Surgical tools localization in 3D ultrasound images

Marian Uhercik

Surgical Tools Localization in 3D Ultrasound Images

Doctoral Thesis under Co-supervision

presented on the 20th April 2011 in Prague to

the Czech Technical University in Prague in partial fulfillment of the requirements for the Ph.D. degree in Study Programme No. P 2612—Electrical Engineering and Information Technology, branch No. 3902V035—Artificial Intelligence and Biocybernetics

and to

the Institut National des Sciences Appliquées de Lyon in partial fulfillment of the requirements for the degree of Docteur, École Doctorale MEGA (Mécanique, Energétique, Génie Civil, Acoustique)

serial number : 2011-ISAL-0035

by

Mgr. Marián UHERČÍK

Panel:		
Co-Advisor	Jan KYBIC	Research Fellow, CTU, Prague
Co-Advisor	Hervé LIEBGOTT	Associate Professor, UCB, Lyon
Co-Advisor	Christian CACHARD	Professor, UCB, Lyon
Reviewer	Jiří JAN	Professor, UT, Brno
Reviewer	Michal KOZUBEK	Associate Professor, MU, Brno
Reviewer	Christophe DOIGNON	Associate professor, ENSP, Strassbourg
	Václav HLAVÁČ	Professor, CTU, Prague
	Marcel JIŘINA	Doctor of Sciences, AS, Prague
	Didier VRAY	Professor, INSA, Lyon
	Luděk MÜLLER	Associate Professor, ZCU, Plzeň

Research Reports of CMP, Czech Technical University in Prague, **No. 17, 2010**
ISSN 1213–2365

Available at:
`ftp://cmp.felk.cvut.cz/pub/cmp/articles/uhercik/Uhercik-TR-2010-17.pdf`

Published by Center of Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University in Prague
Technická 2, 166 27 Prague 6, Czech Republic
fax: +420224357385, phone: +420224357637
`http://cmp.felk.cvut.cz`

Abstract

This thesis deals with automatic localization of thin surgical tools such as needles or electrodes in 3D ultrasound images. The precise and reliable localization is important for medical interventions such as needle biopsy or electrode insertion into tissue.

The reader is introduced to basics of medical ultrasound (US) imaging. The state of the art localization methods are reviewed in the work. Many methods such as Hough transform (HT) or Parallel Integral Projection (PIP) are based on projections. As the existing PIP implementations are relatively slow, we suggest an acceleration by using a multiresolution approach.

We propose to use model fitting approach which uses randomized sample consensus (RANSAC) and local optimization. It is a fast method suitable for real-time use and it is robust with respect to the presence of other high-intensity structures in the background. We propose two new shape and appearance models of tool in 3D US images. Tool localization can be improved by exploiting its tubularity. We propose a tool model which uses line filtering and we incorporated it into the model fitting scheme. The robustness of such localization algorithm is improved at the expense of additional time for pre-processing.

The real-time localization using the shape model is demonstrated by implementation on the 3D US scanner Ultrasonix RP. All proposed methods were tested on simulated data, phantom US data (a replacement for a tissue) and real tissue US data of breast with biopsy needle. The proposed methods had comparable accuracy and the lower number of failures than the state of the art projection based methods.

Keywords: 3D ultrasound images, localization, needle, model fitting, line filtering, real-time application.

Résumé (in French)

Cette thèse traite de la détection automatique d'outils chirurgicaux de géométrie linéaire tels que des aiguilles ou des électrodes en imagerie ultrasonore 3D. Une localisation précise et fiable est nécessaire pour des interventions telles que des biopsies ou l'insertion d'électrode dans les tissus afin d'enregistrer leur activité électrique (par exemple dans le cortex cérébral).

Le lecteur est introduit aux bases de l'imagerie ultrasonore (US) médicale. L'état de l'art des méthodes de localisation est rapporté. Un grand nombre de méthodes sont basées sur la projection comme la transformation de Hough ou la Projection Intégrale Parallèle (PIP). Afin d'améliorer l'implantation des méthodes PIP connues pour être assez lentes, nous décrivons une possible accélération par approche multi-résolution.

Nous proposons d'utiliser une méthode d'ajustement de modèle utilisant une approche RANSAC et une optimization locale. C'est une méthode rapide permettant un traitement temps réel et qui a l'avantage d'être très robuste en présence d'autres structures fortement échogenes dans le milieu environnant. Nous proposons deux nouveaux modèles d'apparence et de forme de l'outil dans les images US 3D. La localisation de l'outil peut être améliorée en exploitant son aspect tubulaire. Nous proposons un modèle d'outil utilisant un filtrage rehausseur de ligne que nous avons incorporé dans le schéma de recherche de modèle. La robustesse de cet algorithme de localisation est améliorée au prix d'un temps additionnel de pré-traitement.

La localisation temps-réel utilisant le modèle de forme est démontrée par une implantation sur l'échographe Ultrasonix RP. Toutes les méthodes proposées on été testée sur des données de simulation US, des données de fantômes (qui sont des tissus synthétiques imitant les tissus biologiques) ainsi que sur des données réelles de biopsie du sein. Les méthodes proposée ont montré leur capacité à produire des résultats similaires en terme de précision mais en limitant d'avantage le nombre d'échecs de détection par rapport aux méthodes de l'état de l'art basées sur les projections.

Mots-clés: imagerie ultrasonore 3D, localisation, aiguille, ajustment de modèle, filtrage rehausseur de ligne, temps-réel

Abstrakt (in Slovak)

Táto téza sa zaoberá automatickou lokalizáciou tenkých chirurgických nástrojov, ako napríklad ihiel alebo elektród, v troj-rozmerných (3D) ultrazvukových obrázkoch. Presná a spoľahlivá lokalizácia je dôležitá pre lekárske zákroky ako biopsia alebo vsunutie elektródy do tkaniny.

Čitateľ je uvedený do základov lekárskeho ultrazvukového zobrazovania. V práci sa skúmajú lokalizačné metódy, ktoré zodpovedajú súčasnému stavu vývoja v oblasti. Je množstvo metód, ako napríklad Houghova transformácia (HT) alebo Paralelná integrálna projekcia (PIP), ktoré sú založené na projekciách. Keďže existujúce implementácie PIP metódy sú pomalé, navrhujeme jej zrýchlenie pomocou prístupu viacnásobného rozlíšenia dát.

Navrhujeme použiť metódu odhadovania parametrov modelu (angl. model fitting), ktorá je založená na hľadaní konsenzu náhodne vybraných vzoriek (skr. RANSAC) a na lokálnej optimalizácii. Je to rýchla metóda vhodná pre použitie v reálnom čase a navyše je robustná voči výskytu iných štruktúr s vysokým jasom v obrázku. Predkladáme dva nové modely nástroja v 3D ultrazvukových obrázkoch, ktoré sú založené na popise tvaru a výzoru. Lokalizácia nástroja môže byť vylepšená s využitím informácie o jeho rúrkovom tvare. Navrhujeme model nástroja, ktorý používa lokálne filtrovanie rúrkových objektov. Parametre tohto modelu sú odhadnuté pomocou RANSAC-u. Navrhnutý lokalizačný algoritmus je robustnejší za cenu zvýšeného času potrebného pre predspracovanie dát.

Lokalizácia v reálnom čase je demonštrovaná implementáciou na 3D ultrazvukovom skeneri Ultrasonix RP. Všetky navrhnuté metódy boli otestované na simulovaných dátach, na ultrazvukových dátach fantómu (umelá vzorka nahradzujúca tkaninu) a na skutočných ultrazvukových dátach prsníku s bioptickou ihlou. Navrhnuté metódy majú v porovnaní s inými sučasnými metódami porovnateľnú presnosť a lepšiu spoľahlivosť meranú počtom správnych výsledkov.

Kľúčové slová: 3D ultrazvukové obrázky, lokalizácia, ihla, odhadovanie parametrov modelu, rúrkovité filtrovanie, aplikácia v reálnom čase.

Acknowledgments

I am grateful to a number of people that helped me accomplish this research. First of all, I would like to thank my advisor Dr. Jan Kybic for a very competent and patient guidance in the world of science. His extensive knowledge, professional methodology and invaluable advices made a significant contribution to the quality of this work. I would also like to thank Prof. Vacláv Hlaváč, the head of the CMP group, who was continously interested in my work. I am grateful to Prof. Mirko Navara and Ing. Martin Dolejší for their comments on the thesis.

I would like to thank my co-advisors from the CREATIS-LRMN in Lyon. I sincerely thank Prof. Christian Cachard whose expert knowledge in medical ultrasound and contacts for clinicians and professionals from industry boosted this research project. My kind thanks go to Dr. Hervé Liebgott whose constant motivation, encouragment and competence created an agreeable working atmosphere.

I would like to acknowledge the WARTHE project[1] and all people who took part in it. I obtained an intensive training in the field of medical imaging and the health engineering. I had an opportunity to discuss my work with the world leading scientists at international conferences and workshops.

My special thanks go to my girlfriend Lucia who was supporting me the whole time by the plenty of love and patience.

[1]EC project MEST-CT-2005-021024 WARTHE

Contents

Notation

Abbreviations

2D, 3D	two-, three-dimensional
CT	computed tomography
FLD	Fisher's linear discriminant
FOV	field of view
GPU	graphical processing unit
GUI	graphical user interface
HT	Hough transform
MR	multi-resolution
MRI	magnetic resonance imaging
PCA	principal component analysis
PIP	Parallel Integral Projection
PSF	point spread function
PVA	polyvinyl alcohol
RANSAC	randomized sample consensus
RF	radio-frequency
RHT	randomized Hough transform
RT	Radon transform
ROC	receiver operating characteristics
ROI	region of interest
SNR	singal to noise ratio
SVM	Support Vector Machine
US	ultrasound

Mathematical symbols

$\mathbb{R}$	set of real numbers
$\mathbb{R}^n$	n-D Euclidean space
x, λ	scalar
α, β	angles in degrees
$\mathbf{x}, \mathbf{X}$	vector, matrix
$\mathcal{X}$	a discreete set
$\|x\|$	absolute value of scalar x
$\|\mathbf{x}\|$	Euclidean norm of vector $\mathbf{x}$
$\|\mathcal{X}\|$	the number of elements of a finite set $\mathcal{X}$
$\mathbf{a} \cdot \mathbf{b}$	dot product of vectors $\mathbf{a}$ and $\mathbf{b}$
$I(\mathbf{x})$	image function $I : \mathbb{R}^3 \to \mathbb{R}$
$I(\mathbf{x}) * K(\mathbf{x})$	convolution of image I with mask K
I_x, I_y, I_z	partial derivatives of the image I
$\mathcal{H}$	Hessian matrix of the second order partial derivatives (5.6)

Chapter 1

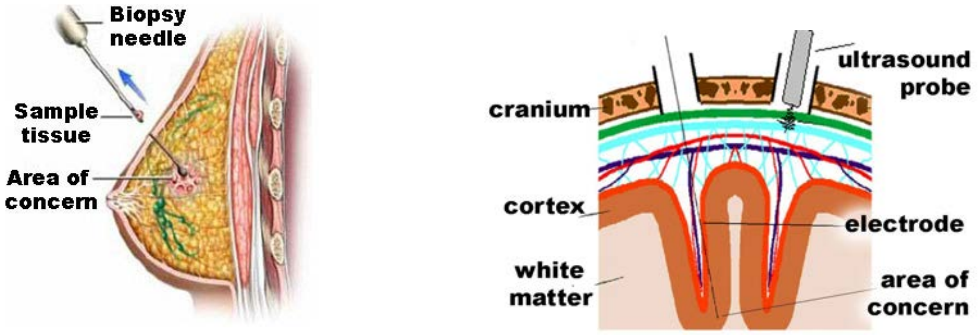
Introduction

Small instruments such as needles and electrodes are inserted into the biological tissue in the surgical practice. Their precise location is required to be known during the interventions which is a motivation for this work (Section 1.1). Our aim is to develop an automatic localization method for tools in 3D ultrasound images (Section 1.2) which shows the location of the tool in surrounding tissue.

We introduce the reader to basics of ultrasound imaging in the remaining part of this Chapter (Section 1.3). The state of the art localization methods are reviewed in the next Chapter 2. We describe two localization methods in detail: the Parallel Integral Projection in Chapter 3, and the model fitting which uses a randomized algorithm in Chapter 4. The line filtering is proposed for improvement of robustness of model fitting in Chapter 5. The model fitting is demonstrated as real-time localization in the last Chapter 6. We conclude the thesis with a summary and future work in Chapter 7.

This work is a result of collaboration between the Center for Machine Perception (CMP)[1], Department of Cybernetics, Czech Technical University (CTU) in Prague and the CREATIS-LRMN[2] in Lyon, which is a common laboratory of Institut National des Sciences Appliquées (INSA), Université Claude Bernard (UCB) Lyon 1, CNRS UMR 5220 and INSERM U630. The project was funded in the framework WARTHE[3] (Wide Applications and Research Training in Health Engineering) which is a part of European Comission's Marie Curie Actions for Early Stage Training FP6 human resource programme. The author started his work in May 2007 and during the PhD studies was supervised by Dr. Jan Kybic at CMP and co-supervised by

[1] `http://cmp.felk.cvut.cz/`
[2] `http://www.creatis.insa-lyon.fr/`
[3] `http://www.warthe.eu`



(a) (b)

Figure 1.1: *Motivation for surgical tool localization: (a) biopsy needle localization, e.g. in breast, (b) electrode localization in brain tissue.*

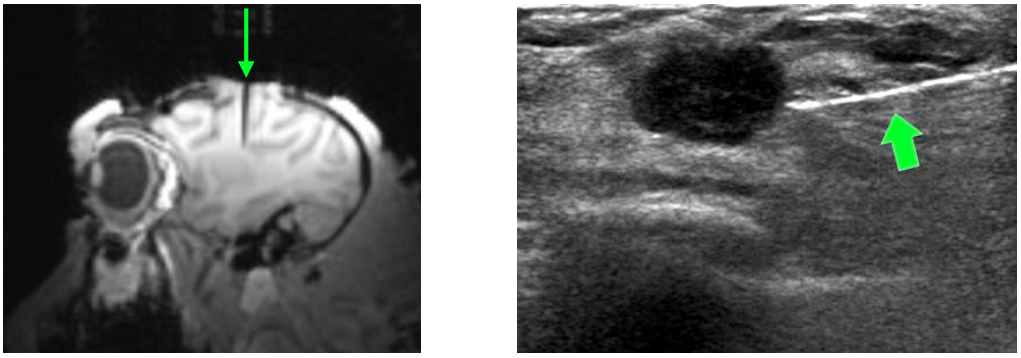
Prof. Christian Cachard and Dr. Hervé Liebgott at CREATIS.

1.1 Motivation and clinical applications

Minimally invasive surgical procedures often involve insertion of a thin tubular instrument into the human body. In a percutaneous biopsy (e.g. breast, liver, prostate), tissue samples are taken from a region of interest by means of a thin needle [1] (Figure 1.1a). In prostate brachytherapy, small radioactive rods are inserted inside the tissue by a hollow shaft [70]. For a neurological research, the electrical activity of a specific group of neurons is recorded by a thin electrode [3] (Figure 1.1b). The electrode is inserted through a hole in the skull and the ultrasound image is observed through a second hole. In peripheral nerve block (PNB) anaesthesia, a needle delivering an anaesthetic medicine is inserted close to the nerve and the arm or leg is desensitized [54]. In all these cases it is important to localize the instruments, to know their precise position which is the subject of our research. The desired localization accuracy depends on the application: for needle biopsy it is 2–3 mm, for electrode insertion, it is around 0.3 mm.

1.1.1 Guidance of surgical instruments

The precise navigation of surgical instruments is vital for reducing the damage to the tissue caused by the insertion to a wrong location. Instrument miniaturization makes the tool placement more challenging because the needle may bend due to steering [26].



(a) (b)

Figure 1.2: *Medical imaging modalities used for guidance of surgical instruments: (a) magnetic resonance image of tungsten electrode inserted into brain (Institut des Sciences Cognitives, Lyon), (b) ultrasound image of a core biopsy needle in the tissue close to a round tumor [103].*

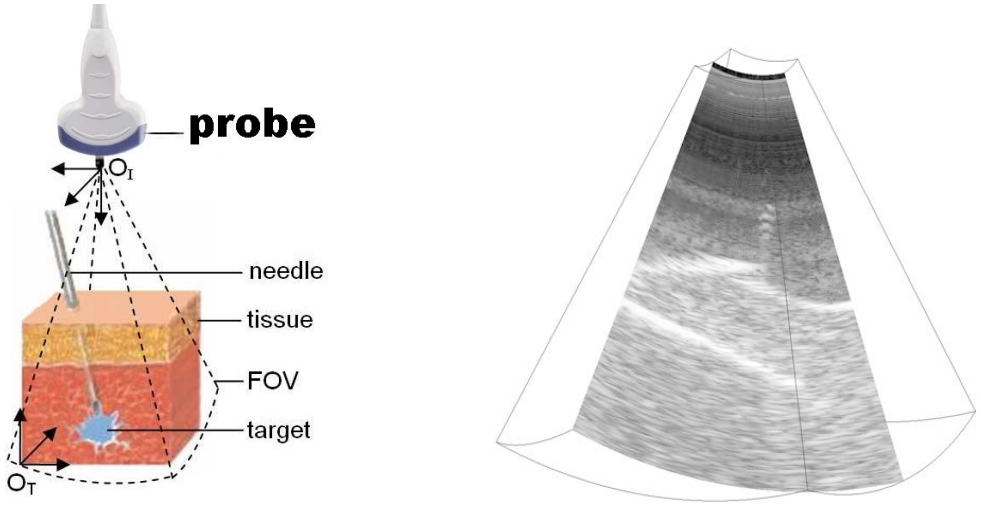
A stereotactic frame for instrument guidance on small animals was first introduced in 1908 [55]. The frame is fixed to a solid part of body, e.g. bone, and allows precise navigation with respect to external anatomical landmarks.

To avoid the limitations of stereotaxy and patient discomfort, frameless techniques were proposed, e.g. the spatial localization of a tool using a radio-frequency signal [105], or optical tracking with two calibrated cameras [35]. Glossop et al. [47] propose a magnetic sensor embedded in the needle tip that allows to track it in the surrounding magnetic field. Nowadays, stereotaxy is sometimes combined with medical imaging techniques [79], e.g. MRI, CT, or ultrasound imaging (Figure 1.2), and the localization accuracy achieves the order of tenth of a millimeter [86].

In this work, we focus on ultrasound (US) imaging of miniature surgical tools as the US modality has several advantages: no ionizing radiation is produced and the cost is relatively modest; ultrasound imaging is a standard technique in clinics and offers a real-time acquisition speed. Our goal is to develop localization method which relies only on image data. Unlike with the MRI that is not compatible with metallic tools, ultrasound does not require any change in the clinical procedure.

1.1.2 Visualization of surgical tools

The surgical instruments are observed in 2D ultrasound images together with the surrounding tissue because the guidance task might require to reach a particular target or avoid some anatomical structures. The physicians must be trained to maintain the observation plane incident with the tool. The problem can be partially



(a) (b)

Figure 1.3: *Ultrasound imaging of surgical tool: (a) probe observing the tissue, (b) example of acquired 3D ultrasound image.*

solved by fixing the needle to the probe [98]. A less restrictive solution is to acquire 3D data, then localize the tool and present a suitable 2D slice to the radiologist. The tool marked by a colour line can be shown in the incident slice and a perpendicular slice can be presented to give a view of surrounding tissue. When 3D data are used, physician does not need to maintain the observation plane anymore. Ultrasound image together with surgical instrument can be shown in augmented reality [74].

1.2 Task definition

The task addressed here is to automatically determine the 3D location of an elongated tool, e.g. needle or electrode, in the US image of the biological tissue. The coordinate system is defined relatively to the probe position (Figure 1.3a). The tool is usually straight or slightly bended due to the lateral forces during insertion and steering [26]. Bending is typical for thin electrodes (diameter around 0.3 mm), unlike biopsy needles which are thicker (diameter around 1 mm) and stay straight. The tool appears as a long thin cylinder with diameter around 1 mm in ultrasound image (Figure 1.3b). The tool voxels usually have the intensity higher than the surrounding voxels, but the appearance is irregular because of the speckle noise which is present in US images.

The localization task is decomposed into two sub-tasks:

- Axis localization (direction of tool) – it is important during insertion/steering,

e.g. for identification of plane incident with the tool.

- Tip localization (endpoint of tool) – it performs a precise identification of tool tip during intervention. Typically, the needle is inserted from outside of observed area so there is at most one endpoint in the image.

The result should be computed from 3D volume in real-time, i.e. faster that the acquisition speed (approximately 1 s for our 3D probe connected to Ultrasonix (Vancouver, BC, Canada) RP scanner and performing a mechanical sweeping in the azimuthal dimension).

1.3 Ultrasound image formation

In this section, the basics of ultrasound imaging are summarized including sound propagation, instrumentation and image formation. Medical ultrasound exploits the backscattering of acoustic energy from biological tissues. A series of pressure pulses are transmitted into the body along pre-determined trajectories, each of them forming a narrow ultrasound beam. Images are computed from backscattered signals received for each beam.

The frequencies used in medical practice are typically in the range from 1 to 20 MHz. The ultrasound acquisition is relatively fast and a real-time imaging is possible with a frame rate from 5 to 80 frames per second for the 2D US. The tissue is not mechanically altered in any way. Ultrasound scanners are relatively cheap compared to other medical imaging modalities.

Ultrasound image acquisition is view-dependent (unlike tomography), and is affected by a strong speckle noise [15]. However, the imaging of some types of tissue such as bones or lungs is difficult, and the interpretation of the ultrasound images needs to be done by a trained specialist.

1.3.1 Sound propagation in medium

The tissue is a medium for the sound propagation where an exchange of the kinetic energy and the potential energy takes place. The waves are primarily longitudinal in water and biological tissues. The waves are propagated by the tissue particles which are alternately compressed and decompressed (Figure 1.4).

The wave equation describes the propagation of pressure $p(x,t)$ which is a function of the position x and time t. A longitudinal wave in homogeneous medium

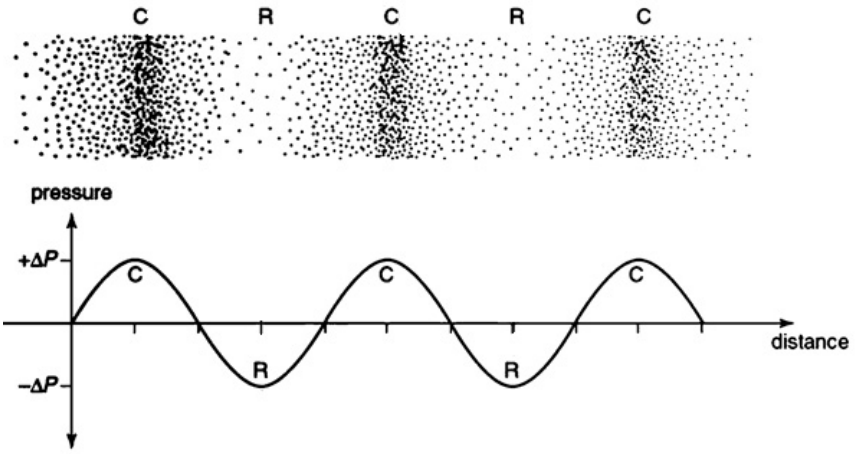


Figure 1.4: *Illustration of the propagation of a 1D longitudinal wave in homogeneous, loss-less medium [5]. The pressure $p(x,t)$ is alternately compressed (C) and decompressed (R).*

without attenuation is described in 1D by a second-order differential equation [5, 6]:

$$\frac{\partial^2 p(x,t)}{\partial x^2} - \frac{1}{c^2}\frac{\partial^2 p(x,t)}{\partial t^2} = 0 \tag{1.1}$$

The pressure $p(x,t)$ is related to the sound velocity c in the medium. Using the medium compressibility κ [Pa^{-1}], the sound velocity c is expressed in terms of medium density ρ:

$$c = \frac{1}{\sqrt{\kappa\rho}} \tag{1.2}$$

The sound velocity for some materials is summarized in Table 1.1. In the case of a harmonic sound wave with frequency f, the propagation velocity is given by

$$c = \lambda \cdot f \tag{1.3}$$

where λ is the sound wavelength in the medium.

The acoustic properties of the medium are characterized by the acoustic impedance which is an opposition to the flow of sound through a surface. The acoustic impedance is defined as the pressure p per velocity v per area S:

$$Z_0 = \frac{p}{vS} \tag{1.4}$$

The specific acoustic impedance Z of is the ratio of sound pressure p to particle

	c [m·s^{-1}]	Z [10^{-6}·kg·m^{-2}·s^{-1}]
Air	343	0.0004
Fat	1450	1.38
Water	1484	1.48
Blood	1550	1.61
Kidney	1560	1.62
Liver	1570	1.65
Skull bone	3360	6.00
Aluminium	6420	17.00

Table 1.1: *Sound velocity c and specific acoustic impedance Z for some materials and biological tissues [5].*

velocity v at a point:

$$Z = \frac{p}{v} = Z_0 S \tag{1.5}$$

The unit rayl defined as 1 [rayl] = 1 [kg·m^{-2}·s^{-1}] is commonly used to quantify the specific acoustic impedance. The specific acoustic impedance of selected materials and biological tissues is listed in Table 1.1.

Reflection and transmission

When the wave is propagated in tissue, a certain portion of the incident energy is reflected (I_r) while the remaining portion is transmitted (I_t) (Figure 1.5) [5, 6]. Assuming a planar boundary of two homogeneous media (with dimensions much greater than the wavelength λ), their portion can be quantified by the intensity reflection coefficient R_I and intensity transmission coefficient T_I

$$R_I = \frac{I_r}{I_i} = \frac{\left(\frac{Z_2}{\cos\theta_t} - \frac{Z_1}{\cos\theta_i}\right)^2}{\left(\frac{Z_2}{\cos\theta_t} + \frac{Z_1}{\cos\theta_i}\right)^2} \tag{1.6}$$

$$T_I = \frac{I_t}{I_i} = 1 - R_I \tag{1.7}$$

where Z_1, resp., Z_2 is the specific acoustic impedance of the first, resp., the second medium. The equations governing the angles of incidence θ_i, reflection θ_r and transmission θ_t are given by

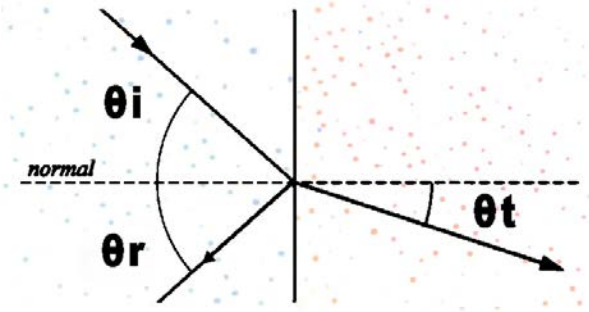


Figure 1.5: *Transmission and reflection of a sound wave on a planar boundary between two media with distinct acoustic properties: the specific acoustic impedances $Z_1 \neq Z_2$ and sound velocities $c_1 > c_2$. The angles of incidence θ_i, reflection θ_r are equal and the angle transmission θ_t is changed because of the different speed of sound c in the medium.*

$$\theta_i = \theta_r, \tag{1.8}$$

$$\frac{\sin\theta_i}{\sin\theta_t} = \frac{c_1}{c_2} \tag{1.9}$$

where c_1 and c_2 are the sound velocities in the first, resp., the second medium. In case of different c_1, c_2, the transmitted wave is deviated from the direction of incident wave leading to refraction artifacts in an image.

For the ultrasound wave propagation inside the human body, the angles θ_i, θ_t are often not too big, so we neglect them $\theta_i = \theta_t = 0$ for the simplicity.

The amount of reflected energy depends only on the ratio between Z_1, Z_2:

$$R_I = \frac{I_r}{I_i} = \frac{(Z_2 - Z_1)^2}{(Z_2 + Z_1)^2} \tag{1.10}$$

Using values of the specific acoustic impedances listed in Table 1.1, it can be shown that the intensity of the reflected sound wave from an interface within soft tissue is typically around 0.1% of the incident intensity. The reflection on other interfaces, e.g. bones, can be stronger because of the higher specific acoustic impedance.

Scattering, absorption and attenuation

Incident wave travels through medium particles with dimensions smaller or comparable to the wavelength, its energy is *scattered* into many directions in the form of a spherical wave and thus propagates in all directions. Such waves interact through

positive and destructive interferences resulting in a typical speckle noise pattern of ultrasound images. The magnitude of scattered energy depends on many factors including the shape, the size and the acoustic properties of scatterers.

Absorption is a process of converting the acoustic energy to heat when a sound wave propagates through the medium. The energy is absorbed due to a relaxation phenomenon, which is a force acting against the movement of displaced particles, is also taking place. This force is trying to restore the equilibrium and it responsible for the energy losses of the absorption process.

The loss of intensity (or energy) of the forward propagating wave because of scattering and absorption is denoted *attenuation*. It is approximated by an exponential function of the distance x and linearly depends on the initial intensity [5]

$$I(x) = I(0) \cdot \mathrm{e}^{-\mu x} \tag{1.11}$$

where μ [cm^{-1}] is the intensity attenuation coefficient. Its value is often given in units of decibels per centimeter μ [$\mathrm{dB \cdot cm}^{-1}$]. The coefficient μ depends on the tissue type and is proportional to the frequency f. The attenuation in biological media is approximately $1\,\mathrm{dB \cdot cm}^{-1} \cdot \mathrm{MHz}^{-1}$. Decreasing the frequency permits to investigate deeper located parts of a body at the price of decreasing the axial resolution which is inversely proportional to the frequency.

1.3.2 Design of transducers

The key component for the ultrasound system is a transducer which converts the electrical signal to the mechanical energy and vice versa. It operates in *a pulse-echo* mode which perform two steps: transmission and recording [4]. In transmission mode, the transducer acts as loudspeaker sending out acoustic pulse which are transmitted into the body as mechanical vibrations. Subsequently, the transducer acts as microphone which records the received sound waves and converts them into electrical signals. Afterwards the time-varying signal is processed and presented as an ultrasound image (Section 1.3.3). Assuming the sound velocity in a tissue to be known (1540 $\mathrm{m \cdot s}^{-1}$), the distance from the transducer is determined from the time elapsed from transmission to the reception.

Transducers are usually composed of an array of small piezoelectric crystals. The dimensions of the single crystal is periodically changed by applying harmonic voltage. The array configuration is used for electronic beam forming and dynamic focusing. The focusing is a beamforming technique which changes the shape of the

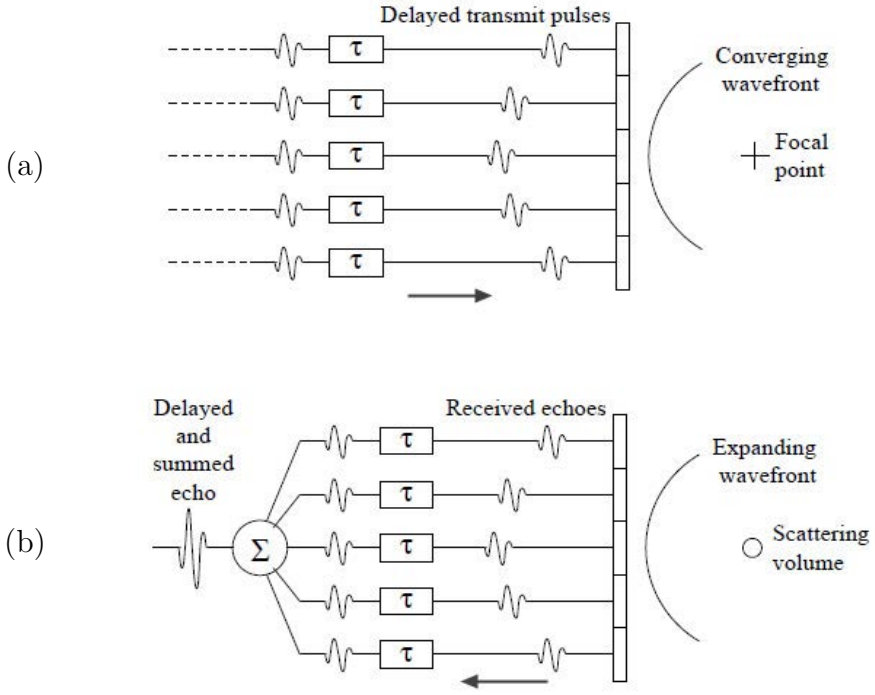


Figure 1.6: *Physical layout of a five-element transducer with a 2D FOV and phased array beamforming: (a) bunch of pulses transmitted with a delay pattern achieves steering and focusing of wave; (b) echoes returning to the transduced are delayed and summed together. [4]*

propagating wave so that the energy is concentrated at specific depth. It allows to increase the resolution of the US scanner. The physical layout of crystals on the transducer determines the shape of the observed field of view (FOV) and its dimensionality which is discussed in the following text.

Linear and sector 2D scanning

Transducers for 2D scanning consist of a large number (64–512) of piezoelectric crystals arranged in a single row. Exciting few adjacent crystals by a voltage pulse, a sound beam is produced (Figure 1.6a). Afterwards the echoes backscattered from the medium are received by the same group of crystals (Figure 1.6b). [5, 6]

The ultrasound probes usually use two configurations of transducers:

1) The *linear transducers* consist of crystals arranged in a line. Parallel beams are produced and they provide 2D images with a rectangular FOV. The FOV can be extended by steering beams at ends of the array. A probe with the linear transducer

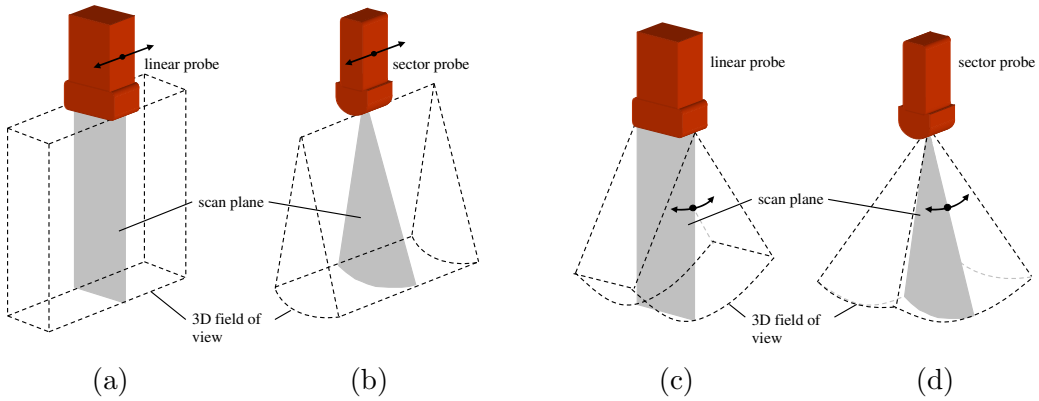


Figure 1.7: *(a) Linear or (b) sector transducer with 2D FOV is translated to acquire a series of parallel 2D images. (c) The scan plane of a linear or (d) sector probe is tilted while 2D images are acquired with a predetermined angular step.*

is suitable, e.g., for observation of shallow objects in the body such as carotid arteries or cysts in the liver.

2) The *sector transducers* use the same pulse excitation as linear transducer but the array of crystals is placed on a convex curve. They produce a fan-shaped field of view and therefore they have a larger FOV. A probe with the sector transducer is suitable, e.g., for observation of the hearth from the space between ribs.

Scanning with a 3D FOV

The matrix transducers have piezoelectric elements arranged in a rectangular grid form. It allows to steer the ultrasound beam electronically in two directions, creating a 3D field of view. The resolution of such systems is better than other 3D probes because the focusing is done in two dimensions. [36]

The matrix transducers are at the level of development and they are not standard yet in industry. Instead, many 3D imaging systems acquire data by scanning 2D scan planes and composing them into 3D volume. The mechanism inside the probe allows to tilt or shift the 2D transducer in the acquisition procedure (Figure 1.7). The acquisition time is increased multiple times (to around 1 s) compared to matrix transducers but is still sufficient for medical examinations.

1.3.3 Image reconstruction

The acquisition of ultrasound data consists of transmission of pulses and recording of echoes from the tissue which explore the space in narrow beams. The beams are

organized in an array of $N_b \times N_p$ in case of 3D data acquisition. There are N_p planes and each plane consists of N_b beams. For each beam $r_{i,j}$, a radio-frequency (RF) signal of length N_s is recorded over time at a sampling rate f_s. The RF signal is digitized to $r_{i,j}[n]$ and it is processed by a sequence of operations (Figure 1.8) in order to generate ultrasound image.

Band-pass filtering. Received RF signals can have some frequency components outside the frequency band of the transducer. These components correspond to noise and they need to be suppressed. Therefore, RF signals are filtered with a band-pass filter whose parameters correspond to the frequency band of the transducer. Let us denote the filtered RF signals by $f_{i,j}[n]$.

Envelope detection. Image intensities shown on the ultrasound screen are derived from the energy of the filtered RF signals. The common way of computing the RF signal envelope is using the Hilbert transform $\mathcal{H}\{\cdot\}$ [5] which is defined for a continuous real-valued signal $x(t)$ as a complex function

$$\hat{x}(t) = \mathcal{H}\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t-\tau} \mathrm{d}\tau \tag{1.12}$$

The Hilbert transformation is interpreted as the output of $\frac{\pi}{2}$ phase shifter with input $x(t)$ and the impulse response $h(t) = \frac{1}{\pi t}$. The amplitude envelope $e(t)$ of a signal $x(t)$ is computed as the complex module of the sum of the original signal and its Hilbert transformation

$$e(t) = |x(t) + j\hat{x}(t)| = |x(t) + j\,(h(t) * x(t))| \tag{1.13}$$

where j is a complex unit and $*$ denotes a convolution. The US scanners often use an approximation of the Hilbert transform such as quadrature demodulation [4].

Logarithmic compression. The dynamic range of the RF signals and envelope signals $e_{i,j}[n]$ is up to 80 dB which makes the simultaneous perception of high and low intensities difficult for a human observer. In most ultrasound imaging systems, the amplitude of envelope signals is compressed before displaying by a logarithmic function

$$c_{i,j}[n] = k_1 \log\left(e_{i,j}[n] + k_2\right); \tag{1.14}$$

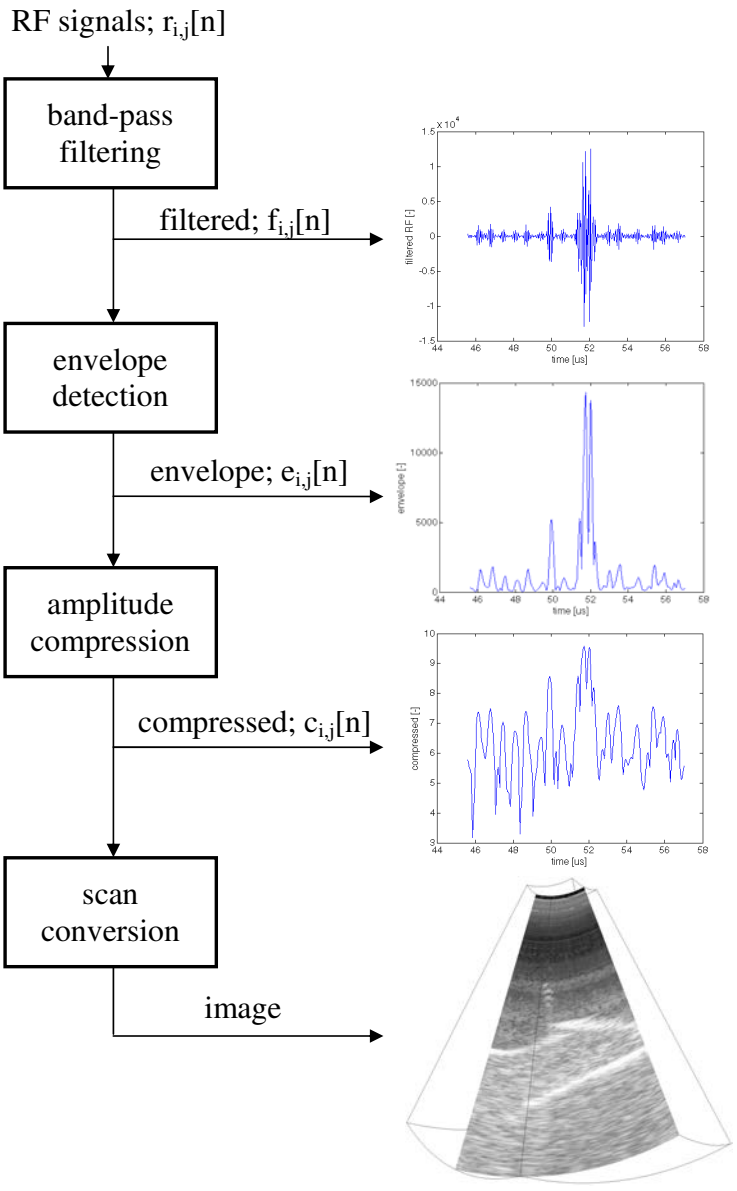
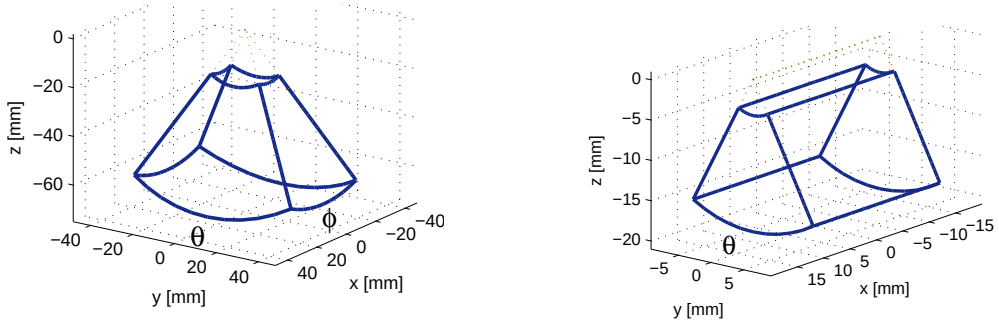


Figure 1.8: *Block scheme [9] illustrating the reconstruction of an ultrasound image from acquired RF signals. Intermediate data are shown on the right side.*



(a) (b)

Figure 1.9: *The geometry of 3D ultrasound data showed as wireframe model of the volume boundary: (a) sectorial geometry (b) book-shape geometry.*

where k_1, k_2 are parameters influencing the degree of amplitude compression. Jensen [60] proposes for the purpose of visualization to set both parameters k_1, k_2 to 1.

Scan conversion. The values in $c_{i,j}[n]$ specify the intensity of voxels aligned along the i-th ultrasound beam on the j-th scan plane. To display the acquired image, we must determine for each $c_{i,j}[n]$ its coordinates in a Cartesian coordinate system. The explicit conversion formula depends on the probe used for acquisition. We give examples of two types of geometry from Figure 1.7: sectorial (c) and book-shaped (d) geometry.

In case of a 3D probe with sector geometry (Figure 1.9a), the conversion formula for computation of the Cartesian coordinates $[x, y, z]$ from the indices $[i, j, n]$ is

$$\begin{bmatrix} x(i,j,n) \\ y(i,j,n) \\ z(i,j,n) \end{bmatrix} = \frac{nc}{2f_s} \begin{bmatrix} \sin(\phi_i) \\ \cos(\phi_i)\sin(\theta_i) \\ \cos(\phi_i)\cos(\theta_i) \end{bmatrix} \tag{1.15}$$

where θ_j is a tilt angle of the j-th plane with respect to axis z and ϕ_i is a scan angle of i-th beam in the j-th plane.

In case of a 3D probe with book-shaped geometry (Figure 1.9b), the conversion formula for computation of the Cartesian coordinates $[x, y, z]$ from the indices $[i, j, n]$ is

$$\begin{bmatrix} x(i,j,n) \\ y(i,j,n) \\ z(i,j,n) \end{bmatrix} = \begin{bmatrix} s_x x(i,j,n) \\ \frac{nc}{2f_s}\sin(\theta_i) \\ \frac{nc}{2f_s}\cos(\theta_i) \end{bmatrix} \tag{1.16}$$

where θ_j is a tilt angle of the j-th plane with respect to axis z and s_x is a scale

factor of with respect to axis x.

The conversion usually includes computation of intensities for each voxel $v_{x,y,z}$ in an axis-aligned 3D grid by backward interpolation into the original volume data $c_{i,j}[n]$. In case of visualization of a section plane, it is enough to compute coordinates of points in the plane and compute their intensities by interpolation. The data are presented to the screen as perpendicular slice sections (Section 1.1.2).

1.4 Observation of tool in ultrasound images

In clinical practice, localization of surgical tools in ultrasound images is mostly done by a human expert [57]. Clinicians need to interact with the tool in the tissue in order to guide it to the target area. However, automatic methods (Chapter 2) can help with the localization.

The 2D ultrasound imaging is a common standard but it is limited by observing only one plane at a time. A larger volume of interest is observed by the 3D ultrasound and the guidance task becomes easier [16]. It is more difficult for radiologist to handle the 3D data, so there is an increased demand for an automatic method.

Various non-image based solutions have been proposed to improve the needle contrast in the ultrasound image [17], [20]: passive markers at the needle tip which enhance the ultrasound signal [91]; the adaptive ultrasound beam steering which enhances the energy of reflected ultrasound wave by sending it more perpendicular to the needle [19]. In this work, we focus on pure ultrasound image processing based methods and therefore modifications of the surgical tools or the clinical protocol are not necessary.

Difficulties of US imaging of metallic tools. A metallic tool usually has a much higher acoustic impedance than the surrounding tissue (Figures 1.10 and 1.11). Thus the reflections from tool appear as high intensity pixels in the ultrasound image. A problem is that varying the angle between the tool and the ultrasound beam might make the signal to noise ratio (SNR) lower because metallic tools are not perfect diffuse reflectors (Figure 1.11b). Some types of tissue (e.g. fat, bones) have a similar high-intensity appearance as the tool (Figure 1.10).

Acoustic shadows, attenuation and other artifacts can also locally obscure the continuous appearance of the tool [56] (Figure 1.10). Moreover, specific combinations of the tool (such as a needle with a hole inside) and the transmission frequency can generate reverberation artifacts [87] (Figure 1.11a).

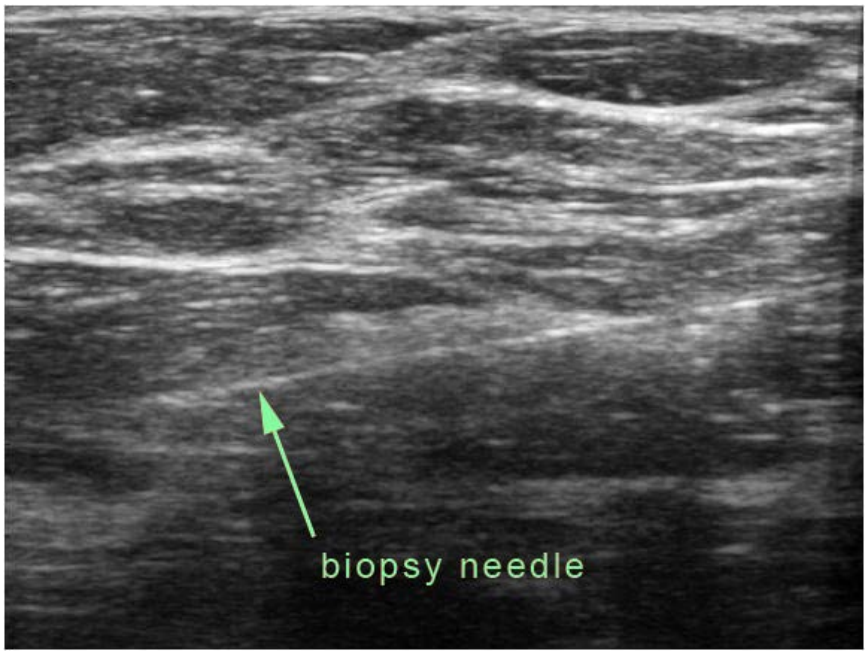
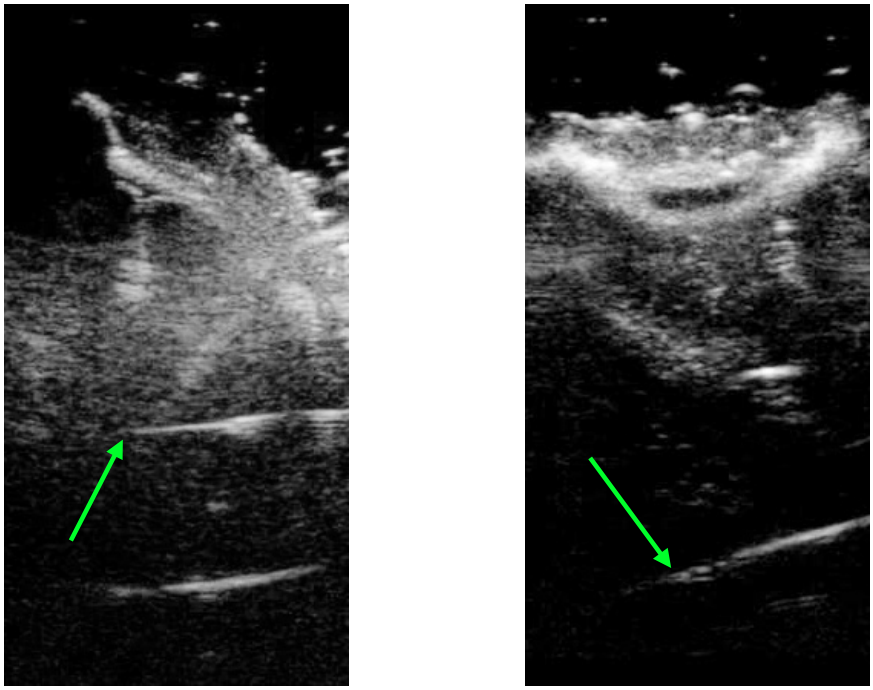


Figure 1.10: *2D slice of 3D ultrasound image of a biopsy needle in the breast tissue. The US image was acquired by GE Voluson E8 scanner with a* 12 MHz *probe and the needle has a diameter* 1.092 mm*. The high intensity structures in the top are 2D layers of fat tissue. The needle is straight (marked by the green arrow) but the appearance is irregular because of speckle noise and shadowing by the particles above the tool.*



(a) (b)

Figure 1.11: *2D ultrasound images of a tungsten micro-electrode in the brain tissue. The US image was acquired by Phillips US scanner with a* 15 MHz *probe. The structures in the top belong to the cortex or white matter. The electrodes are in the bottom part, marked by the green arrow. The electrode in the figure (a) has a tail on the right side. Its appearance is interrupted probably by a shadow. The electrode in the figure (b) has varying contrast. The contrast on the tip (where the green arrow points) is lower than in the middle part of the tool.*

Chapter 2

State of the art

A variety of algorithms for elongated object localization in ultrasound (US) data have been proposed. Most of them assume that the object, i.e. surgical tool in US image, appears brighter than background voxels and that the tool axis is straight.

A method based on the Principal Component Analysis analysis is described in Section 2.1. Many methods use a kind of integral projection: the standard Hough transform (Section 2.2) and its modifications, e.g. the randomized Hough transform (Section 2.3) or the Parallel Integral Projection (Section 2.5). The methods based on the Hough transform are applied to a straight tool localization in US images (Section 2.4). Another approach is a model fitting method (Section 2.6) using a randomized algorithm which is a fast method able to estimate also a curved tool. The method for tool tip localization is described in the last Section 2.7.

The survey of various tool localization methods was written in the author's thesis proposal [P7]. We have presented the review of needle localization methods at the *International Congress on Ultrasonics* (ICU) 2009 [P6].

2.1 Principal component analysis method

The detection of elongated tubular object in thresholded image is done using the *Principal component analysis* (PCA) [62]. The dominant eigenvector is detected (other eigenvectors are assumed to be smaller) using the eigenanalysis of points belonging to cylindrical objects.

Tool detection using PCA in a 2D image. Draper [30] uses a 2D ultrasound (US) image incident with a needle to track the biopsy needle in the breast tissue.

First, the local variance image I_1 of the original US image is computed to enhance the tool voxels and suppress the background noise. The size of the local window and the variance is found as a trade-off of the needle enhancement and computational time.

Then, a binary image is created by adaptive thresholding. The intensity of pixels in I_1 is modelled as a single normal distribution with the mean value μ and variance σ^2. The parameters of distribution are estimated from the training data. An appropriate threshold as $T = \mu + k\sigma$, where the value of $k > 0$ is found experimentally [30].

A morphological closure with a 3×3 symmetrical structural element is applied in order to obtain a smooth tool contour. The morphological closure also reduces the possibility of accidentally connecting the clusters of various structures in the image.

The clusters C_i $(i = 1, .., N)$ are found by a connectivity analysis using an 8-pixel neighbourhood. Let us denote by $\mathbf{R}_i$ the covariance matrix of pixel coordinates in the cluster C_i, Eigenvectors $\mathbf{v}_{i,1}, \mathbf{v}_{i,2}$ of $\mathbf{R}_i$ and the corresponding eigenvalues $\lambda_{i,1} \geq \lambda_{i,2} \geq 0$ are determined using the PCA. The line approximating a cluster C_i is defined by its mean $(\overline{x}, \overline{y})$ and the eigenvector which corresponds to the largest eigenvalue. The needle is then identified among the needle candidates as the cluster with the largest eigenvalue $\lambda_{i,1}$.

The location of the end-point (tip) is found in the binary image on the estimated tool line. Clusters along the line separated by a gap smaller than the electrode diameter are joined together. The tool tip is found as the end-point of the largest cluster.

Tests were performed on a tissue mimicking phantom containing a biopsy needle of diameter 2.1 mm with US scanner operating at 5.5 MHz [30]. The tool axis and tip in 2D image were determined by the algorithm with accuracy of 1 mm for a depth of insertion greater than 15 mm.

Tool detection using PCA in 3D volume. Novotny et al. [75] proposed an extension of Drapper´s method in 3D. The binary pre-segmentation of volume is created by thresholding. Connected components (using a 26-voxel neighborhood) are analyzed using the PCA. The longest and the straightest candidate are selected by maximizing the ratio of the first and second eigenvalue.

The experiment was done with a cylindrical rod of diameter 6.2 mm submerged in water tank close to the tissue sample. The algorithm succeeds to segment the rod from tissue which was located at distance 2 mm. The mean tip accuracy was 0.7 mm and the mean variance 0.6 mm.

The problem with the PCA based methods is that the component belonging to the tool has to be continuous and the appearance of the tool is not continuous in the real case.

2.2 Standard Hough transform

In the following two sections, the Hough transform (Section 2.2) and the randomized Hough transform (Section 2.3.2) are explained. Both of them convert the image into the space of line parameters. Section 2.4 discuss various applications of the HT to tool localization in US image.

The *Hough transform* (HT) was proposed for line detection in 2D images [32]. The idea is to use a voting scheme and dual representation of the line. A line in the image space represented as

$$\text{Line}_{a,b} : \{(x,y) \in \mathbb{R}^2 \mid y = ax + b\} \tag{2.1}$$

corresponds a single point (a,b) in the parameter space (also called the Hough space). Each point (x,y) in the image space is transformed into a set of possible line parameters (a,b) such that a line $\text{Line}_{a,b}$ passes through (x,y):

$$\text{Point}_{x,y} : \{(a,b) \in \mathbb{R}^2 \mid y = ax + b\} \tag{2.2}$$

which is a line in the Hough space. The intersection of all such lines in the Hough space is a single point (a,b) and represents the desired line in the image space.

Practically, the input image $f(x,y)$ is thresholded and transformed point-by-point into a set of lines in the Hough space. The accumulator is incremented for the set of points (a,b) in the Hough space corresponding to the point (x,y) in the image space according to (2.2). The Hough space is discretized using an accumulator array which is a uniform grid over the bounded parameter space (Algorithm 1). The accumulator grid must be sufficiently fine for accurate localization of the thin tool but significantly coarse so that bins get enough votes to allow unambiguous solution. The lines are detected as local peaks in the Hough space. The local maximum is detected in the local neighbourhood window $N \times N$, where N is a parameter, e.g. $N = 5$.

Note that better parametrization of line is using polar coordinates (θ, r) [32]

Input:
$f(x, y)$ — input image.
Output:
$\{(\theta_i, r_i) \mid i \geq 0\}$ – set of detected lines.
forall (θ, r) **do** $A(\theta, r) \leftarrow 0$;
forall $(x_i, y_i) \in \mathbb{R}^2$ *s.t.* $f(x, y) \geq T$ **do**
  Compute the set of bins $\{(a_{ij}, b_{ij})\}$ corresponding to (x_i, y_i) according to (2.2) or (2.3):
  foreach (a_{ij}, b_{ij}) **do**
    $A(a_{ij}, b_{ij}) \leftarrow A((a_{ij}, b_{ij})) + 1$;
  end
end
Find local maxima $\{(\theta_i, r_i)\}$ in accumulator $A(\theta, r)$;

Algorithm 1: *The HT for line detection in a thresholded image* $f(x, y)$.

(Figure 2.1):

$$\text{Line}_{\theta,r} : \{(x, y) \in \mathbb{R}^2 \mid y \sin\theta + x \cos\theta = r\} \tag{2.3}$$

because it avoids the singularity of a line parallel to the y-axis. The parametrization (2.3) leads to a more uniform distribution of points in the parameter space when considering all lines which admissible in the volume boundary.

Relation to the Radon transform. The Hough transform with the parametrization in polar coordinates (2.3) is related to the *Radon transform* (RT) [51] which is defined as integral of the image function $f(x, y) \in \mathbb{R}$ over curve L

$$R(\theta, r)[f(x, y)] = \int_L f\left(x(t), y(t)\right) dL(t) \tag{2.4}$$

where the integration is performed with respect to measure $dL(t)$. We obtain the HT with 2 parameters when doing the RT along a straight line:

$$L(t) : [x(t), y(t)] = t\,[\sin\theta, -\cos\theta] + r\,[\cos\theta, \sin\theta] \tag{2.5}$$

where r is the distance of line from the origin and θ is the angle to x-axis. Note that (2.5) is a parametric form of (2.3). It is differentiable by t which is required by the Radon transform (2.4).

The image transformed into the Radon (parameter) space (θ, r) is often called a *sinogram* and it is used in computer tomography [71] for an image reconstruction.

The HT can be also extended for detection of other parametric shapes [8], e.g.

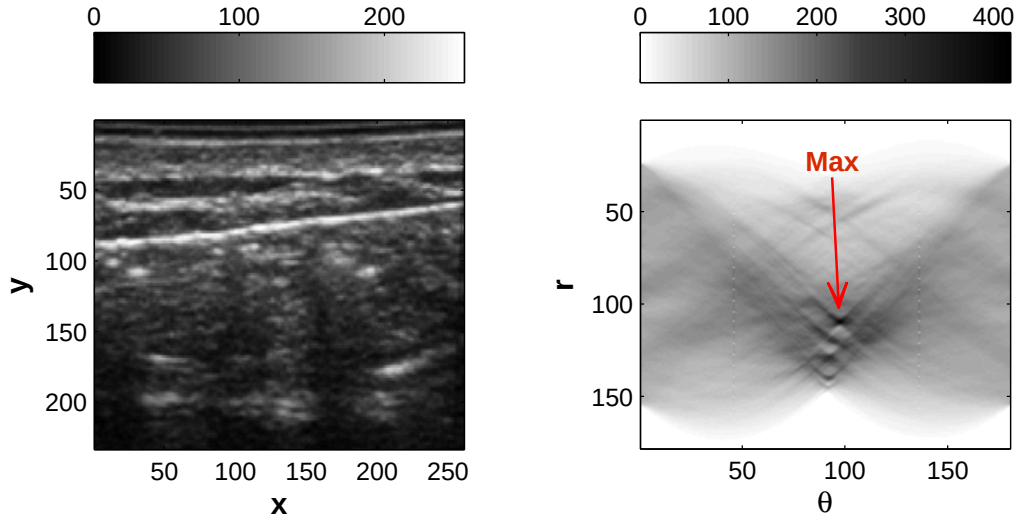


Figure 2.1: *Hough transform using polar coordinates (2.3): (a) the input US image with a linear tool, (b) the output Hough space with parameters* (θ, r).

circle, ellipse. The computational complexity is increased exponentially with the number of dimensions of the Hough space. The HT has been extended for curved object detection [77] (details in Section 2.4) and specifically for thin needle detection in US images.

2.3 Modifications of the Hough transform

2.3.1 The 3D Hough transform

The HT is generalized also for 3D line detection [106, 108]. The line in 3D is defined by at least 4 parameters [82]. The point $\mathbf{p}$ and the vector $\mathbf{v}$ defining the line in 3D is computed from spherical coordinates:

$$\text{Line}_{\phi,\rho,\theta,r} : [x(t), y(t), z(t)] = \mathbf{p}_{\theta,r} + t\mathbf{v}_{\phi,\rho} \tag{2.6}$$

where point $\mathbf{p}_{\theta,r}$ is an intersection of line with base plane (intercept point with the plane xy). The spherical coordinates for $\mathbf{p}_{\theta,r}$ and the azimuth and the elevation for

$\mathbf{v}_{\phi,\rho}$ are

$$\mathbf{p}_{\theta,r} = [r\cos(\theta), r\sin(\theta), 0]\,, \tag{2.7}$$

$$\mathbf{v}_{\phi,\rho} = [\cos(\phi)\cos(\rho), \sin(\phi)\cos(\rho), \sin(\rho)] \tag{2.8}$$

Line detection is done similarly to the 2D case but the 4D accumulator is much bigger.

Bhattacharya et al. [13] decompose the finding of line in 3D into: (i) finding the slope of 3D line in 2D parameter space (ϕ, ρ), and (ii) finding its intercept $\mathbf{p}_{\theta,\, r}$ of a 3D line also in 2D parameter space (so called a (2+2)D Hough space). The experimental results on range images of 3D objects were demonstrated and more moderate costs have been achieved compared to the standard 3D Hough transform.

2.3.2 Randomized Hough transform

The *randomized Hough transform* (RHT) has been proposed by Xu and Oja [100, 102]. The RHT maps a pair of points $(x_1, y_1), (x_2, y_2) \in \mathbb{R}^2$, which define a line, into a single point (a, b) in the parameter space.

$$2 \text{ Points} : \{(x_1, y_1), (x_2, y_2)\} \rightarrow (a, b) \tag{2.9}$$

It is in contrary to the standard HT which maps 1 point (x, y) to a curve in a parameter space. The RHT is more time-efficient than the standard HT.

The RHT (Algorithm 2) iteratively samples random pairs of points in a thresholded image, and increments corresponding bins in the accumulator. The probability distribution of line parameters is represented by a uniform grid accumulator or by an other approximation method using a probability distribution [101]. The solution candidates are found as local maxima in the accumulator.

The RHT has been generalized to the 3D line detection by [108] and used for needle detection in US image. The improved Quick RHT (QRHT) uses a coarse-to-fine strategy to improve the speed [81]. The application of the RHT and the QRHT for tool detection in US images are further discussed in Section 2.4.

2.3.3 Probabilistic Hough transform

In the *Probabilistic Hough transform* (PHT) [65], a randomly selected subset of points is used as input for the standard Hough transform. The time complexity

Input:
$f(x, y)$ — input image.
N — the number of iterations.
Output:
$\{(\theta_i, r_i) \mid i \geq 0\}$ – set of detected lines.
$\mathrm{A}(\theta, \mathrm{r}) \leftarrow 0$; $j \leftarrow 1$;
while $j \leq N$ **do**
  Select randomly a pair of points $(x_1, y_1), (x_2, y_2) \in \mathbb{R}^2$ s.t. $f(x_1, y_1), f(x_2, y_2) \geq T$;
  Compute parameters (θ, r) from $(x_1, y_1), (x_2, y_2)$;
  Increment accumulator $A(\theta, r) \leftarrow A(\theta, r) + 1$;
end
Find local maxima $\{(\theta_i, r_i)\}$ in accumulator $A(\theta, r)$;

Algorithm 2: *The RHT procedure for line detection in an image $f(x, y)$.*

of the PHT is reduced compared to the standard HT at the expense of increased variance. Comparison of the HT methods (RHT, PHT, HT) [67] shows that the RHT is the fastest method for line detection. Several modifications of the RHT are proposed in [67] as well. The performance evaluation of the HT, the PHT, and the RHT is presented in [66]. It is shown that the RHT is better suited for high quality low noise edge images, while the PHT is more successful for noisy low quality images. The experiments were done on synthetic images of points belonging to a line and randomly distributed background points.

2.4 Tool localization with the HT

Okazawa et al. [77] provides an overview of previous work in tool localization in US images. Ding et al. [27, 28] proposed to use 2D projections of 3D US volume for faster detection of a tool. Initially, a region of interest (ROI) which contains the tool is selected. To avoid the computational burden of an exhaustive 2D search for the needle direction, a faster 1D search procedure is proposed. First, a plane which contains the needle direction is determined by finding the line in the initial projection. Subsequently, an adaptive 1D search technique is used to adjust the projection direction iteratively until the projected needle area is minimized. Wei et al. [99] uses the method for 3D guidance for robot-aided prostate brachytherapy. Experimental testing showed that the algorithm finds the 3D needle orientation within 0.54 degree for a chicken tissue phantom, and 0.58 degree for agar phantoms, over a ±15 degrees insertion orientation.

The application of the 3D RHT to a localization of a needle in US images was proposed [106]. The RHT discretizes the parameter space and randomly samples pairs of 3D voxels in thresholded volume (Section 2.3.2). Standard accumulator (uniformly discretized over parameter space) is used. Reported average time of localization is 1 second [107, 108]. Qiu et al. introduced a quick randomized 3D Hough transform [81] which reduces the computational effort by doing the RHT only on coarse resolution images, and subsequently refines the solution locally. The reported angular deviation of the estimated position is less than 1 degree, position deviation less than 1 mm, and the computational time is less than 1 second. All experiments were done in a water tank only.

Okazawa et al. [77] proposes two methods for detection of a tool in 2D US images, both based on the HT. In the first method, a rough approximation of the axis is found. A boundary detection is performed in the direction perpendicular to the axis. The final needle axis is found by the HT. The second method uses a field of possible trajectories of an inserted needle which is assumed to be an arc with constant curvature. The field is used to transform the image with needle so that the needle is straight after the transformation. The proposed method has been tested on US images of a phantom and on photographic images.

Salcudean et al. [83] developed a robotic system for needle guidance for prostate brachytherapy. The reported translational repeatability was 0.06 mm, and the rotational repeatability was within 0.05 degree. An extended needle insertion study using US-based 2D motion tracking was presented in [24].

Aboofazeli et al. [2] propose a scheme which also detects curved needles in 3D US images. A 2D projection is done first similarly to the method of Ding et al. [29]. A 2D parallel projection along specified direction is done using a volume rendering technique (VRT). The tool is segmented in a projected 2D image and then projected back to 3D volume as a surface. The 3D location of the tool is found in the surface. The curved shape is modelled by approximating the tool by few short straight segments.

A modified Radon transform for instrument localization has been proposed by Novotny et al. [76]. They divided the volume into smaller spherical regions and detected the axis in parallel in each of them using a fast method implemented on a graphics processing unit (GPU). They demonstrated a real-time system for tracking of tubular instruments of diameter 5 mm (enhanced with passive markers [90])

in a cardiac sequence with accuracy 0.2 mm [91].

A method based on the generalized HT represents the tool as a Bézier curve of order three which allows the curved shape to be modelled [73]. It increases the number of parameters of the tool and therefore also the time complexity of solution. The speed of the method has been optimized by a parallel implementation on the GPU. The method is demonstrated on 3D US images of brachytherapy needle (of diameter 0.7 mm) in a gelatin agar phantom.

2.5 Parallel Integral Projection

The Parallel Integral Projection (PIP) [P2, 9] is a transform which integrates the volume over parallel lines given a spatial angle (α, β). The integration lines are summed in a projection plane which is perpendicular to lines. The PIP transform can be used for finding the straight tool axis by finding the maximum value of the PIP over varying orientation. When the projection is done parallel to the tool axis, all tool voxels contribute to a single point in the projection plane, and the PIP transform is maximized. An efficient method for maximization of the PIP transform, and thus also finding the angle (α, β) has been proposed [P2, 9]. The details about the PIP-based tool localization and its improvements are in Chapter 3.

2.6 Model fitting methods

A model fitting approach estimates the tool defined by a set of parameters. A method based on the RANSAC procedure [37] is used to solve the problem of a curvilinear object localization in 3D US image. Tao et al. [93] proposed a method based on the RANSAC which was applied to range data from an underwater acoustic camera. He approximates the sought object by superquadrics and with help of the RANSAC, the outliers are filtered out.

Barva et al. [12] developed the idea of model fitting for tool localization in US images. The input 3D US image is thresholded to reduce the amount of data. The location of the tool (defined as a polynomial curve) is estimated using the RANSAC procedure. The precise location of the tool is found by subsequent local optimization of the model. Finally, the tip of the tool is localized by the method described in Section 2.7. Barva proposed in three shape and appearance models of the tool in US images which are summarized in his PhD thesis [9]. We have developed his method

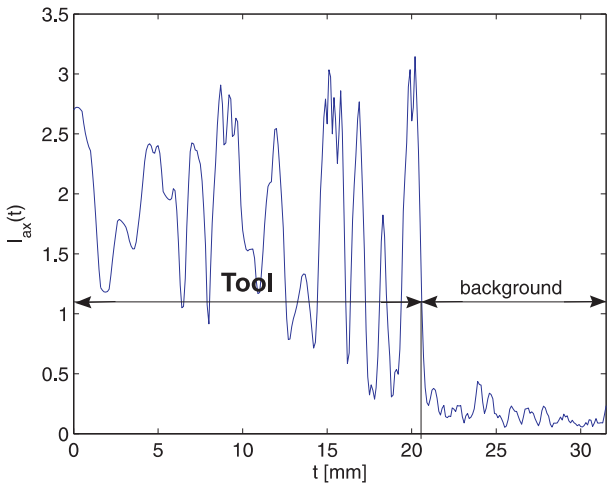


Figure 2.2: *Intensity profile along the estimated axis $a(t)$ is used to identify the part of the axis that passes through the tool.*

further and created new tool models. We give a detailed description of the method in Chapter 4.

2.7 Tip localization

Once the tool axis is identified, we determine the coordinates of tool endpoint. In most cases, the tool is not entirely comprised in the field of view since only a small part of tissue is scanned by the scanning device. Therefore, we only localize one tool endpoint (referred to as a tool tip) that is located inside the field of view.

We use the method [P2, 9] for the tool tip localization in methods proposed in Chapters 3 and 4. We describe it here in more detail. The tool axis is modelled as a parametric line or curve in 3D (Chapter 4, Section 4.1.2). Let $a(t)$ be the tool axis: $a : \mathbb{R} \to \mathbb{R}^3$ with a parameter $t \in \mathbb{R}$. Let $\mathcal{B}(t)$ be the voxel intensity along the estimated axis $a(t)$:

$$\mathcal{B}(t) = \mathcal{I}(a(t)); \qquad \forall t \in \mathbb{R} : a(t) \in \Omega \tag{2.10}$$

Figure 2.2 shows an example of an intensity profile along the estimated axis. While tracing the values of $\mathcal{B}(t)$ for increasing t, let t^* be the first value where $\mathcal{B}(t)$ decreases under a pre-determined threshold T. The coordinates of the tool tip are given by $a(t^*)$.

The threshold value T is determined from two a priori estimated probability

distributions: probability of the tool voxel $p(\mathrm{tl}\,|\,I)$, resp. background voxel $p(\mathrm{tl}\,|\,I)$ given the voxel intensity I. Parameter T is such that $p(\mathrm{tl}\,|\,T) = p(\mathrm{bg}\,|\,T)$. To estimate these distributions, voxels were classified as tool or background voxels in a dataset with known tool position.

The breaks smaller than b_{max} are skipped during the tracing of $\mathcal{B}(t)$. The maximum length of a break b_{max} is determined using a histogram of break lengths, as the 95% quantile.

Chapter 3

Improving the Parallel Integral Projection

The *Parallel Integral Projection* (PIP) was used by Barva et. al [P2, 11] for a straight tool localization in 3D images (Sections 3.1 and 3.2). The PIP is related to the generalized 3D Radon transform [51]. The comparison to other methods [10, P6] and summary of the PIP method is given in Barva's thesis [9].

We propose the multi-resolution improvement of the PIP method (Section 3.3) which was also presented at the *IEEE International Symposium on Biomedical Imaging* (ISBI) 2008 [P3]. We show that the downsampling with use of a maximum decimation filter achieves the better results than the standard downsampling with averaging.

We have published the method based on the PIP transform in the *IEEE Transactions on Ultrasonics, Ferro-electrics, and Frequency Control* (UFFC) [P2]. The author contributed to the project by making final corrections of the paper.

3.1 Definition of the PIP

The PIP is a transform that maps an image function $\mathcal{I} : \mathbb{R}^3 \to \mathbb{R}$ representing the volume data to a function $\mathcal{P}_\mathcal{I} : \mathbb{R}^4 \to \mathbb{R}$ describing its projections as the projection direction determined by two angles (α, β) and a function of the 2D displacement (u, v). More formally, the PIP transformation of $\mathcal{I}(\mathbf{x})$ is defined by an integral along a line passing through the point $Q = [u, v]$ with a direction given by angles α, β:

$$\mathcal{P}_\mathcal{I}(u, v, \alpha, \beta) = \int_{-\infty}^{\infty} \mathcal{I}\left(\mathbf{R}(\alpha, \beta) \cdot (u, v, \tau)^T\right) \mathrm{d}\tau, \tag{3.1}$$

where $\mathbf{R}(\alpha, \beta)$

$$\mathbf{R}(\alpha, \beta) = \begin{pmatrix} \cos\beta & \sin\alpha\sin\beta & -\cos\alpha\sin\beta \\ 0 & \cos\alpha & \sin\alpha \\ \sin\beta & -\sin\alpha\cos\beta & \cos\alpha\cos\beta \end{pmatrix} \tag{3.2}$$

is the rotation matrix representing a rotation around the x-axis by angle α, and around the y-axis by angle β. The range of values $(u, v, 0)$ covers the entire volume in all projections.

The PIP transform is used to identify the axis of a thin tool in a 3D image assuming the intensity of tool is greater than the intensity of background. As the tool diameter tends to zero, the location of the PIP maximum

$$(u_{max}, v_{max}, \alpha_{max}, \beta_{max}) = \arg\max \mathcal{P}_{\mathcal{I}}(u, v, \alpha, \beta) \tag{3.3}$$

approaches the axis of tool with a parametric equation

$$a(t) = \mathbf{R}(\alpha_{max}, \beta_{max}) \cdot (u_{max}, v_{max}, t)^T ; \ \forall t \in \mathbb{R}. \tag{3.4}$$

The integral $\mathcal{P}_{\mathcal{I}}$ is evaluated numerically. The integration line is sampled with a sampling step corresponding to the axial resolution of an ultrasound system.

We shall optimize the PIP transformation (3.3) of a 3D image on a discrete grid. The discretization step Δ for parameters (α, β) and the discretization step Γ for (u, v) must be sufficiently fine in order not to miss the tool. It is necessary that at least one integration line passes through the tool.

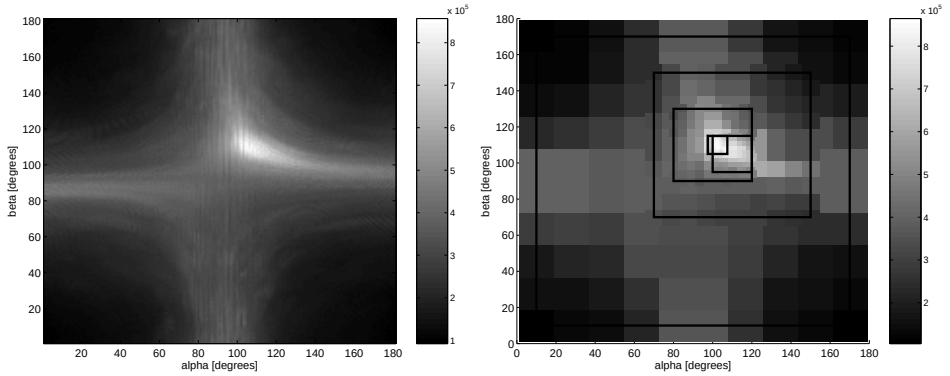
3.2 Optimization of the PIP

We decompose the maximization of $\mathcal{P}_{\mathcal{I}}$ (3.3) to an inner maximization with respect to (u, v) and an outer maximization with respect to (α, β). Let us define a function

$$\mathcal{A}(\alpha, \beta) = \max_{u,v} \mathcal{P}_{\mathcal{I}}(u, v, \alpha, \beta), \tag{3.5}$$

that is referred to as the angle function. We shall now find values $\alpha_{max}, \beta_{max}$ that maximize the angle function $\mathcal{A}(\alpha, \beta)$,

$$(\alpha_{max}, \beta_{max}) = \arg\max_{\alpha,\beta} \mathcal{A}(\alpha, \beta). \tag{3.6}$$



(a) (b)

Figure 3.1: *Maximization of the angle function $\mathcal{A}(\alpha, \beta)$. (a) Exhaustive search with discretization steps evaluates $\mathcal{A}(\alpha, \beta)$ at each grid point of a square grid $\langle 0, 180° \rangle \times \langle 0, 180° \rangle$ uniformly sampled with discretization steps $\Delta = 1°$. (b) Five-level hierarchical mesh-grid search with initial steps $\Delta_1 = 16°$ and final discretization steps $\Delta_5 = 1°$. The size of a search region and discretization steps Δ are decreased by factor of 2 at each level. Black rectangles in the figure delineate the region of search on each level.*

Two approaches to the maximization have been tested: (i) exhaustive search and (ii) hierarchical mesh-grid search. [14] Both approaches do not require an initial solution and they do not need derivates. We decided to use the hierarchical mesh-grid method since its time complexity is lower and it also avoids local maxima to some extent.

Exhaustive search. The interval $(\alpha, \beta) = \langle 0°, 180° \rangle \times \langle 0°, 360° \rangle$ covers all 3D rotations. There are two optimal rotations $(\alpha_{max}, \beta_{max})$ of the angle function $\mathcal{A}(\alpha, \beta)$ (3.6) in this interval which correspond to the projection planes with the opposite orientation. Therefore, it suffices to maximize the angle function in the interval $\langle 0°, 180° \rangle \times \langle 0°, 180° \rangle$. The grid is uniformly sampled with discretization steps Δ, Γ and the angle function is evaluated at each grid point (Figure 3.1a).

The total number of the evaluations of angle function $\mathcal{A}(\alpha, \beta)$ is equal to $\left(\frac{180°}{\Delta}\right)^2$, where Δ is the sampling step of angle space (α, β).

Hierarchical mesh-grid search. The main drawback of the exhaustive search is its computational complexity. We propose to use the hierarchical mesh-grid search method [50] to alleviate it. On the ith level, $\mathcal{A}(\alpha, \beta)$ is evaluated on a rectangular grid of points that are uniformly sampled (Figure 3.1b). Generally, the angle function is evaluated on a grid $\langle \alpha_{max}^{i-1} - \frac{180°}{2^i}, \alpha_{max}^{i-1} + \frac{180°}{2^i} \rangle \times \langle \beta_{max}^{i-1} - \frac{180°}{2^i}, \beta_{max}^{i-1} + \frac{180°}{2^i} \rangle$ with

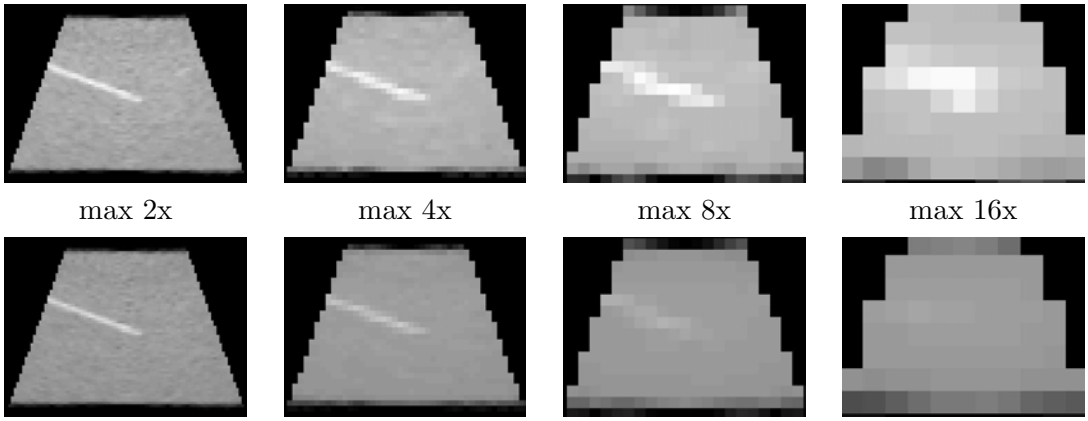


Figure 3.2: *The 2D image of a slice with needle selected from 3D volume, and multiple downsampled images using max and avg functions. The original resolution was* $53 \times 71 \times 262$ *voxels, for each coarser level the resolution has been divided by two. The tool of radius* 0.3 mm *is blurred out in case of the avg function in coarse resolutions, while for the max function the contrast between the tool and the background stays relatively good.*

discretization steps $\Delta^i = \frac{\Delta^1}{2^{i-1}}$. The algorithm continues until the step Δ^i is inferior to a predefined threshold value Δ^{final} that controls accuracy of axis localization.

The asymptotic complexity of the method is logarithmical compared to exhaustive search. The details and the discussion about proposed algorithms can be found in [P2].

3.3 Multi-resolution PIP

As the basic PIP method [P2, 9] is rather slow, we want to improve its speed. We propose to use downsampled 3D images for tool localization with a special decimation filter. We show that more reliable results on downsampled images are obtained by using a maximum function than with a standard averaging of neighbouring elements.

A discrete function $\mathcal{I}_{smp} : \mathbb{N}^3 \to \mathbb{R}$ represents an image $\mathcal{I}$ at the pixel grid. We define the downsampled image $\mathcal{I}_{down}$ of image $\mathcal{I}_{smp}$ by a factor $M_x, M_y, N_z \in \mathbb{N}$. The intensity of voxel (x, y, z) in downsampled image is computed as a function of intensities of neighboring voxels of size $M_x \times M_y \times M_z$:

$$\mathcal{I}^f_{down}(x, y, z) = f \left\{ \begin{array}{c} \mathcal{I}_{smp}(xM_x + i, yM_y + j, zM_z + k), \\ 0 \le i < M_x,\ 0 \le j < M_y,\ 0 \le k < M_z \end{array} \right\}, \tag{3.7}$$

Input: 3D image $\mathcal{I}_{smp}$ with tool, constants $\Delta_{init}, \Delta_{final}, \Gamma_{init}, \Gamma_{final}, K_{init}, K_{final}$
Result: Tool axis: $(\alpha_{max}, \beta_{max}, u_{max}, v_{max})$
Create multiple resolutions $\mathcal{I}_1 ... \mathcal{I}_n$;
$k \leftarrow K_{init}$, $\Delta \leftarrow \Delta_{init}$, $\Gamma \leftarrow \Gamma_{init}$, $R \leftarrow 90°$;
$A \leftarrow (0°, \Delta, 2\Delta, ..., 180°) \times (0°, \Delta, 2\Delta, ..., 180°)$;
while $\Delta > \Delta_{final}$ or $\Gamma > \Gamma_{final}$ or $k > K_{final}$ **do**
    $(\alpha_{max}, \beta_{max}, u_{max}, v_{max}) \leftarrow \arg\max_{\alpha,\beta} \max_{u,v}$ $\mathcal{P}_{\mathcal{I}_k}(u, v, \alpha, \beta)$ where $(\alpha, \beta) \in A$ and $(u, v) \in (u_1, u_1 + \Gamma, ..., u_2) \times (v_1, v_1 + \Gamma, ..., v_2)$;
    $k \leftarrow \max(k - 1, K_{final})$;
    $\Delta \leftarrow \max(\Delta/2, \Delta_{final})$;
    $\Gamma \leftarrow \max(\Gamma/2, \Gamma_{final})$, $R \leftarrow R/2$;
    $A \leftarrow (\alpha_{max} - R, \alpha_{max} - R + \Delta, ..., \alpha_{max} + R) \times (\beta_{max} - R, \beta_{max} - R + \Delta, ..., \beta_{max} + R)$;

Algorithm 3: *Hierarchical mesh-grid search of the MR-PIP. Constants u_1, u_2, v_1, v_2 are boundaries of the projected volume, A contains the set of angles for evaluation of $\mathcal{P}_{\mathcal{I}_k}$, R is the size of the current interval of angles. $\Delta_{init}, \Delta_{final}$ and $\Gamma_{init}, \Gamma_{final}$ are initial and final discretization steps for Δ and Γ respectively. K_{init} and K_{final} is initial and final level of resolution respectively.*

The f should preserve well the contrast between the tool and the background. We examine two functions: the standard averaging ($f = avg$) and the maximum function ($f = max$) of a set of values which is a quantile with the top rank. We discuss some other options in Section 3.5. We set $M_x = M_y = M_z = 2$. A multi-resolution pyramid was constructed by repeated downsampling (Figure 3.2):

$$\mathcal{I}_1 = \mathcal{I}_{smp}, \mathcal{I}_2 = \mathcal{I}_{down}^{max}(\mathcal{I}_1), \ldots \mathcal{I}_n = \mathcal{I}_{down}^{max}(\mathcal{I}_{n-1}). \tag{3.8}$$

Algorithm. We use a hierarchical mesh-grid search (Section 3.2) [P2, 9, 14] for finding the maximum $(\alpha_{max}, \beta_{max}, u_{max}, v_{max})$ of the PIP transform with respect to (3.6). We modify the maximization by using a multi-resolution pyramid (Algorithm 3). The discretization step Δ for (α, β) and the discretization step Γ for (u, v) is iteratively decreased as the resolution level $\mathcal{I}_k$ is refined. The multi-resolution (MR) method is faster than the PIP while preserving the high accuracy.

By stopping the iterative algorithm early we further accelerate the algorithm. We stop on a coarse resolution $\mathcal{I}_{K_{final}}$ and set larger discretization steps Δ_{final} and Γ_{final}. It usually allows to use less iterations but decreases the accuracy. We call this method the fast multi-resolution Parallel Integral Projection (Fast MR-PIP).

Let us analyze the time complexity of the MR-PIP. The number of iterations of while loop in Algorithm 3 is at most $N = \max\{N_1, N_2, N_3\}$ where $N_1 = \lceil \log_2 \frac{2\Delta_{init}}{\Delta_{final}} \rceil$, $N_2 = \lceil \log_2 \frac{2\Gamma_{init}}{\Gamma_{final}} \rceil$ and $N_3 = K_{init} - K_{final} + 1$. The number of evaluations of $\mathcal{P}_{\mathcal{I}_k}$ in each iteration is dependent on maximization of the term in line 5. There are $T_1 = (\frac{180^\circ}{\Delta_{init}})^2$ evaluations of outer part maximizing over (α, β) and $T_2 = \sum_{i=\Gamma_{init}}^{\Gamma_{final}} (\frac{size}{\Gamma_i})^2 = \sqrt[4]{3} \cdot (\frac{size}{\Gamma_{final}})^2$ evaluations of inner part maximizing over (u, v). The total number of evaluations of $\mathcal{P}_{\mathcal{I}}(u, v, \alpha, \beta)$ is $N \cdot T_1 \cdot T_2$ and it depends on the size of volume and discretization steps $\Delta_{init}, \Delta_{final}, \Gamma_{init}, \Gamma_{final}$.

3.4 Results

We show that the MR-PIP method is as accurate as the basic PIP method while being faster. The method was implemented in MATLAB. We did all tests on a Gentoo Linux computer with a 64-bit Intel Core 2 processor at 2400 MHz.

Two measures are used to quantify the accuracy [P2, 9]. The tip localization accuracy is

$$\varepsilon_{tip} = \|T - \hat{T}\|, \tag{3.9}$$

where T is the true tool tip, $\hat{T}$ is the estimated tip and $\|\cdot\|$ is the Euclidean distance. Axis localization accuracy is given by

$$\varepsilon_{axis} = \max\left\{\|E - Q_1\|, \|T - Q_2\|\right\}, \tag{3.10}$$

where E is the intercept point, Q_1 and Q_2 are the orthogonal projections of E and T on the estimated axis with respect to the true axis (Figure 3.3). We consider a result as successful when the axis accuracy is better than 10 mm.

For the evaluation of image quality we define the signal-to-noise ratio:

$$\mathrm{SNR} = 10 \log \frac{E[x_{el}^2]}{E[x_{bg}^2]} \ [\mathrm{dB}] \tag{3.11}$$

where x_{el} are voxels with distance from the axis less than a tool radius and the remaining voxels are considered as background x_{bg}.

3.4.1 Evaluation of parameters influence

The experiments were done on 28 simulated datasets of size $53 \times 71 \times 307$ voxels with varying tool translation and rotation. Simulated data were generated by using the

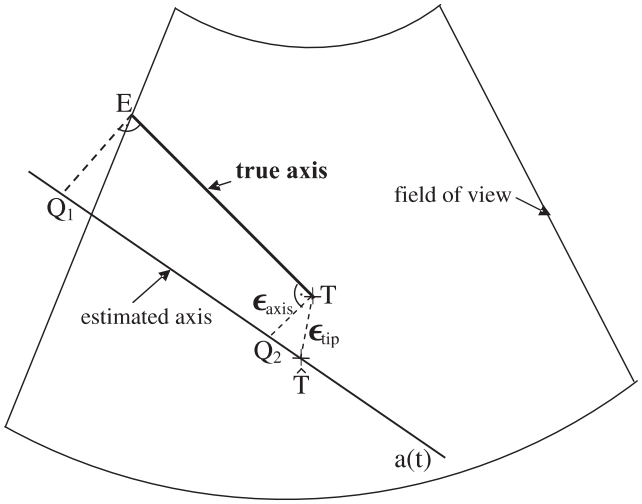


Figure 3.3: *Illustration of the axis and endpoint accuracy evaluation. The ground-truth tool position is determined by an intercept point E and a tool tip T. It is compared with an axis $\mathbf{a}(t)$ and a tip $\hat{T}$ estimated by the proposed method.*

US simulator FIELD II [59]. The parameters of simulation were set to imitate the US scanner Voluson 530D. The discretization parameters were fixed to $\Delta_{init} = 32°$, $\Delta_{final} = 1°$, $\Gamma_{init} = 0.4 \cdot 2^{levels-1}$ mm, $\Gamma_{final} = 0.4$ mm where the parameter *levels* is the number of resolutions.

We compared the *max* and *avg* decimation functions for downsampling. The SNR values of simulated data are in Figure 3.4. Downsampling with the *max* function seems to preserve the tool shape and also the SNR is better than the *avg* function. Figure 3.5a shows the success rate for the MR-PIP on synthetic data. The success rate is steadily close to 100% for the *max* function, and decreasing to 0% for the *avg*. We decided to use the *max* function for downsampling in the rest of experiments.

We varied the number of resolution levels from 1 to 5 and measured the time (Figure 3.5b). As expected, the time has been significantly reduced from the 123 seconds mean time for the basic single resolution PIP method (Table 3.1, row 1). The best mean time was 46 seconds for resolution 3 (Table 3.1, row 2) with the accuracy 0.4 mm which is satisfactory (Figure 3.5c, 3.5d).

We have also evaluated the trade-off between the speed and the accuracy for the Fast MR-PIP method. The results for various values of the final discretization step are shown in Figure 3.6. The error of tool localization was larger for one dataset relatively to the rest of group (three points to the right in Figure 3.6) but still

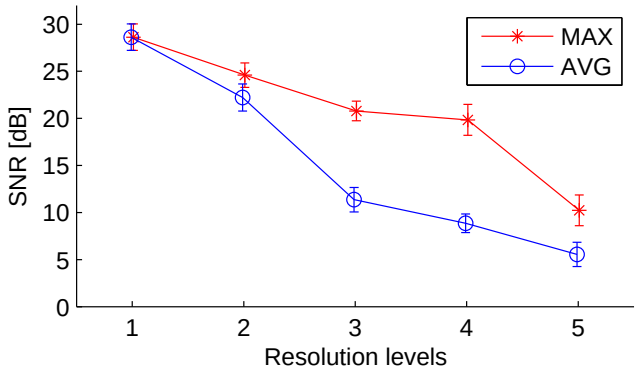


Figure 3.4: *The comparison of SNR of simulated data on various resolution levels for max and avg filtering function.*

considered as a successful run. We reached the mean time 7.8 seconds with axis accuracy around 2 mm for Γ_{final} = 1.6 mm (Table 3.1, row 3).

3.4.2 Experiments on real data

We acquired 8 datasets of a cryogel tissue mimicking phantom with a thin tungsten electrode of 250 μm in diameter and length 20 mm using an US scanner Voluson 530D [69]. The datasets are sector volumes of 40°×40°×depth 6.2 cm with a resolution of $53 \times 71 \times 310$ voxels. The success rate for experiments on this data was 100% in all cases and measured time and accuracy can be found in Table 3.1 (row 4 for the MR-PIP and row 5 for the Fast MR-PIP).

A dataset of breast biopsy was acquired by a 3D US scanner GE Voluson E8 with the 12 MHz probe. The biopsy needle was 1.092 mm in diameter. The geometry of the volume was $30° \times 38$ mm and 19 mm depth with resolution $207 \times 383 \times 208$ voxels. The localization was successful with a discretization step Γ_{final} = 0.3 mm and the result corresponds to the visual identification. The time and achieved accuracy are in Table 3.1 (row 6 for the MR-PIP and row 7 for the Fast MR-PIP).

3.5 Conclusions

We have presented a multi-resolution PIP method for a straight tool axis localization in 3D US data which is much faster than the basic PIP and yet has the same accuracy. Further speed-up is achieved by the Fast multi-resolution PIP based on an early stopping of the hierarchical search algorithm at the expense of worse

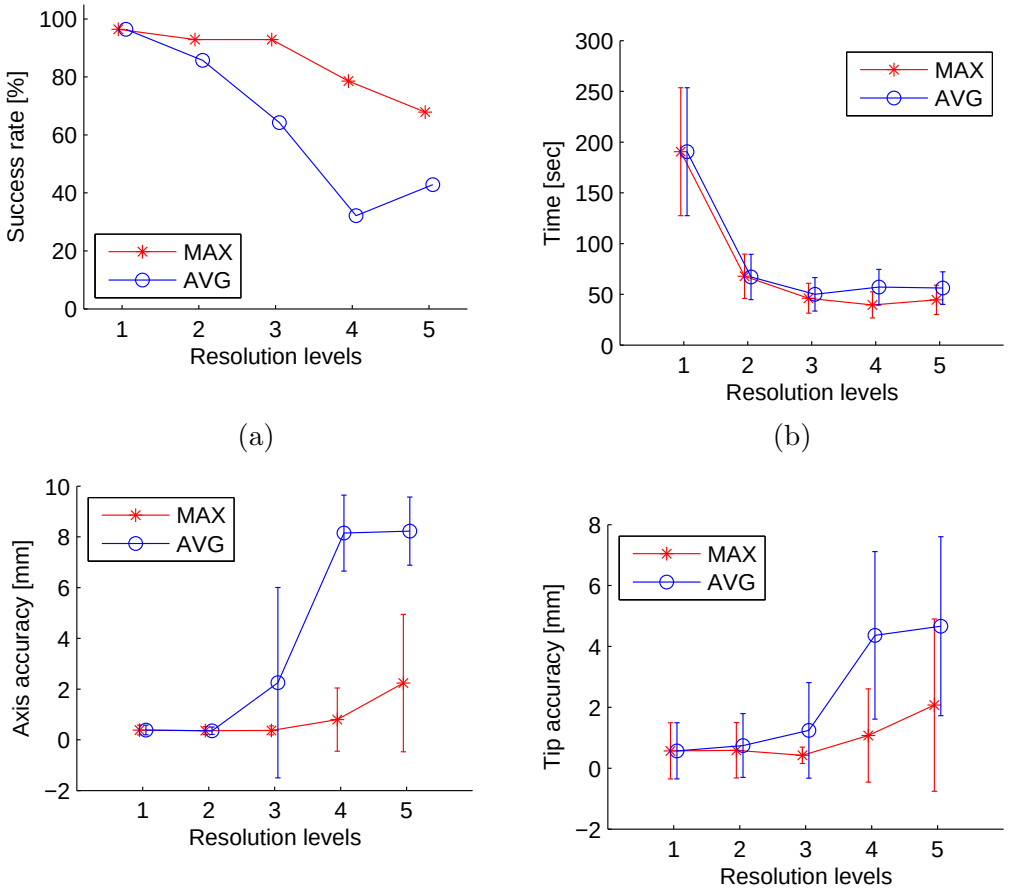


Figure 3.5: *Results from the MR-PIP on simulated data. The original resolution of the data was* $53 \times 71 \times 307$ *voxels. The basic PIP algorithm is equivalent to resolution level* 1*. The success rate considers good results with axis accuracy better than* 10 mm*. The figures show means as points and standard deviations as vertical bars.*

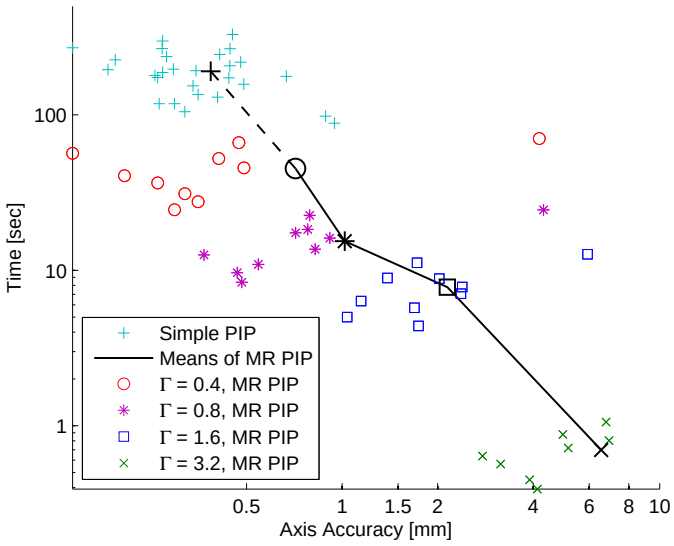


Figure 3.6: *The figures show the dependence of time and accuracy on final discretization* Γ_{final} *for the Fast MR-PIP and the basic PIP without any multi-resolution on simulated data.*

accuracy. Rewriting the method in a compiled language or implementing the method on the GPU would also accelerate it substantially.

The decimation based on averaging or the Gaussian filtering is commonly used. We propose downsampling with the *max* decimation function which preserves tool voxels better than averaging. The other functions similar to maximum could be considered, e.g. a quantile with a high rank which preserves the high intensity tool and could be more robust. The decimation can be used also for other applications requiring fast detection of thin lines, e.g. vessel segmentation in a 3D image.

The PIP methods are not suitable for localization of curved tools. They have also problems with tools of a low contrast to the background, i.e., when there are other biological structures with high intensity. We will propose different localization methods in the following chapters.

Data / Method	Time [sec]	Axis ac. [mm]	tip ac. [mm]
simulated / 1	123 ± 5.8	0.327 ± 0.180	1.732 ± 5.132
simulated / 2	46.1 ± 15	0.371 ± 0.178	0.422 ± 0.268
simulated / 3	7.8 ± 2.6	2.143 ± 1.369	13.89 ± 6.67
phantom / 2	62.5 ± 26	0.443 ± 0.206	0.508 ± 0.175
phantom / 3	7.3 ± 2.1	1.421 ± 0.429	9.335 ± 7.747
br. biopsy / 2	61	0.108	0.569
br. biopsy / 3	5.5	3.270	3.302

Table 3.1: *Results on various data: simulated data (28 datasets), cryogel phantom (8 datasets), breast biopsy (1 dataset). Methods used is the table: 1 - the PIP on single full resolution, 2 - the MR-PIP with 3 resolutions, 3 - the Fast MR-PIP. The mean and standard deviations of elapsed time, axis accuracy and tip accuracy are reported.*

Chapter 4

Model fitting using RANSAC

The proposed localization algorithm is based on a model fitting approach described previously by Barva [9]. It is able to localize the deformed tool because the axis is represented as a polynomial curve. We propose two new models of the tool in US image (Section 4.1.3). Parameters of model are estimated using the RANSAC procedure (Section 4.1.4) and refined using the local optimization (Section 4.1.5). We add experiments on real data of a phantom and breast tissue (Section 4.2). The work described in this chapter has been published in the *IEEE Transactions on Biomedical Engineering* [P1].

4.1 Method

Given a 3D ultrasound image (Fig. 4.1), our method is able to find the position and orientation of thin elongated objects such as electrodes or needles. With respect to the projection-based methods mentioned above (the HT, the RHT or the PIP), the presented method is designed to be faster, more robust to the presence of other high-intensity structures, and allows more generally shaped models.

The algorithm is based on the following two assumptions:

Assumption 1: The intensity of the tool voxels is higher than the surrounding tissue.

Assumption 2: The shape of the tool is a thin, long, and possibly curved cylinder.

The tool might be deformed during insertion and steering. The deformation is caused by lateral forces [26]. Bending is typical for thin electrodes (diameter around

0.3 mm). Biopsy needles are thicker (diameter around 1 mm) and therefore remain straight.

Our goal is to localize the axis and the tip of the tool with sub-millimeter accuracy which is sufficient for all intended applications. The task is challenging for several reasons: the diameter of the tool can be as small as the physical resolution of the ultrasound. The images contain a large amount of speckle noise and some parts of the tissue have a level of intensity similar to the tool. The proposed algorithm consists of four steps:

1. Thresholding — the considered set of voxels is reduced by thresholding using Assumption 1 (Section 4.1.1).

2. Axis localization — an approximate position of the tool axis is estimated by a robust randomized search procedure RANSAC (Section 4.1.4).

3. Local optimization — a more accurate solution is found by using local optimization (Section 4.1.5).

4. Tip localization — the endpoint of the tool is identified along the tool axis (as described in Section 2.7).

Steps 2 and 3 are based on a model describing the tool shape and intensity in the image (Sections 4.1.2 and 4.1.3).

4.1.1 Thresholding

A set of voxels with coordinates $\mathcal{X} \subseteq \mathbb{R}^3$ and intensities $I(\mathcal{X}) \subseteq \mathbb{R}$ is split by thresholding into two disjoint sets: $\mathcal{X}_t$ (tool voxels) and $\mathcal{X}_b$ (background voxels)

$$\begin{aligned} \mathcal{X}_t &= \{\mathbf{x} \in \mathcal{X} : I(\mathbf{x}) \geq T_I\}, \\ \mathcal{X}_b &= \mathcal{X} \setminus \mathcal{X}_t, \end{aligned} \tag{4.1}$$

As an example, a thresholded 3D image $\mathcal{X}_t$ from Fig. 4.1 is shown in Fig. 4.2. All subsequent processing uses only the subset $\mathcal{X}_t$ in order to reduce the processing time.

The threshold T_I is found by minimizing voxel classification error, assuming that labelled training data are available. Otherwise, the threshold is chosen empirically. Barva estimates the threshold as the 95% quantile of the input data by fitting a Gamma distribution [9] which seems to work in practice. It is based on the expectation that the proportion of voxels belonging to the tool is less than 5%.

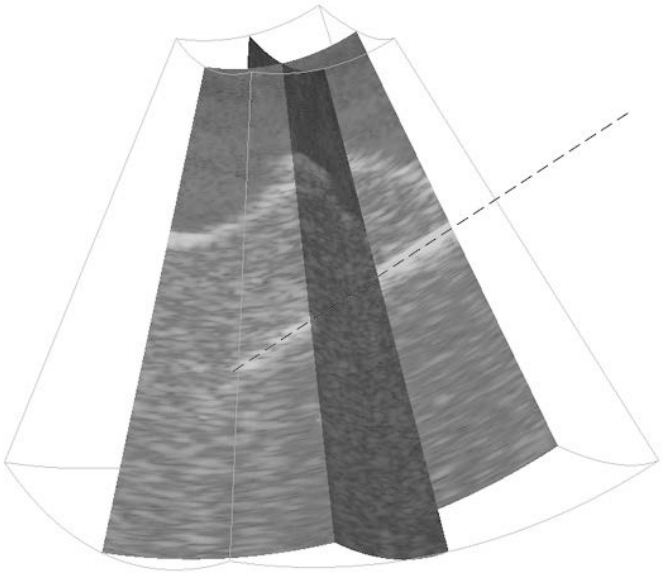


Figure 4.1: *Example of a 3D ultrasound image of the PVA cryogel phantom in water. Inside the phantom there is a tungsten electrode. Two planar sections (one of them passing through the electrode axis) are shown in gray-scale. The diagonal dashed line is the localization result. A thin wire-frame shows the scanned volume boundaries.*

Note that the resulting $\mathcal{X}_t$ also contains some non-tool voxels (outliers) which are filtered out later (Section 4.1.4).

4.1.2 Axis model

The tool axis is represented by a spatial parametric polynomial curve $\mathbf{a}(t;\mathbf{H}) : \mathbb{R} \to \mathbb{R}^3$ of order $n-1$:

$$\mathbf{a}(t;\mathbf{H}) = \underbrace{\begin{pmatrix} h_{11} & \cdots & h_{1n} \\ h_{21} & \cdots & h_{2n} \\ h_{31} & \cdots & h_{3n} \end{pmatrix}}_{\mathbf{H}} \begin{pmatrix} 1 \\ t \\ \vdots \\ t^{n-1} \end{pmatrix}; \quad t \in \mathbb{R} \tag{4.2}$$

We use $n = 2$ to model straight tools, polynomial curves of low order $n = 3$ to model bent tools (C-like shapes) and occasionally higher values of n might be also used (S-like shapes for $n = 4$). The curve is determined by n control points $\mathbf{p}_i \in \mathbb{R}^3$, $i = 1 \ldots n$ through which it is required to pass. First, a principal direction $\mathbf{k}_0$ is determined by fitting a straight line to points $\mathbf{p}_i$, e.g., using a least squares method. Then, we

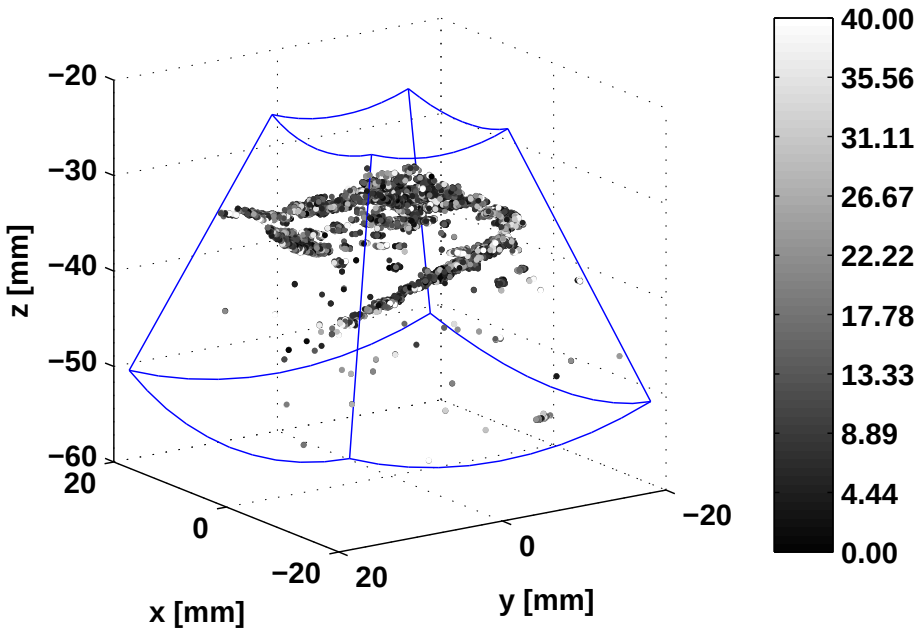


Figure 4.2: *A 3D ultrasound image from Fig. 4.1 after thresholding. Intensity values shown are normalized by the mean of all voxels. A blue wireframe shows the volume boundaries.*

choose the parameters t_i according to a projection onto this line

$$t_i = \frac{(\mathbf{p}_i - \mathbf{p}_1)\mathbf{k}_0}{\|\mathbf{k}_0\|}. \tag{4.3}$$

Finally, the polynomial curve defined by matrix $\mathbf{H}$ is found by solving the following system of $3n$ linear equations:

$$\mathbf{a}(t_i, \mathbf{H}) = \mathbf{p}_i, \ \forall i : 1 \leq i \leq n, \tag{4.4}$$

after substituting $\mathbf{a}(t_i, \mathbf{H})$ from (4.2).

4.1.3 Tool models

Two models (AxShp, IntDstr) are proposed for the tool shape and intensity in 3D ultrasound images to be used in steps 2 and 3 of the algorithm (Section 4.1). Each model consists of a function $q(\mathbf{x}; \mathbf{H}) \in \{1, 0\}$ classifying each voxel $\mathbf{x}$ with intensity $I(\mathbf{x})$ as either a tool ($q = 1$), or a background ($q = 0$); and a cost function $C(\mathcal{X}_{\text{inl}}; \mathbf{H})$ quantifying how well the model parameters $\mathbf{H}$ fit a set of voxel observations $\mathcal{X}_{\text{inl}}$ consistent with the model (inliers).

The tool shape, i.e. curve parameters $\mathbf{H}$, are first estimated roughly by using the function $q(\mathbf{x};\mathbf{H})$ in the RANSAC procedure (Section 4.1.4). The RANSAC maximizes the number of tool voxels (inliers $\mathcal{X}_{\text{inl}}$). Afterwards, the solution is refined by local optimization of the cost function $C(\hat{\mathcal{X}}_{\text{inl}};\mathbf{H})$ on the best set of estimated inliers $\hat{\mathcal{X}}_{\text{inl}}$ (Section 4.1.5).

Model AxShp

The model evaluates only the distances of the points $\mathbf{x} \in \mathcal{X}_t$ to the curve $a(t;\mathbf{H})$. It does not use any a priori information on the intensity values and no training is needed. The classification function $q(\mathbf{x})$ uses a distance $d(\mathbf{x};\mathbf{H})$ of the point $\mathbf{x}$ to the curve $\mathbf{a}(t,\mathbf{H})$

$$q_{\text{AxShp}}(\mathbf{x};\,\mathbf{H}) = \begin{cases} 1, & \text{if } d(\mathbf{x};\mathbf{H}) \le \tau \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

The threshold τ is set as the expected radius of the tool in the image. The model's cost $C(\mathbf{H})$ is the sum of the squared distances to the axis

$$C_{\text{AxShp}}(\mathcal{X}_{\text{inl}};\,\mathbf{H}) = \sum_{\mathbf{x}\in\mathcal{X}_{\text{inl}}} d(\mathbf{x};\mathbf{H})^2. \tag{4.6}$$

The cost function is smooth (unlike $\sum q_{\text{AxShp}}(\mathbf{x};\mathbf{H})$, used in [9]) which is important for the local optimization later.

As the true point-to-curve distance would be prohibitively expensive to calculate, the following approximation is used:

$$\begin{aligned} d(\mathbf{x};\mathbf{H}) &= \|\mathbf{x} - \mathbf{a}(t;\mathbf{H})\|, \\ \text{with} \quad t &= \frac{(\mathbf{x}-\mathbf{p}_1)\mathbf{k}_0}{\|\mathbf{k}_0\|} \end{aligned} \tag{4.7}$$

which is good as long as the curvature of $\mathbf{a}(t;\mathbf{H})$ is small.

Model IntDstr

The model is based on an estimated likelihood $p(d, i \mid c)$ of observing a voxel $\mathbf{x}$ with an intensity $i = I(\mathbf{x})$ at distance d (4.7) from the axis, given its class c (tool 'tl' or background 'bg'). Assuming $p(\text{tl}) = p(\text{bg})$, the classification function q is chosen as

a likelihood comparison test:

$$q_{\mathrm{IntDstr}}(\mathbf{x};\mathbf{H}) = \begin{cases} 1, & \text{if } p(d,i \mid \mathrm{tl}) \geq p(d,i \mid \mathrm{bg}) \\ 0, & \text{otherwise} \end{cases} \tag{4.8}$$

The model's cost function $C(\mathbf{H})$ is a negative logarithmic likelihood of observing the set of inliers $\mathcal{X}_{\mathrm{inl}}$ given a tool position, assuming that voxel observations are independent:

$$C_{\mathrm{IntDstr}}(\mathcal{X}_{\mathrm{inl}};\mathbf{H}) = \sum_{\mathbf{x}\in\mathcal{X}_{\mathrm{inl}}} -\log p(d,i \mid \mathrm{tl}). \tag{4.9}$$

The dependence of $p(d,i \mid c)$ on $\mathbf{H}$ is implicit through d (4.7).

The likelihood $p(d,i \mid \mathrm{tl})$ for tool voxels is decomposed as follows:

$$\begin{aligned} p(d,i \mid \mathrm{tl}) &= p(i \mid d,\mathrm{tl})\, p(d \mid \mathrm{tl}) \\ \text{with} \quad p(d \mid \mathrm{tl}) &= \mathrm{N}^{+}_{0,\sigma}(d) \end{aligned} \tag{4.10}$$

where $\mathrm{N}^{+}_{0,\sigma}(d)$ is the positive part of a normal distribution with zero mean and variance σ (for simplicity) corresponding to the expected radius of the tool in the image. The background intensity $p(d,i \mid \mathrm{bg})$ is assumed to be spatially independent, i.e.,

$$p(d,i \mid \mathrm{bg}) = p(i \mid \mathrm{bg}). \tag{4.11}$$

Both $p(i \mid d,\mathrm{tl})$ and $p(i \mid \mathrm{bg})$ are estimated from a training set of images with a known ground-truth. The tool voxels are first collected into m uniformly sized bins $b_j = \left[j\Delta d;(j+1)\Delta d\right]$ according to a distance d from the tool. The distributions of voxel intensities $p(i \mid \mathrm{bg})$ and $p(i \mid d \in b_j,\mathrm{tl})$ for $j = 0 \ldots m-1$ are modelled as Gamma distributions $\Gamma_{k_{\mathrm{bg}},\theta_{\mathrm{bg}}}$ and Γ_{k_j},θ_j, respectively. The Gamma distribution is sufficiently general to approximate the real distribution well and it was successfully used for ultrasound images in [9, 94]. The parameters $\sigma, k_{\mathrm{bg}}, \theta_{\mathrm{bg}}, k_j$ and θ_j are determined as maximum likelihood estimates [21]. Examples of learned distributions are shown in Fig. 4.3.

4.1.4 RANSAC procedure

The RANSAC (RAndom SAmple Consensus) procedure was introduced by Fischler and Bolles [37] to solve the problem of robust estimation of model parameters given a set of input samples with a large portion of outliers. In our case, the input of the RANSAC procedure is a set of thresholded voxels $\mathcal{X}_t$ and a classification function

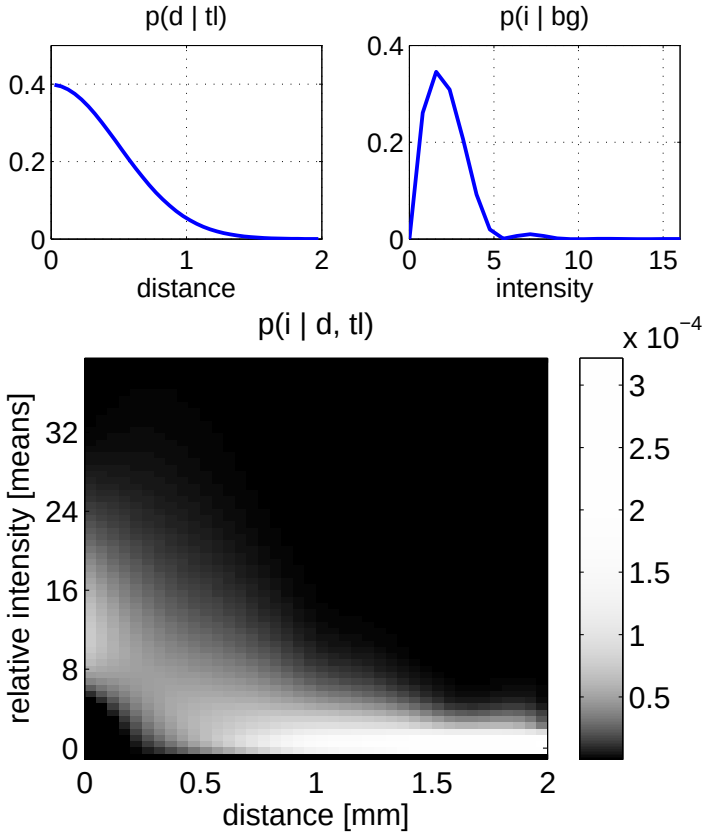


Figure 4.3: *Examples of empirically learned distributions* $p(i|d,\mathrm{tl})$, $p(d|\mathrm{tl})$ *and* $p(i|\mathrm{bg})$ *used in the* IntDstr *model. The distributions were estimated from a training set of nine images of an electrode in the PVA cryogel phantom.*

q_{AxShp} or q_{IntDstr}. The RANSAC outputs are the identified curve parameters $\hat{\mathbf{H}}$ and the corresponding set of points $\hat{\mathcal{X}}_{\mathrm{inl}}$ consistent with the model.

In each RANSAC iteration (Algorithm 4), first a sample consisting of a set $\mathcal{P} = \{\mathbf{p}_i; i = 1 \ldots n\}$ of n distinct points $\mathbf{p}_i$ is randomly selected from $\mathcal{X}_t$ (Step 1 of Algorithm 4). To quickly filter out sets $\mathcal{P}$ leading to excessively curved axes, which would not lead to good solutions anyway, the samples with curvature $\kappa(\mathcal{P}) > \kappa_{max}$ are rejected. The pseudo-curvature $\kappa(\mathcal{P})$ is defined as the maximum orthogonal distance between one of the control points $\mathbf{p}_i$ and a straight line $l(\mathcal{P})$ fitted to all n points $\mathcal{P}$. The threshold κ_{max} is set to maximum expected deformation. The coefficient matrix $\mathbf{H}$ is calculated from $\mathcal{P}$ as described in Section 4.1.2 (Step 2) and a set of inliers is estimated (Step 3):

$$\mathcal{X}_{\mathrm{inl}}(\mathbf{H}) = \{\mathbf{x} \in \mathcal{X}_t \mid q(\mathbf{x}; \mathbf{H}) = 1\}\,. \tag{4.12}$$

Input:
$\mathcal{X}_t$ — thresholded voxels.
q — classification function from AxShp or IntDstr models.
Output:
$\hat{\mathcal{X}}_{\text{inl}}$ – estimated inliers.
$\hat{\mathbf{H}}$ – estimated curve parameters.
$\hat{\mathcal{P}}$ – control points on the estimated curve.
$j \leftarrow 1$; $J \leftarrow J_{\max}$;
while $j \leq J$ **do**
1. Randomly select a subset $\mathcal{P} \subset \mathcal{X}_t$, $|\mathcal{P}| = n$ and compute $\kappa(\mathcal{S}_j)$. If $\kappa(\mathcal{P}) \leq \kappa_{max}$, then continue; otherwise repeat step 1;
2. Compute matrix $\mathbf{H}$ from control points $\mathcal{P}$ (4.2–4.4);
3. Estimate a set of inliers $\mathcal{X}_{\text{inl}}$ (4.12);
4. Update the best-so-far model:
   if $|\mathcal{X}_{inl}| > |\hat{\mathcal{X}}_{inl}|$ **then**
      a) $\hat{\mathbf{H}} := \mathbf{H}$; $\hat{\mathcal{P}} := \mathcal{P}$; $\hat{\mathcal{X}}_{\text{inl}} := \mathcal{X}_{inl}$;
      b) Update the number of iterations J (4.13);
   else $j \leftarrow j + 1$;
end

Algorithm 4: *The RANSAC procedure to robustly estimate parameters of the tool axis from the thresholded points $\mathcal{X}_t$.*

The best curve parameters $\hat{\mathbf{H}}$ found so far based on the number of estimated inliers $|\mathcal{X}_{\text{inl}}|$ (Step 5) are stored together with the corresponding control points $\hat{\mathcal{P}}$ and a set of consistent points $\hat{\mathcal{X}}_{\text{inl}}$.

The number of iterations J to perform is initially set to $J_{\max}$ (typically a few hundred) and it is then adaptively updated (Step 4b) whenever a better model is found [37, 96]

$$J = \frac{\ln(1-\eta)}{\ln(1-\zeta^n)}, \qquad \text{with} \quad \zeta = \frac{|\mathcal{X}_{\text{inl}}|}{|\mathcal{X}_t|} \tag{4.13}$$

where ζ estimates the inlier ratio and η is a user-defined parameter which is a desired probability that the RANSAC succeeds. Success of the RANSAC means that n inliers are selected at least once during J iterations.

4.1.5 Local optimization

The RANSAC procedure gives a robust approximation of the axis position. However, its accuracy is limited, since the model parameters $\hat{\mathbf{H}}$ are computed only from n control points. Therefore a more accurate solution $\mathbf{H}^*$ is found based on the complete estimated set of inliers $\hat{\mathcal{X}}_{\text{inl}}$ by minimizing the cost function $C(\hat{\mathcal{X}}_{\text{inl}}; \mathbf{H})$ (Section 4.1.3).

Instead of optimizing the coefficients $\mathbf{H}$ directly, we optimize the position of the control points $\mathcal{P}$ because it is numerically more stable. Moreover, it is enough to vary the point positions in a direction perpendicular to the axis.

First, the local coordinate system $\mathbf{K}$ is calculated from the set $\hat{\mathcal{X}}_{\text{inl}}$ by PCA. The coordinate system $\mathbf{K}$ consists of principal directions $\mathbf{k}_0$, $\mathbf{k}_1$, $\mathbf{k}_2$, in the decreasing order of corresponding eigenvalue magnitudes ($|\lambda_0| \geq |\lambda_1| \geq |\lambda_2|$)[1]. The position of the control points along the $\mathbf{k}_0$ is not important for the shape of the curve. To reduce the redundancy of the parametrization, the control points $\hat{\mathcal{P}}$ are reparametrized by using a matrix $\mathbf{E}$ with dimensions $2 \times n$

$$\underbrace{\begin{bmatrix}\mathbf{p}_1 & \cdots & \mathbf{p}_n\end{bmatrix}}_{\mathcal{P}(\mathbf{E})} = \underbrace{\begin{bmatrix}\hat{\mathbf{p}}_1 & \cdots & \hat{\mathbf{p}}_n\end{bmatrix}}_{\hat{\mathcal{P}}} + \begin{bmatrix}\mathbf{k}_1 & \mathbf{k}_2\end{bmatrix} \mathbf{E} \tag{4.14}$$

The curve parameters $\mathbf{H}(\mathbf{E})$ are calculated from control points $\mathcal{P}(\mathbf{E})$ by solving the linear system (4.4), as described in Section 4.1.2. The cost function C from Section 4.1.3 is optimized with respect to variable $\mathbf{E}$:

$$\mathbf{H}^* = \arg\min_{\mathbf{E}} C\big(\hat{\mathcal{X}}_{\text{inl}}; \mathbf{H}(\mathbf{E})\big), \tag{4.15}$$

The optimization is done using a derivative-free Nelder-Mead downhill simplex method [72] with an initial estimate $\mathbf{E} = \mathbf{0}$.

4.1.6 Implementation details

We use a straight line ($n = 2$ in (4.2)) or a quadratic curve ($n = 3$ in (4.2)) for cases where tool is expected to be bent (Section 4.2.2, 4.2.3). The desired probability for the successful RANSAC termination was set to $\eta = 0.99$. Increasing η further improves the accuracy by $5 \sim 15\%$ at the expense of an increased number of iterations. Local optimization usually terminates in not more than 40 steps, taking about 26%

[1] In Section 4.1.2 a principal direction $\mathbf{k}_0$ is also calculated but only from the n control points.

of the total time. The majority of the memory was used for storing the 3D volume (90 MB for 3D data of size $53 \times 71 \times 3100$ voxels). The RANSAC localization algorithm takes the thresholded points as input which is only a small fraction of the full set of points so the additional memory requirements for the RANSAC are negligible.

All algorithms were implemented in MATLAB[2] on a standard PC with Intel Core 2 processor at 1.83 GHz. Further speedup should be possible by rewriting them in a compiled language such as C++. Our preliminary experiments with implementation of the RANSAC method in C++ show that the speedup is at least by a factor of three.

4.2 Results

The results of the localization algorithm are presented on simulated data, on real ultrasound data of a phantom with an inserted tool, and on real data from a breast biopsy. The true tool position in the simulated data is known; for the real ultrasound data, it was found as a mean of the estimates of ten human observers. The mean variability for human observers was less than 0.4 mm. All 3D visualizations shown in this section were made in Paraview [52].

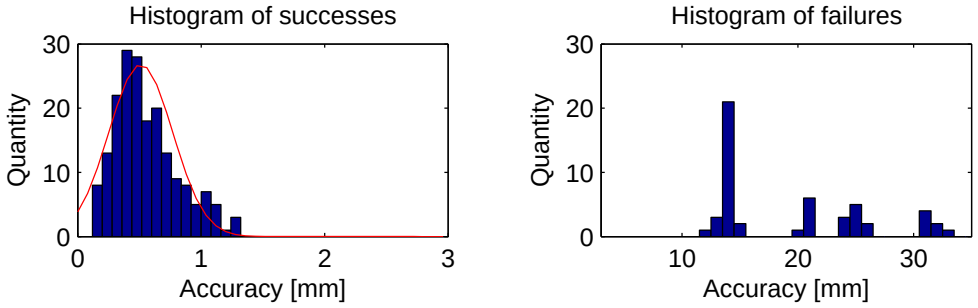
The proposed method is compared to other projection-based localization methods (see also Chapter 2): the randomized Hough transform (RHT) [108], the Quick RHT (QRHT) [81], the PIP method [P2], the multi-resolution PIP (MR-PIP) method [P3] (Chapter 3), and the Fast MR-PIP method [P3]. The number of bins for the RHT and the QRHT is set to 36 and 16, respectively, for each of four dimensions. In the PIP method, an angular resolution of 1° and spatial resolution 0.2 mm is used; its MR version uses three resolution levels, its fast variant (Fast MR-PIP) stops the optimization at the second resolution level. All algorithms use the same method for tip localization (Section 2.7).

In the experiments described below, the randomized algorithms were repeated 30 times (on each test data set) with various random seeds. We report the mean time for each method. For the RANSAC method, we also report the mean number of iterations because the time complexity of the RANSAC depends directly on it. We give an overview of all data sets in Table 4.1. The first three data sets were computed as the envelope of the acquired RF signals with high axial resolution. To study the effect of axial resolution on the localization accuracy, we have performed

[2]The MathWorks, Natick, MA

Data	Resolution	Page	Results
Simulation	53×71×3100	54	Table 4.2
PVA phantom	53×71×3100	56	Table 4.3
PVA phantom	53×71×3100	56	Table 4.4
Turkey Breast	273×376×196	58	Table 4.5
Breast biopsy	273×383×208	59	Table 4.6
Breast biopsy	273×383×208	59	Table 4.7

Table 4.1: *Overview of data sets used for experiments.*



(a) (b)

Figure 4.4: *Example of histograms of axis accuracies measured for* AxShp *model fitting on the PVA cryogel phantom: a) for successful runs ($\varepsilon_{axis} < 3\,\mathrm{mm}$) with a fitted normal distribution shown as red line, b) for failures ($\varepsilon_{axis} \geq 3\,\mathrm{mm}$).*

the localization with the envelope volumes downsampled 10× (in the axial direction); the axis localization accuracy was not significantly affected. The last three data sets were acquired by using a different 3D scanner as standard B-mode images.

4.2.1 Accuracy assessment

Two measures are used to quantify the accuracy of the proposed method [9]. Tip accuracy ε_{tip} (3.9) and axis accuracy ε_{axis} (3.10), measured in millimeters, were defined in Chapter 3, see also Figure 3.3. Angular error, which is also sometimes used to measure localization accuracy, is related to ε_{axis} and the tool length.

The result is considered a failure when $\varepsilon_{axis} \geq 3\,\mathrm{mm}$ or $\varepsilon_{tip} \geq 3\,\mathrm{mm}$. The number of failures is reported separately and accuracy is evaluated only on successful runs. An example of sample histograms for accuracy ε_{axis} in Fig. 4.4a for successful runs shows that accuracies approximately follow a normal distribution, while for failures (Fig. 4.4b) the errors are significantly higher. We have performed statistical significance tests for comparison of the proposed method to the other methods.

Specifically, we do a t-test [38] with a significance level of 5% ($p = 0.05$) for accuracy ε_{axis} and also for the number of failures. The experiments are repeated many times (with various random seeds) so that enough measurements are accumulated for the distribution of the number of failures in order to be approximately normal and the t-test to be applicable. The results of statistical tests are in the following text.

4.2.2 Simulation study

A data set mimicking breast tissue was created using the software package Field II [59]. The simulator parameters were set to imitate an ultrasound scanner Voluson 530D[3] operating at a central frequency of 7.5 MHz. Background scatterers were distributed according to a smoothed real 3D ultrasound image of a breast in order to obtain a realistic inhomogeneous background. A signal corresponding to a 0.6 mm diameter tool (radius 0.3 mm) with metal-like acoustic parameters was created by using highly reflecting scatterers. The background field and the tool field were summed in the radio-frequency (RF) signal domain, and a 3D envelope image was calculated. It is equivalent to adding the tool as scatterers but is faster when many datasets are created with the same background. An example volume of $53 \times 71 \times 3100$ voxels is shown in Figure 4.5.

Parameter learning was performed on a training set of simulated data sets with varying tool locations, distinct from the testing set. The threshold T_I was set to maximize the inlier ratio (Section 4.1.1) to approximately 11 times the mean intensity. The threshold τ was set to 0.6 mm, which was the tool observed radius.

Tool location

Nine training and 19 testing data sets were prepared with varying depths and orientations of the straight tool with respect to the probe. The mean inlier ratio in the testing data was 30% and the mean number of the RANSAC iterations was 60. The results (Table 4.2) show that the proposed methods (the RANSAC with local optimization) are among the fastest and have the best repeatability (smallest standard deviation). The number of failures is nearly zero for the RANSAC, the RHT and the PIP methods. However, the PIP method is still very slow. It can be accelerated (by the MR-PIP) but the robustness is then reduced. There is a statistically significant difference (t-test, $p = 0.05$) between the RANSAC and the RHT

[3] GE Healthcare, UK

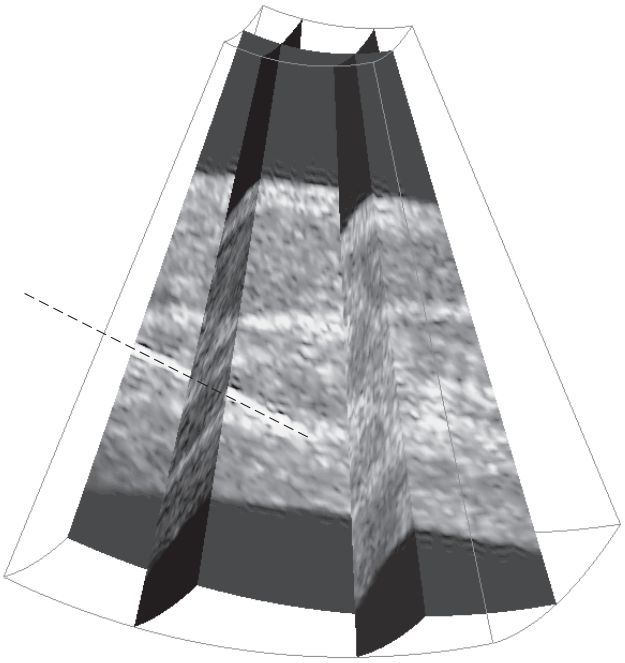


Figure 4.5: *Simulated ultrasound data using Field II with a tool and inhomogeneous background. A 2D slice of a 3D volume with the tool is selected. The estimated tool (marked by a dashed line) is oriented diagonally entering the volume from the left side.*

axis accuracy but not between AxShp and IntDstr results. The RANSAC with the simple AxShp model is the fastest and should be chosen for this data.

Tool deformation

The robustness of the tested methods with respect to tool deformation was evaluated on 21 test data sets. We used parameters learned in the previous experiment. The tool axis was modeled by a quadratic (n = 3) polynomial curve, and the shape was controlled by bending the tool tip. The tip bending offset varied between 0 (straight axis) and 4.0 mm for the total length of 20 mm. The results in Figure 4.6 demonstrate that the axis localization error ε_{axis} (we are not discarding failures in this case) for the RHT and for the QRHT increases when applied on data with a deformed tool. The performance of the proposed method does not deteriorate because it represents the tool axis as a polynomial curve.

Simulated data	Axis acc. [mm]	Tip acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.14 ± 0.09	0.46 ± 0.50	**0%**	0.54
RNS+IntDstr	0.12 ± 0.08	0.48 ± 0.62	**0%**	2.22
RHT	0.38 ± 0.28	0.62 ± 0.59	2%	1.82
QRHT	1.41 ± 0.81	1.42 ± 0.99	28%	1.11
PIP	0.28 ± 0.12	0.53 ± 0.35	5%	69.1
MR-PIP	0.27 ± 0.13	0.45 ± 0.76	11%	29.4
Fast MR-PIP	0.56 ± 0.24	0.85 ± 0.62	11%	12.1

Table 4.2: *Axis accuracy, tip accuracy, percentage of failures and elapsed time for simulated data with varying tool positions and various localization algorithms for the synthetic inhomogeneous background case.*

Signal-to-noise ratio

The robustness of the tested methods with respect to noise was evaluated for data with varying tool intensities with respect to the background. The signal-to-noise ratio (SNR) was defined as:

$$\mathrm{SNR} = \frac{\mathrm{mean}[I(x_{el})]}{\mathrm{mean}[I(x_{bg})]}, \tag{4.16}$$

where x_{el} are voxels whose distance from the true axis is less than a tool radius; the remaining voxels are considered as background x_{bg}. The percentage of failures is reported in Fig. 4.7 for SNR between 1.0 and 2.0 — for higher SNR. All methods localized the tool perfectly and for lower SNR none of the methods found it. A group of nine training and 19 testing data sets were used for each SNR level. Note that the fastest method, the RANSAC+AxShp, has about the same number of failures as the RHT. The RANSAC+IntDstr model is the most robust method.

4.2.3 Experiments on real data

PVA Cryogel phantom

To mimic a biological tissue with a highly reflecting inclusion, a polyvinyl alcohol (PVA) cryogel phantom [45] with size $50 \times 50 \times 50$ mm was created. Inside the phantom there was a thin straight tungsten electrode 150 μm in diameter and 20 mm long. The phantom was scanned eight times from various directions by Voluson 530D ultrasound scanner with a 3D probe operating at a central frequency of 7.5 MHz.

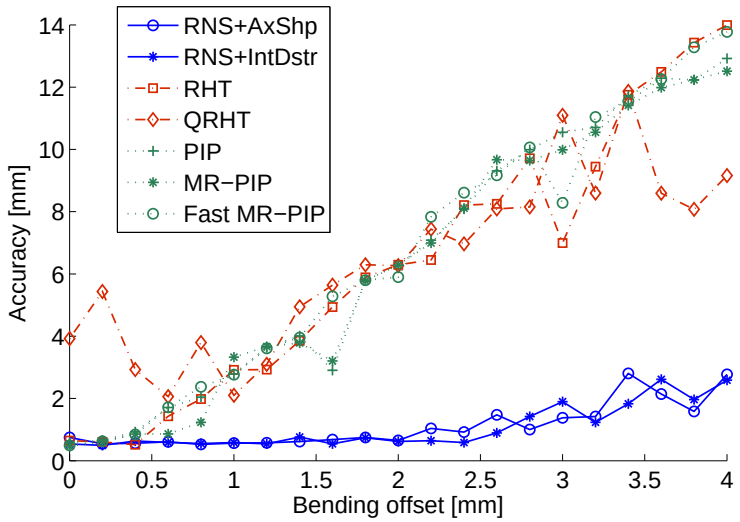


Figure 4.6: *The axis accuracy ε_{axis} for varying tool offsets (bending) in simulated data. The failures were not discarded.*

The axial resolution was approximately 0.4 mm and the lateral resolution 1 mm. The size of acquired volumes was $53 \times 71 \times 3100$ voxels.

All the localization algorithms were tested in two variants: a) we selected a small region of interest (ROI) with the electrode, b) on full volume which also contains the high-intensity phantom boundary. The results of the first experiment are shown in Table 4.3. There is no statistically significant difference (t-test, $p = 0.05$) in the number of failures between the RANSAC and the RHT based methods. The mean inlier ratio was 20% and the number of the RANSAC iterations was between 100 and 200. However, for the second experiment on the full volume in Table 4.4, the percentage of failures is statistically significantly smaller for the RANSAC than for the RHT and the QRHT methods. The mean inlier ratio was only 5% and the number of the RANSAC iterations increased to 1000 ~ 2000. Therefore the total time for the RANSAC also increased.

The proposed RANSAC-based methods are very robust, especially using the IntDstr model, unlike the RHT, the QRHT and the PIP methods which are fooled by the presence of the phantom boundary in the 3D volume. The PIP is the most accurate method, followed by the RANSAC. An example of a localization result can be seen in Fig. 4.1. The AxShp model fitting is the fastest method but using model IntDstr is more robust to the presence of outliers.

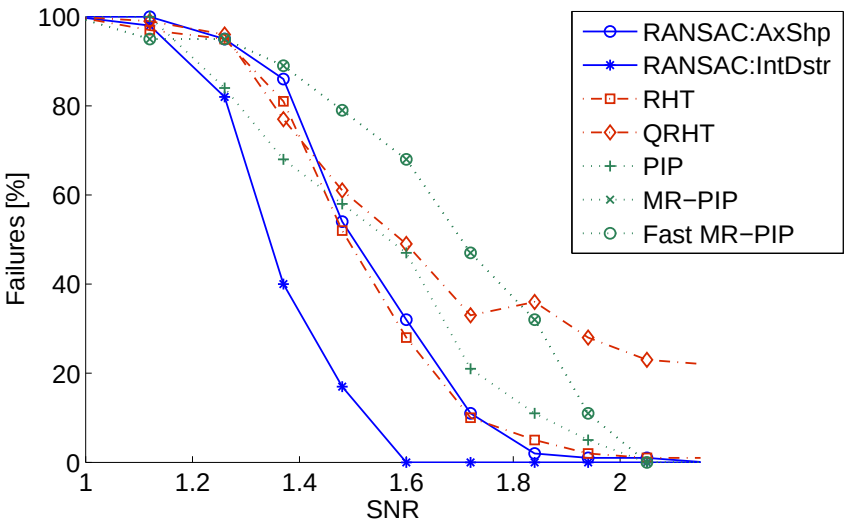


Figure 4.7: *The failure rate for varying SNR (tool contrast) on simulated data.*

Experiments on turkey breast

We acquired three data sets of turkey breasts with a 27-gauge needle (0.41 mm) inserted. The 3D ultrasound images are $273 \times 376 \times 196$ voxels. The central frequency of the 3D probe RSP6-16 RS varied between 10 and 18 MHz so the needle's appearance was different in each data set. Therefore, we had to train on each data set separately. To make training and testing data distinct, we trained on different parts of the image than the part tested. The needle in the breast tissue was deformed in various ways; the maximal bending offset of the tool varied between 0.6 and 1.8 mm. The tool length in the ROI was between 8 and 14 mm.

We selected a ROI containing the needle as well as some high-intensity structures (Fig. 4.8). The results are shown in Table 4.5. The mean inlier ratio was 16% and the number of the RANSAC iterations was between 700 and 1000. There is a statistically significant difference (t-test, $p = 0.05$) between the number of failures for the RANSAC methods and the RHT. The RANSAC has the lowest number of failures but longer processing time compared to other methods. We observed that the RHT often fails on this data because of bent needles, while the proposed methods are successful. The all PIP methods fail on this data.

PVA phantom data	Axis acc. [mm]	Tip acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.47 ± 0.31	0.46 ± 0.26	1%	0.66
RNS+IntDstr	0.91 ± 0.54	0.40 ± 0.32	**0%**	2.62
RHT	0.91 ± 0.44	0.80 ± 0.40	4%	2.13
QRHT	1.44 ± 0.75	0.99 ± 0.60	13%	2.19
PIP	0.33 ± 0.14	0.60 ± 0.21	25%	1268
MR-PIP	0.50 ± 0.31	0.80 ± 0.18	25%	342.1
Fast MR-PIP	0.33 ± 0.14	0.77 ± 0.25	25%	113.4

Table 4.3: *Axis and tip accuracy, the percentage of failures and elapsed time for experiments on the PVA cryogel phantom for the small ROI.*

PVA phantom data	Axis acc. [mm]	Tip acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.56 ± 0.27	0.64 ± 0.27	29%	5.67
RNS+IntDstr	0.55 ± 0.28	0.66 ± 0.31	**14%**	7.90
RHT	0.74 ± 0.33	0.98 ± 0.45	88%	3.81
QRHT	1.52 ± 0.67	0.63 ± 0.03	97%	2.07
PIP	0.34 ± 0.12	0.58 ± 0.12	50%	5615
MR-PIP	0.29 ± 0.12	0.45 ± 0.02	50%	1528
Fast MR-PIP	0.91 ± 0.25	0.62 ± 0.16	75%	537.2

Table 4.4: *Axis and tip accuracy, the percentage of failures and elapsed time for experiments on the PVA cryogel phantom for the full volume.*

Experiments on a breast biopsy

The usability of the proposed method on clinical data is demonstrated on real ultrasound data sets from a breast biopsy acquired by a 3D US scanner GE Voluson E8 with a 12 MHz probe (Figure 4.9). We acquired three data sets with a mostly straight 19-gauge needle (1.092 mm outer diameter) and one data set with a 26-gauge needle. The 3D ultrasound images are $273 \times 383 \times 208$ voxels.

The ROI contains the needle as well as some highly reflecting artifacts. The tip of the needle was located outside of the scanned volume area for three of the data sets, so the tip localization measure was not applicable here. The axis accuracy and the time elapsed for the the various methods are shown in Table 4.6. The last data set contains a thin needle that has a different appearance from the previous group. We trained and tested the localization method on distinct parts of the volume. The results are reported in Table 4.7. In this case, the tip localization accuracy was also

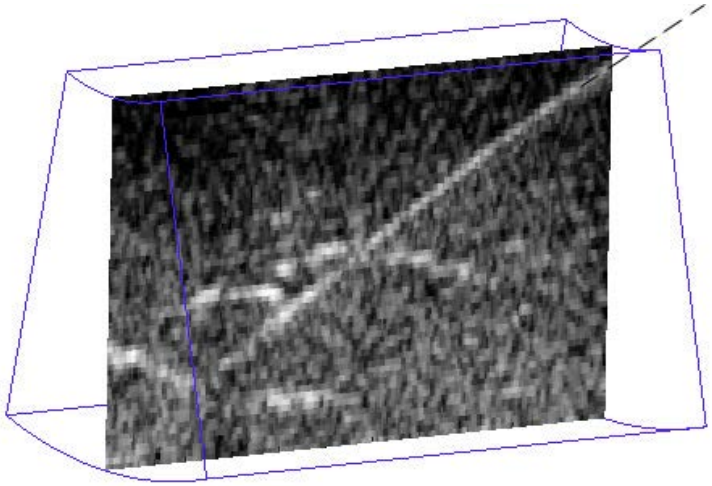


Figure 4.8: *A slice of a 3D data of turkey breast with a needle. A 2D slice of a 3D volume with the needle is selected. The estimated tool (marked by a dashed line) is oriented diagonally entering the volume from the top-right side.*

Turkey breast	Axis acc. [mm]	Tip acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.10 ± 0.05	0.19 ± 0.10	18%	2.24
RNS+IntDstr	0.09 ± 0.05	0.15 ± 0.07	**16%**	4.78
RHT	0.21 ± 0.11	0.13 ± 0.04	74%	1.74
QRHT	1.19 ± 0.14	1.37 ± 0.08	78%	1.37
PIP	*N.A.*	*N.A.*	100%	*N.A.*
MR-PIP	*N.A.*	*N.A.*	100%	*N.A.*
Fast MR-PIP	*N.A.*	*N.A.*	100%	*N.A.*

Table 4.5: *Axis and tip accuracy, percentage of failures and elapsed time for ultrasound data showing a needle in a turkey breast.*

evaluated because the electrode tip is visible.

The data used in this experiment are very challenging, so all the methods have a relatively high failure rate. Nevertheless, the RANSAC-based methods are the fastest, most robust and most accurate. The PIP is also accurate but less robust and two orders of magnitude slower. The RHT is as robust as the RANSAC with speed similar to the IntDstr model but with a worse accuracy.

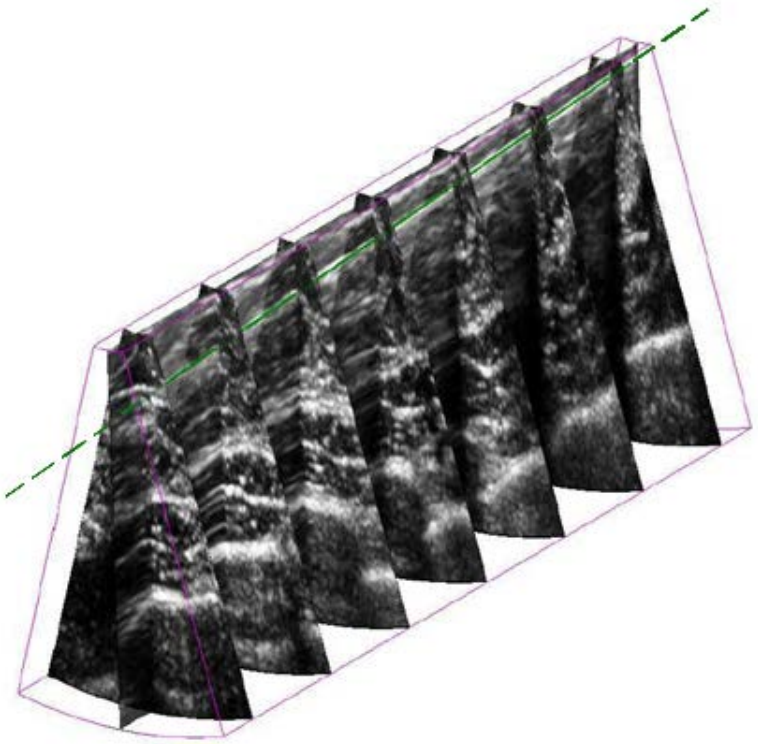


Figure 4.9: *3D view of data from breast biopsy. The boundary geometry of 3D data is marked as a wireframe. There is one planar section of the data with a needle in the upper part and seven perpendicular planar sections. The dashed line shows the estimated needle direction.*

4.3 Conclusions

We proposed a method for the fast and robust tool localization from 3D ultrasound images based on the RANSAC, local optimization. The model fitting uses one simple and one more complicated model. The AxShp model is very fast and easy to implement and works well in most cases. The more complex IntDstr model takes advantage of the learned shape and intensity of the tool and the background. It works better in difficult low-SNR situations but is several times slower.

The proposed RANSAC+AxShp method needs less than 1 s in MATLAB which is approximately the time needed for an acquisition of one 3D volume. A real-time implementation in a compiled language should be perfectly possible. We estimate a throughput of 3 ~ 5 fps on a single processor system. Previous articles [76, 73] showed the feasibility of projection based methods implemented on the GPU. The RANSAC is also suitable for a parallel implementation. The accuracy is better than 1 mm which is enough for all potential applications — visualization, localization and

Biopsy data	Axis acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.20 ± 0.11	33%	0.64
RNS+IntDstr	0.17 ± 0.12	33%	2.29
RHT	0.52 ± 0.57	33%	4.31
QRHT	1.75 ± 0.47	79%	1.88
PIP	0.15 ± 0.0	67%	619
MR-PIP	*N.A*	100%	198
Fast MR-PIP	*N.A.*	100%	28

Table 4.6: *Axis accuracy, tip accuracy, percentage of failures and elapsed time for biopsy ultrasound data with a 19-gauge needle.*

Biopsy data	Axis acc. [mm]	Tip acc. [mm]	Fails [%]	Time [s]
RNS+AxShp	0.25 ± 0.10	0.97 ± 0.25	0%	1.66
RNS+IntDstr	0.24 ± 0.08	1.08 ± 0.49	0%	3.80
RHT	0.35 ± 0.26	0.87 ± 0.48	50%	2.67
QRHT	*N.A.*	*N.A.*	100%	2.5
PIP	*N.A.*	*N.A.*	100%	113
MR-PIP	*N.A.*	*N.A.*	100%	37.0
Fast MR-PIP	*N.A.*	*N.A.*	100%	6.5

Table 4.7: *Axis accuracy, tip accuracy, percentage of failures and elapsed time for biopsy ultrasound data with a 27-gauge needle.*

guidance. The proposed method could be used for initialization of the tracking of the tool in a sequence of 3D ultrasound volumes.

Our experimental results show that the methods proposed here are superior to all other methods tested. They are mostly as robust and fast as the randomized Hough transform (RHT) method while being more accurate, and as accurate as the Parallel Integral Projection (PIP) method while being about two orders of magnitude faster. The proposed methods are also more robust to low tool contrast and tool bending. We have shown that the methods were successfully applied to real ultrasound data.

Chapter 5

Line filtering for tool localization

In this chapter, we present a method for enhancement and localization of surgical tools in 3D ultrasound (US) images. Our first aim is to enhance the visual contrast of the tool to aid with visual localization. Secondly, we want to apply the new appearance description to improve our previously described automatic localization method (Chapter 4), particularly its robustness with respect to the background noise.

Previous localization methods assume the tool to appear as a high intensity cylinder in 3D US images. An additional assumption about the tool, that it is a one-dimensional (1D) object, is used to distinguish the tool from other high-intensity structures. These structures are usually of biological nature (such as bones or fat tissue) and they are zero-dimensional (isolated points) or two-dimensional (layers of echogenic tissue).

A review of line filtering methods is in Section 5.1. The robustness of localization algorithm to background noise is increased by applying an appearance based tool model using line filtering (Section 5.2). The proposed model fitting is fast and accurate enough after pre-processing step which is approved by results (Section 5.3). Initial part of this work has been published at the conference *IEEE International Ultrasonics Symposium* (IUS) 2009 [P4].

5.1 Review of line filtering methods

Line filtering is a method for enhancement of tubular structures. It is used for vessel enhancement and detection, e.g., in the MR angiography (MRA) [78, 95]. Vessel enhancement techniques are used also for improved segmentation and visualization

in medical imaging. The diameter of surgical tool is fixed in the tool localization task (unlike the varying diameter of vessels) which makes the task easier.

Extensive taxonomy of vessel extraction techniques and algorithms can be found in [63]. There are three groups of such methods: 1) matched filters (Section 5.1.1) which use a set of filters with varying orientation; 2) ridge analysis (Section 5.1.2) based on differential geometry and second order analysis; 3) steerable filters (Section 5.1.3) which compute the result of line filtering by interpolation from a fixed set of filter responses. Model based methods, which follow the top-down strategy, offer an alternative to local filtering. References to particular works are in the following sections.

5.1.1 Matched filters

Chaudhuri et al. [18] observed three interesting properties of blood vessels in 2D retinal images: 1) small curvatures; 2) lower mean intensity pixels compared to the background; 3) the intensity profile approximately corresponding to the Gaussian curve in a direction perpendicular to axis. Similar properties are valid also for linear tools in US images.

The local line filter should have a similar shape as the intensity profile itself and it should be bounded. The intensity profile at point $\mathbf{x}$ is defined by using a center line $c_{\mathbf{v}}$ passing through the origin and having orientation $\mathbf{v}$. The 2D filter kernel is expressed as a function of distance $d(\mathbf{x}, c_{\mathbf{v}})$ to the line $c_{\mathbf{v}}$:

$$K_{\mathbf{v}}(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)} e^{-\frac{d(\mathbf{x},c_{\mathbf{v}})^2}{2\sigma^2}}, \text{ for } \|x\| \leq \frac{L}{2} \tag{5.1}$$

where L is the maximum length of the linear segment. We note that other templates can be used for line detection, e.g. ridge template [58] which is similar to (5.1) but smoothed in all directions.

A set of matched filters $K_{\mathbf{v}_i}$ with various orientations is prepared so that the set $\{\mathbf{v}_i\}_{i=1}^{N}$ is sampled on unit sphere. Unknown direction of the line at pixel $\mathbf{x}$ is determined as the direction of the strongest response of all filters:

$$\hat{\mathbf{v}}(\mathbf{x}) = \arg\max_{\mathbf{v}_i} \left(K_{\mathbf{v}_i}(\mathbf{x}) * I(\mathbf{x})\right) \tag{5.2}$$

We find both the orientation $\mathbf{v}$ and the strength J of the linear structure

$$J_{[\mathrm{MF}]}(\mathbf{x}) = K_{\hat{\mathbf{v}}(\mathbf{x})}(\mathbf{x}) * I(\mathbf{x}) \tag{5.3}$$

where $*$ is a convolution.

Chaudhuri uses matched filters for detection of lines in 2D images. Twelve various orientations with angular distance 15° are prepared. The kernel width σ is determined manually but in later works [53], it is found by optimization. Increasing the number of filters improves the accuracy but decreases the speed. The speed is also influenced by the size of convolution mask. The size of the convolution mask in pixels is chosen proportionally to 3σ.

Poli and Valli [80] developed an algorithm based on matched-filters and thresholding with hysteresis which works at real-time. Proposed filters are sensitive to thin vessels in coronary angiograms. Hoover at al. [53] classify each pixel using local and region-based properties. The response of the matched filter is examined by iteratively decreasing the threshold of the classifier.

5.1.2 Hessian based methods

When we look at 2D image as the elevation map, lines can be seen as ridges or valleys. Ridge and valley is a concept of local maxima and minima which is generalized to N dimensions ($N \geq 1$) [34]. 1D Ridge is defined as a curve such that each point is a local extrema in a subspace orthogonal to the ridge, i.e., each point of 1D ridge in 2D image has a local extrema (maximum, resp. minimum) at a direction perpendicular to the ridge.

We are interested in 1D ridges in 3D with an approximately constant intensity. Tubular tools in 3D US images are enhanced by using filtering methods based on 1D ridge analysis mentioned here. The filters are based on directional second derivatives of image smoothed by the Gaussian kernel. We follow a concept of tubular objects [41] when such objects are determined by relationship (5.12) of eigenvalues of the Hessian matrix. Frangi [41] and Sato [84] use a 3D multi-scale enhancement filter to adapt the filter to the scale of tubular structure.

Local properties of linear structure

Intensity variations of voxel $\mathbf{x}$ in its local neighborhood are analyzed via second order derivatives. The image function $I(\mathbf{x})$ is approximated using Taylor expansion in the proximity of point $\mathbf{x}$ as

$$I(\mathbf{x}+\mathbf{d}) \approx I(\mathbf{x}) + \mathbf{d}^{\mathrm{T}} \bigtriangledown I(\mathbf{x}) + \frac{1}{2}\mathbf{d}^{\mathrm{T}}\mathcal{H}(\mathbf{x})\mathbf{d}, \tag{5.4}$$

where $\mathbf{d}$ is a small perturbation vector. The gradient vector

$$\nabla I(\mathbf{x}) = \begin{bmatrix} I_x(\mathbf{x}) & I_y(\mathbf{x}) & I_z(\mathbf{x}) \end{bmatrix} \tag{5.5}$$

and the Hessian matrix

$$\mathcal{H}(\mathbf{x}) = \begin{bmatrix} I_{xx}(\mathbf{x}) & I_{xy}(\mathbf{x}) & I_{xz}(\mathbf{x}) \\ I_{yx}(\mathbf{x}) & I_{yy}(\mathbf{x}) & I_{yz}(\mathbf{x}) \\ I_{zx}(\mathbf{x}) & I_{zy}(\mathbf{x}) & I_{zz}(\mathbf{x}) \end{bmatrix} \tag{5.6}$$

are computed by the convolution of the image function I with derivatives of the isotropic Gaussian [41]:

$$I_\alpha(\mathbf{x}) = I(\mathbf{x}) * \frac{\partial}{\partial\alpha} G(\mathbf{x}, s) \tag{5.7}$$

$$I_{\alpha\beta}(\mathbf{x}) = I(\mathbf{x}) * \frac{\partial^2}{\partial\alpha\partial\beta} G(\mathbf{x}, s) \tag{5.8}$$

where $\alpha, \beta \in \{x, y, z\}$ are directions of derivatives and s is a scale of Gaussian. The Gaussian G is defined as:

$$G(\mathbf{x}, s) = \frac{1}{(2\pi s^2)^{\frac{3}{2}}} e^{-\frac{\|\mathbf{x}\|^2}{2s^2}} \tag{5.9}$$

Scale s of the Gaussian $G(\mathbf{x}, s)$ corresponds to apparent radius of the tool and is learned from training set of images. The learning is done by maximizing the response of the Gaussian filter given ground-truth orientation of linear structure.

The second order directional derivative at $\mathbf{v}$ can be obtained from the Hessian matrix

$$\frac{\partial^2 I}{\partial \mathbf{v}^2}(\mathbf{x}) = \mathbf{v}^{\mathrm{T}} \mathcal{H}(\mathbf{x}) \mathbf{v}, \;\; \mathbf{v} \in \mathbb{R}^3, \; \|\mathbf{v}\| = 1 \tag{5.10}$$

where $\mathbf{v}$ is unit directional vector of chosen orientation. It is used later by oriented line filters (Section 5.1.3) with a priori known principal orientation.

Eigenanalysis of $\mathcal{H}$. Eigenanalysis of the Hessian matrix $\mathcal{H}(\mathbf{x})$ characterizes the second order local structure. Let $|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$ be its eigenvalues and $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ corresponding eigenvectors. The local principal direction is:

$$\hat{\mathbf{v}}(\mathbf{x}) = \mathbf{e}_1 \tag{5.11}$$

If all $\lambda_i \approx 0$ ($i = 1, 2, 3$) are close to zero, there is no noticeable structure. For planar structures, $|\lambda_3| \gg |\lambda_2| \approx |\lambda_1| \approx 0$. For blob-like structures, $|\lambda_i| > 0$ ($i = 1, 2, 3$) and $\lambda_1 \approx \lambda_2 \approx \lambda_3$. For voxels in a tubular structure, one eigenvalue λ_1 is small and the other two λ_2, λ_3 large:

$$(0 \approx |\lambda_1|) \text{ and } (|\lambda_1| \ll |\lambda_2|) \text{ and } (\lambda_2 \approx \lambda_3) \tag{5.12}$$

Various *tubularness* measures $J(\mathbf{x})$ have been proposed for line enhancement in 3D images (see also examples of filtering of 3D US images in Figures 5.1, 5.2 and 5.3). Their experimental comparison is presented in Section 5.3. We assume a bright cylinder on dark background so the voxels which do not satisfy the condition

$$(\lambda_3 < 0) \text{ and } (\lambda_2 < 0) \tag{5.13}$$

are discarded and $J(\mathbf{x})$ set to 0.

Li's measure

A simple formula for line filtering has been proposed by Li et al. [68]

$$J_{[\mathrm{Li}]}(\mathbf{x}) = \frac{|\lambda_2|}{|\lambda_3|}\left(|\lambda_2| - |\lambda_1|\right) \tag{5.14}$$

which penalizes the large $|\lambda_3|$ compared to the $|\lambda_2|$. The term (5.14) is maximized when ratio $\frac{|\lambda_2|}{|\lambda_3|}$ is close to 1 and difference $|\lambda_2| - |\lambda_1|$ is large.

Sato's measure

Sato et al. [84] uses a notation for eigenvalues assuming $\lambda_1 > \lambda_2 > \lambda_3$ which is equivalent to Frangi's notation ($|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$) when all $\lambda_1, \lambda_2, \lambda_3 \leq 0$. Ideal tubular structure in 3D image should have eigenvalues satisfying:

$$(\lambda_1 \approx 0) \text{ and } (\lambda_2 \approx \lambda_3 < 0) \tag{5.15}$$

The absolute values of λ_2 and λ_3 are relatively large compared to λ_1. The condition (5.15) is logically equivalent to conjunction of conditions (5.12) and (5.13) which is used by Li and Frangi [41].

Based on the condition $\lambda_2 \approx \lambda_3 < 0$, a measure of tubularness is introduced

$$\lambda_{23} = \begin{cases} |\lambda_3| \left(\frac{\lambda_2}{\lambda_3}\right)^{\gamma_{23}}, & \text{if } \lambda_3 < \lambda_2 < 0, \\ 0, & \text{otherwise} \end{cases} \tag{5.16}$$

where $\gamma_{23} \geq 0$ controls sharpness of the function. The term (5.16) is maximized when $\frac{|\lambda_2|}{|\lambda_3|}$ is close to 1 and both $|\lambda_3|$ and $|\lambda_3|$ are large. The condition $\lambda_1 \approx 0$ is combined with λ_{23} to

$$\lambda_{123} = \lambda_{23} w_{12}, \qquad \text{if } \lambda_3 < \lambda_2 < 0, \tag{5.17}$$

where w_{12} is a weight function that decreases as λ_1 deviates from zero

$$w_{12} = \begin{cases} \left(1 + \frac{\lambda_1}{|\lambda_2|}\right)^{\gamma_{12}}, & \text{if } \lambda_1 \leq 0, \\ \left(1 - \alpha \frac{\lambda_1}{|\lambda_2|}\right)^{\gamma_{12}}, & \text{if } \frac{|\lambda_2|}{\alpha} > \lambda_1 > 0, \\ 0, & \text{otherwise} \end{cases} \tag{5.18}$$

where $\gamma_{12} \geq 0$ and $0 < \alpha \leq 1$. Parameter α is introduced in order to give w_{12} asymmetrical characteristics in negative and positive regions of λ_1.

The function $J_{[\text{Sato}]}(\mathbf{x}) = \lambda_{123}$ is typically used with parameters $\gamma_{23} = \gamma_{12} = 1$ or $\gamma_{23} = \gamma_{12} = 0.5$ as proposed by Sato [84].

Frangi's measure

An advanced tubularness measure has been proposed by Frangi et al. [41]. He introduces three natural quantities: $\mathcal{R}_\mathcal{B}$ for quantification of the relative amplitude of $|\lambda_1|$, $\mathcal{R}_\mathcal{A}$ for discrimination of tubular structures from planar structures, and S for quantification of strength of all second order features:

$$\mathcal{R}_\mathcal{B} = \frac{|\lambda_1|}{\sqrt{|\lambda_2 \lambda_3|}}, \quad \mathcal{R}_\mathcal{A} = \frac{|\lambda_2|}{|\lambda_3|}, \tag{5.19}$$

$$\mathcal{S} = \| \mathcal{H} \|_F = \sqrt{\sum_{j=1,2,3} \lambda_j^2} \tag{5.20}$$

Two quantities $\mathcal{R}_\mathcal{A}$, $\mathcal{S}$ should be maximized ($\mathcal{R}_\mathcal{A}$ only up to 1) and $\mathcal{R}_\mathcal{B}$ should be low or close to 0. They are combined into a single tubularness measure [41]:

$$J_{[\text{Fra}]}(\mathbf{x}) = \left(1 - e^{-\frac{\mathcal{R}_\mathcal{A}^2}{2\alpha^2}}\right)\left(e^{-\frac{\mathcal{R}_\mathcal{B}^2}{2\beta^2}}\right)\left(1 - e^{-\frac{\mathcal{S}^2}{2c^2}}\right) \tag{5.21}$$

Frangi et al. [41] recommends to set $\alpha = \beta = 0.5$; the parameter c is set experimentally.

Hessian based localization

Aylward et al. [7] approximate the medial axes of tubular vessels as oriented intensity ridges. Ridges are tracked from a user-supplied starting point by estimating the local directions of the tube with respect to the Hessian matrix. The local widths of the object is estimated by using points on the ridges. It requires a fair amount of user intervention (around 100 mouse clicks).

Guo et al. [49] propose a ridge extraction method that first applies median filter and then non-linear anisotropic smoothing. Adaptive thresholding is used to cut the cost of the ridge extraction and reduce false ridges. The vessel centerlines are connected using a curve relaxation process.

Staal et al. [89] use the Hessian based method to detect ridges in 2D images followed by a learn-able selection scheme. Ridge pixels are grouped by the similarity of eigenvector directions, and straight line elements are formed out of them. Image is partitioned according to the closest line element. Pixels are classified according to a set of features computed in a local coordinate frame.

5.1.3 Steerable filters

Matched filter based detection (Section 5.1.1) is computationally demanding because a large number of non-separable filters need to be applied. A more efficient approach is to use only a few filters (corresponding to a few angles) and interpolate between their responses. A term *steerable filter* is used for such a class of filters in which filter of arbitrary orientation is synthesized as a linear combination of a set of *basis filters* [42, 43]. We will show here that elements of the Hessian matrix $\mathcal{H}$ also constitute a basis for a steerable filter.

Basis filters are designed by using partial derivatives of the Gaussian function. For the first order directional derivative of I, it is enough to pre-compute three basis functions I_x, I_y and I_z as in (5.7). Response of the filter at given direction $\mathbf{v}$ is computed by linear combination

$$\frac{\partial I}{\partial \mathbf{v}}(\mathbf{x}) = v_x I_x(\mathbf{x}) + v_y I_y(\mathbf{x}) + v_z I_z(\mathbf{x}) \tag{5.22}$$

where $\mathbf{v} = [v_x, v_y, v_z]$ is unit directional vector of chosen orientation $\|\mathbf{v}\| = 1$.

The second order directional derivative of the image I is computed by convolution with the partial derivatives of the Gaussian $\frac{\partial^2 G(\mathbf{x},s)}{\partial \mathbf{v}^2}$ (5.8). The construction of steerable Gaussian filter requires six basis filters [25]

$$\frac{\partial^2 I}{\partial \mathbf{v}^2}(\mathbf{x}) = \sum_{i=1}^{6} k_i(\mathbf{v}) \left(I(\mathbf{x}) * B_i(\mathbf{x}, s)\right) \tag{5.23}$$

where $B_i(\mathbf{x})$ are the basis functions and $k_i(\mathbf{v})$ are the interpolation scalars

$$\begin{aligned}
B_1(\mathbf{x}, s) &= (2x_1^2 - 1)G(\mathbf{x}, s), & k_1(\mathbf{v}) &= v_1^2 \\
B_2(\mathbf{x}, s) &= (2x_1x_2 - 1)G(\mathbf{x}, s), & k_2(\mathbf{v}) &= 2v_1v_2 \\
B_3(\mathbf{x}, s) &= (2x_2^2 - 1)G(\mathbf{x}, s), & k_3(\mathbf{v}) &= v_2^2 \\
B_4(\mathbf{x}, s) &= (2x_1x_2 - 1)G(\mathbf{x}, s), & k_4(\mathbf{v}) &= 2v_1v_3 \\
B_5(\mathbf{x}, s) &= (2x_2x_3 - 1)G(\mathbf{x}, s), & k_5(\mathbf{v}) &= 2v_2v_3 \\
B_6(\mathbf{x}, s) &= (2x_3^2 - 1)G(\mathbf{x}, s), & k_6(\mathbf{v}) &= v_3^2
\end{aligned} \tag{5.24}$$

where $\mathbf{x} = [x_1, x_2, x_3]$ are three coordinates, $G(\mathbf{x}, s)$ is the 3D Gaussian (5.9) and $\mathbf{v} = [v_1, v_2, v_3]$ is unit directional vector of chosen orientation.

The expression in (5.23) is similar to obtaining the second order directional derivative from the Hessian matrix $\mathcal{H}$ (5.10). The interpolation scalars $k_i(\mathbf{v})$ appears as scalar factors in the expansion of (5.10)

$$\mathbf{v}^{\mathrm{T}}\mathcal{H}(\mathbf{x})\mathbf{v} = \underbrace{v_1^2}_{k_1(\mathbf{v})} I_{xx} + \underbrace{2v_1v_2}_{k_2(\mathbf{v})} I_{xy} + \underbrace{v_2^2}_{k_3(\mathbf{v})} I_{yy} + \underbrace{2v_1v_3}_{k_4(\mathbf{v})} I_{xz} + \underbrace{v_2v_3}_{k_5(\mathbf{v})} I_{yz} + \underbrace{v_3^2}_{k_6(\mathbf{v})} I_{zz} \tag{5.25}$$

and the convolution of the image $I(\mathbf{x})$ and basis functions $B_i(\mathbf{x}, s)$ correspond to the second order partial derivatives of the image $I(\mathbf{x})$:

$$B_1 \leftrightarrow I_{xx} * I(\mathbf{x}), \;\; B_2 \leftrightarrow I_{xy} * I(\mathbf{x}), \;\; B_3 \leftrightarrow I_{yy} * I(\mathbf{x}), \tag{5.26}$$

$$B_4 \leftrightarrow I_{xz} * I(\mathbf{x}), \;\; B_5 \leftrightarrow I_{yz} * I(\mathbf{x}), \;\; B_6 \leftrightarrow I_{zz} * I(\mathbf{x}) \; . \tag{5.27}$$

Since two methods for computation of $\frac{\partial^2 I}{\partial \mathbf{v}^2}(\mathbf{x})$ are similar (equations (5.10) and (5.23)), we implement only the Hessian based computation in our experiments (Section 5.3).

Line detection with oriented filter. We want to design an oriented line filter $\zeta(\mathbf{x}, \mathbf{v})$ for detection of tool at orientation $\mathbf{v}$. The oriented line filter can use one of two methods for computing the second order directional derivatives $\frac{\partial^2 I}{\partial \mathbf{v}^2}(\mathbf{x})$ as de-

scribed above (equations (5.10) and (5.23)). A more informative notation $\zeta(\mathbf{x}, \mathbf{v}, \mathbf{K})$ is used where $\mathbf{K}$ specifies either the Hessian matrix $\mathcal{H}$ or the steerable basis (5.24).

The estimated principal orientation $\hat{\mathbf{v}}$ of the line filter corresponds to the first eigenvector $\hat{\mathbf{e}}_\mathbf{1}$ of the Hessian matrix $\mathcal{H}$. The other two eigenvectors $\hat{\mathbf{e}}_\mathbf{2}$, $\hat{\mathbf{e}}_\mathbf{3}$ form a plane perpendicular to the tubular structure. In case of using steerable technique (5.23), the local direction $\hat{\mathbf{v}}$ is determined by maximization of the filter's response over all directions $\mathbf{v}$ similarly to matched filters (Section 5.1.1).

The 3D line filter oriented along $\mathbf{v} = \hat{\mathbf{e}}_\mathbf{1}$ is defined as [43]:

$$\zeta_1(\mathbf{x}, \mathbf{v}) = \frac{\partial^2 I}{\partial \hat{\mathbf{e}}_\mathbf{2}^2}(\mathbf{x}) + \frac{\partial^2 I}{\partial \hat{\mathbf{e}}_\mathbf{3}^2}(\mathbf{x}), \tag{5.28}$$

Jacob and Unser [58] showed that there is a better alternative using Canny-like criteria by maximization of the signal to noise ratio (SNR). The following line filter is proposed

$$\zeta_2(\mathbf{x}, \mathbf{v}) = -\frac{2}{3}\frac{\partial^2 I}{\partial \hat{\mathbf{e}}_\mathbf{1}^2}(\mathbf{x}) + \frac{\partial^2 I}{\partial \hat{\mathbf{e}}_\mathbf{2}^2}(\mathbf{x}) + \frac{\partial^2 I}{\partial \hat{\mathbf{e}}_\mathbf{3}^2}(\mathbf{x}) \tag{5.29}$$

which is claimed to have a better sensitivity to the orientation of linear structures than filter $\zeta_1(\mathbf{x}, \mathbf{v})$ (5.28).

Line detection with steerable filters. The steerable based line detection has been used also for the 3D MRA images [48, 104]. Weiping et al. [104] created a 3D steerable filter based on dyadic B-spline wavelets. González et al. [48] use steerable features for statistical 3D dendrite detection. Rotationally invariant features that include higher-order derivatives are used. Then a classifier is learned on them which allows to adapt the filter to particular properties of filaments without manual intervention. The method is claimed to be more robust to the presence of noise than Hessian based methods.

5.2 Tool localization with line filtering

The tool localization method proposed here improves the previously described method in Chapter 4 and it consists of two steps.

In the first step, the line filtering method is applied to calculate the tubularness $J(\mathbf{x})$ and the local orientation $\hat{\mathbf{v}}(\mathbf{x})$ using one of the methods in Section 5.1.2. The variable $\mathbf{K}(\mathbf{x})$ contains the Hessian matrix (5.6) which is used later in tool

localization. The attributes

$$\mathbf{m}_1(\mathbf{x}) = \left[I(\mathbf{x})\ J(\mathbf{x})\right], \quad (\mathbf{x} \in \mathcal{X}) \tag{5.30}$$

are used in pre-segmentation to reduce the amount of voxels not belonging to the tool (Section 5.2.1). This is a generalization of the simple thresholding from the method used in Chapter 4.

The subsequent tool localization uses all attributes from the previous step

$$\mathbf{m}_2(\mathbf{x}) = \left[I(\mathbf{x})\ J(\mathbf{x})\ \hat{\mathbf{v}}(\mathbf{x})\ \mathbf{K}(\mathbf{x})\right], \quad (\mathbf{x} \in \mathcal{X}) \tag{5.31}$$

The tool axis is estimated by model fitting based on the RANSAC procedure (Chapter 4). The subsequent local optimization step uses the cost function based on the shape description (Section 4.1.3). Additional variables $\hat{\mathbf{v}}(\mathbf{x})$ or $\mathbf{K}(\mathbf{x})$ are used for computation of consistency of the local orientation with the curve (Section 5.2.2).

5.2.1 Pre-segmentation

Tentative tool voxels $\mathcal{X}_\mathrm{t} \subseteq \mathcal{X}$ are selected according to the attribute vector $\mathbf{m}_1(\mathbf{x})$ (5.30)

$$\mathcal{X}_t = \{\mathbf{x} \in \mathcal{X} \mid h(\mathbf{x}) = 1\} \tag{5.32}$$

where $h(\mathbf{x}) \in \{0, 1\}$ is a binary classifier, determining the tool ($h(\mathbf{x}) = 1$) and background voxels ($h(\mathbf{x}) = 0$). The attribute $J(\mathbf{x})$ was pre-computed by line filtering (Section 5.1.2). We discuss several options for classifier in the following text.

Thresholding

The simple approach is to use thresholding of intensities $I(\mathbf{x})$

$$h_\mathrm{thresh}(\mathbf{x}) = 1 \quad \Leftrightarrow \quad I(\mathbf{x}) \geq I_T \tag{5.33}$$

where I_T is a threshold estimated on the training data.

The thresholding was used for pre-segmentation in Chapter 4 and in [P1]. It is an efficient method for reducing the amount of data. It is much faster than the following methods based on line filtering but it is not so precise in recognition of tubular structures from other, e.g. 2D, structures.

Linear classifier

Decision function is defined as monolithic linear classifier [33] using the attribute vector $\mathbf{m}_1(\mathbf{x})$ (5.30):

$$h_{\text{linear}}(\mathbf{x}) = 1 \quad \Leftrightarrow \quad \mathbf{w} \cdot \mathbf{m}_1(\mathbf{x}) \geq w_0 \tag{5.34}$$

where $\mathbf{w}$ is a weight vector and w_0 is a bias. Parameters $\mathbf{w}$ and w_0 are learned from the training data with ground truth as the Fisher's linear discriminant (FLD) [33] or by using a modified perceptron learning for linearly non-separable case. We use a pocket algorithm [46] which keeps the best solution seen so far. The pocket algorithm returns the solution in the pocket, rather than the last solution (Figure 5.4a).

Pre-segmentation using two attributes gives better results (in terms of true positives and true negatives) than doing thresholding (Chapter 4) at the expense of additional time for line filtering. The results are reported in Section 5.3.

Support Vector Machine

Support Vector Machine (SVM) classifier [33] is defined as:

$$h_{\text{SVM}}(\mathbf{x}) = 1 \quad \Leftrightarrow \quad (\mathbf{w} \cdot \mathbf{k}_S(\mathbf{x}) \geq w_0) \tag{5.35}$$

where discriminant function is defined by using dot product of weight vector $\mathbf{w}$ and the vector of kernel functions

$$\mathbf{k}_S(\mathbf{x}) = [k(\mathbf{m}_1(\mathbf{x}), \mathbf{s}_1), ..., k(\mathbf{m}_1(\mathbf{x}), \mathbf{s}_d)]^T \tag{5.36}$$

centered at support vectors $\mathcal{S} = \{\mathbf{s}_1, ..., \mathbf{s}_d\}, \mathbf{s}_i \in \mathbb{R}^n$ which are usually subset of the training data. The kernel functions $k(\mathbf{x}_1, \mathbf{x}_2) = \boldsymbol{\phi}(\mathbf{x}_1) \cdot \boldsymbol{\phi}(\mathbf{x}_2)$ are defined as a dot product of transformed vectors $\boldsymbol{\phi}(\mathbf{x}_1)$ and $\boldsymbol{\phi}(\mathbf{x}_2)$.

If the kernel function $\boldsymbol{\phi}(\mathbf{x})$ is an identical function then the SVM is, in fact, a linear classifier. The kernel function $\boldsymbol{\phi}(\mathbf{x})$ can use a polynomial mapping which allows to transform a non-linearly separable data into higher dimension and solve it there using a linear classifier. Quadratic mapping is used for the $\boldsymbol{\phi}^q(\mathbf{x})$:

$$\boldsymbol{\phi}^q(\mathbf{x}) = \left[x_1, x_2, \ldots, x_n,\ x_1^2, x_1x_2, \ldots, x_1x_n,\ x_2^2, \ldots, x_2x_n,\ \ldots\ x_n^2\right] \tag{5.37}$$

where x_i denotes the i-th attribute of the vector $\mathbf{x}$.

For training of the SVM, the pattern recognition toolbox from [39] or the faster implementation SVM Light [61] for large datasets can be used. The library LIBOCAS [40] implements the fastest SVM training method known to us.

Cascaded classifier

We have tested a cascade of two classifiers: the first one is thresholding of intensities (5.33); the second one is linear classifier of two features (5.34)

$$f_0 = h_{\text{thresh}}, \quad f_1 = h_{\text{linear}} \tag{5.38}$$

where f_1 is applied only on the voxels which were positively classified by classifier f_0. More time consuming classifier f_1 is run only on a small portion of surviving candidates. Such an architecture saves the computational time by early refusing of a large portion of voxel candidates with low intensities.

The cascade is trained in two phases: first, we find a threshold as the FLD and adjust it so that 80% of the FPs are achieved on the testing set. Then, a linear classifier is learned by perceptron learning of $\mathbf{m}_1(\mathbf{x})$ on a subset of surviving training samples. The found solution is usually close to other tested classifiers (Section 5.3). An example of cascaded classification is in Figure 5.4b.

AdaBoost and WaldBoost

The AdaBoost (adaptive boosting) is a learning meta-algorithm. [44] Given a labelled training set and a set of weak (usually very simple) classifiers $\mathcal{G}$, the AdaBoost produces a strong classifier

$$h_{\text{AdaB}}(\mathbf{x}) = \sum_{i=1}^{I} h^{(i)}(\mathbf{m}_1(\mathbf{x})) \tag{5.39}$$

where $h^{(i)} \in \mathcal{G}$ is selected sequentially during training. The resulting strong classifier (which is binary in our case) maximizes the likelihood ratio of the tool class and the background class.

Šochman et al. [88] proposed an algorithm WaldBoost which integrates the AdaBoost training with the Wald's sequential probability ratio test (SPRT). The SPRT gives the termination criterion for the evaluation in the cascade. Two additional parameters have to be specified: the desired FN and FP rate. The WaldBoost can be viewed as a theoretically justified boosted cascade of classifiers proposed by Viola

and Jones [97].

Weak classifiers, which can be used in the AdaBoost or WaldBoost, can have the accuracy only a little better than random guessing. An example of a weak hypothesis is thresholding of a single feature or randomly oriented linear classifier. Domain partitioning weak classifiers [85] divides the feature into K bins. In each bin, the response of the weak classifier is computed from the sum of positive and negative samples falling into the bin.

5.2.2 Tool model

The localization algorithm introduced in Chapter 4 uses shape and appearance model for the tool in 3D US. Here, we propose an extended appearance model which uses intensities $I(\mathbf{x})$ as well as the data pre-computed by line filtering methods from Section 5.1.2, namely tubularness $J(\mathbf{x})$, principal direction $\hat{\mathbf{v}}$ and the local characteristics $\mathbf{K}(\mathbf{x})$ stored in $\mathbf{m}_2(\mathbf{x})$ (5.31).

Model fitting. We recall that the tool model is used by the localization algorithm (Chapter 4) and it consists of a classification function $q(\mathbf{x}, \mathbf{H}) \in \{0, 1\}$ classifying voxel $\mathbf{x}$ as either a tool ($q = 1$) or a background ($q = 0$); and a cost function $C(\mathcal{X}_{\text{inl}}; \mathbf{H}) \in \mathbb{R}$ quantifying how well the model parameters $\mathbf{H}$ fit the set of observations $\mathcal{X}_{\text{inl}}$ consistent with the model.

The tool shape, i.e. curve parameters $\mathbf{H}$, is first estimated roughly by maximizing the number of tool voxels (inliers set $\mathcal{X}_{\text{inl}}$) as determined by q via RANSAC

$$\hat{\mathbf{H}} = \arg\max_{\mathbf{H}} \sum_{\mathbf{x} \in \mathcal{X}_t} q(\mathbf{x}, \mathbf{H}) \tag{5.40}$$

on the pre-segmented set of voxels $\mathcal{X}_t$. The estimated inliers set $\hat{\mathcal{X}}_{\text{inl}}$ is determined as a subset of $\mathcal{X}_t$ containing only voxels with $q(\mathbf{x}, \hat{\mathbf{H}}) = 1$, i.e. *inliers*. Finally, the tool parameters $\hat{\mathbf{H}}$ are refined by minimization of the cost function

$$\mathbf{H}^* = \arg\min_{\mathbf{H}} C(\hat{\mathcal{X}}_{\text{inl}}, \mathbf{H}) \tag{5.41}$$

on the set of inliers $\hat{\mathcal{X}}_{\text{inl}}$ (Chapter 4). In this chapter, we use the cost function (4.6) from Section 4.1.3, which is based on the sum of squared distances of inliers $\hat{\mathcal{X}}_{\text{inl}}$ to the axis.

Calculation of features. The classification function $q(\mathbf{x}, \mathbf{H})$ uses the curve parameters $\mathbf{H}$ and the vector $\mathbf{m}_2(\mathbf{x})$ (5.31) for decision if $\mathbf{x}$ is an inlier or not. We compute the feature vector as

$$\boldsymbol{\omega} = [I(\mathbf{x})\ J(\mathbf{x})\ d(\mathbf{x};\mathbf{H})\ b(\mathbf{x};\mathbf{H})] \tag{5.42}$$

where functions $d(\mathbf{x};\mathbf{H})$ and $b(\mathbf{x};\mathbf{H})$ are explained in the following text. First, a distance to the curve $\mathbf{a}(t,\mathbf{H})$ is computed for each voxel $\mathbf{x}$. The approximative distance $d(\mathbf{x};\mathbf{H})$, is used as explained in Section 4.1.3. A tangent vector of the curve a at point $\mathbf{a}(t_0,\mathbf{H})$ is also required by our tool model

$$\mathbf{a}_t(\mathbf{x},\mathbf{H}) = c\frac{\partial \mathbf{a}}{\partial t}(t_0,\mathbf{H}), \tag{5.43}$$

where c is a normalization constant such that $\|\mathbf{a}_t(\mathbf{x},\mathbf{H})\| = 1$. Finally, a measure of the consistency $b(\mathbf{x},\mathbf{R})$ of the local image structure and the curve's tangent $\mathbf{a}_t(\mathbf{x},\mathbf{H})$ are computed. We have considered two options:

1) Compute the dot product of the tangent vector with the local principal direction $\hat{\mathbf{v}}$ which was obtained by eigenanalysis:

$$b_1(\mathbf{x};\mathbf{H}) = |\hat{\mathbf{v}}(\mathbf{x}) \cdot \mathbf{a}_t(\mathbf{x},\mathbf{H})|\,. \tag{5.44}$$

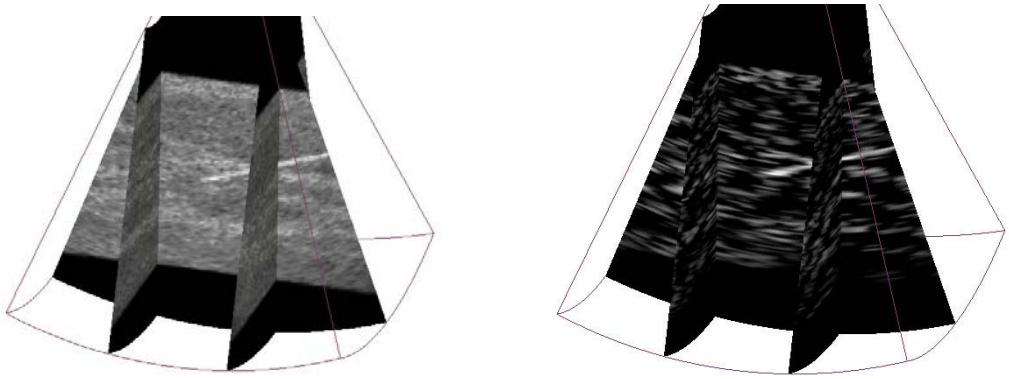
The higher value of b_1 (closer to 1) corresponds to the high consistency of the tangent vector $\mathbf{a}_t$ with the $\hat{\mathbf{v}}(\mathbf{x})$.

2) Apply a line filter ζ_1 or ζ_2 in the direction of the tangent $\mathbf{v} = a_t(\mathbf{x},\mathbf{H})$. Depending on the choice of the equation (5.28) or (5.29) from Section 5.1.3 we have two options:

$$b_2(\mathbf{x};\mathbf{H}) = \zeta_1(\mathbf{x}, a_t(\mathbf{x},\mathbf{H}), \mathbf{K}(\mathbf{x})) \tag{5.45}$$

$$b_3(\mathbf{x};\mathbf{H}) = \zeta_2(\mathbf{x}, a_t(\mathbf{x},\mathbf{H}), \mathbf{K}(\mathbf{x})) \tag{5.46}$$

where $\mathbf{K}(\mathbf{x})$ is the local characteristics, e.g. the Hessian matrix. The consistency of the local structure with the tangent vector is now computed with respect to a local structure tensor.



(a) (b)

Figure 5.1: *Example of the line filtering results: a) original 3D US image simulated using FIELD II, b) the output of line filtering using Frangi's filter.*

Classification function $q(\mathbf{x}, \mathbf{H})$

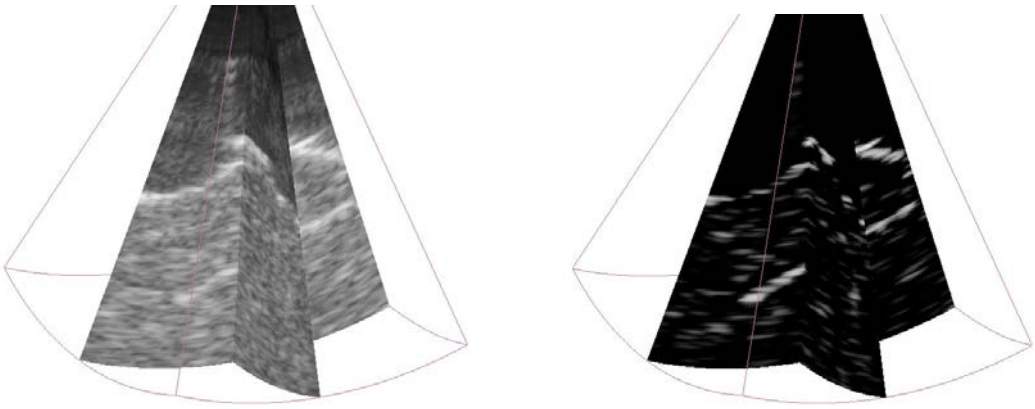
The classification function $q(\mathbf{x}, \mathbf{H})$ uses four features for each voxel $\mathbf{x} \in \mathcal{X}_t$ stored in $\boldsymbol{\omega}(\mathbf{x})$ (5.42) and it is used to evaluate tool hypotheses within the RANSAC procedure (5.40). The classifier $q(\mathbf{x}, \mathbf{H})$ can be thought of as forming a classification cascade together with the pre-segmentation (5.32). Unlike in pre-segmentation, it also depends on the axis parameters $\mathbf{H}$.

The function $q(\mathbf{x}, \mathbf{H})$ can be defined as linear classifier trained by perceptron algorithm, linear classifier trained by the SVM, classifier trained by the SVM with quadratic mapping, cascaded classifier, the AdaBoost or the WaldBoost (Section 5.2.1). The parameters of classifier are learned from the training data with ground truth. We report the results and comparison of all classifiers in Section 5.3.

5.3 Results

We did experiments on various types of data which are similar to data used in previous Chapter 4:

Simulation study. Twenty-eight 3D ultrasound datasets were generated by the ultrasound simulator FIELD II [59] with known ground-truth location of the tool (Figure 5.1a). Numerical phantoms mimic the statistical properties of scatterers of real tissue. Spatial distribution was varied to imitate non-homogeneous response of the breast tissue. An image of tubular tool was created by adding a cylinder of radius 0.4 mm with highly reflecting scatterers. Resolution of each 3D image was



(a) (b)

Figure 5.2: *Example of the line filtering results: a) original 3D US image of the PVA cryogel the phantom acquired with Voluson 530D scanner, b) the output of line filtering using Frangi's method.*

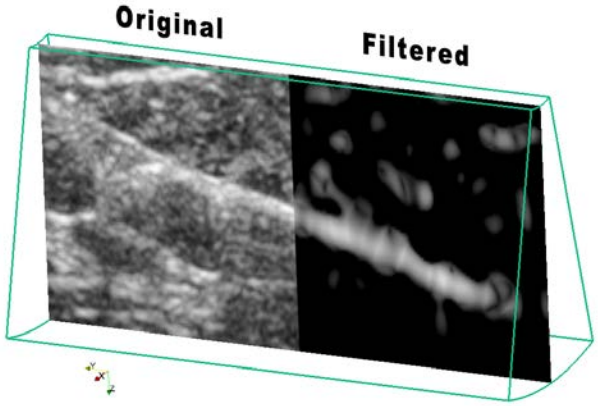


Figure 5.3: *3D volume of breast biopsy with needle. The left part of the slice contains the original data and the right part contains the data filtered by method of Frangi [41]. The original data were acquired with GE Voluson E8 scanner.*

$53 \times 71 \times 160$ voxels.

PVA phantom. Experiments have been done on a PVA cryogel phantom [31, 45] which mimics properties of biological tissue. The PVA cryogel phantom contained an electrode of diameter 0.3 mm (Figure 5.2a). Eight 3D ultrasound images of size $53 \times 71 \times 260$ voxels of the PVA cryogel phantom from various positions have been acquired using the ultrasound scanner Voluson 530D with a probe with central frequency 7.5 MHz.

Breast biopsy. We tested the method on three data sets of the breast tissue (Figure 5.3) with a straight biopsy needle of 11-gauge (1 mm outer diameter). 3D ultrasound images were acquired by the GE Voluson E8 scanner with a probe of 12 MHz. Volume data sets were of dimension $273 \times 383 \times 208$ voxels.

The ground-truth for real data sets has been determined as an average location labelled by eight experts. They were asked to find the tool axis in thresholded set of voxels by fitting the best approximating line. The mean variability for human observers was less than 0.4 mm. Assuming the ground-truth location of the tool to be known, voxels of the volume have been divided into two groups: $\mathcal{X}_{\text{tool}}$ (tool voxels) and $\mathcal{X}_{\text{bg}}$ (background voxels). Radius of the tool has been determined on the training data set as a mean of standard deviations of distances to the ground-truth location of the tool in thresholded data.

The evaluation of classifiers is performed only on simulated data. The reason is that we have the precise ground-truth location of the tool only for the simulated data. The results of classification on real data is similar, but such a high accuracy of evaluation (as for simulated data) cannot be guaranteed. The final localization results are shown on all types of data.

Implementation. Line filtering methods and proposed localization method were implemented in MATLAB (The MathWorks, Natick, MA) and tested on PC with Intel Core 4 processor at 2.83 GHz. The Hessian matrices were computed using a code by Almar Klein[1]. The code estimates partial derivatives of any order on multi-dimensional data by convolution with derivative of the Gaussian. For the fast training of the SVM on large datasets, the library LIBOCAS [40] was used. For the training and evaluation of the AdaBoost, the implementation from the CMP Vision algorithms [92] was used. The implementation of Waldboost with domain partitioning weak classifiers by Jan Šochman was used. The number of weak classifiers was set to five times the number of features, i.e., the number was 10 for classifier h in the pre-segmentation and 20 for the classifier q in the RANSAC. The desired TP rate and FN rate, for the training of the AdaBoost and the WaldBoost, was set to 99%. For weak hypotheses of the AdaBoost, we used randomly generated linear classifiers. Domain-partitioning weak classifiers with eight bins were used in the WaldBoost.

[1] http://www.mathworks.com/matlabcentral/fileexchange/19696

	Frangi's filter		Sato's filter		Li's filter	
Datasets (number)	Ψ^{Impr}	Time	Ψ^{Impr}	Time	Ψ^{Impr}	Time
Simulation (28)	**3.14**	10.64 s	1.96	10.53 s	0.65	10.31 s
PVA phantom (8)	**16.78**	18.80 s	4.04	18.61 s	3.12	18.27 s
Breast tissue (3)	**4.54**	236.7 s	2.29	213.1 s	1.89	201.4 s

Table 5.1: *Measured results of the mean contrast improvement using the mean contrast ratio* $\Psi^{\text{Impr}}(I_f, I_o, \mathcal{X}_{\text{tl}}, \mathcal{X}_{\text{bg}})$ *and the mean elapsed time. The Frangi's filter outperformed the others and its column is marked by bold-face. The time of filtering increased with the size of volume.*

5.3.1 Enhancement evaluation

According to an expert, the images after line filtering by Frangi's filter were better, the improved visual contrast helped identify the tool easier. We have also evaluated the enhancement of the linear structures using a quantitative method. We used the contrast of tool voxels $\mathcal{X}_{tl}$ and background voxels $\mathcal{X}_{bg} = \Omega \setminus \mathcal{X}_{tl}$. The contrast Ψ was measured using the *contrast ratio*:

$$\Psi(I, \mathcal{X}_{\text{tl}}, \mathcal{X}_{\text{bg}}) = \frac{\text{mean}\,(I(\mathcal{X}_{\text{tl}}))}{\text{mean}\,(I(\mathcal{X}_{\text{bg}}))} \tag{5.47}$$

Let us assume that the original image is I_o and the filtered image is I_f. We measure the improvement of contrast by computing the ratio:

$$\Psi^{\text{Impr}}(I_f, I_o, \mathcal{X}_{\text{tl}}, \mathcal{X}_{\text{bg}}) = \frac{\Psi(I_f, \mathcal{X}_{\text{tl}}, \mathcal{X}_{\text{bg}})}{\Psi(I_o, \mathcal{X}_{\text{tl}}, \mathcal{X}_{\text{bg}})} \tag{5.48}$$

Results of enhancement. We performed experiments on all three types of datasets mentioned earlier: twenty-eight simulated datasets ($53 \times 71 \times 164$ voxels), eight PVA phantom datasets ($53 \times 71 \times 310$ voxels) and three datasets of breast tissue ($383 \times 273 \times 208$ voxels). We evaluated the mean contrast improvement Ψ^{Impr} using three types of filters: Frangi, Sato and Li. The processing time was approximately the same for all three types of filters and it increased with the volume size and with the size of convolution mask for partial derivative estimation. The results measured by the contrast ratio are in Table 5.1. The images that were filtered with Frangi's method have the best improvement of the contrast Ψ. Curiously, for the Li's filtering method, the contrast decreased on simulated data (marked by waved underline) because of some difficult configurations of the tool in simulated data mentioned

earlier.

5.3.2 Pre-segmentation evaluation

The task in pre-segmentation is to classify voxels into tool and background using two attributes: intensity and tubularness (Figure 5.4). We have evaluated the precision of classifiers described in Section 5.2.1.

Classifier $h(\mathbf{x})$ evaluation. The predicted segmentation $\hat{\mathcal{X}}_{tl}$ and $\hat{\mathcal{X}}_{bg}$ is compared to the ground-truth sets $\mathcal{X}_{tl}$ and $\mathcal{X}_{bg}$. Table 5.2 summaries the naming convention in the confusion matrix.

Confusion matrix	negative predicted $\hat{\mathcal{X}}_{bg}$	positive predicted $\hat{\mathcal{X}}_{tl}$
actual negatives $\mathcal{X}_{bg}$	True Negative (TN)	False Positive (FP)
actual positives $\mathcal{X}_{tl}$	False Negative (FN)	True Positive (TP)

Table 5.2: *Confusion matrix illustrates the naming convention for the evaluation of predicted class with respect to actual class.*

The basic measures to be evaluated are a TP rate (sensitivity) and a FP rate (1-specificity):

$$\text{TP rate} = \frac{\text{TP}}{\text{FN} + \text{TP}} \tag{5.49}$$

$$\text{FP rate} = \frac{\text{FP}}{\text{TN} + \text{FP}} \tag{5.50}$$

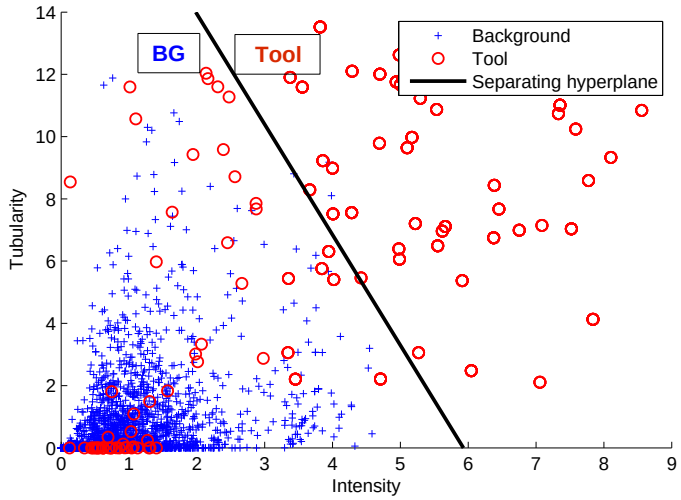
We primarily want to maximize the inliers ratio (Section 4.1.4). It is also called a *precision* and it is defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5.51}$$

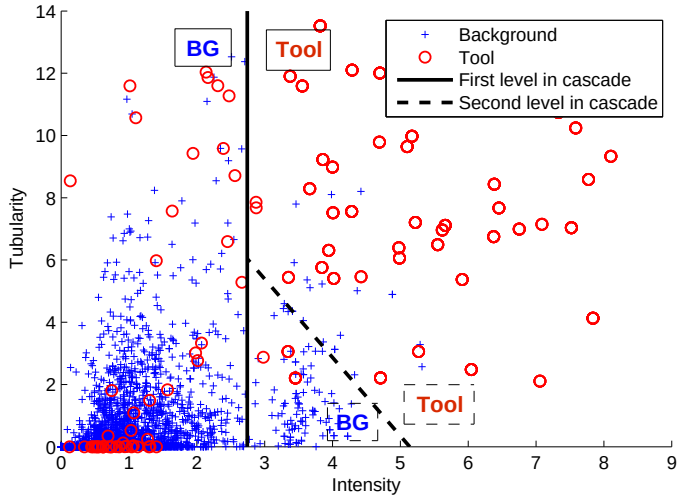
The performance of the RANSAC procedure directly depends on it. The *accuracy* (also called the error rate) is also used in this chapter and it is defined as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{5.52}$$

The sensitivity, specificity and accuracy are informative measures which provide an information complementary to the precision.



(a)



(b)

Figure 5.4: *Feature space composed of two attributes for each voxel: intensity* $I(\mathbf{x})$ *and tubularness* $J(\mathbf{x})$*. (a) monolithic linear classifier found by the perceptron learning; (b) cascaded classifier composed of thresholding and followed by linear classifier.*

Type of Classifier	**Precision** [%]	Sens. [%]	Spec. [%]	Train [s]	Test [$\cdot 10^{-3}$ s]	Filter [s]
Thresholding	66.4	77.5	80	6.91	5.41	0
Linear (Perc.)	85.2	89.5	80	8.71	5.76	10.64
SVM (Lin.)	92.4	91.8	80	1.13	3.42	10.64
SVM2 (+Qm)	**94.5**	92.0	80	3.44	5.36	10.64
AdaBoost	91.8	91.0	80	*142.7*	*81.83*	10.64
WaldBoost	87.6	90.6	80	*3.46*	*345.16*	10.64
Cascade	89.6	83.7	80	3.45	3.82	**0.98**

Table 5.3: *Results of training of pre-segmentation on simulated data. For each classifier, we report the precision (inliers ratio), sensitivity (TP rate), specificity (1 - FP), training time (on* $472 \cdot 10^3$ *samples) and testing time (on* $118 \cdot 10^3$ *samples). The best achieved precision and the best achieved time using line filtering is marked by bold face. The times of training and testing of AdaBoost and WaldBoost (in italics) are higher because they were tested in MATLAB. The time required for line filtering is reported separately.*

Results on testing data. We compared the classifiers by evaluating the precision (inliers ratio), the FP rate and the TP rate, where our primary objective is to maximize the precision. The training set contained $472 \cdot 10^3$ samples made of 9 simulated datasets. Testing set (different from training set) contained $118 \cdot 10^3$ samples. The results on the testing set are reported in Table 5.3. Classifiers were adjusted by looking at the ROC curve so that specificities are 80% (Section 5.3.2) to make sensitivities comparable. The results show that the precision is higher when using the line filtered data, e.g. 85.2% for linear classifier which is in contrast with 66.4% for the simple thresholding. The precision was around 90% for the cascaded classifier, the AdaBoost or the WaldBoost. The highest precision 94.5% was achieved with the non-linear SVM classifier. The times of training and testing of AdaBoost and WaldBoost are higher because they were tested in MATLAB. It is fast enough for our application, but better time can be achieved by re-implementation, e.g. in C.

The time required for filtering is reported in Table 5.4. The time is similar for all three types of Hessian based methods (Frangi's, Li's and Sato's). The line fitlering can be evaluated efficiently in the cascaded classifier (Section 5.2.1). The filter is evaluated only on voxels surviving the first level thresholding and the second level classifier uses also tubularity feature. Using the fast version of line filtering in cascade, we succeeded to reduce the overall time to 10%. The filtering time around 1 s allows the faster implementation of the tool localization. We propose to use

Datasets	Size [voxels]	Full volume	**in Cascade**
Simulation	$53 \times 71 \times 164$	10.64 s	**0.98 s**
PVA phantom	$53 \times 71 \times 310$	18.80 s	**2.05 s**
Breast tissue	$383 \times 273 \times 208$	236.7 s	**14.04 s**

Table 5.4: *Elapsed times for line filtering using Frangi's method of various datasets. The time is given for filtering of the full volume and for filtering of the thresholded data in a cascade. The overall time depends linearly on the size of dataset.*

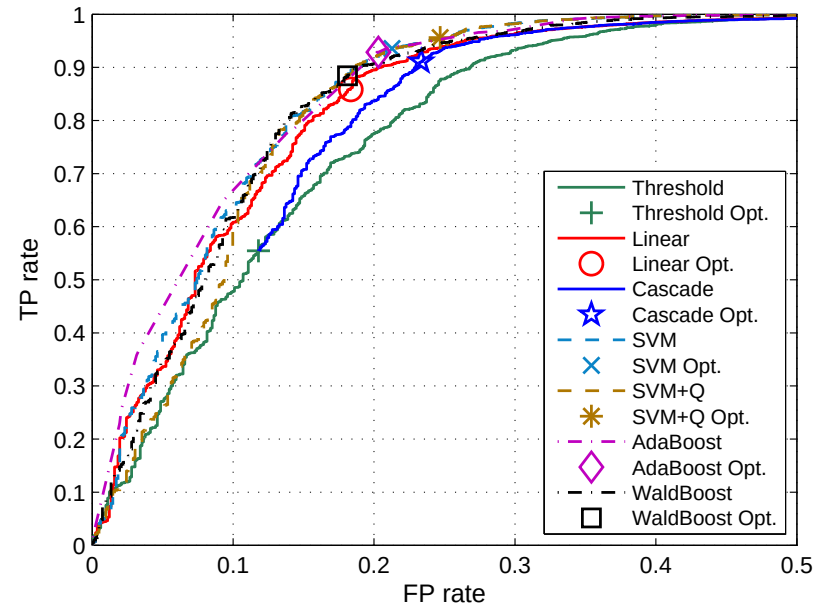
the cascaded classifier because the time of evaluation is lower than for the other classifiers.

ROC curve analysis

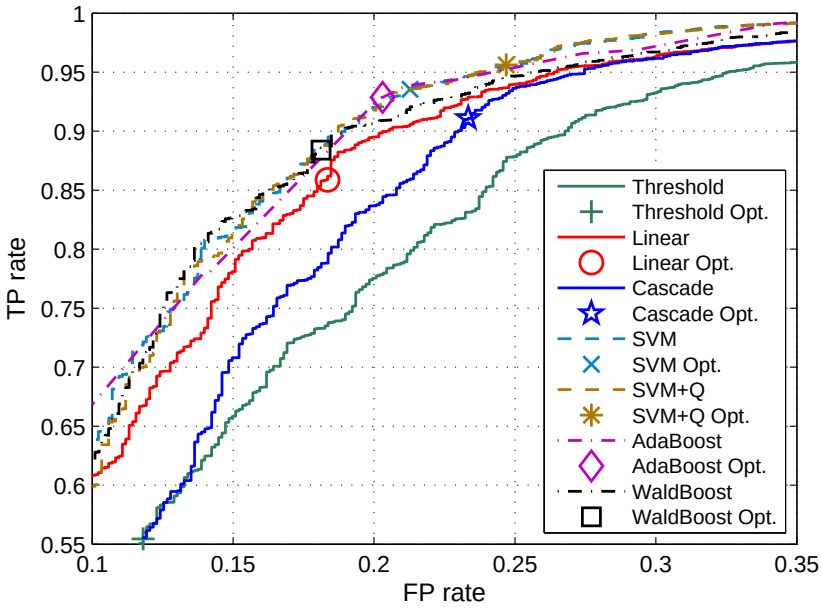
We evaluated the quality of the pre-segmentation (Section 5.2.1) using receiver operating characteristics (ROC) curve [33]. We draw the dependency of the TP rate on the FP rate. The ideal ROC curve has low false alarm rate and high true positive rate. We note that the ROC curve provide only a complementary information and the primary criterion for choosing the best classifier is precision as explained previously.

We do the ROC curve computation by varying the bias of the linear classifier (trained by perceptron or by the SVM) in the range of all possible classifications. The ROC curve for the AdaBoost was plotted by connecting the points corresponding to the weak classifiers. The ROC curve for the WaldBoost was computed by varying the decision threshold. We show the ROC curves in Fig. 5.5 for the testing set. The optimal found classifiers are shown by point marks and the curve represent the result of varying classifier. For convenient comparison, we have set the bias for each classifier so that the FP rate is 20% and we observed the TP rate. The results obtained using this technique are given in Table 5.3. The computed ROC curve shows small differences between tested classifiers. The linear and the cascaded classifier were performing better than the thresholding. The highest TP rate for the varying FP rate around 20% achieved the SVM-based classifiers.

A similar adjustment technique for the classifier $q()$, including the computation of the ROC curve, is performed in the following section and the results are in Table 5.6.



(a)



(b)

Figure 5.5: *The ROC curves for various pre-segmentation classifiers: thresholding, linear classifier, cascade, the SVM, the AdaBoost and the WaldBoost. The optimal classifiers found by learning algorithms are marked by point symbols in legend. The ROC curves were by varying the decision threshold from the optimal classifiers. There are two views of the same ROC curve with a different range of TP and FP.*

feature	$I(\mathbf{x})$	$J(\mathbf{x})$	$d(\mathbf{x};\mathbf{H})$	$b_1(\mathbf{x};\mathbf{H})$	$b_2(\mathbf{x};\mathbf{H})$	$b_3(\mathbf{x};\mathbf{H})$
$\Sigma_{\text{LDA}}(\cdot)$	13.9	32.9	35.4	13.5	16.4	11.9
$\Sigma_{\text{accuracy}}(\cdot)$	56.9%	76.7%	98.5%	56.4%	65.0%	46.2%

Table 5.5: *Analysis of features discriminativity using the LDA.*

5.3.3 Tool model evaluation

In this Section, we evaluate the proposed tool model. First, we do feature analysis to determine good features used later in the classification. Then, we examine the classification function $q(\mathbf{x},\mathbf{H})$ (from Section 5.2.2) which is used in tool localization.

Feature analysis

We have proposed to use several features in vectors $\boldsymbol{\omega}(\mathbf{x})$. The most important one is distance from tool axis $d(\mathbf{x},\mathbf{H})$. We shall also compare two ways for computing the consistency of local vector and the curve direction: by dot product $b_1(\mathbf{x};\mathbf{H})$ (5.44) and by oriented line filter $b_2(\mathbf{x};\mathbf{H})$ (5.45) or $b_3(\mathbf{x};\mathbf{H})$ (5.46).

We measure the discriminative power of features by two measures: 1) Linear discriminant analysis (LDA) which measures the ratio of between-class and within-class variance:

$$\Sigma_{\text{LDA}}(F) = \frac{\sigma^2_{\text{between}}}{\sigma^2_{\text{within}}} \tag{5.53}$$

2) Classification accuracy $\Sigma_{\text{accuracy}}(F)$ (5.52) of a linear trained by FLD [33] for each evaluated feature.

The results of evaluation of Σ_{LDA} and Σ_{accuracy} in Table 5.5 show that feature $b_2(\mathbf{x};\mathbf{H})$ is more discriminative than $b_1(\mathbf{x};\mathbf{H})$ or $b_3(\mathbf{x};\mathbf{H})$, therefore we use the oriented line filter ζ_1 (5.45) in our tool model.

Evaluation of classifier

The training of the classifier $q(\mathbf{x},\mathbf{H})$ is done on datasets separated from the testing set. We compute the vector $\boldsymbol{\omega}(\mathbf{x})$ (5.42) for each voxel $\mathbf{x}$. The ground-truth label is assigned according to distance from the ground-truth location. If $d(\mathbf{x};\mathbf{H}) \geq r$ (where r is radius of tool) then "1" is assigned, otherwise "0" is assigned.

The classifier $q(\mathbf{x};\mathbf{H})$ is used for the robust RANSAC estimation of the tool axis by maximization of the number of inliers. Note that the input data might be contaminated by outliers, therefore we need a classifier robust to large spatial variations of tool axis. We compared all classifiers from Section 5.2.1. We report

Type of classifier	**Precision** [%]	Sens. [%]	Spec. [%]	Train [s]	Test [10^{-3} s]
Linear (Perc.)	94.97	96.11	98	0.10	0.57
SVM (Lin.)	96.99	99.98	98	0.38	0.74
SVM2 (+Q.)	**98.12**	99.99	98	0.99	1.08
AdaBoost	**98.18**	99.29	98	*19.30*	*7.70*
WaldBoost	96.77	98.28	98	*1.68*	*42.37*

Table 5.6: *Results of training of classifier* $q(\mathbf{x};\mathbf{H})$ *on simulated data. For each classifier (with or without quadratic mapping), there are various values reported: precision (inliers ratio), Sensitivity (TP rate), Specificity (1 - FP rate), training time (on* $30 \cdot 10^3$ *samples) and testing time (on* $7.6 \cdot 10^3$ *samples). The best achieved precision is marked by bold face. The times of training and testing of AdaBoost and WaldBoost (in italics) are higher because they were tested in MATLAB.*

results (the precision, the TP rate and the FP rate) in Table 5.6. Classifiers were adjusted by looking at the ROC curve (using a technique from Section 5.3.2) so that specificities are 98%. The best performing classifiers (with the precision over 99%) were the non-linear SVM and the AdaBoost, but the difference between them is not significant.

5.3.4 Localization evaluation

In this section, the proposed localization method is evaluated using the axis accuracy, the number of failures and the elapsed time. We evaluate the axis accuracy ϵ_{axis} as in Chapter 4 (Section 4.2.1). The result is considered to be a failure when the axis accuracy is $\epsilon_{axis} > 3$ mm. The number of failures are reported separately and the accuracy is evaluated only on the successful runs. The reported times include line filtering and pre-processing.

The proposed method is examined with two variations in pre-segmentation: 1) with a cascaded classifier (referred as CASC) and 2) with a classifier learned by the SVM with quadratic mapping (SVM2). The proposed method is examined with four variations of classifier $q()$ in the RANSAC procedure: 1) the basic SVM classifier (SVM); 2) the SVM with quadratic mapping (SVM2); 3) the AdaBoost with weak linear classifier (ADA); 4) the WaldBoost with domain partitioning weak classifiers (WALD). In total we examined 8 variations of the proposed method on each type of data.

The proposed method with line filtering (5.2.2) is also compared to previously described localization methods based on model fitting using the RANSAC with

No. #	Method	Pre-segm. + Tool model	Time [sec]	Fails [%]	Acc. [mm]
1	PIP	None	69.1	5.0 ± 4.7	0.28
2	MR-PIP	None	29.4	11.0 ± 9.8	0.27
3	RHT	Thresholding	1.38	14.5 ± 12.4	0.36
4	Q-RHT	Thresholding	0.89	28.0 ± 20.1	1.41
5	RANSAC	Thrsh. + AxShp	0.41	8.6 ± 7.8	0.14
6	RANSAC	Thrsh. + IntDstr	1.00	1.4 ± 1.4	0.38
7	RANSAC	Casc. + SVM	1.05	**0 ± 0**	0.16
8	RANSAC	Casc. + SVM2	1.07	**0 ± 0**	0.17
9	RANSAC	Casc. + ADA	1.22	**0 ± 0**	0.13
10	RANSAC	Casc. + WALD	2.05	**0 ± 0**	0.23
11	RANSAC	SVM2 + SVM	7.61	**0 ± 0**	0.18
12	RANSAC	SVM2 + SVM2	7.78	**0 ± 0**	0.11
13	RANSAC	SVM2 + ADA	8.57	**0 ± 0**	0.12
14	RANSAC	SVM2 + WALD	9.26	**0 ± 0**	0.18

Table 5.7: *The results of the tool localization on simulated data with the high SNR of the tool.*

models AxShp and IntDstr [P1] (Chapter 4) and also with the PIP [P2], the MR-PIP [P3] (Chapter 3), the RHT and the Q-RHT [108]. The tool models were learned as described in Chapter 4 (Section 4.1.3). The settings for the RHT and the QRHT were the same as in previous Chapter (Section 4.2).

Simulation study

We have compared various tool localization methods on the simulated data with two levels of difficulty: the first group contains datasets with a relatively high contrast of the tool (also referred as the SNR); the second group contains datasets with the SNR decreased by 50% compared to the first group.

The results for the first group are reported in Table 5.7 and example of the input 3D volume is in Figure 5.1a. The methods based on the PIP (rows 1–2) are taking longer than 10 seconds. The RHT-based methods (rows 3–4) are worse (in terms of the robustness) than the RANSAC localization with the model AxShp and IntDstr (rows 5–6). The methods which are using line filtering (rows 7–14) perform better than the others but they require more time for pre-processing. We have successfully tested the cascaded classifier (rows 7–10) which has reduced the filtering time compared to the SVM with quadratic mapping (rows 11–14). All

No. #	Method	Pre-segm. + Tool model	Time [sec]	Fails [%]	Acc. [mm]
1	PIP	None	70.8	100 ± 0	N.A.
2	MR-PIP	None	31.2	100 ± 0	N.A.
3	RHT	Thresholding	5.91	100 ± 0	N.A.
4	Q-RHT	Thresholding	4.08	100 ± 0	N.A.
5	RANSAC	Thrsh. +AxShp	9.8	100 ± 0	N.A.
6	RANSAC	Thrsh. +IntDstr	11.2	100 ± 0	N.A.
7	RANSAC	Casc. + SVM	18.3	44.3 ± 24.7	0.27
8	RANSAC	Casc. + SVM2	23.4	17.1 ± 14.2	0.32
9	RANSAC	Casc. + ADA	26.9	95.7 ± 4.1	0.51
10	RANSAC	Casc. + WALD	22.6	95.7 ± 4.1	0.58
11	RANSAC	SVM2 + SVM	39.6	17.1 ± 14.2	0.36
12	RANSAC	SVM2 + SVM2	41.2	**1.4 ± 1.4**	0.16
13	RANSAC	SVM2 + ADA	39.1	15.7 ± 13.2	0.26
14	RANSAC	SVM2 + WALD	54.6	67.1 ± 22.1	0.41

Table 5.8: *The results of the tool localization on simulated data with the low SNR of the tool.*

variants of proposed method had zero number of failures which was not be achieved by the previous methods without line filtering.

We tested the localization methods on more difficult data in order to show differences between their performance. The results for the case with a low SNR of the tool are in Table 5.8 and example of localization result is in Figure 5.6. The methods, which are not using the line filtering (rows 1–6), are failing in the most cases. The methods with the SVM classifier in pre-segmentation achieved the best performance. The classifier $q()$ based on the SVM with quadratic mapping (row 12) had least number of failures. The cascaded classifier in pre-segmentation decreased the overall time at the cost of increasing the number of failures.

Experiments on real data

The results on real data of the PVA phantom are in Table 5.9 and example of localization result is in Figure 5.7. The methods based on the PIP (rows 1–2) take longer time than 30 seconds and they are not reliable with 50% of failures. The RHT and the QRHT (rows 3–4) mostly fail because of the presence of the 2D layer which confuses them. The same problem had the model fitting with the AxShp and the IntDstr (rows 5–6). The robustness of tested methods was improved when using

No. #	Method	Pre-segm. + Tool model	Time [sec]	Fails [%]	Acc. [mm]
1	PIP	None	126.8	50 ± 25	0.34
2	MR-PIP	None	34.2	50 ± 25	0.50
3	RHT	Thresholding	3.81	88 ± 10.5	0.74
4	Q-RHT	Thresholding	2.07	97 ± 2.9	1.52
5	RANSAC	Thrsh. + AxShp	0.37	97.5 ± 2.4	0.69
6	RANSAC	Thrsh. + InDstr	1.34	91.3 ± 7.9	0.75
7	RANSAC	Casc. + SVM	3.83	2.5 ± 2.4	0.61
8	RANSAC	Casc. + SVM2	4.45	10.0 ± 9	0.72
9	RANSAC	Casc. + ADA	5.30	10.0 ± 9	0.80
10	RANSAC	Casc. + WALD	6.00	12.5 ± 10.9	0.83
11	RANSAC	SVM2 + SVM	16.21	7.5 ± 6.9	0.83
12	RANSAC	SVM2 + SVM2	17.12	**2.5 ± 2.4**	0.68
13	RANSAC	SVM2 + ADA	17.24	**2.5 ± 2.5**	0.68
14	RANSAC	SVM2 + WALD	17.58	40.0 ± 24	0.89

Table 5.9: *The results of tool localization on the PVA cryogel phantom.*

No. #	Method	Pre-segm. + Tool model	Time [sec]	Fails [%]	Acc. [mm]
1	PIP	None	619	100 ± 0	N.A.
2	MR-PIP	None	198	100 ± 0	N.A.
3	RHT	Thresholding	14.31	100 ± 0	N.A.
4	Q-RHT	Thresholding	5.48	100 ± 0	N.A.
5	RANSAC	Thresh. + AxShp	2.21	100 ± 0	N.A.
6	RANSAC	Thresh. + IntDstr	8.21	100 ± 0	N.A.
7	RANSAC	Cascade + SVM	48.6	80 ± 16	0.314
8	RANSAC	Cascade + SVM2	53.1	66.7 ± 22.2	0.168
9	RANSAC	Cascade + ADA	52.6	66.7 ± 22.3	0.174
10	RANSAC	Cascade + WALD	52.6	100.0 ± 0.0	N.A.
11	RANSAC	SVM2 + SVM	272.7	53.3 ± 24.9	0.098
12	RANSAC	SVM2 + SVM2	273.8	6.7 ± 6.2	0.098
13	RANSAC	SVM2 + ADA	280.9	**0.0 ± 0.0**	0.132
14	RANSAC	SVM2 + WALD	276.9	**0.0 ± 0.0**	0.198

Table 5.10: *The results of tool localization on real data of breast biopsy.*

line filtering (rows 7–14). The failure rate decreased to below 10% which makes the localization method more reliable. The best performance was achieved with the SVM classifier with quadratic mapping in pre-segmentation and in in the RANSAC procedure (rows 8). Similar failure rate was achieved with the AdaBoost classifier. The time of localization can be reduced by employing the cascaded classifier in the pre-segmentation.

The results on real data of breast tissue with the biopsy needle are in Table 5.10. The methods, which are not based on the line filtering (rows 1–6), completely fail on this data. The line filtering helped with detection of the tool in the biological tissue. The proposed methods (rows 7–14) achieve less failure rate. The combination of the SVM classifier and the AdaBoost or the WaldBoost in the tool model achieves zero failures on the three data sets that were used in the experiment.

The elapsed times were higher because of the volume size ($273 \times 383 \times 208$ voxels) but it was successfully reduced from 246 s to 35 s using the cascaded classifier.

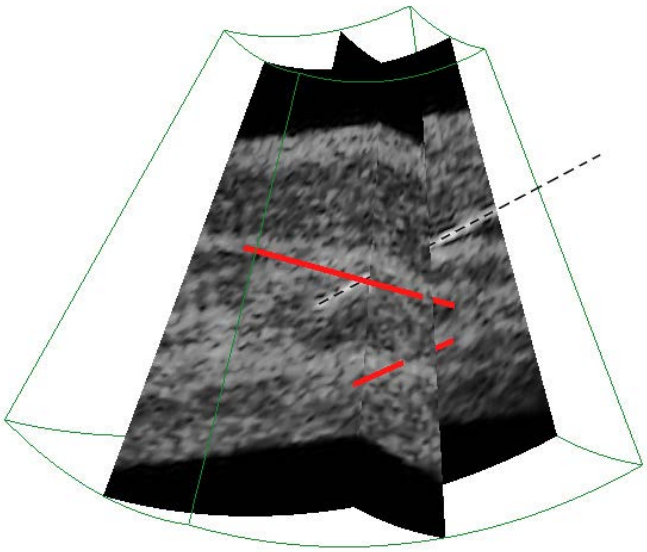
5.4 Conclusions

We proposed the method for tool localization in 3D ultrasound images which exploits its 1D shape. The tool contrast has been enhanced by line filtering methods based on eigenanalysis of the Hessian matrix. We chose the Frangi's method because it provides the best enhancement of the tool contrast.

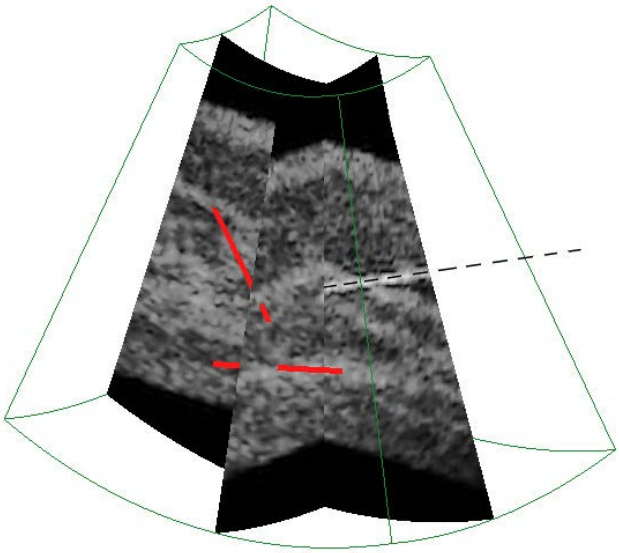
The localization method proposed in this chapter improves the previous model fitting method because of two properties: 1) the line filtering reduces the number of background voxels, so the fraction of tool voxels is higher after pre-segmentation; 2) the classifier in the RANSAC procedure uses additional features pre-computed by line fitlering which makes it more robust. The best performing combination is the cascaded classifier in the pre-segmentation (for fast evaluation of the line filtering) and the SVM classifier in the RANSAC procedure (for the low number of failures).

Additional voxel descriptors could be used for improved robustness (Section 5.2), e.g., the raw acoustic data from the US scanner. We have used the cost function based on the SSD (sum of squared distances) for the local optimization. We shall propose in the future a cost function which corresponds to classifiers described in this chapter.

The line filtering requires an additional pre-processing time (around 10 s for the volume of $53 \times 71 \times 160$ voxels). The time was significantly reduced by employing the cascade classifier to 1 s.

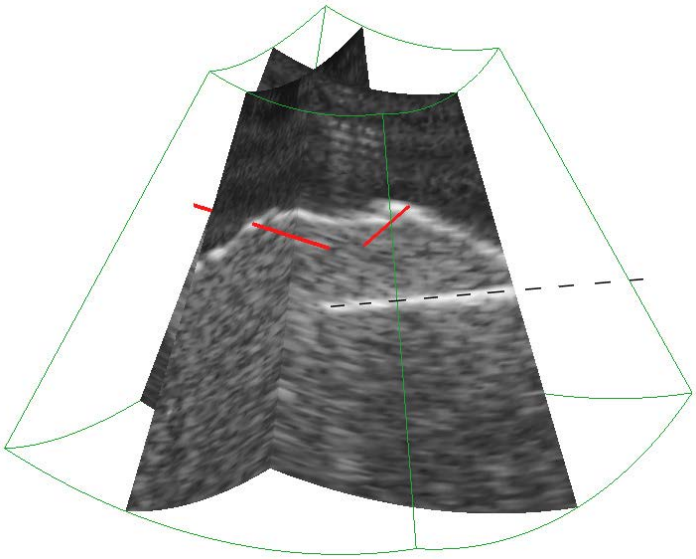


(a)

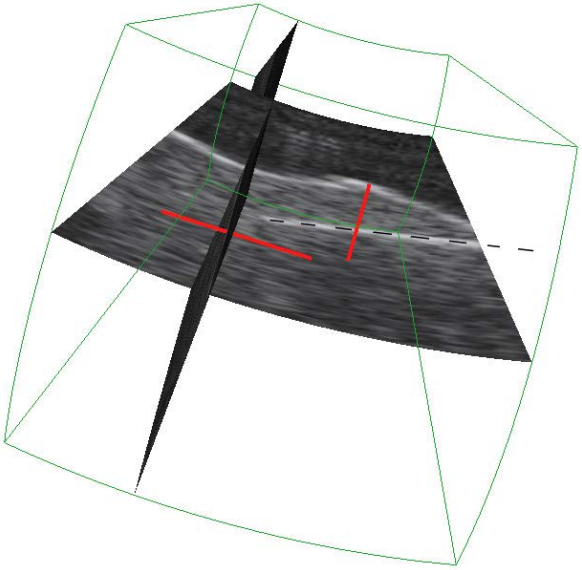


(b)

Figure 5.6: *Visualization of the localization results in simulated 3D ultrasound data. The dashed line shows the correct result of tool localization with line filtering. Two red lines show examples of failure without line filtering. Complete statistics and comparison of various methods can be found in Table 5.7. Two different views (a) and (b) show the same data set for better understanding of the 3D scene. Another 3D visualization in a movie is on the accompanied CD in file* `/Viz3D/animation-simData.avi`.



(a)



(b)

Figure 5.7: *Visualization of the localization results in 3D ultrasound data of the PVA phantom. The dashed line shows the correct result of tool localization with line filtering. Two short red lines show examples of failure without line filtering. The localization method got confused by the high intensity layer caused by the border of the phantom. Complete statistics can be found in Table 5.9. Two different views (a) and (b) show the same data set. Another 3D visualization in a movie is on the accompanied CD in file* `/Viz3D/animation-pvaData.avi`.

Chapter 6

Real-time localization

In this chapter, we propose to assist the surgeon by providing a real-time application that takes the 3D data from the ultrasound scanner, automatically localizes the surgical tool and displays the section plane containing the tool. The axis of the tool is estimated with a method based on model fitting using a Random Sample Consensus (RANSAC) as described in Chapter 4.

The work presented here was presented on the *IEEE International Ultrasonics Symposium* (IUS) 2010 [P5]. The author made a significant contribution to the software presented here while working on it with various master students at CREATIS over two years.

6.1 Method

The algorithm that was used to localize the surgical tool from 3D data is presented in Section 6.1.1. The design of real-time application, overall functioning and implementation in the C++ is given in Section 6.1.2 which is a contribution of this work.

6.1.1 Localization algorithm

The tool localization algorithm based on the model fitting [P1] (Chapter 4) is used. The method has been tested offline and has been validated in various configurations. We implemented the simple tool model based on evaluation of distance to the tool axis. We do not perform the local optimization because we aim at demonstrating the real-time feasibility of the method.

6.1.2 Proposed application

The application is implemented on the Ultrasonix Sonix RP with a CPU speed of 3 GHz and the Microsoft Windows XP operating system. The 3D probe 4DC7-3/40 was used that allows the acquisition of echographic volumes by built-in motorized sweeping of transducers. Three frequencies are available with this probe for the acquisition of B-mode: 2.5, 4.0 and 5.0 MHz. It had an array length of 55 mm and a maximum Field of View (FOV) of 104°.

The demonstration application is built from the open source application Propello provided by Ultrasonix for custom applications. It uses the Porta library (version 4.6.2) which is an interface for communication with the Ultrasonix scanner and for controlling of 3D probe motor. Propello was built-up in the C++ language using the Microsoft Foundation Class (MFC 7.1). The library allows, among others, multi-threading which is used to create a real-time application. The program source code is provided for the Visual Studio .NET environment.

The tool localization algorithm (model fitting using the RANSAC procedure) was implemented in the C++ and it is provided as an open-source static library under the GNU GPL licence. The GNU Scientific library (GSL) version 1.11, an open-source library for scientific computing, was used for mathematical functions. In addition to the localization method (Chapter 4), there is also a function for automatic estimation of the threshold. This function finds such a threshold (from range between 0 and 255) that returns a given portion of voxels over threshold. There are other important parameters of the tool localization: the minimal length of the tool, the maximum radius of the cylinder, the desired probability of the solution and the maximal number of iterations.

Functions and features

Propello allows the acquisition of frame data from motorized 3D transducers. Various types of data can be collected (pre-scan and post-scan converted B-mode data, raw RF data, etc.). With the Propello program, users have full control over the motor. There are two types of capturing method: manual (i.e. step by step) and automatic mode which was used here.

After initialization, the user can modify various parameters for the acquisition, e.g. the number of frames per volume, or the step (in degrees) between frames. After starting the automatic acquisition, all the acquired frames are stored in a cyclic buffer of a fixed size (Figure 6.1) according to the principle first-in, first-out (FIFO).

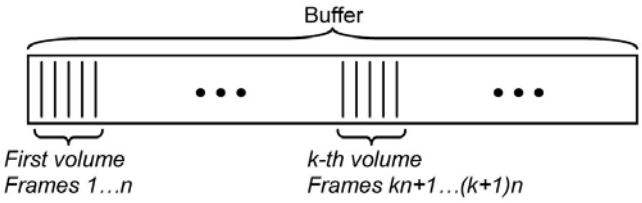


Figure 6.1: *Representation of the cyclic buffer which contains acquired volumes.*

With the original application provided by the Ultrasonix, it is not possible to know when an entire volume is acquired. We implemented the missing feature by testing the number of frames already acquired. Once a full volume has been obtained, it is ready to be processed by the tool localization function. See also a detailed description of the procedure in the next section 6.1.2.

An offline mode has also been implemented. In this mode, user can replay the acquired sequence and the tool localization from the memory or from a file. The feature of replaying from file allows the application to run on any computer without the Ultrasonix computer and the 3D probe connected.

Software architecture

We use a multi-threading architecture which allows running three various tasks in parallel:

- Main Window - for user interaction and displaying of the result of localization.
- Acquisition - for automatic acquisition of frames in the Porta library.
- Worker Thread - for tool localization (using our C++ library).

The synchronisation between them is done by calling functions and exchanging messages (Figure 6.1.2). The acquisition thread consists of: (i) checking if a correct probe (i.e. 3D probe) is connected; (ii) automatically moving the motor using parameters (e.g. spacing between frames and the number of frames) supplied by the user, (iii) acquiring and storing frames in the cyclic buffer as discussed above.

Whenever a new frame is acquired, a callback function from Porta is called and handled in the main dialog (Step 1). Once a full volume has been obtained (Step 2), we test if the worker thread has finished the processing of the previous volume (Step 3). If finished, the main window wakes up the worker thread (Step 4) which works on a copy of the last acquired volume. We ensure that the volume data is not

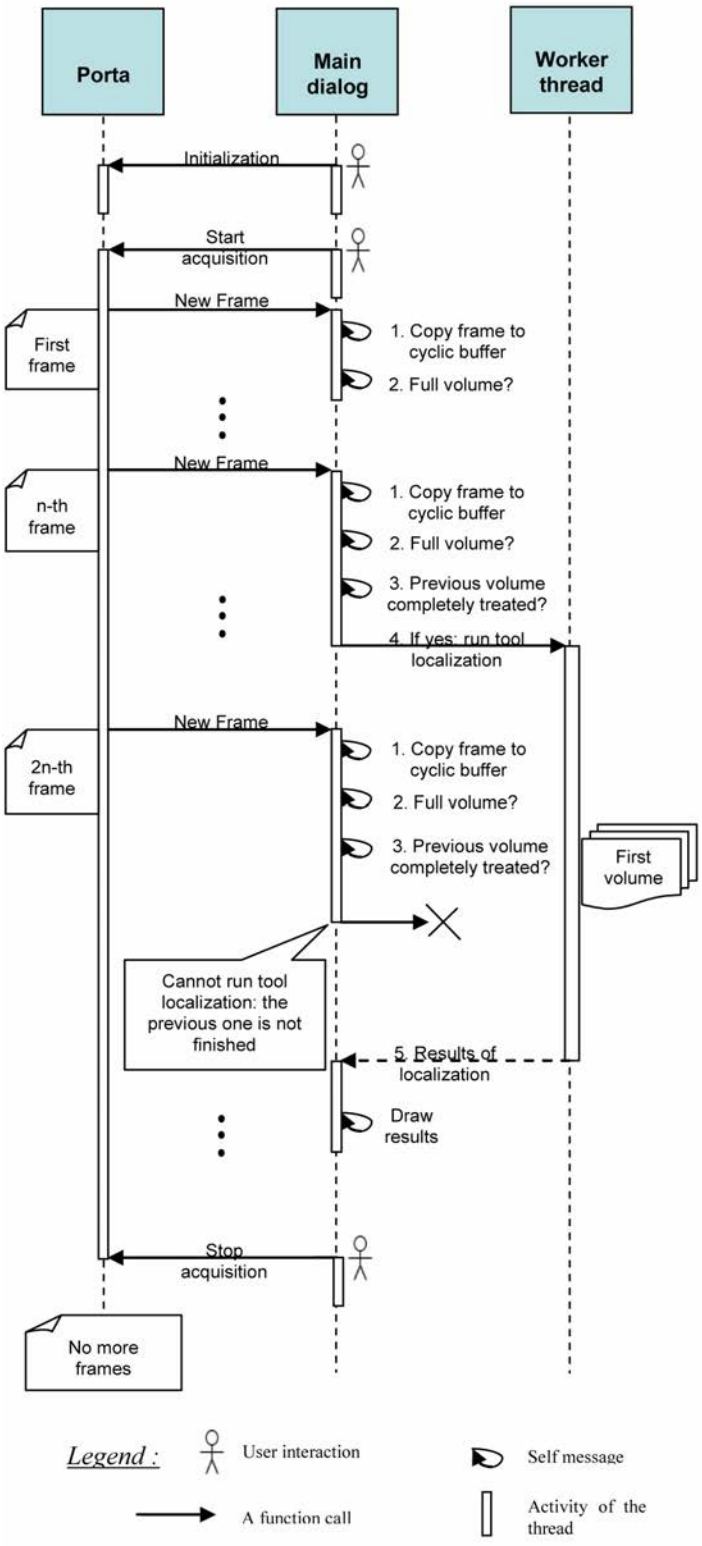


Figure 6.2: *The figure shows how the synchronisation between three threads works: the Porta library, the main dialog and the worker thread. The user controls the acquisition by clicking on the initialize button, the run button and the stop button. Note that here a self message represents one method calling another method belonging to the same object. More explanation about the sequence diagram is given in Section 6.1.2*

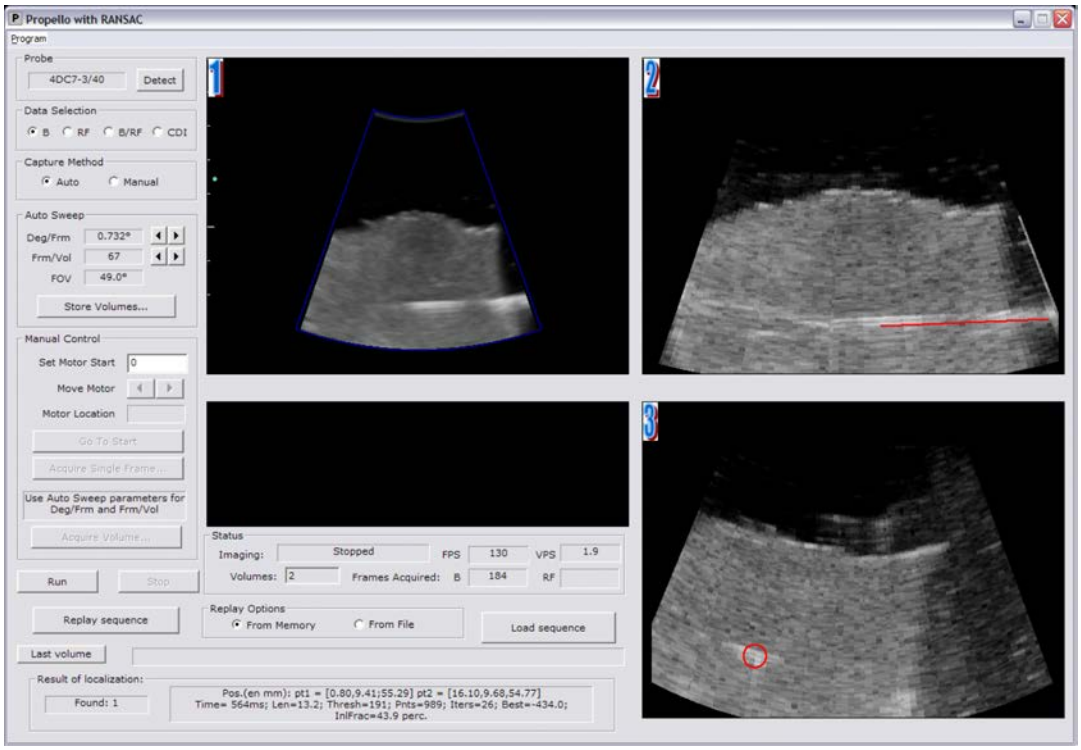


Figure 6.3: *Graphical user interface installed on the Ultrasonix RP. The axis of the surgical tool is indicated by the red line. Several parameters can be changed from the window and the user can replay the acquisition sequence and tool localization either from the memory or from a file. 1) B-mode image. 2) Section plane with the tool. 3) Perpendicular plane.*

overwritten by the acquisition thread. After the worker thread has computed the solution, it returns the result to the main window for visualization (Step 5). The acquisition runs in a loop until the user stops it.

Graphical User Interface

The Graphical User Interface (GUI) shows both the section plane containing the tool (marked by the number 2 in Figure 6.3) and the perpendicular plane to it (marked by the number 3 in Figure 6.3) in order to give a view of surrounding tissue. The section plane (containing the tool) is computed from the coordinates of the tool and from the geometry of 3D volume, so that the 2D slice completely contains the volume and the incident tool. The nearest neighbour interpolation is used to compute the image of the section plane with the tool. The perpendicular plane which intersects the middle of the tool is computed similarly. Our application provides a relevant visualization for a real-time localization (Figure 6.3).

6.2 Results

Experiments were performed on three types of data with surgical tools: (i) submerged in water, (ii) inserted into homogeneous polyvinyl alcohol (PVA) Cryogel phantom [31] and (iii) liver tissue. Conditions under which our experiments were performed are described in Section 6.2.1. The time required for tool localization and the success rate of localization are discussed in Section 6.2.2.

6.2.1 Details of experiments

For our experiments, the B-mode sector angle was 52°. The motor step was set to 0.732° with a corresponding 49° FOV, i.e., 67 frames per volume. The depth of acquisition was 6 cm. The focal point was approximately at the depth of the tool. 3D volumes of size $33 \times 64 \times 224$ voxels have been acquired by using the probe with central frequency 5 MHz.

Concerning the material used in experiments, we used two types of needles: 1) Short needle of a length of 5 cm and a diameter of 1 mm. 2) Longer biopsy needle of length 8 cm and diameter 1.25 mm. The PVA cryogel phantom of size approximately 5 cm × 5 cm × 5 cm was used for insertion of the needles. The distance between the 3D probe and the needle was approximately 4 cm.

We tested the application in a static configuration, i.e., the needle was already inserted and was not moving. When the needle was submerged in water, we tested if we were able to locate the needle in any position relatively to the 3D probe. Then we tried to follow the needle during the insertion inside the phantom. We also performed the experiment with the biopsy needle in the liver tissue sample (bought in a supermarket).

In our experiments, the following parameters were set for tool localization method: minimum length of the tool: 5 mm; maximum radius of the tool: 1 mm; maximum number of points (for automatic threshold estimation): 1000; desired probability: 99.5%; maximum number of iterations: 3000.

6.2.2 Success rate and time

In a static configuration described above, the application displays the correct frame that contains the needle and gives an estimation of the needle position with a high successful rate greater than 98%. When the needle is moving, in any environment, the success rate is decreasing slightly. Although, sometimes the displayed frame

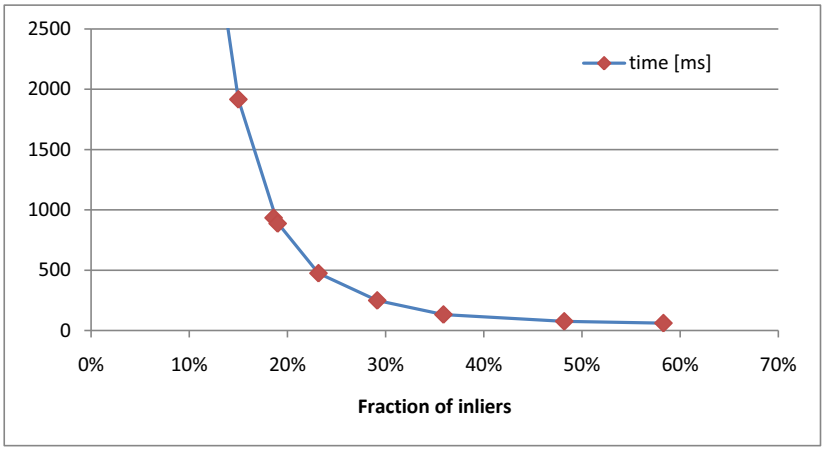


Figure 6.4: *The dependence of localization time on the proportion of inliers.*

is correct, the location of the needle is incorrect. On average, good frames are displayed with a success rate of 95% and good locations of the needle are obtained with a success rate of 75%.

The localization time depends on the environment. It varies linearly with the volume size and the number of points remaining after thresholding (a parameter of tool localization). The time of localization is inversly proportional to the fraction of inliers (true tool voxels) in the thresholded data set (Figure 6.4). For our setting with a needle of diameter of 1 mm, the fraction of inliers is 19% and the localization time is 0.935 seconds on average.

The volumes of size smaller or equal than 67 planes × 128 beams × 224 samples (it corresponds to a depth of 7 cm, a field of view of 104° in horizontal direction and 63° in vertical direction), the localization time is less than the time required for acquisition (about 1 second).

6.3 Discussion

Seeing that it is feasible to localize the tool during its positioning at real-time in various types of an environment, the application can be useful for surgical operations after an optimisation phase of the process. In fact, the automatic needle localization can be difficult because of the images with low contrast, attenuation and artifacts.

It would be also interesting to implement the other tool models based on the intensity profile of the tool as proposed in [P1] (Chapter 4). In order to enhance the contrast of the tool, the second harmonic imaging [5] or the pulse inversion technique [64] or can be used.

The resulting location of the tool has a large variation because the tool localization is performed on each volume of sequence independently. We could use the information computed from the previous volume in order to constrain the estimation for the tracking of the tool in the sequence of volumes.

6.4 Conclusion

We propose to assist surgeons by providing a real-time application which is able to track surgical tools movements. The refresh rate of the tool localization images and displaying the interpolated planes is about one second. It corresponds to the time necessary to acquire a full volume with the 3D probe used here. In the absence of artifacts, our experiments have shown that we are able to locate the surgical tool in water, the PVA cryogel phantom and liver quickly and with a good accuracy. The application is available at our webpage `http://www.creatis.insa-lyon.fr/site/en/ToolDemo`.

Chapter 7

Conclusions and outlooks

This thesis was dedicated to the problem of automatic localization of surgical tools in 3D ultrasound (US) images. The proposed methods were tested on real clinical data of the breast biopsy. They are fast enough to perform the localization in real-time.

The result of localization can be presented as a 2D slice containing the tool because the clinicians are used to observe 2D ultrasound images. Besides, it is not necessary to maintain the observation plane incident with the tool anymore.

The localization method consists of the axis and tip localization. The proposed tool localization method is fast and robust to the deformation of the tool and the presence of other high-intensity structures.

Parallel Integral Projection. The Parallel Integral Projection (PIP) was proposed previously for a straight tool localization. The author contributed to the final publication of Barva et al. [P2] in the *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*.

The existing PIP implementation was rather slow so we proposed to use a pyramid of multiple down-sampled 3D US images which brings the time of the method from the order of hundreds of seconds to below than ten seconds (Chapter 3). The author's proposal was presented at the *IEEE International Symposium on Biomedical Imaging 2008* [P3].

Model fitting using RANSAC. The model fitting is a localization method which estimates the parameters of the axis by using the RANSAC procedure. We proposed two new shape and appearance models in Chapter 4: one simple based on the axis shape and one more complicated model which can better fit the typical intensity profile of the tool. The proposed methods are very fast, the localization time

it depends on the proportion of inliers. In our tests, it was 0.5 s for the simple shape model applied to the volume of $53 \times 71 \times 3100$ voxels. The tool model with a priori known intensity distribution is more robust to the presence of high-intensity structures in the real ultrasound images than the state of the art localization methods. We published the model fitting method in the *IEEE Transactions on Biomedical Engineering* [P1].

We showed that the robustness can be further improved by exploiting a priori information about the one-dimensionality of the tool. We proposed to use the line filtering in pre-segmentation and in the estimation of the tool model (Chapter 5). The robustness of the localization method was further improved on all tested 3D US images including the difficult simulated data, the PVA phantom and the breast tissue. We achieved the zero number of failures, while the other state of the art methods had higher failure rate or they were failing in all cases. The additional time costs with the line filtering were reduced by employing the cascaded classifier from 8 s to 1 s. The preliminary work about line filtering was presented at the *IEEE International Ultrasonics Symposium 2009* [P4] and a publication for a journal is in preparation.

The model fitting is a fast method suitable for real-time localization (Chapter 6). It is demonstrated by a real-time application with the simple shape model implemented on the 3D ultrasound scanner Ultrasonix RP. The localization method was able to follow the movement and insertion of the needle in the phantom. The application was presented at the *IEEE International Ultrasonics Symposium 2010* [P5].

Future work. The model fitting method (Chapter 4) can be further improved by incorporating more a priori knowledge, e.g. by using additional descriptors in method proposed in Chapter 5 or by incorporating constraints for the solution. The other way for improving the method is to replace the RANSAC procedure with variations, e.g., randomized RANSAC [22] or locally optimized RANSAC [23].

A new voxel descriptor based on the radio-frequency (RF) signal could be developed. The RF signal is the raw acoustic data recorded by an ultrasound scanner at high sampling rate (Section 1.3.3). The raw RF data offer a better accuracy than the standard US images which are degraded by post-processing and interpolation. Unfortunately, the most commercial US scanners do not provide an access to the RF data.

A complementary information to intensities in the US image is the acoustic shadow which is relatively strong for the metallic tools. A new tool model could

detect a dark shadow line in the projection under the tool axis. It would be useful in cases when the contrast of the tool is poor because of the specific observation angle.

The tool is usually inserted from the lateral side of the volume so that the tool axis is usually oriented perpendicularly to the sound wave propagation. This can be used as a constraint in the estimation of the tool axis for the rejection of the non-feasible orientations.

The proposed method performs the localization in the single 3D US volume. It can be applied to the sequence of volumes for a real-time tracking (Chapter 6). The tracking can be improved by using the result of localization on the previous volume in sequence. The change of location is expected to be relatively small because of the continuous movement.

Many of the proposed methods are suitable for a parallel implementation: the PIP projections can be computed in parallel, the hyphoteses of the RANSAC can be also tested concurrently. The line filtering uses a local neighborhood so the implementation, e.g., on the GPU, would descrease the time complexity immensely.

More experiments with the real tissue should be performed to validate the proposed method in real clinical cases. We have done experiments with the breast biopsy data and the phantom data, but other types of real tissue should be also considered. The validation of the proposed method with a complementary imaging modality, e.g. MRI or CT, should verify the precision of the method. In the future, the validation of the localization method should be done with a clinician.

Appendix A

Accompanying CD

We provide a supplementary material on the CD Accompanying to this text. The manuscript of the thesis is in the file: `UhercikThesis2010.pdf`.

A.1 Directory structure

Please read the file `ReadMe.txt` first. There are several directories:

- `DemoGUI` – Demonstration application for the tool localization (Section A.2).
- `Papers` – The publications related to this thesis in electronic form.
- `Viz3D` – Supplementary movies with visualization of 3D data used for experiments.

The source code is available at the CMP or the CREATIS laboratory. If you are interrested, please contact the author (Marián Uherčík[1]) or the supervisors (Jan Kybic at CMP, CTU in Prague, Hervé Liebgott and Christian Cachard at CREATIS-LRMN, Université Lyon).

A.2 Demonstration application

We provide a demonstration application on the Accompanying CD. It is different application than the real-time demonstration (Chapter 6). This application is for testing of proposed tool localization methods off-line. The software was implemented in MATLAB and we provide a program with the graphical user interface (GUI) compiled to an executable file.

[1] uhercik@cmp.felk.cvut.cz

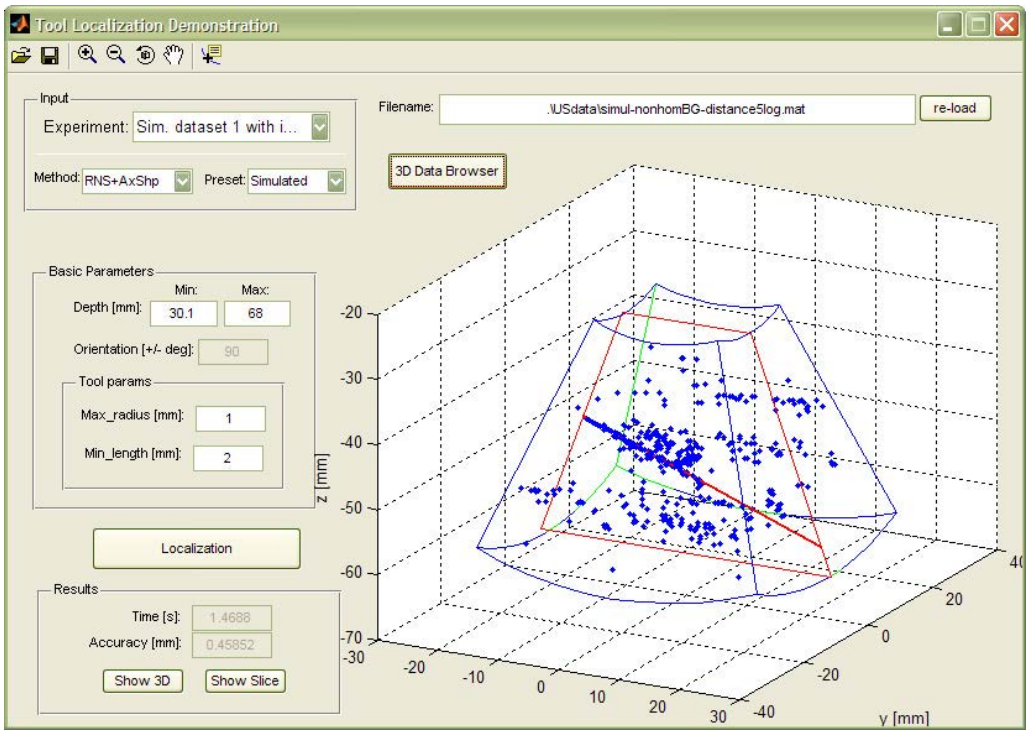


Figure A.1: *The GUI for tool localization in MATLAB.*

Localization methods. The tool localization methods, which are implemented in this application, are based on the model fitting with the RANSAC procedure (Chapter 4). Three shape and appearance models of tool in 3D US have been implemented: the AxShp, the IntDstr and the LineFilter – a tool model with line filtering (Chapter 5).

The models IntDstr and LineFilter were learned on training data, while the model AxShp does not use any learning. We note that the line filtering was performed in advance and the pre-processed data are stored in a separate file. The tool model with line filtering used a cascaded classifier in pre-segmentation and the SVM classifier in the RANSAC estimation.

Software implementation. The application was implemented in the MATLAB and compiled using a MATLAB Compiler. The MATLAB Component Runtime (MCR) is required for running the application. The executable file for the MS Windows is `DemoGUI/ToolDemo/ToolDemo.exe`. There is also a version compiled for Linux in the archive `DemoGUI/ToolDemoLinux.zip`. Please read the file `readme.txt` before starting the program. The 3D US data sets are stored in the sub-directory `USdata` and the tool models are prepared in `USmodels`.

How to use it. The screenshot of the GUI application is in Figure A.2. The main window consists of the 3D view of the thresholded data and the control panel on the left side. The user can select one of the experiments (from list in the top-left corner) and all parameters are automatically set. The 3D US dataset is also loaded and displayed.

The user can modify some parameters, e.g. the ROI, and start the localization by clicking on the button (in left side panel). The result of localization is presented in two ways: (i) as a red line in the main window together with the thresholded set of points, and (ii) as a 2D section plane containing the tool (computed from 3D US data by interpolation) in a separate window. Detailed documentation and a tutorial how to use the application is in the file `DemoGUI/ToolDemoDoc.pdf`.

Set of experiments is prepared in (in the top-left part):

- The first three items (*Sim. dataset ...*) are testing various tool models on simulated data.
- The next group of four items (*PVA-Phantom ...*) uses the real data of the PVA Cryogel phantom with various tool models.
- The last group uses the real US data of breast tissue with two types of needle:
  - Thick needle of diameter 1.067 mm (*Breast Biopsy - thick ...*).
  - Thin needle of diameter 0.4636 mm (*Breast Biopsy - thin ...*).

The very last item (*Breast Biopsy - thin needle - large ROI*) is an example of difficult data set. The most localization methods fail but the tool model with the line filtering is able to localize the tool.

Bibliography

Author's publications

[P1] M. Uherčík, J. Kybic, H. Liebgott, and C. Cachard. Model fitting using RANSAC for surgical tool localization in 3-D ultrasound images. *IEEE Transactions on Biomedical Engineering (BME)*, 57(8):1907–1916, August 2010.

[P2] M. Barva, M. Uherčík, J.-M. Mari, J. Kybic, J.-R. Duhamel, H. Liebgott, V. Hlaváč, and C. Cachard. Parallel integral projection transform for straight electrode localization in 3-D ultrasound images. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control (UFFC)*, 55(7):1559–1569, July 2008.

[P3] M. Uherčík, J. Kybic, H. Liebgott, and C. Cachard. Multi-resolution parallel integral projection for fast localization of a straight electrode in 3D ultrasound images. In *IEEE International Symposium on Biomedical Imaging (ISBI): From nano to macro*, pages 33–36, May 2008.

[P4] M. Uherčík, J. Kybic, C. Cachard, and H. Liebgott. Line filtering for detection of microtools in 3D ultrasound data. In *IEEE International Ultrasonics Symposium (IUS)*, pages 594–597, September 2009.

[P5] F. Gaufillet, H. Liebgott, M. Uherčík, F. Cervenansky, J. Kybic, and C. Cachard. 3D ultrasound real-time monitoring of surgical tools. In *IEEE International Ultrasonics Symposium (IUS)*, October 2010.

[P6] M. Uherčík, H. Liebgott, J. Kybic, and C. Cachard. Needle localization methods in 3D ultrasound data. In *International Congress on Ultrasonics (ICU)*, Book of abstracts, page 60, January 2009.

[P7] M. Uherčík. Surgical tool localization in 3D ultrasound data – PhD thesis proposal. Research Report CTU–CMP–2009–34, Center for Machine Perception,

K13133 FEE Czech Technical University, Prague, Czech Republic, August 2009.

Other references

[1] A. Abati and A. Simsir. Breast fine needle aspiration biopsy: prevailing recommendations and contemporary practices. *Clinics in laboratory medicine*, 25(4):631–654, December 2005.

[2] M. Aboofazeli, P. Abolmaesumi, P. Mousavi, and G. Fichtinger. A new scheme for curved needle segmentation in three-dimensional ultrasound images. In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*, pages 1067–1070, June 2009.

[3] R. L. Alterman, D. Sterio, A. Beric, and P. J. Kelly. Microelectrode recording during posteroventral pallidotomy. *Neurosurgery*, 44(2):315–321, February 1999.

[4] M. E. Anderson and G. E. Trahey. A seminar on k-space applied to medical ultrasound. Technical report, Department of Biomedical Engineering Duke University, February 2006.

[5] B. A. J. Angelsen. *Ultrasound Imaging – Waves, Signals, and Signal Processing*, volume I. Emantes, Norway, 2000.

[6] B. A. J. Angelsen. *Ultrasound Imaging – Waves, Signals, and Signal Processing*, volume II. Emantes, Norway, 2000.

[7] S. Aylward, E. Bullitt, S. Pizer, and D. Eberly. Intensity ridge and widths for tubular object segmentation and description. In *Mathematical Methods in Biomedical Image Analysis (MMBIA), Proceedings of the Workshop on*, pages 131–138, June 1996.

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

[9] M. Barva. *Localization of Surgical Instruments in 3D Ultrasound Images*. PhD Thesis CTU–CMP–2007–12, Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, June 2007.

[10] M. Barva, J. Kybic, H. Liebgott, C. Cachard, and V. Hlaváč. Comparison of methods for tool localization in biological tissue from 3D ultrasound data. In *Proceedings of the IEEE Ultrasonics Symposium*, pages 1983–1986, October 2006.

[11] M. Barva, J. Kybic, J.-M. Mari, and C. Cachard. Radial Radon transform dedicated to micro-object localization from radio frequency ultrasound signal. In *Proceedings of the IEEE International Frequency Control Symposium and Exposition*, pages 1836–1839, August 2004.

[12] M. Barva, J. Kybic, J.-M. Mari, C. Cachard, and V. Hlaváč. Automatic localization of curvilinear object in 3D ultrasound images. In *Medical Imaging 2005: Ultrasonic Imaging and Signal Processing*, volume 6, pages 455–462, Bellingham, Washington, USA, February 2005. SPIE.

[13] P. Bhattacharya, H. Liu, A. Rosenfeld, and S. Thompson. Hough-transform detection of lines in 3-D space. *Pattern Recognition Letters*, 21(9):843–849, 2000.

[14] R. Bulirsch and J. Stoer. *Introduction to Numerical Analysis*. Springer-Verlag, New York, 1991.

[15] C. B. Burckhardt. Speckle in ultrasound B-mode scans. *IEEE Trans. on Sonics and Ultrasonics*, 25(1):1–6, Jan 1978.

[16] J. W. Cannon, J. A. Stoll, and I. S. Salgo. Real time 3-dimensional ultrasound for guiding surgical tasks. *Computer aided surgery*, 8:82–90, 2003.

[17] G. A. Chapman, D. Johnson, and A. R. Bodenham. Visualization of needle position using ultrasonography. *Anesthesia*, 61:148–158, 2006.

[18] S. Chaudhuri, S. Chatterjee, N Katz, M. Nelson, and M. Goldbaum. Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Trans. on Medical Imaging*, 8:263–269, 1989.

[19] S. Cheung and R. Rohling. Enhancement of needle visibility in ultrasound guided percutaneous procedures. *Ultrasound in Medicine and Biology*, 30:617–624, 2004.

[20] K. Chin, A. Perlas, V. Chan, and Brull R. Needle visualization in ultrasound-guided regional anesthesia. *Reg. Anesth. Pain Med*, 33:532–544, Nov-Dec 2008.

[21] S. C. Choi and R. Wette. Maximum likelihood estimation of the parameters of the Gamma distribution and their bias. *Technometrics*, 11(4):683–690, November 1969.

[22] O. Chum and J. Matas. Randomized RANSAC with T(d,d) test. In *Proc. of the British Machine Vision Conference (BMVC), Cardiff, UK*, pages 448–457, 2002.

[23] O. Chum, J. Matas, and J. Kittler. Locally optimized RANSAC. In *Proceedings of 25th Pattern Recognition Symposium*, volume 2781, pages 236–243. DAGM, 2003.

[24] E. Dehghan and S. E. Salcudean. Needle insertion study using ultrasound-based 2D motion tracking. In *Proceedings of MICCAI*, pages 660–667, Berlin, Heidelberg, 2008. Springer-Verlag.

[25] K. G. Derpanis and J. M. Gryn. Three-dimensional Nth derivative of gaussian separable steerable filters. In *Proceedings of ICIP*, volume 3, pages 553–566, Sept. 2005.

[26] S. P. DiMaio and S. E. Salcudean. Needle steering and model-based trajectory planning. In *Proceedings of the MICCAI*, pages 33–40, 2003.

[27] M. Ding, H. N. Cardinal, W. Guan, and A. Fenster. Automatic needle segmentation in 3D ultrasound images. In *Proceedings of SPIE*, volume 4681, pages 65–76, May 2002.

[28] M. Ding and A. Fenster. A real-time biopsy needle segmentation technique using Hough transform. *Medical Physics*, 30:2222–2233, 2003.

[29] M. Ding, Z. Wei, L. Gardi, D. B. Downey, and A. Fenster. Needle and seed segmentation in intra-operative 3D ultrasound-guided prostate brachytherapy. *Ultrasonics*, 44(1):331–336, 2006.

[30] K. J. Draper, C. C. Blake, L. Gowman, D. B. Downey, and A. Fenster. An algorithm for automatic needle localization in ultrasound-guided breast biopsies. *Medical Physics*, 27:1971–1979, 2000.

[31] F. Duboeuf, A. Basarab, H. Liebgott, E. Brusseau, P. Delachartre, and D. Vray. Investigation of PVA cryogel Young's modulus stability with time,

controlled by a simple reliable technique. *Medical Physics*, 36(2):656–661, 2009.

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

[33] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, November 2000.

[34] D. Eberly. *Ridges in Image and Data Analysis*. Kluwer Academic Publishers, 1996.

[35] M. Felber. *Optical measurement system for medical applications*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Switzerland, 2002.

[36] A. Fenster, B. D. Donal, and H. N. Cardinal. Three-dimensional ultrasound imaging. *Physics in Medicine and Biology*, 46:67–99, May 2001.

[37] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *CACM*, 24(6):381–395, June 1981.

[38] R. A. Fisher. *Statistical methods for research workers*. 1970.

[39] V. Franc and V. Hlaváč. Statistical pattern recognition toolbox for MATLAB. Research Report CTU–CMP–2004–08, Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, June 2004.

[40] V. Franc and S. Sonnenburg. OCAS optimized cutting plane algorithm for support vector machines. In *Proceedings of ICML*, 2008.

[41] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. In *Proceedings of MICCAI*, 1998.

[42] W. T. Freeman. *Steerable filters and local analysis of image structure*. PhD thesis, Cambridge, MA, USA, 1992.

[43] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(9):891–906, 1991.

[44] Y. Freund and R. E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.

[45] J. Fromageau, E. Brusseau, D. Vray, G. Gimenez, and P. Delachartre. Characterization of PVA cryogel for intravascular ultrasound elasticity imaging. *IEEE Trans. on UFFC*, 50(10):1318–1324, Oct. 2003.

[46] S. I. Gallant. Perceptron-based learning algorithms. *IEEE Trans. on Neural Networks*, 1(2):179–191, Jun. 1990.

[47] N. D. Glossop, K. Cleary, and F. Banovac. Needle tracking using the aurora magnetic position sensor. In *Proceedings of CAOS*, 2002.

[48] G. González, F. Aguet, F. Fleuret, M. Unser, and P. Fua. Steerable features for statistical 3D dendrite detection. In *Proceedings of MICCAI*, pages 625–632, Berlin, Heidelberg, 2009. Springer-Verlag.

[49] D. Guo and P. Richardson. Automatic vessel extraction from angiogram images. pages 441–444, Sep. 1998.

[50] E. Heitz, P. Perez, and P. Bouthemy. Multiscale minimization of global energy functions in some visual recovery problems. *Computer Vision, Graphics, and Image Processing*, 59(1):125–134, 1994.

[51] S. Helgason. *The Radon Transform*. Birkhauser Boston, New York, USA, 2nd edition, 1999.

[52] A. Henderson. *ParaView Guide*. Kitware Inc., 2007.

[53] A. D. Hoover, V. Kouznetsova, and M. Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Trans. on Medical Imaging*, 19(3):203–210, 2000.

[54] T. T. Horlocker. Peripheral nerve blocks regional anesthesia for the new millennium. *Reg. Anesth. Pain Med*, 23(3):237–40, May-June 1998.

[55] V. Horsley and R. H. Clarke. The structure and functions of the cerebellum examined by a new method. *Brain*, 31:45–124, 1908.

[56] J. Huang, J. K. Triedman, N. V. Vasilyev, Y. Suematsu, R. O. Cleveland, and P. E. Dupont. Imaging artifacts of medical instruments in ultrasound-guided interventions. *Journal of Ultrasound in Medicine*, 26:1302–1322, 2007.

[57] S. Huber, M. Wagner, M. Medl, and H. Czembirek. Real-time spatial compound imaging in breast ultrasound. *Ultrasound in Medicine and Biology*, 28:155–163, 2001.

[58] M. Jacob and M. Unser. Design of steerable filters for feature detection using Canny-like criteria. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26:1007–1019, 2004.

[59] J. A. Jensen. Field: A program for simulating ultrasound systems. In *Medical and Biological Engineering and Computing*, volume 34, pages 351–353. Springer, New York, 1996.

[60] J. A. Jensen. User's guide for the Field II program. Research Report 2.86, Technical University of Denmark, August 2001.

[61] T. Joachims. Making large-scale SVM learning practical. *Advances in Kernel Methods - Support Vector Learning*, 1999.

[62] I. T. Jolliffe. *Principal component analysis*. Springer, 2002.

[63] C. Kirbas and F. Quek. A review of vessel extraction techniques and algorithms. *ACM Comput. Surv.*, 36(2):81–121, 2004.

[64] J. Kirkhorn, S. Frigstad, and H. Torp. Comparison of pulse inversion and second harmonic for ultrasound contrast imaging. In *IEEE Ultrasonics Symposium*, volume 2, pages 1897–1901, October 2000.

[65] N. Kiryati, Y. Eldar, and A. M. Bruckstein. A probabilistic Hough transform. *Pattern Recogn.*, 24(4):303–316, 1991.

[66] N. Kiryati, H. Klviinen, and S. Alaoutinen. Randomized or probabilistic Hough transform: Unified performance evaluation. *Pattern Recognition Letters*, 21:1157–1164, 1999.

[67] H. Klviinen, P. Hirvonen, L. Xu, and E. Oja. Probabilistic and non-probabilistic Hough transforms: overview and comparisons. *Image and Vision Computing*, 13(4):239–252, 1995.

[68] Q. Li, S. Sone, and K. Doi. Selective enhancement filters for nodules, vessels, and airway walls in two- and three-dimensional CT scans. *Medical Physics*, 30(8):2040–2051, 2003.

[69] J.-M. Mari and C. Cachard. Acquire real-time RF digital ultrasound data from a commercial scanner. *Electronic Journal "Technical Acoustics"*, Vol. 3, January 2007.

[70] G. S. Merrick, W. M. Butler, K. E. Wallner, R. W. Galbreath, and E. Adamovich. Monotherapeutic brachytherapy for clinically organ-confined prostate cancer. *West Virginia Medical Journal*, 101(4):168–171, August 2005.

[71] F. Natterer. *The Mathematics of Computerized Tomography*. Society for Industrial and Applied Mathematics, 2001.

[72] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7:308–313, 1965.

[73] H. R. S. Neshat and R. V. Patel. Real-time parametric curved needle segmentation in 3D ultrasound images. In *2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 670–675. IEEE press, 2008.

[74] S. Nicolau, A. Garciaand, X. Pennecand, L. Soler, and N. Ayache. An augmented reality system to guide radio-frequency tumor ablation. *Computer Animation and Virtual World*, 1, 2005.

[75] P. M. Novotny, J. W. Cannon, and R. H. Howe. Tool localization in 3D ultrasound images. In *Proceedings of MICCAI*, volume 2879, pages 969–970, 2003.

[76] P. M. Novotny, J. A. Stoll, N. V. Vasilyev, P. J. del Nido, P. E. Dupont, and R. D. Howe. GPU based real-time instrument tracking with three dimensional ultrasound. *Medical image analysis*, 11(5):458–64, 2007.

[77] S. H. Okazawa, R. Ebrahimi, J. Chuang, R. N. Rohling, and S. E. Salcudean. Methods for segmenting curved needles in ultrasound images. *Medical image analysis*, 10(3):330–342, 06 2006.

[78] M. Orkisz, L. F. Valencia, and M. H. Hoyos. Models, algorithms and applications in vascular image segmentation. *Machine Graphics & Vision International Journal*, 17(1):5–33, 2008.

[79] T. M. Peters. Image-guidance for surgical procedures. *Physics in Medicine and Biology*, 51(14):R505–R540, 2006.

[80] R. Poli and G. Valli. An algorithm for real-time vessel enhancement and detection. *Computer Methods and Programs in Biomedicine*, 52(1):1–22, 1997.

[81] W. Qiu, M. Ding, and M. Yuchi. Needle segmentation using 3D quick randomized Hough transform. In *Proceedings of SPIE Medical Imaging*, pages 449–452, 2008.

[82] K. S. Roberts. A new representation for a line. In *Proceedings of the CVPR*, pages 635–640, June 1988.

[83] S. E. Salcudean, T. D. Prananta, W. J. Morris, and I. Spadinger. A robotic needle guide for prostate brachytherapy. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2975–2981, May 2008.

[84] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis. Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical image analysis*, 2(2):143–168, June 1998.

[85] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37:297–336, December 1999.

[86] L. S. Scott, P. Douglas, G. H. Baltuch, and J. L. Jaggi. Error analysis of MRI and Leksell stereotactic frame target localization in deep brain stimulation surgery. *Stereotactic and functional neurosurgery*, 83(1):1–5, 2005.

[87] B. D. Sites, R. Brull, V. W. Chan, B. C. Spence, J. Gallagher, M. L. Beach, V. R. Sites, S. Abbas, and G. S. Hartman. Artifacts and pitfall errors associated with ultrasound-guided regional anesthesia. Part II: a pictorial approach to understanding and avoidance. *Regional anesthesia and pain medicine*, 32(5):419.

[88] J. Sochman and J. Matas. WaldBoost – learning for time constrained sequential detection. In *Proceedings of the CVPR*, pages 150–156, Washington, DC, USA, 2005. IEEE Computer Society.

[89] J. J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken. Ridge based vessel segmentation in color images of the retina. *IEEE Trans. on Medical Imaging*, 23(4):501–509, 2004.

[90] J. Stoll and P. Dupont. Passive markers for ultrasound tracking of surgical instruments. In *Proceedings of MICCAI*, pages 41–48, 2005.

[91] J. Stoll, P. Novotny, R. Howe, and P. Dupont. Real-time 3D ultrasound-based servoing of a surgical instrument. In *IEEE International Conference on Robotics and Automation*, pages 613–618, May 2006.

[92] T. Svoboda, J. Kybic, and Hlaváč V.

[93] L. Tao, U. Castellani, A. Fusiello, and V. Murino. 3D acoustic image segmentation by a RANSAC-based approach. In *Proceedings of OCEANS*, pages 1098–1101, September 2003.

[94] Z. Tao, C. C. Jaffe, and H. Tagare. Tunnelling descent: A new algorithm for active contour segmentation of ultrasound images. In *Proceedings of IPMI*, pages 246–257, 2003.

[95] X. Tizon. *Algorithms for the analysis of 3D magnetic resonance angiography images*. PhD thesis, Centre for Image Analysis, SLU, 2004.

[96] P. H. S. Torr, A. Zisserman, and S. J. Maybank. Robust detection of degenerate configurations while estimating the fundamental matrix. *Computer Vision and Image Understanding*, 71(3):312–333, 1998.

[97] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *Journal of Computer Vision*, 63(2):153–161, 2005.

[98] V. J. Wedel and R. L. Pruter. Needle guide for ultrasound transducers. Patent no. 5,052,396 USA, 1991.

[99] Z. Wei, M. Ding, D. Downey, and A. Fenster. 3D TRUS guided robot assisted prostate brachytherapy. In *Proceedings of the MICCAI*, pages 17–24, 2005.

[100] L. Xu and E. Oja. Randomized Hough transform (RHT): Basic mechanisms, algorithms, and computational complexities. *CVGIP: Image Understanding*, 57(2):131–154, 1993.

[101] L. Xu and E. Oja. Randomized Hough transform. *Encyclopedia of Artificial Intelligence*, pages 1354–1361, 2008.

[102] L. Xu, E. Oja, and P. Kultanen. A new curve detection method: randomized Hough transform (RHT). *Pattern Recognition Letters*, 11(5):331–338, 1990.

[103] J. H. Youk, E.-K. Kim, M. J. Kim, Ji Y. Lee, and K. K. Oh. Missed breast cancers at US-guided core needle biopsy: How to reduce them. *Radiographics*, 27(1):79–94, January-February 2007.

[104] Weiping Z. and Huazhong S. Detection of cerebral vessels in MRA using 3D steerable filters. In *IEEE Engineering in Medicine and Biology Society (EMBS)*, pages 3249–3252, Jan. 2005.

[105] H. Zhang, F. Banovac, R. Lin, N. Glossop, B. J. Wood, D. Lindisch, E. Levy, and K. Cleary. Electromagnetic tracking for abdominal interventions in computer aided surgery. *Computer Aided Surgery*, 11(3):127–136, May 2006.

[106] H. Zhou, W. Qiu, and M. Ding. Automatic needle segmentation in 3D ultrasound images using 3D Hough transform. In *Proceedings of SPIE Medical Imaging*, 2007.

[107] H. Zhou, W. Qiu, and M. Ding. New realtime needle segmentation technique using grayscale Hough transform. In *Proceedings of SPIE Medical Imaging*, 2007.

[108] H. Zhou, W. Qiu, M. Ding, and S. Zhang. Automatic needle segmentation in 3D ultrasound images using 3D improved Hough transform. In *Proceedings of SPIE Medical Imaging*, 2008.