

UiO : **Department of Informatics**
University of Oslo

Automatic detection of skeletal muscle architecture features

Frida Elén Jalborg
Master's Thesis Spring 2016



Abstract

Muscle architecture features, typically defined by the length, curvature and the angle of insertion of fascicles, are directly related to muscle function. In addition to its applications in fundamental and applied physiology, these features are relevant in several clinical contexts such as aging, obesity, cerebral palsy or critical illnesses in general. These features can be extracted from ultrasound images. Manual analysis of the ultrasound images is, in addition to being time-consuming, associated with a high degree of uncertainty when it comes to reliability. An automatic algorithm for detection of these features would not only be time-saving, but also ensure objectivity which will contribute to the reliability of the estimations.

This thesis proposes an algorithm for estimating pennation angle and fascicle length in single frame ultrasound images of the vastus lateralis. This algorithm includes detection of the region of interest, filtering of the images to reduce speckle noise and enhance the structures, aponeuroses detection, fascicle detection and calculation of the muscle features. The main aim of filtering the image is to enhance the structures in the image while reducing the noise. The best method to achieve this is using the Knutsson tensor filters. The detection of aponeuroses is done using local Radon transform, while the detection of fascicles is done using a normalized local Radon transform.

The pennation angle is estimated based on the angle of the lower aponeurosis and a representative reference fascicle, resulting in one pennation angle. The fascicle length is estimated based on a reconstructed fascicle represented by a 2. degree polynomial, taking the curvature into account. The aponeuroses are represented as straight lines. The aponeuroses and fascicles are extrapolated in cases where the entire fascicle is not within the field of view. The algorithm shows promising results, but further improvements should be made in order to make it more reliable, especially when it comes to how the aponeuroses and fascicles are represented.

Acknowledgements

This thesis has been a collaboration between the Department of Informatics and Norwegian School of Sport Sciences. It was started in January 2015 and completed in May 2016.

First, I would like to thank my supervisors Professor Anne H. Schistad Solberg and Professor Olivier Seynnes for all the feedback and advice throughout the work with this thesis. Their knowledge, ideas and interest in this thesis has been of great value. I also want to thank them for providing the material and data needed for this project.

Finally I wish to thank my friends and family for their encouragement. A special thanks to my dad and my sister for their feedback on this thesis, and my mom for always providing me with great support and food.

Contents

1	Introduction	9
2	Background	11
2.1	Ultrasound imaging	11
2.2	Image Description	12
2.3	Muscle Architecture	13
2.4	Speckle noise	14
2.5	Speckle reducing filters	14
2.6	Previous work	15
2.7	Validity of the measurements	17
3	The data material	19
3.1	Ultrasound images of skeletal muscle	19
3.1.1	Data material without manual analysis	19
3.1.2	Data material with manual analysis	23
3.1.3	Challenges	24
3.2	Simulated ultrasound images of skeletal muscles	24
4	Methods	27
4.1	Automatic detection of region of interest	27
4.2	Directional filtering of the images	28
4.2.1	Sobel filter	28
4.2.2	Frangi Vessel filtering	29
4.2.3	Non-local means	31
4.2.4	Gaussian directional filters along the dominant direction	32
4.2.5	Knutsson tensor filters	33
4.3	Line detection	36
4.3.1	Hough transform	36
4.3.2	Radon transform	37
4.3.3	Choice of line detection method	38
4.4	Detecting upper and lower aponeurosis	38
4.4.1	Approximate aponeuroses detection	39
4.4.2	Exact aponeuroses detection	41
4.5	Detecting fascicles	42
4.6	Construct a reference fascicle	43
4.7	Calculate muscle features	47
4.7.1	Calculate fascicle length	47

4.7.2	Calculate pennation angle	47
4.8	Overview of the algorithm	49
5	Evaluation of results	51
5.1	Evaluation of structure enhancing filtering	51
5.1.1	Frangi vessel filtering	51
5.1.2	Non-local means	51
5.1.3	Gaussian directional filters along the dominant direction	52
5.1.4	Knutsson tensor filters	52
5.1.5	Choice of filter	52
5.2	Results when testing on simulated images	59
5.2.1	Results for aponeuroses detection	59
5.2.2	Results for fascicle detection	64
5.2.3	Results for pennation angle and fascicle length	67
5.3	Results for ultrasound images without manual analysis	71
5.3.1	Results for aponeuroses detection	71
5.3.2	Results for fascicle detection	71
5.3.3	Results for pennation angle and fascicle length	72
5.4	Results for images with manual analysis	77
5.5	Assumption that the results can be transferred to the area outside field of view	82
6	Conclusion	83

Chapter 1

Introduction

This thesis is a cooperation between the Department of Informatics and Norwegian School of Sport Sciences on ultrasound imaging of muscles. Ultrasound images of skeletal muscles are acquired in vivo by using B-mode ultrasound, and are used for characterizing muscle architecture parameters. Muscle architecture refers to the morphological features and the three-dimensional arrangement of muscle fibres, and is directly related to muscle function. It is typically defined by the length, curvature and angle of insertion of fascicles. In addition to its applications in fundamental and applied physiology, these features are relevant in several clinical contexts such as aging, obesity, cerebral palsy or critical illnesses in general. The most common way to acquire measurements of these features today is through manual analysis, but this is both time-consuming and prone to human error and experimenter bias. We would therefore like to have a faster, more objective and accurate approach to this process.

Automated segmentation and/or tracking algorithms have been developed by a few research groups but their use is currently limited because they were not published or because of methodological aspects (e.g. manual segmentation before tracking). There is therefore an unmet need for the development of an automated method to measure architectural parameters in single images and in image sequences.

The aim of this thesis is to develop a fully automated algorithm that is able to estimate the pennation angle and fascicle length. This would ensure objectivity by eliminating user interaction and also save time. This thesis will not cover tracking of fascicles, but the segmentation of fascicles and aponeuroses in a single frame ultrasound image. The pennation angle is usually defined as the angle of insertion between the muscle fascicles and the lower aponeurosis. In this thesis the goal is to estimate one representative pennation angle. The fascicle length should be estimated from a single frame although the entire fascicle is not within the field of view. We also want to take the curvature of the fascicle into account when estimating fascicle length.

Chapter 2

Background

Topics relevant to this thesis are introduced in this chapter. It includes basic information about ultrasound imaging and muscle architecture. A description of the images relevant to this thesis is provided. Finally an overview of a selection of studies relevant to this thesis is presented.

2.1 Ultrasound imaging

The information in this section is mainly gathered from [20], [3] and [12]. Ultrasound is an imaging technique which is widely used in medical applications. The safe and non-invasive nature of this technique provides the ability to image humans in vivo. This thesis will deal with brightness mode (B-mode) ultrasound images. B-mode is a two-dimensional ultrasound presentation of a scan in a single plane. The intensity of the scatterers is represented by the brightness of the pixel, and the position of the scatterer is determined by the transit time of the pressure wave.

The images are formed by emitting short pressure waves into the tissue and detecting the returned echo. The pressure waves are emitted through a transducer. There are several different types of transducers (e.g. phased arrays, linear arrays, etc.). The transducer generates pressure waves by passing an electric current to the piezoelectric crystals in the transducer. The current makes the crystals contract and expand. When an echo returns to the transducer, it will convert the vibration of the wave into electricity. The frequencies of these pressure waves are usually between 2 and 15 MHz depending on the application.

The chosen frequency is one of the factors contributing to the resulting image resolution. Image resolution is defined by range resolution and lateral resolution. Range resolution is limited by the pulse length and determines how close two objects can be located along a scan line and still be detected as separate objects. If two objects are closer together than the pulse length, the echo from these objects will overlap and they will be detected as one object. The lateral resolution is limited by the beam width, and determines the resolution of two objects located side by side. The beam width will be narrower for higher frequency pressure waves, but higher frequencies will also attenuate faster when traveling through the tissue. When deciding what frequency to use for ultrasound imaging we therefore have to make a compromise between the range of the pressure waves and how small details we want to capture [20]. Beam focusing can also be used to adjust the lateral

resolution. The images used in this thesis was acquired using frequencies of 12 MHz, and the resolution was 11.300 pxl/mm, 9.035 pxl/mm and 7.530 pxl/mm for depths of 4, 5, and 6 cm respectively.

The backscattering will occur when the pressure wave hits a boundary between two mediums where some of the energy will be transmitted and some will be reflected. The reflected and transmitted amount is dependent on the acoustic impedance of the mediums. The acoustic impedance in soft tissues and organs is approximately the same because of the high water content. The sound waves will therefore travel at approximately the same speed in these tissues, 1540 m/s. The difference in the acoustic impedance is sufficient to get a reflection, but not so big that it causes geometric distortion. The reflection coefficients of for example bone and air are very high. This means that very little of the energy is transmitted, resulting in inability to see past bone and air when using ultrasound.

2.2 Image Description

Figure 2.1 shows an example of a B-mode ultrasound image of a skeletal muscle. Skeletal muscles are one of three main muscle types in the body. They are attached to bones by tendons. These are the only muscles we can control voluntarily, as opposed to the other muscle types, i.e. smooth muscles (e.g. muscles in the walls of blood vessels) and cardiac muscles. The muscle fibers in skeletal muscles can be seen as bundles of 5 to 50 muscle fibers [17]. These bundles are called fascicles. The connective tissue between the fascicles can be seen as bright areas. The contractile material of the fibers parallel to the connective tissue can be seen as darker areas.

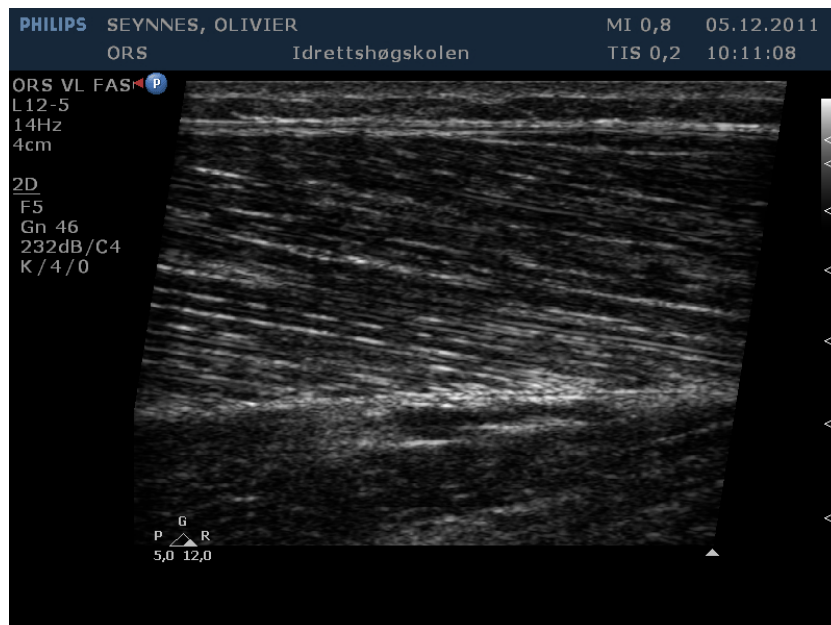


Figure 2.1: Example of what an ultrasound image of the muscle looks like.

The pennation angle is usually defined as the angle between the lower aponeurosis and the fascicles. Aponeuroses are the layer of connective tissue wrapping the whole muscle. Because we are looking at a 2D scan, we see the upper and lower aponeuroses. They are usually thick white lines running horizontally above and under the fascicles of a muscle. Figure 2.2 shows an image of the features that are significant for this thesis. In this image only a portion of the fascicle is visible. This is the case in many ultrasound images of muscles.

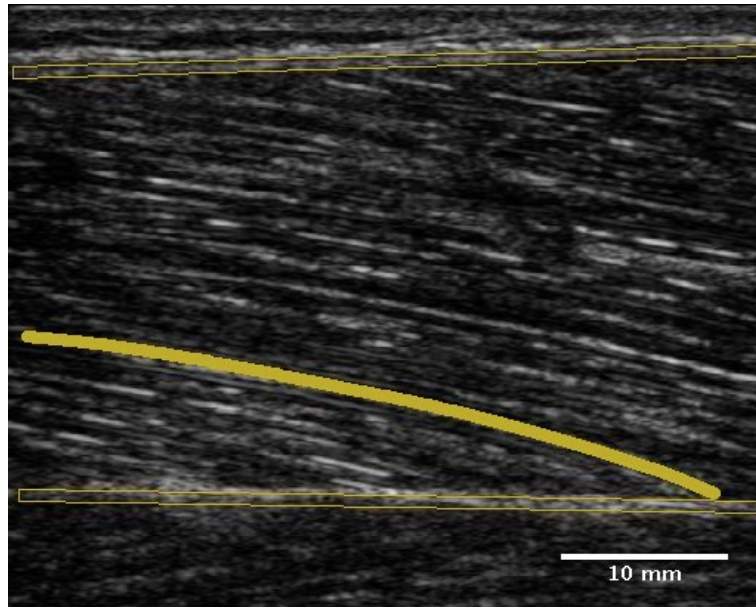


Figure 2.2: Image showing a fascicle (solid line) and the upper and lower aponeurosis (outlined).

2.3 Muscle Architecture

A muscle's architecture is the arrangement of muscle fibers which is related to the muscle function [21]. Every muscle in the body is different, but we can divide them into three different types of muscle architecture; fusiform, unipennate and multipennate architecture. These different muscle architectures differ from each other through the length, the curvature and the pennation angle of the fascicles. These parameters change depending on the joint angle and muscle contraction [17]. Examples of these muscle architectures are shown in figure 2.3.

Fusiform architecture means that the muscle fibers are positioned parallel to the axis of force generation. An example of this architecture is the biceps.

The muscle architecture relevant for this thesis is the *unipennate architecture*. Muscles with this architecture have fibers that are positioned at an angle relative to the axis of force generation. This angle is related to the muscle function, which is the reason why we want to develop a reliable method to estimate this angle. The vastus lateralis is an example of a muscle with this kind of architecture. The pennation angle increases when the muscle is

flexed, while the fascicle length decreases.

Multipennate architecture describes a muscle which has fibers running in multiple different angles relative to the axis of force generation. The gluteus medius has this kind of architecture.

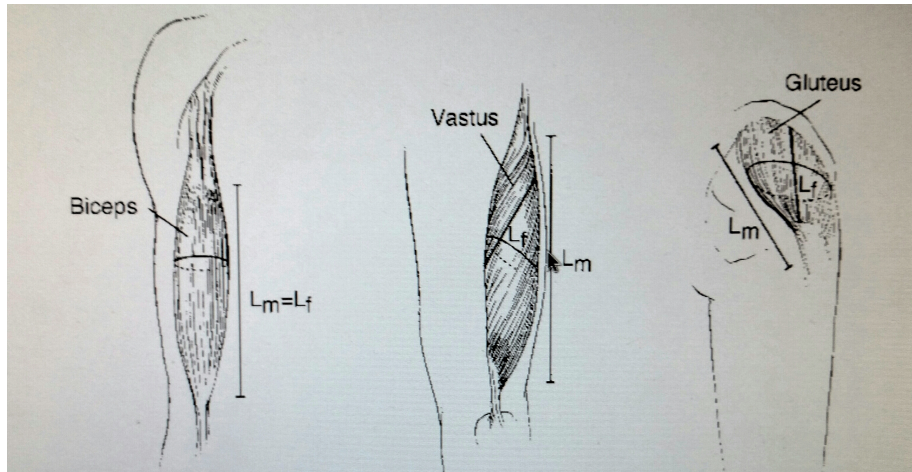


Figure 2.3: Image of respectively longitudinal, unipennate and multipennate architecture [17].

2.4 Speckle noise

One of the main challenges when working with ultrasound images is related to image quality. One of the factors contributing to degraded quality is the occurrence of speckle noise. It complicates image processing tasks such as feature segmentation. It is generated due to interference of backscattered signals originating from closely spaced scatterers in the material being scanned. It appears as a grainy pattern throughout the image. This pattern is caused by the constructive and destructive interference from the scatterers.

Speckle noise reduces both image contrast (signal to noise ratio), and limits the ability to detect small structures. It is ascribed to poor performance of conventional edge detectors in ultrasound images [27].

The information in this section is mainly gathered from [10] and [23].

2.5 Speckle reducing filters

To improve the quality of the images, reducing the speckle noise before further processing is a good idea. When despeckling we have to be careful not to remove relevant information. This can happen if important structures have the same scale as the speckle. In our

case this is not a problem as we are only interested in the larger structures in the images.

Some methods for reducing speckle noise includes the Lee filter [16] and the Kuan filter [15]. These filters were developed for synthetic aperture radar images, which also contain speckle noise. The Lee filter filters the image by looking at a local area around each pixel. Each pixel value is calculated by averaging the neighbouring pixels with intensities within a chosen sigma range compared to the center pixel. The Kuan filter is an extension of the Lee filter. In addition to looking at the local mean it uses the local variance as a weight to how much it should filter this area. Both of these filters use local statistics of the image to preserve edges while reducing noise. Both these filters are however very sensitive to the chosen filter size.

Two other methods are Speckle Reducing Anisotropic Diffusion (SRAD) [25] and Detail Preserving Anisotropic Diffusion (DPAD) [1]. SRAD is a diffusion method developed specifically for removing speckle in ultrasound and radar images. It is based on the same approach as the Lee filter, but it eliminates some of the main disadvantages of these filters by using a partial differential equation approach. While the Lee and Kuan filters only preserves edges, SRAD also enhances edges by performing directional filtering. DPAD is an extension of SRAD that use a diffusion function based on Kuan's filter. While SRAD performs the estimation of statistics and the filtering in parallel, [1] showed that more stable estimates can be obtained by splitting these two procedures. Oriented Speckle Reducing Anisotropic Diffusion (OSRAD) [14] is an oriented diffusion based approach to the despeckling problem. It extends the SRAD method by allowing a directional filtering in the gradient and the principal curvature directions. Variants of the non-local means method have also shown to perform well when it comes to despeckling ultrasound images. This method is further explained in chapter 4.

A full description and comparison of these filters will not be included in this thesis, but can be read in [11].

2.6 Previous work

The studies that will be presented in this section involves both segmentation of single-frame ultrasound images and algorithms for tracking the fascicles. Some of the tracking algorithms mentioned in this section segments the aponeuroses and fascicles manually prior to tracking. These studies can still provide ideas of the best way to define the fascicles and aponeuroses (e.g. linear or quadratic functions).

To better understand dynamic muscle-tendon interaction during human locomotion, fascicle tracking methods to measure the changes in pennation angle and fascicle length have been proposed. Methods proposed in [5] and [18] depend upon a manual segmentation of aponeuroses and fascicles prior to tracking. As mentioned earlier, manual analysis is prone to human error and experimenter bias, and are therefore not always reliable [27]. A study conducted by Rana, Hamarneh and Wakeling [21] showed that manual digitization of muscle fascicles during a dynamic contraction resulted in a standard deviation of angle estimates of 1.41 degrees across ten researchers. This manual step could possibly be replaced by an automated method if a reliable algorithm is developed. Several studies have

proposed methods to automate this analysis.

Zhou, Chan and Zheng [27] proposed a method to determine pennation angle and fascicle length. The gastrocnemius muscle was used for their analysis. They defined the fascicles and the aponeuroses as straight lines. The superficial and deep aponeurosis amounted to the borders of the region of interest. They were found using the normalized Radon transform. The dominant fascicle orientation in the region of interest was detected using the maximum variance in the Hough transform. A bank of Gabor wavelets constructed to find the dominant orientation also contributed with a weight. After these preprocessing steps it was possible to localize and track an individual fascicle and find the pennation angle and fascicle length through basic trigonometry. If the entire fascicle was not present in the region of interest, they estimated the length by extending both the fascicle and the superficial aponeurosis linearly. The length will however often be underestimated by assuming straight fascicles because they in reality often have a curved shape.

Rana, Hamarneh and Wakeling [21] developed a multistage process to automatically quantify the global and local orientations of the fascicles within a muscle from a single-frame ultrasound images. Initially they filtered the image using a multiscale vessel enhancement method proposed by Frangi et al. [8]. The Radon transform was used to identify the dominant fascicle orientation within an image, which can be used for estimating the linear fascicle length. The wavelet analysis provided information on the local fascicle orientations, and can be used to quantify fascicle curvatures and regional differences. They compared the results with simulated images with known orientation and with real ultrasound images analyzed manually.

Wakeling and Randhawa [24] conducted a study for better understanding how muscle fibres change shape during contraction. Their method consisted of an initial manual step where they identified three coordinates for representing the aponeuroses, and two coordinates on a representative fascicle. This study therefore described the aponeuroses using a second-order polynomial and the fascicles using a straight line. They filtered the image using the same multiscale vessel enhancement method as Rana, Hamarneh and Wakeling.

Mersmann et al. [18] did a study on imbalanced muscle strength and tendon properties in adolescent athletes. To conduct the trial, they used a semi-automatic feature-tracking algorithm. This code was not published, but it is described to some extent in the article. Initially they manually segmented one fascicle and the superficial and deep aponeurosis to define the region of interest. Both fascicles and aponeuroses were defined by straight lines. The region of interest was defined as the area between the upper and lower aponeuroses. The visible part of the fascicles was then automatically tracked using a non-linear least squares fitting to analyze the shift of the brightness profiles within horizontal lines. From the behavior of the chosen fascicles they could calculate a reference fascicle to find the features they wanted [18].

Darby et al. [6] proposed a method for tracking changes in fascicle shape and length in B-mode ultrasound images of skeletal muscles. They detected the fascicle plane using the active shape model formulation. The fascicle tracking was done looking at the change in average shape throughout the image instead of tracking one specific fascicle. To achieve

this they used a Bayesian tracking framework which allowed the fascicles to be non-linear. More detailed information about this can be read in [6]. To identify fascicle shapes in a single frame they ran this image in a sequence using the same algorithm as when tracking the fascicles.

Summary

[6], [24], [21] and [18] focused on tracking of fascicles. In addition to tracking, [6] also performed the segmentation automatically, while [18] and [24] performed the segmentation of fascicles and aponeuroses manually before tracking. Even though [21] focused on tracking, the algorithm they developed might as well be used on single frame ultrasound images. [27] was the only one of these studies that were focused only on the segmentation of fascicles and aponeuroses. [6], [24] and [21] used the Frangi vessel enhancement method to enhance the structures in the image.

Darby et al. [6] and Rana, Hamarneh and Wakeling [21] was the only of the mentioned studies that allowed fascicles to take up non-linear shapes. [21] only allowed this when using the wavelet analysis. The other studies, [24], [27], [18], assume straight fascicles.

The methods used in the different studies for detecting aponeuroses includes the active shape model formulation [6] and the normalized Radon transform [27]. I could not find any information about how they detected the aponeuroses in [21]. The methods used for detection of fascicles included the active shape model [6], the Radon transform [21], wavelet analysis [21] and the Hough transform in combination with a bank of Gabor wavelets [27].

2.7 Validity of the measurements

Bénard et al. [2] conducted a study to investigate how the scanning procedure (e.g. tilt and rotation of the probe) could affect the errors in measurement of muscle geometry. The study was conducted on the medial gastrocnemius in both human cadavers and human in vivo subjects. Their results indicated that the ultrasound images yield valid estimates of the true muscle parameters as long as the ultrasound probe is parallel to the true fascicle, and that substantial measurement errors can be made when the ultrasound probe is oriented perpendicular to the skin.

Chapter 3

The data material

We have a few different datasets that each will be used for different types of evaluation. These datasets are introduced in this chapter.

3.1 Ultrasound images of skeletal muscle

Two of the datasets consist of ultrasound images of the vastus lateralis. This is the largest muscle in the quadriceps group which is located in the thigh. These images are acquired using a Philips HD11 XE scanner operating at 12 MHz.

3.1.1 Data material without manual analysis

This dataset contains ultrasound images of different quality for which we do not have a manual analysis. Because of this the evaluation of the performance of the algorithm on this dataset will be based on visual inspection. This dataset will primarily be used to develop the algorithm and find good parameter settings. These images will also be used for demonstrating the methods throughout this thesis.

The first part of this dataset consists of 391 frames from the same scan sequence. These are acquired using an image acquisition frequency of 43 Hz which means that the images are taken over a period of approximately 9 seconds. They are relatively good quality images, meaning that there are a lot of visible fascicles. The fascicles are continuous to some extent. These images also contain a prominent upper aponeurosis and an indistinct lower aponeurosis. Examples of these images are shown in figures 3.1 and 3.2. They are selected from the start and end of the scan sequence respectively.

The other part of this dataset contains 24 lower quality images. Unlike the first part of the dataset, these images do not originate from the same scan sequence. Because of this the image acquisition frequency is not the same in all images. The main characteristics of these images are few visible fascicles, fragmented fascicles, low contrast and patches where fascicles or aponeuroses are smeared. This problem may happen when the operator do not have sufficient training or with unfavourable body composition (e.g. subcutaneous or intramuscular adipose tissue). On the positive side both the upper and lower aponeuroses are prominent and easy to detect. Examples of these images can be seen in figure 3.3, 3.4, 3.5 and 3.6.

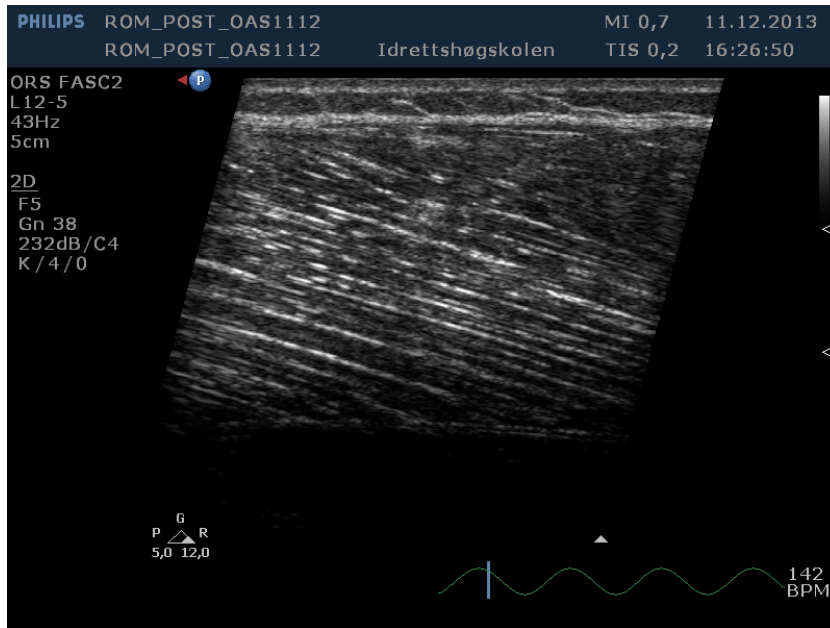


Figure 3.1: Frame number 15 from the scan sequence. The scan sequence is the first part of the dataset without manual analysis.

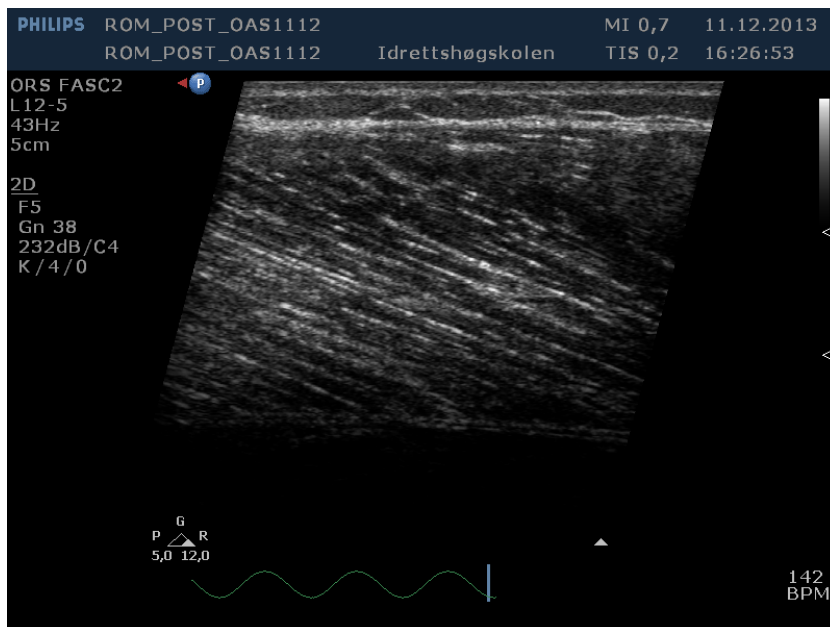


Figure 3.2: Frame number 388 from the scan sequence. The scan sequence is the first part of the dataset without manual analysis.

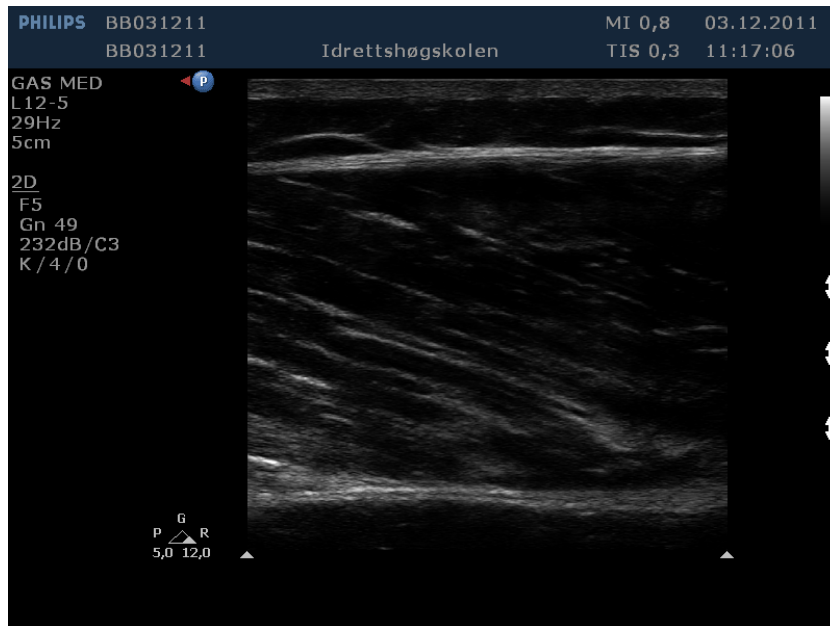


Figure 3.3: Example of lower quality ultrasound image.

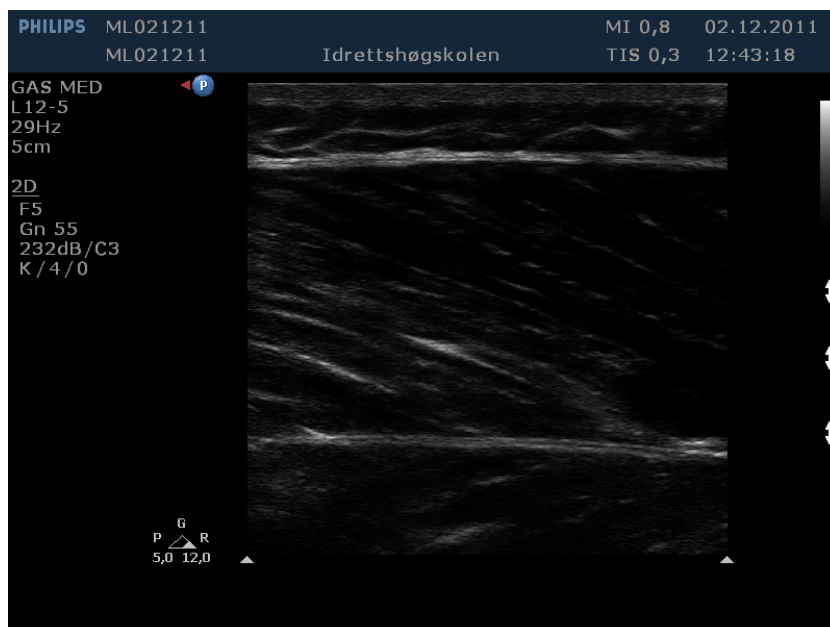


Figure 3.4: Example of lower quality ultrasound image.

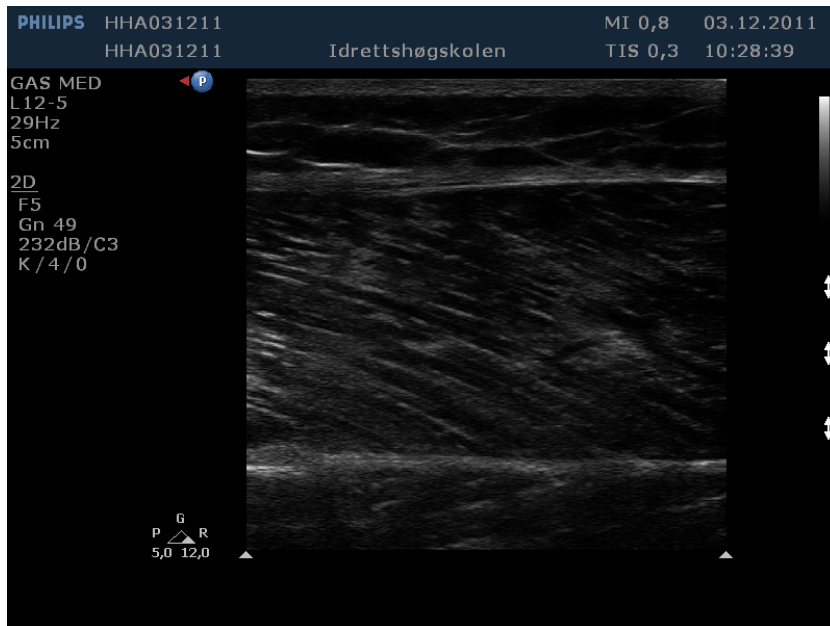


Figure 3.5: Example of lower quality ultrasound image.

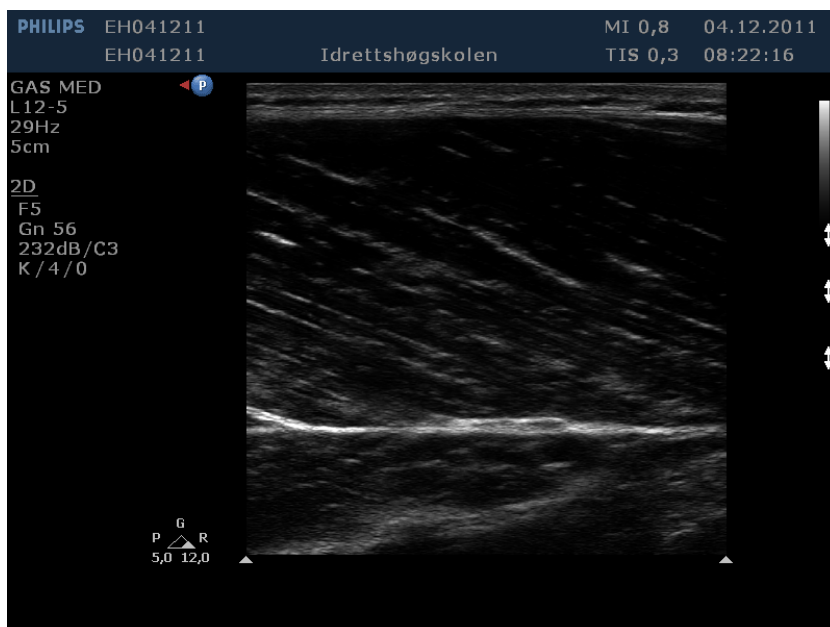


Figure 3.6: Example of lower quality ultrasound image.

3.1.2 Data material with manual analysis

This part of the data material consists of 20 sets of ultrasound images. Each set consists of two images corresponding to two conditions. This data will be used to evaluate the algorithm compared to the manual method. The manual analysis only recorded the pennation angle, so this is the only feature that will be compared to the digital analysis. Figure 3.7 and 3.8 shows two example images from this dataset. This dataset was made available a couple of weeks before the deadline for this thesis. Thus, it was not used to develop the algorithm or tune parameters, but purely to evaluate the results.

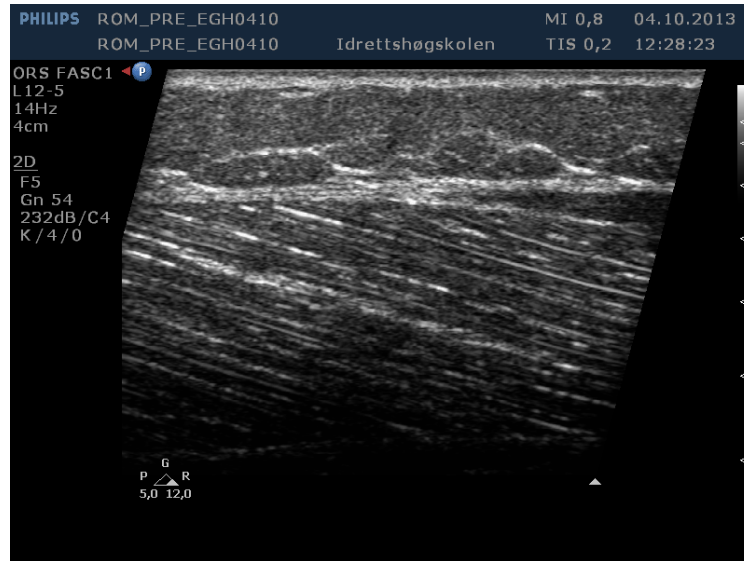


Figure 3.7: Example of image from the dataset with manual analysis. Referred to as subject 6, condition short in table 5.9.

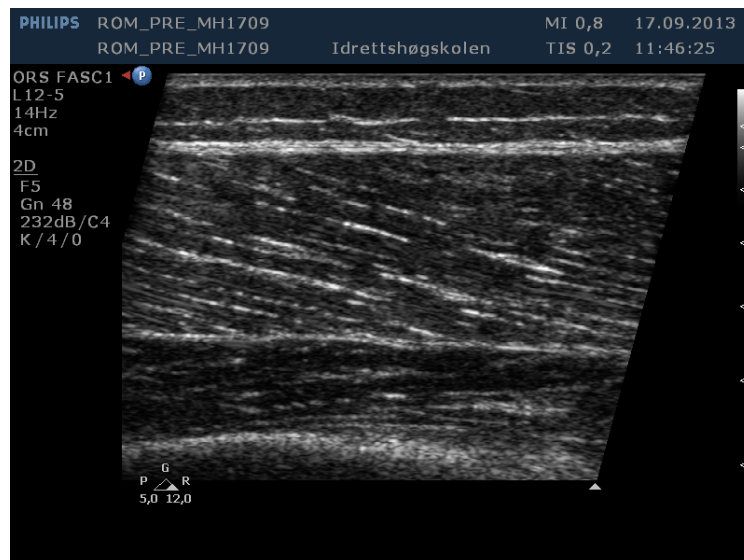


Figure 3.8: Example of image from the dataset with manual analysis. Referred to as subject 9, condition long in table 5.9.

3.1.3 Challenges

Several challenges are faced when working with these two datasets. First of all the images are noisy, meaning both unspecific noise (e.g. speckle noise) and specific noise (e.g. echo from genuine structures like intramuscular blood vessels or connective tissue). This can pose a challenge during line detection. The algorithm needs to be versatile in order to be able to detect both very prominent aponeuroses as seen in figure 3.6, and fainter aponeuroses as the lower aponeuroses in figure 3.1 and 3.7. It also needs to be able to detect the dominant angle in images containing many fascicles, and in images containing very few fascicles. In some of the images the fascicles are fragmented and the contrast between the background and the fascicles are low. A good example of this can be seen in figure 3.4.

3.2 Simulated ultrasound images of skeletal muscles

We also have two datasets containing simulated ultrasound images with varying amounts of noise. This is useful for measuring the robustness of the algorithm because the underlying structures are the same in the images in each of the simulated datasets. The method used for this was proposed by Yu and Acton [26]. The images are constructed by changing the contrast between the fascicles and the background in the phantom image before multiplying this image with a random grid of scatterers. The resulting image is convolved with a point spread function of a certain width. Some of the images are created by taking the average of a certain number of these simulations, called compounding. The absolute value of the Hilbert transformed image is calculated to avoid complex values. Finally a standard logarithmic scaling is done. Table 3.1 illustrates the effect of different parameters on the simulated image. Each row in this table shows images where only the specified parameter was changed.

The first simulated dataset was based on the phantom image shown in figure 3.9. The second simulated dataset was based on the phantom image shown in figure 3.10 and differs from the other in that it has structures going in different directions than the fascicles. Structures like this can be present in ultrasound images because of patches of fat or connective tissue. Both datasets have a Contrast Ratio (CR) of 0.25.

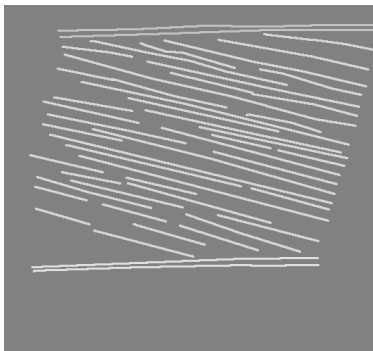


Figure 3.9: Phantom image from which the first simulated dataset is created from.

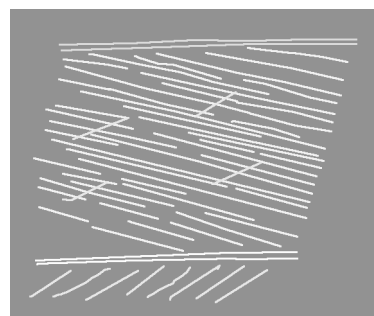


Figure 3.10: Phantom image from which the second simulated dataset is created from.

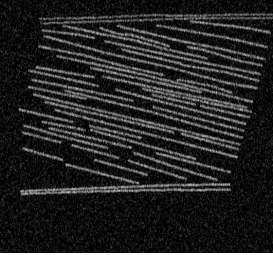
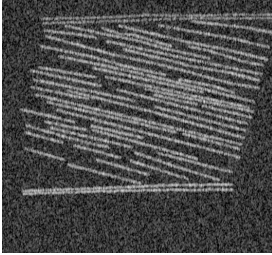
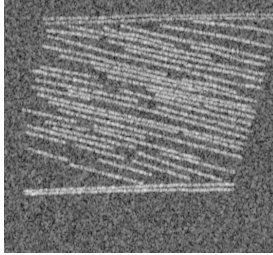
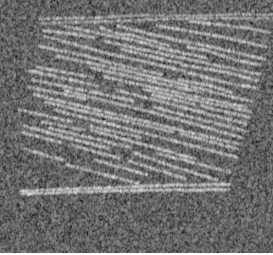
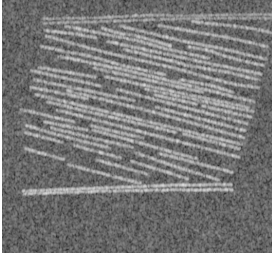
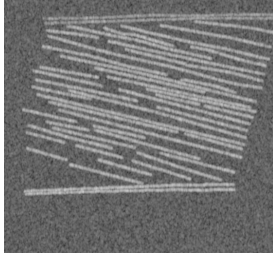
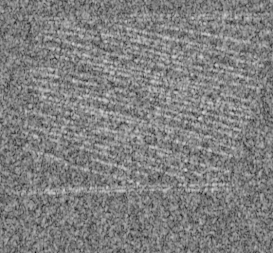
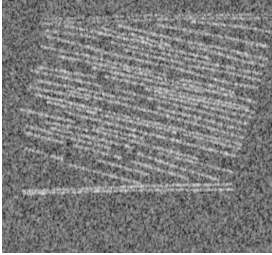
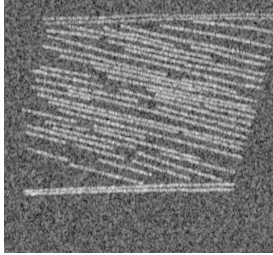
		
Psf width 1	Psf width 1.5	Psf width 2.5
		
Compounding of 3 images	Compounding of 5 images	Compounding of 10 images
		
Scaling factor 1	Scaling factor 1.5	Scaling factor 2

Table 3.1: Each row of this table shows how the different parameters affects the simulated images. In the first row the only varying parameter is the point spread function. In the second row only the number of compounded images varies. In the third row the scaling factor is the only parameter that changes.

Chapter 4

Methods

The algorithm developed consists of six main steps; determining the region of interest, performing directional filtering of the images, detecting aponeuroses, detecting fascicles, reconstructing a fascicle and finally calculation of pennation angle and fascicle length. This chapter will describe methods used to perform these steps.

4.1 Automatic detection of region of interest

Because some of the images in the datasets contain a lot of information in addition to the ultrasound image, we need to define the region of interest (ROI).

The means of the columns and rows in the image are used to determine the region of interest. We start with determining the upper and lower border of the ROI. To do this we start by looking for rows in the upper half of the image that have a mean value below a threshold. If there are no rows with a mean value below the threshold, the threshold is increased and the same procedure is repeated. Otherwise the rows with a mean value below the threshold are located. Of these rows, the row located lowest in the image is chosen. The upper border of the ROI is set to a few rows below this to make sure the region of interest only contains the ultrasound image. This is important because the sharp edge between the background and the ultrasound image can affect the result when we later want to detect the upper aponeurosis. The same thing is done to find the end row of the ROI, but now the lower half of the image is used. The rows with a mean value below the threshold are located, and the row located furthest up in the image is chosen. The border of the ROI is set to a few rows above this.

The image is cropped at the start and end row. This is done to remove the header and the information at the bottom of the image that will affect the mean value of each column. Then the left and right borders of the ROI are determined using the same algorithm as for the rows, using the columns.

In addition to setting a lower threshold, we also need to set an upper threshold for determining when the algorithm should stop looking for the border of the ultrasound image. If the algorithm reaches this upper threshold, the start/end row/column of the image should be set as the borders of the ROI. This allows the algorithm to choose the appropriate ROI in cases where the image contains additional information and in cases where it

only contains the ultrasound image. This makes the algorithm more versatile and tolerant to different types of input images. The thresholds should be determined based on the range of pixel values in the image.

This method requires that the upper and lower borders of the ultrasound image are located in the upper and lower half of the full image respectively. Figure 4.1 shows an example of the detected ROI with the starting mean value threshold set to 1. The pixel values in this image ranges from 0 to 255.

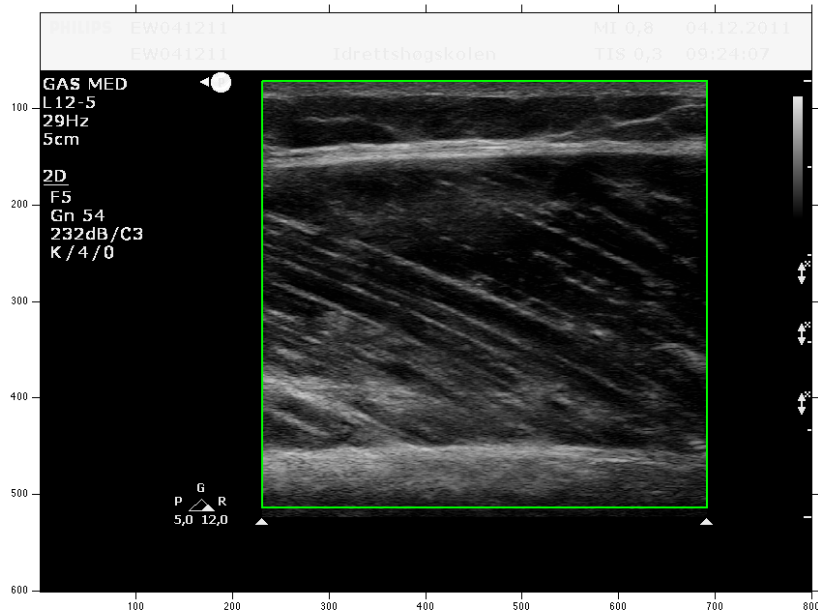


Figure 4.1: Region of interest marked in with the green rectangle. Image with pixel values ranging from 0 to 255 and mean value threshold set to 1.

4.2 Directional filtering of the images

The presence of speckle noise makes the detection of fascicles and aponeuroses more challenging. The goal with this first step is to use directional filtering to enhance the muscle fibers and reduce speckle noise in order to make the fascicle and aponeuroses detection more reliable. The following sections will describe different directional filtering methods tested to achieve this goal.

4.2.1 Sobel filter

The first method considered is using Sobel filters to find the gradient in x- and y-direction to enhance the edges in the image. The Sobel filter can be defined as the following gradient

operators:

$$S_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (4.1)$$

To make the filter less prone to noise, the Sobel filter is convolved with a Gaussian filter, h , resulting in the filters, $h_x = S_x * h$ and $h_y = S_y * h$. The image, I , is convolved with the resulting filters, h_x being the gradient in x-direction and h_y being the gradient in y-direction. The magnitude and direction is calculated from the gradient images, I_x and I_y .

$$\begin{aligned} I_x &= I * h_x \\ I_y &= I * h_y \\ I_m &= \sqrt{I_x^2 + I_y^2} \\ I_a &= \arctan \frac{I_y}{I_x} \end{aligned} \quad (4.2)$$

where I_m is the magnitude of the gradient and I_a is the angle of the direction with the least contrast.

4.2.2 Frangi Vessel filtering

Frangi et al. [8] developed a method for enhancing blood vessels in magnetic resonance angiography images and digital subtraction angiography images. This method searches for geometrical structures which can be regarded as tubular. The structure of vessels and muscle fibers are both tubular, so the method should be able to detect and enhance the muscle fibers in our images. Rana, Hamarneh and Wakeling [21] and Wakeling and Randhawa [24] both used this method with good results in B-mode ultrasound images of muscles. We will only describe how this is done in 2D images, but this can also be applied to 3D images.

While the Sobel-filter uses the first derivative, the Frangi filter uses the second-order partial derivatives/the Hessian matrix. The Frangi algorithm uses Gaussian second derivative filters to achieve this. They are shown in figure 4.2 and are given by the following equations:

$$\begin{aligned} h_{xx}(i, j) &= \frac{1}{2\pi\sigma^4} \frac{i^2}{\sigma^2 - 1} e^{-\frac{i^2+j^2}{2\sigma^2}} \\ h_{xy}(i, j) &= \frac{1}{2\pi\sigma^6} ije^{-\frac{i^2+j^2}{2\sigma^2}} \\ h_{yy}(i, j) &= \frac{1}{2\pi\sigma^4} \frac{j^2}{\sigma^2 - 1} e^{-\frac{i^2+j^2}{2\sigma^2}} \end{aligned} \quad (4.3)$$

where i and j are pixel coordinates. The range of i and j are based on the scale σ . To obtain the Hessian matrix for each pixel the image is convolved with these three filters (equation 4.4). The Hessian matrix, H , for each pixel are then formed from these three matrices:

$$\begin{aligned}
I_{xx} &= \frac{\partial^2 I}{\partial x^2} = I * h_{xx} \\
I_{xy} &= \frac{\partial^2 I}{\partial x \partial y} = I * h_{xy} \\
I_{yy} &= \frac{\partial^2 I}{\partial y^2} = I * h_{yy}
\end{aligned} \tag{4.4}$$

$$H_{i,j} = \begin{bmatrix} I_{xx}(i,j) & I_{xy}(i,j) \\ I_{xy}(i,j) & I_{yy}(i,j) \end{bmatrix}. \tag{4.5}$$

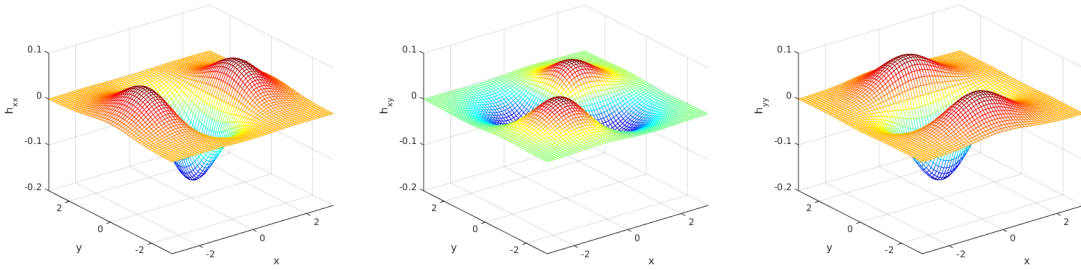


Figure 4.2: Plot of the three filters of size 7×7 ($\sigma = 1$) that the image is convolved with to obtain the Hessian matrix for each pixel. Left: h_{xx} , middle: h_{xy} , right: h_{yy} .

The principal directions are extracted from the eigenvalues and eigenvectors of the Hessian matrix in each pixel. From this we can find the direction with the least variation. The eigenvector corresponding to the eigenvalue with the largest magnitude points in the direction of maximum variance. To find measures that gives information about the structure we are looking for, we also have to know that these are bright tubular structures in a darker environment. We are therefore interested in regions where the smallest eigenvalue is close to zero and the second eigenvalue is negative with a high absolute value. The conditions are given by:

$$\begin{aligned}
|\lambda_2| &\approx 0 \\
\lambda_1 &< 0 \\
|\lambda_2| &\ll |\lambda_1|.
\end{aligned} \tag{4.6}$$

The eigenvalues are sorted in descending order, so $|\lambda_1| > |\lambda_2|$. From these we can define the ratio and "second order structureness" that helps us decide whether there is a tubular structure present or not. R (equation 4.7) indicates whether the area we are looking at contains a tubular structure or not. R will have a large absolute value in areas where there is a tubular structure, and will be closer to zero in areas where there is a blob-like structure or in homogeneous areas. S (equation 4.8) is the Frobenius norm of the Hessian matrix which will be large in areas with high contrast. R and S are given as:

$$R = \frac{\lambda_1}{\lambda_2} \quad (4.7)$$

$$S = \sqrt{\lambda_1^2 + \lambda_2^2}. \quad (4.8)$$

The filtered image is computed using the following equation:

$$I_o = \begin{cases} 0, & \text{if } \lambda > 0. \\ e^{-\frac{R^2}{2\beta^2}}(1 - e^{-\frac{S^2}{2c^2}}), & \text{otherwise.} \end{cases} \quad (4.9)$$

This equation was proposed by Frangi et al. [8] as a vesselness measure. β and c are parameters that control the sensitivity of the filter to the measures R and S respectively. These two criteria, R and S , are combined to ensure that the response of the filter is at a maximum when both criteria are fulfilled.

The approach described above is carried out multiple times with different scales of the Gaussian second derivative filters to be able to detect and enhance muscle fibers of different thickness. This scaling is done by changing the size of the grids X and Y in equation 4.3. The muscle fibers in one image will usually have approximately the same thickness, but this can vary between images. Also, the upper and lower aponeurosis will usually be thicker, and we want to be able to enhance these as well as the muscle fibers. The response of the filter will be maximum for the scale that best fits the width of the muscle fibers. After calculating the response for all different scales we want to choose the maximum value of all scales, see equation 4.10.

$$I_o = \max_{\sigma} I_o(\sigma) \quad (4.10)$$

To get the best result for our images, the different scales of σ was set to the range 1 to 3 with an interval of 0.5. β and c were set to 36 and 2500 respectively.

4.2.3 Non-local means

Non-local means is a denoising method that aims to remove noise while still preserve details. In our case we want to remove noise from the background, but not smooth out the fascicles. There are several different versions of the non-local means (NLM) method, e.g. patchwise NLM, pixelwise NLM, fast NLM and multiscale NLM. The method described in this section is the pixelwise implementation of the non-local means method.

The pixelwise non-local means method compares patches in the images to the neighbourhood around the pixel we want to denoise. The average of all the patches, weighted by the similarity, contributes to the final value of the current pixel.

The weight matrix represents the similarity between the neighbourhoods centered around pixels i and j and are measured using the Euclidean distance of the neighbourhood around each of the pixels, see equation 4.11. Given an image, I , the neighbourhoods are denoted by $I(N_i)$ and $I(N_j)$. σ is the standard deviation of the Gaussian kernel used to weight

the pixels in each neighbourhood, h denotes the degree of filtering and Z is a normalizing constant ensuring $0 \leq w(i, j) \leq 1$.

$$w(i, j) = \frac{1}{Z} e^{-\frac{\|I(N_i) - I(N_j)\|_{2,\sigma}^2}{h^2}} \quad (4.11)$$

The final pixel value is then calculated as an average of the pixel values multiplied with the similarity weight, $w(i, j)$. To avoid having to do too many calculations, we do not compare every single patch of the image for each pixel, but create two windows of different sizes. A smaller window for the size of the neighbourhood and a bigger window containing the patches we use for comparison.

The non-local means method was tested on the muscle images because a study conducted by Hofsvøy Breivik et al. [10] showed that a variant of this method gave very good noise filtering results when used on ultrasound images of the heart.

Parameters

This method was tested using parameters proposed in [19]. The examples that are shown in section 5.1.2 were obtained using a neighbourhood window size of 7×7 and a comparison window size of 35×35 . σ was set to 35 and h was set to 0.35σ . We did however not find a combination of parameters that worked well in reducing speckle noise while still maintaining the thin lines in the image.

4.2.4 Gaussian directional filters along the dominant direction

Find local dominant direction

In the Frangi vessel enhancement method the second-order partial derivatives were used to find lines in the image. In the gradient structure tensor method the covariance of the gradients are used instead. We start by forming gradient filters and convolve the image in x and y direction just like the Sobel method. For more details, see section 4.2.1. We are then left with three gradient images; g_1 , g_2 and g_3 .

$$\begin{aligned} g_1 &= I_x^2 \\ g_2 &= I_x I_y \\ g_3 &= I_y^2 \end{aligned} \quad (4.12)$$

The tensor matrices $T_{i,j}$ for each pixel are formed from these three images. This is done by summing the pixels of the gradient images in a weighted neighbourhood, N , centered around each pixel (i, j)

$$T_{i,j} = \begin{bmatrix} \sum_{x,y \in N} w(x,y) g_1(x,y) & \sum_{x,y \in N} w(x,y) g_2(x,y) \\ \sum_{x,y \in N} w(x,y) g_2(x,y) & \sum_{x,y \in N} w(x,y) g_3(x,y) \end{bmatrix}. \quad (4.13)$$

We find the eigenvectors and eigenvalues of $T_{i,j}$ and sort the eigenvalues, λ_1 and λ_2 , in descending order along with the corresponding eigenvectors, v_1 and v_2 , so that $|\lambda_1| > |\lambda_2|$. v_1 represents the direction with the biggest variance. The direction orthogonal to this

vector points in the direction along the edges. The two following measures are calculated using the eigenvectors and eigenvalues:

$$\begin{aligned} I_\theta(i, j) &= \frac{\text{atan2}(v_1(1), v_1(2)) * 180}{\pi} \\ I_c(i, j) &= \frac{\lambda_1}{\lambda_1 + \lambda_2} \end{aligned} \quad (4.14)$$

where I_θ contains the angle orthogonal to the direction along the muscle fibers in each pixel and I_c contains the certainty that this direction is correct.

Filter the image based on direction

The image is filtered based on the information acquired about the direction and the certainty, I_c , of this direction. A Gaussian vertically elongated filter is rotated to fit the direction found for each pixel. The size of this filter is set according to the certainty; large filter if the certainty is high and smaller filter if the certainty is lower. The scaling of the filter size is only done in the vertical direction, i.e. in g_v in equation 4.15. This will filter the image along the edges given that the direction estimation is reliable. The filter for a certain pixel (i, j) is constructed by convolving two one dimensional Gaussian filters and rotating it based on the dominant angle I_θ . The Gaussian filters g_v and g_h are given by:

$$\begin{aligned} g_h(x) &= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} && \text{for } x = 1, \dots, ls \\ g_v(y) &= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y^2}{2\sigma^2}} && \text{for } y_{i,j} = 1, \dots, lb * I_c(i, j) \end{aligned} \quad (4.15)$$

where $ls \leq lb$.

Parameters

We used a Gaussian matrix as the weight matrix w in equation 4.13 and set ls and lb in equation 4.15 to 5 and 41 respectively. The values in the certainty matrix, I_c , ranges from 0 to 1. σ was set to 4.

4.2.5 Knutsson tensor filters

Find local dominant direction

Knutsson and Andersson [13] proposed a method for estimating local structure and orientation using a set of spherically separable quadrature filters. The general idea of this method is to filter the image with filters spanning different directions in the Fourier space. The output of these filters are combined to estimate the direction with highest variance. This method allows us to detect all possible angles without actually calculating the response for all angles.

The two dimensional quadrature filters, B_{n_i} , are defined in the Fourier domain and given by:

$$B_{n_i}(\omega) = e^{-\frac{4}{b^2 \ln(2)} \ln^2(\omega/\omega_i)} D_{n_i}\left(\frac{\omega}{\|\omega\|}\right) \quad (4.16)$$

$$D_{n_i}(\omega) = \begin{cases} (n_i^T n_i)^2, & \text{if } (n_i^T n_i) > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (4.17)$$

where b is the bandwidth, ω_i is the center frequency and n_i is the vector pointing in the direction we want to filter [22]. The minimum number of filters required is given by $K = \frac{1}{2}d(d+1)$ where d is the dimensionality of the data. In 2D three filters are needed. Figure 4.3 shows the three filters used when processing the 2D ultrasound images.



Figure 4.3: Fourier specter of the quadrature filters with center frequency $\omega = \pi * 2^{-2}$, bandwidth $b\omega = 1$ and direction $n_1 = [1, 0]^T$, $n_2 = [\frac{1}{2}, -\frac{\sqrt{3}}{2}]^T$ and $n_3 = [\frac{1}{2}, \frac{\sqrt{3}}{2}]^T$. The Fourier domain is shifted so that the zero frequency is in the center of the image.

The image, I , is filtered with each of the complex quadrature filters in the Fourier domain and then transformed back to the image domain yielding three images, q_1 , q_2 and q_3 :

$$q_i = |\mathcal{F}^{-1}(\mathcal{F}(I) \circ B_{n_i})|^2 \quad (4.18)$$

The quadrature filters are complex, so we take the absolute value of the filtered images to avoid complex values in q_i . The filtered images, q_i , are convolved with a smoothing filter calculating the weighted average in a neighbourhood around each pixel:

$$q_{ig} = q_i * f \quad (4.19)$$

where f is a smoothing filter. We used a Hamming filter for this.

To find an estimate of the local orientation we use the eigenvector decomposition of the tensor matrix. The tensor matrices for each pixel are formed using equation 4.20 where M_K tensors are given by equation 4.21.

$$T_{i,j} = q_{1g}(i,j) * M_1 + q_{2g}(i,j) * M_2 + q_{3g}(i,j) * M_3 \quad (4.20)$$

$$M_i = \frac{4}{3}(n_i n_i^T) - \frac{1}{3} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (4.21)$$

The local dominant direction and the certainty of this direction are calculated from the eigenvector decomposition of the tensor matrices $T_{i,j}$ as described in the gradient structure tensor method (section 4.2.4).

Filter the image based on directed highpass filters

We create a Gaussian lowpass filter, L , and set of highpass filters, H_i . The lowpass filter is a Gaussian filter defined in the Fourier domain. The highpass filters are based on B_{n_i} and given by:

$$H_i = (1 - L) \circ B_{n_i} \quad (4.22)$$

where \circ denotes the pointwise product. Figure 4.4 shows the four filters used in our 2D images in the Fourier domain.

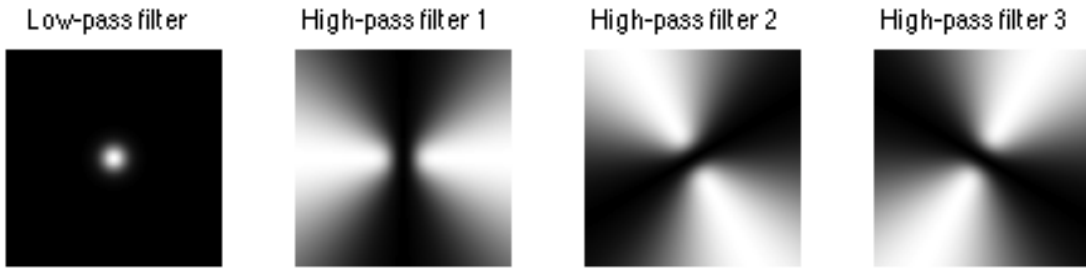


Figure 4.4: Lowpass and highpass filters in the Fourier domain.

The image is filtered in the Fourier domain with these four filters and then transformed back to the image domain. Only the real part of these filtered images are used. This gives us one lowpass image output, I_L , and three directional highpass outputs, I_{H_1} , I_{H_2} and I_{H_3} . A control tensor are used to weight the contribution of each of the directed highpass filtered images to obtain the final highpass filtered output, H , given by equation 4.23. The response of the highpass filters are used as weights to calculate the filter response in the dominant direction.

$$H = \frac{(I_{H_1}(cq_{1g} + dq_{2g} + dq_{3g}) + I_{H_2}(dq_{1g} + cq_{2g} + dq_{3g}) + I_{H_3}(dq_{1g} + dq_{2g} + cq_{3g})) * m}{I_{\lambda_1}} \quad (4.23)$$

I_{λ_1} and I_{λ_2} are matrices containing the largest and smallest eigenvalue computed at each pixel respectively. c , d and m are given by:

$$\begin{aligned} c &= \text{trace}(M_1' M_1) \\ d &= \text{trace}(M_1' M_2) \\ m &= \frac{t^b}{t^{a+b} + n^b} \\ t &= \sqrt{\frac{\sqrt{I_{\lambda_1}^2 + I_{\lambda_2}^2}}{Z}} \end{aligned} \quad (4.24)$$

where a , b and n are adjustable parameters and Z is a normalizing constant. The sum of the lowpass filtered output and the highpass filtered output will give us a directional smoothed image which reduces noise while keeping the sharp edges of the muscle fibers.

Parameters

Different values for the adjustable parameters have been tested to find the ones that gives the best result for the relevant images for this thesis. Bandwidth, b , was set to 1. Center frequency, ω_i , was set to $\pi * 2^{-2}$. The smoothing filter, f , used to obtain q_{ig} prior to orientation estimation was set to a Hamming filter of size 31×31 . σ used to create the lowpass filter in figure 4.4 was set to 11. The parameters a , b , and n was set to 0.01, 2 and 0.1 respectively. a , b and n are usually constants set to these values.

4.3 Line detection

4.3.1 Hough transform

The Hough transform is an algorithm used to detect lines, circles and other objects in an image. The only requirement is that the parametric equation of the object is known. The idea was originally proposed by Hough [4], hence its name, and was further developed by, among others, Duda and Hart. In this section I will present the algorithm proposed by Duda and Hart [7] using polar coordinates instead of the original Cartesian coordinates.

Before applying the Hough transform to the image we have to create a binary gradient image. This can for example be done using edge detection algorithms like Sobel, Canny or Prewitt. A suitable threshold for creating a binary image containing the edges must be applied before moving on. The binary image should contain ones representing the edges in the image and zeros everywhere else. We want to find sets of pixels that combined makes up a straight line. Given a point (x_i, y_i) , the polar representation of a straight line is given by

$$x_i \cos(\theta) + y_i \sin(\theta) = \rho. \quad (4.25)$$

Figure 4.5 shows what θ and ρ represents. ρ is the distance from the left upper corner of the image. θ is the angle from the vertical axis. The line detected is the line orthogonal to the line given by ρ and θ . An accumulator matrix representing the parameter space is created to keep track of the possible lines. For each of the edge pixels in the binary image, the parametric equation 4.25 is calculated for each θ and the accumulator matrix is incremented at entry (ρ, θ) . Each edge pixel will create a sinusoid in the (ρ, θ) -plane. If there is a line present, the sinusoids will intersect

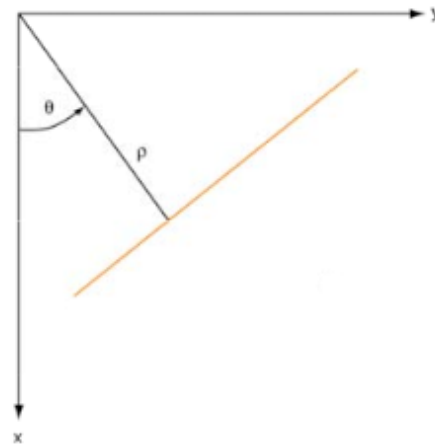


Figure 4.5: The geometry of the Hough transform. The yellow line is the detected line given ρ and θ .

in a point. This results in a local maximum in the accumulator matrix. These peaks give the polar coordinates of the most significant lines in the image.

The Hough transform is robust to noise and works well even if the lines are discontinuous, which they often are in ultrasound images. The main disadvantage with the Hough transform is that the input image needs to be binary. To achieve good results using this method we therefore need to find the optimal threshold value to threshold the gradient magnitude image.

4.3.2 Radon transform

The Radon transform eliminates the problem of finding the optimal threshold value because any gray level image can be used as input. The Radon transform [9] is defined as

$$R(\theta, r) = \iint f(x, y) \delta(x \sin \theta - y \cos \theta - r) dx dy \quad (4.26)$$

where θ is the angle relative to the y-axis, r is the distance from the center of the image, f is the input image and δ is the Dirac delta function.

The basic idea of the Radon transform is to rotate the image an angle θ and then sum the pixels in each vertical line. This is done for all angles, θ , specified. In the final Radon transform matrix the columns represent one angle each and the rows represents the distance from the summed image column to the center of the image. An example of the Radon transform can be seen in figure 4.6. Figure 4.7 shows what r and θ represents in the image.

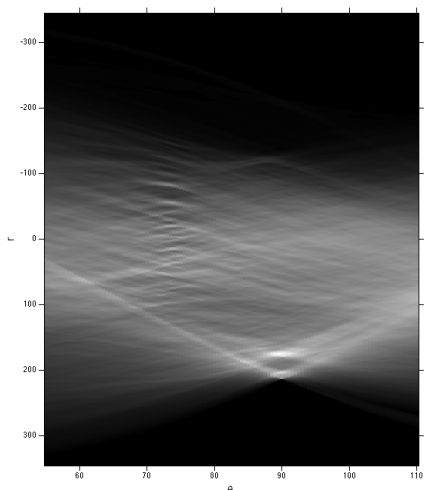


Figure 4.6: An example of how the Radon transform looks like.

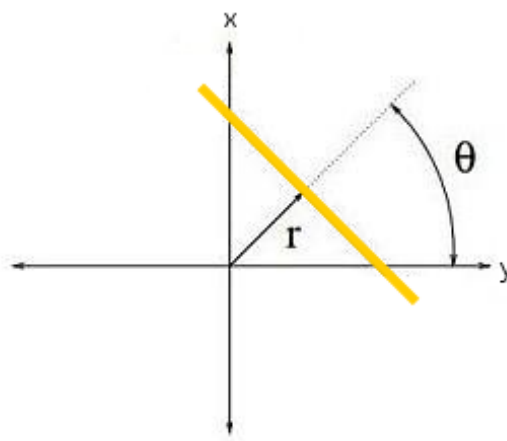


Figure 4.7: The geometry of the Radon transform.

In the Hough transform we could locate the lines in the image by finding the peaks in the Hough transform. This is not a sufficient measure in the Radon transform, but the advantage with this method is that we can read the Radon transform matrix in the way that best fits the structure we are looking for. In the Hough transform this structure (e.g. lines,

circles) must be defined prior to the transform, but with the Radon transform this can be done after the transform. The features we are interested in includes fascicle direction and aponeuroses direction. Because the appearance of these two features are a bit different in the images, two different methods are used to detect these in the Radon transform matrix.

The fascicles will have parallel bright and dark elongated lines. To detect these in the Radon transform, we integrate over each θ to find the angle with the highest variance as proposed by Rana, Hamarneh and Wakeling [21]. When the image is rotated so that it is aligned with the fascicle direction, the summed columns will contain low values for the areas between the fascicles and higher values for the areas containing fascicles. When it is rotated perpendicular to the orientation of the fascicles, all the summed columns will have similar values. The standard deviation or variance of each θ in the Radon transformation matrix are therefore good measures for finding the direction of the fascicles. After finding the θ that fits this observation, the highest value of this column is used to find the location of the most prominent fascicle.

The aponeuroses are often represented as thicker lines in the ultrasound image compared to the fascicles, and will not always have parallel darker areas. This means that the aponeuroses will be represented as blobs in the Radon transform. We therefore chose to use cross correlation, which is a similarity measure, with a Gaussian kernel to locate the blobs in the Radon transform. The highest value within the area most similar to the Gaussian kernel was chosen as the location of the aponeurosis.

Normalized Radon transform

For the Radon transform not to favor longer lines, it can be normalized by dividing each pixel in the Radon transform by the length of the line given by θ and r .

4.3.3 Choice of line detection method

The Radon transform will be used for detecting lines in this thesis. Both the Hough transform and the Radon transform were tested on the images, but it was difficult to create good thresholded binary images. As the Hough transform requires a binary input image, we found this method too limiting, and we therefore chose to use the Radon transform to avoid this limitation.

4.4 Detecting upper and lower aponeurosis

The upper and lower aponeurosis need to be detected in order to estimate the fascicle length and pennation angle. This is also necessary to estimate the region of interest for fascicle detection, which is the area between the aponeuroses called the fascicle-plane. The aponeuroses can be seen in the ultrasound image as the outlined thicker bright lines at the top and bottom of the image in figure 2.2.

As mentioned previously, there are some challenges related to aponeuroses detection. Differences between images was the main challenge. An example of this can be seen in figure 3.1 and 3.3. The upper aponeurosis is quite prominent in both images, but the

lower aponeurosis is very faint in the first image while it is as prominent as the upper aponeurosis in the second image. Another example can be seen in figure 2.2 where neither the upper or lower aponeurosis are particularly prominent compared to the energy in the rest of the image. We have tried a few different methods to detect the aponeuroses that I will explain and discuss in the following sections. The detection is divided into two steps. The first step finds the approximate location of the aponeuroses, and crops the image so that it contains one of the aponeuroses. The second step uses the Radon transform of this cropped image to locate the exact location of the aponeuroses.

4.4.1 Approximate aponeuroses detection

Approximate aponeuroses detection by pixel brightness

The first method considered was based on finding peaks in the pixel values at the bottom and top of the filtered image. This is done for each column, and then a line is fitted to these peaks to represent the aponeuroses. This method assumes that the aponeuroses is the brightest parts of the image, or at least that it stands out compared to the rest of the image. This is not always the case, especially when it comes to the lower aponeuroses. It will often be very faint because the ultrasound waves will attenuate as it travels through the muscle tissue. For the upper neurosis detection, the biggest challenge using this method was to differ between the upper aponeuroses and the skin layer. The skin layer appears as a bright line, usually located close to the upper aponeurosis.

Approximate aponeuroses detection using the Radon transform

A second method considered was to use the Radon transform to determine the approximate location of the aponeuroses. The Radon transform is calculated in a local horizontally elongated window at linearly spaced locations in the image. We used a window size of 30x70 pixels. Since the aponeuroses are fairly horizontal and therefore appear as longer lines in the image, we do not normalize the Radon transform.

An accumulation matrix with columns representing angles and rows representing image depth is used to keep track of the dominant angles at different depths of the image. The dominant angle is located from the Radon transform of each window. The maximum value of the dominant angle, found in the Radon transform matrix, is added to the bin representing the current image depth and the dominant angle in the accumulation matrix. From the resulting accumulation matrix we want to find the approximate depth of upper and lower aponeurosis. Two different methods was considered to locate this approximate depth.

Locating peaks for angles around 90 degrees. The peaks in the accumulation matrix oriented at 90 ± 10 degrees (close to horizontal lines) is located. These peaks provides an approximate location of the upper and lower neurosis. Figure 4.8 shows what the accumulation matrix looks like and what part of this matrix is used to find the upper and lower aponeurosis.

This method requires the assumption that the aponeuroses is approximately horizontal. It also requires that the fascicles have an angle of more than ± 10 degrees for them to

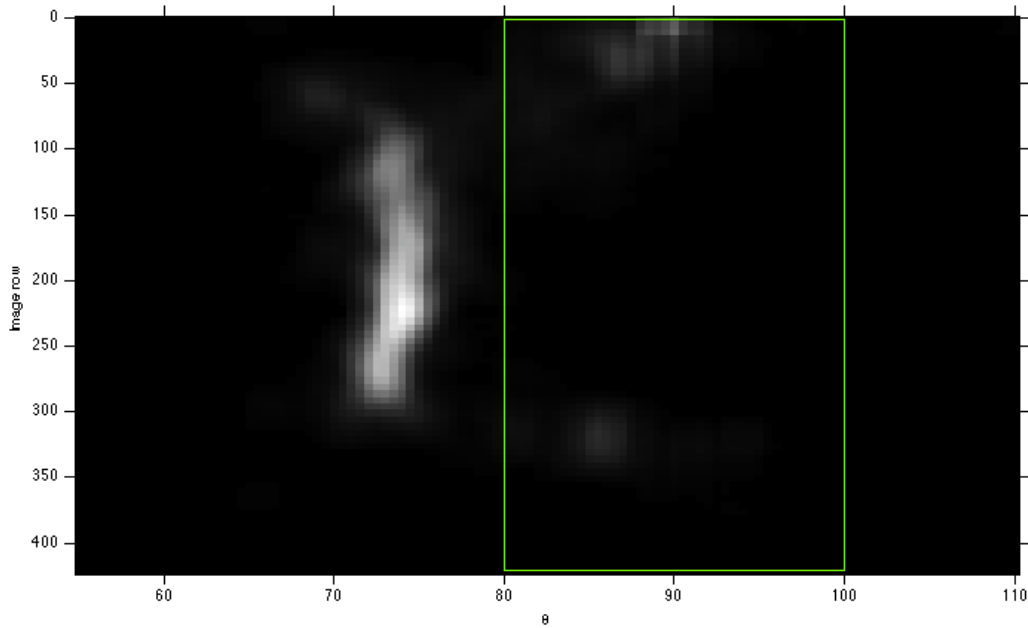


Figure 4.8: Image showing the accumulation matrix, smoothed with a Gaussian filter, used for keeping track of dominant angles at different depths. Green rectangle showing the area used for locating upper and lower aponeurosis when assuming it is within 90 ± 10 degrees.

not be present as peaks in the part of the accumulation matrix we are looking at (the green rectangle in figure 4.8). Both these assumptions will not always hold. If the first approximate localization fails so that the cropped image does not contain the aponeuroses, the more exact localization will obviously fail as well.

Locating changes in dominant angle. To eliminate the assumption that the aponeuroses needs to be within 90 ± 10 degrees we can, instead of looking for lines around 90 degrees, look for a change in dominant angle for each depth. A typical plot showing the most frequent angles for each depth will look something like figure 4.9. The red cross marks the depth believed to contain the aponeuroses. This depth is determined by finding a place where the difference in angle is sufficiently large (determined by a threshold) for us to regard it as a change from fascicle direction detection to aponeuroses direction detection. If there are two such changes in direction, we choose the midpoint between these as the approximate location. The distance between these changes in direction are also measured for later use.

To find the approximate location of the upper aponeurosis we use the most frequently detected angles in each depth of the upper half of the image. For the lower aponeurosis we use the lower half. We therefore assume that the aponeuroses are located at opposite sides of the center of the region of interest. For this method we need to set a threshold for how big the change in angle between fascicles and aponeuroses should be. We have used a threshold of 4 degrees which means that we assume the fascicle direction and aponeuroses direction differs from each other with at least 4 degrees. This is an assumption that holds

for most images, and we therefore chose this method for finding the approximate location of the aponeuroses.

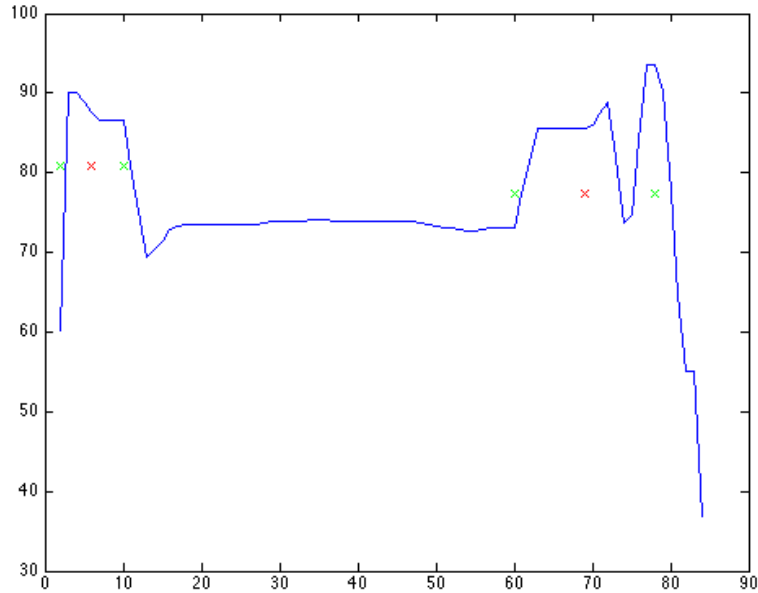


Figure 4.9: Plot showing the most dominant angle for each depth acquired from each row of figure 4.8 (x axis: Window row depth, y axis: dominant angle). Green x marking where a dominant change in angle was detected. Red x marking the middle of the area we believe the aponeurosis is located.

4.4.2 Exact aponeuroses detection

The approximate location of the aponeuroses is used to crop the image so that it contains all columns, but only a certain number of pixels above and below the approximate location of the aponeuroses. The height of the cropped image is determined by the distance between the detected changes in direction (obtained in the approximate aponeuroses detection). If only one significant change in dominant angle was detected, a default height is set for the cropped image. Figure 4.10 and 4.11 shows an example of what the cropped images will look like. The Radon transform is calculated for these cropped images to acquire the more precise location.

We chose to run the Radon transform on the directional highpass filtered output from the Knutsson filter when looking for the lower aponeurosis. The reason for this is that the lower aponeurosis are sometimes not as prominent as the upper aponeurosis, and therefore easier to detect in the highpass image. For the upper aponeurosis we use the highpass filtered image added to the lowpass filtered image because the aponeuroses would often get confused with the skin right above the upper aponeuroses when only using the highpass filtered image.

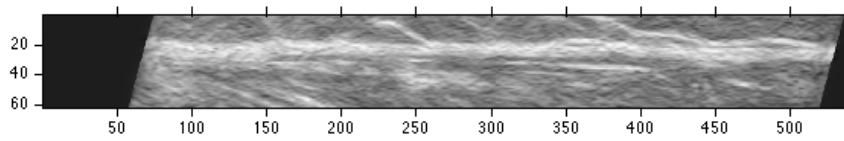


Figure 4.10: Cropped image containing the upper aponeurosis. Acquired using the method where we assume the aponeuroses are 90 ± 10 degrees.

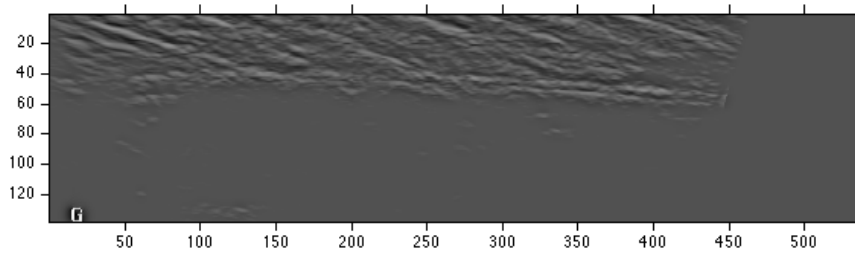


Figure 4.11: Cropped image containing the lower aponeurosis. Acquired using the method where we assume the aponeuroses are 90 ± 10 degrees.

4.5 Detecting fascicles

The goal with this part of the algorithm is to find the orientation of the fascicles in different parts of the image. This data is used for estimating the pennation angle and fascicle length. The image is cropped so that only the part of the image between upper and lower aponeurosis is processed.

To find the dominant orientations in different parts of the image we perform a normalized Radon transform in local windows linearly spaced throughout the filtered image. The normalized Radon transform is used because we want every possible line in each window to have equal probability of being detected. One dominant angle is determined for each window, which is then stored in a matrix for later use. The dominant angle is detected using the method explained in section 4.3.2. For areas where two or more windows are overlapping the mean of these angles are calculated. After this we are left with an image containing the dominant angle in a local area around each pixel.

In some of the windows, especially in homogeneous areas, the detected fascicle will be located in the corners of the images. This is a consequence of the normalized Radon transform. Because there are no prominent lines the values will be similar in the entire Radon transform. When we divide the Radon transform by the normalizing matrix, containing the length of the lines at each angle and location, the corners of the image will get a higher value because of the short length of the lines in these locations. We have chosen to discard these detections to increase the reliability of the algorithm. A repercussion of this choice is that non-prominent lines might be discarded as well. This is one of the reasons why enhancing the fascicles prior to the detection is essential.

The images often contain the presence of random patches of fat and connective tissue. These are sometimes elongated in different directions than the fascicles, and constitutes a significant obstacle to automated analysis. To try to take this problem into account and further improve reliability, we calculate the mean of the most frequent angles at a certain number of row depths. This mean is compared to each of the detected fascicles at the same row depths. If the detected fascicles deviates too much from this angle it is discarded.

We also know that the fascicles have a curved shape, but we do not want the fascicles to have an S-shape. To try to incorporate this requirement into the algorithm we start by calculating the mean angle for different depths of the image. This data gives us an indication of whether the fascicles are convex or concave. The mean angle for the upper grid depth is calculated and used for determining which fascicle detections should be discarded in the next grid depth. This is necessary in order to preserve the convex or concave shape of the fascicle. The detections that are not discarded are used to calculate the mean for the current grid depth, which in turn are used as a threshold for the next grid depth and so on.

Figure 4.12 shows an example of the fascicle detections through each of these three steps.

4.6 Construct a reference fascicle

After detecting the mean angle for different depths we want to reconstruct a fascicle based on the information gathered about the fascicle orientation and location of the upper and lower aponeurosis. In order to do this we need to assume that the dominant angle at different depths of the image can be transferred to the areas outside the field of view.

The fascicle plane is divided into grid cells. The matrix containing the fascicle detections is used to calculate the mean dominant angle in each of these grid cell. Figure 4.13 shows an example of the result after this step. A vertical line in a grid cell means that we do not have enough reliable data for this area.

The mean angle for each depth of the grid is calculated. The outermost columns of the grid are excluded from the calculation of mean angle because of difficulties with the detection in these areas. The resulting mean angles, representing each depth in the muscle, are used for distributing points in the image. These points are used for reconstructing a fascicle. The starting point for the top grid depth is the upper left corner of the grid. Points are distributed in a straight line based on the mean angle. The start point for the next grid depth is the point where the previous mean angle would have continued to the next grid depth. When this is done for all grid depths, the points are relocated so that the center column point are located in the middle of the image. See figure 4.14 for an example of this.

To construct a fascicle we fit a 2. degree polynomial, given by equation 4.27, to the distributed points. This is used to calculate the fascicle length which is explained in the next section. We chose to use a 2. degree polynomial to represent this line because the fascicles usually have a curved shape. Figure 4.15 shows an example of the resulting 2. degree polynomial.

$$f(x) = ax^2 + bx + c \quad (4.27)$$

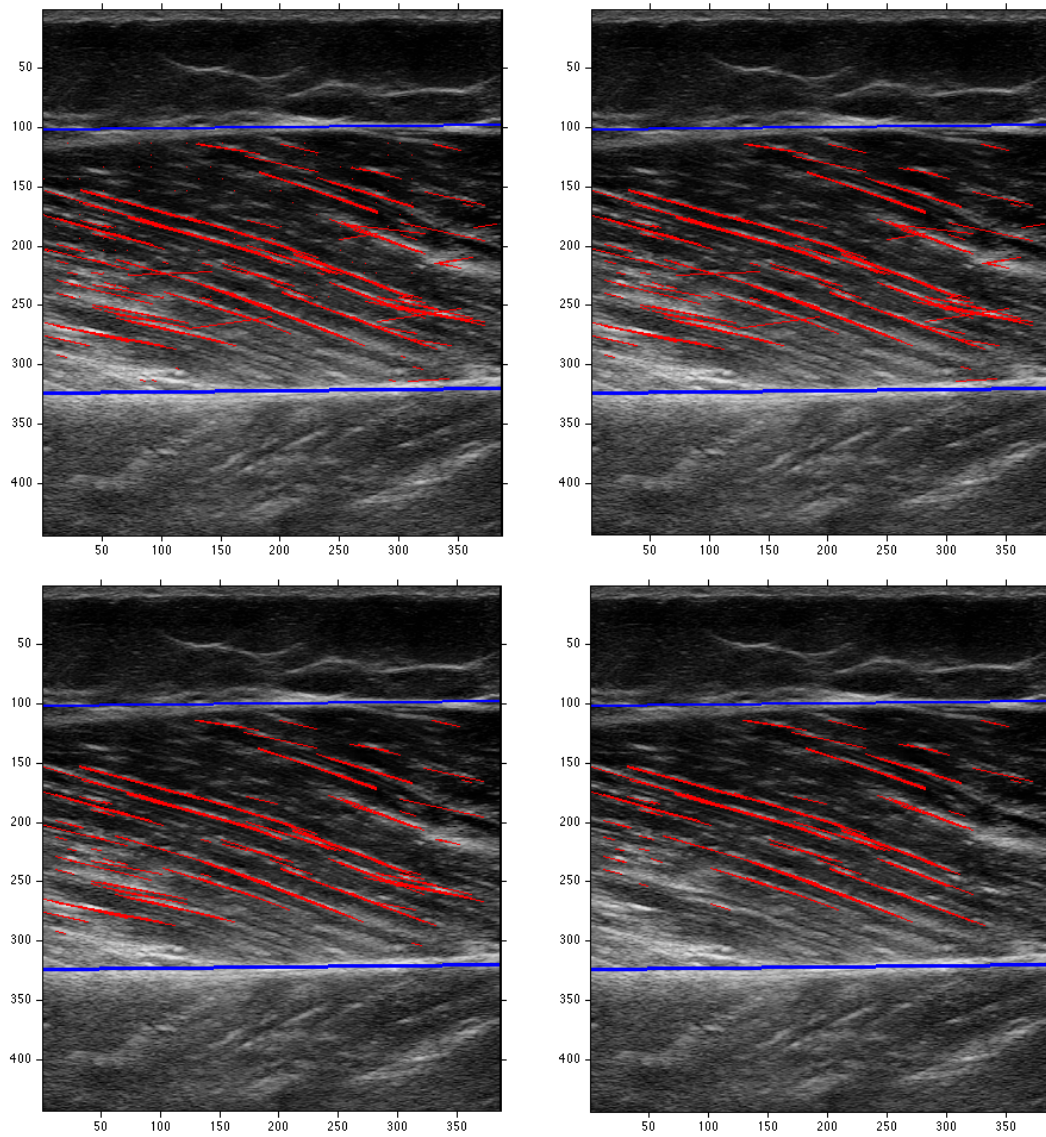


Figure 4.12: Top left: all detected fascicles, top right: removed unreliable fascicle detections, bottom left: removed deviating angles, bottom right: removed fascicles that contribute to an unwanted S-shape

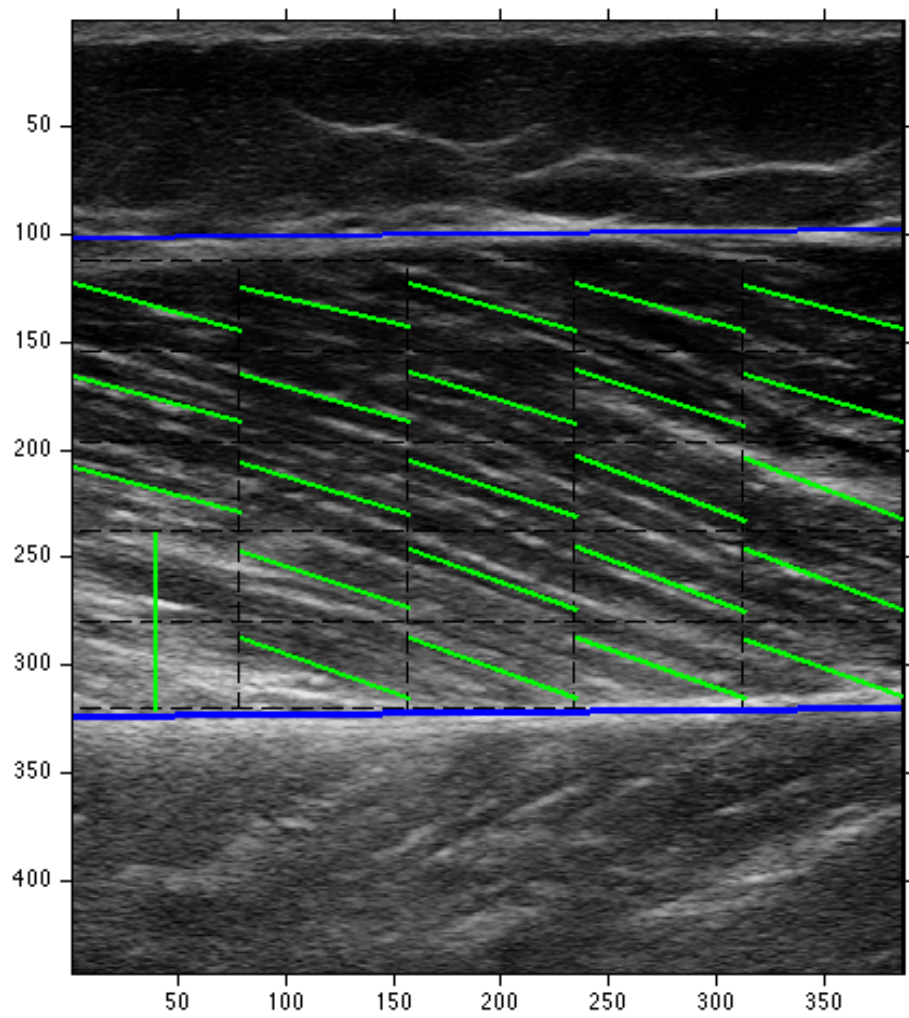


Figure 4.13: The mean angle of each grid plotted as green lines.

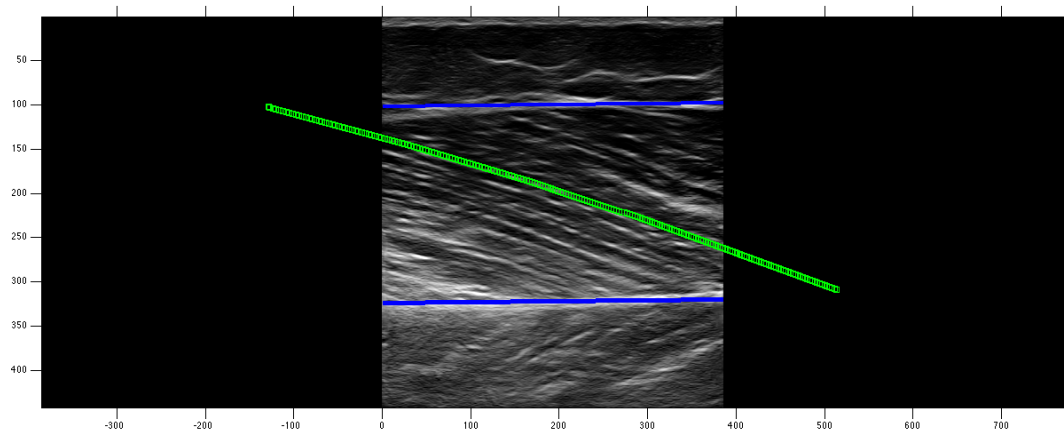


Figure 4.14: Distributed points based on mean angle for different depths marked in green

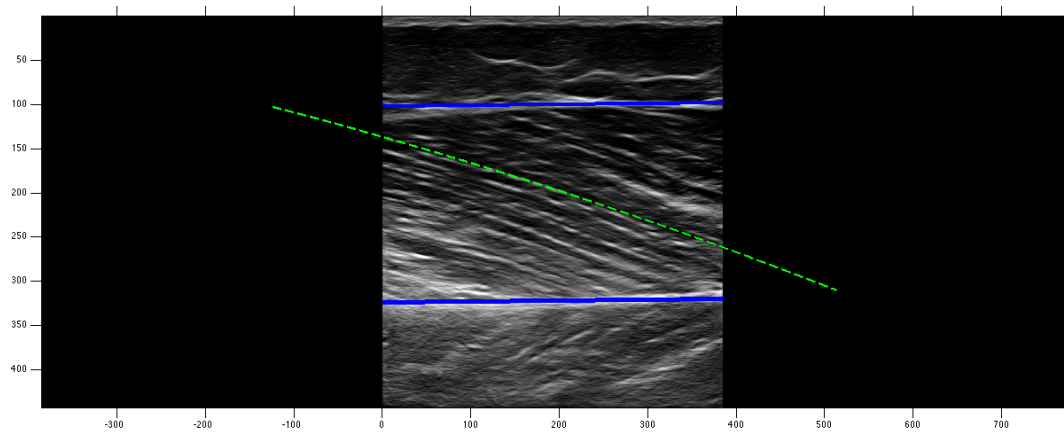


Figure 4.15: 2. degree polynomial fitted to the points in figure 4.14 plotted as green line

4.7 Calculate muscle features

As mentioned earlier, the interesting muscle features are pennation angle and fascicle length. Pennation angle is relatively variable from one fascicle to the other. This poses a challenge because we only want one angle describing this feature. The biggest challenge for estimating the fascicle length is that the entire length is almost never within the field of view, and we rely on geometrical extrapolations or simple trigonometry to estimate their length.

4.7.1 Calculate fascicle length

To calculate the fascicle length we need to find the points where the fascicle connects to the upper and lower neuroses. This was done by simply calculating the intersection of the polynomial and the aponeuroses using the formulas for these lines. The x-coordinate for the intersection with the upper neurosis, x_U , and the x-coordinate for the intersection with the lower neurosis, x_L , constitutes the limits of the integral in formula 4.28. The formula for the estimated fascicle, $f(x)$, is given by equation 4.27. To calculate the length of the curved fascicle, measured in pixels, we use the arc length formula which is given by:

$$length = \int_{x_U}^{x_L} \sqrt{1 - f'(x)^2} dx = \int_{x_U}^{x_L} \sqrt{1 - (2ax + b)^2} dx. \quad (4.28)$$

To convert the pixel length to actual length we need to know the scale of the ultrasound image. This information is present in text on top of the full ultrasound image, but I have not implemented automatic recognition of this.

4.7.2 Calculate pennation angle

The following method used for calculating pennation angle was chosen based on two currently used methods. A manual analysis method for measuring the pennation angle uses an angle at approximately two thirds into the fascicle plane. The method currently used by Norwegian Institute of Sport Science finds the dominant angle among all fascicles in the lower third of the image. Because we only want one estimated pennation angle, we need to calculate a reference fascicle. This is calculated as the median angle of the fascicles located two thirds into the fascicle plane. Figure 4.16 shows this area outlined in yellow. The pennation angle is calculated using the angle of the lower aponeurosis, θ_A , and the angle of the estimated reference fascicle plane, θ_F . The calculation of the pennation angle is given by:

$$angle = \theta_F - \theta_A. \quad (4.29)$$

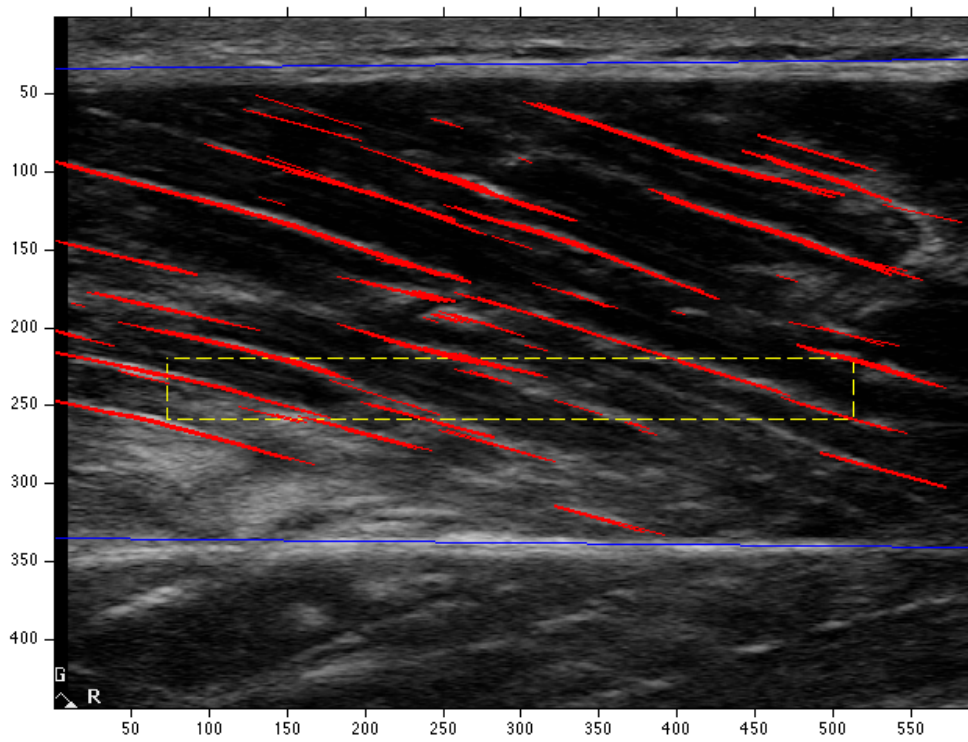


Figure 4.16: Median of the angles located inside the yellow rectangle is used to measure the pennation angle.

4.8 Overview of the algorithm

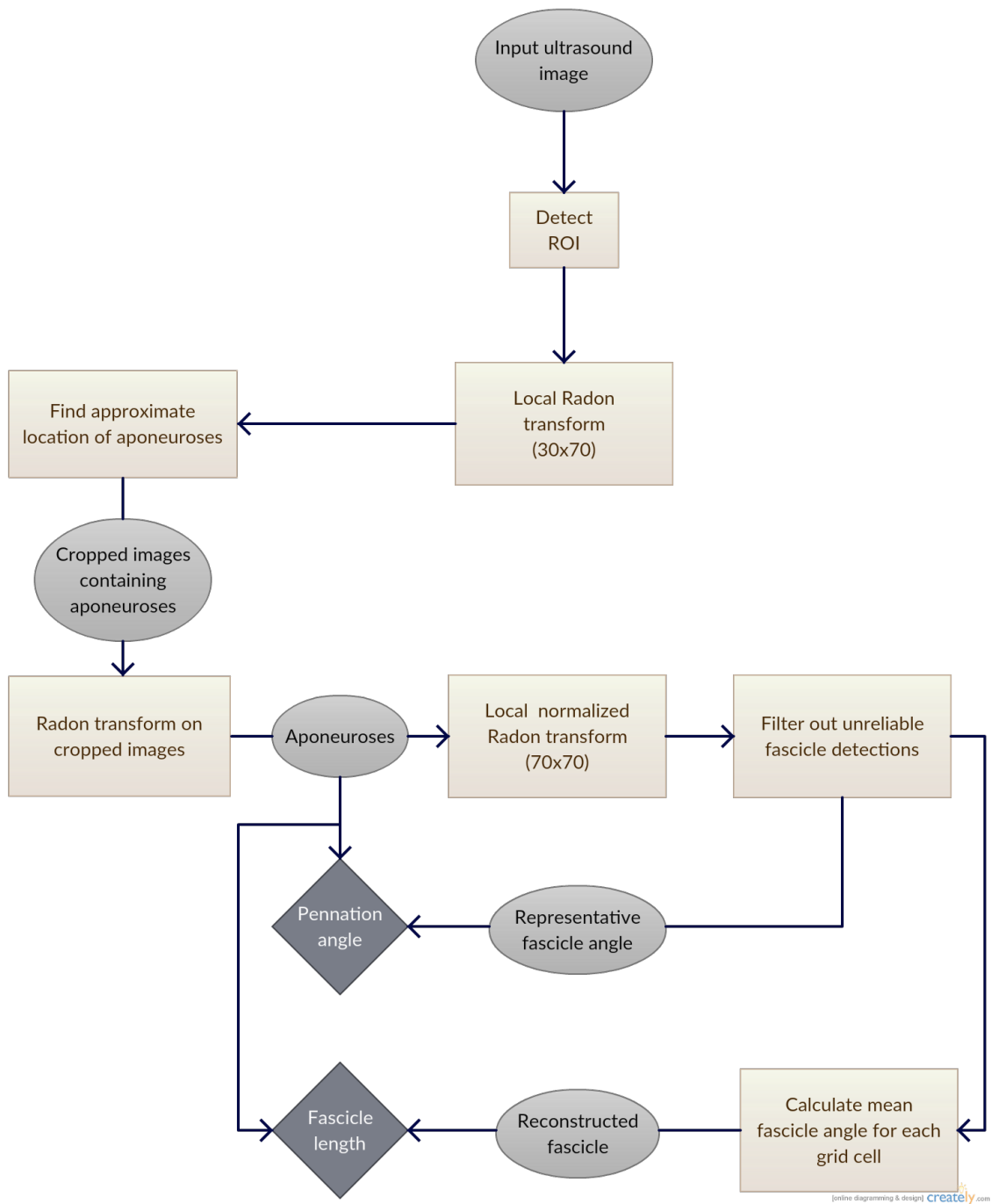


Figure 4.17: Flowchart of the algorithm.

Chapter 5

Evaluation of results

This chapter contains an evaluation of the methods presented in chapter 4. The filtering results are presented and evaluated. Then the results for each of the datasets introduced in chapter 3 are presented and discussed. Because the evaluation method is different for each of the datasets, the results for each dataset are presented separately.

5.1 Evaluation of structure enhancing filtering

This section contains the results for some of the filtering methods introduced in chapter 4. To illustrate the filtering methods, the filtering results are shown on the same two ultrasound images for all methods. The images are obtained from the dataset without manual analysis, one image from the good quality part of the dataset and one image from the part of the dataset with lower quality images. The original images are shown in figure 3.1 and 3.5.

5.1.1 Frangi vessel filtering

The Frangi vessel filter was used to enhance the bright lines in the image, which are the connective tissue of the fascicles. This provided us with an image with enhanced lines that the line detection method were able to detect. Figures 5.1 and 5.2 show two different results when using this method. The parameters used to achieve these results are given in section 4.2.2. The advantage with this method is that it both reduces speckle noise and enhances the fascicles and aponeuroses. The resulting contrast between the fascicles and the background is large, which is good. In areas without prominent fascicles and between fascicles some artefacts arose. These artefacts appeared as either very short lines or winding structures. Although fainter than the fascicles, they were present. The winding structures is useful for the intended purpose of enhancing blood vessels, but this effect is not beneficial when enhancing muscle fibers. An example of the short lines can be seen in figure 5.2, especially in the lower half of the image. The winding structures are visible throughout the entire image in figure 5.1.

5.1.2 Non-local means

Non-local means filtering was tested due to its good results in speckle reduction on ultrasound images of the heart [10]. It did not work as well on the images relevant to this

thesis. It was tested with different sizes of the patch used for comparison. We found that when trying to filter the image "lightly", using a smaller patch size, it created visibly filtered patches in areas where the energy were low, meaning areas without prominent fascicles. When trying to filter the image "harsher", using a bigger patch size, it reduced the contrast ratio between fascicles and the background, although removing speckle noise in a satisfactory way. Figures 5.3 and 5.4 shows the filtered images obtained using the non-local means method. The parameters used to obtain these results are given in section 4.2.3.

5.1.3 Gaussian directional filters along the dominant direction

The Gaussian directional filters worked well in reducing speckle noise and enhancing the fascicles, but the contrast between the fascicles and the background is low compared to the Frangi vessel filtering method. It also produced an undesirable filtering effect in areas where there was little energy. This effect is especially visible in the upper right corner of the image in figure 5.5. This area contains swirls that was not present in the original image. This effect usually occurred in images where parts of the background was inside the region of interest. Visual inspection of the certainty matrix, I_c , showed that the lowest certainty was located in areas outside the actual ultrasound image, i.e. below the lower aponeuroses and on the sides. This effect could probably have been reduced by finding a better scaling method for I_c in these cases. I_c determined the size of the filter used for each pixel. The scaling of I_c worked better in images like 5.6, where the entire region of interest contained the ultrasound image. To get the results shown in figure 5.5 and 5.6, the filter sizes ranged from 41x5 in areas where it was very certain of the direction to 5x5 in areas where the certainty was low. The filter sizes refer to the size prior to the rotation of the filter. The parameters used to achieve the results are given in section 4.2.4.

5.1.4 Knutsson tensor filters

The Knutsson tensor filtering method achieved good direction estimates and it gave us both a lowpass output image and a highpass output image. Visual inspection of the direction estimates indicated that this method gave more reliable direction estimates than the other methods. Figure 5.7 and 5.8 shows the lowpass output added to the highpass output. Because the highpass output is added to the lowpass output this method reduces the speckle noise, but still keeps the sharp edges in the image. Figure 5.9 and 5.10 show the results from the highpass filtering. The parameters used to achieve these results are given in section 4.2.5.

5.1.5 Choice of filter

Initial tests led us to using the Knutsson filtered image for further processing. The main reason for this was that this method provided us with two filtered images that could serve different purposes. The direction estimates was also better using the Knutsson approach. We ended up with using the directional highpass output for detecting the lower aponeurosis and the fascicles, and the lowpass added to the directional highpass output for detecting the exact location of the upper aponeurosis. The reason for this was that the upper aponeurosis detection algorithm sometimes confused the skin layer, which is located right above the upper aponeurosis, with the upper aponeurosis when using the

highpass filtered output. The highpass output was used for lower aponeurosis detection because the lower aponeurosis often are too faint to be detected by the Radon transform in the lowpass filtered image added to the highpass output. From now on I will refer to the lowpass output added to the directional highpass output as the Knutsson lowpass image, and the directional highpass output as the Knutsson highpass image.

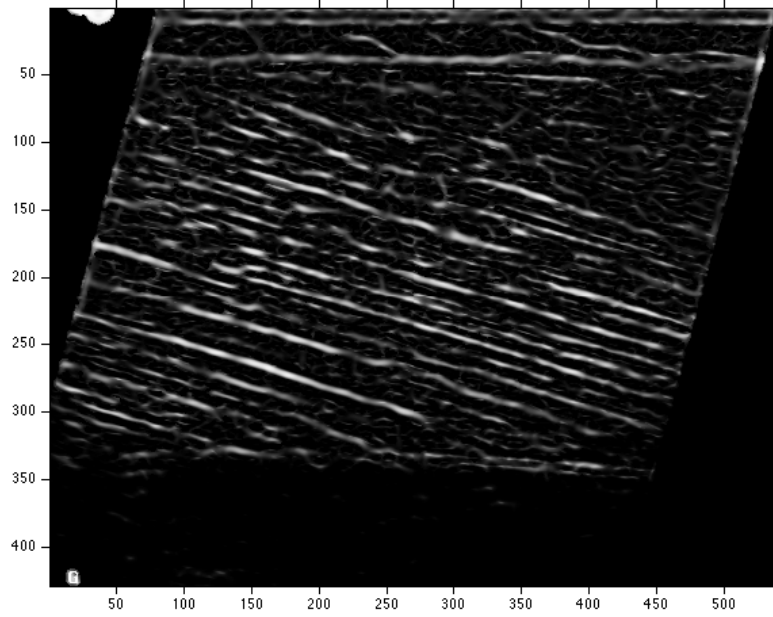


Figure 5.1: Result after filtering image shown in figure 3.1 with Frangi vessel filter.

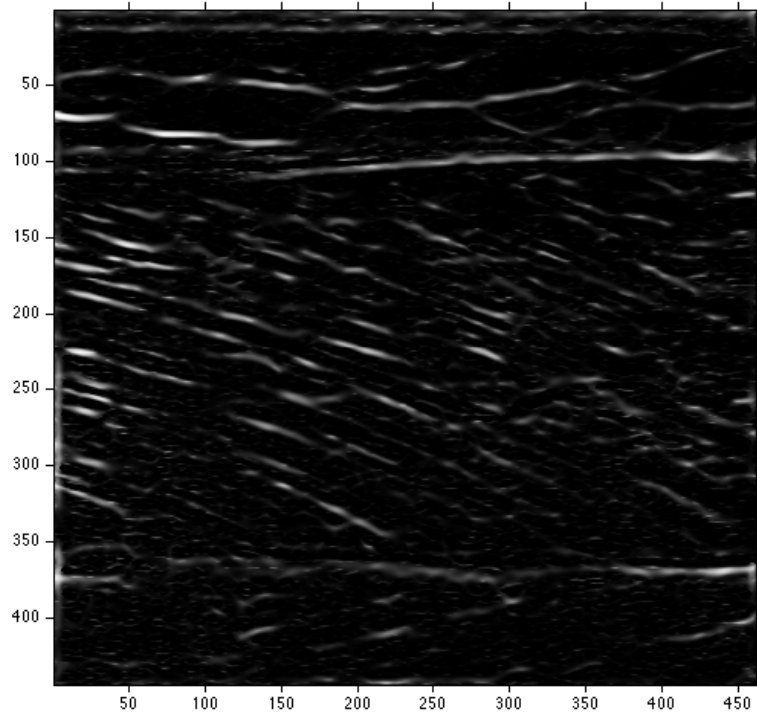


Figure 5.2: Result after filtering image shown in figure 3.5 with Frangi vessel filter.

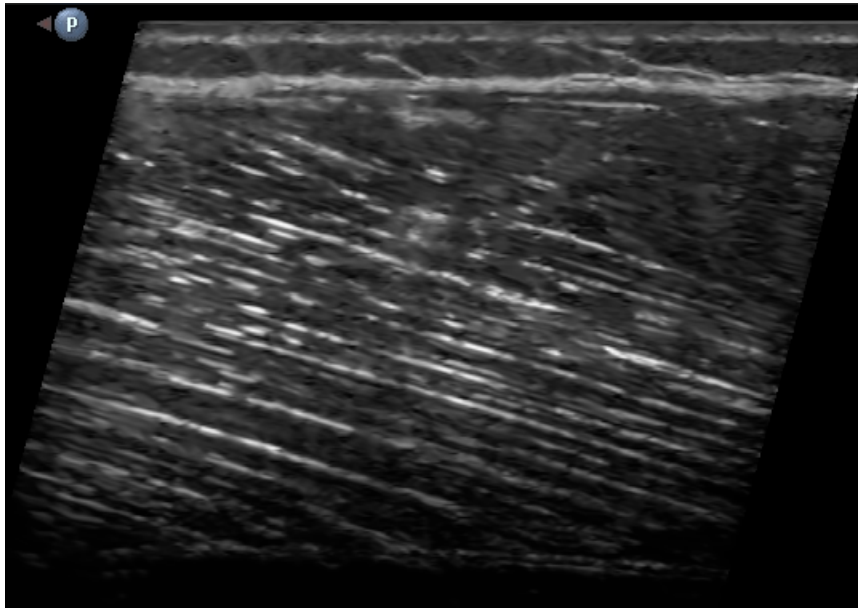


Figure 5.3: Result after filtering image shown in figure 3.1 using the non-local means method.

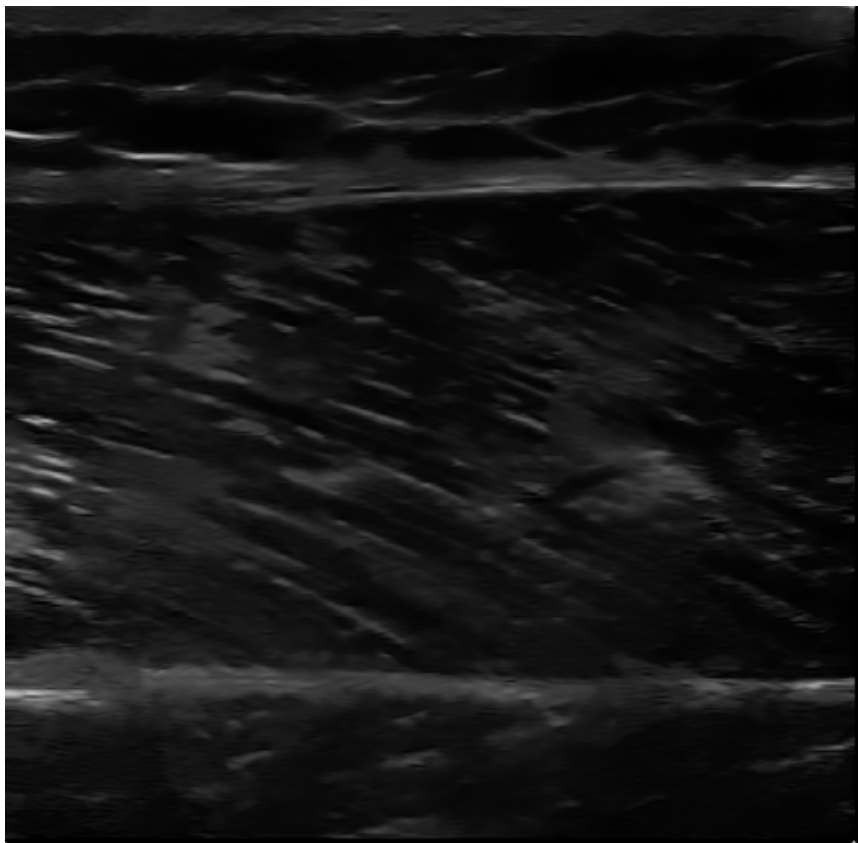


Figure 5.4: Result after filtering image shown in figure 3.5 using the non-local means method.

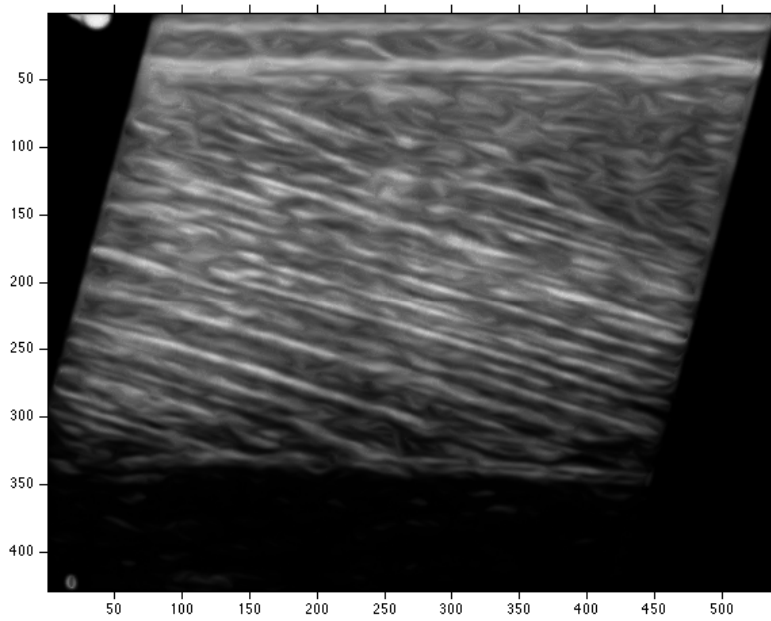


Figure 5.5: Result after filtering image shown in figure 3.1 with a Gaussian directional filter. The filter sizes ranged from 41x5 in areas where it was very certain of the direction to 5x5 in areas where the certainty was low.

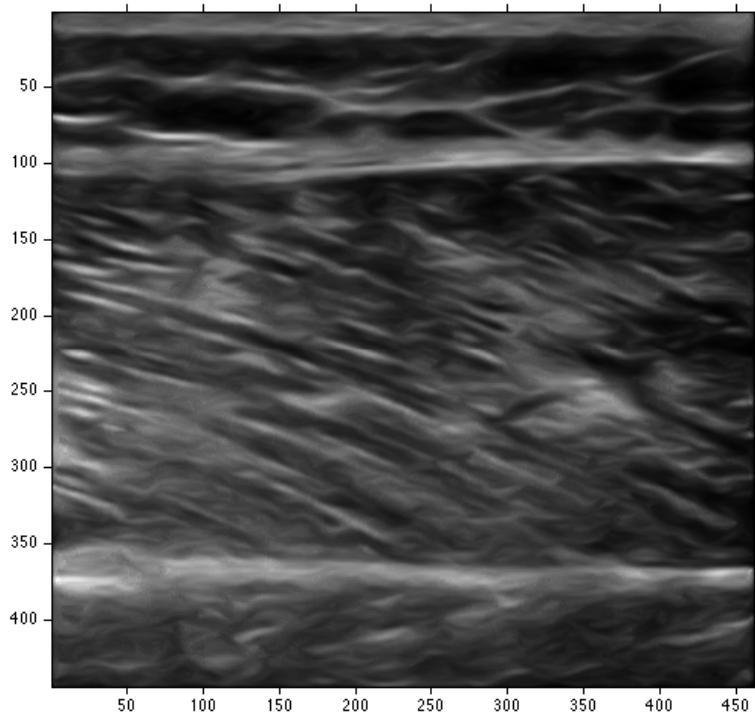


Figure 5.6: Result after filtering image shown in figure 3.5 with a Gaussian directional filter. The filter sizes ranged from 41x5 in areas where it was very certain of the direction to 5x5 in areas where the certainty was low.

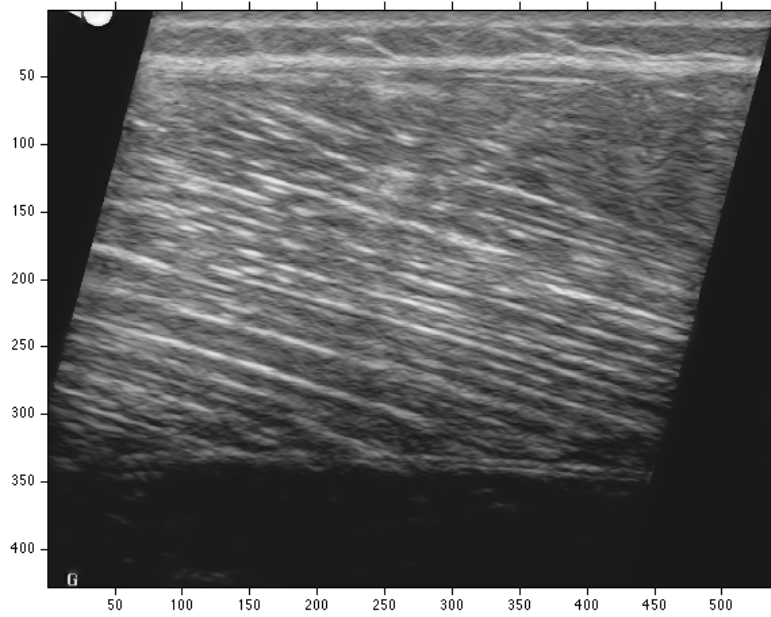


Figure 5.7: The summed lowpass output and highpass output from the Knutsson filtering method. Original image is shown in figure 3.1.

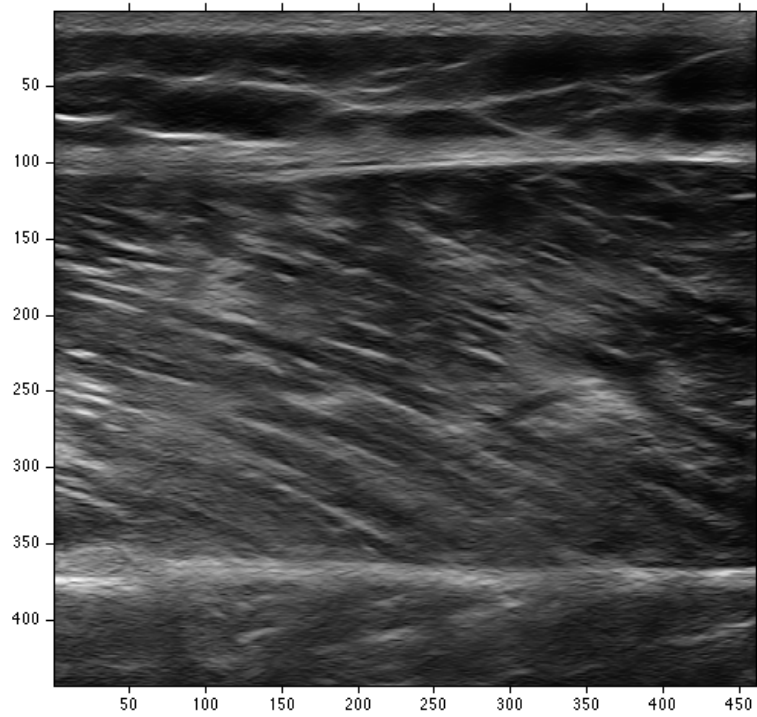


Figure 5.8: The summed lowpass output and highpass output from the Knutsson filtering method. Original image is shown in figure 3.5.

5.2 Results when testing on simulated images

In this section the results of the simulated images section are presented and discussed. As mentioned earlier, the results show how the algorithm responds to noise as the underlying structures are the same in all images. All the simulated images were processed by the algorithm using the same parameters. The results are compared to the "true" values acquired from the corresponding phantom image. The method used for obtaining the "true" values varies for the different parts of the algorithm, and is presented as we go. The dataset created from the phantom image in figure 3.9 is referred to as dataset 1, and the dataset created from the phantom image in figure 3.10 is referred to as dataset 2.

5.2.1 Results for aponeuroses detection

The detection of aponeuroses is a critical step in the algorithm because the estimated fascicle length and pennation angle are dependent on the estimation of upper and lower aponeuroses. If these estimations are inaccurate the final calculations of these features will also be inaccurate.

Because the phantom images for both simulated datasets had two equally prominent lines for the upper and lower aponeurosis, we chose one of these lines as the reference aponeuroses. For dataset 1 the lower line of the upper and lower aponeurosis was chosen as the reference aponeuroses. For dataset 2 the lower line of the upper aponeurosis and the upper line of the lower aponeurosis was chosen as reference aponeuroses. This means that there will be an error if the other line is detected as the aponeurosis. If many images have the same error it will at least show consistency through different amounts of noise. The "true" aponeuroses was defined by the center of mass in vertical direction and the orientation of the upper and lower reference aponeurosis in the phantom images. The center of mass is given in pixel coordinates and is defined as the center of the aponeurosis in vertical direction. The orientation is given as the angle in degrees between the x-axis and the major axis of the ellipse that has the same second-moments as the aponeurosis. The smallest rectangle containing each of the aponeuroses, the bounding box, was also found for later use in the error estimation calculation.

For each processed image the center of mass in vertical direction and orientation of the detected aponeuroses was calculated. The bounding box for the "true" aponeuroses was used to find the endpoints before calculating the center of mass in the vertical direction. The error was measured using the Euclidean distance, measured in pixels, between the "true" center of mass and the detected center of mass. The orientation error was measured in absolute difference in degrees. Table 5.1 and 5.2 shows the error estimates for the upper aponeurosis detection for each of the simulated datasets and table 5.3 and 5.4 shows the error estimates for the lower aponeurosis detection for each of the simulated datasets.

These tables show very good results for the upper aponeurosis detection where the exact same upper aponeurosis was detected in 10 out of 12 images in dataset 1 and 5 out of 7 images in dataset 2. The lower aponeurosis detection was not as consistent. The difference between upper aponeurosis detection and the lower aponeurosis detection is the image used as input to the Radon transform. As mentioned earlier we chose to use the Knutsson highpass image for the lower aponeurosis detection because it sometimes are a

lot less prominent than the upper aponeurosis. The results shows that for images where the lower aponeurosis is prominent, the better choice would be to use the Knutsson low-pass image as input to the Radon transform.

Another way to improve the lower aponeurosis detection could be to use a different method for locating the aponeurosis when the input image is the Knutsson highpass image. The Radon transform for the Knutsson lowpass and highpass output look different, so this detection would probably get more accurate if it was specialized for each of the input images. As it is now, the same method is used regardless of what input image is used.

Figure 5.11 (image number 6 in table 5.2 and 5.4) and figure 5.12 (image number 7 in table 5.2 and 5.4) shows two of the results for aponeuroses detection.

For the upper aponeurosis detection, image number 6 is the only image where a different aponeurosis is detected. The orientation error is small for this image, but the centroid error is big. This can be seen in figure 5.11 where the red line is almost parallel to the true reference aponeuroses, but in fact the detected aponeuroses is the other aponeuroses line. The detected aponeurosis in image number 7 is the same as for most of the other images in dataset 1. This can be seen in image 5.12. Even though the orientation error is relatively big, this is the best fit to the true reference aponeurosis we can get as long as we use a straight line to represent the aponeuroses. This can be seen in image 5.12.

For the lower aponeurosis detection, image number 6 has the largest error in dataset 1 along with image number 5, 9 and 10. As we can see in figure 5.11 this is because the other line in the aponeuroses is detected in these images. The detected fascicle in image 7 has the smallest error in dataset 1, along with 5 other images. This is shown in figure 5.12. This result shows the same problem as the upper aponeurosis detection; to achieve more accurate results we need to use something other than a straight line to represent the aponeuroses.

In retrospect, we have seen that having two equally prominent lines representing the aponeuroses made evaluation unnecessarily complicated. The evaluation of the results would have been easier if we used simulated images that only had one line representing the aponeuroses, but this was discovered too late to redo the simulation experiments.

Image number	Centroid error	Orientation error
1	0.8585	0.5584
2	0.8585	0.5584
3	0.1415	0.5584
4	0.8585	0.5584
5	0.8585	0.5584
6	7.5009	0.0675
7	0.8585	0.5584
8	0.8585	0.5584
9	0.8585	0.5584
10	0.8585	0.5584
11	0.8585	0.5584
12	0.8585	0.5584
Mean error	1.3523	0.5175
Standard deviation	1.9472	0.1417

Table 5.1: Error estimates for the upper aponeurosis detection for the simulated dataset created from the phantom image in figure 3.9. Centroid error is given in pixels and orientation error is given in degrees.

Image number	Centroid error	Orientation error
1	1.0784	0.6008
2	7.9013	0.0935
3	3.4413	1.6023
4	7.9013	0.0935
5	7.9013	0.0935
6	7.9013	0.0935
7	7.9013	0.0935
Mean error	6.2895	0.3815
Standard deviation	2.8360	0.5705

Table 5.2: Error estimates for the upper aponeurosis detection for the simulated dataset created from the phantom image in figure 3.10. Centroid error is given in pixels and orientation error is given in degrees.

Image number	Centroid error	Orientation error
1	0.6872	0.3595
2	0.3128	0.3595
3	0.6872	0.3595
4	0.8559	0.8507
5	8.0023	1.3549
6	8.0023	1.3549
7	0.6872	0.3595
8	0.6872	0.3595
9	8.0023	1.3549
10	8.0023	1.3549
11	0.6872	0.3595
12	0.6872	0.3595
Mean error	3.1084	0.7322
Standard deviation	3.6164	0.4803

Table 5.3: Error estimates for the lower aponeurosis detection for the simulated dataset created from the phantom image in figure 3.9. Centroid error is given in pixels and orientation error is given in degrees.

Image number	Centroid error	Orientation error
1	0.3812	0.4946
2	7.8286	0.5143
3	0.3812	0.4946
4	8.2359	0.0128
5	7.8286	0.5143
6	0.3812	0.4946
7	7.8286	0.5143
Mean error	4.6950	0.4342
Standard deviation	4.0378	0.1861

Table 5.4: Error estimates for the lower aponeurosis detection for the simulated dataset created from the phantom image in figure 3.10. Centroid error is given in pixels and orientation error is given in degrees.

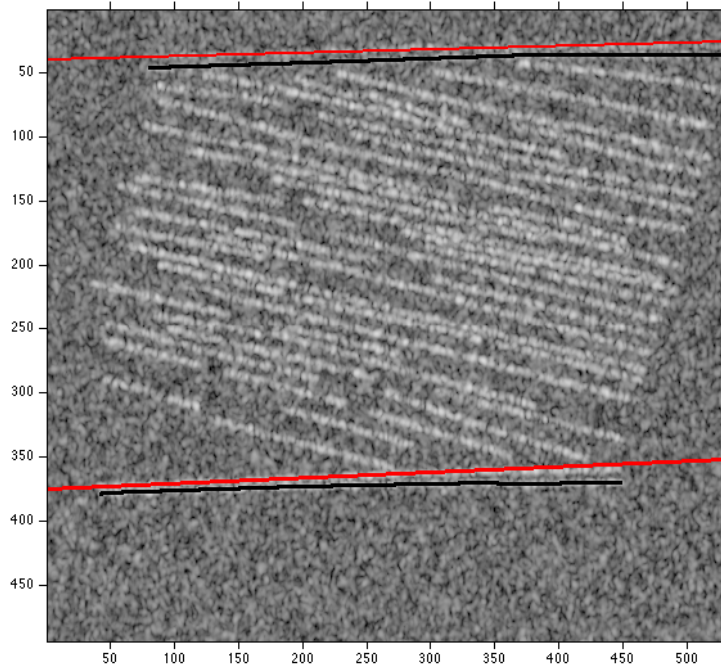


Figure 5.11: Image number 6 in table 5.1 and 5.3. Red line represents the detected aponeuroses. Black line shows the true aponeuroses acquired from the phantom image.

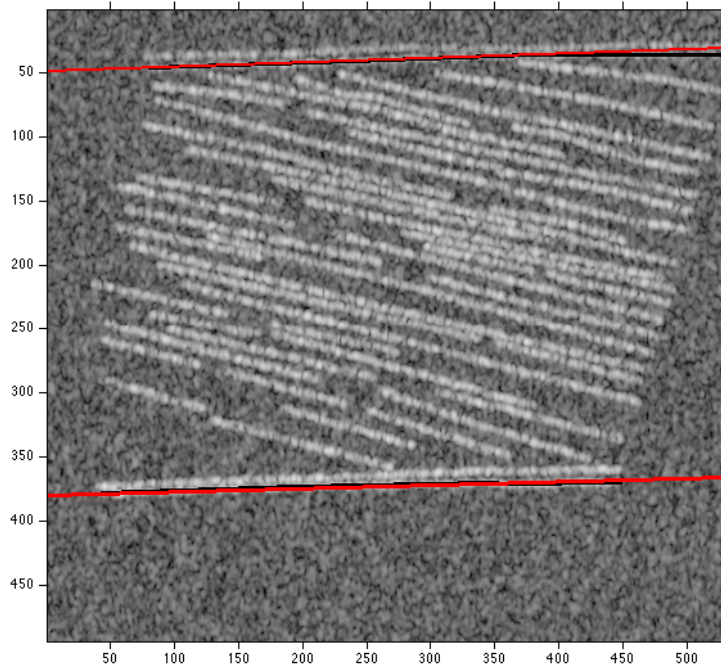


Figure 5.12: image number 7 in table 5.1 and 5.3. Red line represents the detected aponeuroses. Black line shows the true aponeuroses acquired from the phantom image.

5.2.2 Results for fascicle detection

The error in fascicle detection was measured by using the dominant angle in each grid cell used to reconstruct the fascicle. The grid can be seen in figure 4.13. As mentioned in section 4.5, the outermost columns of the grid are excluded when constructing the fascicles because these often contain unreliable detections. These columns are therefore not included in these error measurements. To find the "true" dominant angle the phantom image was processed using the same algorithm as the other images. So the "true" dominant angles in this case was the angles detected on an image without noise.

The error was calculated as the average difference between the dominant angle for each relevant grid cell and the "true" dominant angle in the same grid cells. Table 5.5 and 5.6 shows the error estimates for the two simulated datasets.

Two example images illustrating different results are shown in figure 5.13 (image number 2 in table 5.6) and 5.14 (image number 3 in table 5.6). We chose to illustrate the fascicle detection result with this image instead of the image showing the mean angle in each grid cell. The reason for this is that the fascicle detections shown in figure 5.13 and 5.14 are the data we use for calculating the mean angle in each grid cell. The detected fascicles are shown in red while the "true" fascicles are shown in green. Image number 2 has the lowest error in dataset 2 with an error of 0.2370 degrees, while image number 3 has the biggest error in dataset 2 with an error of 0.8043 degrees.

The most obvious difference between these two results is the amount of fascicles detected in each of the images. Image number 2 has a lot more detected fascicles than image number 3. It is also clear that the result corresponds to the image quality. In image number 2 the contrast between the fascicles and the background is high, but in image number 3 it is very low. Image number 3 is the image with the least prominent fascicles in dataset 2.

These images also show why we discard the outermost columns of the grid. This problem usually occurs when the vertical borders of the ultrasound image are skewed. The fascicle detections on the side of the images are imprecise. This is especially visible on the right side of the images in figure 5.13 and 5.14. This happens when the windows used as input to the Radon transform approaches the edge of the image. The window will then contain only segments of the fascicles which lead to inaccurate detections. This is usually not a problem when the vertical borders of the ultrasound image are straight.

Image number	Grid mean error
1	0.2249
2	0.6005
3	0.3400
4	0.3538
5	0.1900
6	0.1642
7	0.2093
8	0.2249
9	0.1661
10	0.1589
11	0.2616
12	0.5806
Mean error	0.2896
Standard deviation	0.1542

Table 5.5: Table showing the mean difference in degrees between "true" dominant angles and detected dominant angles for the fascicles in the simulated dataset created from the phantom image in figure 3.9.

Image number	Grid mean error
1	0.2694
2	0.2370
3	0.8043
4	0.4959
5	0.3753
6	0.2831
7	0.3280
Mean error	0.3990
Standard deviation	0.1982

Table 5.6: Table showing the mean difference between "true" dominant angles and detected dominant angles for the fascicles in the simulated dataset created from the phantom image in figure 3.10.

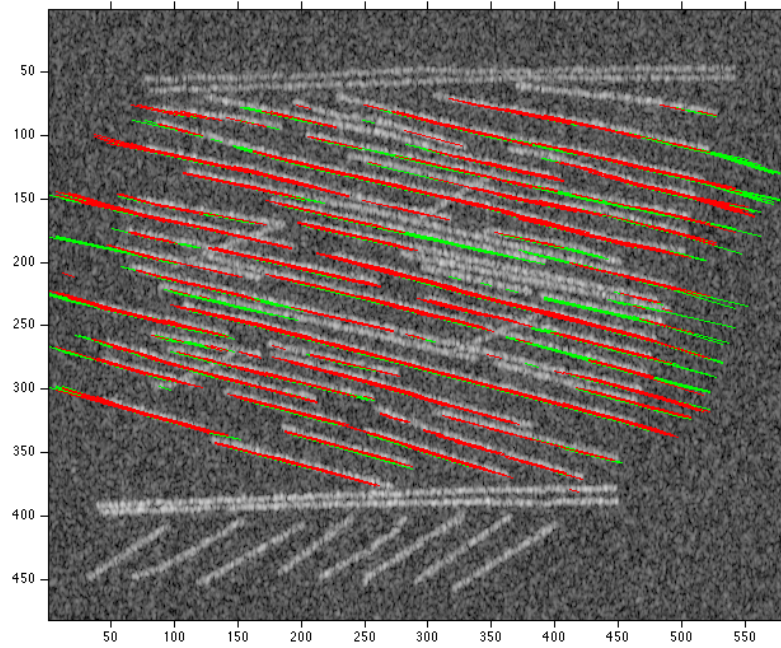


Figure 5.13: Image number 2 in table 5.5. This image had the lowest error for this dataset with an error of 0.2370 degrees. Red lines representing detected fascicles. Green lines showing "true" fascicles detected in the phantom image in figure 3.10.

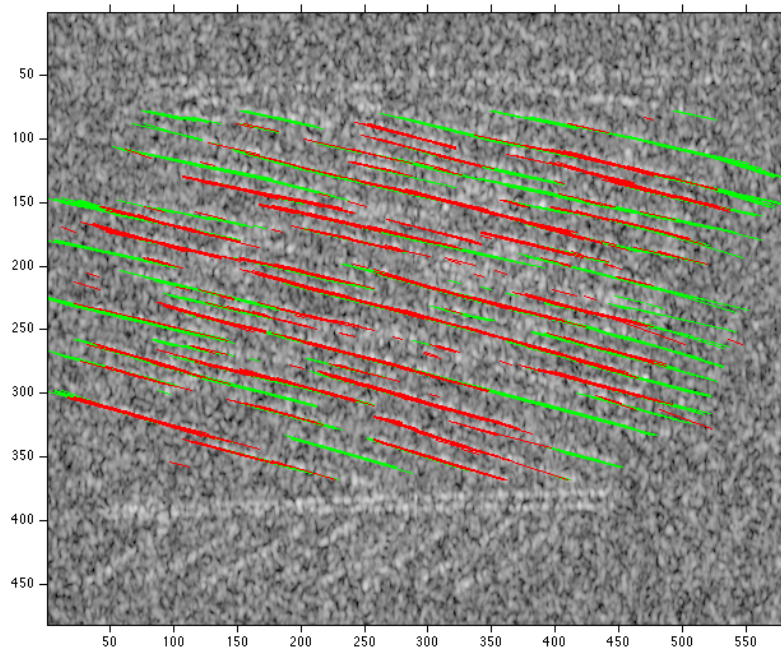


Figure 5.14: image number 3 in table 5.6. This image had the biggest error of all images with an error of 0.8043 degrees. Red lines representing detected fascicles. Green lines showing "true" fascicles detected in the phantom image in figure 3.10.

5.2.3 Results for pennation angle and fascicle length

Table 5.7 and 5.8 shows the absolute value of the difference in calculated fascicle length and pennation angle when compared to the phantom image. The fascicle length error is measured in pixels and the pennation angle error is measured in degrees.

The fascicle length is dependent on the accuracy of both upper and lower aponeuroses detection. As the tables show the fascicle length error quickly accumulates. The fascicle length for dataset 1 and dataset 2 was measured to be 1162 and 1102 pixels respectively. This gives a mean error of 2.54 % and 6.28 % for the two datasets, which is a relatively big error. One of the factors contributing to this error is the fact that the fascicles and aponeuroses are extended outside the field of view. Even a small error in detection of aponeuroses orientation or fascicle orientation can therefore cause a big difference in the intersection point between aponeuroses and fascicle. This can be seen in figure 5.15 and 5.16. Figure 5.15 is image number 1 in dataset 1, and is the image with the smallest fascicle length error. Figure 5.16 is image 2 in dataset 2, which is the image with the biggest fascicle error. These images clearly shows how sensitive the fascicle length estimation is to the accuracy of the aponeuroses detection.

The pennation angle is dependent on lower aponeurosis detection and the calculation of one reference fascicle angle. Two examples corresponding to the smallest and biggest error in dataset 1 can be seen in figure 5.17 (image number 7 in table 5.7) and 5.18 (image number 10 in table 5.7) respectively. The angle of the reference fascicle is detected as 75.0909 degrees in the phantom image, and 75.1458 and 75.2875 degrees in image 7 and 10. This is the angle plotted inside the yellow rectangle in figure 5.17 and 5.18. This means that the error in pennation angle of 0.1966 degrees in image 7 originates only from the calculation of the reference fascicle angle. In image 10 the error in reference fascicle angle is 0.0549 degrees, while the majority of the error originates from the fact that the other line in the lower aponeurosis was detected. This shows that even though the number of detected fascicles located inside the yellow rectangle is less in image 10 than in image 7, it is still able to estimate a fairly good reference fascicle.

Image number	Fascicle length error	Pennation angle error
1	2.3544	0.1998
2	46.9234	0.4239
3	23.9440	0.2750
4	20.7744	0.7509
5	64.9408	0.8260
6	19.1224	1.0266
7	5.4470	0.1966
8	2.3544	0.1998
9	61.8503	1.0830
10	63.9355	1.1417
11	6.4733	0.2875
12	32.5572	0.5018
Mean error	29.2231	0.5760
Standard deviation	24.4905	0.3696

Table 5.7: Table showing the difference in detected fascicle length and pennation angle for each image compared to the detected fascicle length and pennation angle in the phantom image. Dataset corresponding to the phantom image in figure 3.9.

Image number	Fascicle length error	Pennation angle error
1	20.6145	0.1863
2	95.1511	0.6667
3	86.7233	0.5152
4	73.5536	0.1667
5	85.7372	0.6667
6	31.5905	0.1690
7	90.9520	0.6667
Mean error	69.1889	0.4339
Standard deviation	30.3346	0.2490

Table 5.8: Table showing the difference in detected fascicle length and pennation angle for each image compared to the detected fascicle length and pennation angle in the phantom image. Dataset corresponding to the phantom image in figure 3.10.

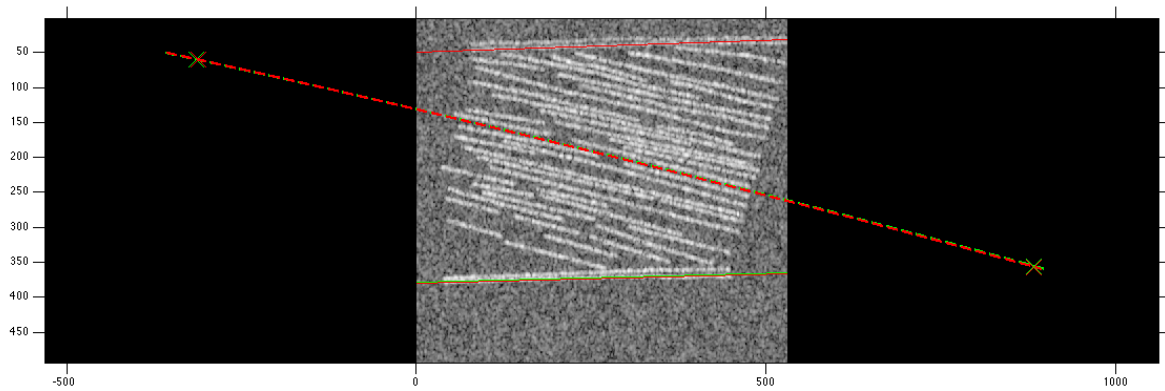


Figure 5.15: Image number 1 in table 5.7. This image had the smallest difference in fascicle length compared to the "true" length. Red line representing the aponeuroses and the reconstructed fascicle for image number 1. Green lines showing "true" aponeuroses and reconstructed fascicle for the phantom image.

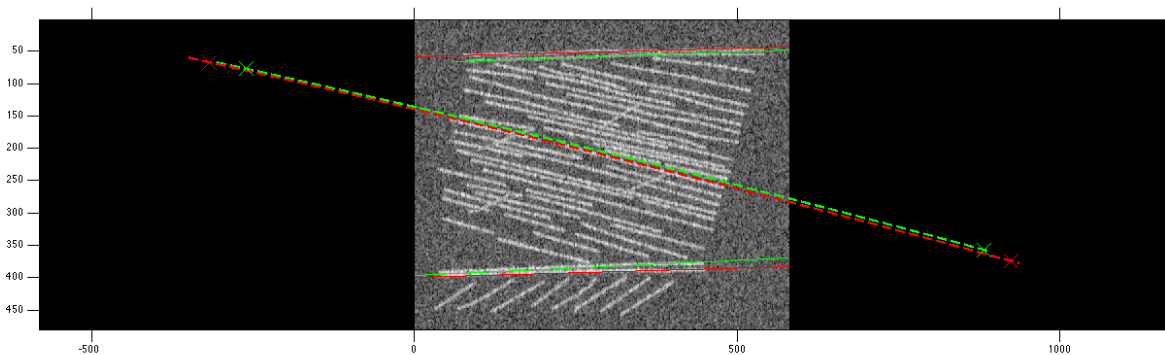


Figure 5.16: Image number 2 in table 5.8. This image had the biggest difference in fascicle length compared to the "true" length. Red lines representing aponeuroses and reconstructed fascicle for this image. Green lines showing "true" aponeuroses and reconstructed fascicle for the phantom image.

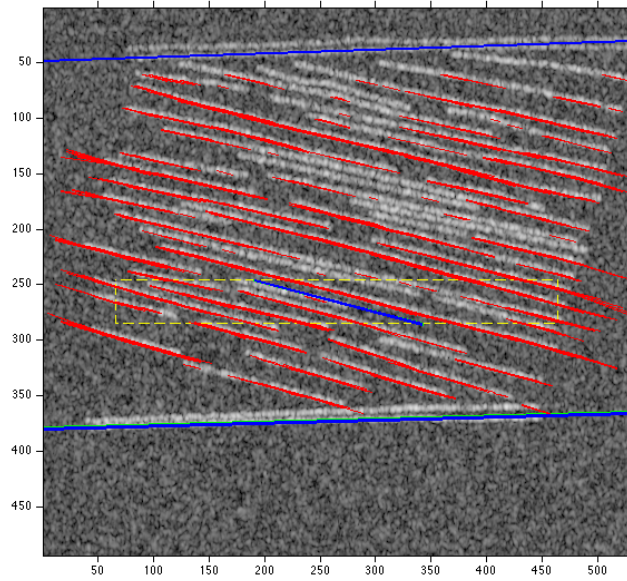


Figure 5.17: Image number 7 in table 5.8. This had the smallest error in pennation angle in dataset 1 with an error of 0.1966. The detected aponeuroses and pennation angle in the phantom image is marked in green. The detected aponeuroses and pennation angle in image number 7 is marked in blue. The green lines are not visible in the upper aponeurosis and the pennation angle because they are plotted underneath the blue lines.

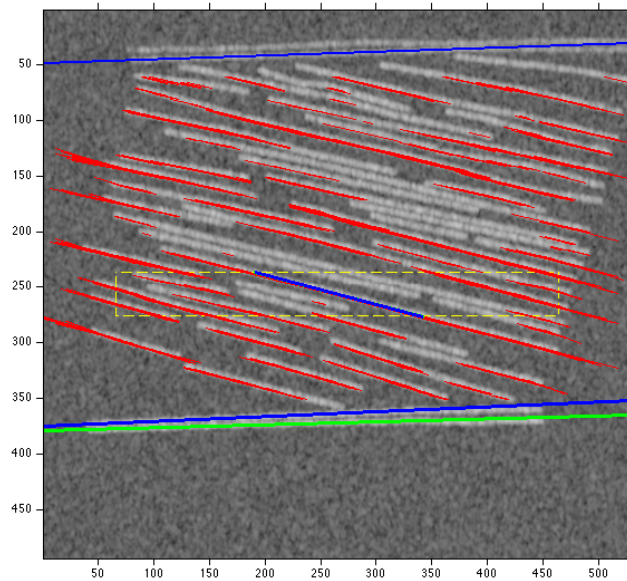


Figure 5.18: Image number 10 in table 5.8. This had the biggest error in pennation angle in dataset 1 with an error of 1.1417. The detected aponeuroses and pennation angle in the phantom image is marked in green. The detected aponeuroses and pennation angle in image number 10 is marked in blue. The green lines are not visible in the upper aponeurosis and the pennation angle because they are plotted underneath the blue lines.

5.3 Results for ultrasound images without manual analysis

Because we do not have any manual interpretation to compare the results of this dataset with, the evaluation of this dataset will be based on visual inspection of the results. We have chosen one representative image from each part of this dataset to illustrate the discussions in this section. The results for the first image, obtained from the part of the dataset containing one scan sequence, is shown in figures 5.20, 5.21, 5.22 and 5.23. This will be referred to as image 1. The results for the second image, obtained from the part of the dataset containing lower quality ultrasound images, is shown in figure 5.24, 5.25, 5.26 and 5.27. This will be referred to as image 2.

5.3.1 Results for aponeuroses detection

Just as in the simulated images, a challenge occurs when the aponeuroses are curved. In these cases the full aponeuroses can not be represented in a good way using a straight line. This problem occurs in both image 1 and image 2. In figure 5.20 the upper neuroses detection looks correct, but the lower aponeurosis representation is not very good. In the right side of the image the line fits with the lower aponeuroses, but because it curves down towards the left side of the image, it does not fit well in this area. In figure 5.24 this problem occurs in both the upper and lower aponeurosis detection as both of the aponeuroses are curved in image 2.

Figure 5.19 shows an image where the lower aponeurosis detection fails. This happens because the approximate aponeurosis location detection is uncertain of where the aponeurosis is located. The cropped image used to detect the exact aponeurosis comes out too big because of this, which causes the wrong line to be detected. Another thing that can cause this to happen is if there are very prominent fascicle close to a faint lower aponeurosis.

5.3.2 Results for fascicle detection

The fascicle detection for each of the images can be seen in figure 5.20 and 5.24. The fascicle detection in image 2 looks accurate. The fascicle detection in image 1 shows some of the challenges in the algorithm. Because the borders of the image is skewed, the detections on the sides of the image are inaccurate. The reason why this happens is explained in more detail in section 5.2.2.

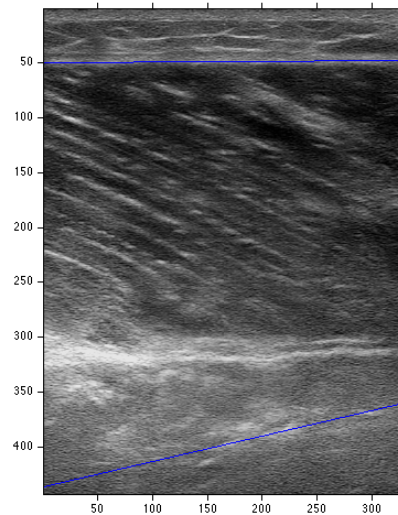


Figure 5.19: Example where the lower aponeurosis detection failed.

We can also see that very few fascicles are detected in the lower part of image 1 (figure 5.20). This is caused by the part of the algorithm that tries to avoid the S-shaped fascicles. The calculation done for determining a convex or concave shape of the fascicles concludes that the fascicles in this image curve upward, which is not a correct observation. In this image the fascicles actually curve in two places; they curve up towards the upper aponeurosis and down towards the lower aponeurosis. The way we tried to solve this in the algorithm leads to no fascicles being removed in the upper part of the image, while too many fascicles often are removed in the lower part of the image. This part of the algorithm therefore needs more work. One suggestion is that instead of deciding between convex or concave we should just make sure the fascicles curve in the same direction in different depths of the image.

The grid showing the mean angle in different areas of the images is shown in figure 5.21 and 5.25 for image 1 and 2 respectively. The detected fascicles discussed in the previous paragraph are used for calculating the mean angle in each grid cell. Even though there were some problems with the fascicle detection in image 1, the mean angle looks representative in most of the grid cells. The vertical line in the bottom right grid cell means that there was not enough data for this area. The mean angles calculated for image 2 also looks representative. Even though the results for the grid cells look precise it might be worth looking into using the median instead of the mean angle for this calculation.

5.3.3 Results for pennation angle and fascicle length

The pennation angle is calculated using the lower aponeurosis detection and the fascicle detections. The results for pennation angle in image 1 and 2 are shown in figure 5.22 and 5.26. Because we only want one estimated pennation angle we need to calculate a reference fascicle. The reference fascicles are plotted as green lines in figure 5.22 and 5.26. Visual inspection of the images shows that the estimated reference fascicle looks representative in both images. The biggest challenge in calculating the pennation angle seems to be the lower aponeuroses detection.

The fascicle length is calculated by reconstructing a fascicle and measuring the length of this. To carry out the reconstruction, parts of the grid data (figure 5.21 and 5.25) is used. As previously mentioned, the outermost columns of the grid are discarded for this calculation because of challenges with fascicle detection near the borders in some of the images. The results of the reconstruction for image 1 and 2 can be seen in figure 5.23 and 5.27. For image 1 we can see that the reconstruction has some problems near the top and bottom of the fascicle. Near the top the reconstructed fascicle should have curved more towards the upper aponeurosis. Near the bottom it should have curved more towards the lower aponeurosis. This shape is not possible to represent with a second degree polynomial. For image 2 the fascicle should have curved more towards the lower aponeurosis, and less towards the upper aponeurosis. These results show that it would be better to use something other than a second degree polynomial to represent the fascicle.

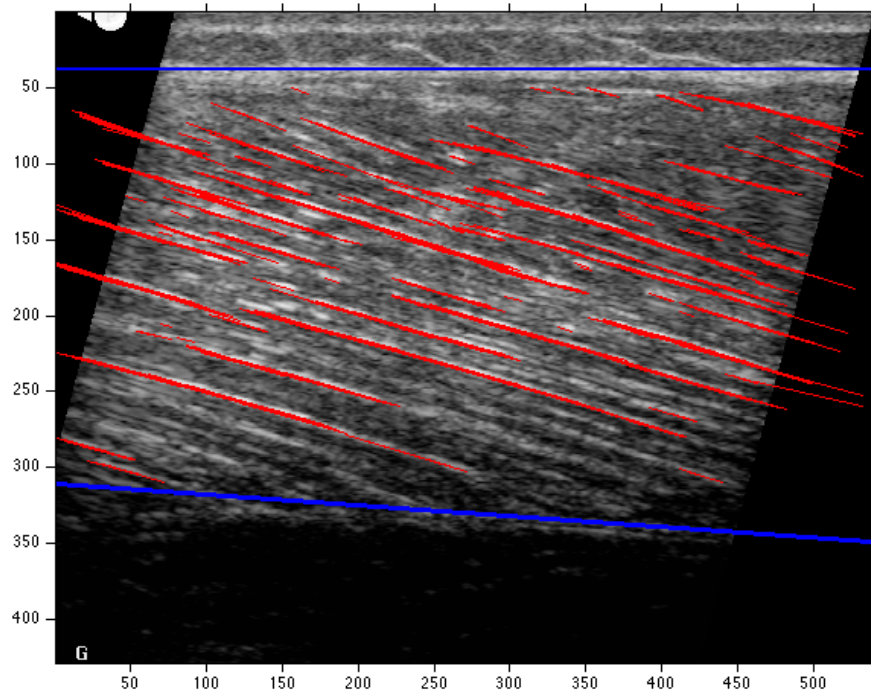


Figure 5.20: The results for fascicle detection for image 1.

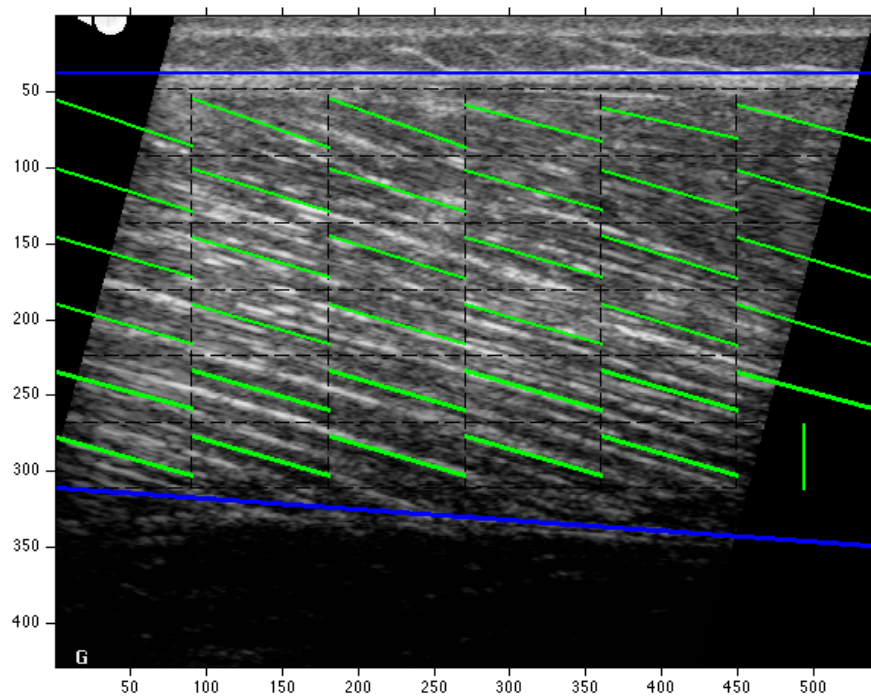


Figure 5.21: The grid containing mean values in different areas of image 1. This is calculated from the detected fascicles shown in figure 5.20.

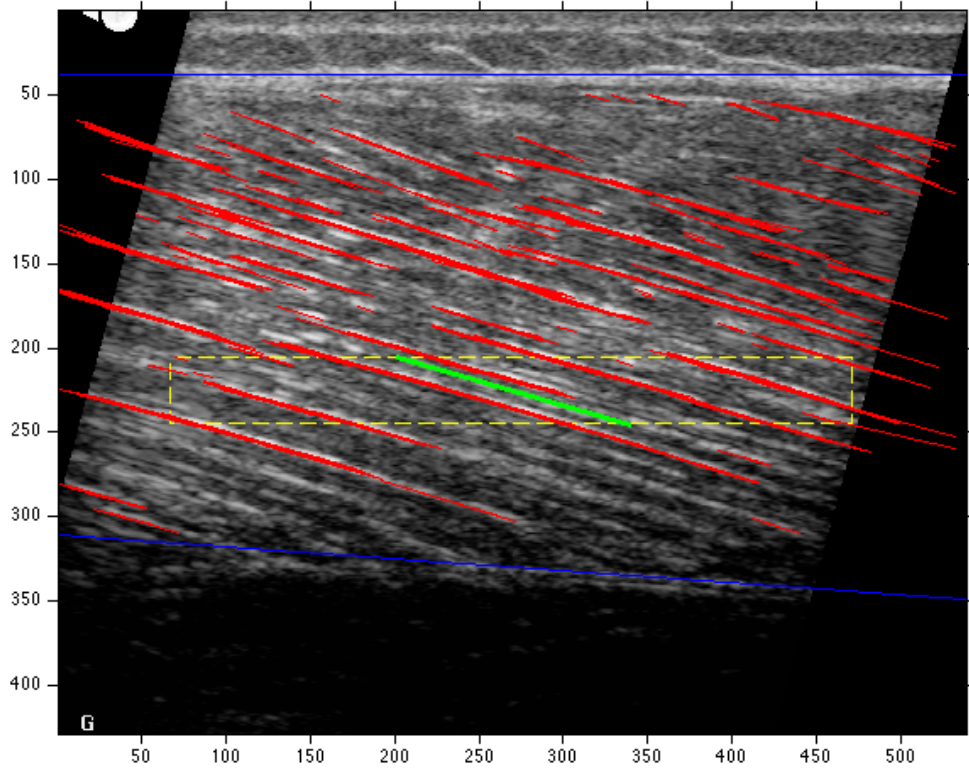


Figure 5.22: The estimated reference fascicle for image 1 plotted in green. This is used for calculating pennation angle.

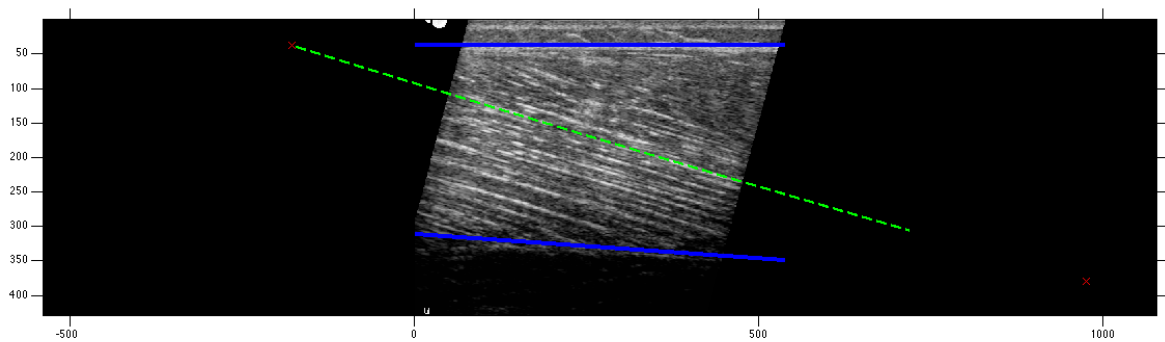


Figure 5.23: Reconstructed fascicle for image 1. This is used for measuring fascicle length, and is constructed based on the data shown in figure 5.21.

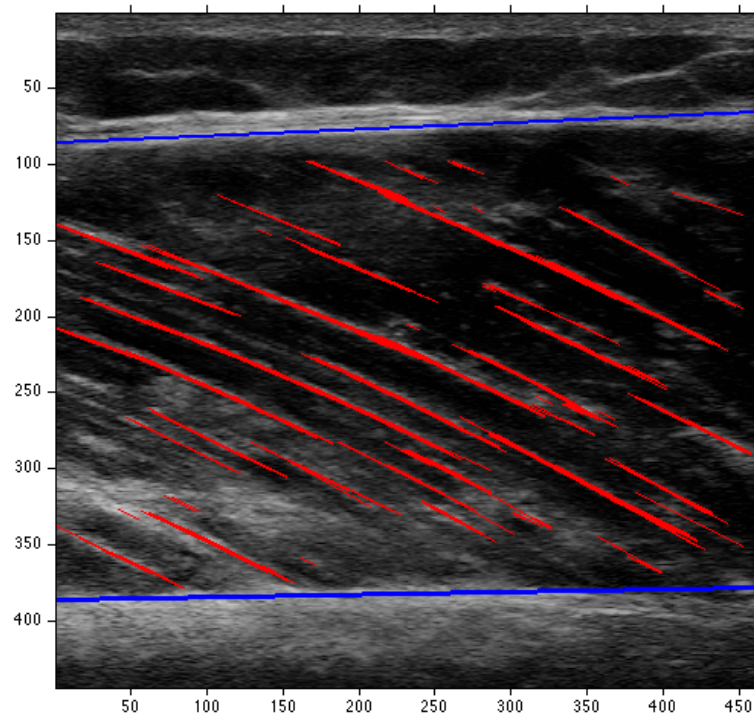


Figure 5.24: The results for fascicle detection for image 2.

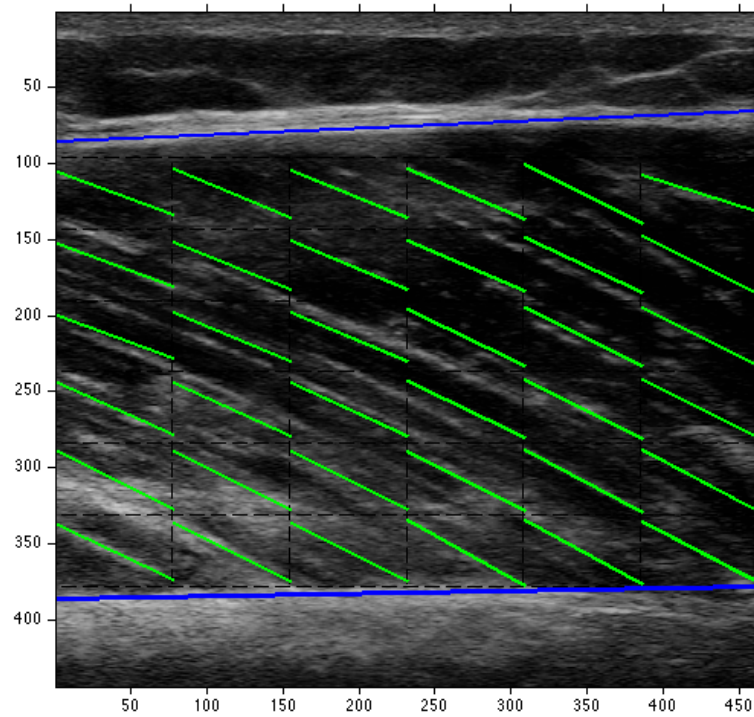


Figure 5.25: The grid containing mean values in different areas of image 2. This is calculated from the detected fascicles shown in figure 5.24.

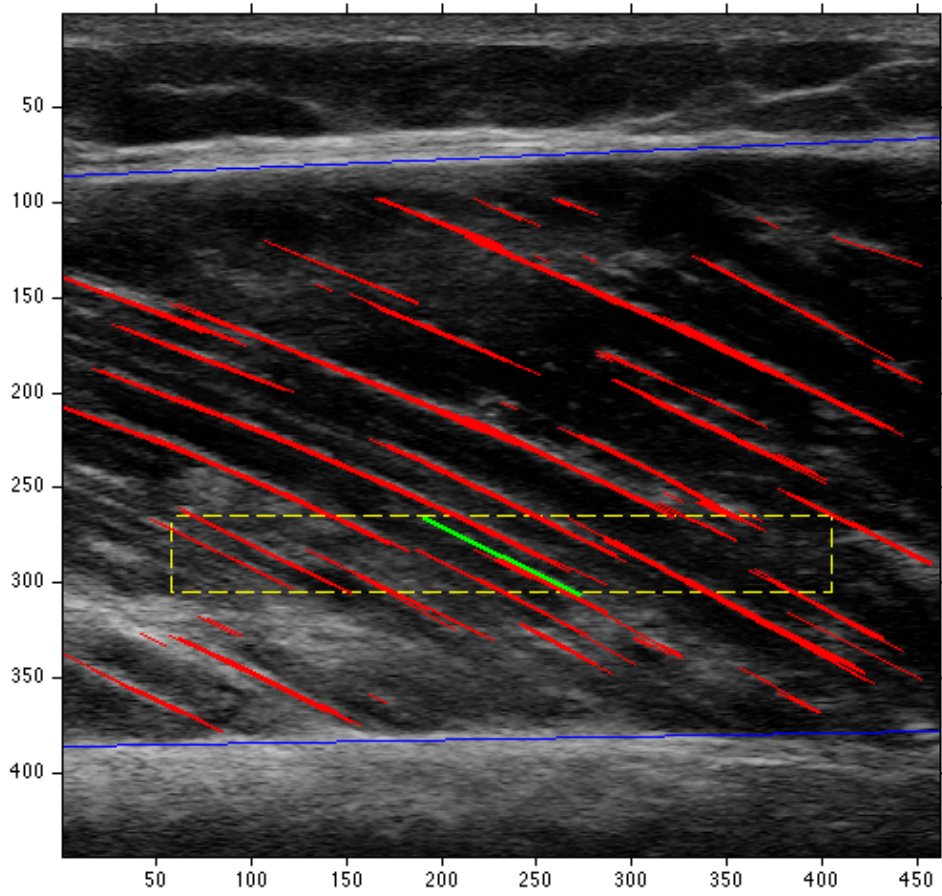


Figure 5.26: The estimated reference fascicle angle for image 2 plotted in green. This is used for calculating pennation angle.

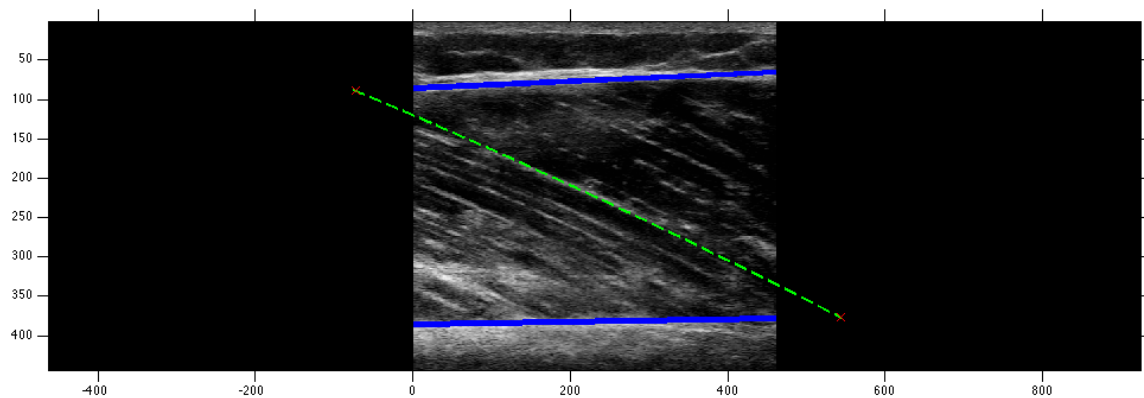


Figure 5.27: Reconstructed fascicle for image 2. This is used for measuring fascicle length, and is constructed based on the data shown in figure 5.25.

5.4 Results for images with manual analysis

Table 5.9 shows the manually detected pennation angle compared to the automatic detected pennation angle. In this section the results shown in these tables are discussed and example images of the results are shown. The manual analysis is only conducted by one person and can not be seen as the only correct pennation angle. As previously mentioned, the manual analysis is subject to human error and bias. It was conducted by fitting a straight line to the lower aponeurosis. Then, the angle between the fitted line and a representative fascicle was measured. The manual analysis only contains the pennation angle, so fascicle length will not be evaluated in this section. Since the manual analysis only provided the pennation angle for each image there is no way to know whether it is the lower aponeuroses detection, the fascicle detection or both that is causing the difference in measured pennation angle.

Some of the images with the biggest errors in table 5.9 (e.g. subject 3 condition short and subject 5 condition short) have a skewed lower aponeurosis. This might cause the imprecise measurements. The top and bottom borders of the grid are horizontal, and is defined by the lowest point of the upper aponeurosis and the highest point of the lower aponeurosis. The error could be reduced by letting the grid adapt to the aponeuroses. By doing this we would also get fascicle detections covering the entire fascicle plane even though the aponeuroses are skewed. The area of the image used to calculate the pennation angle should also be oriented at an angle relative to the fascicle plane. By doing this we would ensure that the angles contributing to the final angle is located at the same depth of the muscle. This would probably reduce the error in these cases since the angle of the fascicle often change in different depths. Figure 5.28 shows an example of this problem.

A different case that can cause a big difference between manual and automatic detection of pennation angle is when the lower aponeurosis is curved. An example showing this situation can be seen in figure 5.30. The difference between the manual analysis and the automatic detection was 8.08 degrees. Because we define the aponeuroses as a straight line the angle estimated for the lower aponeuroses in this image will not fit the entire aponeuroses. The manual analysis probably defined a line leaning to the left while the automatic method detected a line leaning to the right. An aponeuroses detection method that allows curved aponeuroses would perhaps be a better choice than using a straight line. This would however also complicate the pennation angle calculation.

Figure 5.29 shows the image which gave the least difference in pennation angle detection compared to the manual analysis. The difference was 0.09 degrees.

The images in table 5.9 where no automatic measurement and error is recorded is images where the lower aponeurosis detection failed. This happened in 7 out of 40 images, which is a relatively large part of the dataset. The reason why this happens is usually that the first approximate detection is unsure of where the aponeurosis is located. This uncertainty makes the window used for detecting the exact location too big which makes the algorithm detect a prominent fascicle instead of the less prominent aponeurosis.

Subject	Condition	Manual	Automatic	Error
1	short	12.47	12.56	0.09
	long	16.86	16.68	0.18
2	short	15.06	16.57	1.51
	long	14.95	17.81	2.86
3	short	16.00	21.24	5.24
	long	15.43	17.68	2.25
4	short	18.27	18.50	0.23
	long	17.86	18.16	0.30
5	short	15.64	21.24	5.60
	long	14.11	14.93	0.82
6	short	12.90	11.90	1.00
	long	14.28	12.50	1.78
7	short	13.21	14.00	0.79
	long	19.69	20.32	0.63
8	short	15.05	-	-
	long	13.35	4.28	0.93
9	short	14.40	11.88	2.52
	long	15.60	-	-
10	short	17.81	-	-
	long	16.00	-	-
11	short	20.16	18.20	1.96
	long	16.50	15.49	1.01
12	short	16.78	-	-
	long	18.01	17.17	0.84
13	short	15.56	18.79	3.23
	long	19.90	18.75	1.15
14	short	19.58	19.47	0.11
	long	15.41	18.30	2.89
15	short	13.46	-	-
	long	14.78	15.00	0.22
16	short	16.56	16.00	0.56
	long	16.10	12.43	3.57
17	short	16.20	17.70	1.50
	long	18.40	19.43	1.03
18	short	14.14	17.95	3.81
	long	19.17	11.09	8.08
19	short	15.04	10.98	4.06
	long	18.86	-	-
20	short	17.68	22.50	4.82
	long	21.59	18.90	2.69

Table 5.9: The manually measured pennation angle, the automatic detected pennation angle and the absolute difference of these. These are the 20 sets of ultrasound images where each set contains one image of the conditions short and one image of the condition long.

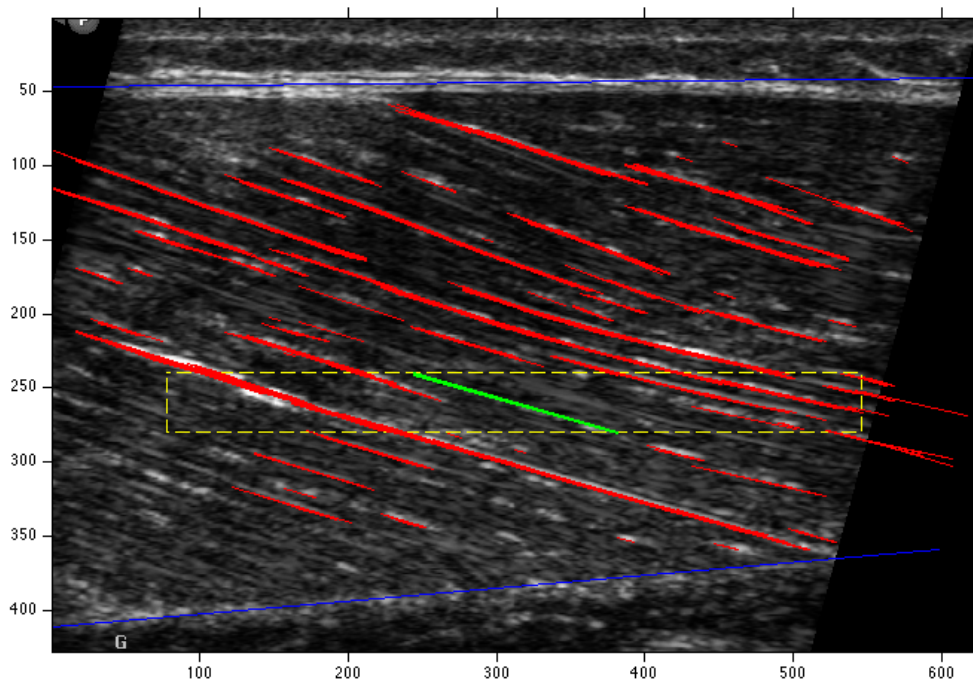


Figure 5.28: One of the images where the aponeurosis is skewed. This image (subject 3, condition short in table 5.9) had a difference of 5.24 degrees between the manual analysis and the automatic detection. Green line showing the angle of the median of all the lines located inside the rectangle. This angle was, together with the lower aponeurosis angle, used to calculate the pennation angle.

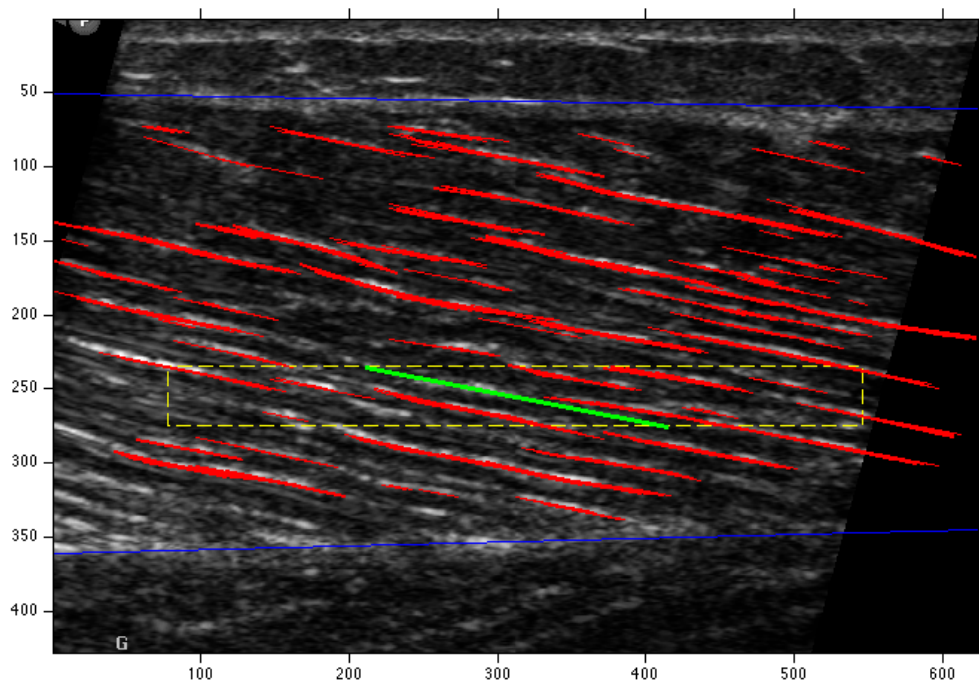


Figure 5.29: Image which had the strongest agreement between manual and automatic detection of pennation angle (subject 1, condition short in table 5.9). The difference was 0.09 degrees. Green line showing the angle of the median of all the lines located inside the rectangle. This angle was, together with the lower aponeurosis angle, used to calculate the pennation angle.

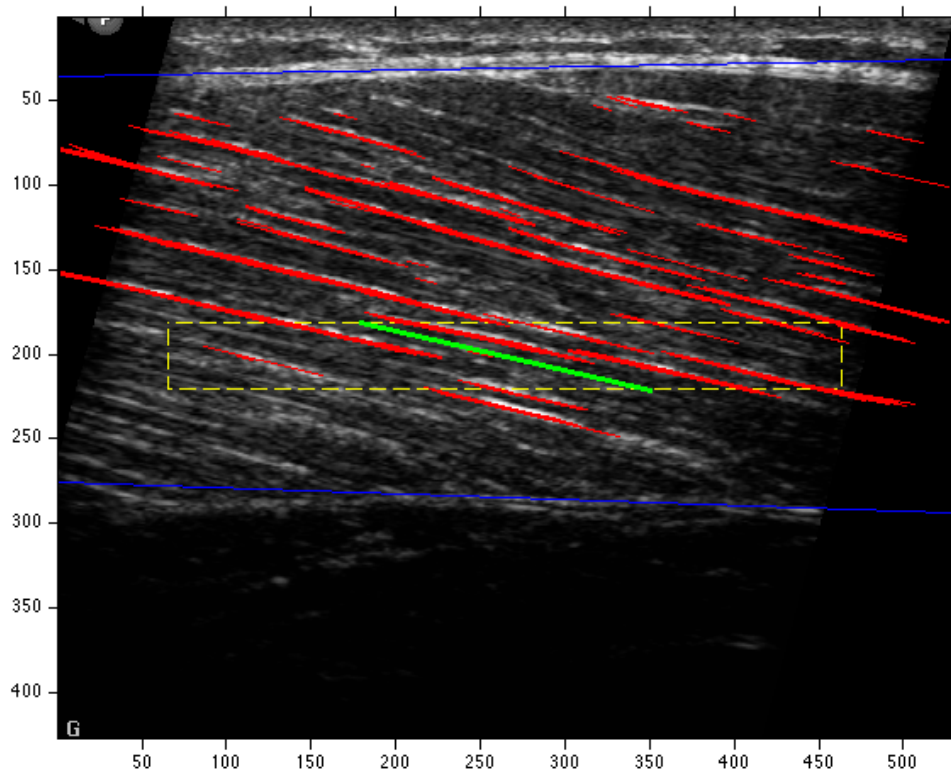


Figure 5.30: Image which had the biggest disagreement between manual and automatic detection of pennation angle (subject 18, condition long in table 5.9). The difference was 8.08 degrees. Green line showing the angle of the median of all the lines located inside the rectangle outlined in yellow. This angle was, together with the lower aponeurosis angle, used to calculate the pennation angle.

5.5 Assumption that the results can be transferred to the area outside field of view

When reconstructing a fascicle we need to assume that the information we have for different depths of the image can be transferred to the parts of the muscle outside the field of view. To test this we used an image of several different scans stitched together to form a bigger image of the muscle. We selected a part of this image and processed this part using the algorithm. The results from this subimage analysis was then plotted on the original full image. The result can be seen in figure 5.31. This image shows that the information can be transferred to the parts outside the image to some degree, but we can also see the effect of the weak parts of the algorithm. Because the upper and lower aponeurosis detection assumes a straight line, the intersection point between the fascicle and the upper neurosis, used for calculating the fascicle length, will not be accurate in this case.

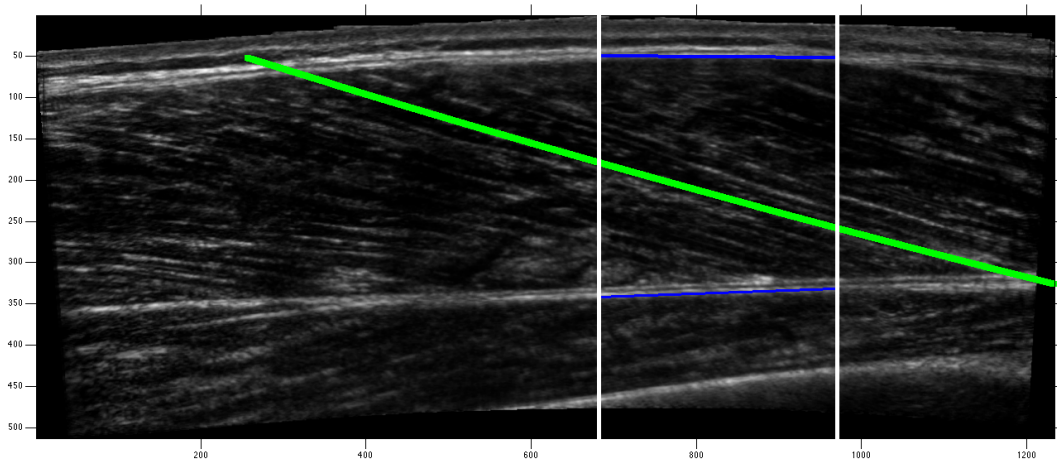


Figure 5.31: Algorithm tested on a subimage of this, shown with the length of the upper and lower neurosis, and plotted on the entire image. The part of the image between the white lines was the part that was processed by the algorithm.

Chapter 6

Conclusion

In this thesis we aimed to develop a fully automated algorithm for estimating the muscle features pennation angle and fascicle length in ultrasound images of the vastus lateralis. The goal was to end up with one representative pennation angle, and to estimate the fascicle length from a single frame. The fascicle angle should be estimated even though the entire fascicle was not within the field of view, and the fascicle curvature should be taken into account. The resulting algorithm proposes an automatic method to estimate these features. This algorithm includes detection of the region of interest, filtering of the images to reduce speckle noise and enhance the structures, aponeuroses detection, fascicle detection and calculation of the muscle features. The pennation angle is estimated based on the angle of the lower aponeurosis and a representative reference fascicle, resulting in one pennation angle. The fascicle length are estimated based on a reconstructed fascicle represented by a 2. degree polynomial, taking the curvature into account. The aponeuroses and fascicles are extrapolated in cases where the entire fascicle is not within the field of view.

The most challenging parts were the aponeuroses detection and the fascicle length estimation. One of the main reasons for this was the formulas used for representing the aponeuroses and the fascicles. The reconstructed fascicle used for measuring the fascicle length was represented by a 2. degree polynomial, which often fell short in representing the shape of the fascicles. The same problem occurred in the aponeuroses detection, where a straight line in many cases was not representative for the shape of the aponeuroses. A critical part of the algorithm was the detection of the lower aponeuroses. If this part of the algorithm fails, estimations of both pennation angle and fascicle length will fail as well. The lower aponeuroses are very faint in some of the images, which makes the detection of this more challenging than the detection of the upper aponeurosis. The results for the fascicle detection are promising, but the part of the algorithm trying to restrain the shape of the fascicle needs more work and should be more adaptive to the different shapes the fascicles can take on/assume. The method used for finding a reference fascicle for pennation angle calculation seems to work well for estimating a representative angle.

Further work should be focused on finding better ways to represent aponeuroses and fascicles, so that it is possible to represent the true shapes of these. The algorithm would also benefit from a more robust way to detect the aponeuroses, and then especially the lower aponeurosis detection which fails more often than the upper aponeurosis detection.

Another thing that could improve the algorithm would be to redefine the grid to better fit the fascicle-plane. The grid as it is now does not follow the angle of the aponeuroses, but is defined by the lowest point of the upper aponeurosis and the highest point of the lower aponeurosis. If the aponeuroses are skewed, parts of the fascicle plane will therefore not be included in the fascicle detection.

Bibliography

- [1] Santiago Aja-Fernández and Carlos Alberola-López. ‘On the estimation of the coefficient of variation for anisotropic diffusion speckle filtering’. In: *Image Processing, IEEE Transactions on* 15.9 (2006), pp. 2694–2701.
- [2] Menno R Bénard et al. ‘Anatomical information is needed in ultrasound imaging of muscle to avoid potentially substantial errors in measurement of muscle geometry’. In: *Muscle & nerve* 39.5 (2009), pp. 652–665.
- [3] Svein Brekke. ‘Techniques for enhancement of temporal resolution in three-dimensional echocardiography’. PhD thesis. Norwegian University Of Science and Technology, 2007.
- [4] H.P.V. C. *Method and means for recognizing complex patterns*. US Patent 3,069,654. Dec. 1962. URL: <https://www.google.com/patents/US3069654>.
- [5] Neil J Cronin et al. ‘Automatic tracking of medial gastrocnemius fascicle length during human locomotion’. In: *Journal of applied physiology* 111.5 (2011), pp. 1491–1496.
- [6] John Darby et al. ‘Estimating skeletal muscle fascicle curvature from b-mode ultrasound image sequences’. In: *Biomedical Engineering, IEEE Transactions on* 60.7 (2013), pp. 1935–1945.
- [7] Richard O Duda and Peter E Hart. ‘Use of the Hough transformation to detect lines and curves in pictures’. In: *Communications of the ACM* 15.1 (1972), pp. 11–15.
- [8] Alejandro F Frangi et al. ‘Multiscale vessel enhancement filtering’. In: *Medical Image Computing and Computer-Assisted Intervention—MICCAI’98*. Springer, 1998, pp. 130–137.
- [9] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. 3rd ed. Pearson Education, Inc, 2008. ISBN: 9780135052679.
- [10] Lars Hofsøy Breivik et al. ‘Multiscale nonlocal means method for ultrasound despeckling’. In: *Ultrasonics Symposium (IUS), 2014 IEEE International*. IEEE. 2014, pp. 1324–1327.
- [11] Lars Hofsøy Breivik et al. ‘Real-time speckle reduction in ultrasound imaging’.
- [12] *Image Resolution*. 2008 (accessed April 21, 2016). URL: <http://usra.ca/imageresolution.php>.
- [13] Hans Knutsson and Mats Andersson. ‘Robust N-dimensional orientation estimation using quadrature filters and tensor whitening’. In: *Acoustics, Speech, and Signal Processing, 1994. ICASSP-94., 1994 IEEE International Conference on*. IEEE. 1994, pp. V–529.

- [14] Karl Krissian et al. 'Oriented speckle reducing anisotropic diffusion'. In: *Image Processing, IEEE Transactions on* 16.5 (2007), pp. 1412–1424.
- [15] Darwin T Kuan et al. 'Adaptive noise smoothing filter for images with signal-dependent noise'. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 2 (1985), pp. 165–177.
- [16] Jong-Sen Lee. 'Digital image smoothing and the sigma filter'. In: *Computer Vision, Graphics, and Image Processing* 24.2 (1983), pp. 255–269.
- [17] Richard L. Lieber and Jan Fridén. 'Functional and clinical significance of skeletal muscle architecture'. In: *Muscle & nerve* 23 (2000), pp. 1647–1666.
- [18] Falk Mersmann et al. 'Evidence of imbalanced adaptation between muscle and tendon in adolescent athletes'. In: *Scandinavian journal of medicine & science in sports* 24.4 (2014), e283–e289.
- [19] *Non-Local Means Denoising*. 2011-09-13 (accessed May 2, 2016). URL: http://www.ipol.im/pub/art/2011/bcm_nlm.
- [20] Jean Pope. *Medical Physics: Imaging*. 1st ed. Heinemann Educational Publishers, 1999. ISBN: 9780435570941.
- [21] Manku Rana, Ghassan Hamarneh and James M Wakeling. 'Automated tracking of muscle fascicle orientation in B-mode ultrasound images'. In: *Journal of biomechanics* 42.13 (2009), pp. 2068–2073.
- [22] Anne Solberg et al. 'Deriving structural information from robust phase attributes'. In: *2013 SEG Annual Meeting*. Society of Exploration Geophysicists. 2013.
- [23] Thomas L. Szabo. *Diagnostic Ultrasound Imaging: Inside Out*. 2nd ed. Academic Press, 2008. ISBN: 9780123964878.
- [24] James M Wakeling and Avleen Randhawa. 'Transverse strains in muscle fascicles during voluntary contraction: a 2D frequency decomposition of B-mode ultrasound images'. In: *Journal of Biomedical Imaging* 2014 (2014), p. 4.
- [25] Yongjian Yu and Scott T Acton. 'Speckle reducing anisotropic diffusion'. In: *Image Processing, IEEE Transactions on* 11.11 (2002), pp. 1260–1270.
- [26] Yongjian Yu and Scott T Acton. 'Speckle reducing anisotropic diffusion'. In: *Image Processing, IEEE Transactions on* 11.11 (2002), pp. 1260–1270.
- [27] Guang-Quan Zhou, Phoebe Chan and Yong-Ping Zheng. 'Automatic measurement of pennation angle and fascicle length of gastrocnemius muscles using real-time ultrasound imaging'. In: *Ultrasonics* 57 (2015), pp. 72–83.