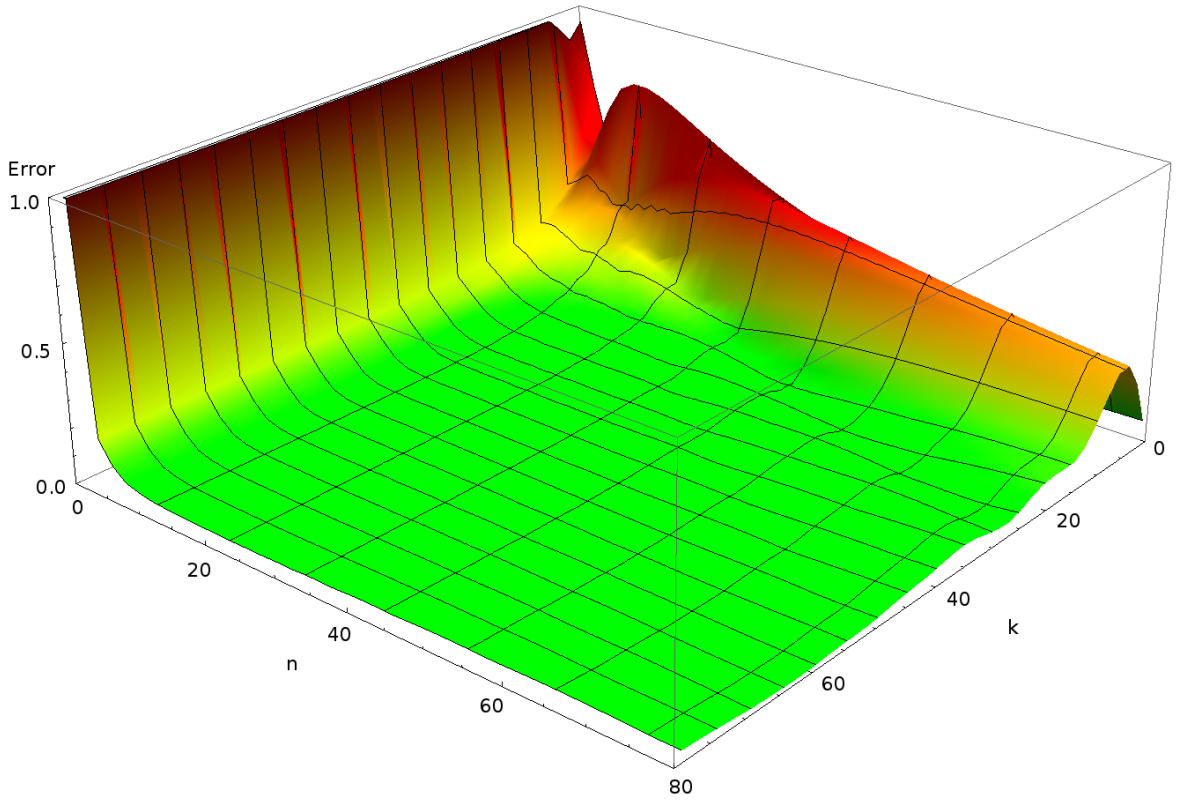


Figure 6.16.: Dependence of the $P_n^k(x)$ approximation error $E = \|P_n^k(x) - \widetilde{P}_n^k(x)\|$ from the parameters $k, n \leq 80$.

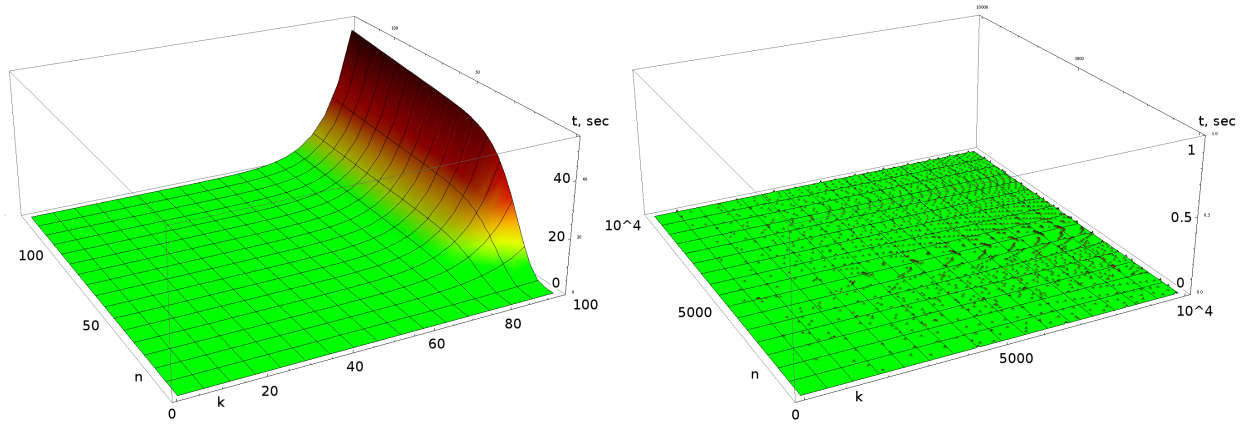


Figure 6.17.: Dependence of the time (in seconds) needed for $F_n^k(x)$ calculation (on an Intel Core i7 2.8GHz workstation) from the parameters $k$ and $n$. Left: using the exact recursive formula (6.24), $k \leq 96$, $n \leq 128$. Right: using the approximate formula (6.32), $k, n \leq 10^4$.

Summarizing the above, we propose to use the following approximation of the probability mass and the cumulative distribution functions and extrapolate it over all $k, n > 0$:

$$\begin{aligned} P_n^k(x) &\simeq \widetilde{P}_n^k(x) = \frac{1}{\beta(k,n)} e^{\frac{\mu(k,n)-x}{\beta(k,n)} - e^{\frac{\mu(k,n)-x}{\beta(k,n)}}}, \\ F_n^k(x) &\simeq \widetilde{F}_n^k(x) = e^{-e^{\frac{\mu(k,n)-x}{\beta(k,n)}}}, \end{aligned} \tag{6.32}$$

with $\mu(k,n)$ and $\beta(k,n)$ defined by (6.31).

Usage of the approximations (6.32) instead of the recurrent relations provided by Theorem 6.8 exponentially decreases the computational time of $P_n^k(x)$ and $F_n^k(x)$ and provide reasonable accuracy for large values of $k$ and $n$ (Figure 6.16 - 6.17).

### 6.3.6. Optimal subdivision strategy for the hierarchical vote analysis

Now, using Theorem 6.11 and the direct formulas (6.32), we can effectively solve the problem (6.13) of selecting a minimal subset $\widetilde{C}$ from a given set of tree nodes $C = \{s_1, s_2, ..., s_N\}$, $s_i \in S$, such that for a predefined value $\Omega \in [0;1]$, condition $P_{\widetilde{C},C} \geq \Omega$ is performed.

First, let us introduce the *comparison operator* for any two tree nodes $s_i$ and $s_j$. We will say that $s_i < s_j$, if the average vote peak in $c_q(s_i)$ is less than in $c_q(s_j)$. Considering the probability mass function (6.23) the average vote peak can be expressed as:

$$E_n^k = \sum_{x=0}^{\infty} P_n^k(x) x.$$

This sum cannot be easily calculated analytically as the closed form solution of the partial recurrence equation (6.19) for $p_n^k(x)$ is unknown. But, similarly to formulas (6.32), it can be approximated with the mathematical expectation of the corresponding Gumbel distribution:

$$E_n^k \simeq \widetilde{E}_n^k = \mu(k,n) + \lambda\beta(k,n), \tag{6.33}$$

where $\lambda \approx 0.5772$ is the Euler–Mascheroni constant.

**Definition 6.10.** *We will say that $s_i < s_j$, $s_i, s_j \in S$, if an only if $E_{|c_q(s_i)|}^{v(s_i)} < E_{|c_q(s_j)|}^{v(s_j)}$.*

This gives us a convenient tool for comparing tree nodes with the different number of votes and descendants (i.e. the nodes with the different node depth). It also should be noted, that for any $s_1, s_2 \in S$, such that $d(s_1) = d(s_2)$, relation $s_i < s_j$ is equivalent to $v(s_1) < v(s_2)$. This quite an intuitive result follows from the linearity of functions $\mu(k,n)$ and $\beta(k,n)$ relatively to $k$.

Using the introduced comparison operator and the formula for the probability of the vote peak in a subset (6.28), the optimal hierarchical subdivision strategy (rule B) can be implemented by the following algorithm:

**Algorithm 1**

---

*a)* Sort the nodes in the initial set $C$ in descending order accordingly to the comparison operator.

*b)* In the sorted set $C$ find the limiting element $s_n$, that divides $C$ into two subsets: $\widetilde{C}$ (starting from the beginning) and $C \backslash \widetilde{C}$, such that $P_{\widetilde{C},C} \geq \Omega$.

---

**End.**

Step (a) needs in average $O(N \log N)$ comparison operations, while step (b) can be performed using the binary search that requires in average $O(\log N)$ evaluations of function $P_{\widetilde{C},C}$.

After $T_1$ construction is finished, we can assign to each edge $e = e(s, s') \in E_1$ the following weight:

$$\omega(e) = P_{\{s'\},c(s)}.$$

It defines the probability that the vote peak occurs in the $q$-depth descendants of $s'$ and is not in the descendants of the other nodes from $c(s)$ (Figure 6.9).

In order to estimate the probability of the global vote peak occurrence in the $q$-depth descendants of a particular node $s \in S$, one has to calculate the product of all edge weights from the path from $s$ to the tree root $s_0$:

$$\Omega(s) = \prod_{e \in [s_0;s]} \omega(e).$$

For estimation of the quality of the final solution provided by the method, we calculate the sum of the global peak occurrence probabilities for all leaf nodes $s \in S_1$ from the last tree $T_1$ level $d(s) = M$:

$$\Omega^* = \sum_{s \in S_1, d(s)=M} \Omega(s).$$

The value $\Omega^*$ acts as the algorithm's accuracy measure and defines the lower estimate of the probability that the unknown global vote peak is presented in the solution.

## 6.4. Results and discussion

We evaluated the version of the method based on the *approximate vote analysis* on cardiac MR images. First, a template model was built and encoded using two reference points that were chosen manually inside the model (Figure 6.18). Before the object detection, the boundary points were extracted using the Canny edge detector [29].

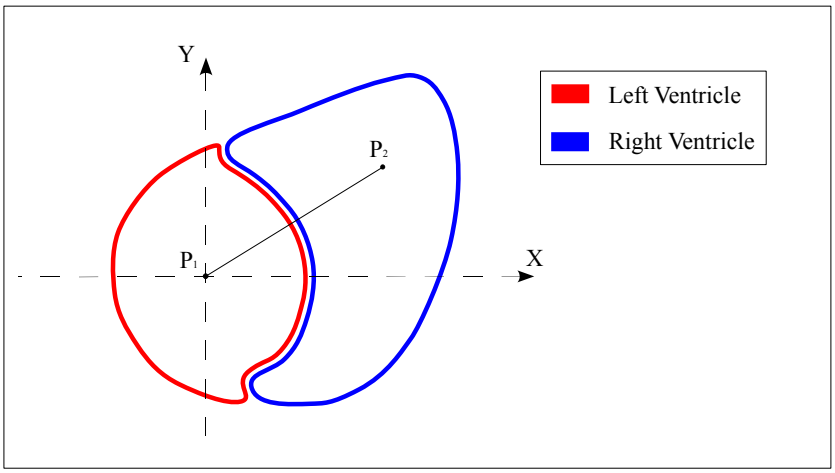


Figure 6.18.: Template model of the human heart (short axis).

The method makes use of the boundary tangent information that reduces the number of false evidences and increases the speed of the voting process. However, the accuracy of the tangent vector calculation must be provided on a high level in order to avoid voting errors and false object detection. In our experiments the tangent vectors for the extracted boundaries were calculated using the 2D principal component analysis (PCA) [151, 78].

The accuracy and the computational complexity of the algorithm based on the approximate vote analysis depend on the threshold value $\tau$ that defines the sizes of projection sets $\mathbb{Q}_1(\tau)$ and $\mathbb{Q}_2(\tau)$ (see Section 6.3.2). Using the high values of $\tau$ we reduce the projection sets and, therefore, decrease the computational time, but on the other hand the probability of the vote peak omission is increased. In our experiments we took the threshold value from the interval 75% - 95% of the corresponding maximal projection value (Figure 6.22).

A combination of both the Hough paradigm and the direct RT-like vote calculation makes the method low memory consumptive and convenient for implementation on a GPU. Low memory usage is partially reached at the expense of the increase of the computational time. Of course, the standard grid vote analysis may provide a better speed, but it is hardly suitable for implementation on GPUs (and even on conventional workstations) because of its extremely high memory usage.

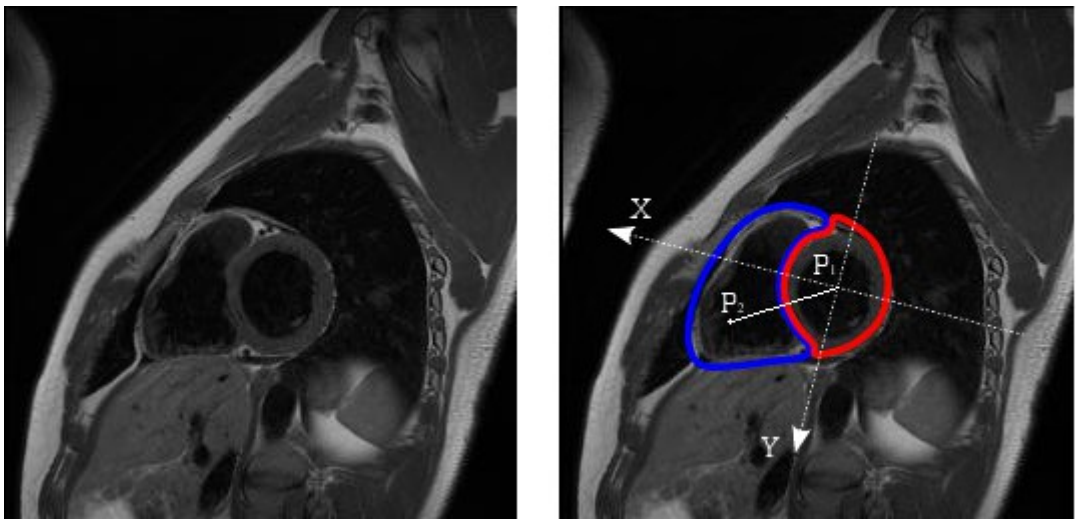


Figure 6.19.: Example of the DPGHT-based heart walls detection on cardiac MRI (a short axis view of the human heart, $256 \times 256$).

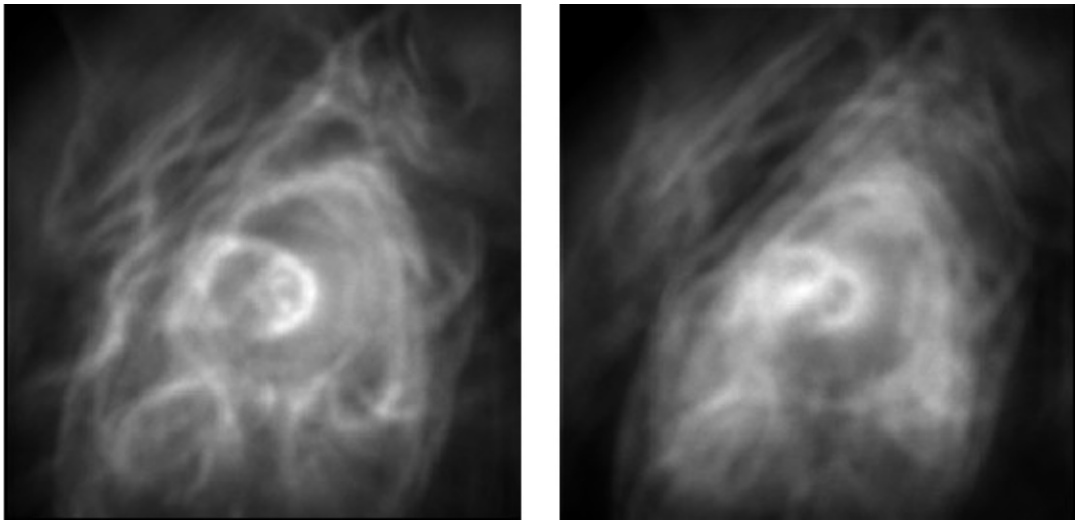

Figure 6.20.: The orthogonal cumulative projections $F_{1,C}(I, P_1)$ (left) and $F_{2,C}(I, P_2)$ (right) of the DPGHT applied to the MRI data from the Figure 6.19.

We parallelized the method on a GPU (GeForce 8800GTX) using nVidia CUDA, that increased its speed in a factor of more than 50 comparing to a CPU version (on a Pentium 4 3.0GHz workstation).

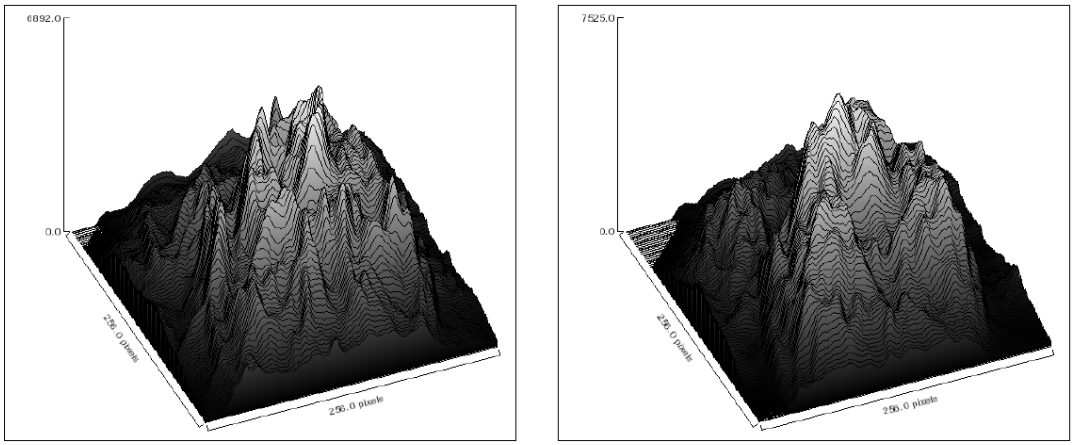


Figure 6.21.: Surface plots of the projections $F_{1,C}(I, P_1)$ (left) and $F_{2,C}(I, P_2)$ (right) of the DPGHT applied to the MRI data from the Figure 6.19.

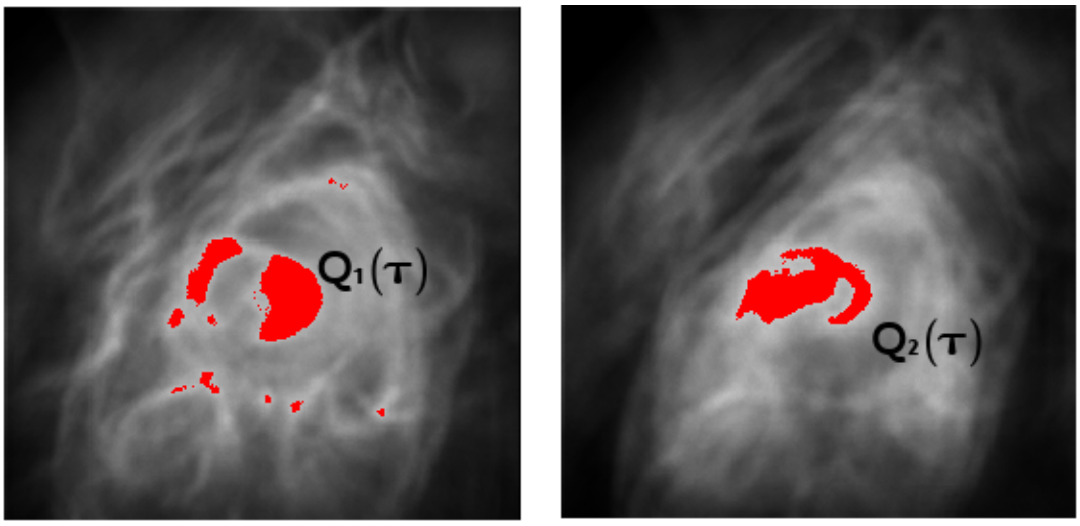


Figure 6.22.: The areas of the projections $F_{1,C}(I, P_1)$ (left) and $F_{2,C}(I, P_2)$ (right) with the most probable reference point location.

The *hierarchical vote analysis* model, in turn, was evaluated on synthetic data. We generated a set $S$ of random series with the different number of uniformly distributed elements, $S = \{S_1, S_2, ..., S_K\}$, $S_i = (s_{i1}, s_{i2}, ..., s_{in_i})$ (see Section 6.3.3). Using this, the accuracy of Algorithm 1 and formula (6.28) for $P_{\widetilde{S},S}$ were tested in the following way.

For each value $P_{opt}$ sampled on $[0; 1]$ with a small step, we run Algorithm 1 and found the minimal subset $\widetilde{S} \subseteq S$, such that $P_{\widetilde{S},S} \geq P_{opt}$. We stored the size $n$ of this $\widetilde{S}$ and checked if the actual global maximum was in $\widetilde{S}$. After repetition of this procedure $N$ times for random data we estimated: a) the dependence of the average size of $\widetilde{S}$ from $P_{opt}$ (Figure 6.23b); b) the actual $P_{\widetilde{S},S}$ for each value $P_{opt}$ (Figure 6.23a, curves $P_{opt1}$ and $P_{opt2}$ for different random generators). The experiments showed that the estimations of $P_{\widetilde{S},S}$ are close to $P_{opt}$ (comparing curves $P_{opt1}$, $P_{opt2}$ and $P_{opt}$ on Figure 6.23a) and, therefore, we ascertained that Algorithm 1 is valid.

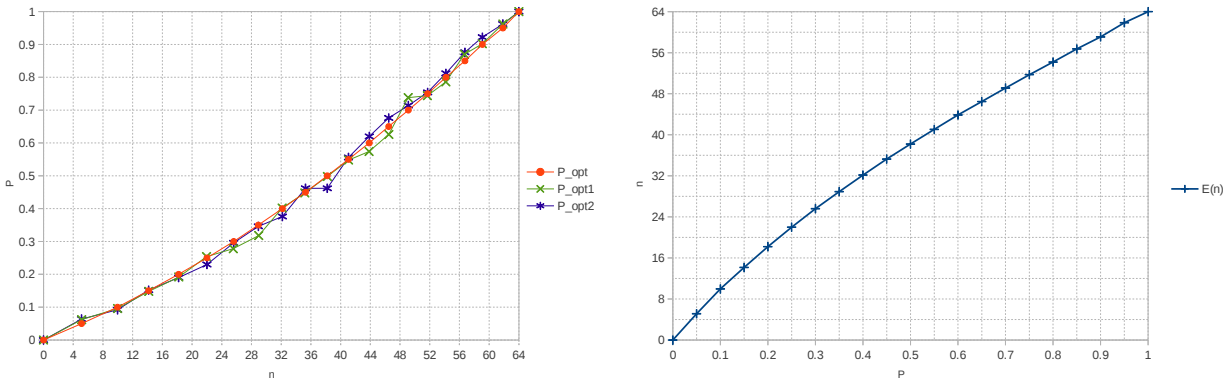


Figure 6.23.: a) Left: comparing the estimated $P_{\widetilde{S},S}$ (curves $P_{opt1}$ and $P_{opt2}$ for different random generators) with threshold $P_{opt}$, such that $P_{\widetilde{S},S} \geq P_{opt}$. b) Right: the dependence of the average size of $\widetilde{S}$ from $P_{opt}$ ($P_{\widetilde{S},S} \geq P_{opt}$). $K = 64$, $|S_i| = 32$, $s_{ij} \leq 1024$, $N = 1000$.

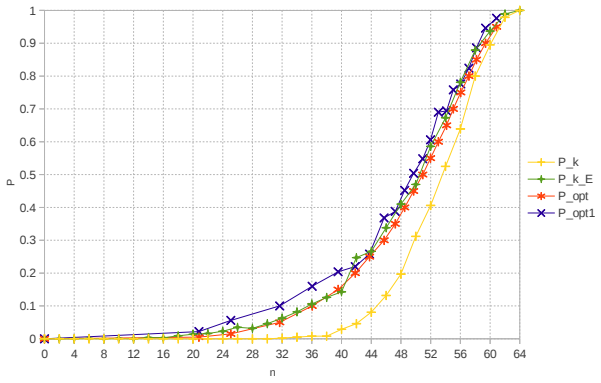


Figure 6.24.: Comparing the optimal strategy (Algorithm 1, $P_{opt}$ and $P_{opt1}$) and the standard strategy (taking $n$ series with the biggest sum of elements, $P_k$). Optimized standard strategy (taking $n$ "biggest" series from $S$ accordingly to the introduced comparison operator, $P_{kE}$).

In spite of these promising results, the further investigation is needed for the proposed hierarchical vote analysis model as well as for the object detection based on it. That is the priority direction for the future work.

# 7. Summary and outlook

In this work we presented two prototypes of the scanning systems based on tracked 3D ultrasound, developed a method for 3D surface reconstruction from noisy and sparse data, and considered the problem of the GHT-based 2D object detection.

The both developed scanning systems showed promising results. Although the first one based on the freehand ultrasound had lower accuracy and precision, it reflected the general feature of the freehand approach (more mobility and scanning freedom at the expense of accuracy). It should be noted, that the validation results were obtained in the close to ideal conditions using a specially designed scanning box and may differ from the results acquired from patients. Therefore, further validation in clinical conditions is obviously needed.

The scanning system based on the sequential triggered scanning approach, in turn, showed high accuracy and precision that is comparable to the state of the art medical imaging modalities. This system has a high potential to be used in clinical practice for X-ray-free diagnostics of fractures of the human extremities, providing numerical data and 2D/3D visualization of bone surfaces for optimal decision making. However, some problems such as automation of the scanning process (the tested prototype was motor-free) and proper extremity fixation during scanning are still actual.

The developed mesh-growing surface reconstruction method was tested within the mentioned scanning systems. It showed appropriate reconstruction quality and good performance, so it can be successfully applied to noisy and sparse data produced by tracked 3D ultrasound scanners. However, the method is relatively sensitive to variations of the input parameters that are, in turn, defined by the properties of input data (e.g. sparseness, surface thickness, noise, etc.). Therefore, the appropriate values of the parameters must be carefully selected before reconstruction. Otherwise, the method can produce artefacts such as holes and mesh overlaps. It should be also noted that in spite of the specialization of the method to noisy and sparse data, it can be applied to arbitrary point clouds. A detailed comparison study with the existing general-purpose surface reconstruction methods is therefore needed. A variation of the mesh-growing method that is better intended for parallelization and application to sliced data is an objective for the future work.

The second part of the work concerns 2D object detection and has a more apparent theoretical orientation. However, several practical modifications of the standard GHT-based approach were proposed. Utilization of the two-point parametrization instead of the single-point one is a convenient tool for introduction of rotation-scale invariance into object detection. Together with an optimal vote analysis principle it can increase effectiveness and flexibility of the method. During the study of the hierarchical vote analysis several interesting problems were solved, including elaboration of the optimal space subdivision strategy based on the Gumbel probability distribution and calculation of volumetric measure of four-dimensional hyperplane - hyperrectangle intersections.

Although medical object detection was not integrated into the developed scanning systems, it can be potentially used in future extensions of the systems for automatic detection of specific anatomical structures. For example, detection of the left ventricle of the heart in the freehand tracked system in pseudo-3D or model-based modes. Anyway, before its practical application a more detailed evaluation study of the proposed method as well as the hierarchical space subdivision strategy is needed. Moreover, the probabilistic properties of the hierarchical space subdivision were investigated on assumption that the votes distribution is uniform, however the case of non-uniform distributions is of great interest. It also should be noted that the GHT-based object recognition may be a suboptimal choice for some medical applications due to high geometric variations of anatomical structures. The deformable-template models therefore appear to be a more suitable approach in this case. Nevertheless, the method can be used in the tasks where the shape of target objects is more rigid (e.g. industry, some computer vision tasks).

# A. Kd-tree based distance computation between meshes and point clouds in 3D

The proposed algorithm is based on a kd-tree search and a prior information from a static mesh. The problem of calculating the closest face of a mesh for an arbitrary point in 3D is well investigated. Besides the brute force methods, the popular approaches are based on Voronoi diagrams [13], decompositions into convex shapes [76], octrees and kd-trees [169], and multiresolution hierarchy of bounding volumes (e.g. the Meshsweeper algorithm [81]).

For an object containing many polygons computing a Voronoi diagram or a decomposition into convex shapes can be quite a complex problem. The methods based on the spatial data structures (octrees, kd-trees and bounding volumes) are considered to be more effective and in many cases can provide better performance [81]. In a general case, the spatial data structures like kd-trees and octrees have such a drawback that the closest vertex in the structure and the closest point on a polygonal mesh may be very distant from one another, therefore an extended search is needed. However, if some additional information is known for the mesh (such as constrains on the polygon size in our case) a convenient kd-tree can become a very effective toolkit for point-to-mesh distance computation. Therefore, for our purposes we use an algorithm based on kd-trees. In order to simplify the point-to-mesh distance calculation procedure we used an intuitive approach that is described below.

Let us define the *shape radius* for a set of points $X = \{x_i\}$ in 3D as the maximal distance from the points to their mass center $\bar{x}$:

$$S(X) = \max_{x_i \in X} \|x_i - \bar{x}\|,$$
$$\bar{x} = \frac{1}{|X|} \sum_{x_i \in X} x_i.$$

For a given triangular mesh $M$ we calculate the maximal shape radius $S$ of its faces and construct a kd-tree $T$ in 3D filling it with the mass centers of the mesh faces. The following intuitive Lemma is utilized for the point-to-mesh distance calculation:

**Lemma A.1.** *Let $M$ be a mesh consisted of convex polygonal faces $F = \{F_i\}$ with the maximal shape radius $S = \max_{F_i \in F} S(F_i)$ and $Z = \{z_i\}$ be the set of the corresponding mass centers of $F$. For a given point $p \in R^3$ let $d$ be the minimal distance from $p$ to $Z$ i.e. $d = \min_{z_i \in Z} \|p - z_i\|$. If $F^* \in F$ is the closest face for $p$ then the following relation is*

*performed for its mass center* $z^* \in Z$*:*

$$\left\|p - z^*\right\| \leq d + S \tag{A.1}$$

*Proof.* Let us suppose the opposite and $\left\|p - z^*\right\| > d + S$ , then for all points on the closest face $x \in F^*$ the following relation is true: $d+S < \left\|p - z^*\right\| = \left\|p - x + x - z^*\right\| \leq \|p - x\| + \left\|x - z^*\right\| \leq \|p - x\| + S$. That is equivalent to $\|p - x\| > d, \forall x \in F^*$. Taking into account that $\min\limits_{z_i \in Z} \|p - z_i\| = d$ and that the faces $F$ are convex (i.e. all $z_i \in F_i$), there exist $F_k \in F$ with $\tilde{x} = z_k \in F_k$ such that $\|p - \tilde{x}\| = d$. From this follows that $F^*$ is not the closest face that is a contradiction. The Lemma is proven.

□

Considering the relation (A.1) the point-to-mesh distance calculation can be done in three steps using two queries to the built kd-tree $T$.

**Algorithm 2**

---

For any point $p \in R^3$:

1) Find the closest element from $T$ and calculate distance $d$ to it;

2) Find all elements $\{z_j\}$ from $T$ within the range $[d, d + S]$ and define the corresponding candidate faces $\{F_j\}$;

3) Calculate the closest face from the set of candidates $\{F_j\}$.

---

**End.**

In the case of high amount of regular faces in the mesh and large point clouds, such a technique provides a significant speed gain comparing to the brute force methods (exhaustive search) and application of general data structures. *Regularity* of a mesh is defined here as the relation $r = \frac{A}{S}$ , where $A$ is the average shape radius and $S$ is the maximal shape radius of the mesh faces. It should be noticed that the complexity of the algorithm is minimal for absolutely regular meshes with $r = 1$ and it grows up with decrease of $r$ approaching the complexity order of the brute force method.

The maximal shape radius $S$ of the processed mesh can be either calculated directly or can be known from the external sources such as specific restrictions for a mesh construction procedure. In our case the maximal shape radius of the reconstructed meshes is inherently defined by the parameters of the mesh reconstruction algorithm ($S = \Delta_{max} + \Delta_p$). Direct computation of $S$ is worth in the case of large point clouds and inefficient in the case of mesh-to-single-point distance calculation, when the brute force method is even more preferable.

# B. Volume of a hyperplane - hyperrectangle intersection

The proposed algorithm for computing the volume of a hyperplane - hyperrectangle intersection consists of two steps: 1) calculate the boundaries of the intersection; 2) calculate the volume of the geometric body constrained by the found boundaries.

As any intersection of a convex set in the n-dimensional space with a hyperplane in this space (a flat subset with dimension n - 1) is a convex set, all intersections of the four-dimensional hyperrectangles $S_i^j$ with the corresponding hyperplanes (Figure B.1) define convex three-dimensional polyhedrons orthogonal to the normals of these hyperplanes. The boundaries of these polyhedrons form the convex hull of the intersection points of the hyperrectangles edges with the hyperplane. So, first we calculate these points using a naive method [120] as follows. Let $H = (P, \vec{n})$ be a hyperplane, $Q_1$ and $Q_2$ define the adjacent vertices of a hyperrectangle $S$ in 4D such that $\vec{e} = \vec{Q_1 Q_2}$ form one of 32 edges of $S$. The intersection point of $H$ and the line containing $\vec{e}$ is defined by the formula:

$$x = Q_1 + t\ \vec{e}, \tag{B.1}$$

where $t = (\vec{n} \cdot \vec{Q_1 P})\ /\ (\vec{n} \cdot \vec{e})$. If $t \in [0, 1]$ then the edge $e$ intersects with the hyperplane $H$ in the point $x$. After testing all the edges of $S$ for intersections with $H$ we obtain the set of points $X = \{X_i\}_{i=0}^{m}$, $m < 32$, that satisfy equation (B.1) with $t \in [0, 1]$ and define the boundaries of the desired intersection.

Having the set $X$ we compute the volume of its convex hull. The most straightforward way to do this is to project $X$ into 3D (as the intersection belongs to some three-dimensional subspace of the parameter space), and to calculate the volume using one of the common methods. For example, using *Delaunay triangulation* (tetrahedralization) [50] of $X$ with further summation of the resulting tetrahedra volumes [18], or calculating the faces of the convex hull for $X$ in 3D and applying the volume formula based on the *Gauss-Ostrogradsky's divergence theorem* [70].

The 3D projection of the intersection points is calculated using the following idea. Having the hyperplane $H = (P, \vec{n})$, $\vec{n} = (n_1, n_2, n_3, n_4)^T$, $|\vec{n}| = 1$, and the set $X$ in 4D we need to calculate its decomposition in the coordinate system of $H$. One of the axes of this coordinate system must be parallel to $\vec{n}$ and the coordinate center must coincide with $P$. We can define such a decomposition by the transformation consisted of the translation vector $P$ and the following skew-symmetric rotation matrix:

$$T(\vec{n}) = \begin{pmatrix} n_4 & -n_3 & n_2 & n_1 \\ n_3 & n_4 & -n_1 & n_2 \\ -n_2 & n_1 & n_4 & n_3 \\ -n_1 & -n_2 & -n_3 & n_4 \end{pmatrix}.$$

Indeed, it is easy to check that $T(\vec{n})$ transforms the canonical orthonormal basis into the orthogonal basis the fourth vector of which is equal to $\vec{n}$.

Let $Y_i$ be the decomposition of $X_i$ in the considered coordinate system defined by $T(\vec{n})$ and $P$. Then the following relation is performed:

$$Y_i = T^{-1}(\vec{n})(X_i - P), \tag{B.2}$$

where the inverse matrix $T^{-1}(\vec{n})$ is explicitly represented as:

$$T^{-1}(\vec{n}) = \frac{1}{|\vec{n}|^2}(2\ diag(n_4) - T(\vec{n})) = 2\ diag(n_4) - T(\vec{n}).$$

So, for each $X_i$, $i \in [0, m]$, we calculate its decomposition $Y_i = (y_1, y_2, y_3, y_4)^T$ defined by (B.2) and obtain the corresponding 3D projection as $Y_i' = (y_1, y_2, y_3)^T$.

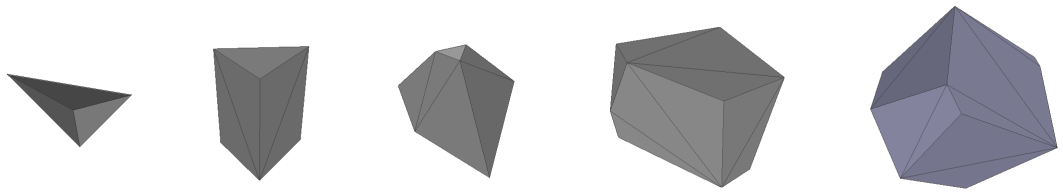

Figure B.1.: Examples of possible polyhedrons generated during intersection of a unit four-dimensional hypercubecube (tesseract) with a random hyperplane. The resulted polyhedrons consist of 4, 6, 8, 10 or 12 vertices (from left to right).

In order to compute the volume of the convex polyhedron defined by the vertices $\{Y_i'\}_{i=0}^m$ we propose to compute the faces of its convex hull $\Omega$ (e.g. using the incremental algorithm [106]) and apply the divergence theorem with the special form of the vector field $\vec{F}(x) = \frac{1}{3}(x_1, x_2, x_3)^T$, whose divergence is equal to 1:

$$\begin{aligned} \int_\Omega \nabla\vec{F}(x)\ d\Omega &= \oint_S \vec{F}(x) \cdot\ \vec{n}\ dS, \\ \int_\Omega d\Omega &= \frac{1}{3}\oint_S \vec{x}\cdot\vec{n}\ dS, \\ Volume(\Omega) &= \frac{1}{3}\sum_{F_i} \vec{b_i}\cdot\vec{n_i}\ A_i, \end{aligned} \tag{B.3}$$

where $F_i$ is i-th face of the convex hull, $b_i$ and $n_i$ are its barycentre and the outer-pointing normal vector correspondingly, and $A_i$ is the area of face $F_i$. The complexity of the proposed volume computation algorithm is $O(n^2)$, where $n$ is the number of the edges in the initial four-dimensional polytope (in our case $n = 32$).

# C. Bö: C++ library for image processing and reconstruction

All presented in this work algorithms have been implemented as a part of a program library, called Bö, developed by our research group. Bö is a collection of basic and specialized methods and data structures for geometrical processing and 2D/3D image analysis including image segmentation, surface reconstruction, mesh processing, object detection as well as basic linear algebra and I/O functionality. Written entirely in C++, the library provides loosely-coupled but mutually compatible modules for a number of problems in computer vision.

The main features of the library:

- Generic representation of 2D images.
- Linear filtering for 2D images.
- Triangular mesh with support of export/import to PLY file format (Stanford Triangle Format).
- Mesh-growing surface reconstruction method.
- Modified Butterfly subdivision surface method.
- Iterative closest point (ICP) registration method.
- Object detection based on the Hough Transform (HT) and the Generalized HT.
- Highly customizable Markov random field (MRF) representation for regular 2D lattices, as well as several predefined prior and likelihood energy functionals ready to use in MRF models.
- Optimization algorithms for MRF: Metropolis–Hastings, Modified Metropolis Dynamics (MMD) and Iterated conditional modes (ICM).
- Various point-to-point, point-to-plane and point-to-mesh metrics in 3D.
- Generic representation of multidimensional vectors and points.
- Matrix operations (e.g. matrix inversion, eigenvectors and eigenvalues of real matrices).
- Transformations in 3D space based on quaternions or transformation matrices.
- Basic multidimensional geometry and topology.
- Convex hull construction for point clouds.

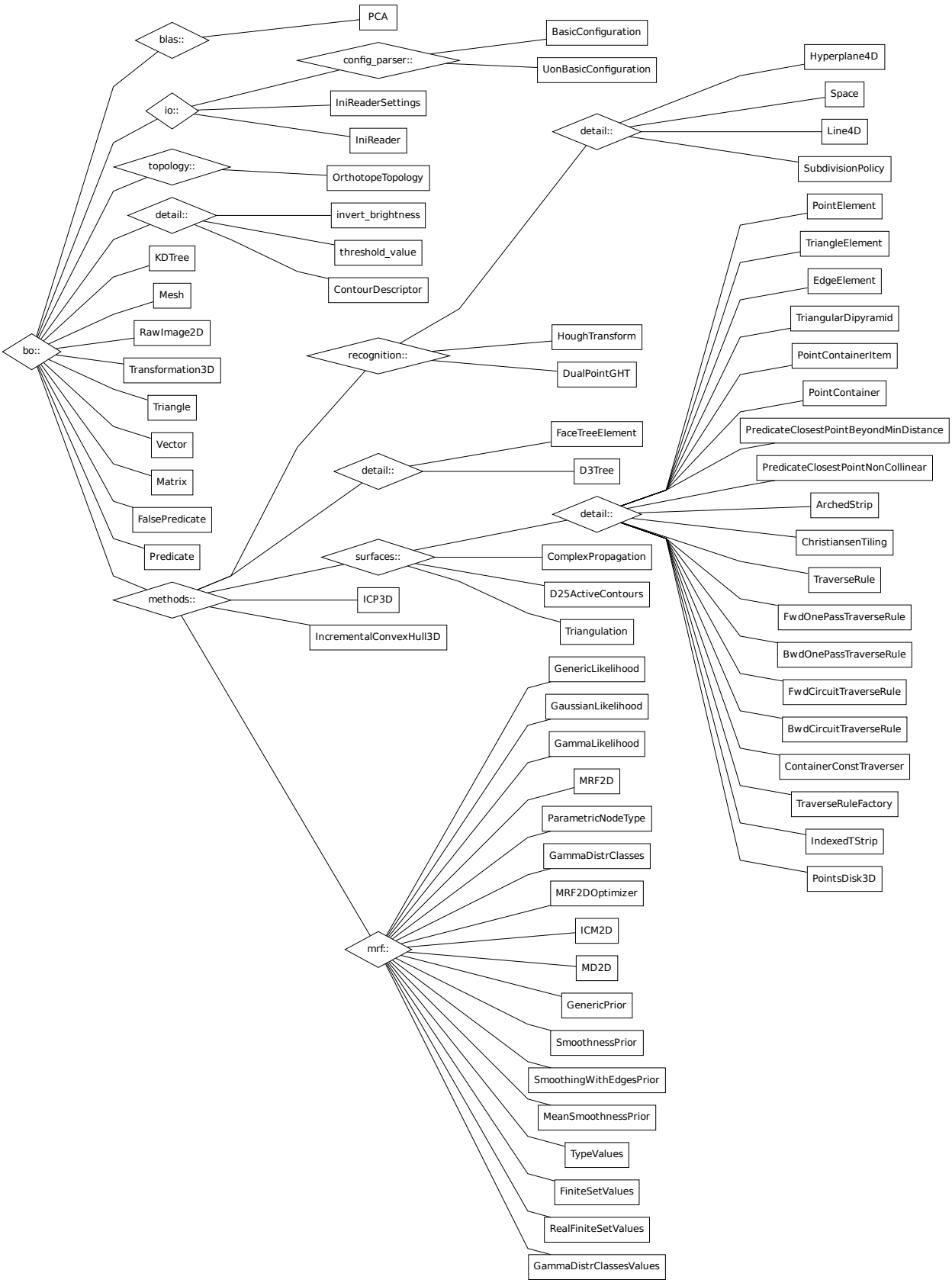


Figure C.1.: Class structure of the Bö library.

The library depends on Boost (minimal required version of Boost is 1.43, although version 1.44 is needed if the unit tests are built). Google Test is used for optional code testing. There is some header-only utility code for OpenCV library, however, it is disabled by default and may be ignored. The library also uses a third party implementation of the k-d tree structure and PLY I/O.

# Bibliography


[1] P. Abolmaesumi and M. R. Sirouspour. Ultrasound image segmentation using an interacting multiple-model probabilistic data association filter. *Medical Imaging 2004: Image Processing. Proceedings of the SPIE*, 530:484–493, 2004.

[2] M. Ackerman. Visible Human Project. `http://vhp.med.umich.edu`. U.S. National Library of Medicine, University of Michigan.

[3] A. S. Aguado, E. Montiel, and M. S. Nixon. Invariant characterization of the Hough transform for pose estimation of arbitrary shapes. In *Proceedings of the British Machine Vision Conference, BMVC2000*, pages 785–794, 2000.

[4] M. S. Aldenderfer and R. K. Blashfield. *Cluster Analysis.* Number 44 in Cluster Analysis. SAGE Publications, 1984.

[5] N. Amenta and M. Bern. Surface reconstruction by Voronoi filtering. *Discrete and Computational Geometry*, 22(4):481–504, 1999.

[6] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. *Proceedings of the sixteenth annual symposium on Computational geometry SCG 00*, pages 213–222, 2000.

[7] N. Amenta, S. Choi, and R. K. Kolluri. The power crust. *Proceedings of the sixth ACM symposium on Solid modeling and applications SMA 01*, 19(2-3):249–266, 2001.

[8] Y. Amit. *2D Object Detection and Recognition: Models, Algorithms, and Networks.* MIT Press, Cambridge, MA, USA, 2002.

[9] B. Andres, J. H. Kappes, T. Beier, U. Köthe, and F. Hamprecht. Probabilistic image segmentation with closedness constraints. In *Proceedings of ICCV*, pages 2611–2618, 2011.

[10] D. Angelova and L. Mihaylova. Contour segmentation in 2D ultrasound medical images with particle filtering. *Machine Vision and Applications*, 22(3):551–561, 2011.

[11] N. Archip, R. Rohling, P. Cooperberg, H. Tahmasebpour, and S. Warfield. Spectral clustering algorithms for ultrasound image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005*, volume 3750 of *Lecture Notes in Computer Science*, pages 862–869. Springer Berlin Heidelberg, 2005.

[12] J. A. R. Artolazabal, J. Illingworth, and A. S. Aguado. LIGHT: Local Invariant Generalized Hough Transform. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 304 –307, 2006.

[13] F. Aurenhammer. Voronoi diagrams — a survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, 1991.

[14] A. Bakulina, D. Hlindzich, and R. Maenner. Fast dual-point Hough transform for medical object recognition. In *2010 IEEE 23rd International Symposium on ComputerBased Medical Systems CBMS*, pages 297–302. IEEE, 2010.

[15] D. H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[16] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using Hough transforms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2233 –2240, 2010.

[17] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346 – 359, 2008.

[18] B. Büeler, A. Enge, and K. Fukuda. Exact volume computation for polytopes: A practical study. In *Polytopes — Combinatorics and Computation*, volume 29, pages 131–154. Birkhäuser Basel, 2000.

[19] S. Berg, H. Torp, D. Martens, E. Steen, S. Samstad, I. Hoivik, and B. Olstad. Dynamic three-dimensional freehand echocardiography using raw digital ultrasound data - transducer design and beam steering. *Ultrasound in Medicine and Biology*, 25(5):745–753, 1999.

[20] M. Bergtholdt, J. H. Kappes, S. Schmidt, and C. Schnörr. A study of parts-based object class detection using complete graphs. *International Journal of Computer Vision*, 87(1-2):93–117, 2010.

[21] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, 1999.

[22] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, B-48:259–302, 1986.

[23] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[24] T. G. Bishop and H. Fuchs. The self-tracker: A smart optical sensor on silicon. *Proceedings of Conference on Advanced Research in VLSI (MIT Press)*, 1984.

[25] F. Bludau, D. Hlindzich, P. Krasopevtsev, A. Kryvanos, U. Obertacke, and M. Schwarz. Ein neuartiger Ansatz zur 3D-Oberflächendarstellung von Knochen mittels Ultraschall. In *Meeting Abstract: Deutscher Kongress für Orthopädie und Unfallchirurgie, DKOU'11*, Berlin, Germany, 25-28 October 2011.

[26] N. Bom, A. F. W. Van der Steen, C. T. Lancee, N. De Jong, and J. R. T. C. Roelandt. Survey of developments in cardiac and cardiovascular ultrasound imaging. *Klinische Fysica*, 3:19–23, 2000.

[27] D. Breitenreicher, J. Lellmann, and C. Schnörr. Sparse template-based variational image segmentation. *Advances in Adaptive Data Analysis*, 3:149–166, 2011.

[28] J. F. Brinkley, W. E. Moritz, and D. W. Baker. Ultrasonic three-dimensional imaging and volume from a series of arbitrary sector scans. *Ultrasound in Medicine and Biology*, 4(4):317 – 321, 1978.

[29] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679 – 698, 1986.

[30] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10(6):350–355, 1978.

[31] T. F. Chan and L. A. Vese. Active contours without edges. *Image Processing, IEEE Transactions on*, 10(2):266–277, 2001.

[32] C.-P. Chau and W.-C. Siu. Generalized dual-point Hough transform for object recognition. In *Image Processing. Proceedings of International Conference on*, volume 1, pages 560–564, 1999.

[33] C.-P. Chau and W.-C. Siu. Adaptive dual-point Hough transform for object recognition. *Computer Vision and Image Understanding*, 96(1):1 – 16, 2004.

[34] T. Chen and D. N. Metaxas. Image segmentation based on the integration of Markov random fields and deformable models. In *Proceedings of the 3rd International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 256 – 265, 2000.

[35] Y.-L. Chen and S.-H. Lai. An orientation inference framework for surface reconstruction from unorganized point clouds. *IEEE Transactions on Image Processing*, 20(3):762–775, 2011.

[36] F.-H. F. Cheng, F.-T. Fan, S.-H. Lai, C.-L. Huang, J.-X. Wang, and J.-H. Yong. Loop subdivision surface based progressive interpolation. *Journal of Computer Science and Technology*, 24(1):39–46, 2009.

[37] S.-H. Chiu, C.-Y. Wen, Lee J.-H., Lin K.-H., and Chen H.-M. A fast randomized generalized Hough transform for arbitrary shape detection. *International Journal of Innovative Computing, Information and Control*, 8(2):1103 – 1116, 2012.

[38] S. M. Choi, J. E. Lee, J. Kim, and Kim M. H. Volumetric object reconstruction using the 3D MRF model-based segmentation. *IEEE Transactions on Medical Imaging*, 16(6):887 – 892, 1997.

[39] H. N. Christiansen and T. W. Sederberg. Conversion of complex contour line definitions into polygonal element mosaics. In *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '78, pages 187–192. ACM, 1978.

[40] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. In *Information Processing in Medical Imaging*, volume 687 of *Lecture Notes in Computer Science*, pages 33–47. Springer Berlin Heidelberg, 1993.

[41] 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.

[42] J. Cornelis, J. De Becker, M. Bister, C. Vanhove, G. Demonceau, and A. Cornelis. Techniques for cardiac image segmentation. In *Engineering in Medicine and Biology Society, 14th Annual International Conference of the IEEE*, volume 5, pages 1906–1908, 1992.

[43] D. Cremers, T. Kohlberger, and C. Schnörr. Shape statistics in kernel space for variational image segmentation. *Pattern Recognition*, 36(9):1929–1943, 2003.

[44] D. Cremers, N. Sochen, and C. Schnörr. Multiphase dynamic labeling for variational recognition-driven image segmentation. *International Journal of Computer Vision*, 66(1):67–81, 2006.

[45] D. Cremers, F. Tischhäuser, J. Weickert, and C. Schnörr. Diffusion snakes: Introducing statistical shape knowledge into the mumford–shah functional. *International Journal of Computer Vision*, 50(3):295–313, 2002.

[46] Y. Dai, T. Jie, D. Dong, G. Yan, and H. Zheng. Real-time visualized freehand 3D ultrasound reconstruction based on GPU. *Information Technology in Biomedicine, IEEE Transactions on*, 14(6):1338–1345, 2010.

[47] R. Damadian. Tumor detection by nuclear magnetic resonance. *Science*, 171(3976):1151–1153, 1971.

[48] L. Danisch, K. Englehart, and A. Trivett. Spatially continuous six degree of freedom position and orientation sensor. *Sensor Review*, 19(2):106–112, 1999.

[49] S. R. Deans. Hough transform from the Radon transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(2):185–188, 1981.

[50] B. N. Delaunay. Sur la sphère vide. *Bulletin of Academy of Sciences of the USSR*, (6):793–800, 1934.

[51] C. Deng and X. Yang. Interpolation over arbitrary topology meshes using Doo-Sabin surfaces. *2009 IEEE International Conference on Shape Modeling and Applications*, pages 52–57, 2009.

[52] C. Deng and X. Yang. A simple method for interpolating meshes of arbitrary topology by Catmull–Clark surfaces. *The Visual Computer*, 26(2):137–146, 2009.

[53] L. Di Angelo and L. Giaccari. A fast algorithm for manifold reconstruction of surfaces. *Proceedings of the International Conference on Innovative Methods in Product Design*, pages 177–186, 2011.

[54] K. K. Djoa, N. de Jong, F. C. van Egmond, J. D. Kasprzak, W. B. Vletter, C. T. Lancee, A. F. W. van der Steen, N. Bom, and J. R. T. C. Roelandt. A fast rotating scanning unit for real-time three-dimensional echo data acquisition. *Ultrasound in Medicine and Biology*, 26:863–869, 2000.

[55] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10(6):356–360, 1978.

[56] Y. Duan, H. Qin, and S. Brook. 2.5D active contour for surface reconstruction. *Proceedings of the 8th International Workshop on Vision Modeling and Visualization VMV 2003*, page 431–439, 2003.

[57] A. Dubouski, A. Kryvanos, and A. Belevich. Automating freehand ultrasound calibration using a plane phanthom. *Transactions on Pattern Recognition and Information Processing*, 9(1):81–86, 2007.

[58] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.

[59] K. T. Dussik. Über die Möglichkeit, hochfrequente mechanische Schwingungen als diagnostisches Hilfsmittel zu verwerten. *Zeitschrift für die gesamte Neurologie und Psychiatrie*, 174(1):153–168, 1942.

[60] 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, 1990.

[61] O. Ecabert. Adaptive Hough transform for the detection of natural shapes under weak affine transformations. *Pattern Recognition Letters*, 25(12):1411–1419, 2004.

[62] C. Ericson. *Real-time collision detection*, volume 1. Morgan Kaufmann, 2005.

[63] A. Fenster, D. B. Downey, and H. N. Cardinal. Three-dimensional ultrasound imaging. *Physics in Medicine and Biology*, 46:R67–R69, 2001.

[64] A. Fisher, R. B. Fisher, C. Robertson, and N. Werghi. Finding surface correspondence for object recognition and registration using pairwise geometric histograms. In *Proceedings of ECCV*, pages 674–686, 1998.

[65] K. Fundana, A. Heyden, C. Gosch, and C. Schnörr. Continuous graph cuts for prior-based object segmentation. In *19th International Conference on Pattern Recognition (ICPR)*, pages 1–4, 2008.

[66] P.-F. Fung, W.-S. Lee, and I. King. Randomized generalized Hough transform for 2-D gray scale object detection. In *Pattern Recognition, Proceedings of the 13th International Conference on*, volume 2, pages 511 – 515, 1996.

[67] S. Galic, S. Loncaric, and E. N. Tesla. Cardiac image segmentation using spatio-temporal clustering. In *Proc. SPIE*, pages 1199–1206, 2001.

[68] J. Gall and V. Lempitsky. Class-specific Hough forests for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*, pages 1022 –1029, 2009.

[69] X. Gao, Y. Su, X. Li, and D. Tao. A review of Active Appearance Models. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 40(2):145–158, 2010.

[70] C. F. Gauss. *Theoria attractionis corporum sphaeroidicorum ellipticorum homogeneorum: methodo nova tractata*, volume 2, pages 355–378. Commentationes societatis regiae scientiarium Gottingensis recentiores (Reprint: Hildesheim 1981), 1813.

[71] F. Gebhard, O. Krivonos, U. C. Liener, P. Keppler, L. Kinzl, and J. Hesser. Bone extraction from ultrasound images and surface matching with CT. In *Proceedings of CAOS: Computer Assisted Orthopedic Surgery*, 2000.

[72] A. H. Gee, R. J. Housden, P. Hassenpflug, G. M. Treece, and R. W. Prager. Sensorless freehand 3D ultrasound using regression of the echo intensity. In *Ultrasound in Medicine and Biology*, volume 29, pages 437–446, 2003.

[73] A. H. Gee, R. J. Housden, P. Hassenpflug, G. M. Treece, and R. W. Prager. Sensorless freehand 3D ultrasound in real tissue: Speckle decorrelation without fully developed speckle, 2006.

[74] E. A. Geiser, L. G. Christie, D. A. Conetta, C. R. Conti, and G. S. Gossrnan. A mechanical arm for spatial registration of two-dimensional echocardiographic sections. *Catheterization and Cardiovascular Diagnosis*, 8:89–101, 1982.

[75] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(6):721–741, 1984.

[76] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *Robotics and Automation, IEEE Journal of*, 4(2):193–203, 1988.

[77] A. M. Goldsmith, P. C. Pedersen, and T. L. Szabo. An inertial-optical tracking system for portable, quantitative, 3D ultrasound. In *Ultrasonics Symposium, 2008. IUS 2008. IEEE*, pages 45–49, 2008.

[78] R. C. Gonzalez and R. E. Woods. *Digital Image Processing (3rd Edition)*. Prentice Hall, 3 edition, 2007.

[79] P. Grangeat. Mathematical framework of cone beam 3D reconstruction via the first derivative of the Radon transform. In *Mathematical Methods in Tomography*, volume 1497 of *Lecture Notes in Mathematics*, pages 66–97. Springer Berlin Heidelberg, 1991.

[80] W. E. L. Grimson. *Object recognition by computer: the role of geometric constraints*. MIT Press, Cambridge, MA, USA, 1990.

[81] A. Gueziec. 'Meshsweeper': Dynamic Point-to-Polygonal-Mesh Distance and Applications. *IEEE Transactions on Visualization and Computer Graphics*, 7(1):47–61, 2001.

[82] E. J. Gumbel. *Statistics of Extremes*. Courier Dover Publications (2004), 1958.

[83] I. Hacihaliloglu, R. Abugharbieh, A. J. Hodgson, and R. N. Rohling. Enhancement of bone surface visualization from 3D ultrasound based on local phase information. In *Ultrasonics Symposium, 2006. IEEE*, pages 21–24, 2006.

[84] G. A. Hansen, R. W. Douglass, and A. Zardecki. *Mesh enhancement*, page 404. Imperial College Press, 2005.

[85] X. Hao, C. Bruce, C. Pislaru, and J. F. Greenleaf. A novel region growing method for segmenting ultrasound images. In *Ultrasonics Symposium, 2000 IEEE*, volume 2, pages 1717–1720, 2000.

[86] J. A. Hartigan and M. A. Wong. Algorithm AS 136: A k-means clustering algorithm. *Applied Statistics*, 28(1):100–108, 1979.

[87] M. Heiler and C. Schnörr. Natural image statistics for natural image segmentation. *International Journal of Computer Vision*, 63(1):5–19, 2005.

[88] W. R. Hendee and E. R. Ritenour. *Medical Imaging Physics*. Wiley, 2003.

[89] D. Hlindzich, P. Krasnopevtsev, T. Poerner, and A. Kryvanos. A flexible medical image archiving and reporting system. *9th International Conference On Pattern Recognition and Information Processing*, pages 298–302, 2009.

[90] D. Hlindzich and A. Kryvanos. A three-dimensional segmentation approach for automatic human heart modelling. In *Proceedings of the 2nd International Conference on Advanced Information and Telemedicine Technologies for Health*, pages 49 – 53, 2008.

[91] D. Hlindzich and A. Kryvanos. Mesh-growing surface reconstruction from sparse data based on the complex propagation scheme. *Proc. SPIE 8334, Fourth International Conference on Digital Image Processing (ICDIP 2012)*, pages 83340N–83340N–6, 2012.

[92] D. Hlindzich, A. Kryvanos, C. Doesch, and R. Maenner. A 3D/4D freehand ultrasound system for cardiac modelling and analysis. *IEEE International Symposium on Computer-Based Medical Systems*, pages 346–351, 2010.

[93] D. Hlindzich and R. Maenner. Medical feature matching and model extraction from MRI/CT based on the Invariant Generalized Hough/Radon Transform. In *4th European Conference of the International Federation for Medical and Biological Engineering*, pages 608 –612, 2009.

[94] A. C. Hodge, A. Fenster, D. B. Downey, and H. M. Ladak. Prostate boundary segmentation from ultrasound images using 2D active shape models: Optimisation and extension to 3D. *Computer Methods and Programs in Biomedicine*, 84(2-3):99–113, 2006.

[95] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *ACM SIGGRAPH Computer Graphics*, 26(2):71–78, 1992.

[96] P. V. C. Hough. Method and means for recognizing complex patterns, 1962. U. S. Patent 3069654.

[97] R. J. Housden, G. M. Treece, A. H. Gee, and R. W. Prager. Hybrid systems for reconstruction of freehand 3D ultrasound data. Technical report, University of Cambridge Department of Engineering, 2007.

[98] D. H. Howry, G. Posakony, C. R. Cushman, and J. H Holmes. Three-dimensional and stereoscopic observation of body structures by ultrasound. *Journal of Applied Physiology*, 9(2):304–306, 1956.

[99] P. W. Hsu, R. W. Prager, A. H. Gee, and G. M. Treece. *Freehand 3D Ultrasound Calibration: A Review*, chapter 3, pages 47–84. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[100] J. Hung, R. Lang, F. Flachskampf, S. K. Shernan, M. L. McCulloch, D. B. Adams, J. Thomas, M. Vannan, and T. Ryan. 3d echocardiography: A review of the current status and future directions. *Journal of the American Society of Echocardiography*, 20(3):213–233, 2007.

[101] J. Illingworth and J. Kittler. The adaptive Hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):690–698, 1987.

[102] J. Illingworth and J. Kittler. A survey of efficient Hough transform methods. In *Proceedings of the Alvey Vision Conference*, pages 43.1–43.8, 1987.

[103] A. K. Jain and R. H. Taylor. Understanding bone responses in B-mode ultrasound images and automatic bone surface extraction using a Bayesian probabilistic framework. *Proc. SPIE 5373, Medical Imaging 2004: Ultrasonic Imaging and Signal Processing*, pages 131–142, 2004.

[104] B. Johnston, M. S. Atkinst, and Booth K. S. Partial volume segmentation in 3D of lesions and tissues in magnetic resonance images. In *Medical Imaging 1994: Image Processing, Proceedings of SPIE 2167*, pages 28–39. SPIE, 1994.

[105] M.-P. Jolly, N. Duta, and G. Funka-Lea. Segmentation of the left ventricle in cardiac MR images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 501–508, 2001.

[106] M. Kallay. The complexity of incremental convex hull algorithms in Rd. *Information Processing Letters*, 19(4):197, 1984.

[107] A. Karamalis, W. Wein, O. Kutter, and N. Navab. Fast hybrid freehand ultrasound volume reconstruction. *Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling. Proceedings of the SPIE*, 7261:726114–726114–8, 2009.

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

[109] Z. Kato, J. Zerubia, and M. Berthod. Satellite image classification using a modified Metropolis dynamics. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 3, pages 573–576, 1992.

[110] M. R. Kaus, J. Berg, W. Niessen, and V. Pekar. Automated segmentation of the left ventricle in cardiac MRI. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003*, volume 2878 of *Lecture Notes in Computer Science*, pages 432–439. Springer Berlin Heidelberg, 2003.

[111] M. Kazhdan, M. Bolitho, and H. Hoppe. *Poisson surface reconstruction*, pages 61–70. Eurographics Association, 2006.

[112] M. M. Khalighi, H. Soltanian-Zadeh, and C. Lucas. Unsupervised MRI segmentation with spatial connectivity. *Proc. SPIE*, 4684:1742–1750, 2002.

[113] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[114] L. Kobbelt. Interpolatory subdivision on open quadrilateral nets with arbitrary topology. *Computer Graphics Forum*, 15(3):409–420, 1996.

[115] O. Krivonos, P. Keppler, F. Gebhard, L. Kinzl, J. Hesser, and R. Manner. From planning of complex bone deformities correction to computer aided operation. In *14th IEEE Symposium on Computer-Based Medical Systems*, pages 210–215, 2001.

[116] H. M. Ladak, D. B. Downey, D. A. Steinman, and A. Fenster. Semi-automatic technique for segmentation of the prostate from 2D ultrasound images. In *BMES/EMBS Conference, 1999. Proceedings of the First Joint*, volume 2, page 1144, 1999.

[117] C. H. Lampert. An efficient divide-and-conquer cascade for nonlinear object detection. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1022–1029, 2010.

[118] A. H. Land and A. G. Doig. An automatic method of solving discrete programming problems. *Econometrica*, 28(3):pp. 497–520, 1960.

[119] G. Langs, P. Peloschek, R. Donner, M. Reiter, and H. Bischof. Active feature models. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 417–420, 2006.

[120] C. Lara, J. J. Flores, and F. Calderon. On the hyperbox – hyperplane intersection problem. *Journal of Computer Science*, 8(4):21–27, 2009.

[121] P. Lauterbur. Image formation by induced local interactions: Examples employing nuclear magnetic resonance, 1973.

[122] V. Leavers. The dynamic generalized Hough transform. In *Computer Vision — ECCV 90*, volume 427 of *Lecture Notes in Computer Science*, pages 592–594. Springer Berlin / Heidelberg, 1990.

[123] A. Lehmann, B. Leibe, and L. Gool. Fast PRISM: Branch and Bound Hough Transform for Object Class Detection. *International Journal of Computer Vision*, 94(2):175–197, 2011.

[124] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.

[125] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal on Computer Vision*, 77(1-3):259–289, 2008.

[126] J. Lellmann, F. Becker, and C. Schnörr. Convex optimization for multi-class image labeling with a novel family of total variation based regularizers. In *IEEE International Conference on Computer Vision (ICCV)*, pages 646 – 653, 2009.

[127] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *The Quarterly of Applied Mathematics*, 2(2):164–168, 1944.

[128] D. Levin. The approximation power of moving least-squares. *Mathematics of Computation*, 67(224):1517–1532, 1998.

[129] D. Levin. Mesh-independent surface interpolation. *Geometric Modeling for Scientific Visualization*, 3:37–49, 2003.

[130] H. Li, M. A. Lavin, and R. J. Le Master. Fast Hough transform: A hierarchical approach. *Computer Vision, Graphics, and Image Processing*, 36(2–3):139 – 161, 1986.

[131] H. Li, M. A. Lavin, and R. J. Le Master. Fast Hough transform based on bintree data structure. *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR86*, pages 640 – 642, 1986.

[132] L. Li, Y. Fu, P. Bai, and W. Mao. Medical ultrasound image segmentation based on improved watershed scheme. In *Bioinformatics and Biomedical Engineering , 2009. ICBBE 2009. 3rd International Conference on*, pages 1–4, 2009.

[133] S. Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag, London, UK, UK, 1995.

[134] U. C. Liener, O. Krivonos, R. Manner, J. Hesser, L. Kinzl, and F. Gebhard. Ultrasound - a new tool for surface matching in computer navigated surgery. *Journal of Military Medicine*, pages 151–154, 2001.

[135] H. Lin. A mesh reconstruction algorithm driven by an intrinsic property of a point cloud. *Computer-Aided Design*, 36(1):1–9, 2004.

[136] O. Linde and T. Lindeberg. Composed complex-cue histograms: An investigation of the information content in receptive field based image descriptors for object recognition. *Computer Vision and Image Understanding*, 116(4):538 – 560, 2012.

[137] C. Loop. Smooth subdivision surfaces based on triangles. Department of mathematics, University of Utah, 1987.

[138] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157, 1999.

[139] Z. Majcenic and S. Loncaric. Algorithm for spatio-temporal heart segmentation. *Proc. SPIE 3979, Medical Imaging 2000: Image Processing*, pages 936–943, 2000.

[140] T. McInerney and D. Terzopoulos. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4d image analysis. *Computerized Medical Imaging and Graphics*, 19:69–83, 1995.

[141] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: A survey. *Medical Image Analysis*, 1:91–108, 1996.

[142] R. Mencl and H. Mueller. Interpolation and approximation of surfaces from three-dimensional scattered data points. In *Scientific Visualization*, pages 223–232, 1997.

[143] P. H. Mills and H. Fuchs. 3D ultrasound display using optical tracking. In *Visualization in Biomedical Computing, Proceedings of the First Conference on*, pages 490–497, 1990.

[144] E. Montiel, A. S. Aguado, and M. S. Nixon. Improving the Hough transform gathering process for affine transformations. *Pattern Recognition Letters*, 22(9):959–969, 2001.

[145] A. Montillo, D. Metaxas, and L. Axel. Automated model-based segmentation of the left and right ventricles in tagged cardiac MRI. *MICCAI*, 1, 2003.

[146] P. Moreels and P. Perona. A probabilistic cascade of detectors for individual object recognition. In *Computer Vision – ECCV 2008*, volume 5304 of *Lecture Notes in Computer Science*, pages 426–439. Springer Berlin Heidelberg, 2008.

[147] M. Mulet-Parada and J. A. Noble. 2D+T acoustic boundary detection in echocardiography. In *Medical Image Computing and Computer-Assisted Interventation — MICCAI'98*, volume 1496 of *Lecture Notes in Computer Science*, pages 806–813. Springer Berlin Heidelberg, 1998.

[148] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[149] R. M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, University of Toronto, 1993.

[150] C. F. Olson. A general method for geometric feature matching and model extraction. *International Journal of Computer Vision*, 45:39–54, 2001.

[151] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572, 1901.

[152] J. Peters, O. Ecabert, C. Meyer, H. Schramm, R. Kneser, A. Groth, and J. Weese. Automatic Whole Heart Segmentation in Static Magnetic Resonance Image Volumes. In *Medical Image Computing and Computer-Assisted Intervention MICCAI'07*, pages 402–410, 2007.

[153] M. Petrou, V. A. Kovalev, and J. R. Reichenbach. Three-dimensional nonlinear invisible boundary detection. *IEEE Transactions on Image Processing*, 15(10):3020 – 3032, 2006.

[154] C. Pluempitiwiriyawej, J. M. F. Moura, Y.-J. L. Wu, S. Kanno, and Chien H. Stochastic active contour for cardiac MR image segmentation. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3, pages II–1097–100, 2003.

[155] R. Pohle and K. D. Toennies. A new approach for model-based adaptive region growing in medical image analysis. *9th International Conference on Computer Analysis and Patterns*, pages 238–246, 2001.

[156] J. Princen, J. Illingworth, and J. Kittler. A hierarchical approach to line extraction. In *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR89*, pages 92 – 97, 1989.

[157] J. Princen, J. Illingworth, and J. Kittler. A formal definition of the Hough transform: Properties and relationships. *Journal of Mathematical Imaging and Vision*, 1:153–168, 1992.

[158] T. Qiu, T. Wen, W. Qin, J. Gu, and L. Wang. Freehand 3D ultrasound reconstruction for image-guided surgery. In *Bioelectronics and Bioinformatics (ISBB), 2011 International Symposium on*, pages 147–150, 2011.

[159] F. H. Raab, E. B. Blood, T. O. Steiner, and H. R. Jones. Magnetic position and orientation tracking system. *IEEE Transactions on Aerospace and Electronic Systems*, 15(5):709–718, 1979.

[160] I. I. Rabi, J. R. Zacharias, S. Millman, and P. Kusch. A new method of measuring nuclear magnetic moment. *Physical Review Letters*, 53:318–318, Feb 1938.

[161] J. Radon. Ueber die Bestimmung von Funktionen durch ihre Integralwerte laengs gewisser Mannigfaltigkeiten. *Berichte Saechsische Akademie der Wissenschaften*, 69:262–277, 1917.

[162] B. R. Radu. Point Cloud Library (PCL). `http://pointclouds.org/`.

[163] K. Rajpoot, V. Grau, and J. A. Noble. Local-phase based 3D boundary detection using monogenic signal and its application to real-time 3D echocardiography images. In *Biomedical Imaging: From Nano to Macro, 2009. ISBI '09. IEEE International Symposium on*, pages 783–786, 2009.

[164] F. Rathke, S. Schmidt, and C. Schnörr. Order preserving and shape prior constrained intra-retinal layer segmentation in optical coherence tomography. In *MICCAI*. Springer, 2011.

[165] M. Riesenhuber and T. Poggio. Computational models of object recognition in cortex: A review. Technical report, Artificial Intelligence Laboratory and Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, 2000.

[166] R. N. Rohling. *3D freehand ultrasound: reconstruction and spatial compounding.* dissertation, University of Cambridge, Cambridge, UK, 1999.

[167] D. Rueckert and P. Burger. Geometrically deformable templates for shape-based segmentation and tracking in cardiac mr images. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 1223 of *Lecture Notes in Computer Science*, pages 83–98. Springer-Verlag, 1997.

[168] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, May 9-13 2011.

[169] H. Samet. *Applications of spatial data structures: Computer graphics, image processing, and GIS.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1990.

[170] O. Schall and M. Samozino. Surface from scattered points: A brief survey of recent developments. *Proceedings of the 1st International Workshop on Semantic Virtual Environments*, pages 138–147, 2005.

[171] S. Schmidt, J. H. Kappes, M. Bergtholdt, V. Pekar, S. Dries, D. Bystrov, and C. Schnörr. Spine detection and labeling using a parts-based graphical model. In *Proceedings of 20th International Conference on Information Processing in Medical Imaging (IPMI 2007)*, volume 4584 of *LCNS*, pages 122–133. Springer, 2007.

[172] B. Schmitzer and C. Schnörr. Modelling Convex Shape Priors and Matching based on the Gromov-Wasserstein Distance. *Journal of Mathematical Imaging and Vision*, 46(1):143–159, 2013.

[173] C. Schnörr. A study of a convex variational diffusion approach for image segmentation and feature extraction. *Journal of Mathematical Imaging and Vision*, 8(3):271–292, 1998.

[174] C. Schnörr. Variational methods for adaptive image smoothing and segmentation. In B. Jähne, H. Haußecker, and P. Geißler, editors, *Handbook on Computer Vision and Applications: Signal Processing and Pattern Recognition*, volume 2, pages 451–484, San Diego, 1999. Academic Press.

[175] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):411–426, 2007.

[176] L. Shapiro and G. Stockman. *Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2001.

[177] Y. A. Sheikh, E. A. Khan, and T. Kanade. Mode-seeking by medoidshifts. In *IEEE 11th International Conference on Computer Vision (ICCV)*, pages 1 – 8, 2007.

[178] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 1997.

[179] J. Shiraishi, S. Katsuragawa, J. Ikezoe, T. Matsumoto, T. Kobayashi, K. Komatsu, M. Matsui, H. Fujita, Y. Kodera, and Doi K. Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists' detection of pulmonary nodules. *American Journal of Roentgenology*, 174:71 – 74, 2000.

[180] K. R. Sloan and J. Painter. From contours to surfaces: testbed and initial results. *SIGCHI Bull.*, 18(4):115–120, 1986.

[181] O. V. Solberg, F. Lindseth, H. Torp, R. E. Blake, and T. A. N. Hernes. Freehand 3D ultrasound reconstruction algorithms — a review. *Ultrasound in Medicine and Biology*, 33(7):991 – 1009, 2007.

[182] K. R. Stickels and L. S. Wann. An analysis of three-dimensional reconstructive echocardiography. *Ultrasound in Medicine and Biology*, 10(5):575 – 580, 1984.

[183] G. Stockman. Object recognition and localization via pose clustering. *Computer Vision, Graphics, and Image Processing*, 40(3):361–387, 1987.

[184] N. Suetake, E. Uchino, and K. Hirata. Generalized fuzzy Hough transform for detecting arbitrary shapes in a vague and noisy image. *Soft Computing*, 10(12):1161–1168, 2006.

[185] P. Suetens. *Fundamentals of Medical Imaging*. Cambridge medicine. Cambridge University Press, 2009.

[186] P. J. Swaney, J. Burgner, T. S. Pheiffer, D. C. Rucker, H. B. Gilbert, J. E. Ondrake, A. L. Simpson, E. C. Burdette, M. I. Miga, and R. J. Webster III. Tracked 3D ultrasound targeting with an active cannula. *Proc. SPIE 8316, Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling*, pages 83160R–83160R–9, 2012.

[187] I. Tobor, P. Reuter, and C. Schlick. Efficient reconstruction of large scattered geometric datasets using the partition of unity and radial basis functions. *Winter School of Gomputer Graphics 2004*, 12(1):467–474, 2004.

[188] F. Tombari and L. Di Stefano. Object recognition in 3D scenes with occlusions and clutter by Hough voting. In *Fourth Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, pages 349 –355, 2010.

[189] G. M. Treece. *Volume measurement and surface visualisation in sequential freehand 3D ultrasound*. PhD thesis, University of Cambridge, 2000.

[190] G. M. Treece, A. H. Gee, R. W. Prager, C. J. C. Cash, and L. H. Berman. High-definition freehand 3-D ultrasound. *Ultrasound in Medicine and Biology*, 29(4):529–546, 2003.

[191] G. M. Treece, J. E. Lindop, A. H. Gee, and R. W. Prager. Freehand ultrasound elastography with a 3D probe. *Ultrasound in Medicine and Biology*, 2007.

[192] Z. Tu, S.-C. Zhu, and H.-Y. Shum. Image segmentation by data driven Markov chain Monte Carlo. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 131–138, 2001.

[193] H. C. van Assen, R. J. van der Geest, M. G. Danilouchkine, H. J. Lamb, J. H. C. Reiber, and B. P. F. Lelieveldt. Three-dimensional active shape model matching for left ventricle segmentation in cardiac CT. *Proc. SPIE 5032, Medical Imaging 2003: Image Processing*, pages 384–393, 2003.

[194] M. van Ginkel, C. L. Hendriks, and L. J. van Vliet. A short introduction to the Radon and Hough transforms and how they relate to each other, 2004.

[195] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In *Computer Vision – ECCV 2008*, volume 5305 of *Lecture Notes in Computer Science*, pages 705–718. Springer Berlin Heidelberg, 2008.

[196] O. von Ramm and S. Smith. Three-dimensional imaging system, Patent: US 4694434, 1987.

[197] O. T. von Ramm and S. W. Smith. Real time volumetric ultrasound imaging system. *Digital Imaging*, 3:261–266, 1990.

[198] T. Werner. A linear programming approach to max-sum problem: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(7):1165–1179, 2007.

[199] O. Woodford, M.-T. Pham, A. Maki, F. Perbet, and B. Stenger. Demisting the Hough transform for 3D shape recognition and registration. In *Proceedings of the British Machine Vision Conference*, pages 32.1 – 32.11. BMVA Press, 2011.

[200] L. Xu and E. Oja. Randomized Hough transform: Basic mechanisms, algorithms, and computational complexities. *CVGIP: Image Understanding*, 57(2):131–154, 1993.

[201] S. Yagel, S. M. Cohen, D. V. Valsky, and B. Messing. 3D and 4D ultrasound in fetal cardiac screening and the evaluation of congenital heart defects. *Medscape CME*, 2009.

[202] P. Yan and A. A. Kassim. Medical image segmentation using minimal path deformable models with implicit shape priors. *Information Technology in Biomedicine, IEEE Transactions on*, 10(4):677–684, 2006.

[203] X. Yang. Surface interpolation of meshes by geometric subdivision. *Computer-Aided Design*, 37(5):497–508, 2005.

[204] R. Yip, P. Tam, and D. Leung. Modification of Hough transform for object recognition using a 2-dimensional array. *Pattern Recognition*, 28(11):1733 – 1744, 1995.

[205] C. Yizong. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790 – 799, 1995.

[206] Z. B. Zabinsky, W. Wang, Y. Prasetio, A. Ghate, and J. W. Yen. Adaptive probabilistic branch and bound for level set approximation. In S. Jain, Roy R. Creasey Jr., Jan Himmelspach, K. Preston White, and Michael C. Fu, editors, *Winter Simulation Conference*, pages 4151–4162. WSC, 2011.

[207] J. Zerubia and R. Chellappa. Mean field annealing using compound Gauss-Markov random fields for edge detection and image estimation. *Neural Networks, IEEE Transactions on*, 4(4):703–709, 1993.

[208] L. Zhukov, Z. Bao, I. Guskov, J. Wood, and D. Breen. Dynamic deformable models for 3D MRI heart segmentation. *Proc. SPIE Vol. 4684, Medical Imaging 2002: Image Processing*, pages 1398–1405, 2002.

[209] D. Zorin, P. Schröder, and W. Sweldens. Interpolating subdivision for meshes with arbitrary topology. *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques SIGGRAPH 96*, 30:189–192, 1996.