MASTER

B-spline surface modelling with coverage path planning towards autonomous vitrectomy

Coerver, Tim

Award date:
2018

Eindhoven University of Technology
Eindhoven, The Netherlands

TU/e Technische Universiteit
Eindhoven
University of Technology

Systems & Control
Department of Mechanical Engineering
Control Systems Technology

B-Spline surface modelling with coverage path planning towards autonomous vitrectomy

Tim Coerver

0773865

CST 2018.108

Y.G.M. Douven
M.J.G. van de Molengraft

November 2018

TU/e Technische Universiteit
Eindhoven
University of Technology

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

23-11-2018

Name

L.T. Coerver

ID-number

0773865

Signature

Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU

B-Spline Surface Modelling with Coverage Path Planning towards Autonomous Vitrectomy

L.T. Coerver, Y.G.M. Douven and M.J.G. van de Molengraft
Department of Mechanical Engineering
Eindhoven University of Technology

***Abstract*—Robotic systems have successfully been used to assist surgeons in various operations. The use of such systems increases instrument accuracy and reduces tremors of the surgeons hands. One of the most occurring eye operations is the vitrectomy. The difficulty of this operation lies in the required high precision movements for prolonged periods of time. Using a robotic system to perform this operation could result in fewer complications and shorter operation times. This paper explores a method for a robotic system to autonomously perform a vitrectomy. To this end, a B-Spline surface model of the eye is created from distance measurements taken from the tip of the robot's instrument. Based on this model a path is planned that allows for the removal of the vitreous without damaging the eye. This path is then executed and adjusted based on model updates from new distance measurements.**

NOMENCLATURE

$\boldsymbol{V}$	Control point vector or matrix.
ϕ	Rotation around the x-axis.
ψ	Rotation around the y-axis.
σ	Standard deviation.
A	Working space within the eye.
$B_{i,k}$	k^{th} order B-Spline at knot interval $[t_i,\ t_{i+k}]$.
C^n	Continuous up to the n^{th} derivative.
C_l	Cone that originates fron the scleral entry and extends around the lens, lying tangent to its surface.
k	Order of the B-Spline (curve or surface).
N	Number of control points.
r_c	Cleaning radius of the vitrectome.
z	Depth of the vitrectome into the eye.

I. INTRODUCTION

IN recent years many breakthroughs have been achieved on the topic of surgical robots [1] [2]. Most of these robotic systems assist the surgeon during surgery by scaling down their movements, filtering out tremor, giving haptic feedback or even guiding the surgery via a vision system. As of yet the most advanced systems, such as the Da Vinci Surgical System, can prevent the surgeon from making harmful movements but cannot move themselves. This is done by analysing the operation area and setting virtual constraints on the robot's movements [3] [4]. Currently no system exists that can perform an operation completely autonomous.

One of the most occurring eye operations is the vitrectomy. During a vitrectomy, the vitreous is removed from inside the eye and replaced with a balanced saline solution. Many reasons exists why a person could require a vitrectomy. The most frequent ones are: retinal detachment, epiretinal membranes, macular holes, blood in the vitreous (due to trauma or diabetic retinopathy) or floaters. Performing a vitrectomy requires very high precision, preferably 1 to 10 μm [2]. The human hand can achieve a precision of about 100 μm for durations of up to 20 seconds. Besides this lack of precision, the surgeon is also limited by the stereoscopic visual input through a microscope and almost non-existent sense of touch. This makes a vitrectomy very difficult to perform.

This paper explores a method for the Preceyes Surgical System (PSS) to autonomously perform a vitrectomy. The system autonomously finds its way through the eye based on distance measurements taken via an optical coherence tomography (OCT) sensor. The OCT-sensor scans the retina and part of the tissue behind it. Each scan is then analysed to determine the location of the retina with a precision of 20 μm. Based on the distance measurements and encoder data of the robot, a model is created of the eye. This model is then used to create a path that will cover the eye to remove the vitreous.

To create the model, a surface has to be fitted on a set of unorganized points. Much research has been done on this problem since it arises in many applications such as computer graphics, 3D scanning and medical imagery. Methods used to approximate surfaces from point clouds range from Coulomb Potentials [5] to B-Splines [6], Radial Basis Functions [7] [8] and even Self-Organizing Neural Networks [9]. However, most of these methods assume the dataset to be entirely available a priori. In this paper, data points are becoming available over time and the surface fitting method used should be able to handle this in an efficient manner. This changes the problem to one of recursive fitting. Solutions to this problem have been found using a Kalman Filter to determine the influence of new measurements on the fitting of a B-Spline curve [10] [11]. This paper will build upon the results from [10] and [11] to recursively fit a B-Spline surface.

To remove the vitreous a path needs to be planned that covers the model. The go to approach for 3D volume coverage path planning (VCCP) is to decompose the volume into a series of 2D layers [12]. A paper on coverage path planning with application to robotic intracerebral haemorrhage evacuation [12] deals with 3D VCPP without a 2D decomposition by minimizing the task space distance travelled. This seems like a promising strategy since minimizing surgery duration and movements inside the eye are important to reduce change of complications. However, besides minimizing the distance travelled, it is also important to minimize contact with the vitreous due to its connection to the retina. Moving it will pull on the retina with the possibility of it tearing. To minimize the chance of this happening, interaction should be kept to a minimum. A second reason to minimize interaction is to prevent the vitreous from moving through the eye. The robot has as of yet no way to detect the vitreous and clears it by simply covering the entire eye. If vitreous would move to an already covered area it would remain after the operation and possibly result in complications. Removing the vitreous layer by layer minimizes interaction to only the tip of the vitrectome. To realize a layer by layer path, a cost function would need to be used in the 3D VCCP algorithm that can accommodate this. The 2D layer decomposition strategy inherently works in this way and because of this reason the coverage path planning used in this paper will build upon this strategy.

This paper is constructed in the following way. Chapter II analyses the procedure of a vitrectomy and describes how it could be performed by an autonomous robotic system. Chapter III describes how measurements of the retina are used to create a model using a B-Spline surface and Kalman Filter. Chapter IV gives the coverage path planning that uses the model to plan a path through the eye that updates online. Finally, chapter V gives the conclusions and future work.

II. The Autonomous Vitrectomy

The vitrectomy consists of many steps, normally all executed by the surgeon. This section will examine the steps that are taken over by the robotic system. From these steps and their translation to autonomy, requirements can be taken that will determine what the capabilities of the model and coverage path planning algorithm should be.

A. The procedure

This paper will focus on removing the vitreous of a patient with a fully attached retina. It is important for the retina to be attached since the PSS has currently no method of detecting and avoiding the retina when it is floating in the eye. The starting state of the robot is assumed to be as follows: any peripherals necessary for the operation have been placed at their position, the system is calibrated and the tip of the vitrectome is positioned just inside of the eye. The operation consists of various steps [2], three of which are of importance here. These are the removal of vitreous around the incision site, the removal of the vitreous core, and the shaving of vitreous close to the retina. Together these steps take approximately 15 to 17 minutes for a specialised surgeon to perform. At the end of these steps, the vitreous is largely removed at which point the surgeon takes over to perform the final steps. One of these final steps is the detachment of the vitreous from he optic nerve. This is a very critical step to the successful removal of the vitreous. During this step there is a higher chance to cause retinal tears due to the stronger connection between the vitreous and the optic nerve. The surgeon gradually detaches this connection bit by bit. What makes this step so difficult to perform is the difficulty of determining if the connection is fully severed. Due to the required involved motions and inability of the robot to detect the connection, this step is better performed by the surgeon. The final step is to check if any vitreous remains by staining it with a special dye. This is also done by the surgeon, as the robot has as of yet no way of detecting the stained vitreous.

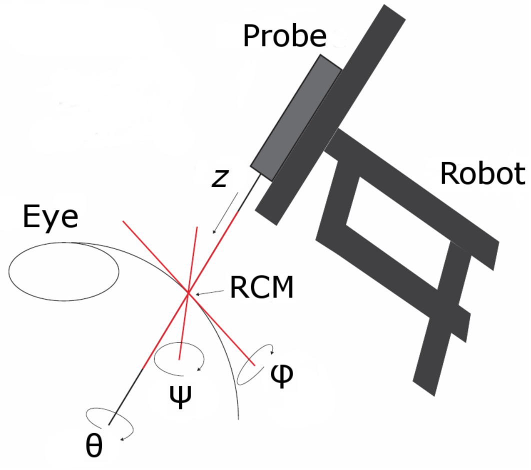


Figure 1: Adapted from [13]. Shows the robot entering the eye at the scleral entry. The scleral entry acts as the remote center of motion around which the robot can rotate over the angles ϕ, ψ and θ. The depth of the into the eye is given by z. Attached to the probe are the vitrectome and the distance sensor.

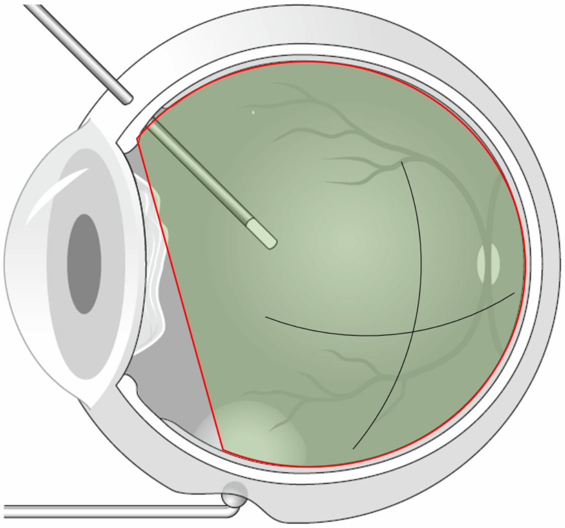


Figure 2: Visualization of the vitrectome entering the eye, adapted from [14]. Based on where the robot can position itself without touching the retina or the lens, a working area is determined and indicated in green. The two black lines inside the working area are the rough trajectories of the measurements shown in Figure 3b and 3c.

B. The working area inside the eye

Figure 2 gives a schematic overview of the eye and the robot entering it. The vitrectome is inserted at the scleral entry. In the case of an eye with a natural lens this point lies 4 mm away from the border between the sclera and the cornea [14]. During the operation, measurements of the retina along with the average dimensions of the human eye [15] are used to create a working area that prevents interaction with anything other than the vitreous. This working area is marked in green in Figure 2 and is restricted by three elements. These are: the retinal surface, the posterior section of the lens and the trochar used to insert the vitrectome into the eye. The retinal surface is indicated in the figure as the circular part of the red line. The limitations of the lens on the working area are indicated by the straight section of the red line. This represents a cone shaped surface that extends from the scleral entry and lies tangent to the surface of the posterior section of the lens. The lens' position cannot be measured by the robot. To ensure it is not touched, this virtual cone is placed around it. The position, shape and size of the cone are based on the dimensions from [15]. Finally the protrusion length of the trocar into the eye limits the vitrectome from clearing the area close to the entry point. Depending on the type, the total length of the trocar varies from 4-6 mm [16]. The thickness of the sclera at the scleral entry approximates to 0.50 ± 0.11 mm [17]. This leaves a radius of at most 5.5 ± 11 mm around the entry point that cannot be reached.

C. Safety requirements

To guarantee the safety of the operation the model needs to represent the retina with a certain level of accuracy. An expert vitreoretinal surgeon stated that the instruments were moved as close as 0.5 mm to the retina. To guarantee safety, the accuracy of the model will need to be high enough to allow for these precise movements to fall within the confidence bounds of the model. In other words, every orientation the robot can take, as close as 0.5 mm from the retina, needs to fall inside the working area with a specified certainty. This required certainty is assumed to be 99.99%, though may change as required.

III. Modelling

The distance measurements taken from the retina can be used to create a model. This model should be able to represent the retina in such a way that allows for the safe execution of the surgery. This means that the model can guarantee movements as close as 0.5 mm to the retina to be safe. The model will be used to plan a path that will be executed by a robot. It is preferred that the model is smooth enough for the robot to follow the edges from as close as 0.5 mm without it having to make adjustments to the path to guarantee a smooth trajectory.

A. Structure of the retinal surface

During the procedure, measurements of the retina are taken using an OCT-probe. This small sensor is fitted on the tip of the vitrectome and creates a 1-dimensional depth scan of a point up to 3.7 millimetres in front of it. If the retina is within range, its position can be determined from the scan. This occurs at a rate of 700 Hz. The noise of these measurements has a Gaussian distribution with a standard deviation of approximately 20 μm.

Figure 3 shows in-vivo distance measurements taken from the retina of a human eye. The y-axis gives the distance between the scleral entry and the position of the retina. When taking a closer look at Figure 3 many details can be seen that the model will have to handle. Figure 3a shows a 19 second stationary measurement of a single point on the retina. It visualizes the influence of the patient's heartbeat and breathing on the retinal position. These repetitive movements cause a shift in the retinal position that with future work will be compensated for, resulting in measurements of a stationary retina. As of yet, the heartbeat and breathing are still included in the measurements and will therefore be assumed to be Gaussian noise. Band-passing the data in Figure 3a to filter out the heartbeat and breathing movements shows that these have a standard deviation of 7.2 and 5.9 μm respectively. Together with the standard deviation of the distance sensor the assumed Gaussian noise on the measurements becomes 22.1 μm.

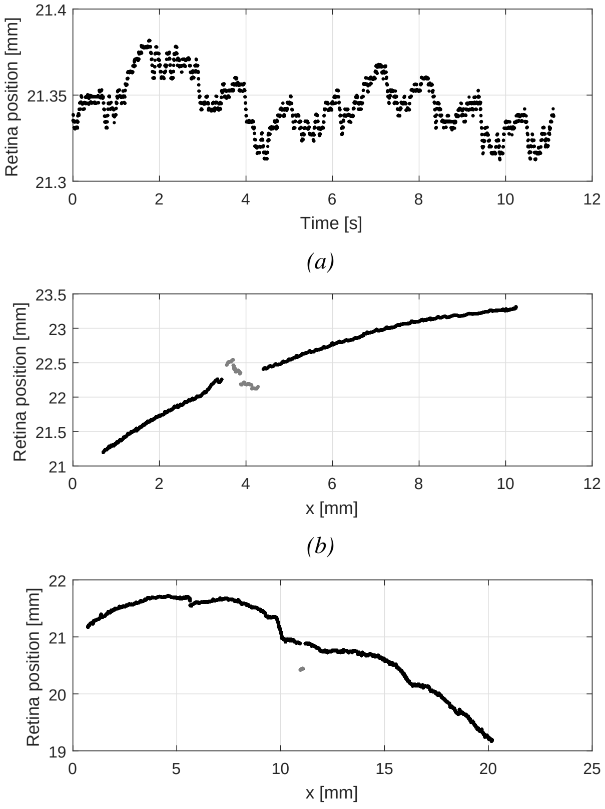
Figures 3b and 3c show measurements taken in a straight line across the retina. The black lines in Figure 2 shows the rough location of these measurements. This visualizes its typical structure and the expected shape that needs to be modelled. Video recording from the surgery shows some measurements to have been disturbed by movements of the patient. Occurrences of these movements can be seen in Figure 3b and 3c around $x = 4$ and $x = 11$ respectively. Since these disturbed measurements do not accurately represent the retinal surface, they have been grayed out and will not be used when testing the surface fitting.

In short, the most important disturbances in measurement data that need to be correctly handled by the model are: gaps between the available measurements caused by the retina not being in range, heartbeat, breathing, movements of the patient and measurement errors.

Besides disturbances and measurement artifacts that need to be excluded by the model, it is also important to know what the model will have to fit. In other words, what is the shape of the retina. According to P. Nogueira et al. (2011) the retina can be approximated as a sphere with an average radius of 10.74 mm. However, since this is an approximation, the model should be able to handle deviations from this shape. These include varying radii within an eye, resulting in more of an ellipsoid. Analysing the measurements from Figures 3a and 3b shows a maximum deviation of 0.41 mm from their respective average radii. The fovea is another element that causes the eye's shape to deviate from a sphere. According to [18], the vertical thickness of the fovea measured between the inner- and external limiting membrane varies from 0.15 mm at the center to 0.40 mm at the rim. This results in a difference of 0.25 mm that occurs over a distance of approximately 1.2-1.5 mm. The model should be able to describe these small deviations from the otherwise almost spherical shape of the eye.

B. Modelling the retinal surface

The model of the retina should be able to describe its almost spherical form and the indentations caused by the fovea at an unknown location. To do this a B-Spline surface will be



(a)

(b)

(c)

Figure 3: Shows three different parts of the measurements from a patients retina. (a) is an 11 second measurement of a single point on the retina. It clearly shows the patients heartbeat as a series of peaks with an amplitude less than $50\ \mu m$. (b) and (c) show measurements in which the robot moved in a straight line across the retina. The distance on the y-axis is measured from the scleral entry to the retina. Video recording from the surgery shows some measurements to have been disturbed by movements of the patient. These measurements have been marked gray.

used. A B-Spline curve or surface is a piece-wise polynomial function [19] [20] [21]. One of its strengths is that coefficient changes only have local effects [22]. Inversely, this means that when new measurements are incorporated into the model the affected coefficients only consist of a small local group based on the order k of the B-Spline. This is a very useful property to keep the computational cost of model updates low and allows for effective scaling. It also has the added benefit of the measurements not influencing areas of which no information is available. The eye is organic after all, there are no guarantees that unmeasured areas equal the average human anatomy.

A B-Spline curve of order k is a piece-wise polynomial function of order $k-1$ that uses N control points to shape N number of k^{th} order B-Splines into a polynomial [19]. The power of this method is that multiple low order polynomials (B-Splines) are used to create a single high order curve. Instead of having to re-evaluate all of the curve's parameters with every measurement, only the local k control points of one polynomial piece have to be adjusted. This is less computationally expensive. The B-Spline curve is defined over a set of knots that form the connections between the separate pieces. These knots are positioned along the curve in a non-decreasing sequence $\boldsymbol{t} := (t_i)$ [19]. A first order B-Spline with $k = 1$ has the following definition

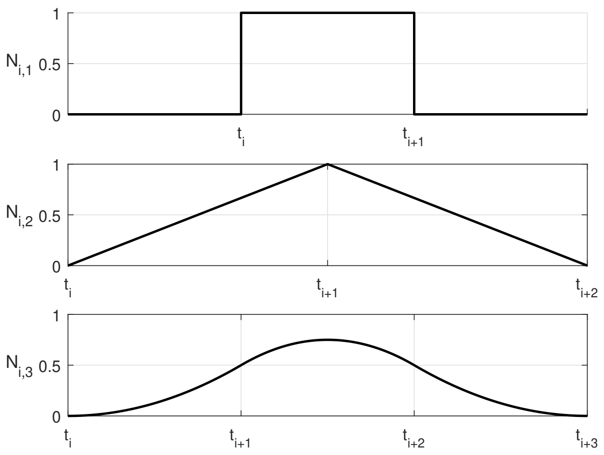


Figure 4: B-Splines of orders 1, 2 and 3.

$$B_{i,1}(t) = \begin{cases} 1, & \text{if } t_i \leq t < t_{i+1} \\ 0, & \text{otherwise} \end{cases} . \tag{1}$$

It has the value one between the two knots t_i and t_{i+1} and zero elsewhere. A B-Spline is defined over a series of knots and outside of this interval has a value of zero. The amount of knots over which the B-Spline is defined depends on its order k. A k^{th} order B-Spline has $k+1$ knots [20]. One of their core strengths is that higher order B-Splines can be created by combining lower order ones through recurrence in the following way [21]:

$$B_{i,k}(t) = \frac{(t - t_i)B_{i,k-1}(t)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - t)B_{i+1,k-1}(t)}{t_{i+k} - t_{i+1}} \tag{2}$$

Figure 4 shows B-Splines of orders 1, 2 and 3 respectively. The first-order B-Spline is defined between two knots and consists of a constant piece. The second-order B-Spline is defined between three knots and consists of two linear pieces. The third order B-Spline is defined over four knots and consists of three quadratic pieces that form a smooth piecewise quadratic function with connections at the knots. Using (2), a B-Spline of any order k can be created in $k-1$ recursive steps.

To create a B-Spline curve, multiple B-Splines can be added together. Each B-Spline adds to a part of the curve over a subset of the knot vector. The individual B-Splines that make up this curve can then be scaled to mould the curve into the desired shape. This is shown in Figure 5, where five B-Splines, shown in the lower sub-plot, are scaled and summed to create the curves shown in upper sub-plot. Creating a B-Spline curve from summing multiple scaled B-Splines can be done by using

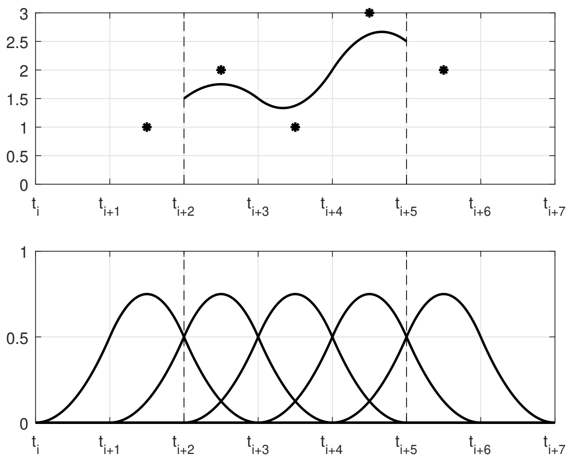


Figure 5: Third order B-Splines scaled with the control point values [1 2 1 3 2]. When summed, this creates the curve seen in the upper sub-plot. Since the B-Splines are third order, the B-Spline curve is only defined at points where three B-Splines are non-zero. This is valid for $t \in [t_{i+2}, t_{i+5}]$.

(3), where t is a position along the knot vector and M is the total number of knots equalling $N + k$ [10].

$$y(t) = \sum_{i=1}^{M} V(t_i) B_{i,k}(t) \tag{3}$$

Determining a point on the B-Spline curve using (3) would require running through all individual B-Spline functions, calculating their value at t and scaling these values with their respective control point $V(t_i)$. Doing this using (2) can be rather computationally expensive due to its recursive nature. To improve the computational cost, the equation can be rewritten to a matrix representation that allows the evaluation of a point on the B-Spline curve by a single calculation. This requires the knots to be equally spaced. To show how this is done, an example using three 3^{rd} order B-Splines is given in Figure 6. In this case, the knot interval $[t_i, t_{i+1}]$ has one section of each of the three 3^{rd} order B-Splines that contribute to it. To determine the value of a point t along the B-Spline curve, its respective knot interval $[t_i, t_{i+1}]$ has to be selected and the point has to be translated to a normalized value inside that knot interval, meaning $t_{norm} = \frac{t - t_i}{t_{i+1} - t_i}$. The functions of the three B-Spline sections in the knot interval are then evaluated for t_{norm} and summed. To find these functions, the B-Splines are determined in (4) using (2). The bold parts of each B-Spline in (4) are the quadratic functions that together create the part of the B-Spline curve that lies between the knots t_i and t_{i+1}.

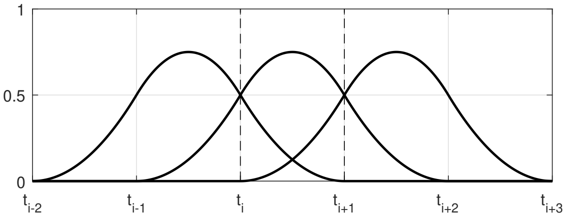


Figure 6: Three third order B-Splines can create a fully defined B-Spline curve section between the knots t_i and t_{i+1}.

$$\begin{aligned}
B_{i-2,3} &= \frac{1}{2}[(t^2 + 4t + 4)B_{i-2,1}(t) \\
&\quad + (-2t^2 - 2t + 1)B_{i-1,1}(t) \\
&\quad + \boldsymbol{(t^2 - 2t + 1)B_{i,1}(t)}] \\
B_{i-1,3} &= \frac{1}{2}[(t^2 + 2t + 1)B_{i-1,1}(t) \\
&\quad + \boldsymbol{(-2t^2 + 2t + 1)B_{i,1}(t)} \\
&\quad + (t^2 - 4t + 4)B_{i+1,1}(t)] \\
B_{i,3} &= \frac{1}{2}[\boldsymbol{(t^2)B_{i,1}(t)} \\
&\quad + (-2t^2 + 6t - 3)B_{i+1,1}(t) \\
&\quad + (t^2 - 6t + 9)B_{i+2,1}(t)]
\end{aligned} \tag{4}$$

The bold parts of (4) can be translated to a matrix that calculates this section of the B-Spline curve for $t \in [0, 1]$. This results in (5) [19].

$$\boldsymbol{M}_2\boldsymbol{T}(t) = \frac{1}{2}\begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} t^2 \\ t \\ 1 \end{bmatrix} \tag{5}$$

For a 4^{th} order B-Spline curve the matrix becomes

$$\boldsymbol{M}_3\boldsymbol{T}(t) = \frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 0 & 4 \\ -3 & 3 & 3 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} \tag{6}$$

These matrices now give the ability to evaluate a point on the B-Spline curve by only inserting the desired position between two knots as $t \in [0, 1]$ and scaling the resulting B-Spline parts with their respective control points. This changes (3) to the new form

$$y_i(t) = \boldsymbol{V}_{i,k}\boldsymbol{M}_k\boldsymbol{T}(t) \tag{7}$$

With $\boldsymbol{V}$ being the $k + 1$ control points around knot interval i defined by $[V_{i-\lfloor\frac{k+1}{2}\rfloor}, \ldots, V_{i+\lceil\frac{k}{2}\rceil}]$.

Equation 7 gives a B-spline curve in 2D. To model a B-Spline surface in 3D an extra dimension is added in which the B-Splines are evaluated. The control points vector $V_{i,k}$ now becomes a $(k + 1) \times (k + 1)$ matrix consisting of all control points that influence a point on the surface. To evaluate the B-splines and their control points in the new dimension $\boldsymbol{T}(y)^\top$ and $\boldsymbol{M}_k^\top$ are used. This results in (8) [19]. Figure 7 gives a visual example of B-Splines evaluated in 3-dimensional space.

$$z(x, y) = \boldsymbol{T}(y)^\top\boldsymbol{M}_k^\top\boldsymbol{V}_{i,j,k}\boldsymbol{M}_k\boldsymbol{T}(x) \tag{8}$$

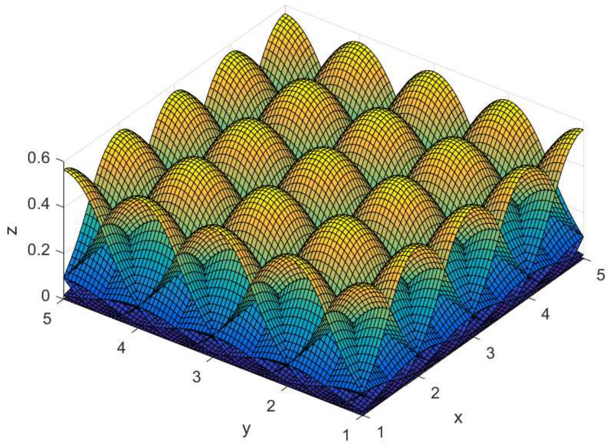


Figure 7: Shows third order B-Splines in 3D space. When scaled with control points and summed these basis functions will form a surface in the same way that was shown in Figure 5.

To allow the B-Spline surface to represent a sphere, a geometric transformation is used in which the three dimensions of the surface represent the angles ϕ and ψ and depth z_r of the robotic system from Figure 1. Using (9), the x, y and z coordinates of the sphere can be determined.

$$\begin{aligned} x &= z_r \cdot sin(\phi) \\ y &= z_r \cdot sin(\psi) \cdot cos(\phi) \\ z &= z_r \cdot cos(\psi) \cdot cos(\phi) \end{aligned} \tag{9}$$

Figure 8 shows a set of 225 control points that form a sphere using this method. The control points are uniformly spaced over ϕ and ψ.

C. B-Spline fitting using an Extended Kalman Filter

In order to fit the model to the distance measurements of the retina an Extended Kalman Filter (EKF) is used [10]. The use of an EKF allows also for the inclusion of movements of the eye into the position of the retina.

The state consists of all control points in the system and the input consists of the current orientation of the instrument given as in (10).

$$\hat{x}_k = \begin{bmatrix} V_{11} \\ V_{12} \\ \vdots \\ V_{mn} \end{bmatrix}, \; u_k = \begin{bmatrix} \phi \\ \psi \end{bmatrix} \tag{10}$$

Measurements are given as

$$z_k = z + d \tag{11}$$

Using (8), the system estimates an expected measurement based on the current position of the robot $[\phi, \psi]$ with

$$\tilde{z}(\phi,\psi) = h(x_{k|k+1}, u_k) = \boldsymbol{T}(\psi)^\top \boldsymbol{M}_k^\top \boldsymbol{V}_{i,j,k} \boldsymbol{M}_k \boldsymbol{T}(\phi) \tag{12}$$

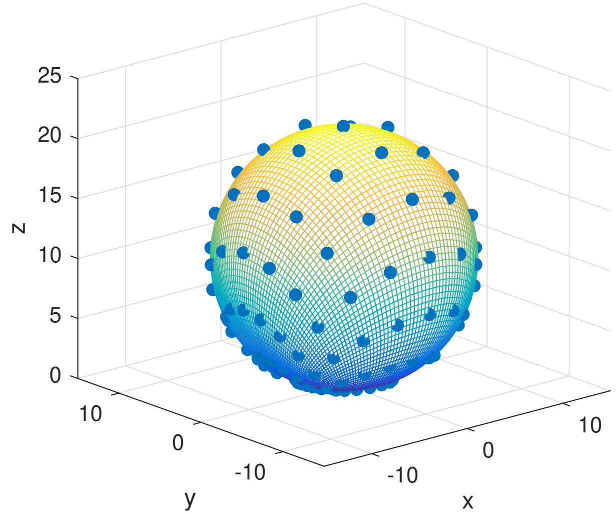


Figure 8: A set of 225 control points used to represent a sphere. The control points are uniformly spaced at fixed intervals over the angles ϕ and ψ of the robot's coordinate frame.

It then compares this to the measurement obtained from the OCT-sensor and calculates the innovation term

$$\tilde{y}_k = z_k - h(\hat{x}_{k|k+1}, u_k) \tag{13}$$

The covariance estimate prediction, its innovation term and the Kalman gain are computed as follows

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{P}_{k-1|k-1} + \boldsymbol{Q}_k \tag{14}$$

$$\boldsymbol{S}_k = \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^\top + \boldsymbol{R}_k \tag{15}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^\top + \boldsymbol{S}_k^{-1} \tag{16}$$

These can then be used to update the state estimate and covariance estimate with

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + \boldsymbol{K}_k \tilde{y}_k \tag{17}$$

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_{k|k-1} \tag{18}$$

The covariance of the observation noise R equals the square of the standard deviation of the measurement noise. As explained in section III-A, assuming the measurement noise is Gaussian, this value equals 0.0221 mm. This results in $R = 0.000488$ mm. The covariance of the process noise is given by the $(m \cdot n) \times (m \cdot n)$ matrix Q. As of yet it is assumed that the environment is static and therefore Q equals 0. In section III-E it will be explained how Q can be used to describe model uncertainties that increase over time.

$$R = 0.000488, \; Q = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \tag{19}$$

The covariance of the state estimate is given by the $(m \cdot n) \times (m \cdot n)$ matrix P.

D. Fitting the Retina with B-Splines and an EKF

To create a proper fit of a surface, two properties of a B-Spline are important. These are the density of the control points and its order k.

1) Choosing the B-Spline order: The continuity at the knots of the B-Spline surface is dependent on its order. With order k the continuity at the knots will be equal to C^{k-1} [23]. Meaning $k-1$ derivatives will be continuous at the knots. This could be beneficial if higher order derivatives need to remain smooth in these areas. The downside of this is that with increasing order the resulting surface will be smoothed out over its control points, which can be seen as a form of low-passing. At the start of this chapter it was stated that the path planned based on the model should be smooth without any adjustments. Creating a path that can be executed smoothly requires the acceleration to be bounded. This results in the path having to be at least C^1 continuous. Since the edges of the path will be following the B-Spline surface model, it is beneficial to have the model also be at least C^1 continuous. To achieve this, a second order B-Spline surface can be used. To further increase smoothness of the path the jerk can also be bounded. The model can take this into account by using a third order B-Spline surface.

2) Choosing the control point density: The second property of the B-Spline surface that needs te be determined is the density of the control points. Choosing an optimal density results in an optimal balance between the resulting fit and the required number of measurements to converge to that fit. Having a too high density allows the model to follow the environment very accurately, but requires many measurements to do so. This is because the affected area of each control point gets smaller as the density increases. Having a too low density allows the model to converge very quickly to a fit, but the fit might not be as accurate.

To determine what the influence of the control point density is in the case of fitting the retina, several different densities have been tested using a second order B-Spline. The test consists of fitting the two sections from Figures 3b and 3c as well as a 9 mm section of the foveal area of the retina. The foveal area has been chosen due to it having the highest rate of change. As explained in section III-A, in this area the position of the retina varies 0.25 mm over approximately 1.2-1.5 mm. The two straight lines are used to test how the model will fit a regular area of the retina. All of these measurement sections have noise laid over them from the patient's heartbeat, breathing and the distance sensor's noise. To create a ground truth from which the model fitting error can be calculated the measurement sections are low passed at just below 0.9 Hz. This removes the noise from the heartbeat and distance sensor and leaves the low frequency breathing and retinal shape. Removing the breathing noise was not possible without removing parts of the retinal structure as well. The resulting assumed ground truths of each data section are shown as the black lines in Figures 9, 10 and 11.

Each section will be fitted using all data points of the ground truth from that section for varying control point densities. These are 1, 3, 6, 9 and 18 mm between consecutive points. Figures 9, 10 and 11 show the resulting fits and their errors for the various densities. An upper bound of 18 mm was chosen due to the length of the measurement section from Figure 11. In the case of Figures 9 and 10 the data section length approximates 9 mm. When having 9 mm spacing between control points, a single knot interval of a second order B-Spline curve is defined over a length of 9 mm. This means that increasing the spacing beyond this point will have no changes on the sections from Figures 9 and 10. The length of the data section from Figure 11 approximates 18 mm and its fit will not change for increasing spacing beyond this point.

TABLE I: Standard deviations σ of the error for varying control point spacings.

	Standard deviation σ [μm]			
Spacing [mm]	Fovea	Line 1	Line 2	99.99% bounds [μm]
1	26.7	8.1	29.8	115.9
3	51.8	25.9	71.4	277.8
6	67.1	27.0	87.5	340.4
9	120.7	27.8	99.0	469.6
18	120.7	27.8	118.0	469.6

Assuming the fitting errors are Gaussian, Table I shows their standard deviations. It shows an increasing error margin for higher control point spacings, though this increase seems irregular and dependent on the particular section. To determine if the fits are accurate enough such that positions as close as 0.5 mm from the ground truth fall within the 99.99% confidence bounds, the bounds are determined using the highest of the three standard deviations for each density. This results in the confidence bounds shown in the last column of Table I. Based on these confidence bounds, all various densities create fits that fulfill the requirements. Therefore a spacing of 9-18 mm is used to keep the number of measurements required for convergence low. The next step would be to simulate even lower densities on larger sets of data to find the point where the confidence bounds cross the 0.5 mm threshold.

E. Uncertainty estimation

The variance of the environment estimate $\hat{z}$ for a certain position $[\phi, \psi]$ can be calculated based on the current state estimate $\hat{x}$. To do this, the correlation between the relevant local control points and the system estimate needs to be determined. This can be done by adapting the method from [24] to a B-Spline surface as follows.

Using (8), a point $[\phi, \psi]$ in the environment is estimated by the model using the $n \times n$ control points from $\boldsymbol{V}$. Where n is the 1+order of the B-Spline surface. To determine how the environment estimate is correlated with these control points at the position, the Jacobian $\frac{\partial z(\phi,\psi)}{\partial \hat{x}_k}$ is calculated. This results in

$$\hat{\boldsymbol{H}}_{\phi\psi} = \begin{bmatrix} \hat{H}_{11} \\ \hat{H}_{12} \\ \vdots \\ \hat{H}_{mn} \end{bmatrix}. \tag{20}$$

Which gives the correlation between between the environment estimate and its influencing control points at $[\phi, \psi]$. Since

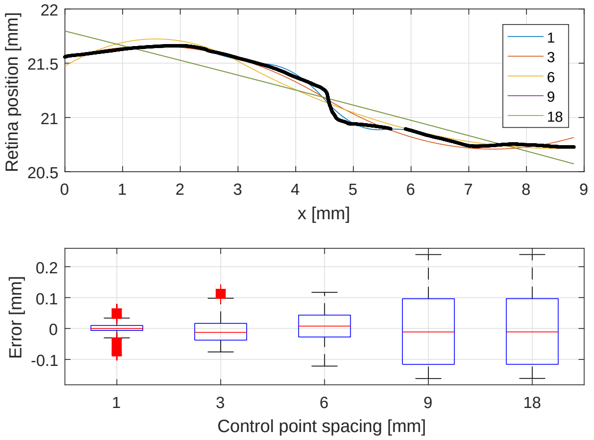


Figure 9: The upper plot shows the fits of 12000 measurements taken from foveal area for varying control point densities. The lower plot shows a boxplot of the errors for each fit. The blue box marks the 25th and 75th percentiles of the errors. The black lines extend to the most extreme values that are not considered outliers. The red lines indicate the outliers.

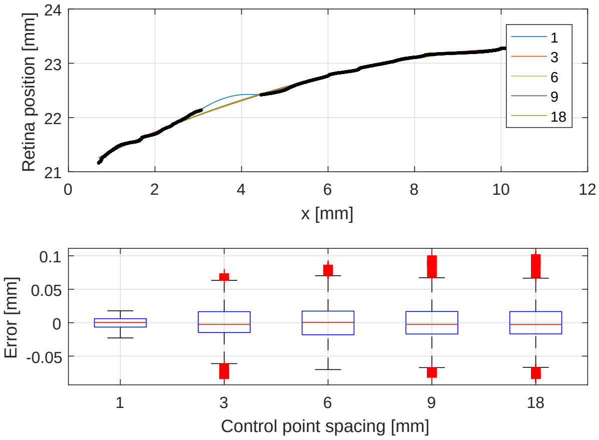


Figure 10: The upper plot shows fits for varying control point densities of 39000 measurements that form a straight line across the retina. The lower plot shows a boxplot of the errors for each fit. The blue box marks the 25th and 75th percentiles of the errors. The black lines extend to the most extreme values that are not considered outliers. The red lines indicate the outliers

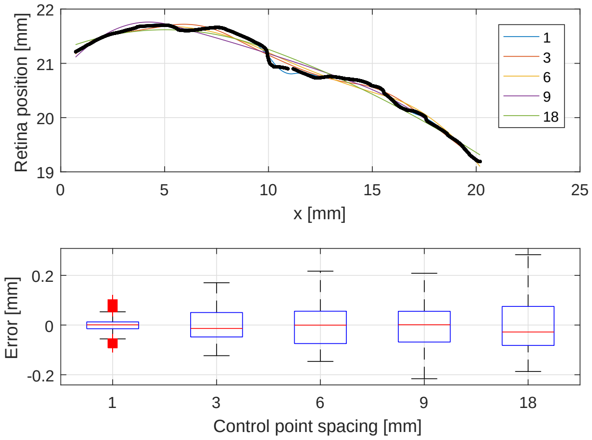


Figure 11: The upper plot shows fits for varying control point densities of 45000 measurements that form a straight line across the retina. The lower plot shows a boxplot of the errors for each fit. The blue box marks the 25th and 75th percentiles of the errors. The black lines extend to the most extreme values that are not considered outliers. The red lines indicate the outliers.

only the control points of $\boldsymbol{V}$ are correlated with this specific location, only their respective entries in $\hat{\boldsymbol{H}}_{\phi\psi}$ are non-zero. Looking at (8), it can be determined that the correlation terms in $\hat{\boldsymbol{H}}_{\phi\psi}$ are only dependent on the coordinates ϕ and ψ and the static matrix $\boldsymbol{M}_k$. This means that they do not change and $\hat{\boldsymbol{H}}_{\phi\psi}$ can be completely determined preemptively for every position in the environment.

With $\hat{\boldsymbol{H}}_{\phi\psi}$ known, the variance of the state estimate $\hat{\boldsymbol{x}}$ at $[\phi,\psi]$ then equals $\Sigma_{\hat{x}(\phi,\psi)} = \hat{\boldsymbol{H}}_{\phi\psi}\boldsymbol{P}$. The variance of $\tilde{z}(\phi,\psi)$ can then be found by

$$\Sigma_{\hat{z}} = |h(\boldsymbol{\Sigma}_{\hat{x}}, \phi, \psi)| = \left| h(\hat{\boldsymbol{H}}_{\phi\psi}\boldsymbol{P}) \right| \tag{21}$$

As an example, the 99.99% certainty interval then equals

$$\hat{z} \pm 3.891\sqrt{\Sigma_{\hat{z}}} \tag{22}$$

1) Choosing an initial P: An important aspect of this certainty interval is its initial value. When the model and EKF are in their initial states, the given certainty interval should be such that the retina will always fall within its bounds. This can be achieved by correctly setting the initial values of $\boldsymbol{P}$. As explained in section II-B the retina can be described as a sphere with an average radius of 10.74 mm. This means that to be sure the retina always falls within the certainty interval, even without any measurements, the initial values of $\boldsymbol{P}$ need to result in a certainty interval greater than 10.74 mm. Figure 12 shows this value to be 21.3.

2) Time dependent uncertainty estimate: The retina is organic material and moves with the patient. Therefore, it cannot be stated with absolute certainty that an area remains exactly the same between two measurements. Especially as time between measurements increases, it should be assumed

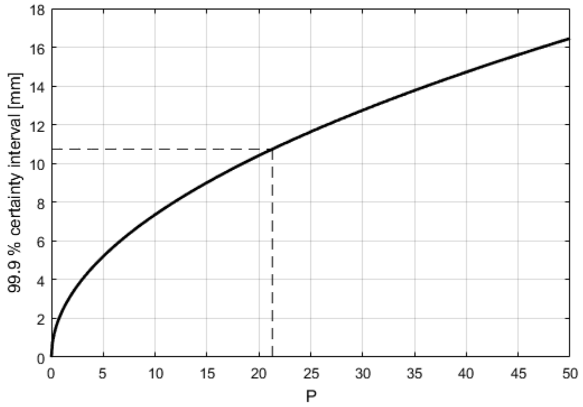


Figure 12: The initial 99.99% certainty interval for increasing values of P. The radius of the sphere that fits the average human retina equals 10.74 mm. P is initialized with 21.3 on its diagonal to create a certainty interval that guarantees to cover the retina even without measurements.

that the shape or position of an area could have changed. To include this time dependent growing uncertainty in the EKF, the covariance matrix of the process noise $\boldsymbol{Q}$ can be used to let $\boldsymbol{P}$ grow every iteration of the EKF by

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{P}_{k-1|k-1} + \boldsymbol{Q}_k. \tag{23}$$

By changing the diagonal terms of $\boldsymbol{Q}$ to be non-zero, the variance of the states given by the diagonal of $\boldsymbol{P}$ will increase over time. This results in an increasing Kalman gain (16), which allows for quicker convergence to new measurements based on (17). Setting the off-diagonal term to non-zero causes the covariance between the states to increase over time. This will cause measurements to influence control points outside of their local group $\boldsymbol{V}$. Doing this can be useful to allow points that remain unmeasured for long periods of time to be modelled based on other areas.

IV. Coverage path planning

Before a path can be planned, it is important to create a strategy that will be applied to remove the vitreous. Moving through the vitreous will have two important effects that need to be considered. The first is that the vitreous is attached to the retina. Moving the vitreous will pull on the retina, which can result in it detaching from the surface. The second effect is that the surgery system cannot detect the vitreous. So the only way to guarantee its entire removal is to cover the workspace in the eye entirely, while making sure the vitreous does not move to previously cleared areas. To minimize both these effects, the vitreous is cleared using a 2D layer decomposition strategy [12]. Starting at the scleral entry the vitreous is removed layer by layer, putting only the tip of the vitrectome into contact with the vitreous.

A. Creating the path

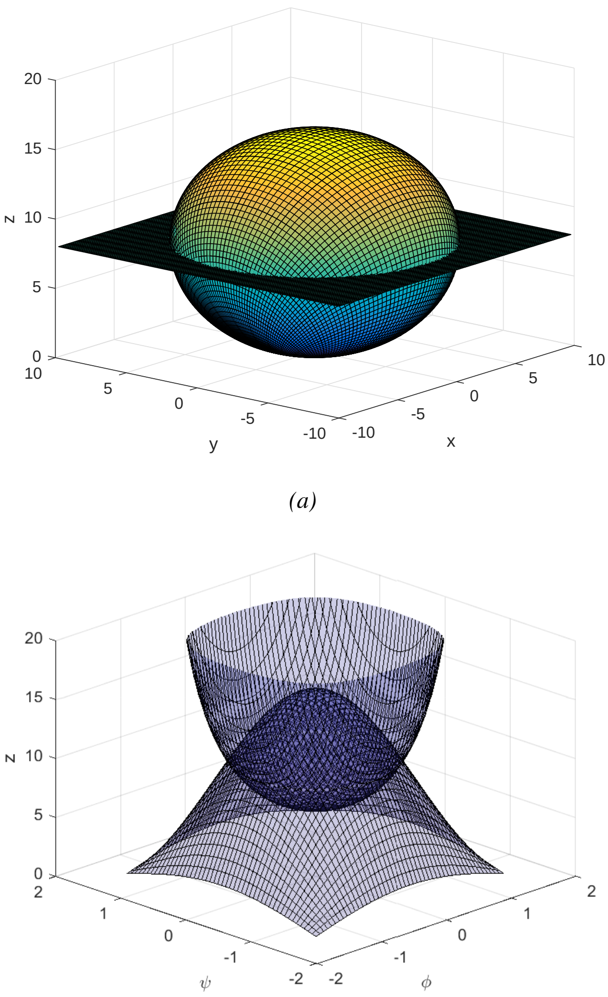
Figure 14 gives an example of the path. To create it, the model is divided into layers based on the radius r_v of the area around the vitrectome that is cleaned when stationary. These layers are positioned at fixed depths $z_l = l \cdot r_v$, with $l = [1, \dots, \left\lfloor \frac{z_{max}}{r_v} \right\rfloor]$. The layered path will be based on the B-Spline model. The exact size and shape of the model varies over time and the path planning should be able to follow these changes. It does this by basing the outer ring each layer on the shape of the model at that layer's depth z_l . Figure 13 visualizes how the models shape at z_l is determined. First, as shown in Figure 13a, a flat surface is positioned at z_l. Then the model and surface are transformed to the coordinate frame of the robot, shown in Figure 13b. Finding the intersecting points is done numerically by subtracting the two surfaces and finding the points with errors closest to zero. The resulting points are then interpolated to create a continuous and smooth outer line for the path. This outer line can then be scaled inwards based on r_v to cover the layer. Once all layers are filled using this method, they are connected to create a single path that covers the entire model. The connections between the lines and layers are made such that the path can 'open up' and encircle the posterior section of the lens. Every time the model is updated, the path will follow by updating the outer lines of each layer. An extra check is performed to guarantee that the path does not collide with the posterior section of the lens. Any points along the path that lie within the lens' cone as explained in section II-B will be removed.

B. Guaranteeing complete coverage

A grid of checkpoints is used to determine if the robot has covered the entire inside of the eye, assuming the model is a perfect fit of the retina. This coverage grid is uniformly spaced throughout the eye. Once the robot cleans a checkpoint it is marked as such. Figure 15 plots the percentage of unmarked cells against the path completion. At the end only six cells (0.0012%) of 1 mm^3 remain unmarked. After completion of the covering path, a new path will be planned visiting all unmarked checkpoints inside the model. To find a near-optimal route for this Traveling Salesman Problem (TSP) an Ant Colony System is used [25]. Once the last checkpoints are cleaned the entire model has been covered and, assuming the model represents the eye, the vitrectomy has been successfully completed.

V. Conclusion and Recommendations

B-Splines have been used in combination with an Extended Kalman Filter to create a modelling algorithm that can create a surface model of the retina built up from distance measurements. Through this model a volume covering path is planned that allows the robotic system to remove as much of the vitreous as possible. With this goal in mind a second or third order B-Spline surface is advised in order to maintain sufficient continuity at the knots. Based on simulations with in-vivo distance measurements a spacing of 9-18 mm between control points is advised to create a fit of the retina using a



(a)

(b)

Figure 13: To determine the outer line of a path layer at depth z_l, the intersecting line between the model and a surface placed at z_l is determined. (a) shows this surface and the model. (b) shows them transformed into the coordinate frame of the robot.

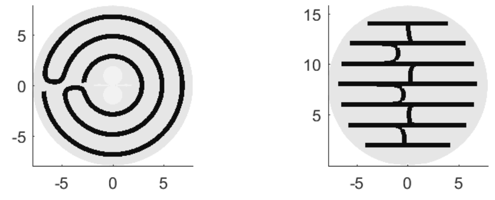


Figure 14: The left figure shows one of the layers that form the path. The outer line takes on the shape of the model. This line is then scaled inwards to cover the entire layer. The right figure shows the various layers that make up the entire path.

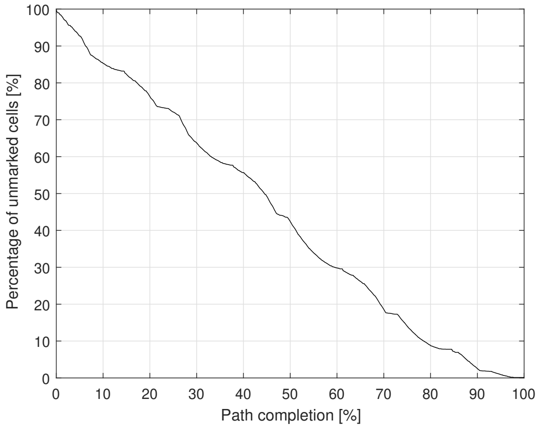


Figure 15: Shows the percentage of unmarked cells plotted against the path completion. The cells have a size of 1 mm^3. At the completion of the path only six cells remain unmarked.

second order B-Spline surface. Though simulations of larger data sets are necessary to check if this spacing can increase further.

The path planning uses a 2D layer decomposition strategy to divide the volume of the eye. This has the advantage of allowing the robot to remove the vitreous layer by layer, minimizing contact between it and the vitreous. This reduces chances of tearing the retina as well as moving vitreous to already cleaned places. To check if the robot has completely covered the eye a coverage grid is used. Once a robot cleans a point on the grid it is marked as such. The remaining checkpoints will be cleaned one by one at the end of the surgery to guarantee complete coverage.

In this paper, the control points were uniformly placed on the retina. To further improve upon this work it is advised to research the advantages of adjusting the control point positions online. The ability to move or even add control points would allow areas with higher detail to use more control points as needed and be fitted better. Areas with low detail could on the other hand use less control points and be fitted faster requiring less measurements to converge.

The intersecting line between each layer of the path and the model is currently determined numerically. This has limitations based on the resolution of the evaluation of the model. Finding the intersection algebraically will possibly yield a more optimal result and could be computationally less expensive.

The model has as of yet no method to cope with movements of the patient. Ideally, when the eye is moved by external factors the model should be able to detect this and shift along to remain accurate. A method of adding extra states to the EKF that describe the position of the center of the eye could be created based on [26].

References

[1] Rosen, J., Hannaford, B., and Satava, R. M., 2011. *Surgical Robotics*. Springer.

[2] de Jonge, N., Willekens, K., Giani, A., Vander Poorten, E., Esteveny, L., Cereda, M., and Faridpooya, K., 2015. "Use Case for European Robotics in Ophthalmologic Micro-Surgery". pp. 1–32.

[3] Camarillo, D. B., Krummel, T. M., and Salisbury, J. K., 2004. "Robotic technology in surgery: Past, present, and future". *American Journal of Surgery,* **188**(4 SUPPL. 1), pp. 2–15.

[4] Burschka, D., Corso, J. J., Dewan, M., Lau, W., Li, M., Lin, H., Marayong, P., Ramey, N., Hager, G. D., Hoffman, B., Larkin, D., and Hasser, C., 2005. "Navigating inner space: 3-D assistance for minimally invasive surgery". *Robotics and Autonomous Systems,* **52**(1), pp. 5–26.

[5] Jalba, A. C., and Roerdink, J. B., 2007. "Efficient surface reconstruction using generalized Coulomb potentials". *IEEE Transactions on Visualization and Computer Graphics,* **13**(6), pp. 1512–1519.

[6] Rouhani, M., Member, S., Sappa, A. D., Member, S., and Boyer, E., 2015. "Implicit B-Spline Surface Reconstruction". pp. 22–32.

[7] Morse, B. S., Yoo, T. S., Rheingans, P., Chen, D. T., and Subramanian, K. R., 2001. "Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions". *Proc. SMI 2001 Int Shape Modeling and Applications Conf.*, pp. 89–98.

[8] Liu, H., Wang, X. I. N., and Turk, S., 2007. "a Fast Method for Implicit Surface Reconstruction Based on Radial Basis Functions Network From 3D Scattered Points". pp. 459–465.

[9] Yu, Y., 1999. "Surface Reconstruction from Unorganized Points Using Self-Organizing Neural Networks". pp. 1–4.

[10] Pokropek, A., 1997. "Spline Approximation using kalman filter state estimation".

[11] Jauch, J., Bleimund, F., Rhode, S., and Gauterin, F., 2017. "Recursive B-spline approximation using the Kalman filter". *Engineering Science and Technology, an International Journal,* **20**(1), pp. 28–34.

[12] Granna, J., Godage, I., Wirz, R., Weaver, K., Webster, R., and Burgner-Kahrs, J., 9998. "A 3D Volume Coverage Path Planning Algorithm with Application to Intracerebral Hemorrhage Evacuation". *IEEE Robotics and Automation Letters,* **1**(2), pp. 876–883.

[13] Weijers, R., 2017. "Retinal Structure Recognition in Images reconstructed from A-Scans of an OCT-probe".

[14] Williamson, T. H., 2013. *Vitreoretinal surgery, second edition.*

[15] Units, S. I., 1994. *SI units in radiation protection and recommendations of the national council on radiation*. No. 82.

[16] Klufas, M. A., 2016. "Trocar/ Cannula Roundup".

[17] Vurgese, S., Panda-Jonas, S., and Jonas, J. B., 2012. "Scleral thickness in human eyes". *PLoS ONE,* **7**(1).

[18] Helga Kolb, 2012. "Photoreceptors. Webvision: The organization of the retina and visual system.".

[19] de Boor, C., 1980. "A Practical Guide to Splines.". *Mathematics of Computation,* **34**(149), p. 325.

[20] de Boor, C. "B(asic)-Spline Basics".

[21] Farin, G., 2002. *Curves and Surfaces for CAGD: A Practical Guide*.

[22] Lyche, T., and Morken, K., 2002. "Spline Methods Draft". p. 36.

[23] Comba, J. L., 1993. Continuity Aspects of Spline Curves.

[24] Douven, Y. G. M. "Uncertainty Estimation using an Extended Kalman Filter". pp. 1–2.

[25] Dorigo, M., 2012. "Ant Colony Optimization". *Expert Review of Clinical Immunology,* **8**(2), pp. 123–133.

[26] Douven, Y. G. M., Naus, G. J. L., Molengraft, M. J. G., and Steinbuch, M. "Observer-based SLAM and a model with interconnected landmarks for the retina". pp. 1–4.