

3D-Druck von fetalen 3D-Ultraschall Daten

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Medieninformatik und Visual Computing

eingereicht von

Julian Wagner

Matrikelnummer 1326108

an der Fakultät für Informatik
der Technischen Universität Wien

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

Wien, 17. März 2017

Julian Wagner

Eduard Gröller

3D-Printing of Fetal Ultrasound

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Media Informatics and Visual Computing

by

Julian Wagner

Registration Number 1326108

to the Faculty of Informatics

at the TU Wien

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

Vienna, 17th March, 2017

Julian Wagner

Eduard Gröller

Erklärung zur Verfassung der Arbeit

Julian Wagner
Juchgasse 30/1/22, 1030 Wien

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

Wien, 17. März 2017

Julian Wagner

Danksagung

Ich möchte mich bei Herrn Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller und dem Insitut für Computergraphic und Algorithmen bedanken, welche mir diese Arbeit ermöglicht haben. Mein besonderer Dank gilt den Mitarbeitern des Institutes, welche mich in jedem Schritt dieser Arbeit unterstützt haben. Das vorhandene Fachwissen und jegliche Bemühungen machten sich immer bezahlt.

Ich möchte mich auch bei meiner Familie für die gesamte Unterstützung während meines Bachelorstudiums bedanken. Besonders hervorheben möchte ich meine Eltern, deren finanzielle Unterstützung mein Studium erst ermöglicht hat, als auch meine Geschwister, die mir immer wieder Mut machen.

Acknowledgements

i would like to thank Mr. Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller and the Institute for Computergraphic and Algorithms for giving me the opportunity to work on this thesis. I am especially thankful for the help of the employees of the institute, who supported me through the whole process. The provided expertise and efforts always paid off.

I would also like to thank my family for supporting me through my whole bachelor. I wish to especially mention my parents, who made my study possible with their financial support, as my siblings, for all their encouragement.

Kurzfassung

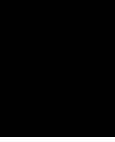
Industrieller 3D-Druck findet bereits seit Jahrzehnten Anwendung. Durch 3D-Druck wird Rapid Prototyping mit niedrigen Kosten ermöglicht. 3D-Drucker für Heimanwendungen wurden ca. im Jahre 2011 relevant. Dank des massiven Anstiegs der Popularität von 3D-Druckern bei Privatpersonen, werden 3D-Drucker immer leistbarer. In dieser Arbeit zeigen wir, wie fetaler 3D-Ultraschall so konvertiert werden kann, dass er durch einen 3D-Drucker gedruckt werden kann. Dazu gehört die Klassifizierung der fetalen Oberfläche, die Extraktion der Isofläche sowie das Glätten des resultierenden 3D Modells. Unser Ansatz basiert auf Schwellwertbildung in Kombination mit Connected Component Analysis, um das Gewebe der Mutter vom Gewebe des Fötus zu unterscheiden. Aus den klassifizierten Daten extrahieren wir darauf die fetale Oberfläche mittels Marching Tetrahedra. Daraufhin wird das Mesh geglättet und in ein Format konvertiert, welches sich zum Druck mit einem 3D-Drucker eignet. Abhängig vom Datensatz, sind wir mit diesem Ansatz fähig ein Fabrikat des Fötus zu erzeugen, auf welchem das Gesicht und periphere Strukturen erkennbar sind.

Abstract

3D printing has been used industrially for decades. It enables rapid prototyping while maintaining low costs. Personal 3D printing became popular approximately since 2011. Since the massive arise of public interest, 3D printers are getting more and more affordable. In this thesis, we show how fetal 3D ultrasound data can be processed to enable 3D printing. Major steps are classification of the tissues, extraction of the isosurface and mesh-smoothing. Our approach uses thresholding, in combination with Connected Component Analysis, to separate the mother tissues from the fetal tissues. From the labeled data, we extract the fetal surface using Marching Tetrahedra. The mesh is then smoothed and converted into a data format suitable for 3D printing. Depending on the quality of the given ultrasound data, we can generate a model with recognizable facial features and peripheral structures.

Contents

Kurzfassung	xi
Abstract	xiii
Contents	xv
1 Introduction	1
2 State of the art	3
2.1 3D Ultrasound	3
2.2 3D Printing	6
3 Methodology	9
3.1 Classification of Fetal Tissue	9
3.2 Segmentation Post-processing	9
3.3 Isosurface Extraction	10
3.4 Mesh Post-processing	11
3.5 STereoLithography	13
4 Implementation	15
5 Results	17
6 Conclusion	23
List of Figures	25
Bibliography	27



Introduction

In this thesis, we utilize existing 3D printing technology to fabricate models of fetuses from 3D ultrasound data. As our goal is to fabricate models with recognizable facial features, our datasets focus on the faces of the fetuses. The resulting models help parents to see and feel their child before birth. Therefore, our focus is not on exact representation of the data, but on delivering aesthetic appearance and recognizable models.

3D printing is not a new technology. Industrial prototyping has been successfully used since decades. The first attempts to create 3D objects using core concepts of 3D printing technology date back to the 1960s [1]. In the last years, the technology has improved in a way, that makes consumer-oriented 3D printers affordable. Filament-based additive 3D printers are now widely available for the consumer market. Nascimento et al. state that „We are witnessing a rise in new Do-It-Yourself (DIY), crafting, manufacturing, hacking, fabbing, or making paradigms where a mix of tools, communities and spaces are increasingly enabling more and more people to produce and share knowledge at a quicker pace, create their own material and symbolic solutions, and define the goals and outcomes of their technological actions“ [2]. This movement, often referred to as maker culture, has rapidly evolved around the development of 3D printing.

3D printing technology starts to be used in the medical domain. For example, an atrial septum has been reconstructed using 3D printing and 3D ultrasound to treat congenital heart disease [3]. The procedure of the evaluation of the atrial septal defect can be seen in Figure 1.1. An application in the pharmaceutical field is 3D printing of tablets containing multiple drugs with defined release profiles [4].

Fetal ultrasound is a routine procedure in prenatal care. It enables the medical specialist to monitor growth and development of the baby. The progress of the pregnancy can be better determined. The doctor can find congenital abnormalities, such as malformed or missing limbs, and determine whether the child is a boy or a girl. In case of problems, the doctor has more time to diagnose and treat them.

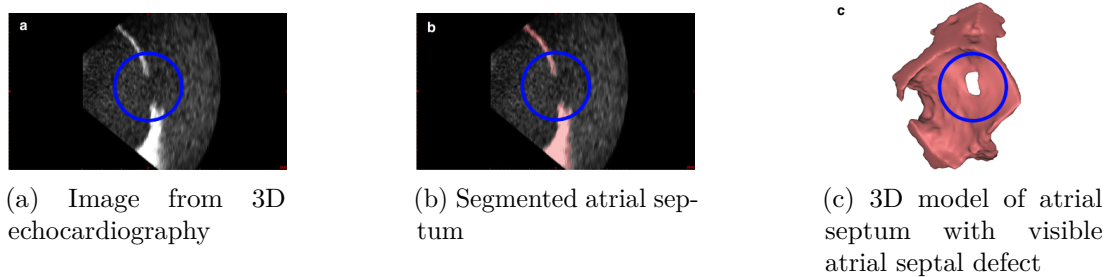


Figure 1.1: Evaluation of atrial septal defect [3]

Problem

This thesis is concerned with transforming scalar volume data into a mesh, suitable for 3D printing.

The datasets in this thesis consist of volume data with a typical dimension of about 300^3 . The values are stored as single bytes, i.e., the range is from zero to 255. Volume data does only implicitly contain information about structure. The structure can be extracted as an isosurface. Volume data contains the intensities of the ultrasound beam in discrete locations, inside the volume, as voxels. In 3D printing, we need models with closed surfaces. Because we need to find a surface that represents the fetus, we define our surface as the enclosure of all voxels belonging to the fetus. We still have to be aware of certain ultrasound artifacts, e.g., speckle and shadowing, which are discussed later in this thesis alongside filtering techniques.

The ultrasound data contains not only the fetal tissues, but also mother tissues and amniotic fluid. Mother tissues have to be discarded, because they occlude the fetus. In order to classify the fetal tissues, we inspect the areas where fetal tissues and other tissues or fluids meet. Ideally, we classify all the fetal tissues correctly and can discard everything else. By discarding everything except the fetus, we make sure that there are no occluders. However, we expect minor imperfections in the classification, e.g., there could be parts of fetal tissues that are not connected to each other due to shadowing [5]. We assume that the largest connected component of classified fetal tissues is the actual fetus and discard smaller elements due to limitations of 3D printing.

After we have classified the voxels, we extract the surface. The most common algorithm is Marching Cubes [6], alternatively Marching Tetrahedra [7]. Marching Tetrahedra as compared to Marching Cubes, has the advantage of not producing holes in the surface due to ambiguities. The result of either algorithm is a triangular mesh, representing the surface of the fetus. But due to imperfections in the ultrasound data and inaccuracies in the surface extraction process, we have to smooth the model before 3D printing. There are many smoothing options with their respective benefits and disadvantages. Main aspects like computing time and mesh shrinkage have to be considered.

CHAPTER 2

State of the art

The two core technologies, this work builds upon, are fetal 3D ultrasound and additive 3D printing. Ultrasound is a well established modality within the medical field. 3D printing is becoming more and more important and affordable. Researchers in the medical field have also discovered the vast applications of 3D printing, such as for implant and tissue design [8].

2.1 3D Ultrasound

2D ultrasound can be traced back to 1952 in the work of Wild and Reid [9]. Today medical 2D ultrasound is widely used and is considered as an indispensable imaging modality. While historically being mostly limited to applications such as cardiology and gynaecology, it is now also being applied to fields such as image-guided surgery and therapy [10].

Conventional 2D ultrasound suffers from limitations that can be addressed by 3D imaging. According to Fenster et al. these are the following [10]:

- In order to examine anatomy and pathology, sonographers have to combine several 2D images in their minds to form a 3D impression. This is an error-prone, time-consuming and subjective process. This makes it more difficult for medical specialists to give correct diagnosis, planning and delivery of therapy.
- There are issues when trying to gather accurate estimations of organ or tumor volume. This is because traditional techniques just take measurements of height, width and length in two orthogonal views, while assuming an idealized (e.g. ellipsoidal) shape.
- Another limitation is the reproducibility of navigation, e.g., for monitoring therapeutic procedures, because of the complexity of manually positioning the transducer.
- Some viewing directions are not physically possible compared to fully tomographic methods, such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI) or Positron Emission Tomography (PET).

In 3D ultrasound imaging, a computer compiles multiple 2D ultrasound images to a 3D volume. The practitioner is then able to view, interact with, measure and manipulate the resulting volume. 3D scanning techniques have to be either gated or rapid, in order to avoid artifacts due to (involuntary internal) motion of the suspect, e.g., breathing.

3D ultrasound also inherits some problems of its 2D counterpart, because 3D ultrasound volumes are composed of multiple 2D images after all. One of the main considerations when dealing with ultrasound is unwanted noise, which shows up in multiple variations. These artifacts generally lead to texture patterns that do not reflect the real tissue.

One major source of ultrasound signal degradation is acoustic speckle [5]. Speckle shows up as a granular structure, that consists of dark and bright spots. Speckle is the result of coherent accumulation of random scattering events within the resolution cell, and does not correspond to the actual tissue microstructure [11]. Burckhard states that „When an object is scanned twice under the same conditions, one obtains identical speckle patterns. Although of random appearance, speckle is therefore not random in the same sense as, e.g., electrical noise. If the same object is, however, scanned under different conditions, say different transducer aperture, pulse length, or transducer angulation, the speckle patterns are different.“ [12]. Speckle hides small differences in grey level. Several filters can reduce speckle noise. Luqman et al. [13] recommend the Wiener filter (2.1d), but also mention median (2.1b) and average filtering (2.1c).

Another common artifact is reverberation. Reverberation occurs when there are two or more parallel strong reflectors in the sound path. The ultrasound beam is then reflected back and forth between the reflectors. Therefore, the waves take longer to return to the transducer. This leads to reflectors in the resulting image that are not present in the tissue [5]. Reverberation can be reduced by repositioning the transducer at an angle so

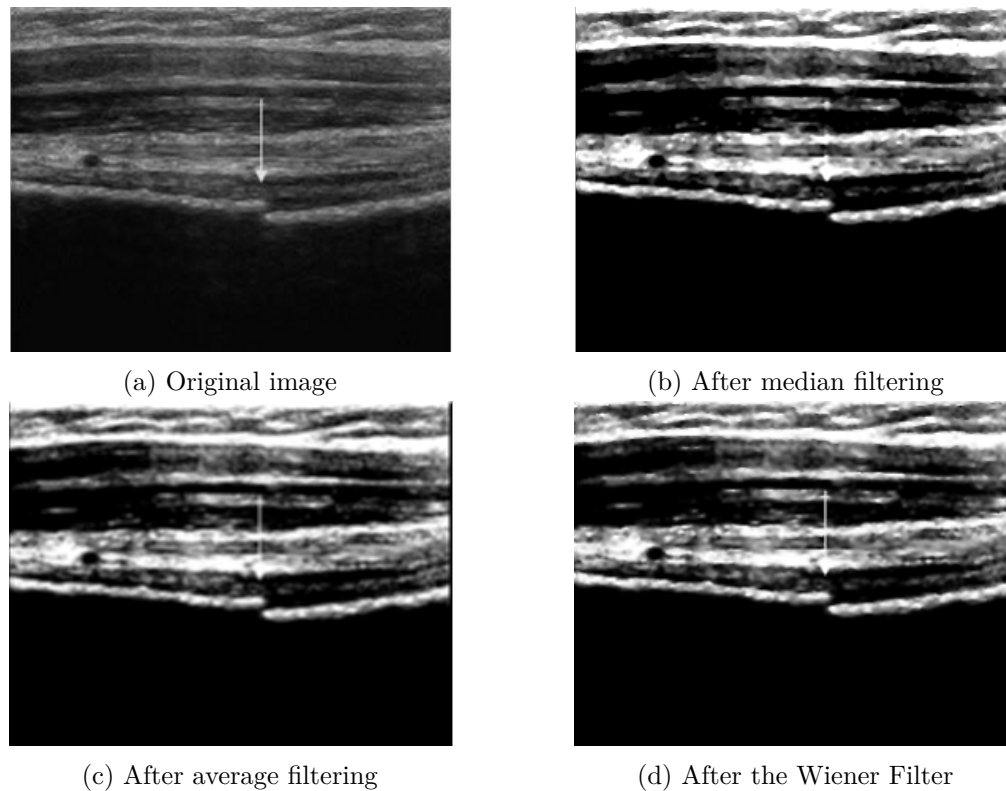


Figure 2.1: Comparison of several filtering techniques on an image of proximal humerus shaft fracture ultrasound [13]

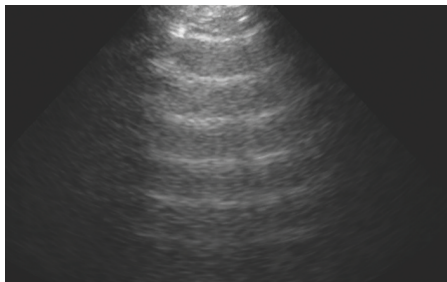


Figure 2.2: An example of reverberation [14]



Figure 2.3: An example of shadowing [15]

that the beam does not hit a second strong reflector. An example of reverberation can be seen in Figure 2.2.

Shadowing is mostly caused by solid structures like bones or stones. Such structures result in high reflection or strong absorption of the ultrasound signal. On the resulting image, the shadow is a signal void behind the structure [5]. An example of shadowing is

shown in Figure 2.3.

2.2 3D Printing

3D printing is enabling rapid prototyping since the late 80's [16]. In the past, it has almost exclusively been used in the industry. It gave designers the opportunity to produce quick prototypes and gather early feedback from customers and stakeholders. Today 3D printing is widely available, easily affordable and has extensively expanded its target audience. Gartner says „Worldwide shipments of 3D printers will reach 496,475 units in 2016, up 103 percent from the predicted 244,533 units in 2015“ [17].

There is a variety of different types of 3D printers. This thesis focuses on 3D printers implementing additive manufacturing processes, in contrast to 3D printers using subtractive processes. Additive manufacturing processes can be divided into selective binding, selective solidification, or selective deposition [18].

Selective binding uses a form of powder as printing material. Used materials for the powder are gypsum, nylon or metal. The fine powder particles are then fused together, using an adhesive medium or heat. Heat could be applied, e.g., from a laser, like in selective laser sintering [19]. The powder is fused together layer by layer. The resolution allows complex and fine prints, because the powder itself acts as supporting medium for the print. Drawbacks are the handling of the fine powder and the expensive price.

Selective solidification generates a solid object from a liquid material. This is done by applying energy to solidify the liquid layer by layer. The first layer is usually fused to a platform that moves down into the liquid. An example of selective solidification is stereolithography, where a resin is solidified using a laser. As light is used to solidify the resin, only photosensitive materials can be used [20]. The models often need to be preserved afterwards, e.g., through hardening with ultraviolet light. Like the powder in selective binding, the resin can be hard to handle.

Printers implementing **selective deposition** place the material only there where it is needed. Standard consumer-faced filament-based printers fall into this category. This is also the type of printers we are focusing on in this thesis.

Models, that are intended to be printed with this type of printer have to meet certain requirements. They have to be 2D-manifolds and watertight. On a 2D-manifold surface, each point is shared by exactly one patch of the surface. Watertight means that the surface of the mesh does not contain holes.

The most common file format for models designed for consumer 3D printers is STereolithography (STL). STL is a very simple format, that consists of only a list of triangles and their normals. It does save neither color nor topological information. The STL file format will be discussed further in Section 3.5.

As the printer generates the model layer by layer, from the bottom to the top, one has to consider support structure. Let us consider the ease of 3D printing a tree with a branch

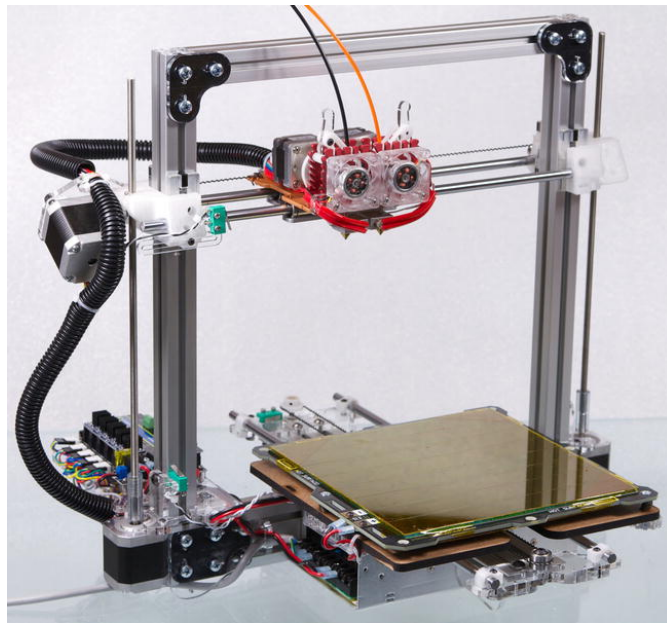
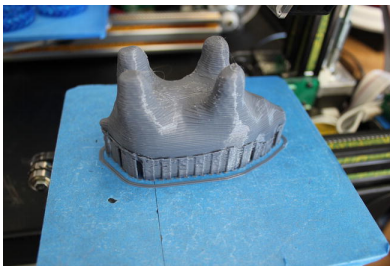


Figure 2.4: A filament-based printer. [18]



(a) A printout of a bear with rounded back [18]



(b) The printout after removal of the support structure [18]

Figure 2.5: Printout process of a bear model

horizontal to the ground. The extruder would have to place the layer of filament in the air. Therefore, support structure is needed. It has to be removed mechanically after the print. For aesthetically pleasing results, techniques like sanding are often used to smooth out irregularities on the printout. Often support structure can be avoided by proper positioning the digital model before 3D printing, so that it is, e.g., printed on its side. In Figure 2.5a the printout of a bear with a rounded back can be seen. This print would not be possible without the support structure. Figure 2.5b shows the printout after the support structure has been removed. The scarring on top shows clearly where the support structure has been.

Methodology

We implemented a semi-automatic approach for 3D printing fetal ultrasound data. The user has to determine a threshold for the classification of the fetal tissue. From there we automatically determine the largest connected component, extract the isosurface, smooth the resulting model and convert it to a format suitable for 3D printing. An overview of our technique can be seen in Figure 3.1.

3.1 Classification of Fetal Tissue

Considering the nature of the datasets, segmentation of the fetal tissues from the mother tissue and amniotic fluid is needed to get a printable representation of the fetus. We combine two techniques. Initially, the user chooses a threshold value. Values that are below this threshold are considered as not belonging to the fetus. This step requires user interaction, because the data contains noise and the threshold values differ from patient to patient.

In our datasets, even manually-tuned thresholds lead to multiple components, because the value ranges of mother tissues and the fetal tissues overlap.

3.2 Segmentation Post-processing

Connected Component Analysis (CCA) allows us to find and label connected components in a graph [21]. By representing the segmentation mask as a graph, we can identify and remove all connected components except the fetus. Each voxel is considered as a vertex, between two neighbouring voxels we define an edge.

The algorithm traverses the voxels. Each unlabeled matching voxel is labeled with a new unique label. Its neighbors that match the criterion, i.e. the chosen threshold, are added to a queue. The process is repeated until the queue is empty.

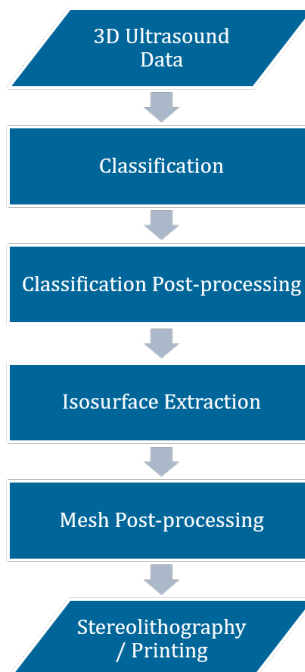


Figure 3.1: Pipeline of our technique

Pseudo-code of CCA can be examined in Algorithm 3.1.

The result of CCA is a segmentation mask, where all connected components are labeled with unique numbers. For the next steps, all connected components, but the largest, are discarded because we consider the largest connected component as the fetus.

3.3 Isosurface Extraction

Marching Tetrahedra is an algorithm we use to extract an isosurface within a 3D volume [22]. Among other applications, the Marching Tetrahedra algorithm is used in medical image processing. Modalities such as CT, MRI and 3D ultrasound produce 3D grids of voxels.

Given a scalar field sampled on a regular 3D grid and an isovalue, the algorithm marches through each grid cell, which is divided into six tetrahedra as shown in Figure 3.2. For each tetrahedron in each cell the algorithm then determines how the isosurface intersects it. To do so, Marching Tetrahedra assigns a one if a tetrahedron’s vertex exceeds or equals a given isovalue, and a zero otherwise. Vertices with the value one are considered to be inside the surface, if the value is zero it is considered outside. If there is at least one vertex inside and one vertex outside, the tetrahedron is considered intersected by the isosurface. The algorithm determines the exact intersection points at the edges of the tetrahedron using a lookup table of intersection configurations.

Algorithm 3.1: Connected Component Analysis

```

1 Input: V - Array - Volume
2       T - Threshold
3 Output: L - Array - Labels
4 Q - queue of voxels
5 P - Array - Flags for already processed voxels, initialised with false
6 currentLabel  $\leftarrow$  0;
7 foreach voxel  $v \in V$  do
8   if  $!P[v]$  and  $v \geq T$  then
9     P[v]  $\leftarrow$  true;
10    Q.enqueue(v);
11    while  $!Q.isEmpty()$  do
12      currentElement  $\leftarrow$  Q.dequeue();
13      L[currentElement]  $\leftarrow$  currentLabel;
14      foreach voxel  $b \in neighbourOf(currentElement)$  do
15        if  $!P[b]$  and  $b \geq T$  then
16          Q.enqueue(b);
17          P[b]  $\leftarrow$  true;
18        end
19      end
20      currentLabel++;
21    end
22  end
23 end

```

There are only 16 possible configurations of how a tetrahedron can be intersected by a surface. Since we labeled each vertex, we use a lookup table to find the right configuration. The lookup table can be examined in Figure 3.3. The exact intersection points of the surface are then found by interpolation.

Marching Tetrahedra is similar to its predecessor Marching Cubes. Marching Tetrahedra solves ambiguity issues that may lead to holes in the resulting mesh, that the Marching Cubes algorithm has. For further information on Marching Cubes, we refer the reader to Lorensen et al. 1987 [6].

3.4 Mesh Post-processing

Laplacian smoothing is one of the most common algorithms for mesh smoothing. The algorithm computes the neighbours v_j for each vertex v_i . The new position of a vertex \hat{v}_i is then computed as the average of all its neighbours. The algorithm runs iteratively and after a few iterations unwanted mesh shrinkage can be observed [23].

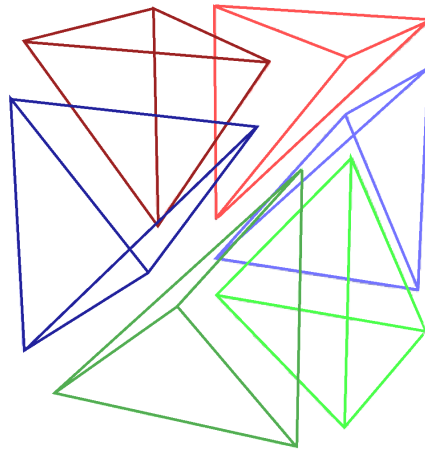


Figure 3.2: Cube split into tetrahedra [22]

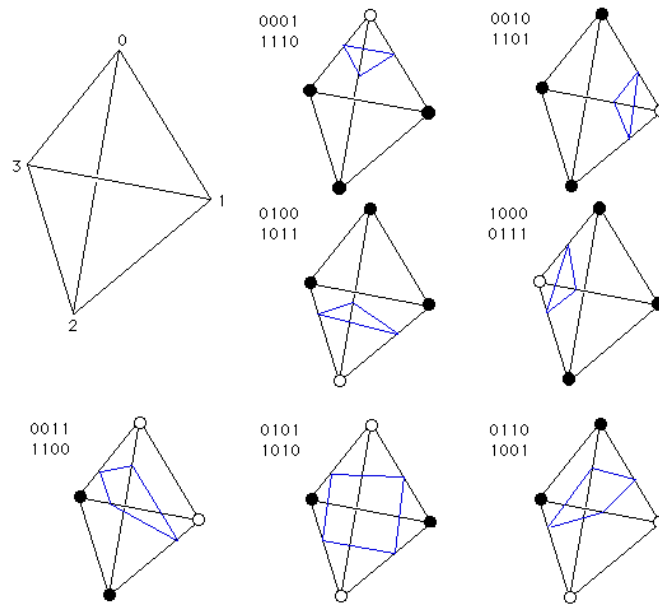


Figure 3.3: Configurations for Marching Tetrahedra [22]

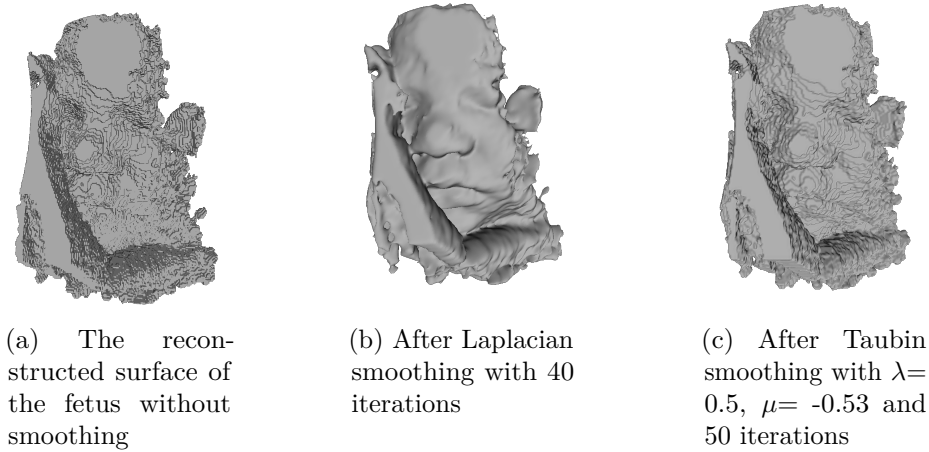


Figure 3.4: Smoothing of the fetus surface

Laplacian smoothing is formally described as

$$\hat{v}_i = \frac{1}{N} \sum_{j=1}^N v_j \quad (3.1)$$

where N is the number of neighbouring vertices v_j .

In order to avoid surface shrinking, we implemented Taubin smoothing [24]. But as shown in Figure 3.4c it produces unwanted ridges. We switched to Laplacian smoothing, which can be seen in Figure 3.4b. The surface shrinkage after 40 iterations is acceptable.

3.5 STereoLithography

The STereoLithography file format, commonly referred to as STL, is used for 3D printing, rapid prototyping and computer-aided manufacturing. It describes the geometric surface as a list of triangles and their normals. Each triangle is described with three vertices. Each vertex and normal is described through a set of X , Y , and Z coordinates. STL files exist in two variations: Binary and ASCII. The exact structure can be seen in Figure 3.5. The first is encoded as ASCII, and the second is binary. The ASCII format is larger in file size, but can be read directly by humans. STL is the de-facto standard for rapid prototyping, but it has some disadvantages. For example, it is not indexed, which means that it carries redundancies [25].

Implementation

The software for this thesis was implemented using C++. For the user interface, we used the QT Framework. We allow the user to traverse through the slices of the ultrasound data on the three major axes. The interface for slice traversal can be seen in Figure 4.1a. Slice traversal is important for finding the appropriate threshold value. In the slice views, we display everything below the chosen threshold with blue color. The process is illustrated in Figure 4.1b. For the real-time rendering of the slices, we use OpenGL.

Besides the controls for the slice views, there are controls to load the volume data, a control to choose the threshold, a progress bar and a button to extract, process and save the STL model. The volume is provided as raw data with a separate header file.

To extract the isosurface, we use the Marching Tetrahedra algorithm. By doing so, we do not have to deal with ambiguities.

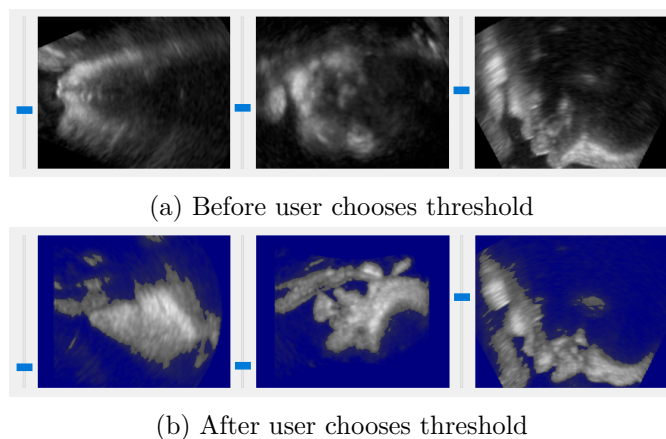


Figure 4.1: Slice views of the data

It is important to note, that we did not use the original volume data for this step, but the binary mask produced by the CCA algorithm instead. As isovalue we chose 0.5. The reason behind this, is to remove numerous small fragments unrelated to the fetus. Doing so we can quickly skip these components while marching the data. Using the real values would not guarantee a smoother result, because small differences may create sharp edges, that we would smooth out anyway. Considering the Laplacian smoothing, the gained additional accuracy within the tetrahedra cells would be negligible.

In Figure 3.4a we show the reconstructed surface after Marching Tetrahedra. The fetus is not occluded by mother tissue and consists of one piece without holes in the surface.

For performance reasons, we implemented the algorithm on the GPU using OpenGL. The algorithm is implemented as vertex shader. After completion, a binary STL file is placed in the same directory as the program.



Results

We show our method with three datasets from actual 3D ultrasound examinations. Each dataset is visualized with a slice view, a DVR image from MeVisLab, mesh view from MeshLab and a photo of the actual printout. Dataset #1 has been printed with a lower resolution than the other two datasets. 3D Printing at a higher resolution seems to cause small ridges, as seen in Figure 5.2 and Figure 5.3. We consider the cause of this to be related to the printing process itself. The faces are recognisable on all three printouts. The DVR images in MeVisLab look quite similar to the final printouts, which indicates that the presented process in this thesis produces models that represent the input data satisfyingly.

5.0.1 Dataset #1

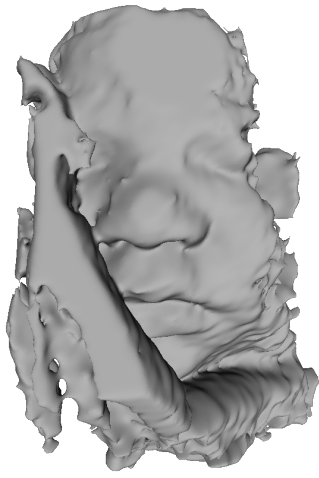
The printout from Dataset #1, as can be seen in Figure 5.1, offers the most recognizable face of all three datasets, despite being printed with a lower resolution than the other two datasets. This is due the superior quality of the input data and the state of applied support structure. There is no scarring from the support structure on the face itself. Scarring can be seen on the inner forearm as shown in Figure 5.1b. One can observe that eyes, nose, mouth and part of the arm are visible. If we compare Figure 5.1a and Figure 5.1b, we can see that the left eye has been partially occluded due to support structure.

5.0.2 Dataset #2

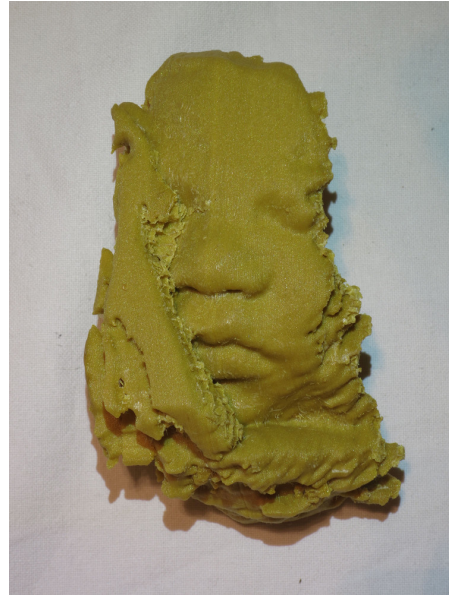
Dataset #2 shows mouth, nose, part of the arm and part of the eyes, as can be seen in Figure 5.2b. The right eye seems to have been occluded during the examination. One can observe that the nose is skewed. This seems to be a result of movement during the examination.

5.0.3 Dataset #3

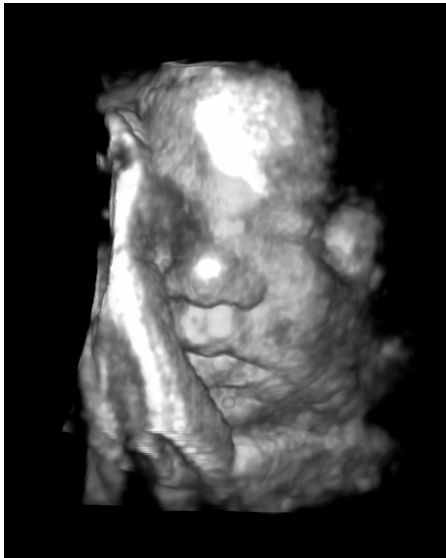
Dataset #3 depicts only half of the face, as can be seen in Figure 5.3b. This is a result of shadowing artifacts during the examination. The basic facial structures, one eye and part of the mouth and nose are still visible.



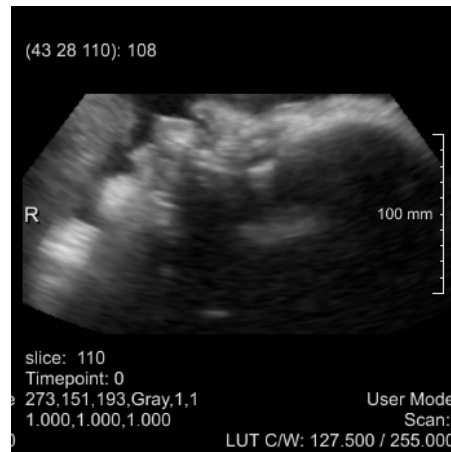
(a) The model in meshlab



(b) The printout

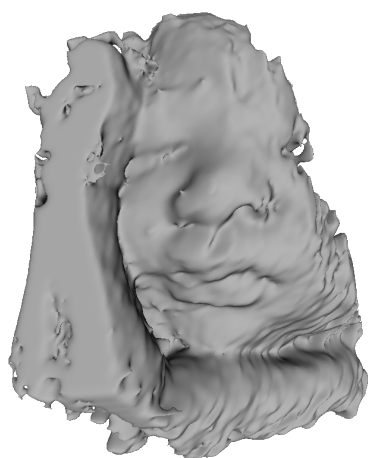


(c) The volume rendered in MeVisLab

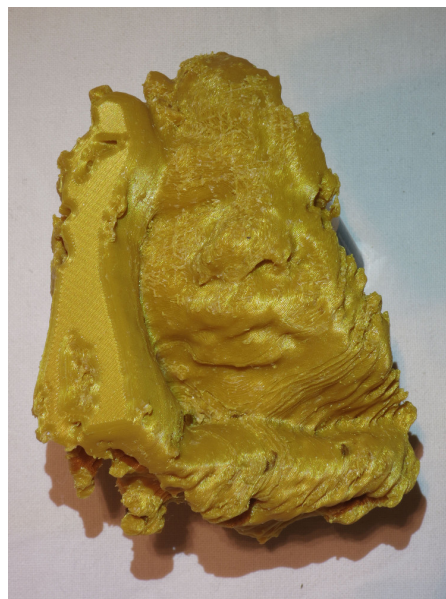


(d) Sliceview in MeVisLab

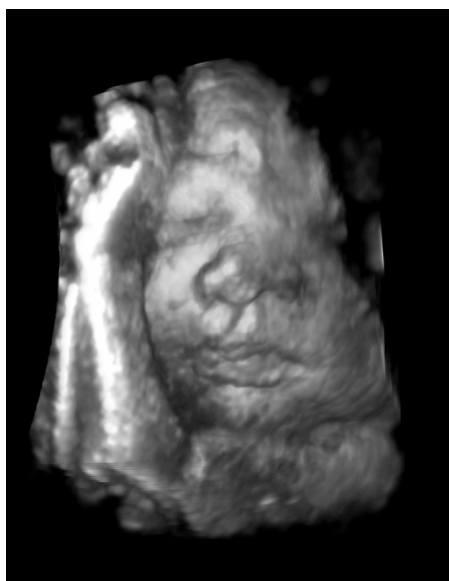
Figure 5.1: Dataset #1



(a) The model in meshlab



(b) The printout

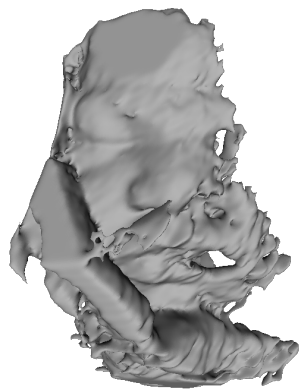


(c) The volume rendered in MeVisLab

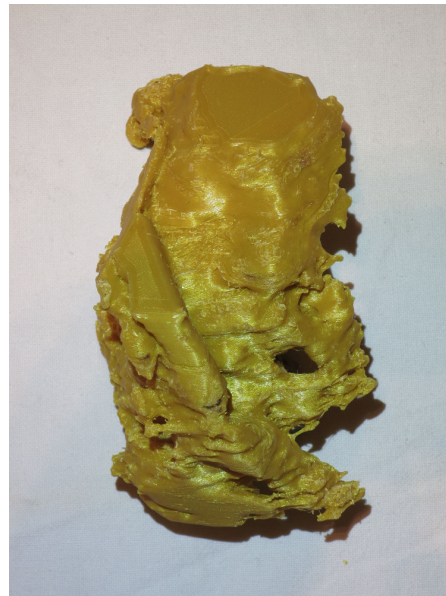


(d) Sliceview in MeVisLab

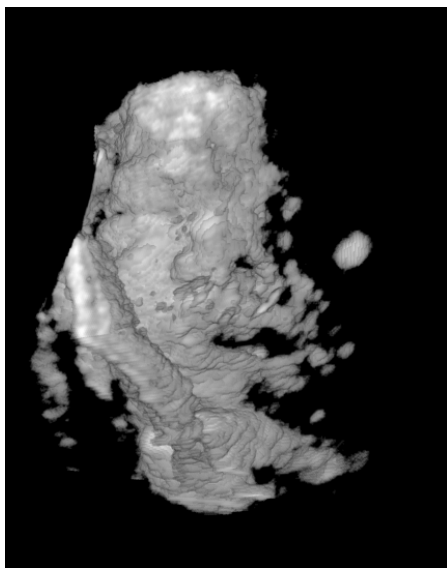
Figure 5.2: Dataset #2



(a) The model in meshlab



(b) The printout



(c) The volume rendered in MeVisLab



(d) Sliceview in MeVisLab

Figure 5.3: Dataset #3

Conclusion

We show that it is possible to fabricate 3D fetal ultrasound data while preserving major facial features, nose, mouth and eyes. The printouts are quite close to the DVR images produced in MeVisLab. during the course of this thesis we created software that covers the entire pipeline from medical volume data, to a printable mesh.

Certain aspects still can be improved. The Marching Tetrahedra step has low performance. It can be improved, e.g., by implementing it on the GPU. A further improvement would be to support the user better during data inspection, e.g., with a preview window that shows the approximate outcome of the mesh and offers interaction.

As expected, the details of the results are not highly medical relevant. Printouts that are medically accurate could be used to inspect anomalies for example. The accuracy of the printouts could be improved by taking tolerances of 3D printing into account in order to perform medically exact measurements. The post-processing step could be optimized to avoid mesh shrinkage.

The fabricates that our approach yields can be used, e.g., to allow a parent that is blind or otherwise visually impaired feel and experience their unborn child like other parents do with ultrasound images.

List of Figures

1.1	Evaluation of atrial septal defect [3]	2
2.1	Comparison of several filtering techniques on an image of proximal humerus shaft fracture ultrasound [13]	5
2.2	An example of reverberation [14]	5
2.3	An example of shadowing [15]	5
2.4	A filament-based printer. [18]	7
2.5	Printout process of a bear model	7
3.1	Pipeline of our technique	10
3.2	Cube split into tetrahedra [22]	12
3.3	Configurations for Marching Tetrahedra [22]	12
3.4	Smoothing of the fetus surface	13
3.5	ASCII and binary STL format	14
4.1	Slice views of the data	15
5.1	Dataset #1	19
5.2	Dataset #2	20
5.3	Dataset #3	21

Bibliography

- [1] George C. Dumitrescu and Ion A. Tanase. 3D printing - a new industrial revolution. *Knowledge Horizons.Economics*, 8(1):32–39, 2016.
- [2] Susana Nascimento and Alexandre Pólvara. Maker cultures and the prospects for technological action. *Science and Engineering Ethics*, pages 1–20, 2016.
- [3] Bennett P. Samuel, Candida Pinto, Todd Pietila, and Joseph J. Vettukattil. Ultrasound-derived three-dimensional printing in congenital heart disease. *Journal of Digital Imaging*, 28(4):459–461, 2015.
- [4] Shaban A. Khaled, Jonathan C. Burley, Morgan R. Alexander, Jing Yang, and Clive J. Roberts. 3D printing of tablets containing multiple drugs with defined release profiles. *International Journal of Pharmaceutics*, 30 October 2015, Vol.494(2), pp.643-650.
- [5] F W Kremkau and K J Taylor. Artifacts in ultrasound imaging. *Journal of Ultrasound in Medicine*, 5(4):227–37, 1986.
- [6] William E. Lorensen and Harvey E. Cline. MARCHING CUBES: A high resolution 3D surface construction algorithm. *Computer Graphics (ACM)*, July 1987, Vol.21(4), pp.163-169.
- [7] Doi Akio and Akio Koide. An efficient method of triangulating equi-valued surfaces by using tetrahedral cells. *nstitute of Electronics, Information and Communication Engineers Transactions on Information and Systems*, 74(1):214–224, 1991.
- [8] F. Rengier, A. Mehndiratta, H. von Tengg-Kobligk, C. M. Zechmann, R. Unterhinninghofen, H.-U. Kauczor, and F. L. Giesel. 3D printing based on imaging data: review of medical applications. *International Journal of Computer Assisted Radiology and Surgery*, 5(4):335–341, 2010.
- [9] John J. Wild and John M. Reid. Application of echo-ranging techniques to the determination of structure of biological tissues. *Science*, 115(2983):226–230, 1952.
- [10] Aaron Fenster, Dónal B Downey, and H Neale Cardinal. Three-dimensional ultrasound imaging. *Physics in Medicine and Biology*, 46(5):R67, 2001.

- [11] R. F. Wagner, S. W. Smith, J. M. Sandrik, and H. Lopez. Statistics of speckle in ultrasound B-Scans. *IEEE Transactions on Sonics and Ultrasonics*, 30(3):156–163, May 1983.
- [12] C. B. Burckhardt. Speckle in ultrasound B-mode scans. *IEEE Transactions on Sonics and Ultrasonics*, 25(1):1–6, Jan 1978.
- [13] M. L. B. Muhd Zain, I. Elamvazuthi, and M. Begam. Comparative analysis of filtering techniques for ultrasound images of bone fracture. In *2010 International Conference on Intelligent and Advanced Systems*, pages 1–4, June 2010.
- [14] Brady Pregeson. Pardon the interference [interactive ultrasound case study]. <http://epmonthly.com/article/pardon-the-interference/>. Accessed October 18, 2016.
- [15] Henry Knipe and Matt Skalski et al. Acoustic shadowing. <https://radiopaedia.org/articles/acoustic-shadowing>. Accessed October 18, 2016.
- [16] Stacy Rauen. 3D printing. *Hospitality Design, Aug 2015, Vol.37(6), pp.49-50*.
- [17] Viveca Woods. Gartner says worldwide shipments of 3D printers to reach more than 490,000 in 2016. <http://www.gartner.com/newsroom/id/3139118>. Accessed October 21, 2016.
- [18] Joan Horvath. *Mastering 3D Printing*. Apress: Berkeley, CA, Berkeley, CA, 2014.
- [19] J-P. Kruth, P. Mercelis, J. Van Vaerenbergh, L. Froyen, and M. Rombouts. Binding mechanisms in selective laser sintering and selective laser melting. *Rapid Prototyping Journal*, 11(1):26–36, 02 2005.
- [20] Jannis Breuninger, Ralf Becker, Andreas Wolf, Steve Rommel, and Alexander Verl. *Generative Fertigung mit Kunststoffen; Konzeption und Konstruktion für selektives Lasersintern*, 2013.
- [21] Michael B. Dillencourt, Hanan Samet, and Markku Tamminen. A general approach to Connected-component Labeling for arbitrary image representations. *Journal of the ACM*, 39(2):253–280, April 1992.
- [22] Paul Bourke. Polygonising a scalar field using tetrahedrons, 1994. [Accessed 27 September 2016].
- [23] Hicham Badri, Mohammed El Hassouni, and Driss Aboutajdine. Kernel-based Laplacian smoothing method for 3D mesh denoising.
- [24] G. Taubin. Curve and surface smoothing without shrinkage. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 852–857, Jun 1995.
- [25] Vinod Kumar and Debasish Dutta. An assessment of data formats for layered manufacturing. *Advances in Engineering Software*, 28(3):151 – 164, 1997.