AIMS | African Institute for Mathematical Sciences CAMEROON

Uveal Melanoma: Optic Nerve Study

Zeinab Almahdi Mohammed Haroon (zeinab.mohammed@aims-cameroon.org)
African Institute for Mathematical Sciences (AIMS)
Cameroon

Supervised by: Dr. Jan-Willem M. Beenakker and Myriam Jaarsma
Leiden University Medical Center (Radiology and Ophthalmology) , Leiden University - Netherlands

31 May 2019

Submitted in Partial Fulfillment of a Structured Masters Degree at AIMS-Cameroon

Abstract

Uveal melanoma is the most common primary intraocular malignancy and the leading primary intraocular disease which can be fatal in adults and cause vision loss. In this essay, we investigated the optic nerve shape changes under different gaze angles to improve the proton therapy planning for uveal melanoma patients by developing a segmentation algorithm in 3T Magnetic Resonance Imaging (MRI). The feasibility of having the optic nerve shape changes under different gaze angles were confirmed. The relative gaze angles have been calculated for the nine gaze images as well. The future research will focus on gaze angles interpolation to find the optimal gaze angle that enables algorithm to be used for further eye-model development which can enhance the proton therapy planning for uveal melanoma patients.

Declaration

I, the undersigned, hereby declare that the work contained in this essay is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Zeinab Almahdi Mohammed, 31 May 2019.

Contents

1 Introduction **1**

1.1 Uveal Melanoma 1

1.2 Literature Review 2

1.3 The Importance Of Optic Nerve Study of Uveal Melanoma Patients 3

2 MeVisLab Software Development Kit (SDK) **4**

2.1 MeVisLab Software Development Kit (SDK) Introduction: 4

2.2 Methodologies 5

2.3 Results 10

3 Three dimensional shape interpolation **12**

3.1 Curve Fitting 12

3.2 Polynomial Regression 13

3.3 Polynomial Fitting in 2D and 3D 13

3.4 Describing Optic Nerve Shape in 3D Fitting 17

3.5 Results 18

3.6 Discussion 24

4 Conclusion **25**

4.1 Future Perspective 25

Acknowledgements **26**

References **28**

List of Figures

1.1 Eye Melanoma 2

2.1 3T MRI images network for uveal melanoma patients - view2D 6
2.2 Separate the nine gaze angle DICOM images 7
2.3 Positions of optic nerve center within the nine gaze angles images 9
2.4 Gaze angle images' comparison 10
2.5 Optic Nerve Network Layout 11

3.2 Third dimensional fitting results of image(1-9) 20
3.3 Gaze angles 23

1. Introduction

1.1 Uveal Melanoma

Melanoma develops from the cells that produces the dark-colored pigment melanin, which is responsible for our skin coloring. These cells, called melanocytes, are found in other places in our bodies, too: our hair, the lining of our internal organs, and our eyes. So while most melanomas do form on the skin, it is possible for a melanoma to form elsewhere. When it forms in the eye, its known as ocular melanoma or, more specifically, uveal melanoma, because it forms in the uveal tract of the eye.

The ocular melanomas, may arise from any of the three parts of the uvea (iris, ciliary body and choroid), and are sometimes referred to by their location, as choroidal melanoma, ciliary body melanoma, or iris melanoma as illustrated in figure (1.1) . The iris opens and closes to change the amount of light entering the eye. The ciliary body changes the shape of the lens inside the eye so it can focus. The choroid layer is next to the retina, the part of the eye that makes a picture. Most uveal melanomas originate in the choroid, followed by the ciliary body, and the iris. It is not clear why eye melanomas develop. We do know that people born with certain growths in or on the eye, as well as those with lighter colored eyes, are at a greater risk for developing ocular melanoma and the blue eye ocular melanoma occurs when the DNA of the pigment cells of the eye develop errors. These errors cause the cells to multiply out of control. The mutated cells collect in or on the eye and form a melanoma [12].

In this study, we look at the possibility of using MRI to measure optic nerve coordinates and describe it in 3D shape . We scanned two volunteers to collect the medical data. The resulting data were manually segmented and processed to describe the optic nerve shape in 3D that showed the feasibility to describe the optic nerve shape in 3D. This results enabled us to confirm the possibility of having the full-scale study being conducted to optimize the proton therapy planning model for uveal melanoma patients for therapy improvement and vision protection.

There are multiple therapy options available which are brachytherapy, proton beam therapy (PBT) and enucleation of the eye. Brachytherapy is often the therapy of choice as it is less expensive than PBT and because the radiation does not traverse healthy tissue which results in better outcomes. However, in medium to large tumours, PBT is the therapy of choice as brachytherapy can only reach a sufficient dose up to 7 mm thickness. PBT is also preferred when tumours surround the optic disk and fovea where plaques cannot be placed directly. The advantage of PBT compared to, for example x-rays is that it can deliver dose with a sharp decline, sparing healthy ocular tissue as much as possible. The disadvantage of PBT is that structures outside the eye are more exposed to radiation compared to brachytherapy, where a gold plate protects the radiation from leaking. Around 50 percent of the patients with uveal melanoma suffer from loss of sight and when planning a treatment the clinician should take into consideration some aspects as the size of the tumour and its location in the eye, how far it has grown or spread the stage of the tumour,how much is it affecting the patient's sight, the pathology report (when available), general health and fitness. The calculations methods do not yet accurately predict the dose delivered to the patient and thus do not always predict well whether loss of sight will occur.

It is hard to implement clinical trials to find the best Uveal Melanoma treatment. In this case patients are often prescribed chemotherapy, immunotherapy, or a combination of treatments (surgery, radiation therapy or both), most often proton beam therapy is used to tackle Uveal melanoma disease.

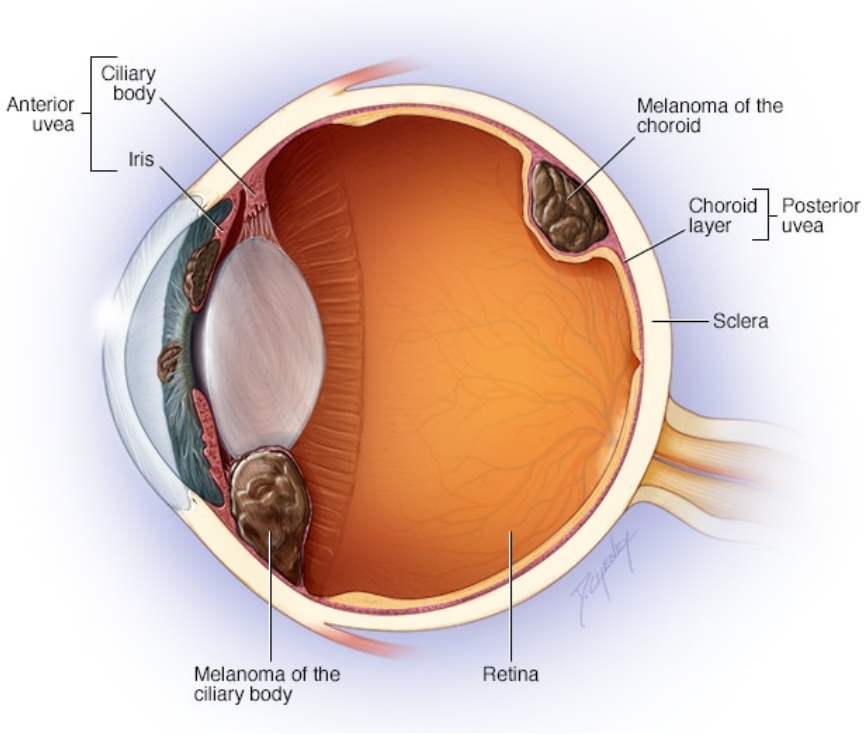


Figure 1.1: Eye Melanoma
(Choroidal melanoma, ciliary body melanoma and iris melanoma arises from any of the three parts of the uvea (iris, ciliary body and choroid [12]))

1.2 Literature Review

Proton beam radiotherapy of uveal melanoma

Proton beam radiotherapy of uveal melanoma can be administered as primary treatment, as salvage therapy for recurrent tumor. The physical properties of proton beams make it possible to deliver high doses of radiation to the tumor with relative sparing of adjacent tissues. This form of therapy is effective for a wider range of uveal melanoma than any other modality, providing exceptionally high rates of local tumor control. This is particularly the case with diffuse iris melanoma, many of which are unresectable. The chances of survival, ocular conservation, visual preservation and avoidance of iatrogenic morbidity depend greatly on the tumor size, location and extent. When treating any side-effects and/or complications, it is helpful to consider whether these are the result of collateral damage or persistence of the irradiated tumor ('toxic tumor syndrome') [19].

The aims of proton beam radiotherapy of uveal melanoma once spread is that preserving the eye itself of Melanoma disease with as much useful vision as possible by killing the tumor cells that can prevent Metastic Disease . As with other ionizing radiation, protons damage cells by disrupting DNA so that tumor cells lose their reproductive ability, hence entering senescence, tissue damage is greatest at the point where the protons stop moving so that there is a 'Bragg Peak' of ionization, with relative sparing of healthy tissues both proximal and distal to the tumor target. Furthermore, the beam can be highly collimated, thereby reducing collateral damage to adjacent structures such as the optic nerve and fovea. [2].

On December 2002, over 3000 patients with uveal melanoma had been treated with protons [13]. The 5-year actuarial local control rate was 96 percent for all sites within the globe, with an 80 percent overall survival. The probability of eye retention at 5 years was estimated to be 90 percent for the entire group and 97, 93, and 78 percent for patients with small, intermediate, and large tumours, respectively.

Egger (2003) [6] reported long-term results of eye retention after treatment of uveal melanoma with proton beam therapy. A total of 2645 patients were treated at Paul Scherrer Institute in Switzerland, between 1984 and 1999. The overall eye retention rates at 5, 10, and 15 years after treatment were 89, 86, and 83 percent, respectively [9].

1.3 The Importance Of Optic Nerve Study of Uveal Melanoma Patients

The optic nerve is the second cranial nerve and is one of the more unusual cranial nerves as it develops from the diencephalon. The optic nerve is a special sensory nerve responsible for vision and it does transfer the vision information from the retina to the brain which any damage on it can not be reversed by any current therapy, close surveillance with structural as well as functional testing, combined with early intervention while balancing the risk of side effects, is key to keeping different diseases under control and to preserving sight [21].

There can be many causes to optic nerve damage and in turn, the nerve damage can be of various forms as well. There are two main types:

- Optic nerve damage caused by trauma or injury
- Optic Nerve damage resulting from some disease or medical condition

Some of the more common eye diseases that cause optic nerve damage include glaucoma, Optic Neurits, Optic Nerve atrophy and Optic nerve head drusen as well as uveal melanoma. Depending on the cause, the treatment and precautions will obviously differ but, unfortunately, as mentioned above, for most cases of optic nerve damage, there are no treatments that can restore the sight. For most cases Knowing that the shape of the optic nerve (ON) in the therapy planning model may improve the accuracy of the prediction of the loss of sight. Due to the speed in which Proton Beam Therapy (PBT) is performed compared to brachytherapy, the gaze-angle is often adapted to minimize the dose for healthy tissue already. Therefore, studying the optic nerve shape changes under different gaze angles to improve the proton therapy planning for uveal melanoma patients by developing a segmentation algorithm in 7T MRI is important for this study.

A study enrolled 242 patients with uveal melanoma found that the uveal melanoma infiltrates the optic nerve in only 0.6 percent to 5 percent of patients and has been associated with high intraocular pressure, non-spindle cell type, juxtapapillary location, and blindness [11]. Therefore protecting the optic nerve of being damaged is one of the study goals that meant to prove algorithm feasibility to perform segmentation and to track the optic nerve shape all along the different images using MeVisLab software for image processing.

2. MeVisLab Software Development Kit (SDK)

2.1 MeVisLab Software Development Kit (SDK) Introduction:

In more than 20 years of development, MeVisLab has become one of the most powerful development platforms for medical image computing research. MeVisLab is defined as a rapid prototyping and development platform for medical image processing and visualization. With its image processing library, it fulfills the following requirements:

- Able to handle large, six-dimensional images (x, y, z, color, time, user-defined).
- Offers easy ways to develop new algorithms or changing/improving existing ones in a modular C++ interface, perfect for a fast-developing research area.
- Offers easy ways of combining algorithms to algorithm pipelines and networks.
- Fast and easy integration into clinical environments due to standard interfaces, for example to DICOM.
- Fair performance for clinical routine due to a page-based, demand-driven approach in the image processing.

The implementation of MeVisLab makes use of a number of well known third-party libraries and technologies, most importantly the application framework Qt, the visualization and interaction toolkit Open Inventor, the scripting language Python, and the graphics standard OpenGL. In addition, modules based on the Insight ToolKit (ITK) and the Visualization ToolKit (VTK) are available [8].

MeVisLab offers the possibility to create a graphical user interface (GUI) for an application consisting of various modules using an abstract module definition language (MDL). Dynamic functionality can be added to an application via scripting using JavaScript and Python. This way, it is possible to react on user interactions, to manipulate the processing pipeline (modules, networks or the user interface) or even to calculate and save results. Furthermore, the application logic is typically implemented using scripting.

MeVisLab has been used in a wide range of medical and clinical applications, including surgery planning for liver,lung,head and neck and other body regions, analysis of dynamic,contrast enhanced breast and Prostate MRI, quantitative analysis of neurologic and cardiovascular image series, orthopedic quantification and visualization,tumor lesion volumetry and therapy monitoring, enhanced visualization of mammograms, 3D breast ultrasound and tomosynthesis image data, and many other applications. MeVisLab is also used as a visualization techniques as well as a training and teaching tool for image processing as implementing MRI scanned images to visualize and process the images to track the eye optic nerve shape in 3D and 2D view directions.

2.2 Methodologies

2.2.1 Magnetic Resonance Imaging (MRI) Images.

The 3T magnetic resonance imaging (MRI) images have been obtained from two healthy volunteers in the age range of (25-30) years old. The measurements were performed on a Philips Achieva 3-Tesla MRI using a custom-built eye-coil as described in Beenakker 2016[1]. The MRI settings were optimized to achieve very short scanning times while maintaining the contrast the between optic nerve and surrounding tissue in the phase encoded images. Each gaze-angle -the image projection on the eye retina as observed from different points of view (viewing angles), and head position-if viewing of an object requires eye movement amplitude- is measured for 3 seconds with the three second blink break. Directly after scanning the images, the contrast should be checked for, because the movement of the eye can distort the image.

The images were obtained by using a cued-blinking protocol, as described in Wezel 2017 [20] to reduce unwanted eye movement. The volunteers had to look in an S shape, from right to left above, then from left to right in the middle and then from bottom right to left with nine viewing angles in total.

The 3T Magnetic Resonance Imaging (MRI) images have been extracted from a medical data into separate nine images with different gaze angles at S-shape looking - right to left above, left to right in the middle and bottom right to left- and shape coordinates through using MeVislab Software that meant for medical image processing and visualization.

2.2.2 MeVisLab Implementation.

In MeVisLab, the programming of image processing algorithms or interactive image / 3D scene manipulation is basically done by establishing networks that consist of modules and connections between modules. Modules encapsulate certain algorithms written in C++ and provide an interface in MeVisLab in the form of fields. Those fields can represent simple data like numbers or strings and also represent more complex data like three to six dimensional images.

MeVisLab screen above starts with an available workspace and some Views on the right (like the Output Inspector) and bottom of the screen (usually the Debug Output). In the Debug Output, you can find information about your MeVisLab installation and start-up, preferences and license file are loaded, and whether all packages are loaded correctly or with errors. The great thing about the Output Inspector is that it displays the output of any connector (or data connection) in the process chain (as long it is a format the inspector can interpret) and also to view the results of every step in the processing pipeline . The Module Inspector View can view and edit module field values [5].

Knowing that the Networks are connections between modules with which you can implement complex processing tasks from sets of standard ML, Inventor, WEM, CSO, ITK, or VTK modules. It is edited and saved as *.mlab files in MeVisLab. The figure (2.1) titled as " 3T MRI images Network for Uveal Melanoma- view2D" is the internal network of MRI images module is shown. The network consists of all three types of modules - image processing module (blue), one visualization modules (green) and some macro modules (brown)- and shows data connections as well as parameter connections.

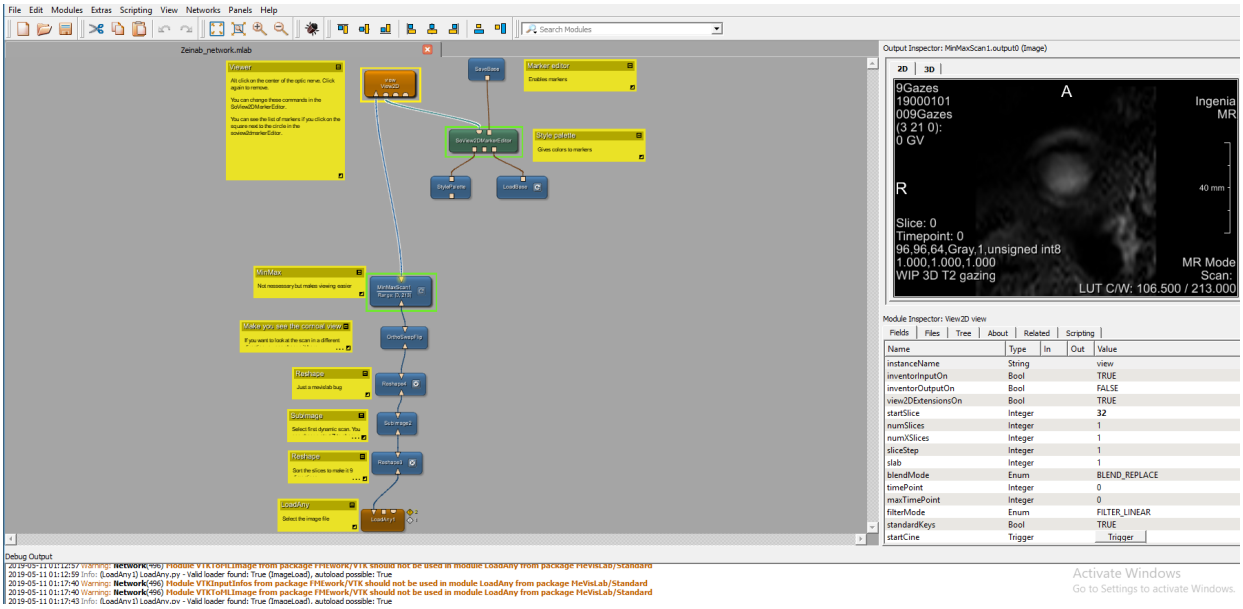


Figure 2.1: 3T MRI images network for uveal melanoma patients - view2D.
(The network consists of all three types of modules - image processing module (blue), one visualization modules (green) and some macro modules (brown)- that uses to display MRI images in 3D and 2D)

The images were processed within MeVisLab using the following steps

- DICM image data - eye images - has been loaded (**loadany1** module) to model the optic nerve shape in different gaze directions.
- Sort the image slices to nine different dimensions (**Reshap(3)**) module.
- Select the first image dynamic (**(**Subimage2) module).
- Select first image dynamic scan starting with (start z = 0) ending by (start z= 8) for the nine different images of the uveal melanoma patient (**Subimage2** module).
- View the scanned image in various directions (Transversal,Coronal and Sagittal), swapping and flipping the image horizontally, vertically and in depth as well ((**Subimage**) followed by (**Reshape(4)**) and (**OrthoSwapFlip**) modules).
- Make the image view easier and clearer (**MinMaxScan**).
- View the images at output Inspector in second and third dimensional shapes (**SoView2DMarkerEditor** and **View2D**).
- Upload and save the scanned images for checking (**LoadBase** and **Saveimage**).

Following the previous steps enabled us to separate the gaze angle images into nine various images that viewed in three different directions (transversal, coronal and sagittal) following the network layout (2.2). The separated images are displayed at output inspector individually in both second and third dimensional shapes, to collect the optic nerve coordinates in third dimension all along the nine images in different gaze angles that happened by clicking on the optic nerve center of the eye image as illustrated in (2.3).

The 3D volume processed images, that reconstructed in the transversal, sagittal and coronal view , has been saved (**SaveBase**) for later uses (**LoadBase**).

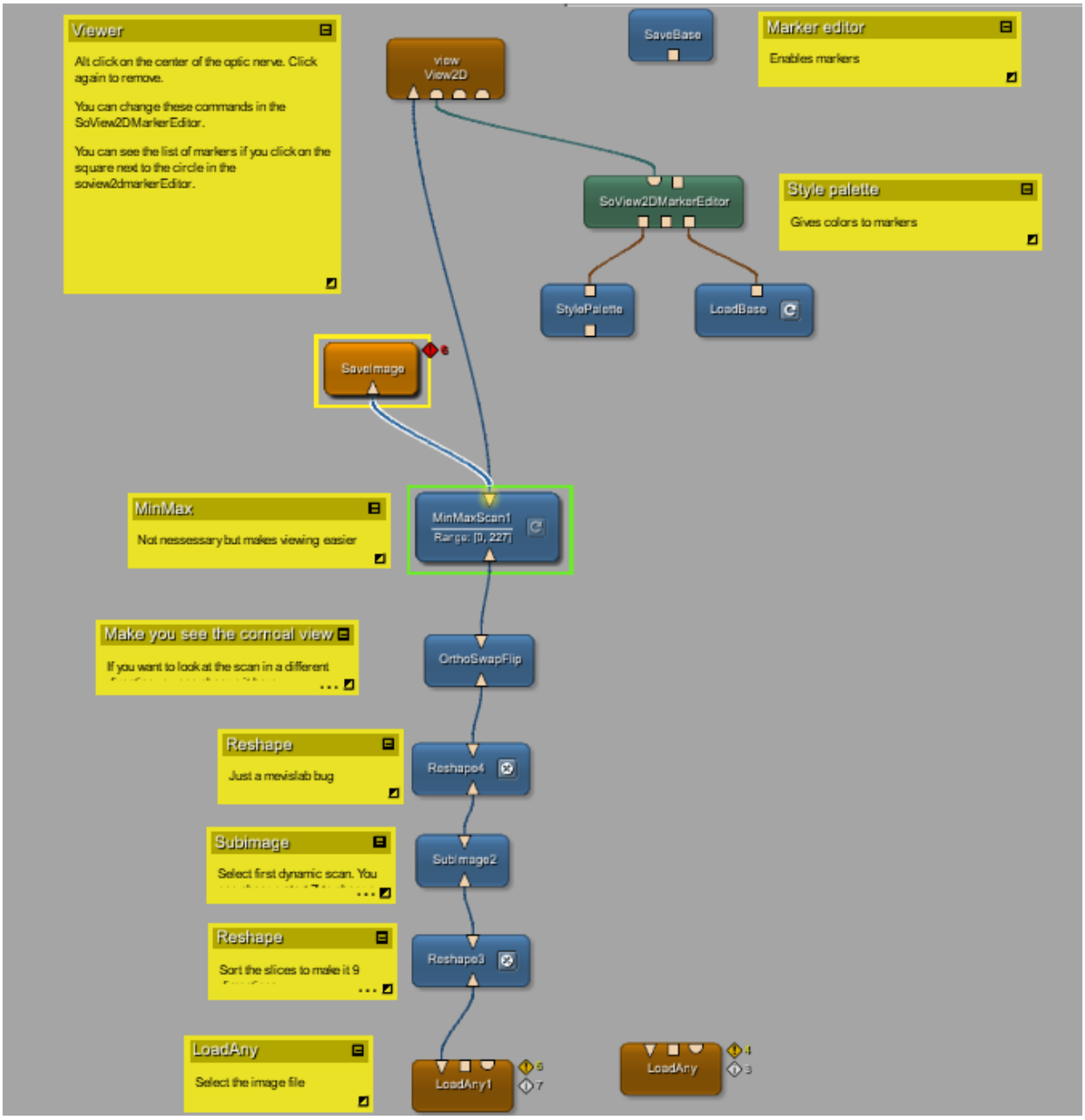
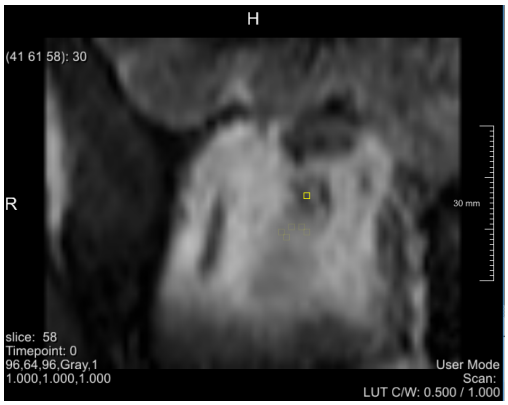


Figure 2.2: Separate the nine gaze angle DICOM images in various directions.
(Network layout used to separate the gaze angle images into nine various images that viewed in three different directions (transversal, coronal and sagittal))

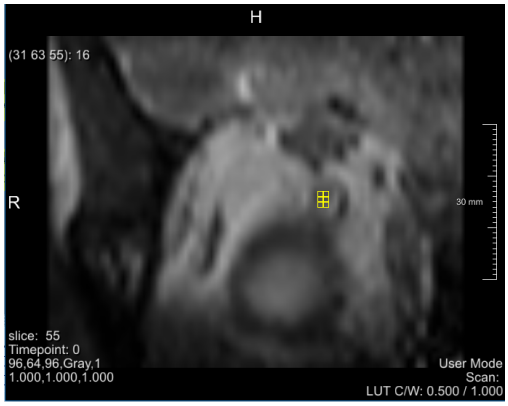
As mentioned, the following images illustrates the optic nerve center position within the nine gaze angle images in coronal view direction (2.3), attached to the network layout that followed in (2.2) which used to visualize the images through different nine eye movements that made the optic nerve (ON) shape changes .


H
(41 61 58): 30
R
30 mm
slice: 58
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000

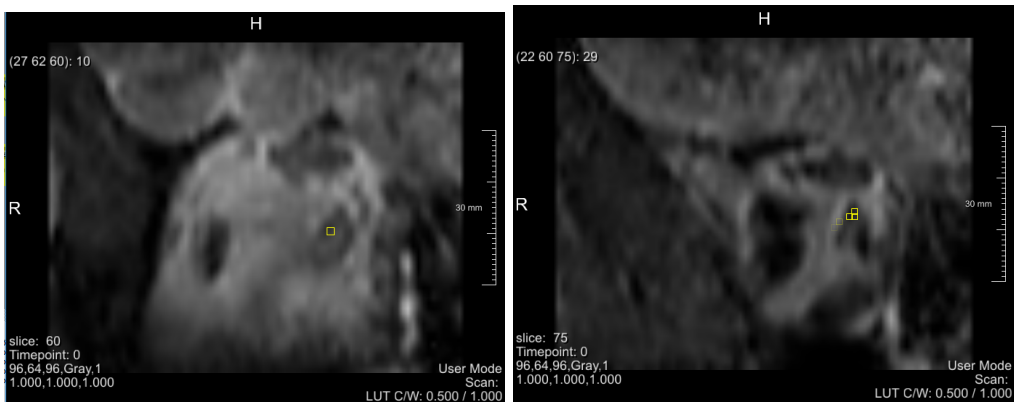

H
(11 63 57): 17
R
30 mm
slice: 57
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000


H
(20 62 57): 4
R
30 mm
slice: 57
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000


H
(34 62 57): 22
R
30 mm
slice: 57
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000


H
(31 63 55): 16
R
30 mm
slice: 55
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000


H
(15 51 58): 5
R
30 mm
slice: 58
Timepoint: 0
96,64,96,Gray,1
1.000,1.000,1.000
User Mode
Scan:
LUT C/W: 0.500 / 1.000



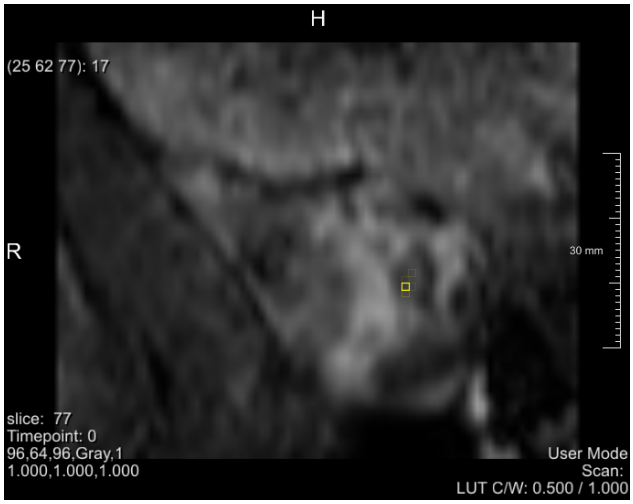


Figure 2.3: Positions of optic nerve center within the nine gaze angles images
(Nine gaze angle images illustrates the optic nerve center position, displayed in coronal view direction)

The nine gaze images are different, though look the same at output Inspector when viewed in 2D/3D view. The nine images have been saved in different files and images' comparison has been made for image one and two, using the two **loadAny** blocks and the **synchroviewer** module as shown in the following figure 2.4.

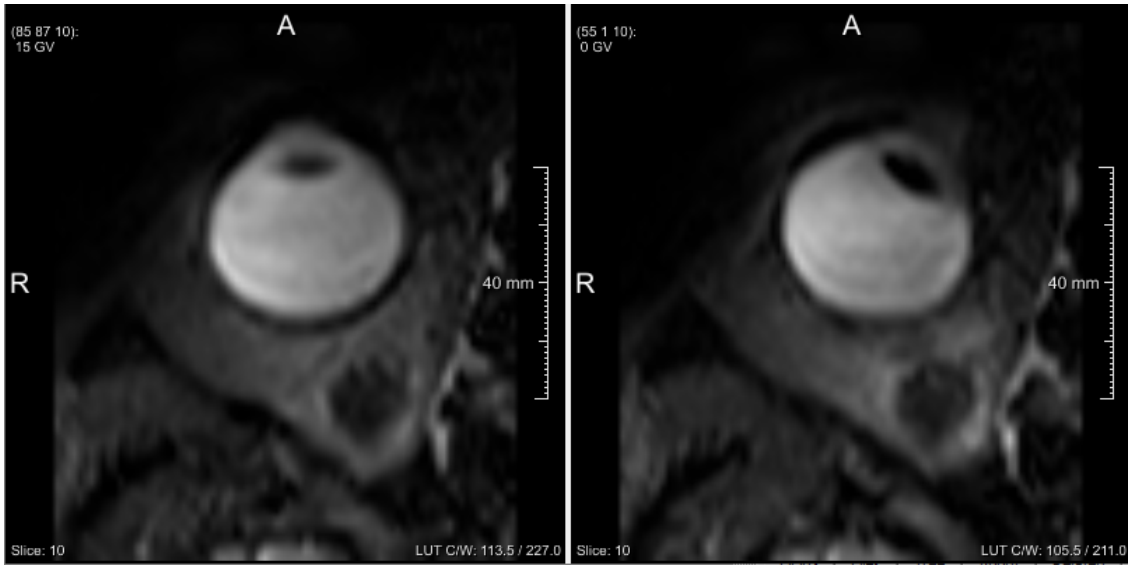


Figure 2.4: Gaze angle images' comparison.
(Showing the differences between image (1) and (2))

2.3 Results

The nine gaze angle images have been saved separately. The obtained images will be used to find the optic nerve (ON) coordinates of the nine images which have been determined by clicking on the center of the optic nerve for each image individually (Figure 2.3) following the networks layout (2.5).

The optic nerve shape has been divided into small pieces in different slices for the same image when displayed in 2D view in MeVislab, therefore for collecting the ultimate complete shape of the optic nerve we swapped and flipped the various slices of the same image horizontally, vertically and in depth ((XYZ), (YZX), (ZXY), (XZY), (ZYX) and (YXZ) orientations) in different view directions (transversal,coronal and sagittal) , using cornol view for checking the optic nerve position and the transversal view for tracking along the optic nerve shape.

The optic nerve coordinates have been collected in 3D (x_i, y_i, z_i) of the nine gaze images and the processed images have been saved along with their coordinates that used later for third polynomial curve fitting in 3D.

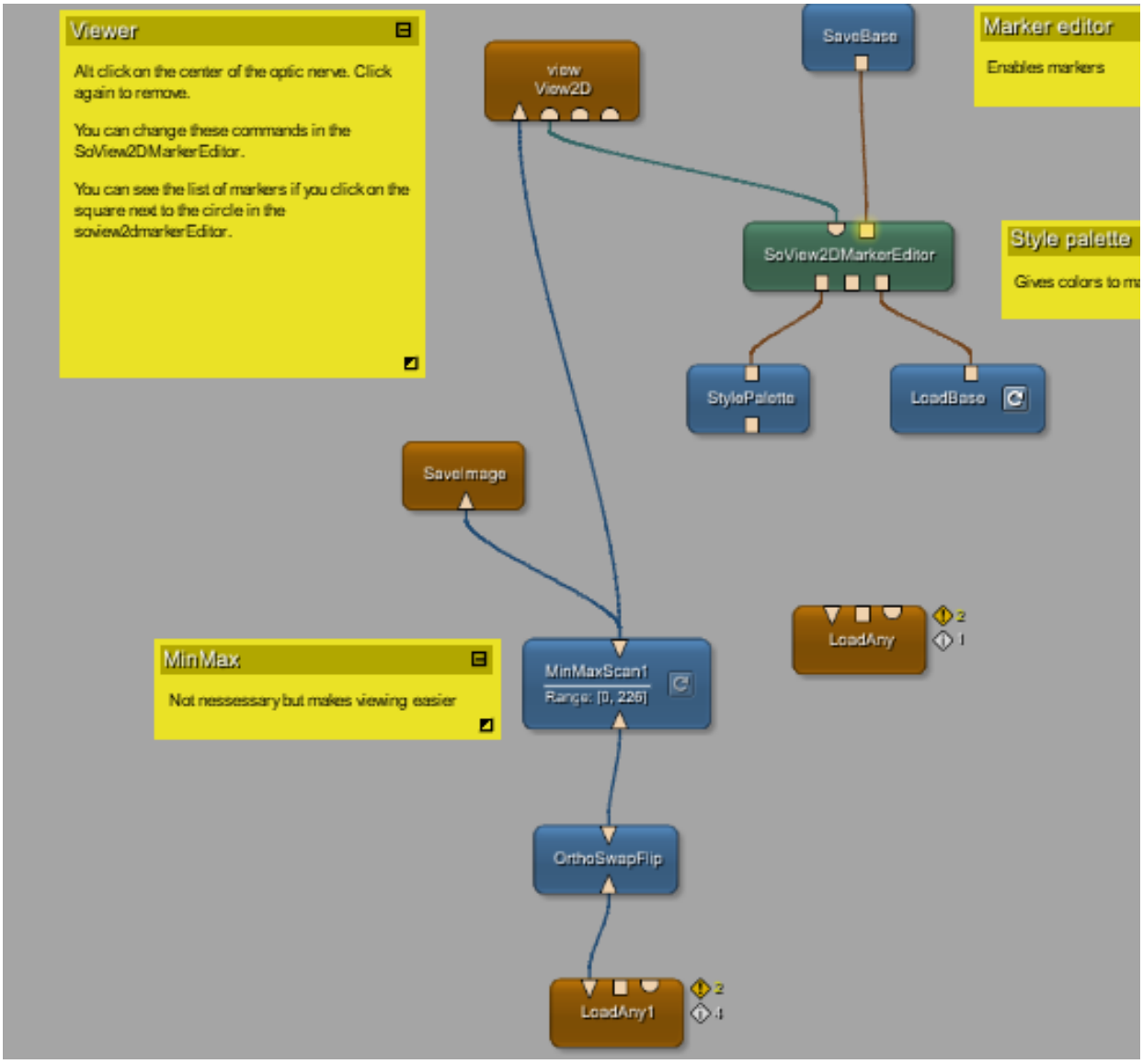


Figure 2.5: Optic nerve network layout.
(The optic nerve network that used to obtain the Optic nerve (ON) coordinates of the nine images which have been determined by clicking on the center of the optic nerve for each image individually)

3. Three dimensional shape interpolation

3.1 Curve Fitting

Curve fitting [10], also known as regression analysis, is used to find the 'best fit' line or curve for a series of data points. Most of the time, the curve fit will produce an equation that can be used to find points anywhere along the curve. In some cases, you may not be concerned about finding an equation. Instead, you may just want to use a curve fit to smooth the data and improve the appearance of your plot.

Regression Analysis refers to the study of the relationship between a response (dependent) variable y, and one or more independent variables the x_i. When this relationship is reasonably approximated by a straight line, it is said to be linear, and we talk of linear regression. When the relationship follows a curve, we call it curvilinear regression [16].

Usually, you assume that the independent variables are measured exactly (without random error) while the dependent variable is measured with random error. Frequently, this assumption is not completely true, but when it cannot be justified, a much more complicated fitting procedure is required. However, if the size of the measurement error in an independent variable is small relative to the range of values of that variable, least squares regression analysis is one of curve fits types among many others that can be used.

3.1.1 Types of Curve Fits.

The curve fits can be divided into three main categories [14]: Least Squares curve fits, nonlinear curve fits, and smoothing curve fits as following [17]:

1) Least Squares Curve Fits

Least Squares is a method of curve fitting [7] that has been popular for a long time. Least Squares minimizes the square of the error between the original data and the values predicted by the equation. While this technique may not be the most statistically robust method of fitting a function to a data set, it has the advantage of being relatively simple (in terms of required computing power) and of being well understood.

The major weakness of the Least Squared method is its sensitivity to outliers in the data. If a data point is widely different from the majority of the data, it can skew the results of the regression. For this reason, the data should always be examined before fitting [3].

2) Nonlinear Curve Fits

This is the most powerful fitting option because you can specify virtually any equation to be fitted to the data. It is calculated using an iterative procedure, starting with the initial guesses for the unknown parameters that were supplied with the equation. It then calculates a Chi Square value that represents the sum of the squared error between the original data and the calculated fit.

3) Smoothing Curve Fits

The fits are different from the other two types of fits, these curve fits do not generate an equation for the resulting curve, this is because there is no single equation that can be used to represent the curve. These curve fits are useful when you just want to improve the appearance of the plot by drawing a smooth curve through the data which can be either second or third dimensions data. Third order polynomial regression that use cubic regression curve-fit has been used to find the best fit for the optic nerve coordinates.

3.2 Polynomial Regression

3.2.1 Cubic Regression Curve-fit (third order polynomial regression).

Polynomial regression is [15] a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modeled as an nth degree polynomial in x as in the first order polynomial is just a straight line, second order polynomial is a parabola, third order is a cubic line, Basically adding one order to the polynomial adds another bend in the line. Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y, denoted $E(y|x)$, and has been used to describe nonlinear phenomena such as the growth rate of tissues, the distribution of carbon isotopes in lake sediments, and the progression of disease epidemics.

Polynomial regression models are usually fit using the least square method that known as a procedure for finding the best-fitting curve to a given set of points by minimizing the sum of the squares of the offsets ("the residuals") of the points from the curve. The sum of the squares of the offsets is used instead of the offset absolute values because this allows the residuals to be treated as a continuous differentiable quantity. However, because squares of the offsets are used, outlying points can have a disproportionate effect on the fit, a property which may or may not be desirable depending on the problem at hand [22]. This method is needed when it comes to cubic regression method that is a process in which the third-degree equation is identified for the given set of data. The cubic regression equation is (3.2.1) that in order three to be approximated mathematically later :

$$y = ax^3 + bx^2 + cx + d. \tag{3.2.1}$$

3.3 Polynomial Fitting in 2D and 3D

There are two types of fitting depending on the given data , the fitting can be linear when given 2D set of data, otherwise it is a curve fitting in 3D. Here is an explanation of polynomial approximation in second and third dimensions.

3.3.1 Polynomial Fitting in 2D.

To solve the linear fitting of 2D points we need to calculate the coefficients to find the function that best fits the curve to the data by using matrices form.

Consider the general form for a polynomial of order (j) :

$$f(x_k, a_1 \cdots a_j) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots + a_j x^j \ . \tag{3.3.1}$$

Which can be written in form,

$$f(x) = a_0 + \sum_{k=1}^{j} a_k x^k \ . \tag{3.3.2}$$

The general expression for any error using the least squares approach is :

$$err = \sum (d_i)^2 = \sum_{i=1}^{n} \Big(y_i - (a_0 + a_1 x_i + a_2 x_i^2 + a_3 x_i^3 + \cdots + a_j x_i^j) \Big)^2 \ . \tag{3.3.3}$$

The square of the deviations d_i^2 of the set of N-data points is given by:

$$\sum (d_i)^2 = \sum_{i=1}^{n} \Big(y_i - (a_0 + a_1 x_i + a_2 x_i^2 + a_3 x_i^3 + \cdots + a_j x_i^j) \Big)^2 \ . \tag{3.3.4}$$

To find the best line we should minimize the error (squared distance) between line and data points. to find the set of coefficients (a_k, a_0) so we can minimize (3.3.4) .

To minimize (3.3.4) , take the derivative with respect to each coefficient (a_k, a_0) , $k = 1, 3, \cdots j$ set each to zero.

$$\frac{\partial err}{\partial a_0} = -2 \sum_{i=1}^{n} \left[y_i - \left(a_0 + \sum_{k=1}^{j} a_k x_i^k \right) \right] \quad = 0 \ . \tag{3.3.5}$$

$$\frac{\partial err}{\partial a_1} = -2 \sum_{i=1}^{n} \left[y_i - \left(a_0 + \sum_{k=1}^{j} a_k x_i^k \right) \right] x_i = 0 \ . \tag{3.3.6}$$

$$\frac{\partial err}{\partial a_2} = -2 \sum_{i=1}^{n} \left[y_i - \left(a_0 + \sum_{k=1}^{j} a_k x_i^k \right) \right] x_i^2 = 0 \ . \tag{3.3.7}$$

$$\vdots \tag{3.3.8}$$

$$\frac{\partial err}{\partial a_j} = -2 \sum_{i=1}^{n} \left[y_i - \left(a_0 + \sum_{k=1}^{j} a_k x_i^k \right) \right] x_i^j = 0 \ . \tag{3.3.9}$$

Rewriting these (j+1) equations and put them into matrix :

$$\begin{bmatrix} n & \sum x_i & \sum x_i^2 \cdots & \sum x_i^j \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \cdots & \sum x_i^{j+1} \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \cdots & \sum x_i^{j+2} \\ \vdots & \vdots & \vdots & \vdots \\ \sum x_i^j & \sum x_i^{j+1} & \sum x_i^{j+2} \cdots & \sum x_i^{j+j} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_j \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum (x_i y_i) \\ \sum (x_i^2 y_i) \\ \vdots \\ \sum (x_i^j y_i) \end{bmatrix}$$

where all summations above are over $i = 1, \cdots n$ data points.

No matter what the order (j) , we always get linear equations with respect to the coefficients that enables us to use the following solution method, where we can find (A) stands for the first variables matrix , (X) is the coefficients matrix and (B) is the last solution matrix :

$$AX = B \ . \tag{3.3.10}$$

Then the coefficients $(a_0 \cdots a_k)$ are solved by multiplying the inverse of matrix (A) to matrix (B) as the below example explain this:

$$X = A^{-1}B \ . \tag{3.3.11}$$

3.3.2 Linear Fitting .

When the data represents measurements where the y-component is assumed to be functionally dependent on the x -component we introduce the least square fitting method . Given a set of points $(x_i, y_i)_{i=1}^m$ we determine the values of (a) and (b) so that the line ($y = a + bx$) best fit the points in the sense that the sum of the squared errors between the actual value (y_i) and the line values $(a + bx)$ is minimized when the error is being measured in only y-direction.

Breaking these steps into simpler way:

knowing that the measured quantity y (depended variable) is a linear function of x (independent variable):

$$y = a + bx \ . \tag{3.3.12}$$

The values of the coefficients $[a, b]$ can be estimated from a set of pairs as mentioned previously (x_1, y_1) , (x_2, y_2) , $(x_3, y_3) \cdots (x_n, y_n)$ which this estimation is known as least-squares linear regression that is considered a partial case of least-squares polynomial regression analysis.

Following the same previous steps as in (3.3.9)

$$-2\sum_{i=1}^{n} y_i + 2\sum_{i=1}^{n} a + 2b\sum_{i=1}^{n} x_i = 0 \ . \tag{3.3.13}$$

$$-2\sum_{i=1}^{n} y_i x_i + 2a\sum_{i=1}^{n} x_i + 2b\sum_{i=1}^{n} x_i^2 = 0 \ . \tag{3.3.14}$$

$$na + b\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \ . \tag{3.3.15}$$

$$a + b\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \ . \tag{3.3.16}$$

$$\begin{bmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum (x_i y_i) \end{bmatrix}$$

$$\Delta = n\sum x_i^2 - (\sum x_i)^2 \tag{3.3.17}$$

Find the coefficients (a and b) as following:

$$a = \frac{\begin{bmatrix} \sum y_i & \sum x_i \\ \sum x_i y_i & \sum x_i^2 \end{bmatrix}}{n\sum x_i^2 - (\sum x_i)^2} \ . \tag{3.3.18}$$

(3.3.19)

$$a = \frac{\sum y_i \sum x_i^2 - \sum x_i \sum x_i y_i}{n\sum x_i^2 - \sum (x_i^2)} \tag{3.3.20}$$

$$b = \frac{\begin{bmatrix} n & \sum y_i \\ \sum x_i & \sum x_i y_i \end{bmatrix}}{n\sum x_i^2 - (\sum x_i)^2} \ . \tag{3.3.21}$$

(3.3.22)

$$b = \frac{n\sum x_i y_i - \sum y_i \sum x_i}{n\sum x_i^2 - (\sum x_i)^2} \tag{3.3.23}$$

The final function that give us the best fit is (3.3.24) :

$$y = \frac{\sum y_i \sum x_i^2 - \sum x_{i0} \sum x_i y_i}{n\sum x_i^2 - (\sum x_i)^2} + \frac{n\sum x_i y_i - \sum y_i \sum x_i}{n\sum x_i^2 - (\sum x_i)^2} x \ . \tag{3.3.24}$$

This least square method is used to find the best fit for 3D optic nerve coordinates with the aid of the computer (programming languages) due to huge amount of Data we have that make the matrix method ineffective manually.

3.3.3 Polynomial fitting in 3D .

In 3D data, the z-component of the data is functionally dependent on the x- and y-components. Given a set of points $(x_i, y_i z_i)_{i=1}^{m}$ we determine the values of a_0 , a_1 and a_2 so that the line ($z = a_0 + a_1 x + a_2 y$) best fit the points in the sense that the sum of the squared errors between the actual value (y_i) and the line values $(a_0 + a_1 x_i + a_2 y_i)$ is minimized when the error is being measured in only Z-direction.

Then Similarly as in the Linear Fitting of 2D Points but by far more complicated expressions we are obtaining the coefficients of polynomials of higher degrees $(a_0, a_1, a_2.....a_n)$ yet the direct use of the previous expressions for $m > 1$ are almost never used. Instead, the system of normal equations is set and the solution vector of (a_0, a_1, a_2) coefficients is calculated usually with the aid of a computer (Programming languages).

Then the solution provides the least squares solution is when finding the coefficients $(a_0, a_1, a_2.....a_n)$ that will be substituting within the polynomial equation ($z = a_0 + a_1 x + a_2 y$), Knowing that the coefficients calculates by computers and in python programming language case there are multiple methods that can do the analytical solution, depending on the function we are trying to fit. To fit a straight line (numpy.linalg.lstsq) package is used . To fit a polynomial (numpy.polyfit) does. For an arbitrary function, use (scipy.optimize.fmin) then those functions return the parameters for the best possible fit when given the data.

3.4 Describing Optic Nerve Shape in 3D Fitting

In this project, we are fitting a line through the optic nerve coordinates for the nine gaze angles in parametric representation to approximate the given optic nerve coordinates that meant to be parametrized. It strongly affects the quality of approximation and one of the methods to be used is to iteratively modify the initial parametrization to minimize the error of approximation that will enable us to describe the optic nerve perfectly by using Python.

We have a three dimensional data for each gaze image $P_i = (x_i, y_i, z_i)^T \in R^3$, first we link them with the parameter values u_i and then the approximation curve in parametric representation has found,

$$f(u) = \begin{bmatrix} f_x(u) \\ f_y(u) \\ f_z(u) \end{bmatrix} \tag{3.4.1}$$

The parametric curve that has been found is represented by polynomial functions of degree (k).

$$f_x(u) = \sum_{r=0}^{k} a_{x_r} u^r, \quad f_y(u) = \sum_{r=0}^{k} a_{y_r} u^r \tag{3.4.2}$$

$$f_z(u) = \sum_{r=0}^{k} a_{z_r} u^r.$$

The least square method is used here to minimize the sum of squares of residuals between the curve and the given data to create curve $f(u)$, that fits as closed as possible to the points at parameter values u_i.

The Euclidean norm that used to formulate the Approximation error is:

$$s = \sum_{i=0}^{n-1} w_i[(f_x(u_i) - x_i)^2 + (f_y(u_i) - y_i)^2 + (f_z(u_i) - z_i)^2]. \tag{3.4.3}$$

Where w_i are values used to weight the importance of the points. We set the end-points weights higher than the other points to achieve exact fit to the end-points.

The python code used the centripetal parametrization method to obtain the initial parametrization values beginning with $(u_0 = 0)$ as a starting point and then calculates the next parameters accordingly.

The approximation method used here is about computing the approximation curve as polynomial functions (f_x, f_y, f_z), using the least square method, and the minimizing the approximation error $s(u_i)$ for each (u_i).

To find the minimum of function we used **golden search method** [18], that iteratively narrowing the initial interval where the minimum exists for determining a good shape parameter of higher degree polynomial equation. The numpy package "numpy.polyfit" [4], is used to implement the least square method with three polynomial degree.

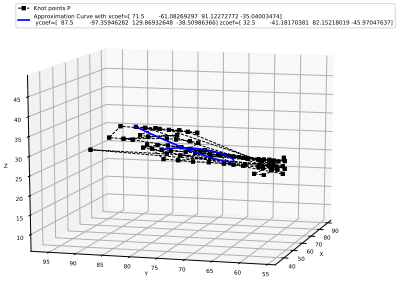
3.5 Results

Following up the previous students' pilot study on ocular opthalmology which confirmed the feasibility of using algorithm to segment the optic nerve from the surrounding tissue in different gaze angles, this study shows that the optic nerve does change shape under different gaze angles -figure (3.3). Though, the ON seems similar for each angle that is close to another angle yet it does change under various gaze angles when describe in 3D -plots (3.2).

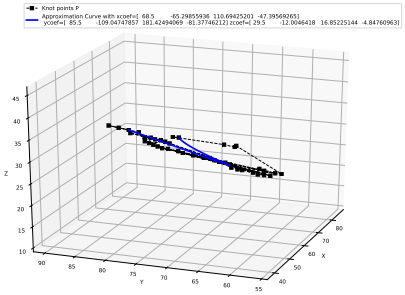
Python code has been used to determine optic nerve plots in second and third dimension for the nine different gaze images. Apparently, for most of the images that view in different planes, we got straight line in (X-Y) plane, curve or loop in (X-Z) plane, and similar shapes in (Y-Z) plane attached with two other plots that explain the Local minimization of approximation error $s(u_i)$ for each u_i and approximation error (s) for each iteration.

Third dimensional fitting plots described the optic nerve shape clearly for the nine gaze images separately which vary around X , Y and Z axes with better description of the ON shape in images (2 to 9) among the nine images where we got lines with bending or a curve for the optic nerve coordinates, figure (3.2). Having had the approximation curves plotted in 3D for all the images confirmed the possibility of describing the ON in 3D that does change under different gaze angles.

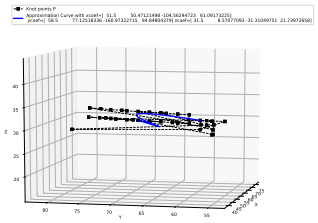
The third dimensional fitting results of the nine gaze images illustrated as following:



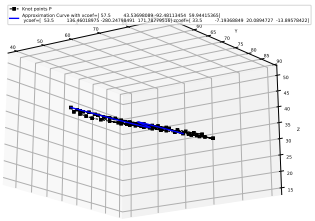
(a) Third dimensional fitting results of image(1)



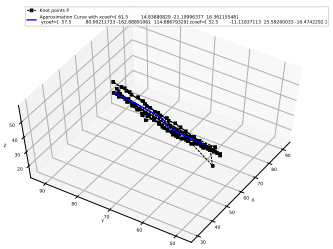
(b) Third dimensional fitting results of image(2)



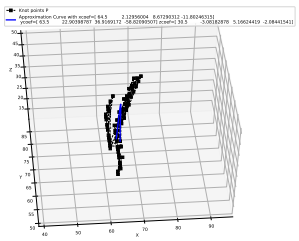
(c) Third dimensional fitting results of image(3)



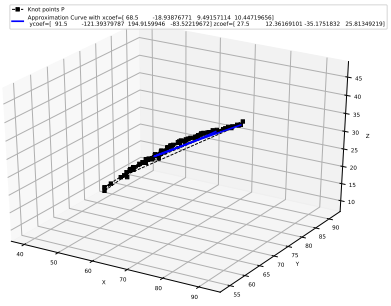
(d) Third dimensional fitting results of image(4)



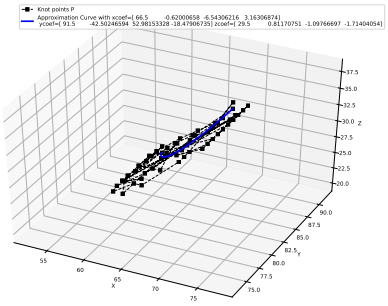
(e) Third dimensional fitting results of image(5)



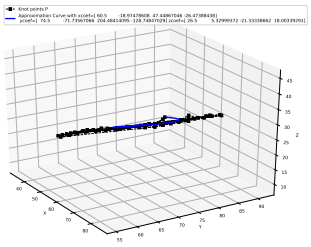
(f) Third dimensional fitting results of image(6)



(a) Third dimensional fitting results of image(7)



(b) Third dimensional fitting results of image(8)



(c) Third dimensional fitting results of image(9)

Figure 3.2: Third dimensional fitting results of image(1-9)

Though the best parametrization method is unknown, the values for initial parametrization have been obtained that fluctuates between the different images. The least-squares method used package NumPy as (numpy.polyfit) which used for fitting polynomial functions (3rd order) through calculating the coefficients for each image singly.

Table (3.1) represents the approximated curve functions coefficients for the nine gaze images:

Image number	Image coefficients coordinates
Image(1)	$f_x = [71.5,\ -61.08,\ 91.12,\ -35.04]$ $f_y = [87.5,\ -97.35,\ 129.86,\ -38.50]$ $f_z = [32.5,\ -41.18,\ 82.15,\ -45.97]$
Image(2)	$f_x = [68.5,\ -65.29,\ 110.69,\ -47.39]$ $f_y = [85.5,\ -109.04,\ 181.42,\ -81.37]$ $f_z = [29.5,\ -12.00,\ 16.85,\ -4.84]$
Image(3)	$f_x = [51.5,\ 50.47,\ -104.56,\ 61.09]$ $f_y = [58.5,\ 77.12,\ -160.97,\ 94.84]$ $f_z = [31.5,\ 8.57,\ -31.31,\ 21.73]$
Image(4)	$f_x = [57.5,\ 43.53,\ -92.48,\ 59.94]$ $f_y = [53.5,\ 136.46,\ -280.24,\ 171.78]$ $f_z = [33.5,\ -7.193,\ 20.08,\ -13.89]$
Image(5)	$f_x = [32.5,\ -11.11,\ 25.59,\ -16.47]$ $f_y = [57.5,\ 80.002,\ -162.88,\ 114.88]$ $f_z = [32.5,\ -11.11,\ 25.59,\ -16.47]$
Image(6)	$f_x = [64.5,\ 2.12,\ 8.67,\ -11.80]$ $f_y = [63.5,\ 22.90,\ 36.91,\ -58.82]$ $f_z = [30.5,\ -3.08,\ 5.16,\ -2.08]$
Image(7)	$f_x = [68.5,\ -18.93,\ 9.49,\ 10.44]$ $f_y = [91.5,\ -121.39,\ 194.91,\ -83.52]$ $f_z = [27.5,\ 12.36,\ -35.17,\ 25.81]$
Image(8)	$f_x = [66.5,\ -0.62,\ -6.54,\ 3.16]$ $f_y = [91.5,\ -42.50,\ 52.98,\ -18.47]$ $f_z = [29.5,\ 0.81,\ -1.09,\ -1.71]$
Image(9)	$f_x = [60.5,\ -18.97,\ 47.44,\ -26.47]$ $f_y = [74.5,\ -71.73,\ 204.48,\ -128.74]$ $f_z = [26.5,\ 5.32,\ -21.33,\ 18.00]$

Table 3.1: 3D coefficient coordinates for the nine gaze images

The nine images gaze angles have been determined (Figure 3.3) by calculating the relative gaze angles, using the angel between (ON center in eye) and the (lens) with respect to the scanner direction [left right, (X-Y) direction , up down, (Y-Z) direction]. Additionally, the angles calculated through using **CalculateAngle** module that calculates the angle between two vectors in both the center of the optic nerve and the eye ball through MeVisLab Software.

Table (3.2) illustrates the gaze angles for the nine images in both (X-Y) and (Y-Z) directions.

Image number	Gaze angle (X-Y)-direction	Gaze angle (Y-Z)-direction
Image (1)	25 °	18 °
Image (2)	12 °	21 °
Image (3)	15 °	22 °
Image (4)	18 °	23 °
Image (5)	17 °	16 °
Image (6)	13 °	17 °
Image (7)	15 °	14 °
Image (8)	16 °	14 °
Image (9)	14 °	16 °

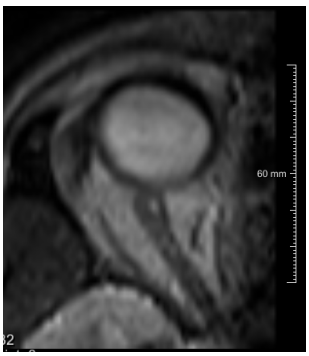
Table 3.2: Gaze angles for the nine gaze images

The images gaze angles showed that there is no slight differences between the distinctive nine images angles yet the optic nerve shape does change under the different gaze angle (Figure (3.3)) .

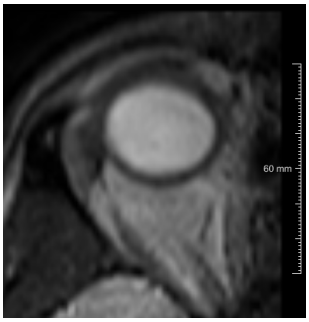
In figure (3.3) it is clear that the direction of the optic nerve changes when the eye ball moves in different directions under various gaze angles, and this latter result confirmed what we got previously in 3D plots results (3.2). The gaze angles collected for further interpolation between them to find the optimal gaze angle which will be used to improve the proton therapy planing for Uveal melanoma patients.

The figure (3.3) shows the images gaze angles in both (X-Y) and (Y-Z) directions for the nine DICOM images.

(a) Gaze angle image(1)
XY: 25 °
YZ: 18 °



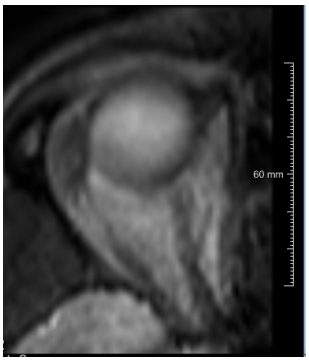
(b) Gaze angle image(3).
XY: 15 °
YZ: 22 °



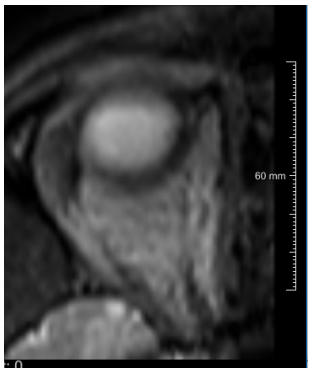
(c) Gaze angle image(2)
XY: 12 °
YZ: 21 °

(d) Gaze angle image(4)
XY:18 °
YZ: 23 °

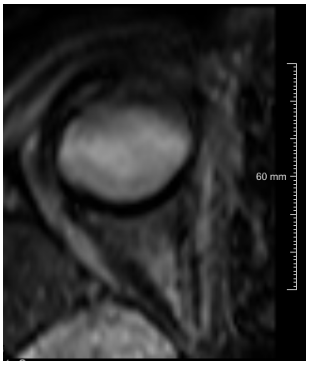
(e) Gaze angle image(5)
XY: 17 °
YZ: 16 °



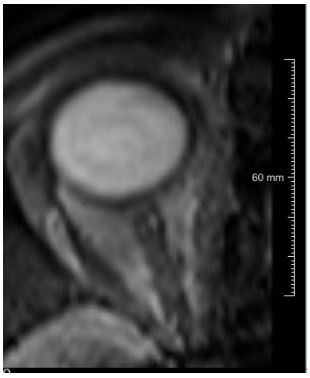
(f) Gaze angle image(6)
XY:13 °
YZ: 17 °



(g) Gaze angle image(7)
XY: 15 °
YZ: 14 °



(h) Gaze angle image(8)
XY:16 °
YZ: 14 °



(i) Gaze angle image(9)
XY: 14 °
YZ: 16 °

Figure 3.3: Gaze angles

3.6 Discussion

A healthy volunteers were examined to collect the medical data (eye images) in three dimension shape, using High field ocular (MRI). The resulting data were manually segmented and processed to describe the optic nerve shape. Through this study, we confirmed that the full-scale study can be conducted successfully depending on the two facts we obtained which are; the feasibility of using segmentation algorithm to investigate the optic nerve shape, and describe the optic nerve changes under different gaze angles (2.3) by displaying it in three dimensions.

The previous studies within ocular opthalmology topic verified that the accuracy of the segmentation by manually selecting the optic nerve and comparing it to the algorithm is essential which can benefit patient sight outcomes. The segmented MRI images showed ,in this current study, a clear difference in shape of the optic nerve under different gaze angles when plotted in 3D -figure (3.2). The calculated gaze angles in two vision directions (up down, left right)-figure (3.3) will be used to find out the optimal gaze angle that will add to the future possibility of optimizing the proton therapy planning for uveal melanoma patients by developing a segmentation algorithm in 3T MRI machine.

To sum up, the MRI images displayed in 3D showed the optic nerve changes under different gaze angles. Studying the optic nerve shape in the therapy planning may improve the accuracy of the prediction of the loss of sight and adapting gaze angles will minimize the dose to the optic nerve apart of the surrounding healthy tissue to improve the proton therapy planning for uveal melanoma patients.

4. Conclusion

The previous study of ocular opthalmology confirmed the algorithm validity to perform optic nerve segmentation in different gaze-angles and therefore the patients can be able to remain a better vision if the gaze-angle is further interpolated and optimized which would be the future pilot study that will add up to the ultimate project objective which meant to improve the proton therapy planning for uveal melanoma patients .

The main study is feasible. Through this essay we confirmed that the optic nerve shape should be taken into consideration when using the therapy planning model as it did change under various gaze angles when described in 3D. The images gaze angles have been calculated for further interpolation to find the optimal gaze angle to optimize the proton beam therapy planning.

4.1 Future Perspective

The general objective of the study is to investigate the optic nerve shape changes under different gaze angles to improve the proton therapy planning for uveal melanoma patients by developing a segmentation algorithm in 3T MRI, with the confirmed facts of the algorithm feasibility to perform segmentation of the optic nerve in nine various gaze angles optic nerve shape changes under different gaze angles we obtained with a future recommendations to interpolate between the different gaze angles to determine the optimal gaze-angle which can enhance the proton therapy planning for uveal melanoma patients.

The gaze angles interpolation along with the optic nerve shape description are needed in order to improve the accuracy of the prediction of the sight loss and to minimize the dose to the optic nerve, therefore the proton therapy treatment will be enhanced for uveal melanoma patients

Acknowledgements

This work is dedicated to Myriam Jaarsma for her supervision and dedication alongside with Dr. Jedrek Burakiewicz and Dr.Nyawo Pelerine for their unlimited supports and assistance as well as Dr.Jan-Willem M. Beenakker. Special thanks to my two families back there in sudan and at AIMS-cameroon institute.

References

[1] Jan-Willem M Beenakker, Teresa A Ferreira, Karina P Soemarwoto, Stijn W Genders, Wouter M Teeuwisse, Andrew G Webb, and Gregorius PM Luyten. Clinical evaluation of ultra-high-field mri for three-dimensional visualisation of tumour size in uveal melanoma patients, with direct relevance to treatment planning. *Magnetic Resonance Materials in Physics, Biology and Medicine*, 29(3):571–577, 2016.

[2] Bertil Damato, Andrzej Kacperek, Doug Errington, and Heinrich Heimann. Proton beam radiotherapy of uveal melanoma. *Saudi Journal of Ophthalmology*, 27(3):151–157, 2013.

[3] ANDRE De Troyer, SAMUEL Borenstein, and RENE Cordier. Analysis of lung volume restriction in patients with respiratory muscle weakness. *Thorax*, 35(8):603–610, 1980.

[4] Nolan J Dyck and Anthony G Straatman. A new approach to digital generation of spherical void phase porous media microstructures. *International Journal of Heat and Mass Transfer*, 81:470–477, 2015.

[5] Jan Egger, Markus Gall, Jürgen Wallner, Pedro Boechat, Alexander Hann, Xing Li, Xiaojun Chen, and Dieter Schmalstieg. Htc vive mevislab integration via openvr for medical applications. *PloS one*, 12(3):e0173972, 2017.

[6] Matthias Egger, P Juni, C Bartlett, F Holenstein, J Sterne, et al. How important are comprehensive literature searches and the assessment of trial quality in systematic reviews? empirical study. *Health Technol Assess*, 7(1):1–76, 2003.

[7] Armin Gruen and Devrim Akca. Least squares 3d surface and curve matching. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(3):151–174, 2005.

[8] Frank Heckel, Michael Schwier, and Heinz-Otto Peitgen. Object-oriented application development with mevislab and python. *GI Jahrestagung*, 154:1338–51, 2009.

[9] WP Levin, H Kooy, JS Loeffler, and TF DeLaney. Proton beam therapy. *British journal of Cancer*, 93(8):849, 2005.

[10] EC Levy. Complex-curve fitting. *IRE transactions on automatic control*, (1):37–43, 1959.

[11] Jens Lindegaard, Peter Isager, Jan Ulrik Prause, and Steffen Heegaard. Optic nerve invasion of uveal melanoma: clinical characteristics and metastatic pattern. *Investigative ophthalmology & visual science*, 47(8):3268–3275, 2006.

[12] MultiMedia LLC. MS Windows NT kernel description, 1999.

[13] John E Munzenrider and Norbert J Liebsch. Proton therapy for tumors of the skull base. *Strahlentherapie und Onkologie*, 175(2):57–63, 1999.

[14] WH Press, BP Flannery, SA Teukolsky, and WT Vetterling. Numerical recipes in c, 490 pp, 1988.

[15] John O Rawlings, Sastry G Pantula, and David A Dickey. *Applied regression analysis: a research tool*. Springer Science & Business Media, 2001.

[16] JM SATANTON and Pearson Galton. the peas: A brief history of linear regression for statistics instructors. *Journal of Statistical Education*, 9(3), 2001.

[17] Russell W Stineman. A consistently well-behaved method of interpolation. *Creative Computing*, 1980.

[18] CH Tsai, Joseph Kolibal, and Ming Li. The golden section search algorithm for finding a good shape parameter for meshless collocation methods. *Engineering Analysis with Boundary Elements*, 34(8):738–746, 2010.

[19] Britta Weber, Katherine Paton, Roy Ma, and Tom Pickles. Outcomes of proton beam radiotherapy for large non-peripapillary choroidal and ciliary body melanoma at triumf and the bc cancer agency. *Ocular oncology and pathology*, 2(1):29–35, 2016.

[20] Joep Wezel, Anders Garpebring, Andrew G Webb, Matthias JP van Osch, and Jan-Willem M Beenakker. Automated eye blink detection and correction method for clinical mr eye imaging. *Magnetic resonance in medicine*, 78(1):165–171, 2017.

[21] Sunil Kumar Yadav, Ella Maria Kadas, Seyedamirhosein Motamedi, Konrad Polthier, Frank Haußer, Kay Gawlik, Friedemann Paul, and Alexander Brandt. Optic nerve head three-dimensional shape analysis. *Journal of biomedical optics*, 23(10):106004, 2018.

[22] Derek York. Least-squares fitting of a straight line. *Canadian Journal of Physics*, 44(5):1079–1086, 1966.