Master thesis on Computational Biomedical Engineering
Universitat Pompeu Fabra

Ultrasound segmentation for vascular network reconstruction in twin-to-twin transfusion syndrome

Enric Perera Bel
Supervisor: Miguel Ángel González Ballester
Co-supervisor: Mario Ceresa

July 2017
Master thesis on Computational Biomedical Engineering
Universitat Pompeu Fabra

Ultrasound segmentation for vascular network reconstruction in twin-to-twin transfusion syndrome

Enric Perera Bel
Supervisor: Miguel Ángel González Ballester
Co-supervisor: Mario Ceresa

July 2017
# Table of Contents

1. Introduction .......................................................................................................................... 1
   1.1. Twin-to-twin transfusion syndrome ............................................................................. 1
   1.1.1. Current treatments ................................................................................................. 2
   1.2. Mapping of the placenta vascular network .................................................................. 3
   1.2.1. Image acquisition techniques ............................................................................... 4
   1.2.2. Segmentation ......................................................................................................... 4
   1.3. Justification ................................................................................................................... 7
   1.4. Goal ............................................................................................................................... 8
2. Methods ................................................................................................................................ 9
   2.1. Integration of methods ............................................................................................... 9
   2.2. Segmentation ............................................................................................................... 10
   2.2.1. Random walker segmentation method ................................................................... 12
   2.2.1.1. Algorithm implementation ................................................................................. 15
   2.2.1.2. GPU optimization ............................................................................................. 17
   2.2.2. Placenta segmentation ......................................................................................... 18
   2.2.3. Vascular network segmentation .......................................................................... 20
   2.2.4. Placenta and vessel fusion .................................................................................... 21
3. Results .................................................................................................................................. 24
   3.1. Segmentation ............................................................................................................... 24
   3.2. Segmentation time ....................................................................................................... 28
   3.3. GPU optimization ....................................................................................................... 30
4. Discussion .............................................................................................................................. 34
5. Conclusions ........................................................................................................................... 38
6. Bibliography .......................................................................................................................... 40
Abstract

In this work we present a placenta and vessel segmentation method for a medical application for Twin-to-Twin Transfusion Syndrome (TTTS). TTTS is a fetal disease that occurs in twin monochorionic pregnancies and can be fatal if left untreated. Right now it is treated with fetoscopic laser coagulation. This method highly improves prognosis, but still presents some risks since the intervention is critical in order to avoid abortion risks. Therefore, it can benefit from image segmentation techniques for surgery planning and guidance. Placenta segmentation is not easy due to a high variability on its location and shape, thus semiautomatic methods are the ones that have shown better results for ultrasound (US) segmentation. We implement one of them, the random walker (RW) algorithm, and include it in a graphic user interface for medical use. Thirty-one sets of US and Doppler US images were available in this study, but four are discarded due to poor gradient quality between tissues. Individual segmentation of placenta and vessel from different images is performed (US and Doppler US, respectively), as well as combined in a multi-label segmentation (Doppler US). The implemented method is compared with previous studies, and it is modified in order to accelerate its computation using a graphics processing unit (GPU). We show that this algorithm offers a fine boundary adherence for US images for both placenta and vessel segmentation, mostly in regions with high tissue gradients, but it is dependent and sensitive on the protocol followed for the manual initialization, which is in concordance with the literature study. We also observe that single and multiple segmentation show similar segmentation results, mostly in vessel and not so much in placenta. The GPU implementation shows faster computation rates, but needs of more iterations to converge to a solution, compared to the already optimized CPU implementation. However, using a high end graphics card accelerates the overall computation, while there is still room for improvement.

The RW algorithm had already been used for placenta segmentation and we have validated its accuracy. However, there does not exist a gold standard in this procedure so we plan on including more methods in the medical application, so the clinician can choose the approach that fits the best to each anatomy and image characteristics.

In this project we tightly collaborate with BCNatal | Barcelona Center for Maternal Fetal and Neonatal Medicine Hospital Clínic and Hospital Sant Joan de Déu, Universitat de
Barcelona. We aim to create a surgery planning and tracking tool that can be used to improve fetoscopic laser coagulation prognosis and, later, that can be extended to other surgeries.

**Keywords:** Placenta segmentation; Vessel segmentation; Random walker algorithm; GPU optimization; Medical application
1. Introduction

Fetal surgery is increasingly being used to improve the outcome of pregnancies with complications. One of the most common is twin-to-twin transfusion syndrome (TTTS) which can be fatal if left untreated. In order to properly plan the intervention, the surgeon needs a complete map of the vascular network on the surface of the placenta. Unfortunately, this is a difficult problem due to limited image quality and high variability in the position and orientation of the placenta and the low visibility of vessel in most modalities of medical imaging.

The proposed project will advance the state of the art of placenta and vascular detection for ultrasound (US) images and contribute significantly to an active project in collaboration with our clinical partners from BCNatal | Barcelona Center for Maternal Fetal and Neonatal Medicine Hospital Clínic and Hospital Sant Joan de Déu, Universitat de Barcelona.

1.1. Twin-to-twin transfusion syndrome

TTTS is a disease that occurs only in monozygotic monochorionic pregnancies (both twins sharing a single placenta), where one twin, the donor, transfuses blood to the other twin, the recipient (TTTS can also happen in pregnancies with more than two fetuses). This syndrome shows significantly fewer vascular anastomoses (connections between blood vessels) than non-pathological twin pregnancies. This constitutes a risk for both twins who can suffer from neurologic, cardiac and pulmonary sequelae.

The rate of monozygotic pregnancies worldwide is about 0.35%, where there is a 70% chance of it being monochorionic. From these, up to a 25% can manifest TTTS.

During pregnancy, the donor twin suffers from hypovolemia (low blood volume) and the recipient suffers from hypervolemia. In cases with diamniotic placentation (each twin having its own individual amniotic sac), blood volume differences lead to oligohydramnios (deficiency of amniotic fluid) in the donor, that can develop into anemia, and polyhydramnios (excess of amniotic fluid) in the recipient. In both cases, the brain can suffer from oxygen shortage and result in fetal brain ischemia (the donor is more susceptible). Both twins are at risk of developing hydrops fetalis (apparition of edema in
at least two fetal compartments) and suffer from critical cardiac issues (e. g., heart failure or hypertension). Moreover, the birth is at risk of being premature, highly worsening the prognosis of the babies the earlier they are born.

1.1.1. Current treatments

Leaving the disease untreated usually results in an early natural abortion or in premature pregnancies where fetuses need to stay a long time in an incubator. The treatments increase the gestation period leading to a healthier development and better prognosis. It has mortality rates up to 90% if not treated, and the survivors having a high probability of developing neurological sequelae. A factor that makes this syndrome so critical is that it reduces the gestation period, and none of the twins can develop properly. There are some treatments that are currently used and can improve the outcome, by extending the gestation period, which let the fetuses grow well. Quintero et al (1999) proposed a staging classification of the syndrome with prognostic significance. The application of a treatment is selected in function of which stage the syndrome is at. For low stages the excess of amniotic fluid around the recipient twin can be drained through amnioreduction, under ultrasonographic guidance, to improve again the circulation of blood, because the pressure on the placental surface is reduced, and situations of anemia or plethora are avoided. This procedure may need to be performed many times during the pregnancy.

The most advanced procedure currently being used is fetoscopic laser photocoagulation. It is used mainly in medium to advanced stages. The method consists in coagulating the anastomoses connecting the fetuses’ vascular systems through the placenta. The anastomoses coagulated are usually found between both umbilical cords and there is only one vessel visible (if a vessel connects the umbilical cord to one of the fetuses, there is another vessel, one for the inlet and the other for the outlet flows). It is a more aggressive procedure than amnioreduction but it shows better survival results; in early stages of the disease amnioreduction remains a good alternative. Fetoscopic laser coagulation shows an overall survival rate of 75% with a 85% probability of one twin surviving, being the donor twin at a higher risk of death. Even though there is an improvement in the prognosis, there have been found a few cases where neurological impairment occurs during infancy, therefore a follow up has to be done on all babies born after this procedure.
is performe. An improved method of laser coagulation, called the Solomon technique, is used to dissect the whole vascular equator, including small anastomoses that might not be visualized. This technique has shown to reduce recurrence of TTTS and lower the probability of postoperative fetal morbidity.

There is a method used only in very advanced stages where one of the fetuses is presumed moribund and it is endangering the health of the other fetus. In this situation the umbilical cord of the dying twin is ligated and the blood flow between twins is interrupted. It is risky to leave the dying fetus’s umbilical cord not ligated since it increases the chance that the surviving fetus develops brain ischemia, leading to neurological sequelae or even death. Umbilical cord ligation is a safe and succeeding procedure to perform but only when one of the fetus is moribund, because it interrupts the flow to that twin leading to death.

1.2. Mapping of the placenta vascular network

A critical point for fetal interventions is to plan properly the incision. It is vital not to interfere with the fetuses and/or the placenta, and to stay the minimum possible time inside the uterus. Surgeries longer than 30 minutes highly increase the risk of an abortion. The blood vessels needed to be dissected are observed previously to the surgery using ultrasound techniques. It is important to coagulate all the vessels, as its partial removal might lead to recurrent TTTS and increase fetal death rates and morbidity rates. Therefore, it is of utmost importance to preoperatively know where these vessels are

![Figure 1. Schematic illustration of endoscopic fetal surgery for twin-to-twin transfusion syndrome (from Luks, 2009)](image-url)
placed and plan the surgery in order to maximize the anastomoses coagulated while reducing the intervention time. Using imaging techniques preoperatively to observe the anatomical structures of interest will help to plan the incision point of the fetoscope and decide the path to the anastomoses that will be coagulated.

The procedure to obtain this vascular mapping is complex, and merges many imaging techniques and image processing methods. The pipeline consists on first acquiring some images of the placenta containing its anatomy and vascularization. Then these regions get segmented and registered on the same coordinate frame to finally fuse their information on a single image output.

1.2.1. Image acquisition techniques

There are many imaging techniques for fetal imaging. Ultrasound and magnetic resonance (MR) are already used to diagnose pathologies in pregnancies. In one hand, US have been long used in prenatal imaging and it is an inexpensive procedure fast enough to acquire motion free 3D images which show the anatomical structure. However, it shows weak tissue gradient making it difficult to detect element boundaries. On the other hand, in magnetic resonance images (MRI) the gradient between anatomical regions is much higher, but lacks the speed of US and is a much more expensive procedure. Thus, sometimes US is used in first instance and, in case of doubt of the diagnosis, MRI is later used to corroborate or discard the findings from US. Ultrasound techniques based on the Doppler effect (US Doppler) allow the visualization of body fluids (e.g. blood). Other techniques, that use X-rays such as CT, even though they can be used with a contrast agent for imaging of the villous vascular tree with a very clear visualization, are not usually used because of the side effects of irradiation on the fetuses, thus endangering the children life. However, in some critical situations it is worth taking this risk since the mother or the fetuses could die otherwise.

1.2.2. Segmentation

Placenta segmentation has been for long a hard task. A major problem appears due to high inter-subject variability of placenta shape, size, position and orientation. The
lack of tissue gradient between uterine tissue and placenta, as well as other effects such as hypo-echoic placental lakes, make the placenta appearance heterogeneous hence adding more factors that compromise the segmentation. Moreover, bones and other anatomies of the fetus or the mother can create shadows and blur the boundaries or fully hide them, being posterior placenta disposition more susceptible. For this reason, automatic methods rarely work, although Alansary et. al. obtained a Dice score of 71.95±19.79% for healthy fetuses from magnetic resonance imaging (MRI), using a convolutional neural network (CNN) as a classifier with a conditional random field (CRF) for refinement 29.

Interactive segmentation has been widely used to overcome these issues. These methods are initialized by manually seeding the foreground and the background of the region to be segmented. A good approach would be the one with minimal user interaction that obtains an accurate segmentation. Different algorithms can benefit from this prior to approach the segmentation problem from different perspectives.

Non-supervised semiautomatic tasks focus more on a geometrical approach. The virtual organ computer-aided analysis (VOCAL , General Electric Healthcare, Milwaukee, WI, USA) has already been used for US placenta segmentation. It is a semiautomatic method used to obtain organ segmentations. It is based on the geometric interpolation of 2D seeded contours defined at constant intervals of a rotation angle around a predefined axis. The fact that the interpolation is purely geometric can lead to incorrectly segment irregular shapes 32,33. The random walker (RW) segmentation algorithm is another semi-automatic approach 34. It evaluates the image as a graph and calculates the probability that a random walker particle placed at an unseeded node will reach a seeded node in function of spatial and intensity (or color) distances. This RW particle is expected to move to one of its neighbors depending on the weights that join them. The highest probability that an unseeded node reaches a labeled node will determine the final segmentation of the region. It has shown an improvement in segmentation accuracy over VOCAL 35. The RW algorithm benefits from a tradeoff between geometric shape of the prior seeding and RW particle motion in function of image intensity gradients. It has also been extended to the lazy random walk algorithm, which behaves similarly but there is a small probability that the RW particle stays at its current node 36. It shows a slight improvement in boundary adherence compared the classic RW approach.
Other methods are based on midlevel discriminative image representations, such as graph cuts (GC), which detect some low level features to segment the images. Based on the max-flow min-cut theorem, it minimizes a cost function defined over the graph setting each pixel as a node and weighting the edges between nodes proportionally to their pixel similarity, and it accurately detects weak boundaries \(^{37}\). Other similar approaches can also be used for image segmentation but have not been tested for placenta yet. Simple linear iterative clustering (SLIC), generates superpixels efficiently by initializing \(k\) clusters and assigning each pixel to the closest neighbors measuring the weighted Euclidean distance combining color and spatial proximity \(^{38}\). Newer superpixel methods are being developed which seem to slightly improve segmentation accuracy and/or computation efficiency. Linear spectral clustering (LSC), developed by Chen et al. (2017) \(^{39}\) combines normalized cuts and weighted k-means (a combination of GC and SLIC), improving segmentation boundary accuracy compared to SLIC, at the expense of slightly higher computational time. Other methods similar methods, e.g. quick shift or turbopixel, are available but show a less accurate boundary adherence and require of more computation time, even for small 2D images \(^{38}\).

Some learning tasks, such as Markov random fields (MRF) or CRF have been used within interactive methods showing fine segmentation accuracy in US \(^{30,31}\). Wang et al. (2016) segmented the placenta from MRI using a prior initialization on a single slice of the 3D image and then propagating the labels across neighboring slices. The algorithms detect and extract some features from the pixels that were indicated with the prior, and then they use this information to classify the remaining pixels of the image to finally obtain a segmentation. These algorithms can benefit from a prior segmentation to better classify the pixels, allowing to combine different algorithms; e.g. as done by Alansary et al. (2016) using a CNN to obtain a prior and refining the segmentation with a CRF. The other explained segmentation methods can be explored to be used as a prior segmentation for these learning tasks.

Vessel segmentation is approached differently due to its natural elongated shape. In Doppler US images vessels appear in hyperintense colors. Otsu’s method can successfully segment the vessels in very neat images, but get corrupted in noisy ones. Hessian filters have been used to detect elongated objects (e.g., vessels) as well as RW with an intensity
threshold prior. Morphological filters are later applied to link unconnected regions due to noise and obtain a fine segmentation \(40-42\).

1.3. Justification

The current prognosis of fetuses diagnosed with TTTS has improved over the years but there still are some risks for the fetuses. Our clinician collaborating partners from BCNatal are currently using photocoagulation therapy to treat TTTS. They are the most experienced team in Spain performing this surgery. This procedure has been shown to improve the prognosis in any stage, but there are still some drawbacks in it that can lead to adverse effects. A reason to this is that some anastomoses that should be coagulated are missed during the intervention and left uncoagulated. The incision point is planned in situ and the intervention is guided through the camera of the fetoscope and live US imaging. No pre-operative planning is done, thus there are still some risk factors that can endanger the procedure; the most critical one is the intervention duration, which the faster it is the higher the success rate also is.

Many considerations can be taken into account pre-operatively that can help during intervention. The most useful planning would be to have segmentations of the placenta and the whole vascular network, differentiating the vessels needed to be coagulated to those that supply nutrients to the fetuses. This is currently an impossible task, no medical imaging techniques have such precise resolution and no image processing algorithms can detect these anatomies so accurately. Despite, there are alternative, and realistic, approaches that can be used as planning tools. Some anatomical landmarks can be of use to focus on the region of the intervention. Actually, the anastomoses that will be coagulated are located between the two fetuses’s umbilical cords. Locating these two landmarks, in addition to fine segmenting placenta surface (between the cords) will lead to knowing pre-operatively the region of interest for the photocoagulation procedure. The incision point can then be planned to navigate faster through that zone, and, therefore, reduce the intervention time. In addition, this can be used as a benchmark where further improvements can be added, such as the segmentation of vessels between the cords or just the exact location of the anastomoses.
Despite the pre-operative planning, there can also be intra-operatory aid. Once all the anastomoses are coagulated, the surgeon navigates again the region to check no anastomose was left uncoagulated. Thus, a system that tracks the fetoscope and records its path, can be used to travel the way back and easily check if all the anastomoses are coagulated. Again, this can reduce the intervention duration.

Compared to the currently in situ planning and no navigation aid, any change that can contribute to reduce the intervention duration, even baby-step changes, can improve the intervention prognosis.

1.4. Goal

In this project we aim to create a surgery planning and tracking tool that can be used to improve fetoscopic laser coagulation prognosis and, later, that can be extended to other surgeries. We are working on different branches with different approaches to get to this objective. One branch focuses on different image processing techniques for US and MR images to obtain placenta and vessel segmentations, umbilical cord insertion point and other anatomical markers. Another branch investigates those aspects for surgical guidance, such as intra-operative imaging and tracking systems. This exact study is embedded within the image processing techniques branch, aiming to develop a method for placenta and vessel localization and segmentation in US images, and fusion of these different labels in a single frame. A graphic user interface (GUI) is developed including segmentation and label fusion methods.

We propose different approaches for the viable methods for prior initialization and fusion. We implement and optimize the RW algorithm which has been already used for US placenta segmentation\textsuperscript{35}, and that can also be used for prior fusion. We use this algorithm to validate the accuracy of placenta segmentation and to set as an initial method to integrate in the medical application. With this benchmark done, many more methods can be easily integrated. The final goal is to merge different optimized methods, for both prior initialization and the fusion of placenta and vessel, and include them in the application to let the clinician chose the best approach for each patient.
2. Methods

2.1. Integration of methods

The inter-patient anatomy deviation of the placenta makes it difficult for a single algorithm to successfully segment different data sets. We have seen that there are automatic and semi-automatic methods. In one hand, the former performs fast without user interaction but still lacks boundary adherence. On the other hand, the latter do obtain better boundary adherence especially in the seeded slices, but lose accuracy when getting farther from the labeled slice \(^{30}\). Moreover, semi-automatic methods show inter-user dependability, which is undesired. That is why we propose the idea of merging state-of-the-art methods in a single interface. We can use semi-automatic methods to segment the placenta and obtain good boundary adherence, but for that we need a prior which is accurate for most of the slices (or all of them). We propose to use different methods to initialize a prior for each of the various regions of interest (in this case placenta and vessel), combine them and let another method compute the final segmentation and fusion of labels, given the information provided by the priors (Figure 2). This way we can reach the requirements to segment different anatomies with unique properties (e.g. shape). A good approach would be to use a first method that detects the overall shape of the region and a second method that adheres to the boundaries. Of course, any prior can be corrected by the user before running the fusion algorithm. The process for the surgery planning

![Figure 2. Scheme of the application.](image)
ends with a curved planar reformation (CPR) for better visualization of surface vessels in the placenta. We have implemented a GUI using the Medical Imaging Interaction Toolkit (MITK) which right now it only includes the RW segmentation algorithm. This implementation has been used for different tasks, as shown in 2.2.1.

The whole fusion workflow is illustrated in Figure 3. In this project, however, only placenta and vessel segmentation has been developed, from different images and from the same images and then fusing both labels (single and multi-label segmentation).

2.2. Segmentation

The data used on the project is composed of 31 datasets with 3D US and 3D Doppler HD-flow ultrasonography acquisitions obtained from our partners in BCNatal. The equipment used as a Voluson E10 from GE Healthcare, with probe eM6C (curved electronic matrix 4D). The settings for the 3D US were a preset routine Th37/Tr40 with speckle reduction (V-SRI 4) and 90° of acquisition angle. The Doppler HD-flow US were acquired with pulse repetition frequency (PRF) 0.6 to 1.8 Hz, with a wall motion filter (WMF), Gn 0, and PWR 94%. The subjects fall within a large range of gestational age, from 16 to 36 weeks. The placenta position is divided in 15 anterior placentas and 16 posterior placentas. Four of them have been discarded because of poor tissue gradient due to the presence of shadows; the four with posterior placenta disposition.
The US images we are working with are represented in a gray scale (from 0 to 255, 1 byte precision). Doppler images are originally in color to represent flows, but since we are not interested in the blood flow, just the presence of vessels, images are turned into gray scale where vessel appear hyperintense. On one hand, in US images, anything outside the conic field of view of the ultrasound is pure black. Liquids (e.g. blood or amniotic fluid) poorly reflect the sound waves, thus appearing in very dark gray. Soft tissues have different reflection capabilities and get represented in a wide range of the gray scale, but not too dark nor too light. Harder tissues, such as bone, reflect most of the sound wave, thus appearing in bright grays. Moreover, these are prone to produce shadows, since when reflecting most of the wave, the tissue beneath it receives waves low in power and their reflection does not get much representation in the image (appearing dark and with low tissue gradient). On the other hand, Doppler US images behave the same way with the difference that moving fluids get represented in blue or red, depending on the flow direction (let us remind that we convert these images to gray scale, then the red and blue colors are turned into bright gray). In addition, these images usually show a poorer tissue gradient compared to US.

*Figure 4* left shows an example of US image. There are three different anatomies that can be observed: the fetus (green line), the placenta in anterior position (yellow line), and the uterus and other regions belonging to the mother (what remains non-circled in mid-range gray). What appears dark but falls within the conic field of view is amniotic fluid. *Figure 4* right shows an example of Doppler US image. We observe the placenta in posterior position (yellow line) and a shadow (red line). Vessels are also easily appreciated as they
are shown in hyperintense gray. This image is zoomed in and everything on top of the placenta is amniotic fluid. Also, on the left of the image some amniotic fluid is present, but the boundary with the placenta is not clear because of the shadow.

2.2.1. Random walker segmentation method

The random walker algorithm (RW) for placenta segmentation has been previously tested in other studies and has shown good results. Doctors initially seed the image with foreground (placenta) and background (e.g., uterus walls, fetus). A fast manual seeding can be done that highly increases the accuracy of the segmentation. It is also resistant to weak boundaries (as long as the two different regions have been indicated during the seeding). This method can be an asset on US image segmentation. It is resistant to weak boundaries and, if seeded properly, it can segment regions not visible because of a shadow in the image. Moreover, this method can segment single or multiple structures as long as they are initialized with different labels. The different pathways this algorithm has been used for is shown in Figure 5.

A random walker algorithm has been implemented for placenta segmentation. This method is the one proposed by L. Grady (2006)34. The RW algorithm is a graph based method which converts the image into an undirected graph with each pixel assigned to a node $v$ and each edge $e$ connecting two neighboring nodes. Given a prior seeding, the

![Figure 5. Applications of the random walker algorithm in this project](image-url)
algorithm computes the probability that a random walker particle placed at an unseeded node will reach a seeded node. Since the prior contains at least two regions each with a unique label, the probability is computed for all different labels, and the node is assigned to the label with the highest probability (Figure 6).

Each edge is assigned a weight, computed using a Gaussian weighting function based on the intensity gradient between two neighboring nodes:

$$w_{ij} = e^{-\beta(g_i - g_j)^2}$$

where $g_i$ and $g_j$ represent the image intensity at that given node (pixel). $\beta$ is the only user set parameter (inversely proportional to the standard deviation), limiting the ease of the random walker propagation. A small constant ($10^{-6}$) is added to the weight to avoid zero weighting, thus disconnecting that edge.

The graph is seeded by manually indicating some nodes corresponding to different regions, labeling each region with a different value. Then the probability that a random walker starting at an unseeded node will reach a seeded node is computed. If a random walker starting at node $v_i$ has a higher probability to reach a node with seed $s_k$ than any

![Figure 6. Representation of the approach of the segmentation with three different labels for a 2D image.](image)
other node with a different seed, node $v_i$ will be assigned to region $S_k$. The solution can be approached by solving the combinatorial Dirichlet problem. Let us define the Dirichlet integral:

$$D[u] = \frac{1}{2} \int_\Omega |\nabla u|^2 \, d\Omega$$

for a field $u$ over a region $\Omega$. The problem is aimed to find a harmonic function with the boundary conditions of the function being the seeded nodes. The harmonic function that satisfies the boundary conditions minimizes the Dirichlet integral.

Let us define the vector $x$ which denotes the probability of the nodes to belong to label $s$ and reorder it so the marked (seeded) nodes ($x_M$) appear first and the unmarked ($x_U$) appear second.

$$x = \begin{bmatrix} x_M \\ x_U \end{bmatrix}$$

Let us define the $N \times N$ combinatorial Laplacian matrix, with $N$ being the total number of nodes.

$$L_{ij} = \begin{cases} 
  d_i, & \text{if } i = j \\
  -w_{ij}, & \text{if } i \text{ and } j \text{ are neighbors} \\
  0, & \text{otherwise}
\end{cases}$$

$$d_i = \sum w_{ij}, \quad \forall \text{ edges incident on } v_i$$

Then this matrix is reordered to match vector $x$ (marked nodes first and unmarked nodes after). It gets partitioned in four submatrices, $L_M$, $L_U$, $B$ and $B^T$ ($^T$ denoting it is the transpose matrix).

$$L = \begin{bmatrix} L_M & B \\ B^T & L_U \end{bmatrix}$$

where $L_U$ corresponds to the matrix having in its rows and columns the unseeded nodes, $L_M$ to the matrix having in its rows and columns the seeded nodes, and $B^T$ being the matrix with unseeded nodes at rows and seeded nodes at columns.

The energy function defined by the combinatorial Dirichlet integral can be formulated as:

$$D[x] = \frac{1}{2} x^T L x$$
Since $L$ is positive semidefinite, the only critical points of $D[u]$ will be minima. The previous equation can be rearranged in the same way as $x$ and $L$.

$$D[x_U] = \frac{1}{2} [x_M^T x_U^T] \begin{bmatrix} L_M & B \\ B^T & L_U \end{bmatrix} [x_M]$$

Differentiating $D[x_U]$ with respect $x_U$ and finding the critical points leads to the solution of the problem:

$$L_U x_u = -B^T x_M$$

Setting $x_M$ to have a value of 1 to those nodes with seed $s_k$ and 0 to the nodes with other seeds, and solving for $x_U$ will give the probabilities for unmarked nodes to reach (to belong to) region $S_k$. This equation can be rewritten to include all the different labels:

$$L_U X = -B^T M$$

$X$ being the matrix with the probabilities of unlabeled nodes (rows) to reach each label (column), and $M$ the matrix having marked nodes at rows and different labels at columns.

$$M_{is} = \begin{cases} 
1 & \text{, if node } i \text{ has label } s \\
0 & \text{, if node } i \text{ does not have label } s 
\end{cases}$$

The probabilities of each node adds to unity, therefore there is only need to solve for $S-1$ linear systems, as one can be obtained by subtraction. For two regions (i.e. two different labels), if the probability of an unseeded node $v_i$ to reach a seeded node with label $s_k$ is higher than 0.5 that node will be assigned to region $S_k$. For more than two regions, the probability for a given unmarked node to belong to each region is computed. Then, the maximum of these probabilities is found and the label corresponding to that probability is assigned to the current unmarked node.

### 2.2.1.1. Algorithm implementation

The algorithm has been implemented in C++, using Image segmentation and registration ToolKit (ITK) and Eigen3 libraries. To reduce the size of the linear system, hence accelerate the computation, the images have been cropped to the size containing seeds (outside these boundaries the image is considered as background). $L_U$ is a highly sparse diagonally dominant, symmetric, and positive semidefinite matrix, with a sparsity of $1 -$
Since \( N \) is of the order tenths or hundreds of millions, the sparsity is almost close to 1. For these reasons, Eigen’s biconjugate gradient stabilized method (BiCGStab) with diagonal preconditioner (or Jacobi preconditioner) has been used as the solver of the problem. BiCGStab offers a faster convergence than other methods (e.g., conjugate gradient method) and works fine on symmetric positive semidefinite matrices. The diagonal preconditioner helps the solver obtaining an even faster solution for diagonally dominant matrices \( |L_{U_{ii}}| \geq \sum_{j \neq i} |L_{U_{ij}}|, \forall \ i \). It has also been compiled with OpenMP to multithreaded the solver. To accelerate the system preparation and reduce memory usage, matrices \( L_U \) and \( B^T \) have been directly built, instead of building the whole \( L \) matrix and then partition it. The whole algorithm workflow is illustrated in Figure 7.

**Figure 7.** Workflow of the implemented algorithm. Postprocessing step is only performed for placenta segmentation.
2.2.1.2. GPU optimization

$L$ is a highly sparse matrix, containing only 7 non-zero elements per row (for a 3D image), the weights to the 6 neighboring nodes and the degree of each node in the diagonal. $L_U$ is slightly smaller, since most of the nodes are unmarked and will contain up to 7 non-zero elements per row. Given the size of the images, even though they are cropped to the region of the placenta, the number of unmarked nodes (i.e. the matrix dimension) is between 10 to 100 million (depending on the image). These properties make this system as a strong candidate for GPU solving.

Let us define the system in a more conventional way $Ax = b$, where $A = L_U$ and $b = -B^TM$. Left hand side $A$ is a $N_U \times N_U$ matrix, $N_U$ being the quantity of unmarked nodes, and right hand side $b$ is a $N_U \times S$ matrix, $S$ being the different number of labels. Assuming we are solving the system for to regions ($S-1 = 2-1 = 1$), $b$ will be a column vector with size $N_U$. On the left hand, $L_U$ is stored in compressed sparse row format (CSR) for faster solving (sparse solvers are optimized for CSR matrices). This format stores three arrays: i) all the values of the non-zero elements, ii) all the column indexes of each non-zero value, and iii) the accumulated quantity of non-zero elements per row. In total the dimensions of these three arrays for the current situation is up to $7N_U$ for the first two arrays and $N_U+1$ for the last array. On the right hand, $b$ can be stored in dense and sparse formats; given that most of the elements are non-zero $b$ is stored as a dense matrix, containing $N_U$ elements. $x$ has the same properties as $b$. We do not need double precision accuracy, so single precision floating point is used to save memory. The total byte size that will need to be stored is:

$$4 \cdot (7N_U + 7N_U + (N_U + 1) + N_U + N_U) = 4 \cdot (17N_U + 1) \approx 68N_U \text{ Bytes}$$

In the case of using the Jacobi preconditioner, which requires an extra storage of a CSR $N_U \times N_U$ sparse matrix with $N_U$ non-zero elements, this number rises to $80N_U$ bytes. To this value, we still have to reserve some memory for the inner computations of the solver. We said that the images for this project have between 10 to 100 million elements, so the approximate size without preconditioner is approximately between 0.6 GB and 6.6 GB, and with preconditioner between 0.8 GB and 8 GB.
We implemented a BiCGStab solving method with CUBLAS and CUSPARSE (CUDA libraries). We used a laptop card NVidia gt 740m with 1.6 GB of available memory. To fulfill GPU memory restrictions, the dimensions of the matrices is reduced to occupy less than 1 GB of memory. Therefore, $N_U \approx 10\cdot10^6$ so no memory shortages occur. Only one case has been used because it is the only one that satisfies this restriction. After successful GPU implementation, we have had also the opportunity to test its efficiency on a high end NVidia Maxwell Titan X card.

Moreover, Matlab’s BiCGStab solver has been used on similar artificial smaller problems to observe the effects of preconditioning on CPU and GPU, and it has also been compared with Eigen and CUDA implementations.

### 2.2.2. Placenta segmentation

Placenta has been segmented from US images. Using the RW algorithm leaves two main unknowns in its procedure that affect the outcome, which are the parameter $\beta$ and the prior used.

For a Gaussian distribution $\beta$ is also defined as the inverse of twice the variance:

$$\beta = \frac{1}{2\sigma^2} \rightarrow \sigma = \sqrt{\frac{1}{2\beta}}$$

Smaller values of $\sigma$ (i.e. higher $\beta$) will enhance intensity similarities over the image whereas lower values will augment the importance of the geometry of the prior labeling. Moreover, the latter will assign very low weights to all intensities gradients but 0, so the random walker particle will likely stay on the same node, thus making it more difficult to find a solution and needing to reduce the stopping criterion for the linear solver, which

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Product name</th>
<th>Cores/CUDA cores</th>
<th>Clock speed</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU</strong></td>
<td>Intel i5-4570</td>
<td>4</td>
<td>3.2GHz</td>
<td>32GB</td>
</tr>
<tr>
<td><strong>GPU</strong></td>
<td>NVidia gt 740m</td>
<td>384</td>
<td>900MHz</td>
<td>2GB</td>
</tr>
<tr>
<td></td>
<td>gtx Titan X</td>
<td>3072</td>
<td>1075MHz</td>
<td>12GB</td>
</tr>
</tbody>
</table>
highly increases the computation time. We want to find a trade-off point where we give enough weight to similar neighbors but highly penalize heterogeneities, because we want to reduce user dependency and find a good boundary adherence. The US images have a pixel data type of 1 byte (from 0 to 255). For intensity gradients higher than $3\sigma$, the weights are at least 100 times smaller compared to two pixels having the same intensity, therefore the propagation of a random walker through that edge is highly unlikely. We consider intensity gradients out of the $3\sigma$ range to have a poorly significant weight. Looking at the images, many weak boundaries have an intensity gradient around 5. We then set $3\sigma = 5$. In this situation we highly restrict heterogeneities, as only 2% (5 out of 255) of intensity gradients are assigned to have a non-poorly significant weight. Thus, $\beta = 0.18$.

As initially stated in the beginning of this section, segmentation results can vary in function of the prior used. A labeling protocol has been established to check the segmentation accuracy of this method (similar to the one proposed by Stevenson et. al (2015) $^{35}$) (Figure 8). It consists on labeling a set of axial slices through the volume. The slice previous to the initial slice with placenta on it is seeded only as background (if the volume ends within the image boundaries). The next tenth slice (towards the placenta) is also seeded. The same procedure is performed at the other extreme of the volume. The remaining volume is labeled every twentieth slice, except if there are sudden changes in

Figure 8. Placenta prior initialization. Foreground prior (placenta) in red and background prior in green.
shape where the labeling will be done in no further than 10 slices, to indicate the two dominant shapes. The boundaries of the placenta do not need to be drawn accurately, but the foreground must include only placenta and the background must include all the elements that are not placenta and are in contact with it. In case there is a small intensity gradient between placenta and another tissue, accurately indicating the boundaries will improve segmentation accuracy. If there are parts of the placenta shadowed, these parts can be approximately manually seeded and it will be geometrically segmented by the RW algorithm.

Once the segmentation is obtained a connected components filter is applied to leave only the biggest component (i.e., with highest quantity of voxels) which corresponds to the placenta, and erase miscomputed regions by RW algorithm (usually in the bounding box boundaries).

2.2.3. Vascular network segmentation

Blood vessels in US Doppler-flow acquisitions are shown in blue (arteries) and red (veins). When transforming the video to a gray volumetric image, the color of the vessels is lost, but they appear hyperintense. The segmentation can be performed by applying Otsu’s method. The problem using this segmentation method is that noise can be classified as vessel, and many boundaries can be confused with other bodies. The RW segmentation algorithm, already used for the segmentation of the placenta, can also be used to segment the vascular network using an appropriate prior. Since this approach does not only take into account the intensity of the image, but also the proximity to seeded pixels, and the vessels are connected, the algorithm should obtain an accurate segmentation. A fast prior initialization for both the foreground (vessels) and background is done using Otsu’s method with four regions, assigning the brightest one as vessel and the two darkest regions as background. The third region is left unseeded and the RW algorithm computes its segmentation (Figure 9 top). In case that some vessels do not appear hyperintense, or the segmentation leaves unconnected vessels, the prior can be manually corrected for a successful outcome (Figure 9 bottom pictures).

Vessels show a higher intensity range compared to placenta. Otsu’s prior might not connect all hyperintense vessels, therefore easier propagation over more different
Intensity gradients must be allowed. This range is increased using visual inspection to $3\sigma=15$, thus $\beta = 0.02$. Moreover, lower $\beta$ also accelerates the computation of the probabilities.

2.2.4. Placenta and vessel fusion

As mentioned in section 2.2.1, the RW algorithm can be used for segmentation of multiple labels. Until now, we only used it to differentiate two regions: placenta from background or vessels from background. In the project we actually want to fuse in a single image
both placenta and vessel. In Doppler-flow US not only the vessels are present, but also
the placenta (with less quality than in US). From these images we can segment both
regions at the same time. In order to do so we need three priors: one for the placenta, one
for the vessels and another one for the background.

The placenta and background priors are computed with the RW algorithm. The seeding
is done following the same slice protocol as in placenta segmentation, but in this case we
add a third label for the vessels. Vessels are over-labeled, meaning that also some pixels
outside the vessels walls are seeded. The RW segmentation is computed but only placenta
and background segmentations are written. Then the vessel prior is computed using
Otsu’s method and then computing the RW segmentation. At this stage, both priors are
checked for some miss-segmented regions and they are finally merged. Since the manual
prior for placenta had over-labeled vessels, there should not be any overlapping of the
two priors, but in case there is, the overlapped region is left unseeded. Finally, the fused
prior (Figure 10) is ready and the RW algorithm computes the segmentation of the three
regions. This process is illustrated in Figure 11.

As there is placenta present in the image, we use again $\beta = 0.18$, since placenta boundary
gradient is the smallest (compared to vessel) and we want to avoid geometric solutions.
We compare individually the placenta and vessel segmentations with the ones obtained individually. Vessels are compared using the Dice coefficient since the images are in both cases the Doppler US. However, placenta was not acquired before in the Doppler US, so if we wanted to compute the Dice coefficient we would also need to register the two segmentations and acquaint for registration errors. For this reason, only segmentation volumes are compared.

$$\text{Dice coefficient} = \frac{2|V_1 \cap V_2|}{|V_1| + |V_2|}$$

$$\text{Volume similarity} = 1 - \frac{|V_1 - V_2|}{|V_1| + |V_2|}$$

$V_1$ and $V_2$ are the number of elements in each segmentation.

Figure 11. Workflow of fusion of labels from US Doppler images and segmentation of placenta and vessels from the same image.
3. Results

3.1. Segmentation

Placenta segmentation results from US show the difficulty of this task. Total placental volume is hardly detected and segmentation gets underestimated. The regions with high intensity gradients get segmented accurately, parts such as the separation with the amniotic fluid (which in US appears with a very low intensity). In other regions where these gradients are much smaller, the algorithm has trouble to accurately adhere to the boundaries. These regions are usually the junction with the uterus and the outermost sides of the placenta. In Figure 12 the middle image shows the probability map of the
segmentation. The probability gradients are much higher at the separation with amniotic fluid than in other regions. This gets reflected in

*Figure 12* and *Figure 13* where different placenta segmentations are shown. It can be easily appreciated how the overall shape of the organ is segmented, but also the lack of boundary adherence in some sides. Despite this low adherence in the mentioned regions, we observe a precise weak boundary adherence in the case where the fetus is in contact with the placenta (yellow line in *Figure 12* top picture).

Shadows are another conflicting point. Partial to total occlusions can occur, mainly in posterior placenta. In case of partial occlusion, the seeding can usually be performed correctly and therefore no accuracy is lost upon segmentation. In case of total occlusion, the seeding has to be done intuitively in the region and the segmentation will be purely geometric since there will be no intensity gradients. Moreover, if the shadows appear in the middle of the placenta, it is possible to visually interpret the shape during the manual labeling, but this task gets harder when the shadow is at an end of the placenta. These shadows are mostly found in cases with posterior placenta disposition, thus getting finer segmentations in anterior dispositions. *Figure 15* shows two different segmentations where the original US images had shadows. On the image on the right the segmentation was not much affected since only partial occlusion occurred and the labeling could be done following the anatomy. However, on the left image, there is a total occlusion of one side of the placenta, being extremely difficult to label properly that region. In addition, no intensity gradients are present leading to miscomputed segmentations because of purely geometrical interpretation of the region (green line).

![Figures 12 and 13 showing placenta segmentations](image)

*Figure 13. Segmentations of the placenta in red, showing the accurate boundary adherence in the region in contact with the amniotic fluid. Lack of boundary adherence in other boundaries (yellow).*
Better results are accomplished in vessel segmentation. As long as vessels have been initialized using Otsu’s prior, the RW algorithm detects the remaining unlabeled vessels from the corresponding regions, but any isolated vessel that has no seeding will never be detected. The umbilical cord is well segmented as well as the main vessels that are born from it. Other thick vessels are also accurately detected, however thin vessels that were not initialized are not segmented. The Doppler US also detects some noise from other
sources of waves besides the one sent by the probe. This noise, if detected in the prior, is then present on the segmentations. Therefore, these segmentations are very sensitive to noise, since no filtering step has been applied. Moreover, flows that are parallel to the probe are not detected either by the Doppler US. Thus, they do not appear on the segmentation, leaving unconnected vessels. Luckily, as mentioned in section 2.2.3, this can be easily corrected manually, and a more accurate segmentation is obtained. Figure 14 shows the Otsu initialization for vessel segmentation and the volumetric rendering of the segmentation. It shows the prior correction of a dark vessel (yellow line, due to parallel flow with respect to the US probe), and how it gets properly segmented after this correction. Moreover, some noise is also present on prior initialization and is appreciated in the bottom pictures as small unconnected particles.

We get a combination of these two results in the case where we segment both anatomies from a single image in a multi-label segmentation. This situation combines the techniques applied for placenta and vessel segmentation. What can be observed is that vessel segmentation is very similar to when segmenting the vessels alone (Figure 16), obtaining a Dice coefficient of 94.18±1.14%. In all the cases the volume is higher in the fused segmentation, because the vessel prior for the fused segmentation is the output of the vessel segmentation alone (Figure 14). Therefore, in the fused approach, vessels can expand an extra time, occupying more space. Again, thick vessels and umbilical cord are detected. In 78% of cases, umbilical cord insertion to placenta is detected and segmented. Visually, it can easily be appreciated were this occurs (Figure 16 yellow line). This detection is independent of placenta location; in three cases the placenta was in anterior position and in the other three in anterior position. Placenta segmentation, however, tends to be smaller, and underestimate even more the organ outermost sides and junction with
the uterus compared to the segmentations where only placenta was obtained. The volume similarity parameter returns a value of 79.64±8.17%, in all the cases being higher the volume obtained in placenta segmentation alone (Figure 13 and Figure 15). In Doppler images intensity gradients of non-flowing anatomies tends to be smaller, this making more difficult placenta segmentation. In addition, in very noisy Doppler acquisitions, the RW algorithm segments most of it as vessels. This turns into an increase of false positive for vessel segmentation and an increase in false negative for placenta segmentation.

3.2. Segmentation time

The workflow of the segmentation process is divided mainly into two steps, prior initialization and segmentation computation. The first part takes an average of 409±100s (6.83±1.68min). Compared to the average computation time of 182±112.31s and 112±50.82s (computing the linear system on one and four threads, respectively), it is clear that the first part takes most of the total segmentation time.

There is also a direct linear dependency between both times and the size of the image. The images are cropped to a bounding box which fits the placenta, thus each case will have a unique dimension. Figure 17 shows how the longer the placenta is (in the axial axis) the longer it takes to label the image. The images are labeled every twentieth slice, so the more slices a placenta occupies, the more slices will need to be manually seeded. However, in this case there is another factor which increases the initialization time, which are the cases with shadows or poor boundary gradient. In addition, each image has only

Table 2. Measures of CPU solving times using 1 and 4 threads in all different placenta segmentation cases.

<table>
<thead>
<tr>
<th></th>
<th>Matrix dimension (voxels x10⁶)</th>
<th>Iterations to convergence</th>
<th>Solver time (s)</th>
<th>Time/dimension (s / (voxels x10⁶))</th>
<th>Speedup (4 vs. 1 threads)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 thread</td>
<td>4 threads</td>
<td>1 thread</td>
</tr>
<tr>
<td>Mean</td>
<td>49.90</td>
<td>120.26</td>
<td>182.27</td>
<td>112.31</td>
<td>3.64</td>
</tr>
<tr>
<td>Std. dev.</td>
<td>22.99</td>
<td>16.53</td>
<td>83.46</td>
<td>50.82</td>
<td>0.50</td>
</tr>
<tr>
<td>Max.</td>
<td>101.71</td>
<td>173</td>
<td>342.76</td>
<td>217.57</td>
<td>5.05</td>
</tr>
<tr>
<td>Min.</td>
<td>14.33</td>
<td>103</td>
<td>39.98</td>
<td>26.92</td>
<td>2.79</td>
</tr>
</tbody>
</table>
Figure 17. Prior initialization times in function of placental length in axial direction.

Figure 18. Solving time of the linear system for different images using one and four threads in function of left hand side matrix dimension.
been labeled once, so other factors could influence this period, which cannot be observed nor analyzed with the available data.

*Figure 18* shows how the higher the dimension of the left hand side matrix of the linear system is, the longer it takes to solve it, and *Table 2* shows the average times for these computations. In this table the average solving times are just to show how long the segmentation of the dataset we are working with takes, but we have to keep in mind that this time is dependent on the system size. Therefore, it is more interesting to look at the solving time in function of the left hand matrix size. Then we obtain the real speed up of the algorithm when using a multithreaded solver with four threads compared to using a single core, which is of 1.62 times in average. The remaining time correspond to preprocessing, preparing the linear system and post processing steps, which take only 10% of the computation time or 3% of the total time (prior initialization plus computation times), taking an average of 19.07±6.44s.

### 3.3. GPU optimization

We mentioned that the characteristics of the problem, a large sparse one, is a good candidate for parallelization to obtain a substantial speed up. First we used Matlab, which already includes a CPU and GPU BiCGStab solver, to observe the behavior of the parallelized method, both with and without preconditioning. Using the Jacobi preconditioner, we observe that the number of iterations gets reduced, but it does not get reflected in total computation time in Matlab’s implementations. In the CPU approach there is not much difference in time (slightly higher with preconditioning), but in GPU computation the total amount of time drastically increases with left hand matrix size (*Figure 20*). The preconditioner is used twice in this method at each iteration, in an inverse matrix product computation. In CPU it is rather fast, but it is observed that matrix inversion is a very expensive procedure for the low-end graphics card employed, so expensive that even though the number of iterations can be reduced a 25%, the total computation time gets multiplied by two. However, we do not observe such a difference time penalty when using a preconditioned system CUDA implementation (*Table 3* and *Figure 21*). On the contrary, preconditioned systems converge faster both in iterations and time. These two implementations instead of inverting the preconditioner twice per
iteration they invert it once before starting the iterative process, hence reducing the complexity of the following computations. In addition, non-preconditioned CUDA implementation has a similar value of iterations compared to Matlab’s implementation, but takes half the time to obtain a solution. The best result is obtained when using the preconditioned CUDA approach. In this case the number of iterations gets reduced seven to nine times, reducing the total time needed with respect Matlab’s up to twenty times (Figure 19).

Let us remind that only one full case could fit in the GPU’s memory and only it was studied. The GPU implementation runned on the low end card gt 740m, according to Table 3, is slower than the CPU approach. It can be seen that the iteration period for both cases is very similar, but CUDA implementation requires of five to ten times more iterations to converge, making it a slower option. Since. This difference in performance

![Graphs](image)

*Figure 20. Time and iterations to convergence in function of left hand side dimension using Matlab’s BiCGStab on different small sized problems with preconditioning (red) and without preconditioning (blue). Left column: CPU computation. Right column: GPU computation.*
between the two methods gets also reflected on the segmentation result. Both methods converge to an error equal or smaller than $10^{-3}$, but with slight differences, obtaining a Dice coefficient between them of 94.96%. The fastest results, though, are accomplished using the high end Titan X card. Even though it asks for more iterations to converge, it reduces so much the iteration period that in the end it shows similar speed results as CPU if not preconditioned, but accelerates the computation three to four times when

Figure 21. Time and iterations to convergence in function of left hand side dimension using CUDA implemented BiCGStab solver on different small sized problems (same as in Figure 13) with preconditioning (red) and without preconditioning (blue), solved using the gt740m card.

Figure 19. Speed up of CUDA BiCGStab implementation compared to Matlab’s GPU (red) and CPU (blue) approaches. Left: Speed up without preconditioning. Right: Speed up with preconditioning.
preconditioned. In this case the Dice coefficient with the CPU result is 94.97%. Both GPU segmentations are very similar, showing a Dice score between them of 99.99%.

There is an extra time that must be taken into account in GPU implementation, which is the memory allocation and data copy time to the graphics card. Results found show that for the gt 740m it takes 0.35±0.08s and for the Titan X 0.43±0.07 (0.2% and 4.2%, respectively, of total computation times in preconditioned systems). Although the linear system is the smallest of all the dataset, and the larger ones will take more time, this time barely affects the total computation time.

Table 3. Comparison of average solving times between BiCGStab implementations using Eigen3 (CPU) and CUDA (GPU) libraries over 30 runs over the same problem.

<table>
<thead>
<tr>
<th>Processor</th>
<th>Time (s)</th>
<th>Iterations</th>
<th>Average iteration period (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Without preconditioning</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>1 thread</td>
<td>41.82±0.84</td>
<td>153</td>
</tr>
<tr>
<td></td>
<td>4 threads</td>
<td>27.44±0.53</td>
<td></td>
</tr>
<tr>
<td>GPU</td>
<td>gt 740m</td>
<td>378.24±0.13</td>
<td>1482</td>
</tr>
<tr>
<td></td>
<td>Titan X</td>
<td>32.95±0.25</td>
<td>2315</td>
</tr>
<tr>
<td><strong>With preconditioning</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>1 thread</td>
<td>27.23±0.29</td>
<td>108</td>
</tr>
<tr>
<td></td>
<td>4 threads</td>
<td>19.08±0.23</td>
<td></td>
</tr>
<tr>
<td>GPU</td>
<td>gt 740m</td>
<td>162.06±0.03</td>
<td>554</td>
</tr>
<tr>
<td></td>
<td>Titan X</td>
<td>10.44±0.19</td>
<td>594</td>
</tr>
</tbody>
</table>
4. Discussion

We have seen that the accuracy of the segmentations varies depending on the side of the placenta, being clearly more accurate where intensity gradients are higher. This is the result of using $3\sigma = 5$. We wanted to reduce the user effect on the prior labeling and get the boundaries more accurately. Stevenson et al. (2015)\textsuperscript{35} used a higher deviation leading to more geometric results, therefore their approach of the RW algorithm approximates better the overall volume but might not detect properly sudden changes in shape. In addition, they used B-mode Us images, which can show different tissue gradient compared to the 3D US images we were using. In this case, we were expecting to get a precise boundary detection but we only get it on the separation with the amniotic fluid, where the intensity gradient is much higher. However, in the current study this does not imply immediately a wrong result. The vessels that will be coagulated in the surgery are all in the placenta surface, so as long as the surface (the one in contact with the amniotic fluid) gets accurately segmented, it does not matter if other sides (e.g. the separation with the uterus) are not well computed. In some cases the fetus is in contact with the placenta surface, but we have also seen that this boundary gets fine segmented

\textit{Figure 12}.

We have also found that the presence of shadows makes it more difficult to obtain a fine segmentation. These are more present in posterior placenta. In these cases, the fetus is placed between the probe and the placenta, so there is more tissue in between capable of reflecting most of the incoming wave, mainly the fetus’s bones. Therefore, posterior placenta segmentation has been found to be a harder task than anterior placenta.

The vessels that must be segmented are the ones in the placenta surface, as they are the ones that are potentially being coagulated. The best results would be to segment only all the superficial vessels. Not all of them get detected by the Doppler US, hence the RW algorithm cannot segment them. Therefore, no full planning of the surgery can be done using this technique because the final decision of which anastomoses to coagulate will still be done intra-operatively (only the coagulation of those anastomoses present in the US can be planned). Moreover, there are some segmented thick inner vessels of the placenta that must not be coagulated. If these go out to the surface it can be confusing to know whether to coagulate it or not during the intervention. The effects of recurrent TTTS
are worse than coagulating a vessel which should not be coagulated, as in this last case the blood can find other paths to the fetus. Therefore, it is better to coagulate a vessel if it is not sure whether it is an anastomose connecting the fetuses or not. However, the best case scenario for the surgery is the one that coagulates all the anastomoses, but not any other vessel. So having also the segmentation of the vessels that must not be coagulated will rise the probability of getting closer to this goal.

We observed that in 78% of the images the insertion point of the cords was detected. This detection was independent of placenta location. Some discontinuities on the cord occur due to parallel flow of the blood, hence appearing black in the Doppler acquisitions. In some cases, this happened near the placenta surface, around the area were the cord insertion was, thus avoiding the detection of this point. Other cases where the cord was not observed was due to noise. Some acquisitions were very noisy, and the algorithm could not differentiate between vessel and noise. If this noise appeared near the cord’s insertion, again, it could not be detected. This anatomical point is of utmost important for fetoscopic laser coagulation since the anastomoses are located between the two cords’ insertion points. Knowing their position can help planning the intervention since the doctors can know pre-operatively the region were the vessels more likely to be coagulated are located. Then they can plan the incision point of the fetoscope, and the region to cover during the intervention, which will potentially reduce de intervention time and reduce its risks.

The similarity in vessel segmentation between single and multi-label segmentation (placenta and vessel fusion) is because the vessel prior initialization is exactly the same (using Otsu’s method plus RW) and the images used are also the same (Doppler US). However, this similarity does not occur with placenta. Intensity gradients in Doppler images for regions other than vessels are lower, thus outermost sides and junction with the organ get undersegmented as well. In addition, vessels take some volume that otherwise is assigned to placenta. Finally, noisy Doppler acquisitions get a high number of false negative regarding placenta segmentation, thus reducing its segmentation volume. These are the reason why we find volume differences between the two cases.

Prior initialization takes 409±100s using a seeding protocol which requires of major annotations on seven to fifteen slices. It takes longer than in Stevenson et al. (2015)
were they recorded initialization times of $175 \pm 33$ s. Two reason can arise from these differences: i) The sizes of the images were different, thus we had more slices to initialize, and ii) the seeding was performed by no clinician, taking longer to decide the boundaries of the placenta. This average time of almost 7 minutes is clearly too long and needs to be reduced. According to the doctors we are collaborating with, we should implement a semiautomatic method which requires of just weak annotations on three or four slices.

Stevenson et al. (2015) used, among others, the BiCGStab solver with Jacobi preconditioner from Eigen library and obtained a mean solving time of 43.60s and a standard deviation of 15.23s. Compared to the times obtained in this project of $182.27 \pm 83.46$ s (single core) and $112 \pm 50.82$ s (four cores), their times are much faster. The CPU used in both cases are similar in computation power, thus the difference cannot be because of it. We have seen that there is a linear dependency between the matrix $L_U$ and the solving time. Therefore, given that the same solving implementation has been used in both studies, the only possibility for these times difference must be that the images used in this case are larger than the ones used by Stevenson et al. (2015).

We observed a speed up in segmentation time when using a preconditioned system, though not in Matlab’s implementation, where performing a sparse inverse matrix product twice per iteration is so expensive that it actually increases the computation time. Luckily, this is not the case with the implemented methods using Eigen and CUDA libraries, where preconditioning accelerates the segmentation. We were expecting GPU to be faster, which is the case for the high end graphics card. However, we still find some shortages in the parallelized implementation, as it requires of more iterations to converge. The reason is that Eigen is a highly optimized linear algebra library, but the GPU implementation is not optimized. Therefore, optimizing CUDA implementation so it converges in similar number of iterations as CPU approach (some differences could still arise due to hardware and software precision) is expected to reduce even more the total computation time (about ten times, the average speed up found in iteration periods).

A strange behavior is found in Eigen CPU solver implementation. The preconditioned approach has a faster iteration period compared to the non-preconditioned one (Table 3). This should be the other way around, as in BiCGStab preconditioned systems two extra matrix multiplications have to be computed at each iteration. We hypothesize that this
conduct is because each iteration requires slightly more time to compute that the previous one. Since non-preconditioned system needs of more iterations to converge, the described situation can occur. Eigen library does not give the option to return the time to compute each iteration, thus the average iteration time is computed by dividing the total solving time over the number of iterations. Consequently, we cannot be sure if this accumulation of computing time after each iteration exists.

Another strange behavior is found in the GPU implementation. We expected some differences on converging number of iterations (hardware capabilities are different, since they are different models from different generations) as found in the preconditioned system needing 554 and 594 iterations (1.07 increase). However, this does not happen in the non-preconditioned approach with 1482 and 2315 iterations (1.56 increase). Moreover, the card requiring of more iterations is the newer and high end Titan X. We would expect the Titan X to not require more iterations than the gt 740m because its CUDA computing capability is much more optimized. In the end, the preconditioned GPU approach on a high end card shows the fastest segmentation results.
5. Conclusions

In this project we implemented a method for placenta and vessel localization and segmentation from US images, the RW algorithm. It can be used for single or multiple label segmentation and has been integrated inside a GUI application for medical use. This application has been designed to include many segmentation and fusion methods. For the moment it only includes the random walker algorithm, but future work can be focused on other automatic and semiautomatic placenta segmentation methods. It will improve current methods and merge them inside a same platform that can be used to specifically select the best approach for each patient.

A good approach to improve the workflow of the whole fusion method is to combine automatic and semiautomatic methods. Two possible ways to initialize the process are weak manual annotations and automatic methods. Adapting the CNN approach from Alansary et al. (2016) to US images can lead to a fast initial placenta segmentation, which can be latter used as a prior for semiautomatic methods. Another way could be to use algorithms that assign to each pixel a probability to belong to each region (such as RW or GC). We could bias the algorithm to down estimate the volume of the segmentation, by only assigning labels to those pixels with probabilities higher than a threshold. This way we should get a segmentation which approximates the total region volume and that can be used as a prior (and combined with other priors, e.g. as explained in 2.2.4) for a semiautomatic method.

We validated the practicality of this RW algorithm for placenta segmentation, but there is still room for improvement. Boundaries with high tissue gradients get properly segmented, but US is a modality with poor differentiation of structures, therefore many boundaries of the placenta are not detected accurately resulting into an undersegmentation of the total volume. No ground truth dataset was available, thus we do not have quantitative data to evaluate the accuracy of this method yet. Further research can be directed into this pathway. Despite, all the results have been corroborated by our clinician partners. We have seen that this same method can also be used for vessel segmentation, but that it is sensitive to prior initialization. Otsu’s method proves to detect most of the vessels for initialization, but it is also very noise sensitive. In general, mostly the thickest vessels (umbilical cord and main ramifications) were segmented. Different methods can
be explored to improve both of these segmentations (single and multiple) to include them in the application.

We obtained a segmentation of the vessels of the placenta that can be used for TTTS surgery planning. Despite no full placenta segmentation nor vascular network are obtained, we get a good boundary adherence in placenta surface and umbilical cord insertion in most of the cases. As previously explained, the anastomoses are located in the placenta surface between the two cords. Getting the segmentation of these anatomical regions provides information that can be used to plan the incision point. Then all of the region with anastomoses can be covered reducing the probability of encountering parts of the fetus that could difficult the vision and the mobility of the fetoscope. These segmentations can still be improved to include the whole vascular network (or at least all the anastomoses) which would help in the planning and reduce the intervention time. In addition, Doppler acquisitions were in many cases very noisy, which made it difficult to detect the cord’s insertion point, and also corrupted placenta segmentation. Filtering should be investigated to reduce noisy images before segmentation processing. Still not ideal, but what we have obtained can be used as a benchmark where further improvements can be added.

The implementation of the algorithm has been improved to reduce memory usage and to accelerate the computation. Two branches were developed, one optimized for CPU and another one for GPU. Both of them have been proved useful showing similar results, but the latter computing three to four times faster. However, this second one has the potential to become even faster, as the critical part of the algorithm (solving the linear system) does not perform as optimized as the CPU approach (requiring of five times more iterations to converge). By optimizing it, we approximate that we the speed up of this method can outperform by ten times the CPU implementation. Moreover, other preconditioner matrices can be studied to adapt better to the size, shape and sparsity of the current problem.
6. Bibliography


31. Martín-Fernández, M. & Alberola-López, C. An approach for contour detection


40. Yan, J., Liu, H. & Cui, Y. A random walk-based method for segmentation of intravascular ultrasound images. in *Biomedical Applications in Molecular, Structural, and Functional Imaging* (eds. Molthen, R. C. & Weaver, J. B.) 903825 (International Society for Optics and Photonics, 2014). doi:10.1117/12.2043478


