3D Segmentation & Measurement of Macular Holes

Amar V. Nasrulloh

A thesis presented for the degree of
Doctor of Philosophy at Durham University

Durham University

Department of Computer Science
Durham University
United Kingdom

May 14, 2020

3D Segmentation & Measurement of Macular Holes

Amar V. Nasrulloh

Submitted for the degree of Doctor of Philosophy
August 2019

Macular holes are blinding conditions where a hole develops in the central part of retina, resulting in reduced central vision. The prognosis and treatment options are related to a number of variables including the macular hole size and shape. In this work we introduce a method to segment and measure macular holes in three-dimensional (3D) data.

High-resolution spectral domain optical coherence tomography (SD-OCT) allows precise imaging of the macular hole geometry in three dimensions, but the measurement of these by human observers is time consuming and prone to high inter- and intra-observer variability, being characteristically measured in 2D rather than 3D. This work introduces several novel techniques to automatically retrieve accurate 3D measurements of the macular hole, including surface area, base area, base diameter, top area, top diameter, height, and minimum diameter. Specifically, it is introducing a multi-scale 3D level set segmentation approach based on a state-of-the-art level set method, and introducing novel curvature-based cutting and 3D measurement procedures. The algorithm is fully automatic, and we validate the extracted measurements both qualitatively and quantitatively, where the results show the method to be robust across a variety of scenarios.

A segmentation software package is presented for targeting medical and biological applications, with a high level of visual feedback and several usability enhancements over existing packages. Specifically, it is providing a substantially faster graphics processing unit (GPU) implementation of the local Gaussian distribution fitting (LGDF) energy model, which can segment inhomogeneous objects with poorly defined boundaries as often encountered in biomedical images. It also provides interactive brushes to guide the segmentation process in a semi-automated framework. The speed of implementation allows us to visualise the active surface in real-time with a built-in ray tracer, where users may halt evolution at any timestep to correct implausible segmentation by painting new blocking regions or new seeds. Quantitative and qualitative validation is presented, demonstrating the practical efficacy of the interactive elements for a variety of real-world datasets.

The size of macular holes is known to be one of the strongest predictors of surgical success both anatomically and functionally. Furthermore, it is used to guide the choice of treatment, the optimum surgical approach and to predict outcome. Our automated 3D image segmentation algorithm has extracted 3D shape-based macular hole measurements and described the dimensions and morphology. Our approach is able to robustly and accurately measure macular hole dimensions.

This thesis is considered as a significant contribution for clinical applications particularly in the field of macular hole segmentation and shape analysis.

Declaration

The work in this thesis is based on research carried out within the Bioimage Informatics research laboratory, part of the Innovative Computing Group at the Department of Computer Science, Durham University, UK. No part of this thesis has been submitted elsewhere for any other degree or qualification, and it is all the author's work unless referenced to the contrary below.

Note on Publications Included in this Thesis

At the time of submission, three chapters of this thesis had been submitted for publication or published in journals.

Chapter 2 Nasrulloh, A. V., Willcocks, C. G., Jackson, P. T. G., Geenen, C., Habib, M. S., Steel, D. H. W. & Obara, B. Multi-scale segmentation and surface fitting for measuring 3D macular holes. *IEEE Transactions on Medical Imaging* **37,** 580–589 (2018)

Chapter 3 Willcocks, C. G., Jackson, P. T., Nelson, C. J., Nasrulloh, A. V. & Obara, B. Interactive GPU active contours for segmenting inhomogenous objects. *Journal of Real-Time Image Processing,* 1–14 (2017)

Chapter 4 Chen, Y., Nasrulloh, A. V., Wilson, I., Geenen, C., Habib, M. S., Obara, B. & Steel, D. H. W. Macular hole morphology and measurement using an automated three dimensional image segmentation algorithm. *BMJ Open Ophthalmology.* In press (2020)

These chapters are presented as published, although referencing and notation has been altered and cross-referencing added for consistency throughout this thesis. Some stylistic changes have been made for consistency.

For consistency additional results chapters (Chapters 2–4) have been written in the style of papers/proceedings and can be considered self-contained (*c.f.* Chapters 2–4).

Chapter 2 were completed in partnership with Chris G. Willcocks, Philip T.G. Jackson, C. Geenen, M.S. Habib and D.H.W. Steel. Amar led research implementation, testing and validation, wrote the majority of the paper. He collected all the results, contributed towards the multi-scale idea. Chris helped formulate the initial idea and contributed towards one of the measurements. He helped write significant part of the paper and helped with its implementation. Phil helped towards the implementation, helped formulate the mathematical description and helped write part of the paper. M.S. Habib and D.H.W. Steel provided datasets, helped write part of the paper and in revising the paper. C. Geenen did manual segmentation used as a ground truth.

Chapter 3 were completed in partnership with Chris G. Willcocks, Philip T.G. Jackson and Carl J. Nelson. Amar helped in conducting the experiment, pre-processing and acquiring data and in revising the paper. Chris led the research, did the majority of the GPU implementation, the experiments and the figures. Wrote half of the paper and formulated the description of GPU kernels. Phil wrote the other half of paper, helped with the implementation and formulated the mathematical description and helped with organising GPU kernels. Chas helped revise the paper, made the supplementary video, and contributed in discussions and the method.

Chapter 4 was completed in partnership with Yunzi Chen, Ian Wilson, Caspar Geenen, Maged S. Habib and David H. W. Steel. Amar helped conducting the experiment, research implementation and analysed and collected all measurement results from the automated segmentation analysis. David Steel led the research, collected the data, analysed results and wrote the paper. Yunzi Chen and Caspar Geenen collected datasets and did the manual measurements. Maged Habib provided datasets and reviewed the manuscript critically. Ian Wilson performed the statistical analysis and produced the diagrams and graphs.

Acknowledgements

Amar V. Nasrulloh was sponsored by the Indonesian Endowment Fund for Education, Indonesia (LPDP Indonesia) No.20150422012873. I would like to thank my supervisor, Dr Boguslaw Obara, for the patient guidance and the never ending support.

Chapter 2 A.V. Nasrulloh, C.G. Willcocks, P.T.G. Jackson and B. Obara are with Department of Computer Science, Durham University, Durham, UK. C. Geenen and M.S. Habib are with Sunderland Eye Infirmary, Queen Alexandra Road, Sunderland, UK. D.H.W. Steel is with Sunderland Eye Infirmary, Queen Alexandra Road, Sunderland and Institute of Genetic Medicine, Newcastle University, Newcastle Upon Tyne, UK.

Chapter 3 The authors are with Department of Computer Science, Durham University, Durham, UK. The authors offer warm thanks to NVIDIA for providing a GTX TITAN X for this research. Table 5 1a shows fixed HaCaT human cell culture cells stained with SiR-Actin (Spirochrome) RED, rat anti-tubulin antibody/secondary anti-rat Alexa488 antibody GREEN and DNA DAPI BLUE. The cells are imaged with a Zeiss 880 Airyscan LSM confocal microscope, prepared, and imaged by Miss Bethany Cole, Miss Joanne Robson and Dr Tim Hawkins. Durham Centre for Bioimaging Technology, Department of Biosciences, Durham University, UK.

Chapter 4 A.V. Nasrulloh and B. Obara are with Department of Computer Science Durham University, Durham, UK. Y. Chen, C. Geenen and M.S. Habib are with Sunderland Eye Infirmary, Queen Alexandra Road, Sunderland, UK. I. Wilson is with Institute of Genetic Medicine, Newcastle University, Newcastle Upon Tyne, UK. D.H.W. Steel is with Sunderland Eye Infirmary, Queen Alexandra Road, Sunderland and Institute of Genetic Medicine, Newcastle University, Newcastle Upon Tyne, UK.

lpdp
lembaga pengelola dana pendidikan

SUNDERLAND EYE INFIRMARY
FOUNDED 1836

Dedication

To my parents.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis is exploring a method to segment and measure macular holes in three-dimensional space (3D). The robustness of this method is demonstrated by the variety of hole shapes that it is able to segment and measure automatically. We also implement our method on graphics processing unit (GPU) to increase procesing speed. Further development of the measurement method for supporting macular hole morphology in clinical analysis is also introduced.

Firstly, we introduce a multi-scale 3D level set segmentation approach based on a state-of-the-art level set method (local Gaussian distribution fitting energy model), and novel curvature-based cutting and 3D measurement procedures. Secondly, we provide a substantially faster GPU implementation of the local Gaussian distribution fitting (LGDF) energy model, which can segment inhomogeneous objects with poorly defined boundaries as often encountered in biomedical images such as 3D OCT imagery of macular holes. Finally, we present an approach for further clinical application of 3D measurement calculation and shape analysis.

This chapter will introduce the technology that enables acquisition of 3D imagery of the retina and the importance of image analysis for clinical applications. The structure of this thesis will also be outlined.

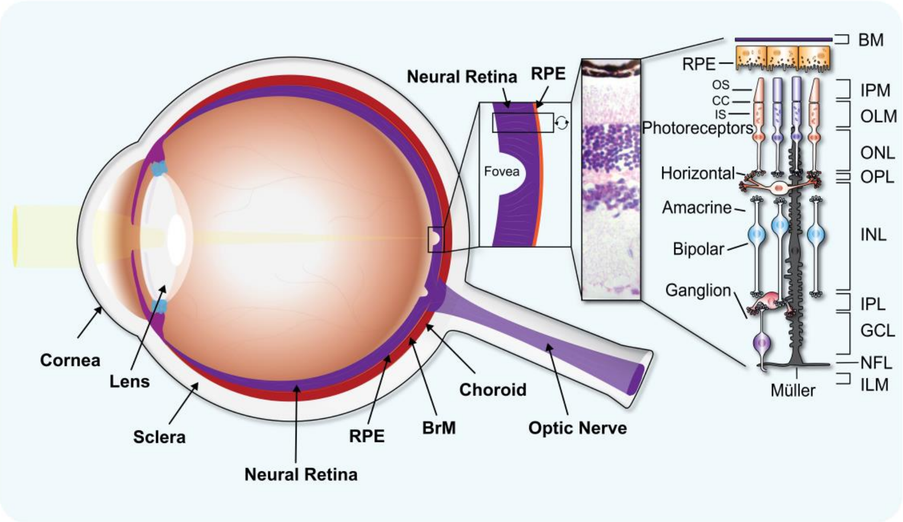


Figure 1.1: Structure of human retina (with permission of David Steel in [4]).

1.1 Retinal Imaging and Retinal Image Analysis

Human retina, part of the human eye, converts energy as light into electrical signals and transmitted into the brain but also performs the initial processing of information about the visual scene [4]. Figure 1.1 shows retinal structure and its parts. The somas (cell bodies) of the photoreceptors occupy the outer nuclear layer (ONL). In the inner nuclear layer (INL) can be found the somas of the Horizontal cells, Amacrine cells, Bipolar cells, and Müller glial cells. Meanwhile the somas of Retinal ganglion cells (RGCs) lie within the ganglion cell layer (GCL). The processes and synaptic terminals of photoreceptors, Horizontal cells, and Bipolar cells lie within the outer plexiform layer (OPL), whilst the inner plexiform layer (IPL) contains the processes and terminals of Amacrine cells, Bipolar cells, and RGCs. The processes of Müller glial cells span from the internal limiting membrane (ILM) of the retina to the external limiting membrane at approximately the level of the inner segments (IS) of the photoreceptors and the nerve fibre layer (NFL) comprising RGC axons, converge to the optic nerve which then travels to the brain. (Connecting cilia (CC), Bruchs membrane (BrM), Outer segment (OS), Inner segment (IS), Retinal pigment epithelium (RPE)) [4].

Retinal imaging has risen highly since the invention of the ophthalmoscope more than 150 years ago [5, 6]. Since the ophthalmoscope needs to be close to the eye of a patient, the recorded

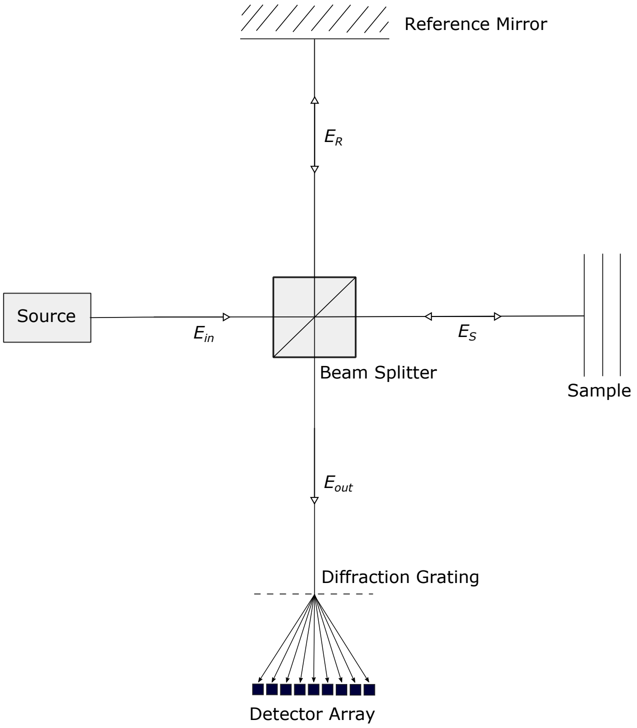


Figure 1.2: Schematic diagram of SD-OCT system (adapted from [8]).

image of the retina gave birth to retinal imaging technology from digitized slides of fluorescein angiograms and fundus camera that mainly provide two-dimensional (2D) imagery to optical coherence tomography (OCT) for 2D and 3D images [7].

OCT technology is based on an optical physics interferometric technique whereby interference from the splitting of light (to reference and sample) is re-combined again and then extracted by photodetector [8]. The depth (axial) image resolution is determined by the echo time coherence length of the light source and OCT retinal imaging, first demonstrated in 1991, has become a common noninvasive procedure in ophthalmology [9]. With the ability of axial scan, it enabled 3D scanning of the retinal tissue and in particularly the macula (central area of the retinal for central vision and colour differentiation [10]). OCT has provided the noninvasive cross sectional imaging in the peripapillary area of the retina [11], detecting and monitoring a variety of macular diseases, including macular holes [12].

Next to OCT technology is spectral-domain OCT (SD-OCT), which has become the standard of

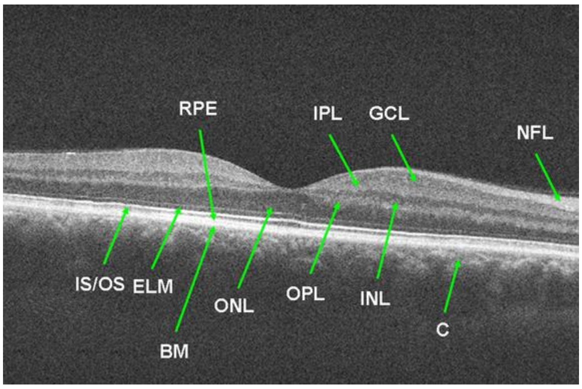


Figure 1.3: A normal human fovea generates by SDOCT. (with permission of David Steel in [4]).

care [7], allowing 3D reconstructions for better visualisation of specific tissue structures [13, 14] and better diagnosis and treatment of ocular conditions [15]. Physics of SD-OCT is almost the same as OCT. The input electric field (E_{in}) passing through the beam splitter and then split into E_r (the reference electric field) and E_s (the sample electric field). Those E_r and E_s travel back through beam splitter again and the resulting E_{out} travels into the detection array [8] (see figure 1.2).

The differences between OCT and SD-OCT are that for SD-OCT machine, the reference path are fixed and the photodetector is replaced with a spectrometer. The spectrometer is a detector array capturing the intensity spectrum (component frequencies) and then Fourier transformed into the time domain to reconstruct the depth resolved sample optical structure. This principle has the advantage that no moving parts are required to obtain axial scans such as in OCT machine [8].

SD-OCT images describe the presence of normal macula and each of the three pathologies (macular hole, macular edema and Age-related macular degeneration) separately [16]. In macular holes, SD-OCT is useful for size classification [17, 18] as well as for quantifying macular hole surgery outcomes. [19]. SD-OCT also enables the extraction of constructive 3D textural information for clinical applications [20]. The horizontal SD-OCT image of a normal fovea (Figure 1.3) illustrates retinal layer [4].

Image processing has been an essential tool in retinal imaging and has contributed to the diagnosis of the human retina for around 40 years [7, 21]. Modern retinal imaging technology stores

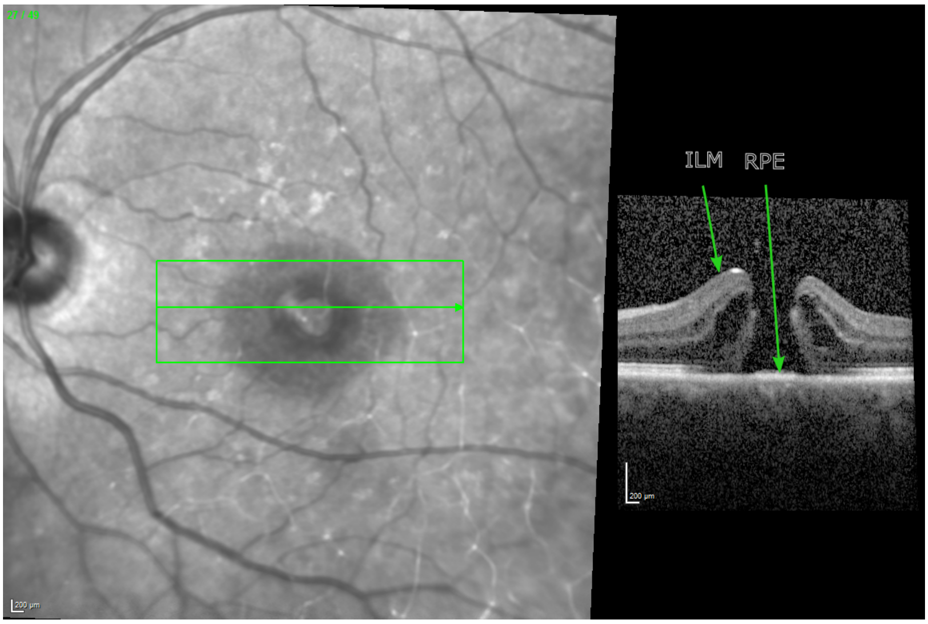


Figure 1.4: SD-OCT image of the fovea in the retina (left) and the form of the macular hole between ILM layer and RPE layer (right). Scale bar: $x = 11.58$ $[\mu m]$ and $y = 3.87$ $[\mu m]$.

photographed images as digital images that keep the same quality over time. Thus, it can be processed using image processing techniques such as enhancement and restoration to improve retinal images and segmentation to localise regions of interest for further information extraction [22, 23].

In this thesis, images of macular hole cases are used from spectral domain optical coherence tomography (SD-OCT).

1.2 Macular Hole Segmentation and Measurement

A macular hole is a hole that develops in the central part of the retina, and is most commonly caused by age related vitreous traction on the central fovea [24, 25] (see Figure 1.4).

Early in retinal analysis, the rise of cystoid in the retina, first described in 1865, has become a focus of attention since then, including the formation of macular holes [26]. As retinal imaging generates digital retinal images including macular hole images, development of image processing technique to segment macular holes has risen prominently. Macular hole segmentation was proposed in 2D based techniques to segment and separate the hole from the retina/vitreous boundary [27–30]. Furthermore, [30] proposed a technique to extract shape and volume of macular holes through slice-by-slice segmentation.

The OCT based for the measurement of macular hole minimum diameter [31] shows correlation with postoperative visual acuity. In macular hole surgery, base diameter and minimum diameter have been shown to be valuable for prediction [32] and classification of macular holes [33]. A selected single slice-based approach or 2D measurement are common techniques for calculating macular hole diameter [34–37]. Meanwhile, measurement of a macular hole will be more accurate if 3D shape is considered.

To accelerate 3D segmentation, graphics processing units (GPUs) provide energy efficient parallel computing and enable real-time interactive segmentation for larger 2D or 3D datasets [38, 39]. The GPU based segmentation can be applied to medical images including macular holes.

In this thesis, new approaches are introduced for 3D segmentation, for separating macular holes from the vitreous body and measuring 3D macular holes (Chapter 2 (*p.* 9)), and also acceleration of 3D segmentation with GPU (Chapter 3 (*p.* 35)).

1.3 Macular Hole Image Analysis

Macular morphology changes continuously before and after surgery. We need to monitor macular contour improvement with time. Minimum and basal (base) diameter measurements are performed for classifying macular hole type before surgery based on a 2D selected image from SD-OCT [40] or for classifying macular hole size and morphological features [41].

A variety of the size of macular holes has been used to guide the choice of treatment, the optimum surgical approach and to predict outcome [42–47]. Using a single 2D slice image, all these size measurements have typically been measured by a human grader using calipers [As the real form of macular hole morphology is a 3D shape and dimensions of macular holes].

In this thesis, a 3D measurement calculation and shape analysis approach (Chapter 4 (*p.* 60)) is introduced for an automated 3D segmentation of macular holes (Chapter 2 (*p.* 9)).

1.4 Image Segmentation

Image segmentation subdivides an image and isolates its object of interest [48]. As medical images play an important part in diagnosis and treatment, medical image segmentation is becoming a crucial element in medical-image analysis and extracting clinical-related information for the physicians for decision making [49, 50].

3D retinal image acquired from SD-OCT provide a 3D view of macular hole. The macular holes segmentation aims holes that occur in the macula as regions of interest. Therefore, a cropping procedure is applied before the segmentation process. Active contours and level set approaches have been used to segment OCT data [51]. The level set method was originally developed for curves in 2D space and surfaces in 3D space [52], so it is compatible with the segmentation of 3D SD-OCT of macular hole images.

The recent method of segmentation applied in image processing is deep learning (a subset of machine learning), where millions of parameters of deeply layered convolutions are learnt using backpropagation [53]. Convolutional neural network approach in deep learning is able to perform image recognition and classification in OCT-based of ophthalmology (mainly in maculae edema, age-related macular degeneration, retinal disease management, glaucoma and corneal diseases) [54]. Deep convolutional neural networks models are capable of learning abstract features in the data; however, their current reliance on such large and strong training datasets as a reference standard in order to produce accurate outcomes makes them limited for a number of applications.

In this thesis, an approach to 3D segmentation is described for macular holes captured using SD-OCT (Chapter 2 (*p.* 9)). This pipeline includes a semi-automated segmentation stage using state-of-the-art graphical processing unit (GPU) level-set technologies (Chapter 3 (*p.* 35)).

1.5 Overview of Thesis and Research Contributions

This thesis is divided as follows: the first section is introducing a multi-scale 3D level set segmentation approach based on a state-of-the-art level set method, as well as novel curvature-based cutting and 3D measurement procedures. Next we describe our segmentation software package targeting medical and biological applications, with a high level of visual feedback and several usability enhancements over existing packages, providing interactive brushes to guide the segmentation process in a semi-automated framework. Finally, we introduce an application of the automated 3D segmentation and measurement to describe the 3D morphology of macular holes. The chapters are outlined as follows:

Chapter 2 Accurate 3D segmentation and measurement of macular holes

Chapter 3 A novel GPU implementation of a specific active contour model: the local Gaussian distribution fitting (LGDF) energy for a variety of real-world datasets including a macular hole dataset

Chapter 4 3D measurement and shape analysis of macular holes

Chapter 2

Multi-scale Segmentation and Surface Fitting for Measuring 3D Macular Holes

Prologue

This chapter introduces a multi-scale 3D level set segmentation approach based on the local Gaussian distribution fitting (LGDF) method, and introduces novel curvature-based cutting and 3D measurement procedures. This measurement technique automatically retrieves accurate 3D measurements of the macular hole.

Declaration: This chapter is based on the following publication: Nasrulloh, A. V., Willcocks, C. G., Jackson, P. T. G., Geenen, C., Habib, M. S., Steel, D. H. W. & Obara, B. Multi-scale segmentation and surface fitting for measuring 3D macular holes. *IEEE Transactions on Medical Imaging* **37,** 580–589 (2018). This chapter is presented as published, although referencing and notation has been altered and cross-referencing added for consistency throughout this thesis. Some stylistic changes have been made for consistency.

2.1 Introduction

A macular hole is a hole that develops in the central part of the retina, and is most commonly caused by age related vitreous traction on the central fovea [24, 25]. The condition affects approximately 2 in every 1000 individuals over the age of 40 [55], and can have a devastating impact on the quality of life and on an individual's independence [25]. Surgery offers a 90% likelihood of closing the hole but the improvement in vision depends on the duration of the hole, and is related to a number of variables including macular hole size and shape [47, 56]. Recently, [30] have proposed the first approach to segment and measure the macular hole; however, their approach acquires the measurements of the 3D macular hole by combining individual 2D graph-cuts segmentations of slices in the 3D image, rather than considering the overall 3D geometry.

Our approach includes a fully 3D segmentation algorithm using a state-of-the-art level set method based on the local Gaussian distribution fitting (LGDF) energy functional [57]. However, despite the high computational expense with level sets, we achieve fast convergence without relying on high-end hardware acceleration. This is accomplished by employing a 3D multi-scale approach that exploits the fact that the macular holes are considered large objects within the 3D image, processing initial updates at lower spatial resolutions.

This is followed by a novel curvature-based surface cutting procedure, which separates the macular hole from its background, allowing for fully-automatic measurement of the shape and volume. The method is shown to be stable to the various 3D input images of different macular holes without requiring retuning of the parameters, and is shown to be more accurate than existing graph cut segmentation approaches.

2.2 Contributions

The objective of this paper is to develop an automatic approach to efficiently extract precise and robust measurements of 3D macular holes. To this purpose, we have developed a level set segmentation approach that improves the efficiency of the state-of-the-art LGDF energy by considering multiple scales. Further, we have introduced a novel curvature-based surface cutting procedure to separate the segmented hole from its background, and proposed novel procedures for automatically calculating the desired measurements robustly.

Specifically, we have:

- Composed an automatic pipeline for measuring 3D macular holes.
- Introduced a 3D multi-scale active surface which is 61 times faster than the original LGDF implementation.
- Introduced an automatic and novel curvature-based surface cutting procedure to separate the 3D macular hole from the vitreous body.
- Introduced automatic and novel procedures for capturing specific macular hole measurements, based on our robust centerline definition.
- Provided quantitative and qualitative validation of the algorithm's robustness and performance accross a variety of different 3D macular hole images.

2.3 Related Work

2.3.1 Retinal Imaging

The analysis in retinal disease is a large multidisciplinary research area with collaboration from researchers and clinicians [58, 59]. A comprehensive review by [7] shows that analysis in retinal disease has led to significant advances in preventing blindness and visual loss, using 2D digital fundus photography and 3D optical coherence tomography (OCT). The macula refers to the central part of the retina with the fovea, a specialized part of it with high photoreceptor density, sited at its centre. It is responsible for fine detailed vision including reading and facial recognition. OCT is considered a powerful technique for imaging macular disease including macular edema and macular holes [60]. In contrast, compared to macular edema, research into automatic macular hole analysis is very limited.

2.3.2 Macular Hole Imaging

Optical coherence tomography allows clinicians to characterize the shape and volume of macular holes [61], leading to new insights into the pathogenesis of macular hole formation [62]. Spectral domain optical coherence tomography (SD-OCT) provides higher resolution images with reduced artefacts compared to time domain OCT [25]. Recently, [27] propose a length-adaptive graph search metric that accurately segments the retina/vitreous boundary of a 2D OCT image, whereas [28] use auto-thresholding in ImageJ [29] and measure the binary segmentation of the forward-facing view of a macular hole.

To calculate the volume of a macular hole, [63] propose fitting a truncated cone to a 2D view of the macular hole, which fails to capture irregularities and depth information. Alternative 3D methods apply 2D segmentation algorithms to each slice in the 3D geometry [30] and then accumulate the results to provide analysis of macular hole's 3D shape and volume; however, the slice-by-slice approach requires verification by a human operator to confirm the segmentation at each slice. The minimum diameter measurement is also calculated based on single 2D slice [30, 35], which may fail to consider profiles that are elliptical in cross-section.

Similarly, [64] distinguish normal macula and multiple macular pathologies which are macular edema, macular hole, and age-related macular degeneration from the foveal slices in retinal OCT images on a slice-by-slice basis. They employ a simple edge-based technique with a canny edge-detector, which has known limitations where edges are weak, blurred and/or broken; however, they apply a multi-scale spatial pyramid and identify local binary patterns in texture and shape encoding to efficiently infer geometric features at multiple scales and spatial resolutions.

In contrast to these approaches, 3D level set methods are able to enforce continuity along all three axes with subpixel accuracy and smooth surfaces, while remaining robust to noise. They can segment objects with weak, broken and/or distorted edges by not relying on image gradients [65].

2.3.3 Active Contours

The snakes active contour model, first introduced in [66], is widely used in the image segmentation field to deform an initial curve to lock onto edges, lines and endpoints interactively. This concept was extended by [67], who considered the contour neighbourhood as a region-based energy. Malladi et al. in [68] formulated the energy update rule using level sets, which allows a deformable implicitly defined region to surround or fill inside the object of interest. By introducing a curvature term [69], the evolving speed of the deformable region (contour or surface) can be controlled such as to prevent leaking through small gaps in the object boundary allowing for segmentation of more complex shapes.

The Mumford-Shah energy functional [70], establishes an optimal partitioning scheme to divide an image into parts, which are piecewise-smooth within segments. Early active contour models solved the Mumford-Shah under the assumption that segments had piece-wise constant intensity [65], which was later addressed by [71] who define a pixel's energy based on its local neighborhood within a predetermined scale rather than the mean intensity of an entire segment. [57] improves on this by modifying the data fitting term to consider the intensity variance, through a *maximum a posteriori probability* formulation based on Gaussian approximation of the intensity distribution, called the local Gaussian distribution fitting (LGDF) energy. The LGDF energy functional is considered a state-of-the-art in the active contour segmentation literature; however, it has currently only been applied to small 2D images due to its high computational requirements, in particular relying on multiple Gaussian convolutions per update iteration.

Further, it is difficult for standard segmentation approaches to automatically separate the macular hole from the vitreous body. Previous work individually applies a 2D graph cut algorithm to each 3D slice of the macular hole, where the graph cut boundaries are specified with morphological erosions of adjacent slices. However, this approach requires manual verification of each slice by a human operator to check the segmentation is plausible [30]. We perform a fully automatic 3D segmentation algorithm with an automatic procedure to cut the relevant volume of the hole from the background vitreous body in order to obtain the required measurements.

2.4 Materials

In this paper, we used 30 images of macular hole cases that were provided by the Sunderland Eye Infirmary, Sunderland, UK. Patients underwent spectral domain optical coherence tomography (SD-OCT) on the Heidelberg Spectralis (Heidelberg Engineering, Heidelberg, Germany) immediately preoperatively as part of routine care. A high density central horizontal scanning protocol with 30 [μm] line spacing was used in the central 10 by 15 [degrees]. All scans used a 25 automatic real time (ART) setting enabling multisampling and noise reduction over 25 images. The images were exported anonymised and in a non-compressed format. All of our input images are $200 \times 200 \times 49$ voxel sub-regions of the raw data, chosen such that the macular hole is in the centre (see Figure 2.1).

2.5 Method

We present a novel segmentation approach based on level set methods and regression analysis, for extracting specific measurements of the 3D shape and volume of the macular hole. The proposed approach consists of three main parts, discussed in Subsections 2.5.2, 2.5.4 and 2.5.5 respectively.

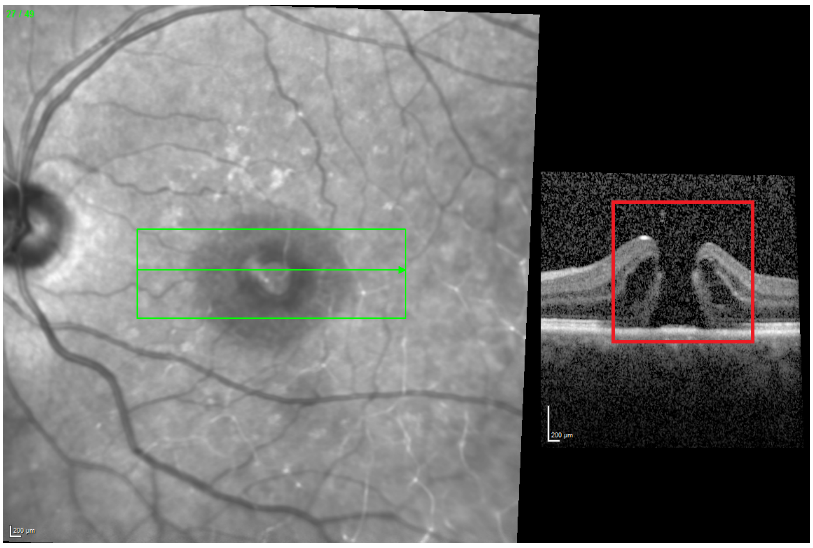


Figure 2.1: 2D view of a 3D SD-OCT image of the retina (left) and its 2D cross-section along the green arrowed line. The region of interest is marked by the red box (right). Scale bar: $x = 11.58$ $[\mu m]$ and $y = 3.87$ $[\mu m]$.

2.5.1 Overview

Level set segmentation methods, including the LGDF energy functional in [57], implicitly define the boundary of a segmentation region as the zero level set of a scalar field $\phi(\vec{x})$. That is, the interior of the region is defined as $\{\vec{x} : \phi(\vec{x}) < 0\}$, with $\phi(\vec{x}) = 0$ on the boundary. Given an input image I, ϕ is iteratively deformed using variational methods so as to minimise the energy functional $E(\phi, I)$, resulting in a smooth boundary that separates regions of different properties.

The initial value of ϕ is set automatically as a small spherical seed region $\phi(\vec{x}) = |\vec{x} - \vec{x}_0| - r$, where $\vec{x}_0$ is $(0.5, 0.6, 0.5)$ in normalized image coordinates, slightly below the midpoint of the image, so as to be close to the base of the macular hole (Figure 2.2a). This is then clamped with a binary step function using a constant value $c = 2$ outside the sphere and $c = -2$ inside the sphere. The segmentation process involves iteratively updating ϕ so as to minimise the LGDF functional $E^{\text{LGDF}}(\phi, I)$ yielding an implicit boundary between vitreous humor and retina tissues (Figure 2.2b-c).

To increase performance, we initially segment the cavity at low-resolution, then when convergence is met we progressively upscale both ϕ and I and iterate until ϕ and I are at their original resolution and fully converged (Figure 2.2b-d). We then cut the segmented cavity from its opening in a separate procedure (Figure 2.2e) before extracting the final measurements.

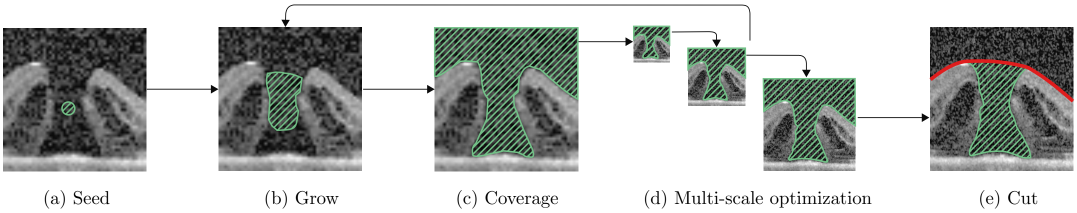


Figure 2.2: The proposed 3D multi-scale approach iteratively deforms an initial region (a) to fill (b-c) the cavity of the hole at low-resolution, and then progressively upscales (d) and reiterates until convergence. This is followed by a curvature-based cutting procedure (e).

2.5.2 3D Local Gaussian Distribution Fitting (LGDF) Energy

We adopt the LGDF energy functional in [57] (originally implemented in 2D) for a 3D active surface. The total energy $\mathcal{E}(\phi)$ is given by the summation of three terms: energy $E^{\mathrm{LGDF}}(\phi, I)$ which drives the contour to fit along salient image edges, a length term $\mathcal{L}(\phi)$ which penalizes the length of the contour to ensure smoothness, and a regularization term $\mathcal{P}(\phi)$ which keeps ϕ approximating a signed distance function to ensure numerical stability:

$$\mathcal{E}(\phi, I) = \alpha E^{\mathrm{LGDF}}(\phi, I) + \nu \mathcal{L}(\phi) + \mu \mathcal{P}(\phi)\,, \tag{2.1}$$

where α, ν, $\mu > 0$ are weighting constants. $E^{\mathrm{LGDF}}(\phi, I)$ is the sum of the energy of each voxel $E^{\mathrm{LGDF}}(\phi, I, \vec{x})$:

$$\begin{aligned} E^{\mathrm{LGDF}}(\phi, I, \vec{x}) = &- \int_{\Omega} \omega(\vec{y} - \vec{x}) \log(p_{1,\vec{x}}(I(\vec{y})) M_1(\phi(\vec{y})) \mathrm{d}\vec{y} \\ &- \int_{\Omega} \omega(\vec{y} - \vec{x}) \log(p_{2,\vec{x}}(I(\vec{y})) M_2(\phi(\vec{y})) \mathrm{d}\vec{y}\,, \end{aligned} \tag{2.2}$$

where $\omega(\vec{x} - \vec{y})$ is a Gaussian weighting function centered on $\vec{x}$, $p_{1,\vec{x}}$ and $p_{2,\vec{x}}$ are the likelihoods assigned to the pixel intensities by Gaussian models of the intensity distributions inside and outside the contour, and M_1 and M_2 are indicator functions separating the regions inside and outside the contour. This is further elaborated in [57]. $\mathcal{P}(\phi)$ penalizes the deviation of ϕ from signed distance function [71]:

$$\mathcal{P}(\phi) = \int \frac{1}{2} \left(|\nabla \phi(\vec{x})| - 1\right)^2 \mathrm{d}\vec{x}\,, \tag{2.3}$$

and $\mathcal{L}(\phi)$ penalizes the length of the contour [72] for a smooth surface:

$$\mathcal{L}(\phi) = \int |\nabla H(\phi(\vec{x}))| \mathrm{d}\vec{x}\,, \tag{2.4}$$

where $\vec{x} = [x, y, z]$ is a voxel in an image I. H is a Heaviside function which we discretize by:

$$H(x) = \frac{1}{2}\left[1 + \frac{2}{\pi}\arctan(x)\right]\,. \tag{2.5}$$

The energy $E^{\mathrm{LGDF}}(\phi, I)$ is able to segment objects with inhomogenous local intensity mean $u(\vec{x})$ and variance $\sigma(\vec{x})^2$ (Equations 22-23 in [57]) allowing for slow changes in intensity accross an object but penalizing sudden changes within it. The image force term is calculated for both inside and outside the contour $i = 1, 2$ respectively, by applying the calculus of variations [57], yielding:

$$e_i(\vec{x}) = \int_\Omega \omega(\vec{y} - \vec{x}) \left[\log(\sigma_i(\vec{y})) + \frac{(u_i(\vec{y}) - I(\vec{x}))^2}{2\sigma_i(\vec{y})^2}\right] \mathrm{d}\vec{y}\,. \tag{2.6}$$

To compute the image force term $e_1 - e_2$, we expand the brackets in Equation 2.6 to get:

$$\begin{aligned}
e_i(\vec{x}) = & \int_\Omega \omega(\vec{y} - \vec{x}) \left[\log(\sigma_i(\vec{y})) + \frac{u_i(\vec{y})^2}{2\sigma_i(\vec{y})^2}\right] \mathrm{d}\vec{y} \\
& - I(\vec{x}) \int_\Omega \omega(\vec{y} - \vec{x}) \frac{u_i(\vec{y})}{\sigma_i(\vec{y})^2} \mathrm{d}\vec{y} \\
& + I(\vec{x})^2 \int_\Omega \omega(\vec{y} - \vec{x}) \frac{1}{2\sigma_i(\vec{y})^2} \mathrm{d}\vec{y} && (2.7) \\
& = G_\sigma * \left[\log(\sigma_i(\vec{y})) + \frac{u_i(\vec{y})^2}{2\sigma_i(\vec{y})^2}\right] \\
& - I(\vec{x}) \left[G_\sigma * \frac{u_i(\vec{y})}{\sigma_i(\vec{y})^2}\right] + I(\vec{x})^2 \left[G_\sigma * \frac{1}{2\sigma_i(\vec{y})^2}\right]\,, && (2.8)
\end{aligned}$$

where $G_\sigma *$ denotes convolution with a Gaussian kernel of standard deviation σ.

This local energy is applied locally with the delta function $\delta(x)$ (derivative of Equation 2.5) $-\delta(\phi)(e_1 - \lambda e_2)$ with a weighting parameter $\lambda > 1$ to give preference to external force e_2 and hence the contour is preferable to grow to fill the hole cavity. The functional can then be minimized by solving the gradient descent flow equation [57] which yields the following update rule:

$$\frac{\partial \phi}{\partial t} = -\alpha \delta(\phi)(e_1 - \lambda e_2) + \nu \delta(\phi)\kappa + \mu \left(\nabla^2 \phi - \kappa\right) \,, \tag{2.9}$$

where κ is the curvature [69]:

$$\kappa = \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) \,. \tag{2.10}$$

To segment the image, we iteratively update the level set function ϕ according to Equation 2.9. However, this is computationally expensive, especially in the case of 3D OCT images requiring tens of thousands of iterations before convergence.

2.5.3 3D Multi-scale LGDF

We improve performance by initially segmenting the hole cavity at a small scale, by downsampling I and ϕ (Figure 2.3a). Solving Equation 2.9 at this small scale converges much more quickly and captures the overall shape of the macular hole. However, the zero-crossing $\phi = 0$ lacks the finer high-frequency surface details. Therefore we progressively upscale ϕ and re-solve Equation 2.9, for multiple scale levels:

$$S = \{s_i\},\ \forall i \in [1, n],\ s_i < s_{i+1} \,, \tag{2.11}$$

where $s_i = (s_x, s_y, s_z)$ and s_x, s_y, s_z are scaling factors for each the x, y, z image dimensions accordingly. Solving Equation 2.9 for these subsequent scales now only takes a small number of iterations given that $\phi = 0$ is already near the object boundary in each case (Figure 2.3b). Finally we are able to process ϕ at the original resolution ($s_n = (1, 1, 1)$) with a small number of iterations, capturing the finer surface details (Figure 2.3c).

Downscaling ϕ and I for the initial multi-scale stage s_1 is able to segment the macular hole, exploiting the fact that the hole is a large object in the OCT image. However, noise introduced by OCT propagates through the downscaled I, which may prevent the initial evolution from evolving to fill the hole cavity. Therefore we denoise the I for the smallest scale s_1 using a Wiener filter by [73, 74], which is effective at removing the speckle noise encountered in OCT imaging [75], and the result is shown in Figure 2.4a. In subsequent scales s_i where $2 \leq i \leq n$, we do not denoise the image as (1) we assume the contour is already filling the hole cavity at this stage, and (2) we wish to capture the finer surface details unhindered by the denoising process.

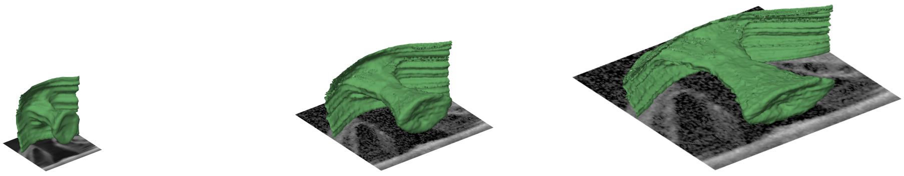


Figure 2.3: Result of 3D macular hole based on our multi-scale level set segmentation approach. In our experiment, we choose $n = 3$ scale levels with scale parameters: (a) $s_1 = (0.25, 0.25, 1)$, (b) $s_2 = (0.5, 0.5, 1)$, (c) $s_3 = (1, 1, 1)$. We do not alter the z-dimension in our case, as it is already shallow in the original OCT input images.

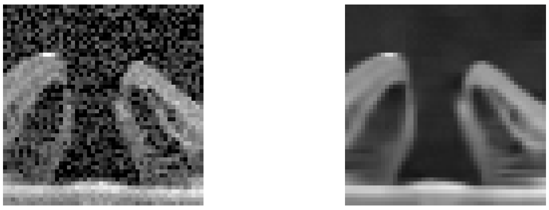


Figure 2.4: (a) OCT image from red box in Figure 2.1 downscaled by $s_1 = (0.25, 0.25, 1)$, (b) Result of denoising using the Wiener filter with default parameters [73, 74].

Implementation of E^{LGDF} in 3D requires 6 × 3D Gaussian convolutions per iteration, required for calculating intermediate variables in Equation 2.8. These convolutions are a performance bottleneck, therefore we use an optimized implementation of Gaussian convolution in the Fourier domain [76] which supports arbitrarily large standard deviation σ without impacting the performance.

2.5.4 Curvature-based Cutting Surface

The problem of LGDF is that it will only separate the vitreous humour from the retinal tissue, and cannot separate the macular hole from the main body of vitreous humour. We need to cut the previously segmented vitreous humour in such a way as to only capture the macular hole shape (Figure 2.2e). In our approach, we initially attempted to cut the macular hole based on intensity information in the image, but we found that a curvature-based approach is more robust across the different images.

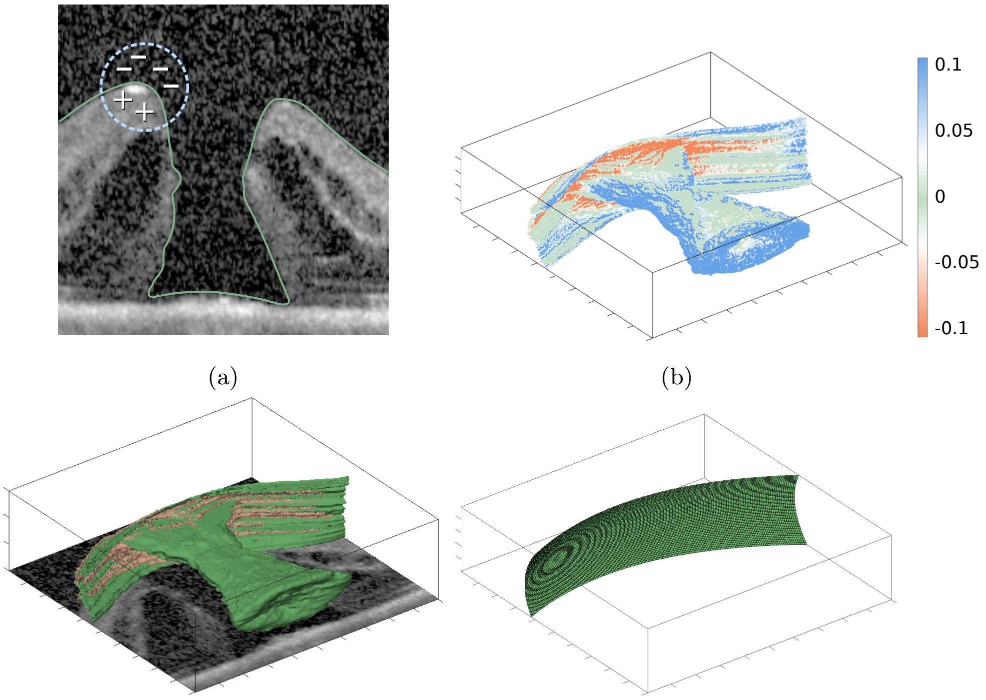
We compute curvature κ of the 3D active surface by using an efficient integral invariant approach [77, 78] based on [79], which uses a spherical kernel in 3D. We apply the signum function to ϕ and then convolve with a spherical kernel of radius r (Figure 2.5a). This effectively computes the difference between the inside and outside volume within a sphere centered at every point on the surface. The radius of the kernel determines the scale of the curves that it responds strongly to. In our case we find that $r = 116.10$ $[\mu m]$ captures the lip of the macular hole (in 200×200×49 [voxels] OCT images), while smoothing out high-frequency surface noise.

The curvature κ (Figure 2.5b) is defined over all points in the OCT image; however, we are only interested in the surface curvature at the lip of the macular hole. We therefore binarize ϕ and select only the surface voxels where $\kappa < 0$ (brown voxels in Figure 2.5c). We then fit a two variable 2^{nd} order polynomial to these voxels (Figure 2.5d). The mathematical function of the polynomial fitting $y(x, z)$ from the surface voxels with negative κ:

$$y(x, z) = p_{00} + p_{10}x + p_{01}z + p_{20}x^2 + p_{11}xz + p_{02}z^2, \tag{2.12}$$

where p_{00}, p_{10}, p_{01}, p_{20}, p_{11}, and p_{02} are coefficients represent the free parameters of the model.

In some pathalogical cases (e.g. Image 25, Table 2.3), the curvature based cutting surface is too high, and does not cleanly separate the macular hole. Therefore, we use Euclidean distance transform calculation function for 3D binary image [80] to translate the surface down by a fixed



(a) (b)

(c) (d)

Figure 2.5: (a) Spherical kernel applied at the surface $\phi = 0$, (b) Curvature κ on the 3D surface, (c) Surface voxels with negative κ (brown), (d) Fitting a 2^{nd} order polynomial to these voxels.

amount of pixels (in x, y and z dimensions) and then convert into microns as listed in the SD-OCT images property (we choose 154.80 [μm]), and finally "re-grow" the macular hole up to a fixed distance (we choose 100.62 [μm]) on all images. We find that these extra steps increase our overall accuracy.

2.5.5 Measurements

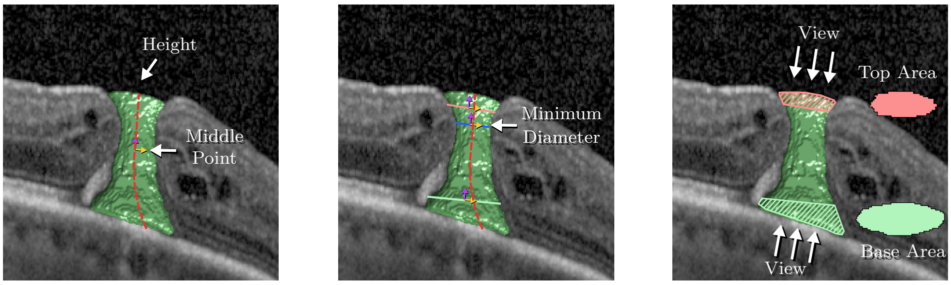
From a 2D side-view of the 3D macular hole, ophthalmologists require precise measurements of the minimum diameter (most narrow point on the hole) and base diameter. To compute these measurements robustly, we first extract a smooth 'centerline' of the macular hole, denoted $C = \{\vec{c}_i\}$, to act as a frame of reference for our measurements (Figure 2.6 red dashed line). The centerline is calculated using the approach proposed in [81], but extended to 3D. Specifically, each coordinate of the 'centerline' is defined to be the centroid of the slice:

$$\vec{c}_i = \left(\int_{\Psi_i} \vec{x} \, \mathrm{d}\vec{x}\right) \Big/ \left(\int_{\Psi_i} \mathrm{d}\vec{x}\right) = \frac{1}{|\Psi_i|} \sum_{\vec{x} \in \Psi_i} \vec{x} \,, \tag{2.13}$$

where $\Psi_i \subset \Omega \subset \Re^2$ is a set of the binary segmented pixels for the i^{th} slice (in the y-axis) in the macular hole after the cutting procedure. The centerline is then smoothed using robust local regression [82], as in the paper [81], using a smoothing parameter (in our case we choose 0.9 to represent a span of 90% of the signal). This is important as it ensures the centerline acts as a descriptor for the overall shape and direction of the hole (especially in the middle as we are not concerned about the centerline veering off at the top and base of the hole) and ensures that it remains highly insensitive to surface noise.

We compute the height measurement of the macular hole as the length of C, and then take the normal of the middle point along C (shown by purple and yellow arrows in Figure 2.6 left) to act as a basis for the three planes (Figure 2.6 middle: pink, blue, and green lines). The remaining measurements (top area, base area, and smallest area) are inferred from these planes: by examining the cross-section at distances 20% and 80% along the curve, and then finding the minimal cross-sectional area between these two planes accordingly. Since these cross-sections are generally elliptical, rather than circular, we define their diameters as their major axis lengths, as in [30].

Ophthalmologists currently use the minimum diameter from a 2D side view of the macular hole. Therefore we first find the smallest cross-sectional area between the top and bottom planes,



(a) (b) (c)

Figure 2.6: (a) We initially compute a smooth centerline (dashed red line), (b) normal at the central point is used as a basis for finding the top region and bottom region, as well as the minimal diameter between these two regions, (c) we view orthogonal projections of the cropped 3D volumes (striped regions) separately from above and below, giving us 2D areas (opaque regions) measuring the top and bottom of the hole accordingly.

and then calculate the diameter from this cross-section's centroid to act in place of the 2D measurement.

2.5.6 Method Parameters and Suggested Default Values

We found stable default level set evolution parameters through both our own empirical tuning across 30 images provided by the Sunderland Eye Infirmary, Sunderland, UK, and also through an experiment by minimizing the mean Jaccard Index across the 30 images with ground-truth segmentations using the built-in MATLAB genetic algorithm implementation [83]. We found that the results of the genetic algorithm coincide with our own findings, whereby the default values are chosen to be: $\sigma = 4$, $\alpha = 20$, $\mu = 1$, $\nu = 39$, $\lambda = 1.04$, across all images. We conducted a cross-validation experiment to evaluate the robustness of these suggested parameter values (σ, α, μ, ν, and λ) using leave-one-out validation [84] (suitable for such small datasets). The leave-one-out cross-validation across all 30 images results in a mean Jaccard Index of 0.9644 and standard deviation 0.0015. The small standard deviation values indicate the stability of our proposed parameters.

2.6 Results and Discussion

2.6.1 Qualititive Validation

In this section, we present the results of our automatic segmentation approach for 30 real-world OCT images provided by the Sunderland Eye Infirmary, Sunderland, UK. Tables 2.1 to 2.6 show the results of our multi-scale 3D segmentation after the curvature cutting procedure for each patient, and also a 2D z-slice showing the maximal hole cross-section of the 3D segmented hole. The results demonstrate that our approach is able to successfully capture the macular hole in all the images, in particular several challenging cases such as the narrow hole in Image 8 and the complex scenario in Image 12. Furthermore, our curvature-based cutting procedure is robust and can correctly identify the lip of holes that are valley-shaped (Image 2, Image 25), hill-shaped (Image 1, Image 8, Image 14, ...), flat (Image 6, Image 7, Image 16), and slanted (Image 9). The curvature cutting procedure is excluded for Image 30 as the hole is fully enclosed and without any opening (Table 2.6).

2.6.2 Quantitative Validation

In order to validate our method quantitatively, we compare the shape of our automatically segmented macular hole with 3D ground truth segmentations. These ground truth segmentations were acquired by a clinical eye surgeon manually sketching a 2D region for each slice of the macular hole for 10 images (Table 2.1). The results in Table 2.7 show the accuracy, sensitivty, Jaccard index (ratio of intersection and union between our segmentation and ground truth) and the Dice similarity coefficient (DSC), as defined in [85, 86]. These metrics are calculated (1) between the 3D ground truth sketched region, (2) for the same dataset of our 3D multi-scale LGDF (MS-LGDF) segmentation, (3) for a 3D continuous max-flow (CMF) graph cuts with default parameters [87–89] based approach using a 5 × 5 × 5 [voxels] 3D median filter pre-processing as with [30]. We also constrain the top of the hole manually as a constraint boundaries (as with [30]). Table 2.7 also show a side-by-side comparison with the ground truth, our method, and the existing approach in Figure 2.7. In general, we see that our method scores highly in all cases and that our method more closely captures the shape of the macular hole, in particular at the macular hole boundary.

We show the 3D measurements extracted with our approach in Table 2.8. Pixels are converted to [mm] metric scaling according to the OCT image metadata, yielding the real size and form of the macular hole (volume, surface area, base area, base diameter, top area, top diameter, height, and minimum diameter). The Bland and Altman method [90, 91] was used to calculate the mean difference ($\bar{\mathrm{d}}$) and standard error (se) for the base diameter (BD) and minimum diameter (MD). This includes 30 datasets (Table 2.9) and ground truths acquired from the average of two manual measurements, taken by a clinical eye surgeon at 4 month intervals (intra-observer validation). The validation between the ground truth and MS-LGDF has smaller difference than validation

Table 2.1: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure.

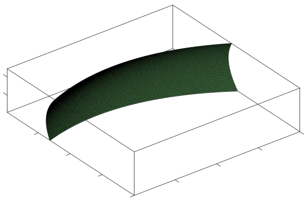
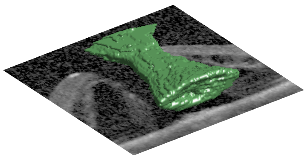
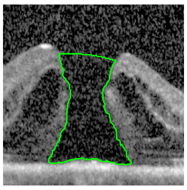
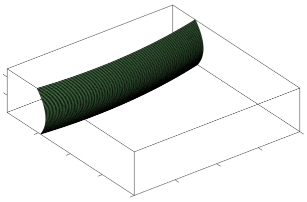
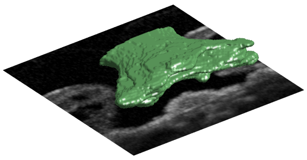
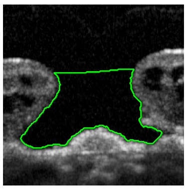
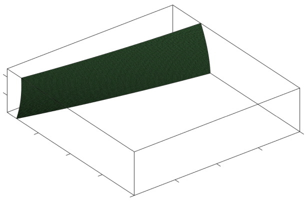
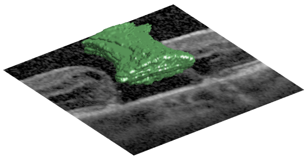
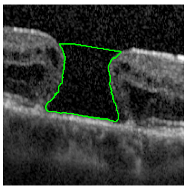
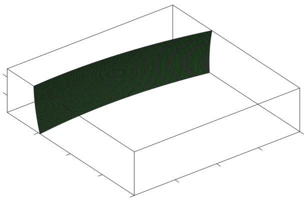
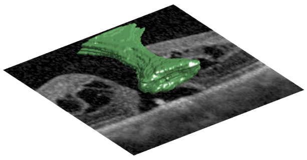
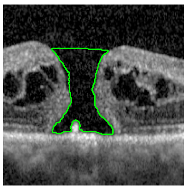
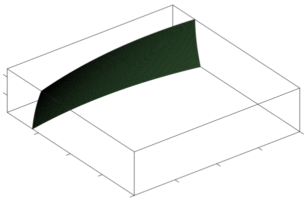
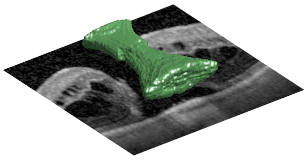
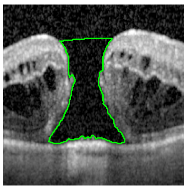
	Surface Fitting	**3D Segmentation**	**2D Cross-Section**
Image 1			
Image 2			
Image 3			
Image 4			
Image 5			

Table 2.2: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure (continued).

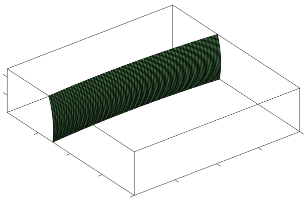
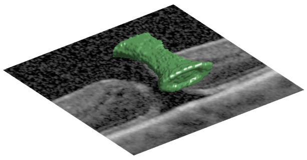
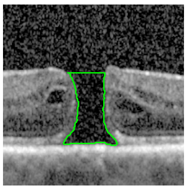
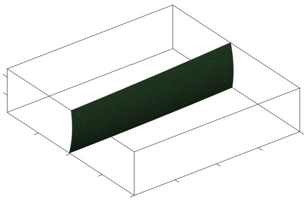
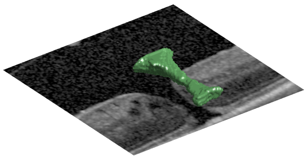
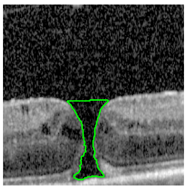
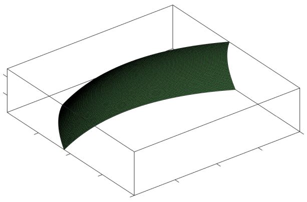
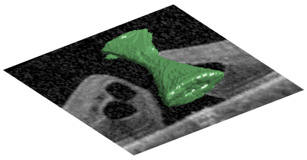
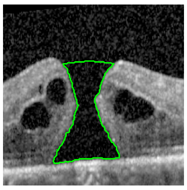
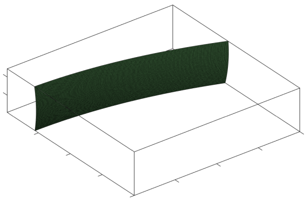
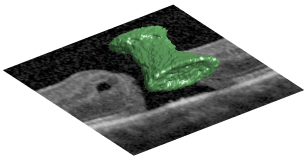
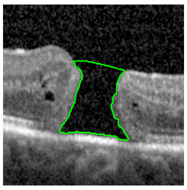
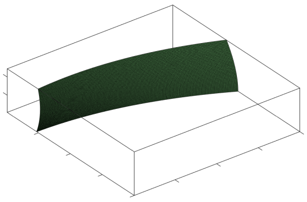
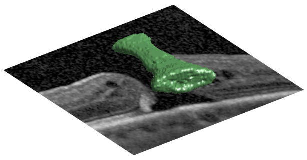
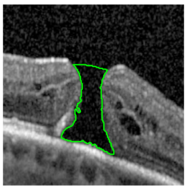
	Surface Fitting	**3D Segmentation**	**2D Cross-Section**
Image 6			
Image 7			
Image 8			
Image 9			
Image 10			

Table 2.3: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure (continued).

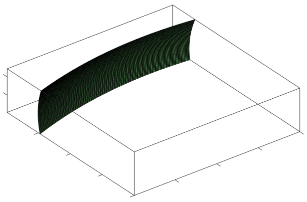
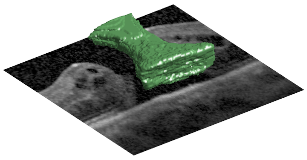
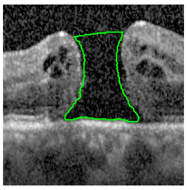
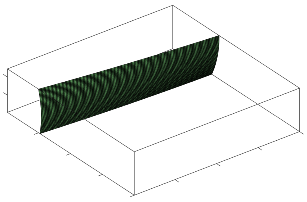
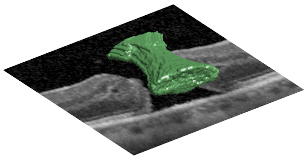
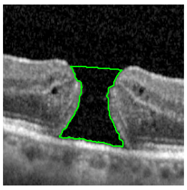
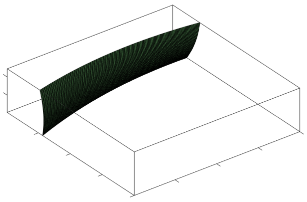
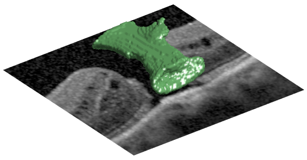
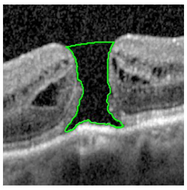
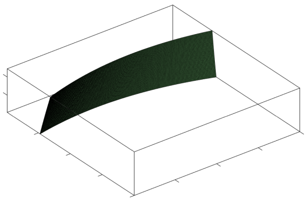
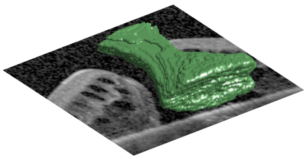
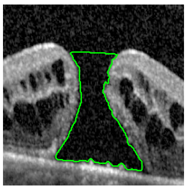
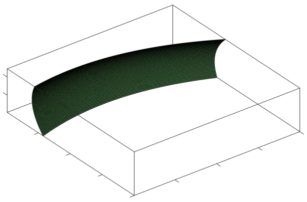
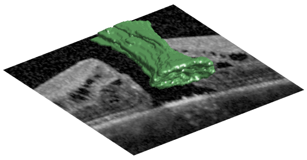
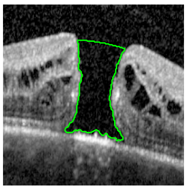
	Surface Fitting	**3D Segmentation**	**2D Cross-Section**
Image 11			
Image 12			
Image 13			
Image 14			
Image 15			

Table 2.4: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure (continued).

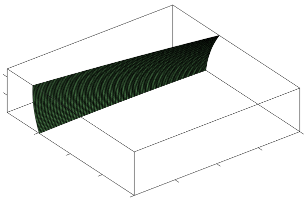
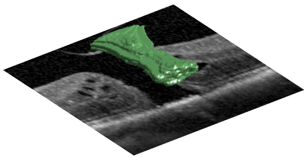
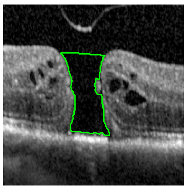
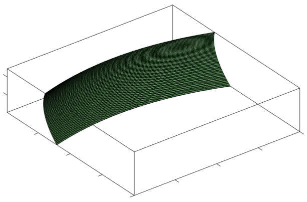
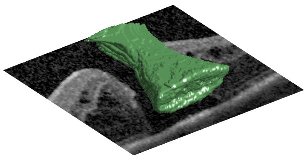
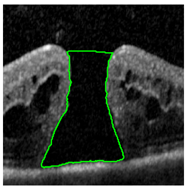
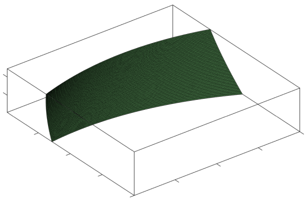
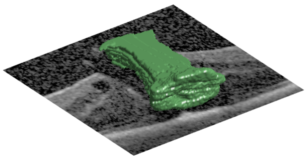
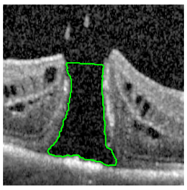
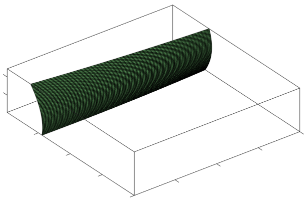
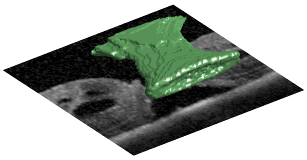
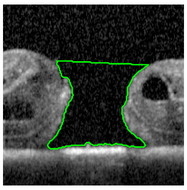
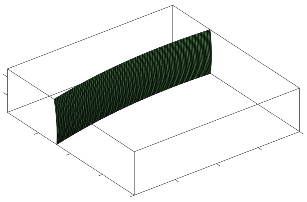
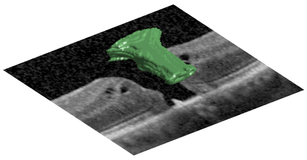
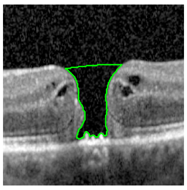
	Surface Fitting	**3D Segmentation**	**2D Cross-Section**
Image 16			
Image 17			
Image 18			
Image 19			
Image 20			

Table 2.5: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure (continued).

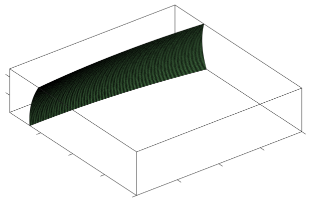
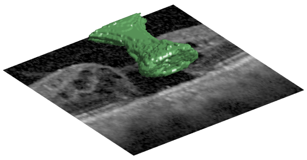
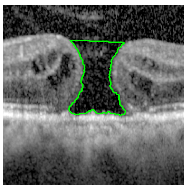
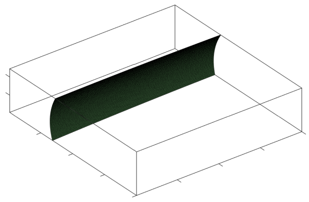
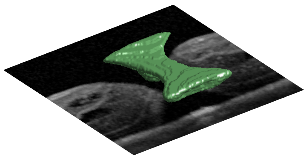
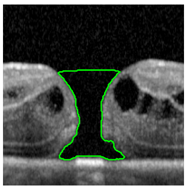
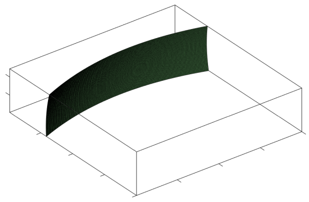
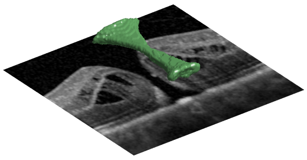
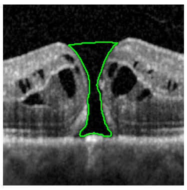
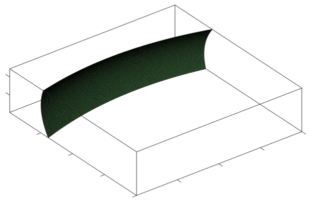
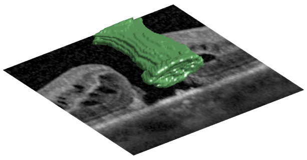
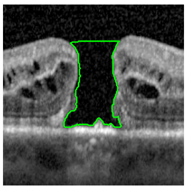
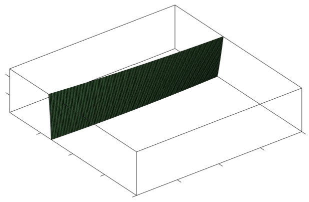
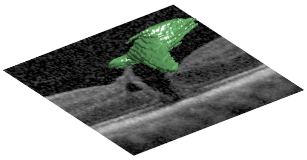
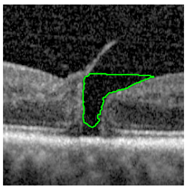
	Surface Fitting	**3D Segmentation**	**2D Cross-Section**
Image 21			
Image 22			
Image 23			
Image 24			
Image 25			

Table 2.6: Segmentation results of our multi-scale 3D segmentation after the curvature cutting procedure (continued).

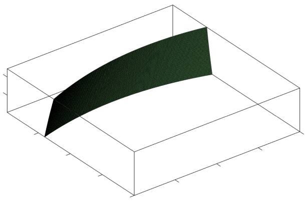
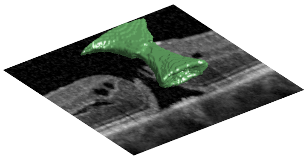
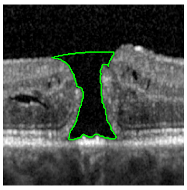
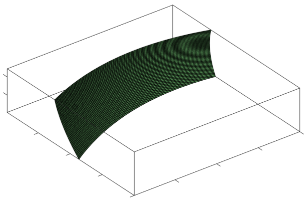
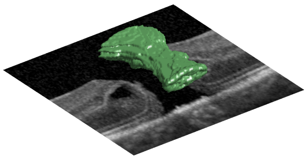
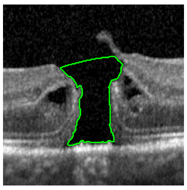
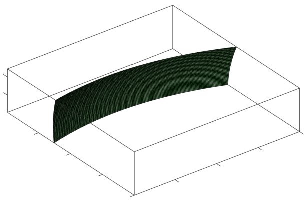
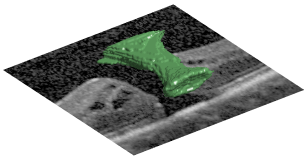
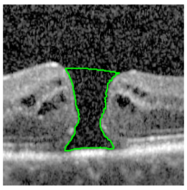
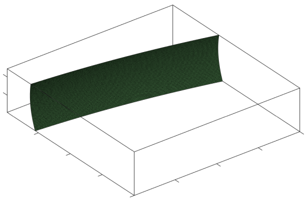
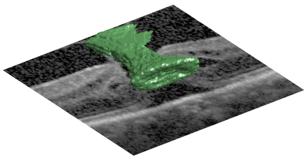
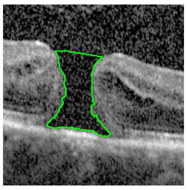
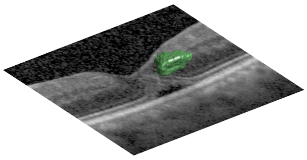
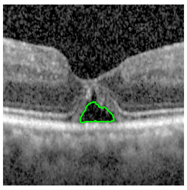
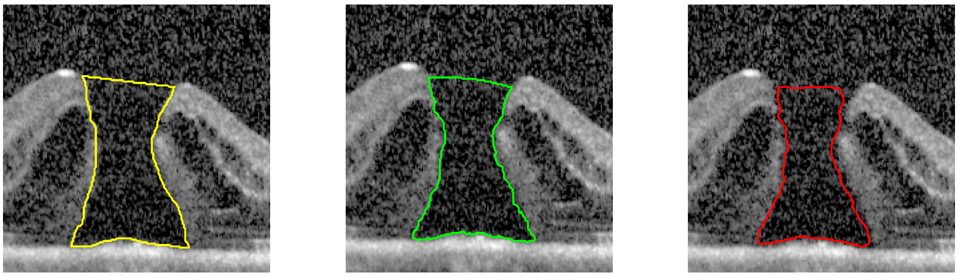
	Surface Fitting	3D Segmentation	2D Cross-Section
Image 26			
Image 27			
Image 28			
Image 29			
Image 30			

Table 2.7: Validation of macular hole segmentation results (Mean±Standard Deviation) with the ground truth between MS-LGDF and 3D CMF.

Segmentation Method	**Accuracy** %	**Sensitivity** %	**Jaccard Index** %	**DSC** %
MS-LGDF	$99.19_{\pm 00.56}$	$85.18_{\pm 04.63}$	$76.34_{\pm 10.31}$	$86.19_{\pm 07.55}$
CMF	$98.83_{\pm 00.74}$	$71.89_{\pm 07.61}$	$66.31_{\pm 10.51}$	$79.27_{\pm 08.33}$



(a) Ground truth (b) MS-LGDF approach (c) CMF approach

Figure 2.7: 2D cross-sections comparing macular hole segmentations with the ground truth by an expert, our method, and a 3D CMF graph-cuts approach.

between the ground truth and CMF. MS-LGDF is shown to be robust and stable, producing the same results when run on the same data. The Bland and Altman plots in Figure 2.8 - Figure 2.10 from Table 2.9 show the agreement between the manual clinician measurements, our approach, and the graph cuts approach. The intraclass correlations two-way model Case 2, ICC (A,1) [92], was calculated to show correlation of intra-observer, manual - MS-LGDF, and manual - CMF (Table 2.9). The correlation coefficient shows greater agreement between clinician measurements and our approach than with the CMF method. It is worth noting that the manual 2D measurements are fundamentally limited by their assumption that the cross-section of the macular hole is circular. This assumption fails in cases where the cross-sections are elliptical. Therefore we expect to see some disagreement between the proposed automated 3D measurement, which is not limited to such cases, and the manual measurements.

2.6.3 Performance

In our experiments, we measure the performance for all input OCT images with size 200 × 200 × 49 [voxels] at all the stages in our approach. In particular, the average segmentation

Table 2.8: 3D measurements of macular holes with our approach.

3D Image Data	**Volume**	**Surface Area**	**Base Area**	**Base Diameter**	**Top Area**	**Top Diameter**	**Height**	**Minimum Diameter**
	$[10^{-3}mm^3]$	$[mm^2]$	$[mm^2]$	$[mm]$	$[mm^2]$	$[mm]$	$[mm]$	$[mm]$
Image 1	148.55	2.72	0.266	0.97	0.141	0.69	1.43	0.46
Image 2	85.45	1.61	0.199	0.83	0.153	0.56	0.50	0.52
Image 3	82.63	1.73	0.164	0.70	0.128	0.66	0.98	0.46
Image 4	57.94	1.48	0.145	0.71	0.122	0.62	1.06	0.24
Image 5	95.95	1.97	0.219	0.86	0.118	0.65	1.27	0.39
Image 6	45.33	1.19	0.123	0.66	0.065	0.50	0.95	0.34
Image 7	20.25	0.79	0.036	0.34	0.073	0.52	0.97	0.13
Image 8	66.43	1.50	0.164	0.73	0.092	0.55	1.27	0.20
Image 9	61.99	1.35	0.175	0.74	0.098	0.55	0.92	0.31
Image 10	49.36	1.31	0.124	0.70	0.058	0.49	1.21	0.31
Image 11	99.61	2.12	0.247	0.92	0.109	0.69	1.11	0.39
Image 12	62.04	1.41	0.156	0.73	0.103	0.59	1.03	0.34
Image 13	59.46	1.54	0.108	0.54	0.107	0.73	1.19	0.34
Image 14	187.48	3.36	0.374	1.26	0.125	0.67	1.52	0.43
Image 15	76.86	1.90	0.147	0.74	0.092	0.56	1.27	0.44
Image 16	48.72	1.26	0.067	0.49	0.096	0.58	1.08	0.32
Image 17	135.13	2.61	0.253	0.90	0.093	0.63	1.47	0.40
Image 18	54.84	1.29	0.239	1.05	0.086	0.63	0.63	0.52
Image 19	110.10	1.89	0.221	0.64	0.166	0.52	1.12	0.32
Image 20	41.62	1.13	0.036	0.34	0.123	0.62	0.97	0.35
Image 21	64.17	1.58	0.163	0.84	0.121	0.63	0.95	0.31
Image 22	25.95	0.77	0.087	0.37	0.102	0.53	0.58	0.20
Image 23	24.11	0.83	0.043	0.39	0.091	0.56	1.18	0.15
Image 24	65.96	1.67	0.135	0.72	0.089	0.52	1.07	0.39
Image 25	38.80	1.07	0.011	0.20	0.265	1.10	0.70	0.20
Image 26	53.74	1.35	0.091	0.55	0.140	0.58	1.11	0.23
Image 27	59.82	1.43	0.085	0.52	0.138	0.69	1.07	0.28
Image 28	51.59	1.34	0.107	0.63	0.125	0.64	1.07	0.25
Image 29	56.02	1.44	0.160	0.73	0.085	0.52	1.03	0.35
Image 30	7.25	0.26	0.056	0.47	0.011	0.22	0.26	0.22

Table 2.9: Validation of the base diameter (BD) and minimum diameter (MD). For intra-observer comparison, $\bar{\text{d}}$ is the old measurement minus the recent. Otherwise, $\bar{\text{d}}$ is the manual measurement minus our approach or the graph cuts approach.

Observer		$\bar{\text{d}}$ $[mm]$	se $[mm]$	**Confidence** $[mm]$	**ICC**
Intra-Observer	**BD**	0.023	0.012	0.025	0.9539
	MD	-0.012	0.010	0.020	0.9153
Manual-MS-LGDF	**BD**	0.033	0.018	0.036	0.9032
	MD	0.023	0.014	0.028	0.7849
Manual-CMF	**BD**	0.105	0.020	0.041	0.7744
	MD	0.040	0.015	0.031	0.6778

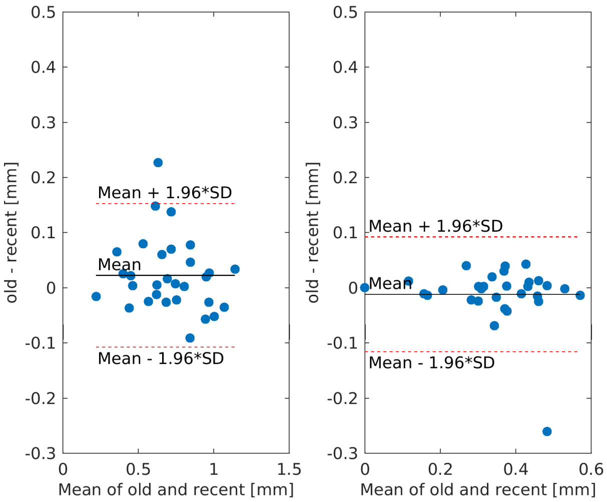


Figure 2.8: Bland-Altman plot of base diameter (left) and minimum diameter (right) for intra-observer from Table 2.9.

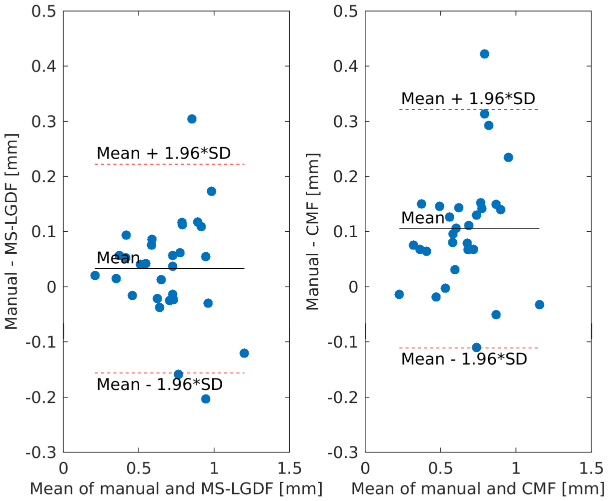


Figure 2.9: Bland-Altman plot of base diameter for the manual measurement minus our approach (left) and the graph cuts approach (right) from Table 2.9.

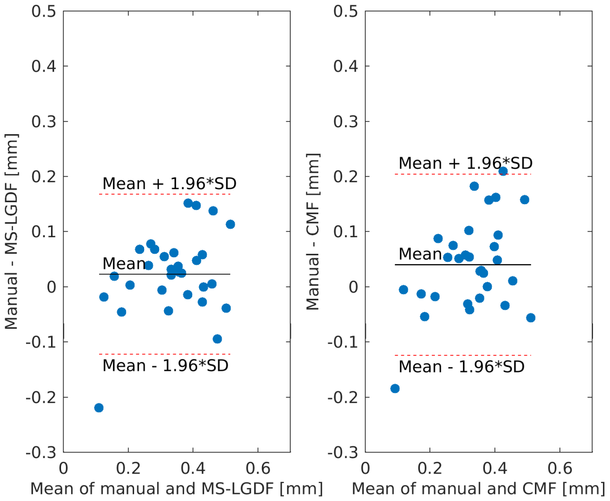


Figure 2.10: Bland-Altman plot of minimum diameter for the manual measurement minus our approach (left) and the graph cuts approach (right) from Table 2.9.

performance of our 3D MS-LGDF is 147.65 seconds (2.46 minutes), which is significantly faster than the original 3D LGDF with a mean of 9122.87 seconds (2.5 hours) to converge. The mean time for the surface cutting procedure is 19.84 seconds. While the timings in our unoptimized MATLAB CPU implementation are considered acceptable, we believe our method can be heavily optimized if rewritten in C/C++. Additionally, the average segmentation performance of our MATLAB GPU-Array [83] implementation is 91.70 seconds for 3D MS-LGDF and 11.45 seconds for the surface cutting procedure (significantly faster than manual segmentation, which take 28.10 minutes in average).

The mean time for our automatic procedures to acquire all of the measurements is 1.15 minutes per OCT image, which is largely due to calculating volumetric analysis metrics in Table 2.8 such as volume, surface area, base area, base diameter, top area, top diameter, height and minimum diameter. The automatic minimum diameter calculation takes the longest time due to finding the minimal cross-sectional area between 20% and 80% of these two planes. The average times to measure base diameter and minimum diameter manually are 19.06 seconds and 22.73 seconds respectively (total for both is 41.79 seconds). This exclude calculating the areas, compared to 1.15 minutes for calculating all the measurement using our unoptimized automatic method.

2.7 Conclusion and Future Work

In conclusion, we have proposed an automatic and robust method to both segment and extract measurements from 3D OCT images of macular holes. In particular, we found that the lip or opening of the macular hole can be automatically cut based on curvature information, and that a significant performance increase can be obtained over state-of-the-art level set methods for large objects through multi-scale techniques. Furthermore, we have proposed a novel method to collect various macular hole measurements through our definition of a robust centerline, and the method has been validated both quantitatively and qualitatively.

In the future, we would like to extend our approach to automatically extract more recent measurements, such as the area ratio factor (ARF) shown to be an effective predictive factor for diagnostic and treatment outcomes [93].

Epilogue

In this chapter, we demonstrated an extended 3D segmentation method with a novel curvature-based cutting and 3D measurement procedure for macular holes.

Chapter 3

Interactive GPU Active Contours for Segmenting Biological and Medical Images

Prologue

In this chapter, a faster GPU implementation of the LGDF energy model will be presented, which can segment inhomogeneous objects with poorly defined boundaries as often encountered in biomedical images. This method demonstrates the practical efficacy of our interactive elements for a variety of real-world datasets including a macular hole dataset.

Declaration: This chapter is based on the following publication: Willcocks, C. G., Jackson, P. T., Nelson, C. J., Nasrulloh, A. V. & Obara, B. Interactive GPU active contours for segmenting inhomogenous objects. *Journal of Real-Time Image Processing,* 1–14 (2017). This chapter is presented as published, although referencing and notation has been altered and cross-referencing added for consistency throughout this thesis. Some stylistic changes have been made for consistency.

3.1 Introduction

Image segmentation is a large research field with many practical applications, including but not limited to:

- Biosciences:
    - Cellular, developmental and cancer biology.
    - Plant biology, including plant-pathogen interactions.
    - Animal biology, including virus-host interactions and bacterial infections.
    - Microbiology, including food safety.
    - Neuroscience, including connectome projects and developmental neuroscience.
- Medicine:
    - Automated differential diagnosis.
    - Diagnostic measurements, shape and volume, of:
        - ∗ Macular holes in retinal degeneration.
        - ∗ Aneurysms, clotting and infarction.
        - ∗ Tumors, neoplasia and dermatological moles.
        - ∗ MRI segmentation in dementia and Alzheimer's.
    - Computer Assisted Surgery:
        - ∗ Pre-surgical planning and surgery simulation.
        - ∗ Guided surgical navigation.

The primary problems with current segmentation approaches are that they are either: (1) too limited, e.g. only able to segment objects by simple criteria, such as objects with consistent mean intensity [94, 95], (2) using too much memory or too slow, taking several hours to segment large 2D or 3D objects [57], (3) lacking in interactivity with the segmentation process in response to visual feedback [96], (4) requiring copious training data [53], or (5) difficult to use, requiring complex interfaces and multiple algorithms [97].

The oldest and most widely cited segmentation approaches are active contours [66]; these are variational frameworks which allow users to define an initial open or closed curve that deforms so as to minimize an energy functional, outlining or surrounding the object of interest. While active contours have been realized as fully automatic approaches without initial contours [98], their original foundation as an assisted approach is still important today as it allows users, such as clinicians, to extract precise measurements from specific objects of interest within a complex image. However, such interactivity relies on real-time visual feedback; therefore they must also be computationally efficient.

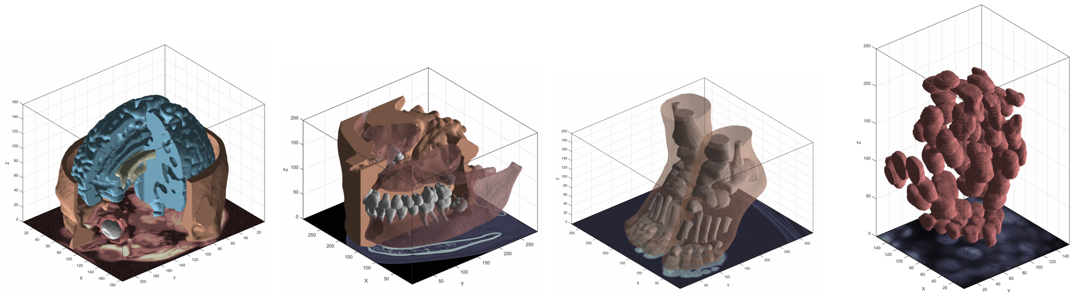
Graphics processing units (GPUs) provide energy efficient parallel computing and enable real-time interactive segmentation for larger 2D or 3D datasets [38, 39], but existing GPU segmentation methods currently rely on simple segmentation criteria restricting their usage and applications. The popular local Gaussian distribution fitting (LGDF) energy model [57] is much more powerful and able to segment a wider variety of general objects. However, it requires several intermediate processing steps that must be implemented sequentially, making it challenging to efficiently implement on graphics hardware. The current implementation of the LGDF energy model can segment small 2D images (99 × 120 pixels in 27.37 seconds), but requires several hours of processing for larger 2D or 3D images [57]. For a 3D image of size 256 × 256 × 160 voxels, this would take 6.6 hours if the implementation were available for 3D, preventing usage in many practical applications.

3.1.1 Contributions

In our approach, we: (1) significantly increase the performance of the LGDF energy model through an optimized GPU implementation, handling much larger 2D images and even 3D images at interactive performance, (2) introduce a novel set of interactive brush functions that are integrated into the GPU kernels such as to modify and constrain the evolving level set in real-time, (3) provide a ray tracer to view the segmentation results at each time-step, and (4) expose a simpler and more intuitive parameter space to the user, with suggested values and ranges. The combination of these four enhancements greatly improves the practicality of what is already considered a state-of-the-art level set method of particular relevance to the biomedical image processing communities. Our software is shown to be stable with respect to its input parameters and robust to noise through a large experiment on synthetic data, and is further evaluated through segmenting a wide variety of real-world images, such as those in Figure 3.1.

3.2 Related Work

The field of active contours first gained mainstream adoption with the ‘active snakes’ model published by [66]. This seminal work proposes iterative evolution of an initial spline curve, with



(a) Brain and ventricles (b) Dental scan (c) Foot bones and tissues (d) Zebrafish cells

Figure 3.1: A selection of 3D objects segmented by our software. Our interactive method allows users to efficiently capture specific objects (coloured separately) within the data, such as the teeth in (b). Image (a) is a simulated brain MRI [99], images (b) and (c) are CT scans [100] and (d) shows selective plane illumination microscopy (SPIM) of zebrafish eye lens cells [101]. (a) Brain and ventricles. (b) Dental scan. (c) Foot bones and tissues. (d) Zebrafish cells.

the evolution being governed by the minimization of an energy functional, the local minima of which correspond to curves that fit along prominent edges in the image. Level set methods (core theory explained in [102]) model contours implicitly as the zero-crossing of a scalar field. Originally they were proposed in [69] to model the evolution of inter-region boundaries in physical simulations. Malladi et al. [68] applied level sets to active contours, with the evolution of the contour being governed by its local mean curvature and the intensity gradient magnitude of the image, in such a way that local curvature is reduced and the motion of the contour stops as it approaches an edge. In [65], the authors develop a level set based active contour framework in which the energy functional is based on the Mumford-Shah model, rather than image edges, which in practice are often faint, blurred or broken. The Mumford-Shah energy model [70] is minimized by an optimal partition of an image into piecewise smooth segments, and high-quality implementations exist on the GPU [103]. The global optimum can be found using a primal-dual algorithm [104] resulting in a cartoon-like rendering of the original image. Local solutions, such as with a trust-region approach [105], have applications in interactive segmentation, where local edits need to be made frequently.

Deep convolutional neural networks are the state-of-the-art in image segmentation, where millions of parameters of deeply layered convolutions are learnt using backpropagation [53]. These models are capable of learning abstract features in the data; however, their current reliance on such large datasets makes them unusable for a number of applications.

The influential public datasets with groundtruth segmentations (such as BSDS, MSRC, iCoseg, FlickrMFC, SegTrack) include RGB videos or 2D images such as cars, chairs, and people. Of these, the interactive approaches take as input a set of scribbles where objects follow similar colour distributions [106]. Graph cut segmentation is popular in this field, where Grady [107] and Vineet and Narayanan [108] propose GPU implementations. For interactive segmentation in the biosciences, we find the main limitations being (1) the initialization of the foreground-background scribbles in 3D datasets such as networks and (2) the opaque intermediate steps of the cutting algorithm making it difficult to obtain a high level of visual feedback. While popular and easy to validate, these approaches address a different problem to grayscale 3D segmentation as with imaging modalities (such as CT, PET, SPECT, MRI, fMRI, ultrasound, optical imaging and microscopy) in the biosciences [38]. There is still a need for benchmark medical datasets with well-defined interactive performance evaluation [109].

Accelerating image segmentation with GPUs is a large research field with several comprehensive surveys [38, 39, 110, 111]. The survey by [38] covers a broad range of algorithms and different imaging modalities, whereas Smistad et al. [39] focuses more on GPU segmentation with a detailed discussion on the current GPU architecture.

The GPU level set methods in the literature focus on limiting the active computational domain to a small region near the zero-crossing of the level set function, such as the traditional narrow band algorithm [112]. More recent extensions classify the active region using simple operations on the spatial and temporal derivatives of the level set function [95], and then discard unimportant regions through parallel stream compaction. While limiting the active computational domain produces excellent performance with lower memory usage, the current implementations all use simple speed functions that attract the level set to make it grow and/or shrink within a fixed intensity range [94, 95, 113]. In contrast, the LGDF model proposed by [57] is able to segment much more challenging images, in which objects exhibit intensity inhomogeneity or even have the same mean intensity as their background, being distinguished only by intensity variance. However, to date the only existing implementation runs on the CPU, likely due to the sequential dependency of convolutions in the intermediate steps. Further, the LGDF model is derived from [65] who introduce C^∞ regularization of the Heaviside and Dirac functions which are non-zero everywhere, unlike the C^2 regularized Heaviside (proposed in [114]) which is non-zero only in the vicinity of the contour. C^∞ regularization restrains the algorithm from converging on local minima, but precludes traditional narrow band or sparse field algorithms because it requires the level set to update at all points on each time step.

GPU active contour methods parallelize the calculation of the energy forces described in the original snakes paper [66]. Traditional methods rely on simple intensity gradients and are prone to converging on local minima; however, [115] introduced a diffusion of the gradient vectors called gradient vector flow (GVF) to address this problem. [116] were one of the first GPU active contour implementations using GVF, and more recent optimizations in Open Computing Language (OpenCL) exploit cached texture memory which has spatial locality in multiple dimensions [117]. The active contour can also be approximated by a surface mesh, such as in [118] who use Laplacian smoothing on local neighborhoods in conjunction with driving mesh vertices with gradient and intensity forces. However, these approaches still rely on the image gradient being a reliable indication of object boundaries, which is not the case in many real-world images [65].

Ever since the original snakes paper, active contours have gained popularity through being able to interactively edit the contour, or setup constraints to guide its motion [66]. Region-based active contour methods provide the option to initialize with a simple primitive shape, or sketch a starting region [119]. The more advanced approach by [120] introduces non-Euclidean radial basis functions, which are weighted by the image features and blended to form an implicit function whose sign can be fixed at user-defined control points. The tool by [97] provides an interactive interface with geodesic active contours [121] and region competition [122]. Region competition favors a well-defined intensity range, whereas the geodesic approach is better suited for images with clear edges; by combining both approaches, [97] can segment a broad range of images, yet it requires significant tuning and can still fail in complex images with neither a well-defined intensity range nor clear edges.

There are several GPU approaches that produce segmentation without relying on initialization of a seed region [98]. Clustering methods join regions of a high-dimensional feature space [123] and superpixel approaches [124] form clusters that are deliberately over-segmented into more manageable regions. These approaches are good at simplifying complex images, yet they do not capture specific objects. In contrast, active shape and appearance methods fit a model to the data based on prior knowledge; however, this inherently makes assumptions of the overall shape of the objects and fails when these assumptions are not met.

3.3 Method

The LGDF model, originally proposed in [57], builds on existing active contour literature by introducing a new energy functional based on the local Gaussian distributions of image intensity.

This functional drives a variational level set approach which is able to segment objects whose intensity mean and variance are inhomogeneous. Rather than creating segments whose intensity is as uniform as possible, this algorithm allows slow changes in intensity across an object, penalizing only sudden changes within it, without relying on a gradient based edge detector [65].

The segmentation is represented by a level set function $\phi(\vec{x})$. The foreground region is the set of points $\{\vec{x} : \phi(\vec{x}) < 0\}$ and the exterior (or background) is $\{\vec{x} : \phi(\vec{x}) \geq 0\}$. The contour itself (or surface in 3D) is thus defined implicitly as the zero level set, $\{\vec{x} : \phi(\vec{x}) = 0\}$. Segmentation is achieved by minimizing a global energy functional:

$$E = E^{\mathrm{LGDF}}(I, \phi) + \mu \mathcal{P}(\phi) + \nu \mathcal{L}(\phi)\,, \tag{3.1}$$

where $\mu, \nu > 0$ are weighting constants, E^{LGDF} is the LGDF energy term which drives the contour to fit along salient image edges, $\mathcal{P}$ avoids the need to periodically re-initialize ϕ to a signed distance function [72], and $\mathcal{L}$ penalizes the contour length to ensure smoothness. The E^{LGDF} term is the sum of the individual LGDF energies for each pixel $\vec{x}$:

$$\begin{aligned} E^{\mathrm{LGDF}}(I, \phi, \vec{x}) = &- \int_{\Omega} \omega(\vec{y} - \vec{x}) \log(p_{1,\vec{x}}(I(\vec{y}))) M_1(\vec{y}) \mathrm{d}\vec{y} \\ &- \int_{\Omega} \omega(\vec{y} - \vec{x}) \log(p_{2,\vec{x}}(I(\vec{y}))) M_2(\vec{y}) \mathrm{d}\vec{y}\,, \end{aligned} \tag{3.2}$$

where $\omega(\vec{y} - \vec{x})$ is a Gaussian weighting function centered on $\vec{x}$, $p_{1,\vec{x}}$ is a Gaussian approximation of the intensity distribution for the part of the neighborhood of $\vec{x}$ lying outside the contour (and inside for $p_{2,\vec{x}}$), and M_1 equals one outside the contour, zero inside (vice-versa for M_2). This quantity is smaller when the intensity distributions in the parts of the neighborhood of $\vec{x}$ lying outside and inside the contour are well approximated as Gaussian distributions, which can only be achieved by deforming the contour so that it separates regions of different intensity mean and variance.

The mean and variance parameters for these local Gaussian distributions are denoted $u_i(\vec{x})$, $\sigma_i(\vec{x})$ where $i \in \{1, 2\}$ for regions outside and inside the contour, respectively:

$$u_i(\vec{x}) = \frac{\int \omega(\vec{y} - \vec{x}) I(\vec{y}) M_i(\phi(\vec{y})) \mathrm{d}\vec{y}}{\int \omega(\vec{y} - \vec{x}) M_i(\phi(\vec{y})) \mathrm{d}\vec{y}}\,, \tag{3.3}$$

$$\sigma_i(\vec{x})^2 = \frac{\int \omega(\vec{y}-\vec{x})(u_i(\vec{x}) - I(\vec{y}))^2 M_i(\phi(\vec{y}))\mathrm{d}\vec{y}}{\int \omega(\vec{y}-\vec{x}) M_i(\phi(\vec{y}))\mathrm{d}\vec{y}} \,. \tag{3.4}$$

Specifically, they express for each pixel the mean and variance of neighboring grey values that lie outside and inside the contour (for pixels whose entire neighborhood lies on one side of the contour, only one pair of these values is defined). The size of each pixel's neighborhood is determined by the standard deviation of the Gaussian weighting function, ω. This is a user-defined parameter, denoted σ. A larger neighborhood increases the range from which a pixel may influence the contour. This results in faster evolution, greater capture range, and a greater tendency to produce segments whose boundaries separate large regions of different mean intensity.

The internal energy term $\mathcal{P}$ penalizes the contour's deviation from a signed distance function [72] to ensure numerical stability [125]:

$$\mathcal{P}(\phi) = \int_\Omega \frac{1}{2}\left(|\nabla\phi(\vec{x})| - 1\right)^2 \mathrm{d}\vec{x} \tag{3.5}$$

and $\mathcal{L}$ penalizes the contour length to ensure smoothness:

$$\mathcal{L}(\phi) = \int_\Omega |\nabla H(\phi(\vec{x}))|\, \mathrm{d}\vec{x} \,, \tag{3.6}$$

where H is the C^∞ regularized Heaviside function, discretized to operate on a regular grid, first proposed by [65]:

$$H(x) = \frac{1}{2}\left[1 + \frac{2}{\pi}\arctan(x)\right] \,. \tag{3.7}$$

The total energy functional (Equation 3.1) can be minimized by applying the calculus of variations [57] yielding the following PDE:

$$\frac{\partial\phi}{\partial t} = -\delta(\phi)(\lambda_1 e_1 - \lambda_2 e_2) + \mu\left(\nabla^2\phi - \kappa\right) + \nu\delta(\phi)\kappa \,, \tag{3.8}$$

where δ is the regularized Dirac function $\delta(x) = H'(x)$ [65], λ_1, λ_2, ν and μ are parameters controlling the weight of the terms, and κ is the contour's local curvature [69]:

$$\kappa = \mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) \tag{3.9}$$

and $-\delta(\phi)(\lambda_1 e_1 - \lambda_2 e_2)$ is the force due to E^{LGDF}:

$$e_i(\vec{x}) = \int_\Omega \omega(\vec{y} - \vec{x}) \left[\log(\sigma_i(\vec{y})) + \frac{(u_i(\vec{y}) - I(\vec{x}))^2}{2\sigma_i(\vec{y})^2} \right] \mathrm{d}\vec{y}\,. \tag{3.10}$$

The data fitting term $e_1(\vec{x})$ quantifies how badly the pixel $\vec{x}$ would fit with the outside-contour parts of its neighbors' neighborhoods. When e_1 is high and $\vec{x}$ does not belong outside, $\frac{\partial \phi}{\partial t}$ is made more negative, so ϕ lowers at that point and the contour grows outwards, swallowing $\vec{x}$. The same applies in reverse for e_2.

Due to the smooth form of the C^∞ regularized Heaviside (Equation 3.7), $\delta(\phi) = H'(\phi)$ is non-zero everywhere. This allows ϕ some freedom to change at any point in the image, not just in a narrow band around the contour. This helps prevent convergence on local energy minima [65].

3.3.1 GPU Implementation

The goal of the implementation is to iteratively solve Equation 3.8 for $\phi(\vec{x}, t)$, and visualise the results at each iteration. This is done by discretizing ϕ with respect to time and applying numerical integration: starting with $\phi(\vec{x}, t = 0)$ (which is specified by the user), an update loop computes $\phi(\vec{x}, t + \Delta t)$ by computing $\frac{\partial \phi}{\partial t}$ according to Equation 3.8 and assuming this quantity stays constant during the short time step Δt. Existing GPU level set methods implement their update rule inside a single kernel function; however, E^{LGDF} is more challenging as relies on intermediate stages with neighborhood operations, such as convolutions and derivatives, whose sequential dependencies must be considered such as to avoid race conditions.

The update rule in Equation 3.8 requires convolutions (Equation 3.10) of intermediate variables that themselves rely on other convolutions (Equations 3.3-3.4). The relationships of these variables are shown in Figure 3.2, where an arrow from A to B indicates that A is required in the computation of B. Wherever they appear, I denotes the input image and H the smooth Heaviside function (Equation 3.7). All variables of the form GX represent the n-dimensional Gaussian convolution of X.

We compute the means and variances (Equations 3.3-3.4) from GIH, GH, GI^2H, GI and GI^2

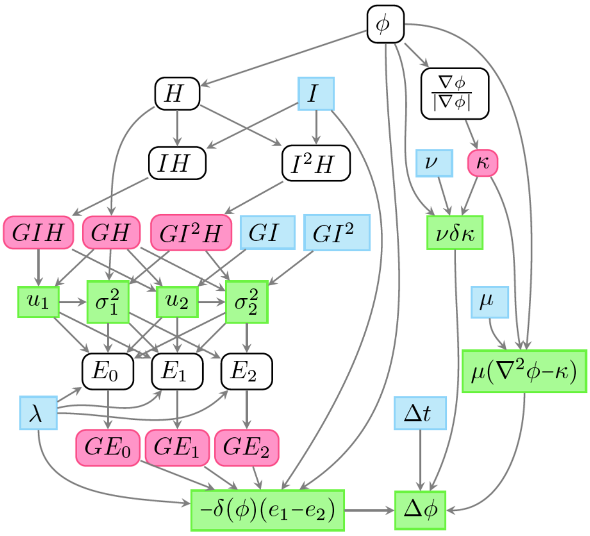


Figure 3.2: Dependency graph between variables in the update process. The red variables require neighborhood computations whereas the blue variables represent constants. All variables except for the parameters ν, μ, λ and Δt are spatially varying fields. The green variables are quantities that are computed 'on the fly' and never stored in a texture.

using the following formulas:

$$u_1 = \frac{GIH}{GH}\,, \qquad \sigma_1^2 = \frac{GI^2H}{GH} - u_1^2\,, \tag{3.11}$$

$$u_2 = \frac{GI - GIH}{1 - GH}\,, \qquad \sigma_2^2 = \frac{GI^2 - GI^2H}{1 - GH} - u_2^2\,. \tag{3.12}$$

For σ_i^2 we have used the alternative variance formula $\mathrm{Var}[X] = E[X^2] - E[X]^2$, and for u_2 and σ_2 we have used $G_\sigma * (1 - H) = 1 - G_\sigma * H$ in the denominators, where $G_\sigma*$ denotes convolution with a Gaussian kernel of standard deviation σ. This is not to be confused with σ_1 and σ_2, the local intensity standard deviations outside and inside the contour. By exploiting these tricks we are able to compute Equations 3.11 - 3.12 using only three convolutions per update cycle (since GI and GI^2 are constant). To compute the image force term $e_1 - e_2$, we expand the brackets in

Equation 3.10 to get:

$$\begin{aligned}
e_i(\vec{x}) = &\int_\Omega \omega(\vec{y}-\vec{x}) \left[\log(\sigma_i(\vec{y})) + \frac{u_i(\vec{y})^2}{2\sigma_i(\vec{y})^2}\right] \mathrm{d}\vec{y} \\
&- I(\vec{x}) \int_\Omega \omega(\vec{y}-\vec{x}) \frac{u_i(\vec{y})}{\sigma_i(\vec{y})^2} \mathrm{d}\vec{y} \\
&+ I(\vec{x})^2 \int_\Omega \omega(\vec{y}-\vec{x}) \frac{1}{2\sigma_i(\vec{y})^2} \mathrm{d}\vec{y} \qquad (3.13) \\
&= G_\sigma * \left[\log(\sigma_i(\vec{y})) + \frac{u_i(\vec{y})^2}{2\sigma_i(\vec{y})^2}\right] \\
&- I(\vec{x}) \left[G_\sigma * \frac{u_i(\vec{y})}{\sigma_i(\vec{y})^2}\right] + I(\vec{x})^2 \left[G_\sigma * \frac{1}{2\sigma_i(\vec{y})^2}\right] . \qquad (3.14)
\end{aligned}$$

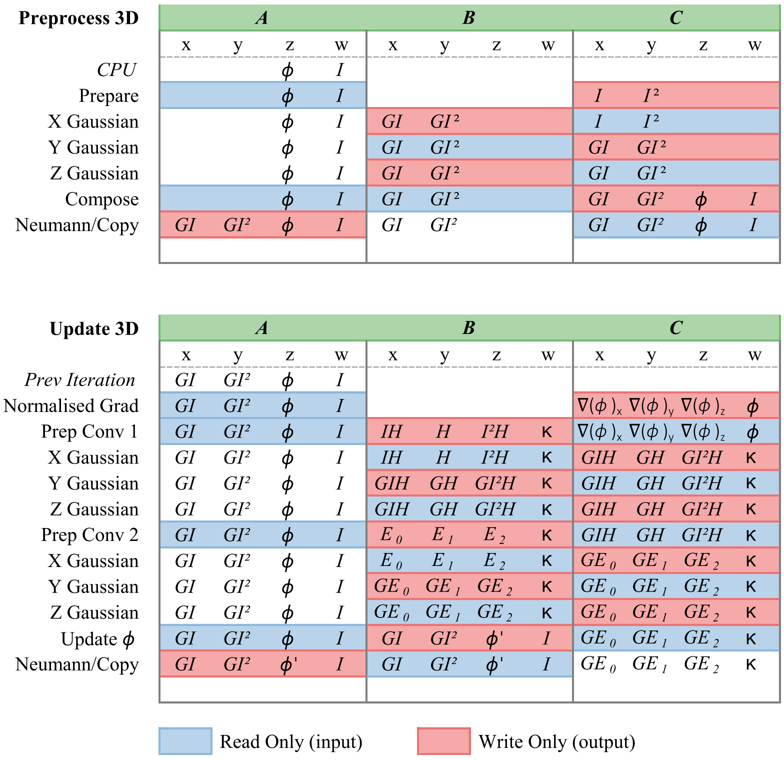
To compute the three terms in Equation 3.14, we first pre-compute the operands of the Gaussian convolutions (E_0, E_1 and E_2 in Figure 3.2), then convolve them (GE_0, GE_1 and GE_2 in Figure 3.2), then weight them by 1, I and I^2 and sum them. This results in just six convolutions altogether. Note that e_1 and e_2 are not computed separately; the variables E_0, E_1 and E_2 are the three corresponding parts of $e_1 - e_2$.

3.3.2 GPU Architecture

The six required Gaussian convolutions require a large number of buffer reads. However, an n-dimensional Gaussian filter can be separated into the matrix product of n vectors allowing us to convolve with n 1D filters instead of one very large n-dimensional filter. This reduces l^2 texture samples to $2l$ in 2D or l^3 texture samples to $3l$ in 3D, for a truncated Gaussian kernel of length l. Therefore our overall algorithm complexity is $O(n \cdot l)$ for an input of size n.

The buffer reads for the horizontal Gaussian pass are coalesced, but for the vertical and depth passes the reads are not coalesced and therefore very slow. This could be alleviated by transposing the image between convolutions, making the buffer reads coalesced for vertical and depth passes. However, transposing the image three times per convolution is slow, even when this is optimized by using local/shared memory. In our architecture we instead make use of texture memory, which preserves spatial locality among neighbouring pixels in all three dimensions, making access time for all three passes comparable to coalesced buffer reads. This allows us to skip the transpositions altogether and convolve up to four images at once in the available texture memory channels, yielding faster overall performance than local/shared memory approaches.

Texture memory buffers must either be read-only or write-only within a given kernel function; therefore, results computed from data in a texture buffer must be written to a different buffer. The memory layout for our architecture includes kernels for the separable X, Y and Z Gaussian passes accordingly, which we show in Figure 3.3. This figure lists our kernels in the order they are called and shows their inputs and outputs (corresponding to the nodes in Figure 3.2) within the available 4×32-bit channels per GPU texture buffer. Besides the convolutions, the rest of our implementation is straightforward; we store the 1D convolution filter weights in constant memory and all intermediate values reside in registers.



Preprocess 3D	***A***				***B***				***C***			
	x	y	z	w	x	y	z	w	x	y	z	w
CPU			ϕ	I								
Prepare			ϕ	I					I	I^2		
X Gaussian			ϕ	I	GI	GI^2			I	I^2		
Y Gaussian			ϕ	I	GI	GI^2			GI	GI^2		
Z Gaussian			ϕ	I	GI	GI^2			GI	GI^2		
Compose			ϕ	I	GI	GI^2			GI	GI^2	ϕ	I
Neumann/Copy	GI	GI^2	ϕ	I	GI	GI^2			GI	GI^2	ϕ	I

Update 3D	***A***				***B***				***C***			
	x	y	z	w	x	y	z	w	x	y	z	w
Prev Iteration	GI	GI^2	ϕ	I								
Normalised Grad	GI	GI^2	ϕ	I					$\nabla(\phi)_x$	$\nabla(\phi)_y$	$\nabla(\phi)_z$	ϕ
Prep Conv 1	GI	GI^2	ϕ	I	IH	H	I^2H	κ	$\nabla(\phi)_x$	$\nabla(\phi)_y$	$\nabla(\phi)_z$	ϕ
X Gaussian	GI	GI^2	ϕ	I	IH	H	I^2H	κ	GIH	GH	GI^2H	κ
Y Gaussian	GI	GI^2	ϕ	I	GIH	GH	GI^2H	κ	GIH	GH	GI^2H	κ
Z Gaussian	GI	GI^2	ϕ	I	GIH	GH	GI^2H	κ	GIH	GH	GI^2H	κ
Prep Conv 2	GI	GI^2	ϕ	I	E_0	E_1	E_2	κ	GIH	GH	GI^2H	κ
X Gaussian	GI	GI^2	ϕ	I	E_0	E_1	E_2	κ	GE_0	GE_1	GE_2	κ
Y Gaussian	GI	GI^2	ϕ	I	GE_0	GE_1	GE_2	κ	GE_0	GE_1	GE_2	κ
Z Gaussian	GI	GI^2	ϕ	I	GE_0	GE_1	GE_2	κ	GE_0	GE_1	GE_2	κ
Update ϕ	GI	GI^2	ϕ	I	GI	GI^2	ϕ'	I	GE_0	GE_1	GE_2	κ
Neumann/Copy	GI	GI^2	ϕ'	I	GI	GI^2	ϕ'	I	GE_0	GE_1	GE_2	κ

🟦 Read Only (input) 🟥 Write Only (output)

Figure 3.3: Memory layout of our GPU kernels for the 3D case. Each row represents a kernel operating on 4-channel texture objects A, B, C. The kernels read variables from one or two of the textures (blue) and write into a single texture (red).

The three Gaussian convolutions of the image and Heaviside (GIH, GH, GI^2H, Figure 3.2) are the result of neighborhood operations, but are not dependent on each other. This is also the case with the three Gaussian convolutions GE_0, GE_1, GE_2. We therefore create kernels shown in Figure 3.3 to perform each set of three Gaussian convolutions simultaneously, and two more kernels to prepare for them (called 'Prep Conv 1' to compute H, IH, I^2H, and 'Prep Conv 2' to compute E_0, E_1, E_2). The curvature field κ (Equation 3.9) requires all three (two in 2D) gradient components to be first stored in texture memory in order to avoid race conditions, since all differential operations are computed by central finite differences, a neighborhood operation. This is why we compute κ early on and pass it through the Gaussian convolution kernels in the conveniently available w channel of the texture buffer; computing κ immediately before 'Update ϕ' would require an extra texture buffer since there is only one unused channel at that point. After updating, we force the partial derivatives of ϕ to be zero at their corresponding image boundaries (in the 'Neumann/Copy' kernel) to prevent numerical instability, and copy the result back into buffer A for the next iteration.

3.3.3 Interactive Brushes

There are many applications in the biosciences, computer vision, medical, and pattern recognition communities where guidance by human experts is required [66, 97, 119, 120, 126]. The current interactive GPU level set methods, such as [95], provide interfaces to (1) initialize ϕ inside/outside the object, (2) dynamically adjust parameters, and in some cases (3) allow ϕ to be edited (a union operator on new objects/regions, followed by rerunning of the algorithm); however, it is difficult to refine evolution such as to prevent contour leaking or constrain the evolution. The graph-cuts and radial-basis function approaches [107, 120] allow users to sketch lines or define control points which are tagged to both the desired object and the undesired regions, but we find the process difficult to refine where the segmented boundary lies somewhere between the input locations, where there may not be discernible image intensity features (see Figure 3.4 top-left).

To address these issues, we follow the strategies outlined in the survey [127] with similar functions to the modeling/graphics literature [128]; however, we closely integrate brush functions with our segmentation kernels with the goal of editing and constraining ϕ during the iterative evolution process itself. Specifically, we provide functions to initialize, append, erase, and constrain (locally stop evolution of ϕ) after each iteration of the update step (Equation 3.8), and visualise the results after each iteration. Note that for simplicity we define our functions with circular (2D) or spherical (3D) regions, but there is nothing to prevent implementing more bespoke functions, such as surface pulling [128].

All brush functions are centered at the mouse position $\vec{p}$ with radius r, and are implemented in the 'Compose' kernel (Figure 3.3). We have deliberately arranged the read buffer B to link to

ϕ from the previous update iteration. To complete a brush action, we relaunch the 'Compose' kernel with the brush parameters followed by the 'Neumann/Copy' kernel between each update iteration. The initialization brush sets ϕ to a binary step function with a small positive constant (we choose 2 empirically):

$$\phi(\vec{x}) := 2 \cdot \operatorname{sgn}(\|\vec{x} - \vec{p}\| - r)\,, \tag{3.15}$$

where := denotes assignment. The user can continue to 'paint' new foreground regions using the additive brush:

$$\phi(\vec{x}) := \begin{cases} \phi(\vec{x}) & \text{if } \|\vec{x} - \vec{p}\| - r > 0 \\ \min(\|\vec{x} - \vec{p}\| - r, \phi(\vec{x})) & \text{otherwise}\,. \end{cases} \tag{3.16}$$

To erase a foreground region, we simply reassign any values inside the brush region with a small positive constant:

$$\phi(\vec{x}) := \begin{cases} \phi(\vec{x}) & \text{if } \|\vec{x} - \vec{p}\| - r > 0 \\ 2 & \text{otherwise}\,. \end{cases} \tag{3.17}$$

However, while the erase brush is useful for undoing undesired strokes, it will not stop the contour from leaking into undesired regions, as ϕ will continually update and burst through the previously erased region again. Therefore, we introduce a 'barrier' brush to persistently block the level set from growing into a fixed region. Rather than define this region in another buffer, we set ϕ to ∞ and check for ∞ values when computing $\Delta\phi$ in the 'Update ϕ' kernel:

$$\phi(\vec{x}) := \begin{cases} \phi(\vec{x}) & \text{if } \|\vec{x} - \vec{p}\| - r > 0 \\ \infty & \text{otherwise}\,, \end{cases} \quad \text{(compose kernel)} \tag{3.18}$$

$$\Delta\phi(\vec{x}) := \begin{cases} 0 & \text{if } \phi(\vec{x}) = \infty \\ \Delta\phi(\vec{x}) & \text{otherwise}\,. \end{cases} \quad \text{(update } \phi \text{ kernel)} \tag{3.19}$$

In our implementation, we found it useful to allow users to pause and unpause evolution with $\Delta t = 0$ and $\Delta t = 0.1$, while still allowing users to commit brush strokes. This makes it easier to guide the contour without having to compete against its growth. Furthermore, by using the previous value of ϕ stored in the B buffer z-channel in combination with the rendered value of ϕ stored in the A buffer z-channel, we can display the currently brush size and position without committing the stroke.

In Figure 3.4 we illustrate two simple use-cases of our interactive brushes. In the top row, the user paints using the 'barrier' brush to cover the full image region, shown in blue. This is followed by the 'erase' brush (Equation 3.17), to cut a permissible region in which a new seed region is

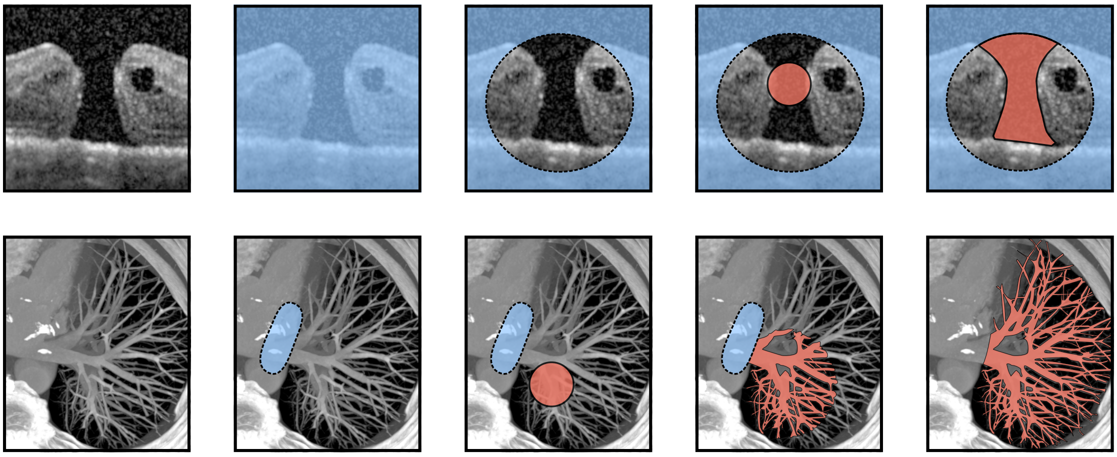


Figure 3.4: Figure illustrating interactive use of our brush functions. The blue region represents the barrier brush $\phi = \infty$ and red regions are where $\phi < 0$ and otherwise $\phi > 0$.

placed (Equation 3.16), which evolves to segment the macular hole without leaking into the opening (we show this in 3D in the accompanying video). Similarly, in the lower row, the vessels are segmented without leaking into the heart (see also Table 3.5 2b-c).

3.3.4 Real-Time Rendering

To render the zero-crossing of the level set function ϕ in 3D, we launch a render kernel after the Neumann/Copy step in the update loop (Figure 3.3). We send a camera matrix to initialize each pixel with a ray origin $\vec{o}$ and direction unit vector $\hat{d}$. We parameterize the ray's position by $\vec{r} = \vec{o} + \hat{d}s$ and, assuming ϕ to be the signed distance to the zero-crossing, advance the ray in steps by $s_{i+1} = s_i + \phi(\vec{r})$. However, ϕ is not a perfect signed distance function; therefore, we must divide our step size by the maximum derivative of ϕ; this value is not known precisely, but in practice we find we can obtain sufficiently small visual artifacts at good performance by choosing a constant step size $\Delta s = 0.3\phi(\vec{r})$. Further, given that ϕ is not defined outside of the image boundaries, we initially advance s_0 to the start of the image axis-aligned bounding box (where the s_0 is calculated using an analytical ray-box intersection function [129]). To increase visual quality, we implement 3D ambient occlusion and soft-shadows by marching the ray in the directional of the normal and light source once it has hit a surface [130].

The output of our real-time rendering implementation, using hardware trilinear interpolation to sample ϕ and with $\Delta s = 0.3\phi(\vec{r})$, is shown in Figure 3.5 (the render kernel has negligible impact on performance).

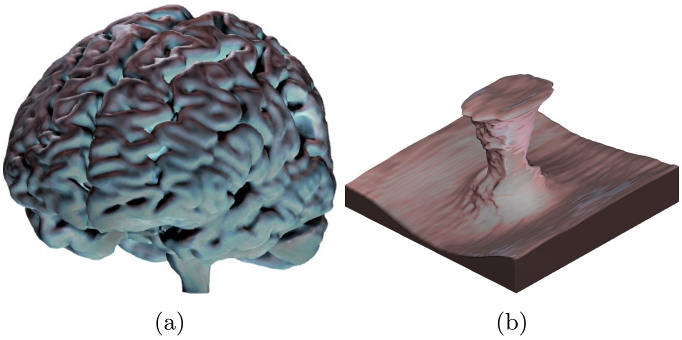


Figure 3.5: 3D views during segmentation rendered in real-time. (a) 3D segmented brain. (b) 3D segmented macular hole.

3.4 Results and Validation

In this section we provide quantitative results validating our algorithm's performance, parameter insensitivity, and robustness to noise. We also provide qualitative results to justify the utility of our interactive brushes and assess the segmentation of real-world images from various domains.

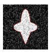
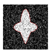
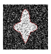
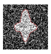
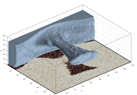
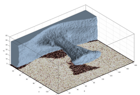
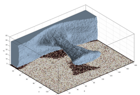
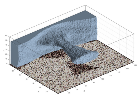
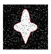
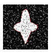
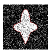
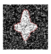
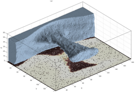
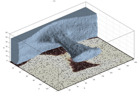
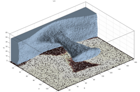
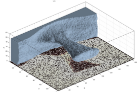
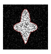
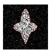
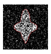
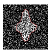
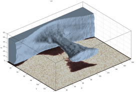
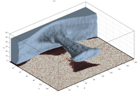
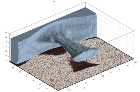
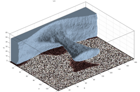
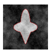
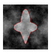
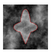
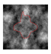
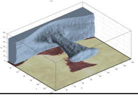
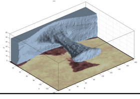
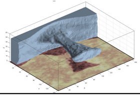
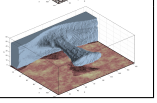
To confirm that our algorithm implements the LGDF energy model correctly, we measure the Jaccard index between the resulting segmentations from the original sequential CPU implementation and our GPU implementation, and show the results in Table 3.1.

Table 3.1: Comparing the Jaccard index for our GPU implementation with the CPU implementation.

Image	Jaccard index
Synthetic Objects 2D	1
Tumour (small) 2D	1
Tumour (large) 2D	0.981
Macular Hole 3D	0.990
Brain 3D	0.984
Tumour 3D	0.993

These results show the GPU to be near-identical to the CPU implementation; we find small discrepancies at the boundary at sub-voxel precision caused by different implementations of low-

Table 3.2: Segmentation without interactive brushes attained from a single circular seed region inside the object.

	PSNR				PSNR			
	15	12.5	10	7.5	15	12.5	10	7.5
Gauss								
Salt and Pepper								
Speckle								
Clouds								

level math library functions and different (mathematically equivalent) algebra in the intermediate steps (Equations 3.11 and 3.12).

3.4.1 Noise and Parameter Insensitivity

We conducted a large number of noise experiments on a synthetic 2D object, which has sharp and smooth features, and plot the mean and standard deviation of the results in Figure 3.6. These experiments all use the same parameters and initialize ϕ to a small circle inside the synthetic object. We also qualitatively show a subset of the experiments in Table 3.2 from the same synthetic 2D object, and for a 3D macular hole [25].

The results in Figure 3.6 show that the method can segment severely noisy images, corrupted with a PSNR of about $10^{1.05}$, under a constant parameter assignment. While the results in Figure 3.6 show the method is more robust to Gaussian noise than speckle noise, it is important to understand that this is only within the parameters chosen; improvements can generally be made by adjusting the parameters for individual scenarios. In addition to Gaussian, salt and pepper, and speckle noise, we implemented a multi-frequency 'cloud' noise at a target PSNR, which simulates intensity inhomogeneity. In Figure 3.6, it appears that the cloud noise improves under a PSNR of $10^{0.81}$; however, this is caused by the cloud-like objects inside the synthetic object being captured. In such cases, we can still segment the underlying object, but only through decreasing σ or using the interactive brushes.

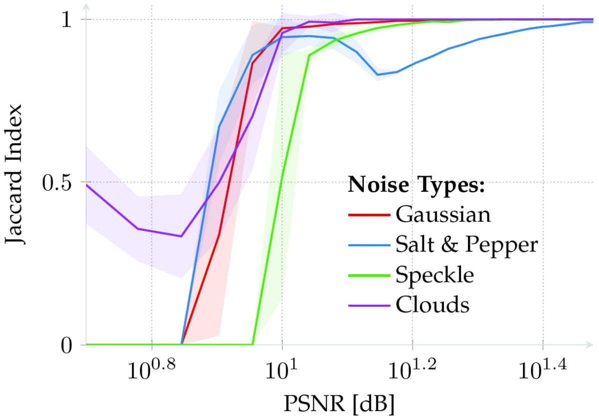


Figure 3.6: The Jaccard index of a synthetic ground-truth segmentation and our segmentation result using the same parameters on 4 different types of noise. The standard deviation is shown by the error envelopes (transparent shaded regions); our method is robust to several noise types heavily corrupting the object to a PSNR of about $10^{1.05}$.

By systematically adjusting the parameters to maximize the mean Jaccard index over all noise types, we found the following defaults: $\sigma = 3$, $\nu = 50$, $\lambda_1 = 1$, $\lambda_2 = 1.05$, $\Delta t = 0.1$, $\mu = 1$ (these are the parameters used in Figure 3.6 and Table 3.2). We also found, through our synthetic experiments and in segmenting real-world images, that across all of the encountered images we only need to adjust σ, ν, and λ, where $\lambda_1 = 1 + \max(0, -\lambda)$ and $\lambda_2 = 1 + \max(0, \lambda)$. To make these parameters more intuitive, we assign more meaningful descriptions to them in Table 3.3.

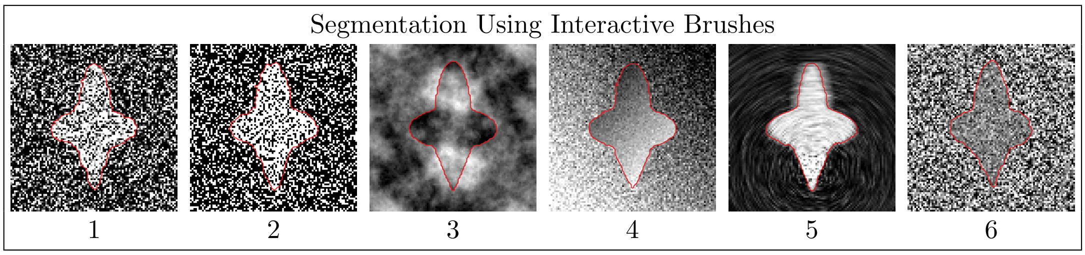
Table 3.3: Our proposed parameters for controlling the method. All images in this paper are generated using these three parameters within their suggested range and constants $\Delta t = 0.1$ and $\mu = 1.0$.

Description	Symbol	Suggested Range	Default
Capture Range	σ	$[1.01,\ 10]$	3
Smoothing Weight	ν	$[10,\ 90]$	50
Shrink or Grow	λ	$[-0.1,\ 0.1]$	0.05

We call σ a ‘capture range’ parameter as it describes the range from which a pixel’s energy may be affected by the contour (see Equations 3.2-3.4), and therefore determines the capture range. The parameter ν penalizes the length of the contour (Equation 3.6 and 3.8); a larger ν value results in a smoother contour which is less likely to burst through small gaps or capture small/sharp features. Traditionally many active contour methods have been designed to grow or shrink until they reach the object boundary and then stop; the parameter λ optionally enables this behaviour by weighting the image terms e_1 and e_2 by λ_1 and λ_2 respectively (Equation 3.8), biasing the contour towards shrinking or growing. By adjusting these parameters in real-time, inexperienced users quickly learn to intuitively manipulate them in combination with our interactive brushes. In most cases, we set $\lambda = 0.05$ to prefer contour growth, and adjust only σ and ν.

To further justify the importance of our interactive brushes, we construct 6 extreme synthetic scenarios in Table 3.4. Images 1-3 show Gaussian, salt and pepper, and cloud noise corrupted to a severe PSNR of 5 (fail cases in Figure 3.6). By adjusting the parameters and constraining the contour with our brushes, we can easily (3-5 seconds per image) segment the underlying object. Images 4-5 show that the LGDF energy can segment noisy objects with intensity inhomogeneity and weak/blurred edges. Image 6 shows an object whose intensity mean is the same as its background, with the only difference being in intensity variance.

Table 3.4: The following challenging scenarios are quickly and easily segmented with our interactive brushes. Images 1-5 have a PSNR of 5 for Gaussian, salt and pepper, and multi-frequency noise accordingly, and images 4-6 show extreme scenarios of poorly defined and/or blurred boundaries.



3.4.2 Segmenting Real-World Images

We evaluate our software against several different imaging modalities using real-world images and show the results in Table 3.5. In all our results, we only adjust the parameters σ, ν, and λ as described in Table 3.3. By initializing $\phi(\vec{x}) = 2$ uniformly, we are able to automatically segment small objects without an initial seed region, such as for images of cells. This works because $\delta(2)$ is large enough that ϕ can still be deformed by image forces, allowing new segments to appear anywhere in the image; this is not possible with a narrow band approach. In general the default parameters suggested in Table 3.3 work well for most object segmentations; however, in challenging cases (such as multiple objects or thin objects) the parameters σ and ν can be dynamically adjusted in real-time where the user can 'slide' the parameter within the suggested range until the motion of the contour is satisfactory to achieve the desired result.

Many of the segmentations (Table 3.5 1a, 3a-b, and 5b-c) are not possible with the current GPU level set segmentation approaches, which use simple speed functions to attract and/or shrink the contour within a fixed intensity range [94, 95, 113]. For example, when painting an initial seed region inside a vessel network with intensity inhomogeneity, the active contour will not grow along the vessel. In contrast, the adopted LGDF energy model allows us to paint a simple initial sphere anywhere on the object which then spreads through the network of vessels. In cases where the contour evolution misses a vessel or oversegments part of the object, evolution is temporarily halted ($\Delta t = 0$), local amendments are made, and then evolution is resumed ($\Delta t = 0.1$). By making local adjustments with a high level of visual feedback, we can spot such issues and make amendments immediately.

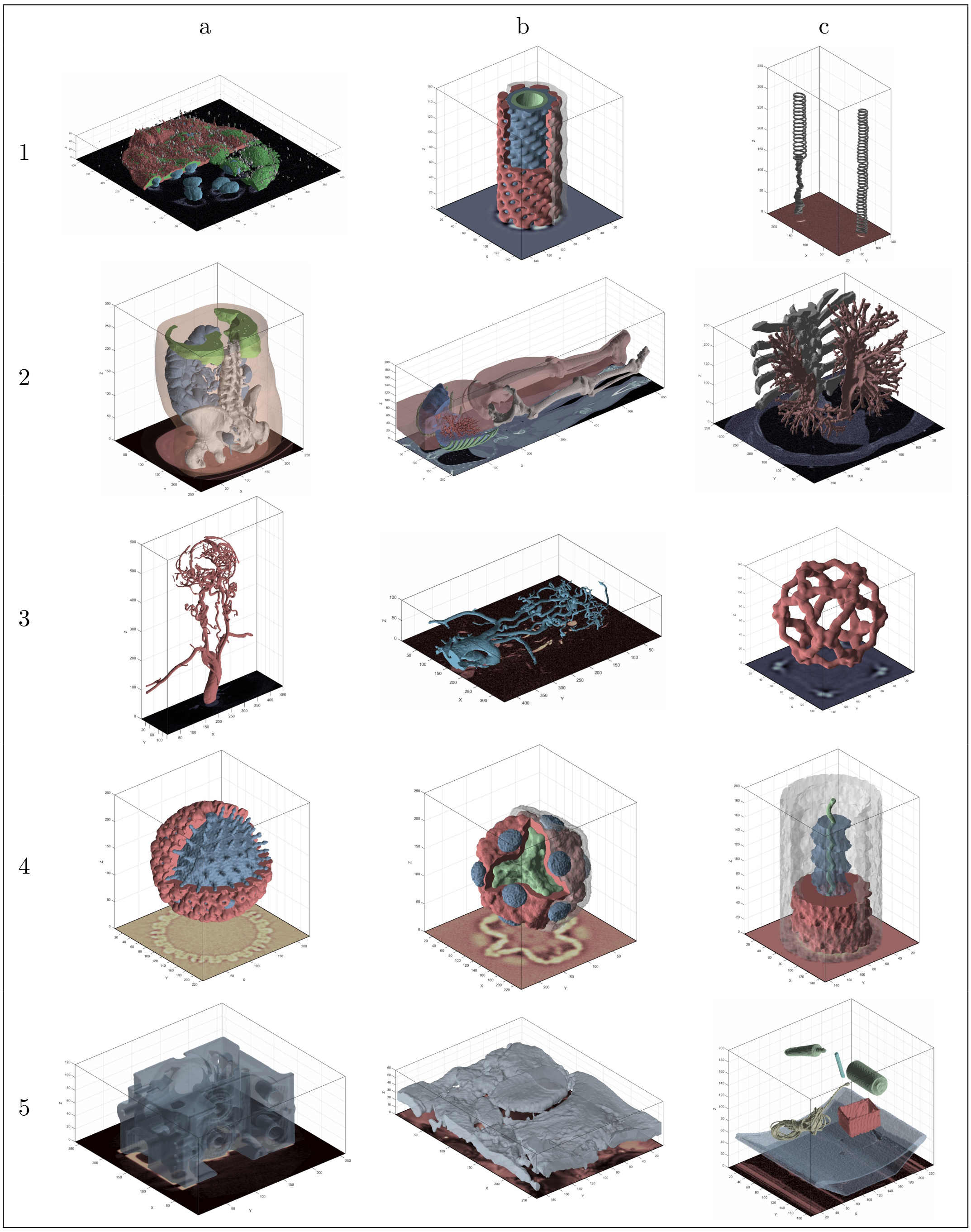
3.4.3 Performance and Memory Usage

In our cross-platform C++/OpenCL application, we measure the mean kernel timings over 100 frames for different sized images on a GTX TITAN X and show the results in Figure 3.7. We can see that the overall algorithm performance is approximately linear in the number of pixels/voxels, since we process the full dataset as the C^{∞} Heaviside and Dirac functions are non-zero everywhere. This agrees with our expected complexity of $O(n \cdot l)$ for an input of n voxels and a truncated 1D Gaussian kernel of length l.

Figure 3.8 shows how the overall running time increases with larger σ and that the performance in the z-axis becomes more similar to the y- and x-axes with larger σ. In the practical and suggested range of σ [1.01, 10] (Table 3.3), it can be seen that the running time increases in small steps (zoom to the lower-left of the graph). This is because running time is primarily influenced by the size of the 1D Gaussian filter buffer, whose size is $\lfloor 4\sigma + 1 \rfloor$ to approximate the Gaussian function with reasonable support.

We also investigated other optimizations given that the Gaussian convolution is the primary bottleneck of our approach. We implemented Gaussian convolution in the Fourier domain using MATLAB GPU arrays. While Fourier convolution allows for a lower order of growth, the benefits are outweighed by the large constant factor due to the algorithm complexity; this takes 400ms per frame using a GTX TITAN X, which is off the scale in Figure 3.8.

Table 3.5: Segmentation results of multiple objects displayed in different colours. 1a shows a segmented image of HaCaT human cell culture cells using confocal microscopy, 1b shows the interdigitation of segmented layers of eisosome proteins from cryo-EM tomography data [131], 1c shows a malaria sporozoite [132]. Row 2 shows medical CT scans of the abdomen, body, and thorax [100]. 3a shows an MRI of a cerebral aneurysm, and 3b an XA angiogram [100]. 3c shows the structure of the Sec13/31 COPII coat cage from cryo-EM data [133]. Row 4 shows the herpes simplex virus capsid [134], phi procapsid [135], and the mumps virus [136], all from cryo-EM data. Row 5 shows applications outside of biology and medicine: 5a is a CT scan of an engine block [137], 5b sintered alumina [132], and 5c shows a selection of objects from a CT scan of a backpack [137].



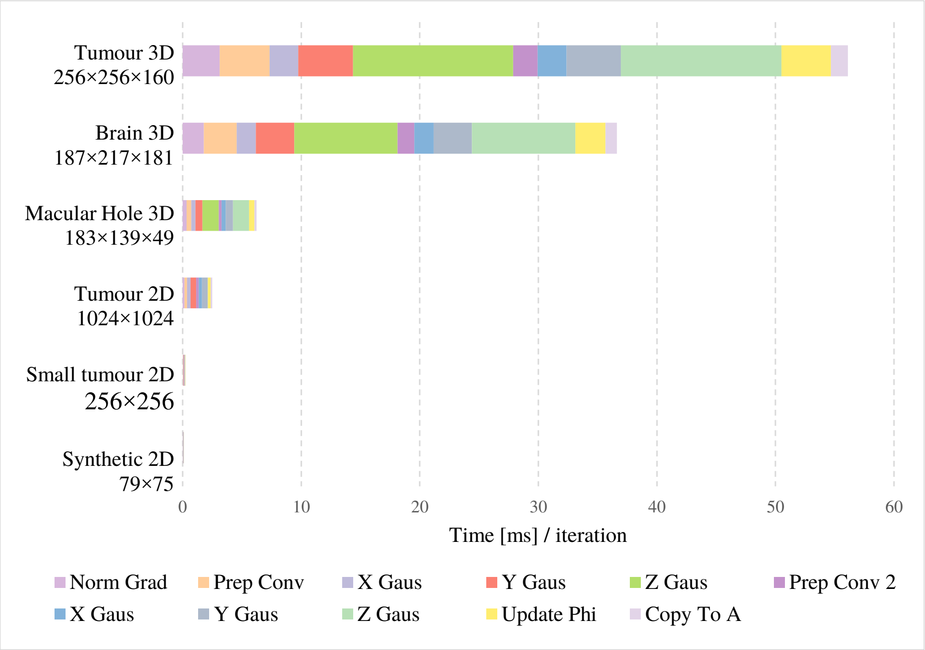


Figure 3.7: Mean kernel timings over 100 frames for different images of different sizes. $\sigma = 3$ in all cases. Despite using texture memory, which is cached and has spatial locality in multiple dimensions [39], and fast constant memory to store the 1D separable Gaussian coefficients, convolution in the z-axis is significantly slower than the y and x axes.

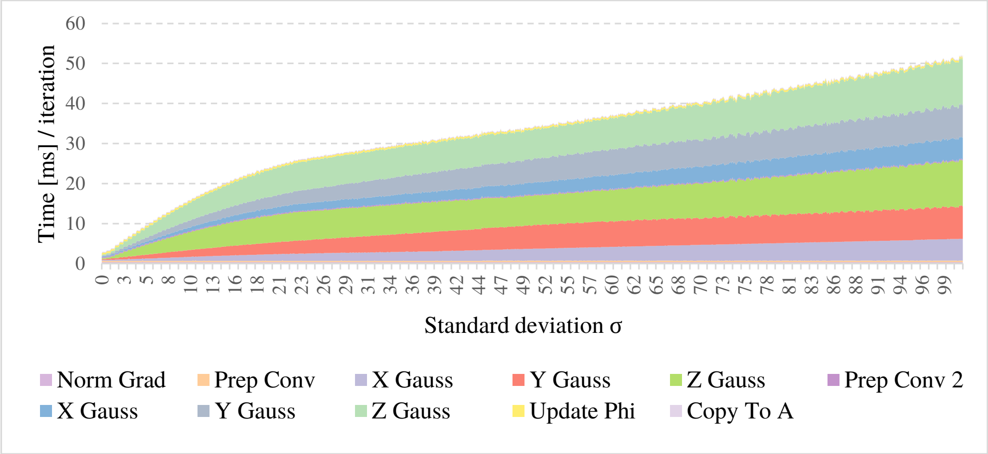


Figure 3.8: Mean kernel timings over 100 frames with increasing σ for 3D macular hole. In practice we rarely require $\sigma > 10$.

The mean time of 100 iterations with our C++ OpenCL implementation is evaluated across different hardware and compared to our GPU Fourier implementation and the original MATLAB version on the CPU (which is vectorized and calls code written in C for the Gaussian convolution). These results are shown in Figure 3.9.

In Figure 3.9, our algorithm substantially outperforms the original implementation in all images. Given that we process the entire dataset with compact kernels and separable convolutions, we can fully utilize high-end GPU hardware to obtain a substantial speedup of up to three orders of magnitude from the original version, and 1-2 orders of magnitude from our GPU Fourier convolution version. This means that segmentations which previously took over an hour can now be achieved in a few seconds, without any trade in quality.

With high-end GPU hardware, our algorithm is limited by memory consumption. We require 48 bytes of texture memory per pixel or voxel for the entire image (4 bytes per channel in Figure 3.3). In cases where the image does not fit into the available GPU memory, we must either downsample or crop the region of interest before segmentation.

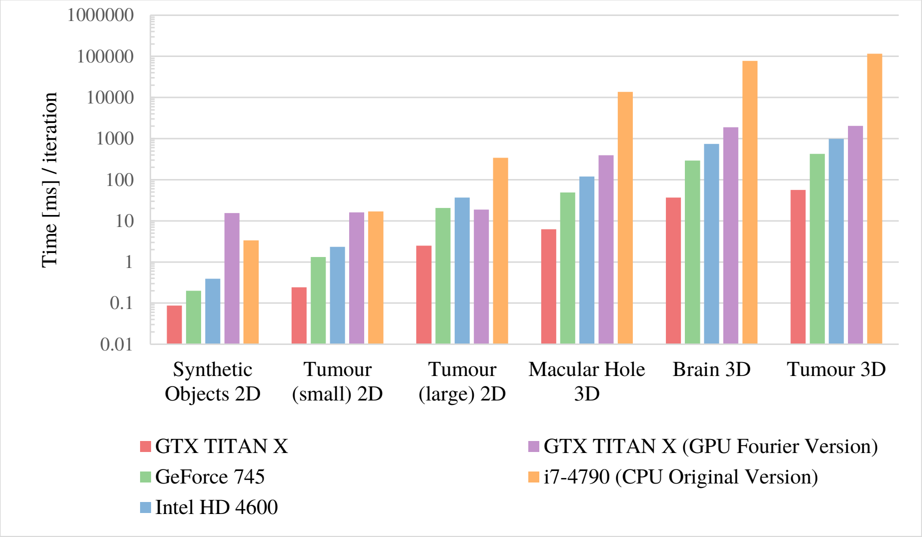


Figure 3.9: Mean time [ms] over 100 iterations on different GPU hardware, compared to the original MATLAB implementation and our implementation using fast Fourier convolution on the GPU. Our OpenCL implementation achieves over a $\times 10^3$ speedup over the original vectorized MATLAB version in larger images, scaling up with the parallel hardware accordingly.

3.5 Discussion

The primary limitation of our implementation is that we require storing the full dataset at the original resolution in GPU texture memory, as the C^∞ Heaviside and Dirac functions are non-zero everywhere to reduce convergence on local minima [65]. This also limits the algorithm's speed. In future work, we will investigate dynamically adjusting the resolution away from the zero-crossing of the C^∞ Heaviside, to reduce the memory requirements and improve performance, and evaluate the impact of this approach on segmentation quality.

While there are some excellent publicly available datasets for interactive segmentation of real-world 2D colour images and videos [106], the problem of segmenting everyday objects in colour photographs, e.g., with a graph cut approach on distributions of colour information, is fundamentally different to segmenting a tissue or organ. In the latter case, the challenge is more often due to intensity inhomogeneity or poorly defined edges, rather than complex backgrounds or discontinuities within the object. As with [109], we would like to see benchmark 3D biological and medical datasets for evaluating interactive performance.

3.6 Conclusion

In conclusion, we have shown that sophisticated level set segmentation energy models, with sequential dependencies amongst intermediate processing steps, can be implemented efficiently on the GPU through careful structuring of the GPU kernels within the constraints of the GPU memory architecture. While active contours are used in unsupervised algorithms, they continue to benefit from interactive approaches that enable users to guide and constrain the contour to capture specific parts of more challenging objects. We have shown that the LGDF energy model proposed by [57] requires little parameter tuning, is robust against different types of noise, and can be generalized to a broad range of real-world 3D images from biology and medicine. Segmenting many of these images was not possible with existing GPU level set algorithms due to their simple energy functionals. We have greatly enhanced the LGDF model's performance, making it practical in many more use-cases than before (including 3D images). We also extended its functionality through interactive brush functions that give direct influence over the dynamic contour evolution. In the future, we believe GPU adaptations of advanced segmentation algorithms will continue to proliferate, using similar design processes to ours.

3.7 Availability

We release our C++/OpenCL software and source code under the GNU General Public License Version 3, alongside an optional MATLAB wrapper. The implementation is cross-platform using GLFW with few dependencies, where binaries for Linux and Windows are also available [138].

Epilogue

In this chapter, a faster GPU implementation of local Gaussian distribution fitting (LGDF) has been presented. It is applicable to a variety of real-world datasets including macular hole datasets.

Chapter 4

Macular Hole Morphology and Measurement Using an Automated Three Dimensional Image Segmentation Algorithm

Prologue

This chapter introduces macular hole morphology and measurements based on 3D shape viewing. This macular hole shape analysis is automatically retrieved from accurate 3D segmentation and measurements of the macular hole.

Declaration: This chapter is based on the following publication: Chen, Y., Nasrulloh, A. V., Wilson, I., Geenen, C., Habib, M. S., Obara, B. & Steel, D. H. W. Macular hole morphology and measurement using an automated three dimensional image segmentation algorithm. *BMJ Open Ophthalmology.* In press (2020). This chapter is presented as submitted, although referencing and notation has been altered and cross-referencing added for consistency throughout this thesis. Some stylistic changes have been made for consistency.

4.1 Introduction

Full thickness macular holes (MH) are a common cause of blindness with a prevalence of up to 0.5% in the over 60-year-old age group and are bilateral in 7-16% [41, 139]. Vitrectomy surgery is an established and successful treatment, with Ocriplasmin and expansile gas also effective in a lower proportion of selected patients [56, 140]. Macular holes are classified partly by the presence of vitreoretinal adhesion at the fovea and optic disc but principally by their size [42]. Indeed, size is used to guide the choice of treatment, the optimum surgical approach and to predict outcome. A variety of size measures have been described with minimum linear diameter being used to divide holes into small, medium and large [42]. Ratios of various size parameters have also been suggested, including diameter hole index (DHI), macular hole index, and macular hole closure index [43–46]. Similarly, the difference between base diameter and the minimum linear diameter (MLD) has been shown to predict response to Ocriplasmin [47]. All these measures have typically been made using a single two dimensional (2D) slice of a horizontal optical coherence tomography (OCT) image, and measured by a human grader using calipers. This is known to be prone to high intra and inter-observer error and also vulnerable to further error from off centre scan location [141]. Furthermore, the true three dimensional measures of the macular hole are not measured, with symmetry in the x/y axis being assumed.

We have designed a three dimensional automated image processing algorithm which is able to segment macular holes with high accuracy. We describe the dimensions and morphology of a consecutive cohort of 104 macular holes from patients prior to surgery, and compare them to clinician acquired measurements in 2D.

4.2 Method

The spectral domain OCT images of a consecutive cohort of patients assessed for vitreoretinal surgery for idiopathic primary full thickness macular hole over a two-year-period in a single eye hospital, were prospectively collected as part of routine care and retrospectively analysed. Secondary, myopic, and persistent holes after previous surgery were all excluded, as were eyes with axial lengths of less than 22 $[mm]$ and greater than 25.5 $[mm]$. The fellow eyes of patients already included were excluded. All had undergone spectral domain optical coherence tomography (SD-OCT) imaging using the Heidelberg Spectralis (Heidelberg, Germany) as part of routine care, using the same imaging protocol. A high density central horizontal scanning protocol with 30 $[\mu m]$ line spacing was used in the central 10 by 15 [degrees]. All scans used a 20 automatic real time setting enabling multisampling and noise reduction over 20 images. Prior to image export two independent experienced clinicians measured the minimum linear diameter (MLD) and maximum base diameter (BD). Observer 2 also measured hole height as previously described and the height above the inner surface of the retinal pigment epithelium (RPE) at which the minimum linear diameter was measured. MLD was defined as the horizontal minimum hole diameter in the approximate mid zone of the hole away from any operculum, in the OCT slice with the widest dimensions.

The presence of any vitreomacular attachment was noted. In the case of any of the measurements being greater than 15% different between the two observers' measurements, the two observers were asked to independently check their measurements to ensure no transcription or identity errors had occurred. Differences in the presence of vitreomacular adhesions (VMA) were arbitrated by a third experienced grader. The volume of the hole was calculated using the volume of a truncated cone formula as previously used [63], and diameter hole index as MLD/BD [44].

Each person's scan was exported as a folder of anonymised non-compressed .tiff files with the accompanying data file containing the image information including the x, y and z axis pixel/[μm] conversion ratio. Basic demographic data accompanied each image including age and gender.

The images were then analysed using an automated 3D segmentation algorithm as previously described. The system uses a state-of-the-art level set method based on the local Gaussian distribution fitting (LGDF) energy functional, employing a 3D multi-scale approach. This is followed by a novel curvature-based surface cutting procedure, which separates the macular hole from its background, allowing for fully-automatic measurement of the shape and volume [1]. We have previously shown that the method is stable to a variety of different macular hole shapes and more accurate than other existing graph cuts segmentation approaches, with an accuracy of segmentation of 99.19% (Table 2.7 in Chapter 2) as compared to a ground truth manual segmentation approach by an experienced clinician. The procedure is also highly repeatable.

A 3D model of the macular hole was produced with the following axes (Figure 4.1):

- x: the axis along the base of the macular hole in the horizontal line scan.
- y: the axis representing the retinal height, from the RPE to internal limiting membrane (ILM).

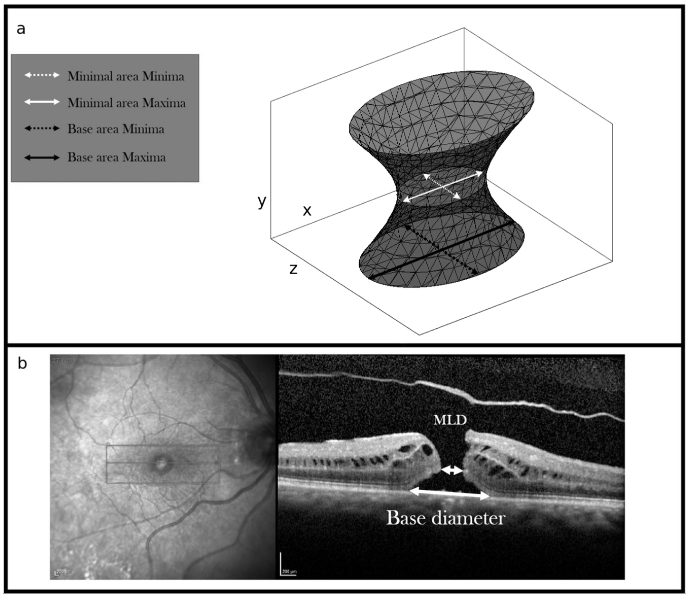


Figure 4.1: Schematic diagram of the MH 3D model (a) and SD-OCT of a MH with 2D labels (b).

- z: the axis across the macular hole slices and at right angles to x and y.

The following parameters were derived from the 3D model and expressed in $[\mu m]$ units:

1. The maximum and minimum diameters of the base area (BA) of the hole i.e. the area of the hole in the plane of the RPE, and the axes of the maxima in the xz and xy planes of the scan. The maximum diameter represents the base dimension most commonly measured clinically.

2. The maximum and minimum diameters of the minimum area (MA) of the hole defined as the minimum area in the central 20% - 90% of the hole height. The minima of the MA represents the minimum linear diameter (MLD) as used clinically and in the international vitreo-macular interface (VMI) classification. The meridian of the minimum axis in the xz and xy planes were also measured as well as the height of the minimum area as measured perpendicularly above the RPE.

3. The total surface area and volume of the extracted 3D macular hole shape measured in pixel areas and voxels respectively.

4. The diameter hole index defined as the minimal dimension of the minimum area divided by the maximal dimension of the base area.

All data and scans were collected as part of routine care and fully anonymised, and as such under UK guidelines the investigation was categorised as service evaluation and did not require ethical approval. Preoperative visual acuities were recorded either using Snellen charts or early treatment diabetic retinopathy study (ETDRS) letter charts and converted to logarithm of minimal angle resolution (logMAR) visual acuity for analysis.

4.2.1 Statistical Analysis

Descriptive and statistical analysis was performed using R [142] and plots using ggplot2 packages [143].

Macular hole variables are presented in terms of mean, standard deviation (SD) and range when normally distributed, and percentage as appropriate. Distribution plots are given for a variety of parameters.

Association between continuous data were assessed using correlations and between categorical data using two sample t-tests. Stepwise multiple regression was used to analyse the effect of multiple variables. Statistical significance was considered with a p-value of 0.05 or less.

4.3 Results

Images and clinical data on 104 eyes from 104 patients were analysed. The mean age was 70-years-old (SD 6.6, range 48-84), 85 (82%) were female and 52 (50%) were right eyes. Vitreomacular traction (VMT) was present in 27 (26%).

4.3.1 3D Image Analysis

The parameters as measured by the image analysis approach are presented in Table 4.1. The mean diameter of the MA was 384.7 [μm] (SD 155.1, range 133-899 [μm]).

There was a mean difference of 54.87 [μm] between the maximal and minimal dimensions of the MA and 87.14 [μm] for the BA, representing 17% and 12% of the mean dimensions, but ranging up to 48% and 36% respectively.

There was no clear trend for these meridian differences to vary in extent in holes of different sizes. (See Figure 4.2)

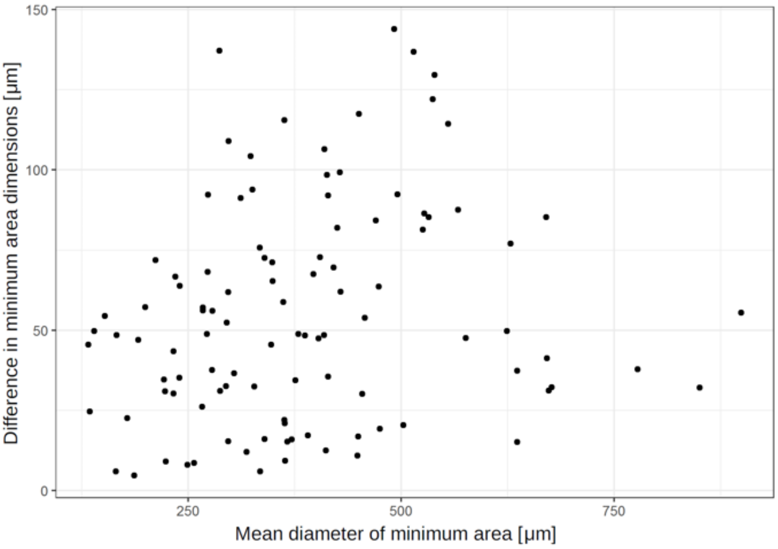


Figure 4.2: Difference in the maximal and minimal dimensions of the MA compare to the mean diameter of the MA.

The mean angle to the x axis of the minimal dimension of the MA was approximately 90 [degrees] in the xz plane i.e. at nearly right angles to the horizontally acquired SD-OCT scan, with a slight tilt downward. Only 10 of the 104 images had a minimal dimension within 10 [degrees] of the horizontal. There were small variances, mostly within 2.5 [degrees] in the minimal dimension in the xy plane. (Figure 4.3)

By distinction the maximum dimension of the BA was approximately in line with the x axis, with 40 of the 104 images (38%) being within 10 [degrees] of the horizontal.

The minimum dimensions of the MA were related to the maximum of BA with a quadratic relationship. The rate of increase of the maximum BA was smaller relative to the rate of increase of the minima of the MA particularly for larger holes. The fit of the quadratic (shown as solid line) was better than the linear fit (shown as dashed line with slope beta = 0.5, p=0.004)(Figure 4.4).

The height of the MA increased with the width of the MA. Wider holes had MAs that were higher from the RPE than narrower ones. (Figure 4.5)

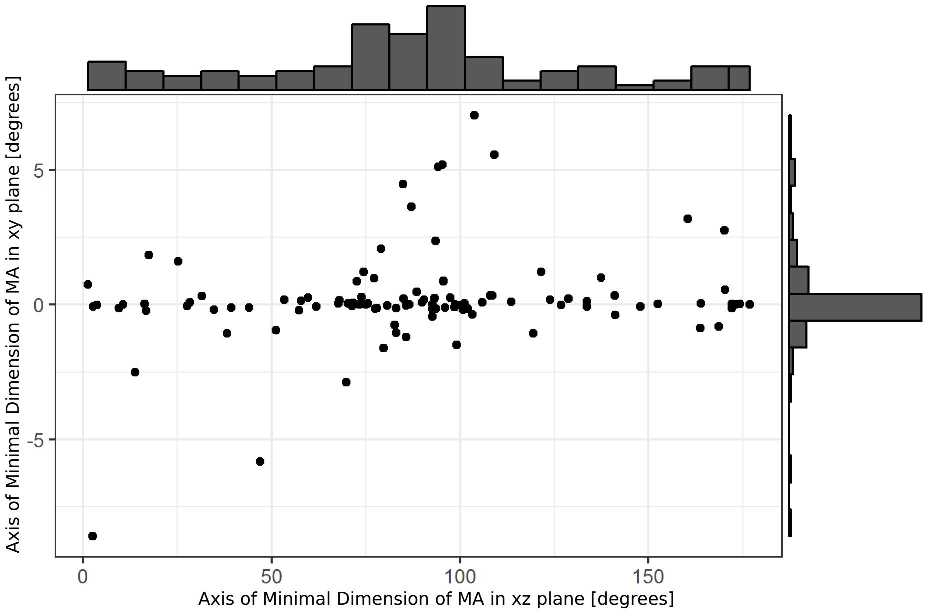


Figure 4.3: Axis of minima of minimum area in xz plane compare to its axis of in xy plane.

Table 4.1: Baseline parameters of MH derived from image analysis algorithm.

Parameter	**Mean**	**SD**	**Minimum**	**Maximum**
Retinal height at centre of Hole [μm]	366.28	52.73	223.11	487.65
Minimum dimension of MA [μm]	357.27	152.77	110.26	871.27
Maximum dimension of MA [μm]	412.14	159.17	146.91	926.70
Height of centre of MA above RPE [μm]	189.14	64.11	50.31	341.45
Difference between maxima and minima of MA [μm]	54.87	34.01	4.71	143.92
Difference expressed as a percentage of mean dimension of MA [%]	15.69	10.05	1.78	47.89
Angle in xz plane of axis of minima of MA [degrees]	89.58	44.15	1.85	175.73
Angle to xy plane of MA [degrees]	-0.37	2.09	-6.98	9.97
Minima of BA [μm]	716.09	250.42	147.58	1,408.65
Maxima of BA [μm]	803.23	267.13	192.57	1,474.14
Difference between maxima and minima of BA [μm]	87.14	47.41	15.29	266.73
Difference expressed as a percentage of mean dimension of BA [%]	12.22	6.57	1.79	36.21
Angle in xz plane of BA maxima [degrees]	4.36	33.12	-86.49	85.32
Surface area [mm^2]	1.66	0.81	0.25	4.28
Volume [$\times 10^{-3} mm^3$]	0.74	0.48	0.068	2.36
Diameter hole index	0.45	0.12	0.22	0.81

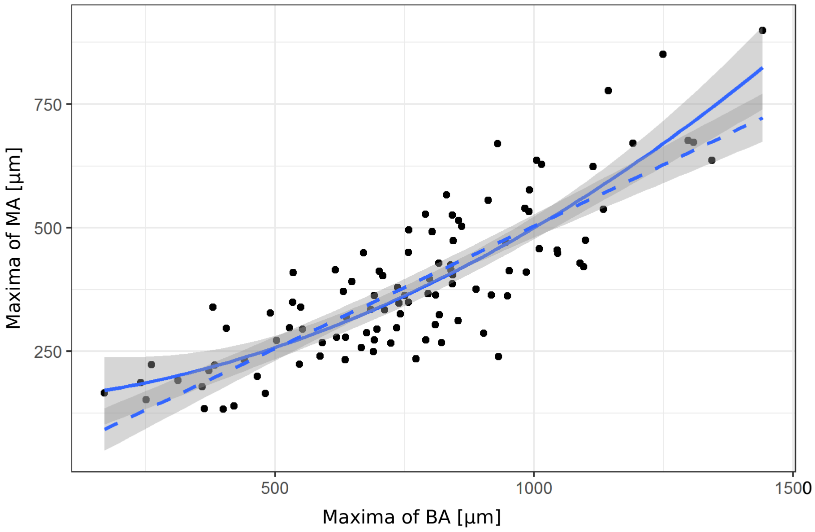


Figure 4.4: The relationship between the mean dimension of BA and the mean dimension of MA of the MH.

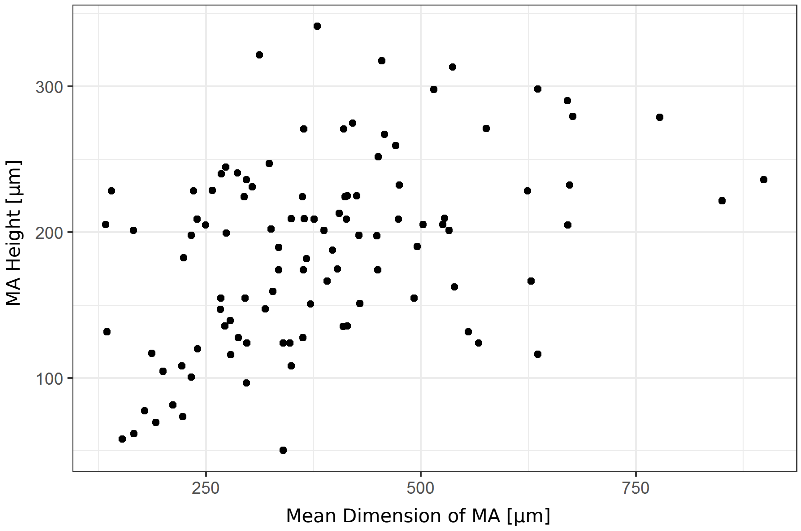


Figure 4.5: Mean width of MA compared to the height of MA of the MH.

Several of the macular hole dimensions were highly collinear in particular the MA, BA, surface area (SA) and volume. DHI correlated poorly to most other parameters with the exception of a weak association with MA. Retinal height similarly was weakly correlated with BA, volume and SA and negatively correlated with DHI. (Figure 4.6)

4.3.2 Macular Hole Shape

Heat maps with colour coding of the number of overlapping hole outlines in the xy, xz and yz profiles are shown in Figure 4.7a, 4.7b, 4.7c, giving a representation of shape in the three planes.

To assess the vertical symmetry of the macular hole, the centre point of the hole on the inner surface was mapped to the base area (Figure 4.7d), with the centre line of the holes shown schematically in (Figure 4.7e), showing that several of the holes did not have an orientation perpendicular to the retinal surface.

Hole shaped varied widely as shown in Figure 4.8 where the holes are ordered according to BA. The presence of VMT (shown as darker shading) was not related to hole size.

4.3.3 Human Measurements and Their Relationships to Algorithm Values

The measured values for the macular holes for observers 1 and 2 are shown in Table 4.2.

Table 4.2: Observers 1 and 2 macular hole measurements based on 2D OCT images.

Observer	**Parameter**	**Mean**	**SD**	**Minimum**	**Maximum**
1	MLD [μm]	406.6	151.9	122	885
	BD [μm]	787.1	255.1	242	1416
2	MLD [μm]	403.8	179.5	105	1122
	BD [μm]	825.8	268.1	225	1507
	Hole height [μm]	389.6	55.2	240	515
	MLD height [μm]	190.5	55.8	71	341
Mean of observer 1 and 2	MLD [μm]	405.3	162.2	113.5	965
	BD [μm]	806.5	259.9	233.5	1461.5
Difference between mean of observer 1 and 2 versus algorithm derived values	MLD [μm] ([%] difference from algorithm)	+47.9 (11.9)	53.4 (13.3)	-90.8 (-26.9)	212.8 (49.2)
	BD [μm] ([%] difference from algorithm)	+87.1 (12.2)	47.4(6.6)	15.3 (1.79)	266.7 (36.2)

There was no significant difference for MLD between Observer 1 and 2 (p=0.69) but there was a significant difference in measured BD between the two observers (mean of the differences 38.75

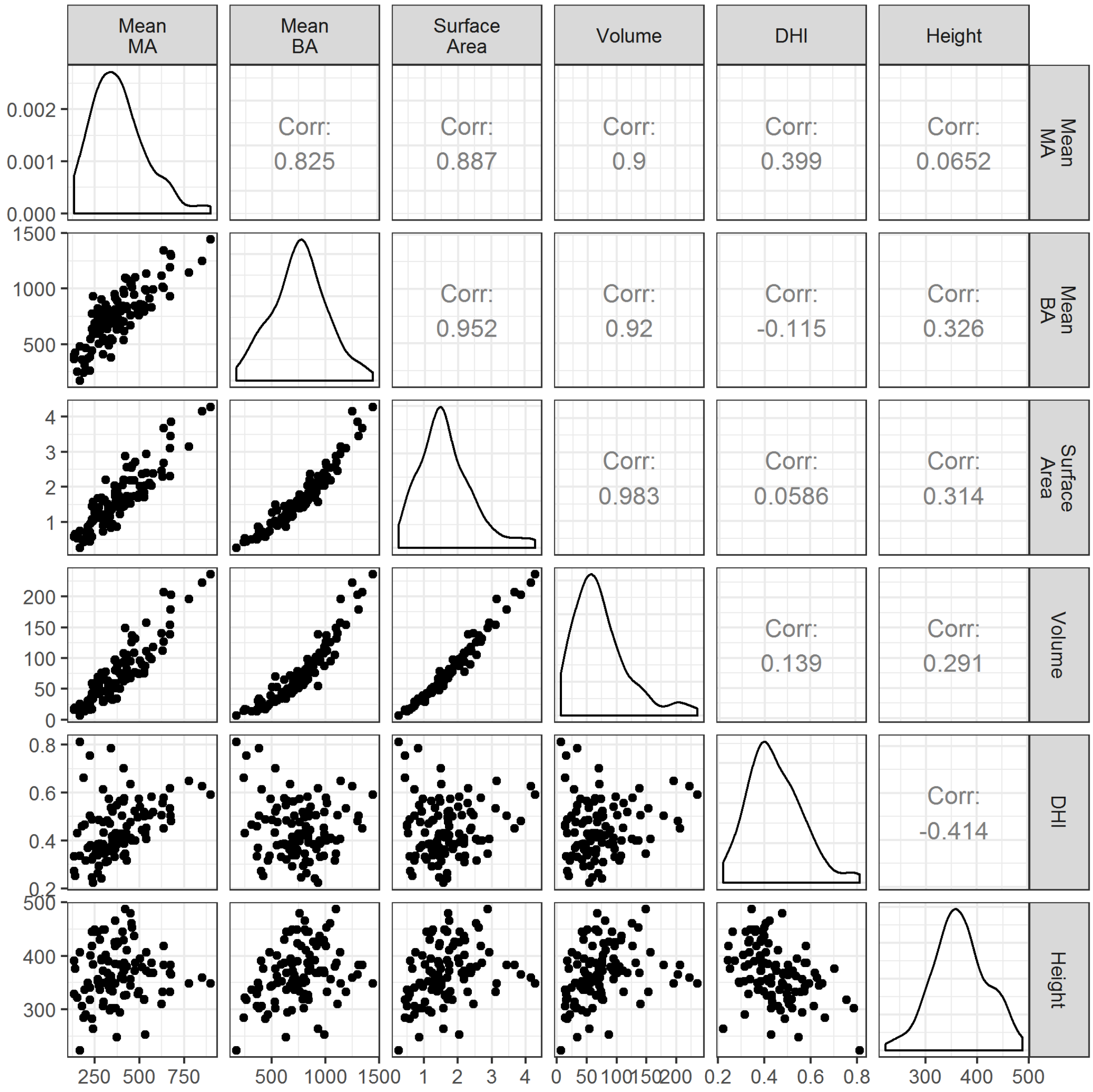
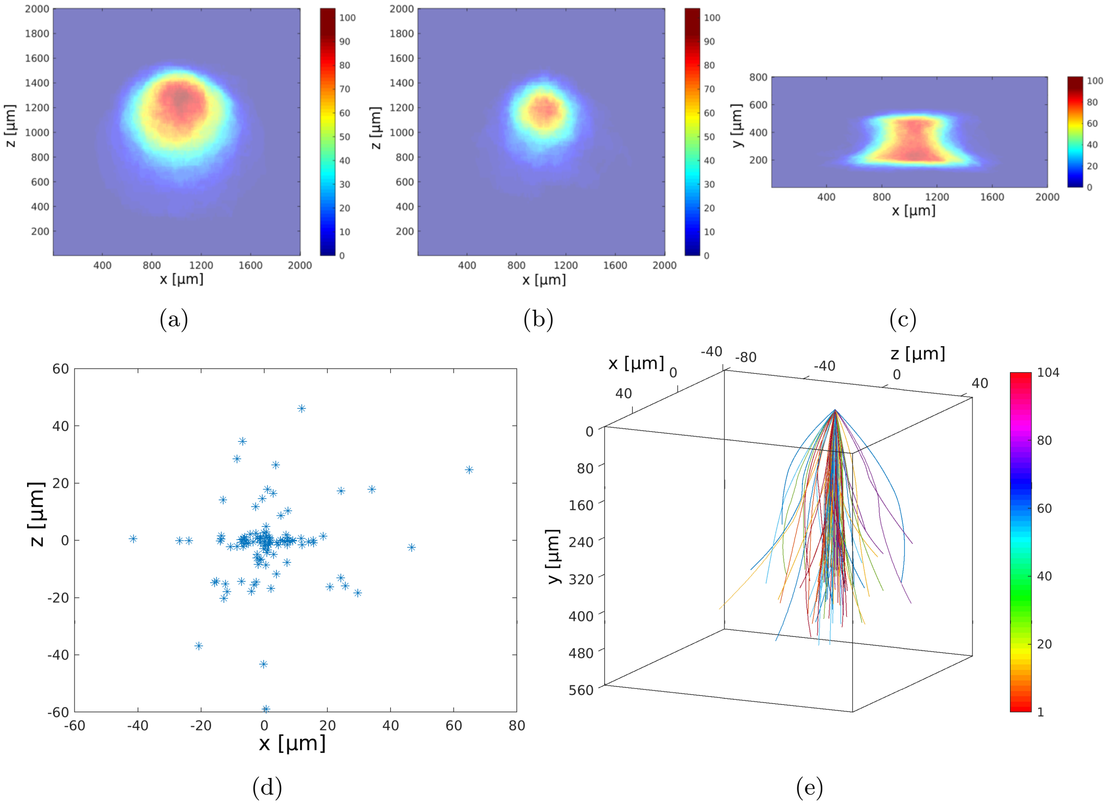


Figure 4.6: Correlation between mean dimension of MA and BA, surface area, volume, DHI and height of the MH.



(a) (b) (c)

(d) (e)

Figure 4.7: Heat maps with colour coding of the number of overlapping hole outlines in the xy, xz and yz profiles (a, b, c); mapping of centre point of hole to base area (d); centre line of hole at apex compared to retinal surface (e).

$[\mu m]$ with observer 2 overestimating BD compared to observer 1, P<0.0001). (Figure 4.9). The 95% limits of agreement between the two observers for MLD were -140.3 $[\mu m]$ (-164.6 $[\mu m]$, -115.9 $[\mu m]$) to 145.9 $[\mu m]$ (121.6 $[\mu m]$, 170.3 $[\mu m]$) and those for BD were -161 $[\mu m]$ (-181.8 $[\mu m]$, -140.2 $[\mu m]$) to 83.5 $[\mu m]$ (62.7 $[\mu m]$, 104.3 $[\mu m]$).

The mean of both observers for MLD and BD differed significantly from the algorithm acquired measurements (p<0.0001 for both) with both observers overestimating MLD and BD significantly. (Figure 4.9)

The MLD height was not significantly different to the algorithm derived values (mean 189.1 $[\mu m]$

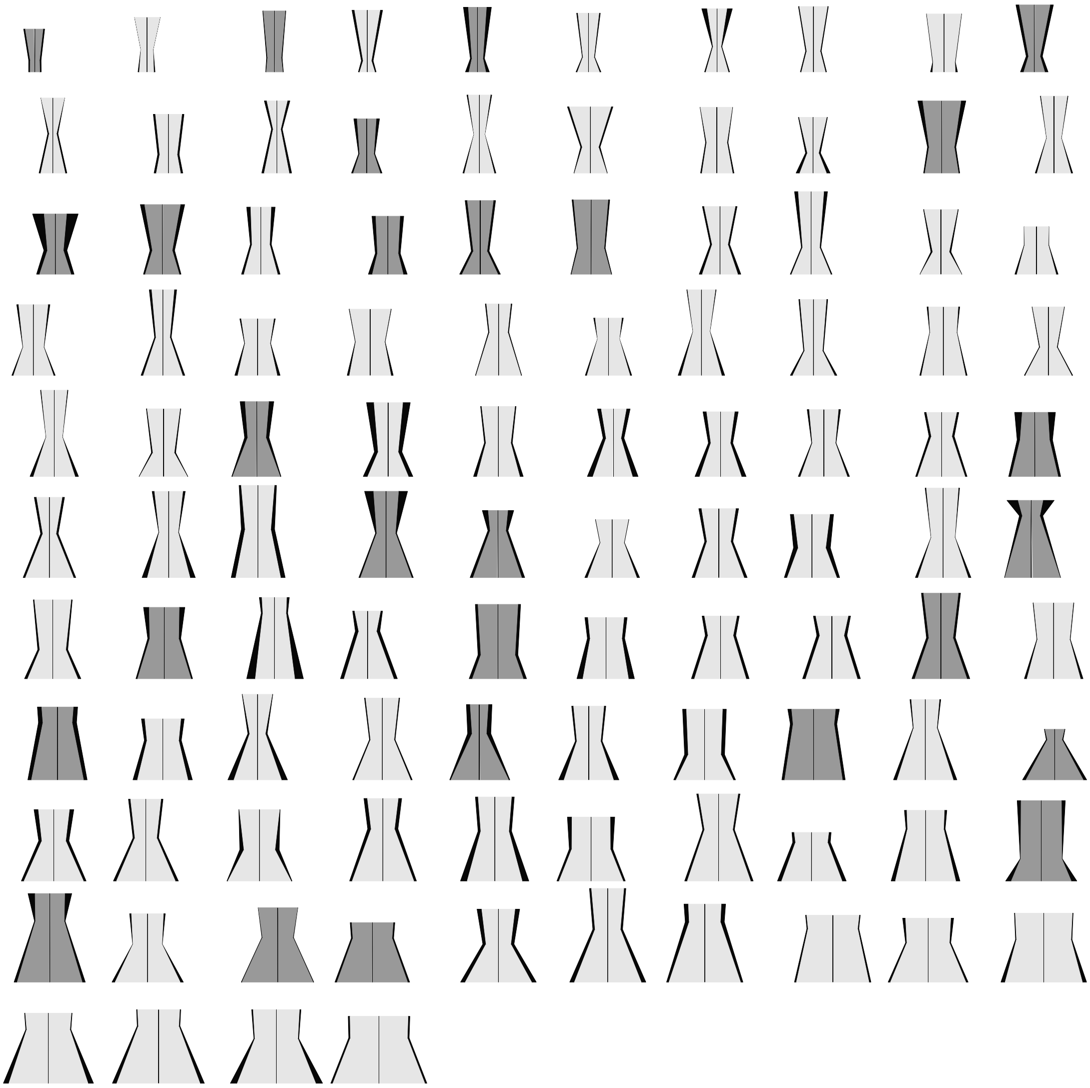


Figure 4.8: Schematic diagram of hole shapes ordered according to base area, with presence of VMT as darker shading.

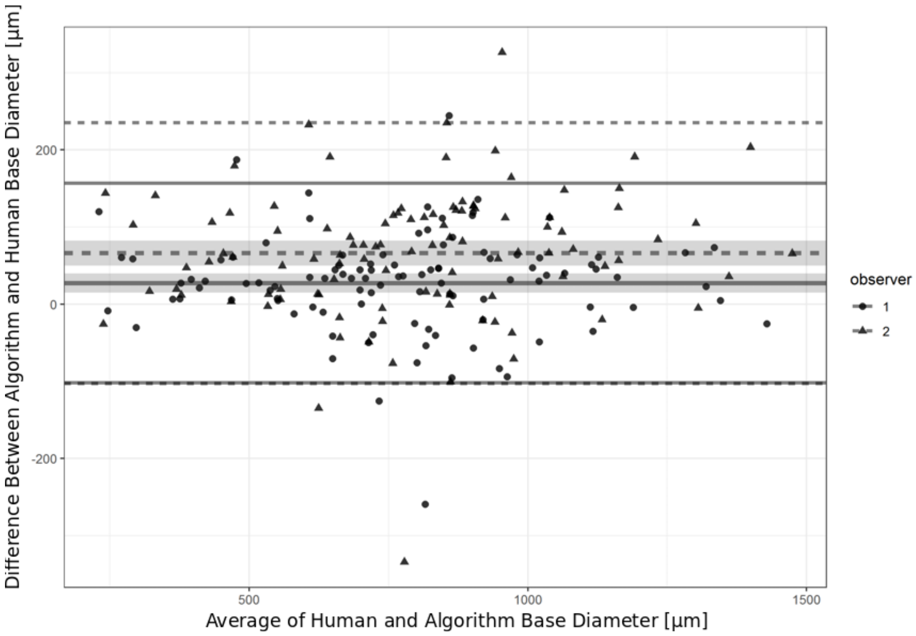


Figure 4.9: Bland Altman plots for BD diameter measurements with observer 1 against the algorithm and observer 2 against the algorithm superimposed. The 95% confidence intervals are shown for the mean differences (shaded) and 95% intervals for the differences (lines).

versus 190.5 [μm], p=0.76) but as can be seen in Figure 4.10 there was wide variability between the values.

The retinal height was significantly different to the algorithm acquired measurement. (mean 366.3 [μm] algorithm versus 389.6 [μm] Human p<0.001)

Using the human measured MLD, 16 holes were classified as small (<250 [μm]), 44 medium (251-400 [μm]) and 44 large (>400 [μm]). Using the algorithm measures of the minimum dimension of the MA as compared to the mean of the human measured MLD resulted in a change in classification in 25 of the 104 eyes, with 26 classified as small, 48 medium, and 30 large. (Table 4.3).

The human derived DHI and volume differed significantly from the algorithm values with high variability. The human observers overestimated DHI compared to the algorithm derived values. Volume was slightly underestimated using the human observers measurements but with a trend to increased variability as hole size increased. (Figures 4.11 and 4.12).

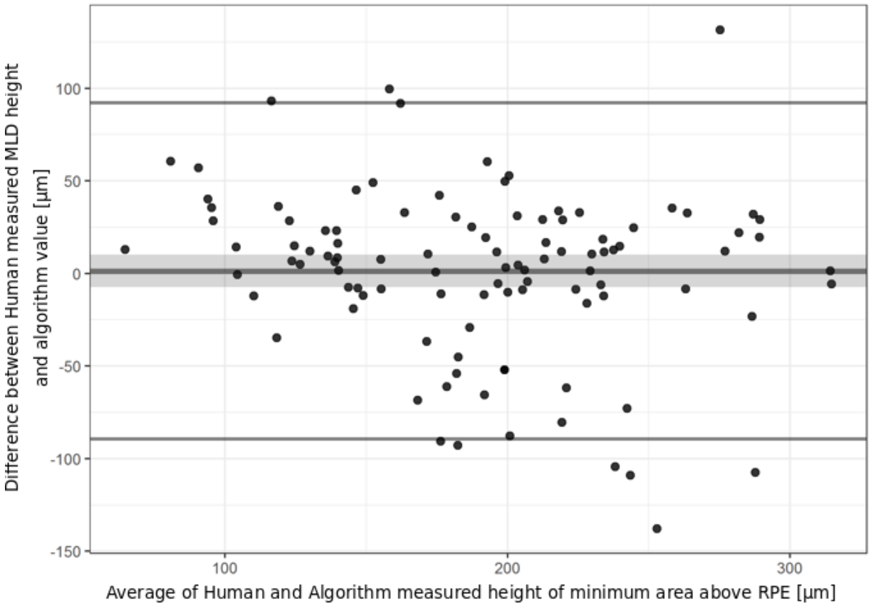


Figure 4.10: Bland Altman plots for algorithm derived measured height of minimum area above RPE and human measurements of MLD height. The 95% confidence intervals are shown for the mean differences (shaded) and 95% intervals for the differences (lines).

Table 4.3: Classification of MH based on size. Algorithm derived measurements in black, human measurements in grey.

	<250 [μm]	**250-400** [μm]	**>400** [μm]	**Algorithm**
<250 [μm]	16	9	1	26
250-400 [μm]		33	15	48
>400 [μm]		2	28	30
Human	16	44	44	104

4.3.4 Associations with Preoperative Variables

Age, gender, VMT presence and laterality were not associated with any size variable nor DHI. Preoperative visual acuity was positively associated with size both for MA $r^2 = 0.57$, higher for minimum than maximum diameter) and BA ($r^2 = 0.56$, higher for maximum diameter than minimum) as well as volume ($r^2 = 0.54$), surface area ($r^2 = 0.57$, and height of the MA ($r^2 = 0.45$) but not retinal height ($r^2 = 0.08$) nor DHI ($r^2 = 0.02$) (Figures 4.13). Using the mean human measurements, the correlation values were slightly lower at MLD (0.49) and BD (0.48).

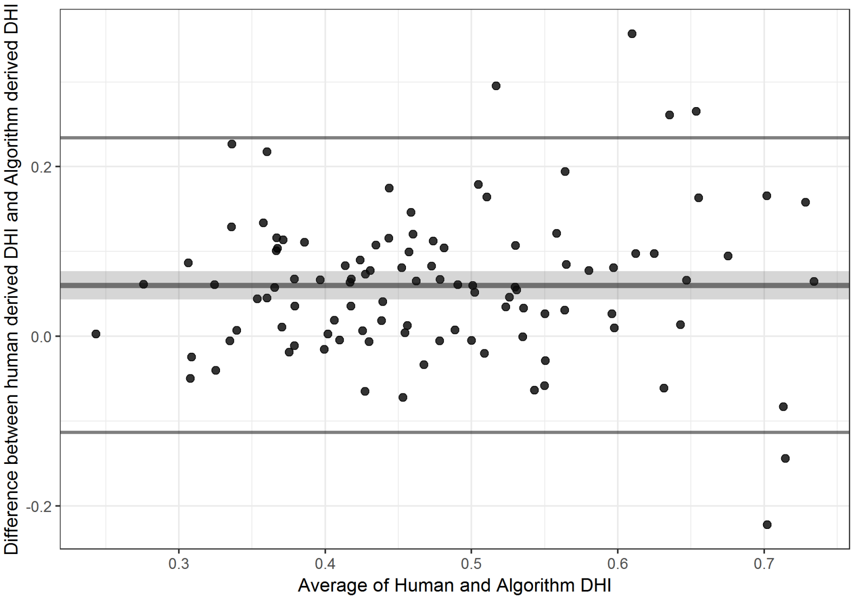


Figure 4.11: Bland Altman plots for human and algorithm derived DHI measurement. The 95% confidence intervals are shown for the mean differences (shaded) and 95% intervals for the differences (lines).

4.4 Discussion

We have described the 3D morphology of MH using a novel and validated automated 3D segmentation algorithm. The algorithm is robust and was able to accurately segment the full consecutive series of 104 SD-OCTs included in the study, including when there was VMT present. We used a high density scanning protocol with 30 [μm] line spacing and averaging 20 A scans per line, reducing noise and meaning that the scan lines were more likely to include the maximum hole dimensions [141]. We have previously shown that the 3D image analysis methodology we used can very accurately segment out the macular hole boundaries as compared to a human observer, and can therefore be regarded as providing a ground truth for macular hole dimensions and shape. MH are shown to be complex shapes with significant asymmetry, meaning that conventionally acquired clinician measurements fail to represent their key parameters accurately. For example, we found that the xz meridian of the minima of the MA was only within 10 [degrees] of the conventionally measured horizontal x axis in 10% of cases, and differed from the human measured MLD by a mean of nearly 50 [μm], and up to 200 [μm]. Similarly, the true maximum base diameter varied from the mean of the human measurements by 87 [μm] or 12%.

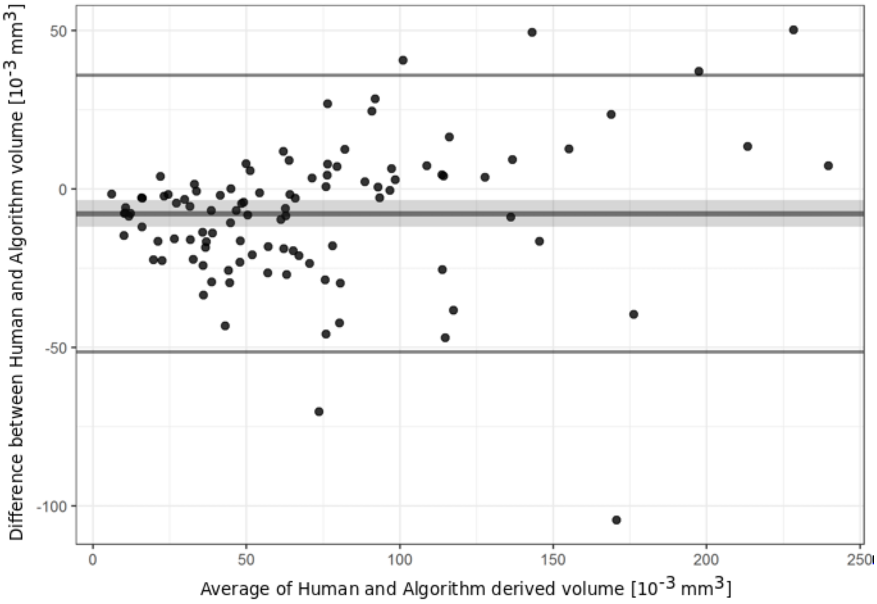


Figure 4.12: Bland Altman plots for human and algorithm derived MH volume. The 95% confidence intervals are shown for the mean differences (shaded) and 95% intervals for the differences (lines).

The resultant differences led to a reclassification in size using the International Vitreomacular Traction Study Group classification in a quarter of the patients [42, 144, 145]. This has significant implications for studies using macular hole measurements to predict outcomes and to act as cut off points for deciding on treatments.

The human measurements had a consistent tendency to overestimate the widths of the holes. To measure a macular hole MLD, a human observer must first accurately locate the scan line with the greatest dimensions and then pick the minimum hole dimension, avoiding the area of the operculum if present. The minimal dimension is typically measured parallel to the RPE. Measuring MH using a horizontal line scanning protocol relies on the MH being symmetric but we show that the holes were significantly asymmetric in all dimensions. There was a mean difference of 55 [μm] in maximum and minimum dimensions of the MA and 87 [μm] for the same measures of the BA. These differences concur with those found by Philippakis et al. [144] using en face SD-OCTs to measure MH dimensions, although they did not comment on the orientation of the maximum/minimum measurements. The minima of the MA were typically approximately

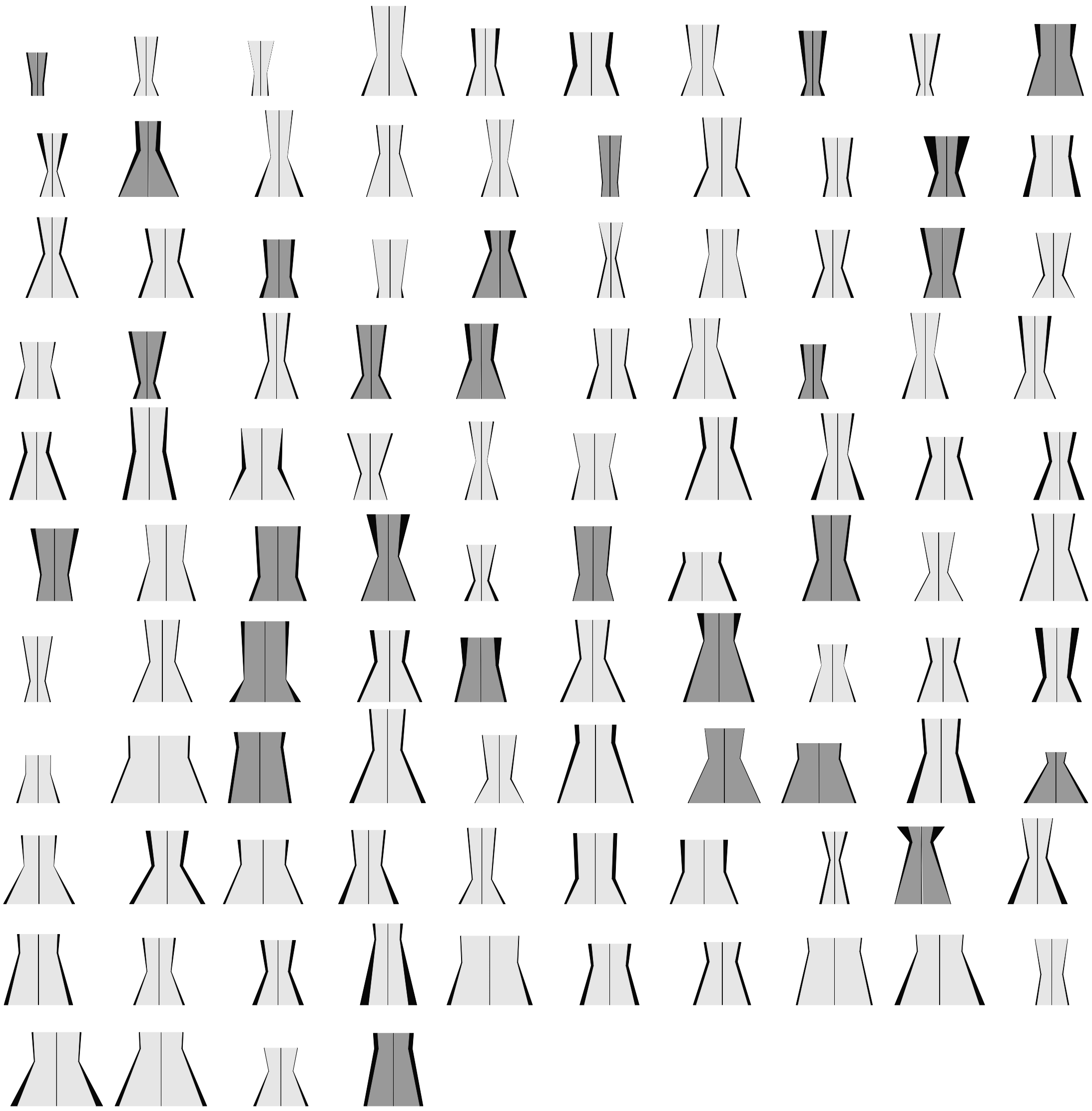


Figure 4.13: Schematic diagram representing all 104 holes by their height, base, minimal and top dimensions and ordered by preoperative visual acuity. The variability in the maxima and minima of the MA and BA is shown by the thickness of the borders of the holes. Holes with VMT are shaded in darker grey. The association between hole size and preoperative visual acuity is clearly seen with a trend towards bigger holes at the base of the diagram. The lack of association between hole size and VMT and preoperative visual acuity and VMT is also seen.

90 [degrees] to the horizontal, whilst the maximum of the BA was predominantly horizontal. The holes were therefore oval with their maximum dimension in the xz axis at the horizontal meridian. Interestingly this corresponds to asymmetries found in the foveal vascular zone (FAZ) where previous studies have found an approximate 30 [μm] difference with the horizontal diameter being widest [146]. It is known that FAZ size is closely related to foveal floor size, and a recent study has suggested an association between macular hole size and foveal floor width [147, 148].

The minimal area had a tilt to the horizontal ranging from -7 to +10 [degrees], and over 5 [degrees] in 7%. Furthermore, although the holes were generally vertical the centre point of the MA and BA were misaligned by over 150 [μm] in 70% of the eyes. This therefore adds to the measurement error of human graders who have tended to measure the MLD and maximum BD on the same SDOCT slice when in reality this occurrence will rarely occur. These asymmetries further explain the human measurement error compared to the true measurements found by the algorithm, but the two observers also varied significantly between themselves, as others have noted. The 95% limits of agreement between the two observers for MLD were -140 [μm] to 146 [μm], which is in broad agreement with the values found by Banerjee et al. [149]. We asked observer 2 to record the height above the RPE at which they measured the MLD and found that although it was not significantly different to the height the algorithm measured the minimal area at, it varied from the algorithm by more than 40 [μm] in 29% of eyes, and it is likely that this is another source of error between observers. The two observers were both experienced in measuring macular holes and from the same institution and it is likely that less experienced observers, with different training may have had even greater differences between them. If different scanning protocols and SD-OCT machines were added, then the differences would be greater again. We did not assess intra-visit variability which would have increased the variability further.

A number of derived values have been proposed to assess macular holes with including DHI and hole volume. We automatically calculated these using the algorithm and compared them to the values generated by human measures. Both significantly varied from the human derived values particularly in larger holes which might explain some of the conflicting results in the literature for their utility. A number of other novel predictive factors for hole closure and visual acuity have been suggested including area ratio factor [93] (surface area of hole/ base area) which will optimally rely on an automated segmentation algorithm as we have described. We did not attempt to measure external limiting membrane (ELM) height which has been shown to be predictive of outcome on its own [150], and could therefore also not measure the macular hole closure index which has also been shown to be useful in predicting surgical results [46].

The accurate algorithm derived measurements we have presented allow us to examine some shape relationships. We found that retinal height was only weakly associated with the BA but not with MA, and height was negatively related to DHI. In other words as holes widened at their base they became higher, and the DHI increased with height with the increase in BA. There was a quadratic relationship between MA and BA, with holes getting wider in both dimensions with an approximate 1:2 ratio, but with the rate of increase of the BA being smaller relative to the rate of increase of the MA, particularly for larger holes. It is known that macular holes enlarge with time. Two hypotheses have been proposed to account for MH enlargement and shape change. One is based on hydration of the para-hole retinal tissue [151], and another that the retina at the fovea is bistable and tangential traction after hole formation results in the hole edges everting akin to an umbrella in the wind [152]. Based on the bistable theory, BA would be expected to increase more than MA as enlargement occurs. This is not what we describe although we present a snap shot of holes rather than a longitudinal study which may have produced different results.

It is thought that most macular holes are formed by the effects of antero-posterior vitreoretinal traction and VMT. 26% of the holes in this series had VMT which is keeping with figures from the same population area that we have previously reported on [153]. We found no significant association between the presence of VMT and any of the size parameters measured which is in keeping with the findings of Philippakis, Amouyal et al. [144]. Forsaa et al. found that a small MLD combined with a large BD was inversely associated with the presence of VMT and thus hypothesised that after VMT release macular holes tended to assume a more triangular rather than rectangular shape [41]. Using our data fitting a logistical model to the presence of VMT gave a model where VMT was associated with increased MLD and decreased BD, but neither variable was statistically significant. This is in the same direction as Forsaa et al. [41], but not statistically significant. The lack of a relationship can be seen on our shape diagram, Figure 4.13.

We assessed the association of age, gender, and preoperative vision with the algorithm and human measured values. Age and gender were not significantly associated with any parameter but preoperative vision was as other authors have found [154, 155]. Interestingly the strongest relationships were all those derived from the algorithm as opposed to the human observers. However, some of the preoperative visions were checked without a protocol refraction and on a Snellen chart before conversion to logMAR and hence are clustered and exact relationships are uncertain.

Three other approaches have been suggested to evaluate macular hole shape and dimensions beyond that achieved by measuring from standard SD-OCT line scans. Philippakis et al.

elegantly demonstrated the use of en face reconstruction to measure macular hole minimal area and dimensions [144]. The technique; however, had a high technical failure rate of ∼50%, often had to be manually adjusted when VMT was present, and was unable to measure other hole parameters. Problems may also be encountered with this technique in cases where holes are misaligned vertically as we have already observed above. Geng et al. used a manual segmentation technique combined with MATLAB to produce a 3D representation of the hole from which a variety of 3D parameters could be measured but involves a very time consuming manual mark up [93]. Xu et al. have described an approach of automatically measuring macular hole dimensions based on the sum of 2D images [30]. In contrast to the solution by Xu et al., our algorithm considers the overall 3D geometry of the hole and is significantly faster. We have also validated the accuracy of the system used in this study against human segmentation in a set of 30 eyes and showed very high accuracy.

Our study has several limitations. We did not correct the measurements for axial length although we restricted the entry criteria to eyes with axial lengths between 22 $[mm]$ and 25 $[mm]$ to minimise errors introduced by this. Furthermore, inaccuracies introduced by doing this would only affect absolute measurements, not the differences in dimension we describe nor differences from the human measures. The exclusion of eyes with more extreme axial lengths limits the applicability of our results. We used a specified scanning protocol which also limits the applicability of our technique and although a consecutive cohort, our sample was restricted to patients undergoing surgery in one centre which may not be representative of all idiopathic macular holes or other ethnicities and populations.

In conclusion we present a 3D automated MH segmentation system that is able to accurately measure MH dimensions and present the detailed dimension findings from a cohort of 104 consecutive macular holes, showing significantly different results from experienced human graders. Macular hole size is known to be one of the strongest predictors of surgical success both anatomically and functionally. Evaluation of the measurements generated from this automated system, in a prospectively collected dataset of eyes undergoing surgery with outcomes analysis will be of great interest.

4.5 Implementation

All codes (Chapter 2 and Chapter 4) were implemented and written in MATLAB 2018a [83] on Linux Ubuntu 64-bit PC running an Intel Core i7-4790 CPU (3.60 GHz) with 16GB RAM and GeForce GTX 1080. The source code is available in a GitHub repository [156].

4.6 Epilogue

In this chapter, we have provided an application of an automated 3D segmentation and measurements of macular hole in the analysis of macular hole morphology.

Chapter 5

Concluding Remarks

In this thesis, new techniques have been introduced for the 3D segmentation and measurement of macular hole objects in 3D OCT images. These methods have been built upon the ideas and concepts of a multi-scale 3D level set segmentation approach based on a state-of-the-art level set method, and novel curvature-based cutting and 3D measurement procedures (Chapter 2 (*p.* 9)).

A substantially faster GPU implementation of our level set method (Chapter 3 (*p.* 35)) has been developed. It can segment inhomogeneous objects with poorly defined boundaries as often encountered in biomedical images including 3D macular hole images.

Finally, an approach has been developed for the 3D shape-based measurements and shape analysis of macular holes. Here, further application and measurements from Chapter 2 (*p.* 9) have been applied to describe the dimensions and morphology (Chapter 4 (*p.* 60)).

5.1 Contributions to the Research Field

This thesis constitutes a series of contributions to the research area of image processing and shape analysis, especially in the field of macular hole clinical applications.

In Chapter 2 (*p.* 9), we described an automatic pipeline for segmenting and measuring 3D macular holes. We introduced a 3D multi-scale active surface which is 61 times faster than the original

LGDF implementation. We introduced an automatic and novel curvature-based surface cutting procedure to separate the 3D macular hole from the vitreous body. We introduced automatic and novel procedures for capturing specific macular hole measurements, based on our robust centerline definition. We provided quantitative and qualitative validation of the algorithm's robustness and performance accross a variety of different 3D macular hole images.

In Chapter 3 (*p.* 35), we significantly increase the performance of the LGDF energy model through an optimized GPU implementation, handling much larger 2D images and even 3D images at interactive performance. We introduce a novel set of interactive brush functions that are integrated into the GPU kernels such as to modify and constrain the evolving level set in real-time, provide a ray tracer to view the segmentation results at each time-step, and expose a simpler and more intuitive parameter space to the user, with suggested values and ranges. The combination of these four enhancements greatly improves the practicality of what is already considered a state-of-the-art level set method of particular relevance to the biomedical image processing communities.

Finally, in Chapter 4 (*p.* 60), we demonstrate an application of our 3D automated macular hole segmentation system. Using a dataset of 104 consecutive macular holes, we accurately measure dimensions and present our detailed findings. As macular hole size is known to be one of the strongest predictors of surgical success both anatomically and functionally, evaluation of the measurements generated from this automated system in a prospectively collected dataset of eyes undergoing surgery with outcomes analysis will be of great interest.

5.2 Conclusions

In this thesis, we have demonstrated an automatic and robust method to segment, perform curvature information-based cuts, and extract measurements (Chapter 2 (*p.* 9)) from 3D OCT images of macular holes. Segmenting through careful structuring of the GPU kernels within the constraints of the GPU memory architecture was not possible with existing GPU level set algorithms due to their simple energy functionals (Chapter 3 (*p.* 35)). The LGDF model's performance has been greatly enhanced, making it practical in many more use-cases than before including 3D macular hole images. 3D shape-based macular hole measurement is able to robustly and accurately measure macular hole dimensions and present the detailed dimension findings (Chapter 4 (*p.* 60)).

These new techniques have been shown to work well on many different types of 3D macular hole images and can be considered a significant contribution for clinical applications.

The results of these new techniques compare to deep learning method as the current state-of-the-art approach has pros and cons of each method. In general, the main advantage of the traditional image processing methods such as level set approach compared to the deep learning method is from the ability to trace back the algorithms step by step so it is easier to develop and/or to revise compared to the deep learning method that is end-to-end learning [157]. Another advantage of the traditional image processing methods specifically in the 3D segmentation of macular holes is it does not need large number of 3D image datasets; however, in the deep learning method it can be mitigated with transfer learning approach. Meanwhile, one of the main advantage of the deep learning models is no more manual parameter tuning compare to the traditional image processing; in our 3D macular holes segmentation case, it will be no more method parameters and suggested default values (chapter section 2.5.6 (*p.* 22)) and curvature-based cutting surface procedure (chapter section 2.5.4 (*p.* 19)).

5.3 Future Work

The work in this thesis, particularly macular hole images has been developed for 3D images. With the rise of deep learning method, the 3D segmentation of macular holes can be done by combining both traditional image processing methods and the deep learning method. This hybrid approach will offer the advantages of both methodologies.

Further measurements of macular holes may also be possible by combining the macular hole and macular layers for medical diagnosis purposes. The hybrid approach will be more efficient for 3D macular holes and retinal layer segmentation across a variety of different 3D macular hole and macular layers images. Meanwhile the clinical measurements will be done by traditional image processing; in our case such as based on 3D centerline definition.

Bibliography

1. Nasrulloh, A. V., Willcocks, C. G., Jackson, P. T. G., Geenen, C., Habib, M. S., Steel, D. H. W. & Obara, B. Multi-scale segmentation and surface fitting for measuring 3D macular holes. *IEEE Transactions on Medical Imaging* **37,** 580–589 (2018) (*pp.* iii, 9, 62).

2. Willcocks, C. G., Jackson, P. T., Nelson, C. J., Nasrulloh, A. V. & Obara, B. Interactive GPU active contours for segmenting inhomogenous objects. *Journal of Real-Time Image Processing,* 1–14 (2017) (*pp.* iii, 35).

3. Chen, Y., Nasrulloh, A. V., Wilson, I., Geenen, C., Habib, M. S., Obara, B. & Steel, D. H. W. Macular hole morphology and measurement using an automated three dimensional image segmentation algorithm. *BMJ Open Ophthalmology.* In press (2020) (*pp.* iii, 60).

4. Steel, D. H. W. *Macular holes with emphasis on the internal limiting membrane of the retina and the vitreoretinal interface.* PhD thesis (Institute of Genetic Medicine, Newcastle University, Newcastle, May 2018) (*pp.* 2, 4).

5. Keeler, C. R. Evolution of the British ophthalmoscope. *Documenta Ophthalmologica* **94,** 139–150 (1997) (*p.* 2).

6. Keeler, C. R. The ophthalmoscope in the lifetime of Hermann von Helmholtz. *Archives of Ophthalmology* **120,** 194–201 (2002) (*p.* 2).

7. Abràmoff, M. D., Garvin, M. K. & Sonka, M. Retinal imaging and image analysis. *IEEE Reviews in Biomedical Engineering* **3,** 169–208 (2010) (*pp.* 3–4, 11).

8. Tomlins, P. H. & Wang, R. Theory, developments and applications of optical coherence tomography. *Journal of Physics D: Applied Physics* **38,** 2519 (2005) (*pp.* 3–4).

9. Drexler, W. & Fujimoto, J. G. *Optical coherence tomography: technology and applications* (Springer Science & Business Media, 2008) (*p.* 3).

10. Bailey, I. L., Hall, A. & Lueck, A. H. *Visual impairment: an overview* (American Foundation for the Blind, New York, USA, 1990) (*p.* 3).

11. Huang, D., Swanson, E. A., Lin, C. P., Schuman, J. S., Stinson, W. G., Chang, W., Hee, M. R., Flotte, T., Gregory, K., Puliafito, C. A., *et al.* Optical coherence tomography. *Science* **254,** 1178–1181 (1991) (*p.* 3).

12. Puliafito, C. A., Hee, M. R., Lin, C. P., Reichel, E., Schuman, J. S., Duker, J. S., Izatt, J. A., Swanson, E. A. & Fujimoto, J. G. Imaging of macular diseases with optical coherence tomography. *Ophthalmology* **102,** 217–229 (1995) (*p.* 3).

13. Popescu, D. P., Flueraru, C., Mao, Y., Chang, S., Disano, J., Sherif, S., Sowa, M. G., *et al.* Optical coherence tomography: fundamental principles, instrumental designs and biomedical applications. *Biophysical Reviews* **3,** 155 (2011) (*p.* 4).

14. Sakata, L. M., DeLeon-Ortega, J., Sakata, V. & Girkin, C. A. Optical coherence tomography of the retina and optic nerve–a review. *Clinical & Experimental Ophthalmology* **37,** 90–99 (2009) (*p.* 4).

15. Yi, K., Chen, T. C. & de Boer, J. F. Spectral domain optical coherence tomography. *Techniques in Ophthalmology* **4,** 170–174 (2006) (*p.* 4).

16. Kanagasingam, Y., Bhuiyan, A., Abràmoff, M. D., Smith, R. T., Goldschmidt, L. & Wong, T. Y. Progress on retinal image analysis for age related macular degeneration. *Progress in Retinal and Eye Research* **38,** 20–42 (2014) (*p.* 4).

17. Gattoussi, S., Buitendijk, G. H., Peto, T., Leung, I., Schmitz-Valckenberg, S., Oishi, A., Wolf, S., Deák, G., Delcourt, C., Klaver, C. C., *et al.* The European eye epidemiology spectral-domain optical coherence tomography classification of macular diseases for epidemiological studies. *Acta Ophthalmologica* (2018) (*p.* 4).

18. Yu, Y., Liang, X., Wang, Z., Wang, J. & Liu, W. Clinical and morphological comparisons of idiopathic macular holes between stage 3 and stage 4. *Graefe's Archive for Clinical and Experimental Ophthalmology,* 1–7 (2018) (*p.* 4).

19. Inoue, M., Watanabe, Y., Arakawa, A., Sato, S., Kobayashi, S. & Kadonosono, K. Spectral-domain optical coherence tomography images of inner/outer segment junctions and macular hole surgery outcomes. *Graefe's Archive for Clinical and Experimental Ophthalmology* **247,** 325 (2009) (*p.* 4).

20. Quellec, G., Lee, K., Dolejsi, M., Garvin, M. K., Abramoff, M. D. & Sonka, M. Three-dimensional analysis of retinal layer texture: identification of fluid-filled regions in SD-OCT of the macula. *IEEE Transactions on Medical Imaging* **29,** 1321–1330 (2010) (*p.* 4).

21. Walter, T., Klein, J.-C., Massin, P. & Erginay, A. A contribution of image processing to the diagnosis of diabetic retinopathy-detection of exudates in color fundus images of the human retina. *IEEE Transactions on Medical Imaging* **21,** 1236–1243 (2002) (*p.* 4).

22. Patton, N., Aslam, T. M., MacGillivray, T., Deary, I. J., Dhillon, B., Eikelboom, R. H., Yogesan, K. & Constable, I. J. Retinal image analysis: concepts, applications and potential. *Progress in Retinal and Eye Research* **25,** 99–127 (2006) (*p.* 5).

23. Marrugo, A. G. & Millan, M. S. *Retinal image analysis: preprocessing and feature extraction* in *Journal of Physics: Conference Series* **274** (2011), 012039 (*p.* 5).

24. Smiddy, W. E. & Flynn, H. W. Pathogenesis of macular holes and therapeutic implications. *American Journal of Ophthalmology* **137,** 525–537 (2004) (*pp.* 5, 10).

25. Steel, D. & Lotery, A. Idiopathic vitreomacular traction and macular hole: a comprehensive review of pathophysiology, diagnosis, and treatment. *Eye* **27,** S1–S21 (2013) (*pp.* 5, 10–11, 51).

26. Croll, L. J. & Croll, M. Hole in the macula. *American Journal of Ophthalmology* **33,** 248–253 (1950) (*p.* 5).

27. Keller, B., Cunefare, D., Grewal, D. S., Mahmoud, T. H., Izatt, J. A. & Farsiu, S. Length-adaptive graph search for automatic segmentation of pathological features in optical coherence tomography images. *Journal of Biomedical Optics* **21,** 076015–076015 (2016) (*pp.* 6, 11).

28. Shahlaee, A., Rahimy, E., Hsu, J., Gupta, O. P. & Ho, A. C. Preoperative and postoperative features of macular holes on en face imaging and optical coherence tomography angiography. *American Journal of Ophthalmology Case Reports* **5,** 20–25 (Apr. 2017) (*pp.* 6, 11).

29. Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH image to ImageJ: 25 years of image analysis. *Nature Methods* **9,** 671–675 (2012) (*pp.* 6, 11).

30. Xu, D., Yuan, A., Kaiser, P. K., Srivastava, S. K., Singh, R. P., Sears, J. E., Martin, D. F. & Ehlers, J. P. A novel segmentation algorithm for volumetric analysis of macular hole boundaries identified with optical coherence tomography. *Investigative Ophthalmology & Visual Science* **54,** 163–169 (2013) (*pp.* 6, 10, 12–13, 21, 23, 79).

31. Alkabes, M., Padilla, L., Salinas, C., Nucci, P., Vitale, L., Pichi, F., Burès-Jelstrup, A. & Mateo, C. Assessment of OCT measurements as prognostic factors in myopic macular hole surgery without foveoschisis. *Graefe's Archive for Clinical and Experimental Ophthalmology* **251,** 2521–2527 (2013) (*p.* 6).

32. Ullrich, S., Haritoglou, C., Gass, C. t., Schaumberger, M., Ulbig, M. & Kampik, A. Macular hole size as a prognostic factor in macular hole surgery. *British Journal of Ophthalmology* **86,** 390–393 (2002) (*p.* 6).

33. Ch'ng, S. W., Patton, N., Ahmed, M., Ivanova, T., Baumann, C., Charles, S. & Jalil, A. The Manchester large macular hole study: is it time to reclassify large macular holes? *American Journal of Ophthalmology,* 36–42 (2018) (*p.* 6).

34. Philippakis, E., Legrand, M., El Sanharawi, M., Erginay, A., Couturier, A. & Tadayoni, R. Measurement of full-thickness macular hole size using en face optical coherence tomography. *Eye* (2017) (*p.* 6).

35. Teng, Y., Yu, M., Wang, Y., Liu, X., You, Q. & Liu, W. OCT angiography quantifying choriocapillary circulation in idiopathic macular hole before and after surgery. *Graefe's Archive for Clinical and Experimental Ophthalmology* **255,** 893–902 (2017) (*pp.* 6, 12).

36. dell'Omo, R., Vogt, D., Schumann, R. G., De Turris, S., Virgili, G., Staurenghi, G., Cereda, M., Costagliola, C., Priglinger, S. G. & Bottoni, F. The relationship between blue-fundus autofluorescence and optical coherence tomography in eyes with lamellar macular holes. *Investigative Ophthalmology & Visual Science* **59,** 3079–3087 (2018) (*p.* 6).

37. dell'Omo, R., Virgili, G., Bottoni, F., Parolini, B., De Turris, S., Di Salvatore, A., dell'Omo, E. & Costagliola, C. Lamellar macular holes in the eyes with pathological myopia. *Graefe's Archive for Clinical and Experimental Ophthalmology,* 1–10 (2018) (*p.* 6).

38. Eklund, A., Dufort, P., Forsberg, D. & LaConte, S. M. Medical image processing on the GPU – past, present and future. *Medical Image Analysis* **17,** 1073–1094 (2013) (*pp.* 6, 37, 39).

39. Smistad, E., Falch, T. L., Bozorgi, M., Elster, A. C. & Lindseth, F. Medical image segmentation on GPUs – a comprehensive review. *Medical Image Analysis* **20,** 1–18 (2015) (*pp.* 6, 37, 39, 56).

40. Michalewska, Z., Michalewski, J. & Nawrocki, J. Continuous changes in macular morphology after macular hole closure visualized with spectral optical coherence tomography. *Graefe's Archive for Clinical and Experimental Ophthalmology* **248,** 1249–1255 (2010) (*p.* 6).

41. Forsaa, V. A., Lindtjørn, B., Kvaløy, J. T., Frøystein, T. & Krohn, J. Epidemiology and morphology of full-thickness macular holes. *Acta Ophthalmologica* **96,** 397–404 (2018) (*pp.* 6, 61, 78).

42. Duker, J. S., Kaiser, P. K., Binder, S., de Smet, M. D., Gaudric, A., Reichel, E., Sadda, S. R., Sebag, J., Spaide, R. F. & Stalmans, P. The international vitreomacular traction study group classification of vitreomacular adhesion, traction, and macular hole. *Ophthalmology* **120,** 2611–2619 (2013) (*pp.* 6, 61, 75).

43. Kusuhara, S., Escaño, M. F. T., Fujii, S., Nakanishi, Y., Tamura, Y., Nagai, A., Yamamoto, H., Tsukahara, Y. & Negi, A. Prediction of postoperative visual outcome based on hole configuration by optical coherence tomography in eyes with idiopathic macular holes. *American Journal of Ophthalmology* **138,** 709–716 (2004) (*pp.* 6, 61).

44. Ruiz-Moreno, J., Staicu, C., Pinero, D., Montero, J., Lugo, F. & Amat, P. Optical coherence tomography predictive factors for macular hole surgery outcome. *British Journal of Ophthalmology* **92,** 640–644 (2008) (*pp.* 6, 61–62).

45. Wakely, L., Rahman, R. & Stephenson, J. A comparison of several methods of macular hole measurement using optical coherence tomography, and their value in predicting anatomical and visual outcomes. *British Journal of Ophthalmology* **96,** 1003–1007 (2012) (*pp.* 6, 61).

46. Liu, P., Sun, Y., Dong, C., Song, D., Jiang, Y., Liang, J., Yin, H., Li, X. & Zhao, M. A new method to predict anatomical outcome after idiopathic macular hole surgery. *Graefe's Archive for Clinical and Experimental Ophthalmology* **254,** 683–688 (2016) (*pp.* 6, 61, 77).

47. Steel, D. H. W., Parkes, C., Papastavrou, V. T., Avery, P. J., El-Ghrably, I. A., Habib, M. S., Sandinha, M. T., Smith, J., Stannard, K. P., Vaideanu-Collins, D. & Hillier, R. J. Predicting macular hole closure with ocriplasmin based on spectral domain optical coherence tomography. *Eye* **30,** 740–745 (2016) (*pp.* 6, 10, 61).

48. Gonzalez, R. C., Woods, R. E. & Eddins, S. L. *Digital image processing using MATLAB* Second Edition (McGraw Hill Education, New Delhi, India, 2010) (*p.* 7).

49. Pham, D. L., Xu, C. & Prince, J. L. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering* **2,** 315–337 (2000) (*p.* 7).

50. Patil, D. D. & Deore, S. G. Medical image segmentation: a review. *International Journal of Computer Science and Mobile Computing* **2,** 22–27 (2013) (*p.* 7).

51. Kafieh, R., Rabbani, H. & Kermani, S. A review of algorithms for segmentation of optical coherence tomography from retina. *Journal of Medical Signals and Sensors* **3,** 45 (2013) (*p.* 7).

52. Osher, S. & Fedkiw, R. P. Level set methods: an overview and some recent results. *Journal of Computational Physics* **169,** 463–502 (2001) (*p.* 7).

53. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521,** 436–444 (May 2015) (*pp.* 7, 36, 38).

54. Kapoor, R., Whigham, B. T. & Al-Aswad, L. A. Artificial intelligence and optical coherence tomography imaging. *The Asia-Pacific Journal of Ophthalmology* **8,** 187–194 (2019) (*p.* 7).

55. Thapa, S. S., Thapa, R., Paudyal, I., Khanal, S., Aujla, J., Paudyal, G. & van Rens, G. Prevalence and pattern of vitreo-retinal diseases in Nepal: the Bhaktapur glaucoma study. *BMC ophthalmology* **13,** 9 (2013) (*p.* 10).

56. Madi, H. A., Masri, I. & Steel, D. H. Optimal management of idiopathic macular holes. *Clinical Ophthalmology* **10,** 97–116 (2016) (*pp.* 10, 61).

57. Wang, L., He, L., Mishra, A. & Li, C. Active contours driven by local Gaussian distribution fitting energy. *Signal Processing* **89,** 2435–2447 (2009) (*pp.* 10, 13–17, 36–37, 39–40, 42, 58).

58. Mul, M. D., Bont, A. D. & Berg, M. IT-supported skill-mix change and standardisation in integrated eyecare: lessons from two screening projects in The Netherlands. *International Journal of Integrated Care* **7,** 1–10 (2007) (*p.* 11).

59. Marrugo, A. G., Millán, M. S., Cristóbal, G., Gabarda, S., Sorel, M. & Sroubek, F. *Image analysis in modern ophthalmology: from acquisition to computer assisted diagnosis and telemedicine* in *SPIE Optics, Photonics, and Digital Technologies for Multimedia Applications II* **8436** (Brussels, Belgium, Jan. 2012), 84360C (*p.* 11).

60. Puliafito, C. A., Hee, M. R., Lin, C. P., Reichel, E., Schuman, J. S., Duker, J. S., Izatt, J. A., Swanson, E. A. & Fujimoto, J. G. Imaging of macular diseases with optical coherence tomography. *Ophthalmology* **102,** 217–229 (1995) (*p.* 11).

61. Hee, M. R., Puliafito, C. A., Wong, C., Duker, J. S., Reichel, E., Schuman, J. S., Swanson, E. A. & Fujimoto, J. G. Optical coherence tomography of macular holes. *Ophthalmology* **102,** 748–756 (1995) (*p.* 11).

62. Gaudric, A., Haouchine, B., Massin, P., Paques, M., Blain, P. & Erginay, A. Macular hole formation: new data provided by optical coherence tomography. *Archives of Ophthalmology* **117,** 744–751 (1999) (*p.* 11).

63. Ozturk, T., Karahan, E., Er, D., Kaya, M., Kocak, N. & Kaynak, S. Effect of macular hole volume on postoperative central macular thickness. *Arquivos Brasileiros de Oftalmologia* **79,** 137–142 (2016) (*pp.* 12, 62).

64. Liu, Y.-Y., Chen, M., Ishikawa, H., Wollstein, G., Schuman, J. S. & Rehg, J. M. Automated macular pathology diagnosis in retinal OCT images using multi-scale spatial pyramid and local binary patterns in texture and shape encoding. *Medical Image Analysis* **15,** 748–759 (2011) (*p.* 12).

65. Chan, T. F. & Vese, L. A. Active contours without edges. *IEEE Transactions on Image Processing* **10,** 266–277 (2001) (*pp.* 12, 38–43, 58).

66. Kass, M., Witkin, A. & Terzopoulos, D. Snakes: active contour models. *International Journal of Computer Vision* **1,** 321–331 (1988) (*pp.* 12, 36–37, 40, 47).

67. Ronfard, R. Region based strategies for active contour models. *International Journal of Computer Vision* **13,** 229–251 (1994) (*p.* 12).

68. Malladi, R., Sethian, J. A. & Vemuri, B. C. Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17,** 158–175 (1995) (*pp.* 12, 38).

69. Osher, S. & Sethian, J. A. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics* **79,** 12–49 (1988) (*pp.* 12, 17, 38, 42).

70. Mumford, D. & Shah, J. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics* **42,** 577–685 (1989) (*pp.* 12, 38).

71. Li, C., Kao, C.-Y., Gore, J. C. & Ding, Z. Minimization of region-scalable fitting energy for image segmentation. *IEEE Transactions on Image Processing* **17,** 1940–1949 (2008) (*pp.* 12, 16).

72. Li, C., Xu, C., Gui, C. & Fox, M. D. *Level set evolution without re-initialization: a new variational formulation* in *Conference on Computer Vision and Pattern Recognition* **1** (Washington DC, USA, 20-25 June 2005), 430–436 (*pp.* 16, 41–42).

73. Maggioni, M. & Foi, A. *Nonlocal transform-domain denoising of volumetric data with groupwise adaptive variance estimation* in *IS&T/SPIE Electronic Imaging, Computational Imaging X* **8296** (California, USA, Oct. 2012), 82960O (*pp.* 17–18).

74. Maggioni, M., Katkovnik, V., Egiazarian, K. & Foi, A. Nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE Transactions on Image Processing* **22,** 119–133 (2013) (*pp.* 17–18).

75. Ozcan, A., Bilenca, A., Desjardins, A. E., Bouma, B. E. & Tearney, G. J. Speckle reduction in optical coherence tomography images using digital filtering. *Journal of the Optical Society of America A* **24,** 1901–1910 (2007) (*p.* 17).

76. Law, M. W. & Chung, A. Efficient implementation for spherical flux computation and its application to vascular segmentation. *IEEE Transactions on Image Processing* **18,** 596–612 (2009) (*p.* 19).

77. Coeurjolly, D., Lachaud, J.-O. & Levallois, J. Implementation of integral based digital curvature estimators in DGtal. *Imagen-A* **3,** 27–31 (5 2013) (*p.* 19).

78. Coeurjolly, D., Lachaud, J.-O. & Levallois, J. *Integral based curvature estimators in digital geometry* in *International Conference on Discrete Geometry for Computer Imagery* (Seville, Spain, 20-22 March 2013), 215–227 (*p.* 19).

79. Pottmann, H., Wallner, J., Huang, Q.-X. & Yang, Y.-L. Integral invariants for robust geometry processing. *Computer Aided Geometric Design* **26,** 37–60 (2009) (*p.* 19).

80. Mishchenko, Y. A fast algorithm for computation of discrete Euclidean distance transform in three or more dimensions on vector processing architectures. *Signal, Image and Video Processing* **9,** 19–27 (2015) (*p.* 19).

81. Willcocks, C. G., Jackson, P. T. G., Nelson, C. J. & Obara, B. Extracting 3D parametric curves from 2D images of helical objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39,** 1757–1769 (2017) (*p.* 21).

82. Cleveland, W. S. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association* **74,** 829–836 (1979) (*p.* 21).

83. The Mathworks, Inc. *MATLAB* https://www.mathworks.com. Massachusetts, USA, 2018 (*pp.* 22, 34, 79).

84. Kohavi, R. *A study of cross-validation and bootstrap for accuracy estimation and model selection* in *International Joint Conference on Artificial Intelligence* (Montreal, Quebec, Canada, 20-25 August 1995), 1137–1145 (*p.* 22).

85. Ahirwar, A. Study of techniques used for medical image segmentation and computation of statistical test for region classification of brain MRI. *International Journal of Information Technology and Computer Science* **5,** 44–53 (2013) (*p.* 23).

86. Taha, A. A. & Hanbury, A. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging* **15,** 29 (2015) (*p.* 23).

87. Yuan, J., Schörr, C. & Steidl, G. Simultaneous higher-order optical flow estimation and decomposition. *SIAM Journal on Scientific Computing* **29,** 2283–2304 (2007) (*p.* 23).

88. Yuan, J., Bae, E. & Tai, X.-C. *A study on continuous max-flow and min-cut approaches* in *Computer Vision and Pattern Recognition* (San Francisco, USA, 13-18 June 2010), 2217–2224 (*p.* 23).

89. Yuan, J., Bae, E., Tai, X.-C. & Boykov, Y. *A continuous max-flow approach to Potts model* in *European Conference on Computer Vision* (Crete, Greece, May 2010), 379–392 (*p.* 23).

90. Bland, J. M. & Altman, D. G. Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet* **327,** 307–310 (1986) (*p.* 23).

91. Giavarina, D. Understanding Bland Altman analysis. *Biochemia Medica* **25,** 141–151 (2015) (*p.* 23).

92. McGraw, K. O. & Wong, S. P. Forming inferences about some intraclass correlation coefficients. *Psychological Methods* **1,** 30–46 (1996) (*p.* 30).

93. Geng, X.-Y., Wu, H.-Q., Jiang, J.-H., Jiang, K., Zhu, J., Xu, Y., Dong, J.-C. & Yan, Z.-Z. Area and volume ratios for prediction of visual outcome in idiopathic macular hole. *International Journal of Ophthalmology* **10,** 1255–1260 (2017) (*pp.* 34, 77, 79).

94. Jeong, W. K., Beyer, J., Hadwiger, M., Vazquez, A., Pfister, H. & Whitaker, R. T. Scalable and interactive segmentation and visualization of neural processes in EM datasets. *IEEE Transactions on Visualization and Computer Graphics* **15,** 1505–1514 (2009) (*pp.* 36, 39, 54).

95. Roberts, M., Packer, J., Sousa, M. C. & Mitchell, J. R. *A work-efficient GPU Algorithm for level set segmentation* in *Proceedings of the Conference on High Performance Graphics* (Eurographics Association, Saarbrucken, Germany, June 2010), 123–132 (*pp.* 36, 39, 47, 54).

96. Zhu, L., Karasev, P., Kolesov, I., Sandhu, R. & Tannenbaum, A. Interactive image segmentation from a feedback control perspective. *ArXiv e-prints.* arXiv: 1606.08008 `[cs.CV]` (June 2016) (*p.* 36).

97. Yushkevich, P. A., Piven, J., Hazlett, H. C., Smith, R. G., Ho, S., Gee, J. C. & Gerig, G. User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *Neuroimage* **31,** 1116–1128 (2006) (*pp.* 36, 40, 47).

98. Li, M., He, C. & Zhan, Y. Adaptive level-set evolution without initial contours for image segmentation. *Journal of Electronic Imaging* **20** (2011) (*pp.* 37, 40).

99. Cocosco, C. A., Kollokian, V., Kwan, R. K.-S., Pike, G. B. & Evans, A. C. BrainWeb: online interface to a 3D MRI simulated brain database. *NeuroImage* **5,** 425 (1997) (*p.* 38).

100. Rosset, A., Spadola, L. & Ratib, O. OsiriX: an open-source software for navigating in multidimensional DICOM images. *Journal of Digital Imaging* **17,** 205–216 (2004) (*pp.* 38, 55).

101. Jarrin, M., Young, L., Wu, W., Girkin, J. M. & Quinlan, R. A. *Chapter twenty-one - in vivo, ex vivo, and in vitro approaches to study intermediate filaments in the eye lens* in *Intermediate Filament Proteins* (eds Omary, M. B. & Liem, R. K.) **568** (Academic Press, 2016), 581–611 (*p.* 38).

102. Osher, S. & Fedkiw, R. *Level set methods and dynamic implicit surfaces* (Springer New York, 2002) (*p.* 38).

103. Pock, T., Cremers, D., Bischof, H. & Chambolle, A. *An algorithm for minimizing the Mumford-Shah functional* in *IEEE International Conference on Computer Vision* (Kyoto, Japan, 29 September-2 October 2009), 1133–1140 (*p.* 38).

104. Chambolle, A. & Pock, T. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision* **40,** 120–145 (2011) (*p.* 38).

105. Gorelick, L., Schmidt, F. R. & Boykov, Y. *Fast trust region for segmentation* in *2013 IEEE Conference on Computer Vision and Pattern Recognition* (Portland, OR, USA, 23-28 June 2013), 1714–1721 (*p.* 38).

106. Zhu, H., Meng, F., Cai, J. & Lu, S. Beyond pixels: a comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *Journal of Visual Communication and Image Representation* **34,** 12–27 (2016) (*pp.* 39, 58).

107. Grady, L. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28,** 1768–1783 (Nov. 2006) (*pp.* 39, 47).

108. Vineet, V. & Narayanan, P. *CUDA cuts: fast graph cuts on the GPU* in *IEEE Computer Society Conf. Computer Vision and Pattern Recognition* (Anchorage, AK, USA, 23-28 June 2008), 1–8 (*p.* 39).

109. Zhao, F. & Xie, X. An overview of interactive medical image segmentation. *Annals of The British Machine Vision Association (BMVA)* **2013,** 1–22 (Apr. 2013) (*pp.* 39, 58).

110. Pratx, G. & Xing, L. GPU computing in medical physics: a review. *Medical Physics* **38,** 2685 (2011) (*p.* 39).

111. Shi, L., Liu, W., Zhang, H., Xie, Y. & Wang, D. A survey of GPU-based medical image computing techniques. *Quantitative Imaging in Medicine and Surgery* **2,** 2223–2292 (2012) (*p.* 39).

112. Adalsteinsson, D. & Sethian, J. A. A fast level set method for propagating interfaces. *Journal of Computational Physics* **118,** 269–277 (1995) (*p.* 39).

113. Lefohn, A. E., Kniss, J. M., Hansen, C. D. & Whitaker, R. T. A streaming narrow-band algorithm: interactive computation and visualization of level sets. *IEEE Transactions on Visualization and Computer Graphics* **10,** 422–433 (2004) (*pp.* 39, 54).

114. Zhao, H.-K., Chan, T., Merriman, B. & Osher, S. A variational level set approach to multiphase motion. *Journal of Computational Physics* **127,** 179–195 (1996) (*p.* 39).

115. Xu, C. & Prince, J. L. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing* **7,** 359–369 (1998) (*p.* 40).

116. He, Z. & Kuester, F. *GPU-based active contour segmentation using gradient vector flow* in *International Conference on Advances in Visual Computing* (Lake Tahoe, NV, USA, Nov. 2006), 191–201 (*p.* 40).

117. Smistad, E., Elster, A. C. & Lindseth, F. Real-time gradient vector flow on GPUs using OpenCL. *Journal of Real-Time Image Processing* **10,** 67–74 (2012) (*p.* 40).

118. Schmid, J., Iglesias-Guitián, J. A., Gobbetti, E. & Magnenat-Thalmann, N. A GPU framework for parallel segmentation of volumetric images using discrete deformable models. *The Visual Computer* **27,** 85–95 (2010) (*p.* 40).

119. Chen, H. L. J., Samavati, F. F., Sousa, M. C. & Mitchell, J. R. *Sketch-based volumetric seeded region growing* in *Proceedings of the Third Eurographics Conference on Sketch-Based Interfaces and Modeling* (Eurographics Association, Vienna, Austria, Sept. 2006), 123–130 (*pp.* 40, 47).

120. Mory, B. *Interactive segmentation of 3D medical images with implicit surfaces* eng. PhD thesis (STI, Lausanne, 2011) (*pp.* 40, 47).

121. Caselles, V., Catté, F., Coll, T. & Dibos, F. A geometric model for active contours in image processing. *Numerische Mathematik* **66,** 1–31 (1993) (*p.* 40).

122. Zhu, S. C. & Yuille, A. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18,** 884–900 (1996) (*p.* 40).

123. Fulkerson, B. & Soatto, S. *Really quick shift: image segmentation on a GPU* in *Trends and Topics in Computer Vision* (Springer, 2010), 350–358 (*p.* 40).

124. Ren, C. Y. & Reid, I. *gSLIC: a real-time implementation of SLIC superpixel segmentation* technical report (University of Oxford, Department of Engineering, Technical Report, 2011) (*p.* 40).

125. Peng, D., Merriman, B., Osher, S., Zhao, H. & Kang, M. A PDE-based fast local level set method. *Journal of Computational Physics* **155,** 410–438 (1999) (*p.* 42).

126. Whitaker, R., Breen, D., Museth, K. & Soni, N. *Segmentation of biological volume datasets using a level-set framework* in *Volume Graphics 2001* (New York, USA, 21-22 June 2001), 249–263 (*p.* 47).

127. Olabarriaga, S. & Smeulders, A. Interaction in the segmentation of medical images: a survey. *Medical Image Analysis* **5,** 127–142 (2001) (*p.* 47).

128. Eyiyurekli, M. & Breen, D. Interactive free-form level-set surface-editing operators. *Computers & Graphics* **34,** 621–638 (2010) (*p.* 47).

129. Kay, T. L. & Kajiya, J. T. *Ray tracing complex scenes* in *Conference on Computer Graphics and Interactive Techniques* (ACM, New York, NY, USA, Aug. 1986), 269–278 (*p.* 49).

130. Evans, A. *Fast approximations for global illumination on dynamic scenes* in *ACM SIGGRAPH 2006 Courses* (ACM, Boston, Massachusetts, July 2006), 153–171 (*p.* 49).

131. Karotki, L., Huiskonen, J. T., Stefan, C. J., Ziółkowska, N. E., Roth, R., Surma, M. A., Krogan, N. J., Emr, S. D., Heuser, J., Grünewald, K. & Walther, T. C. Eisosome proteins assemble into a membrane scaffold. *The Journal of Cell Biology* **195,** 889–902 (2011) (*p.* 55).

132. Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J.-Y., White, D. J., Hartenstein, V., Eliceiri, K., Tomancak, P. & Cardona, A. Fiji: an open-source platform for biological-image analysis. *Nature Methods* **9,** 676–682 (2012) (*p.* 55).

133. Stagg, S. M., Gürkan, C., Fowler, D. M., LaPointe, P., Foss, T. R., Potter, C. S., Carragher, B. & Balch, W. E. Structure of the Sec13/31 COPII coat cage. *Nature* **439,** 234–238 (2006) (*p.* 55).

134. Chang, J. T., Schmid, M. F., Rixon, F. J. & Chiu, W. Electron cryotomography reveals the portal in the herpesvirus capsid. *Journal of Virology* **81,** 2065–2068 (2007) (*p.* 55).

135. Sen, A., Heymann, J. B., Cheng, N., Qiao, J., Mindich, L. & Steven, A. C. Initial location of the RNA-dependent RNA polymerase in the bacteriophage Phi6 procapsid determined by cryo-electron microscopy. *The Journal of Biological Chemistry* **283,** 12227–12231 (2008) (*p.* 55).

136. Cox, R., Pickar, A., Qiu, S., Tsao, J., Rodenburg, C., Dokland, T., Elson, A., He, B. & Luo, M. Structural studies on the authentic mumps virus nucleocapsid showing uncoiling by the phosphoprotein. *Proceedings of the National Academy of Sciences of the United States of America* **111,** 15208–15213 (2014) (*p.* 55).

137. Bartz, D. *Volvis datasets* http://www.volvis.org. Accessed: 2016-03-30. 2005. (2016) (*p.* 55).

138. Willcocks, C. G. *Implementation of interactive gpu active contours (IGAC)* GitHub. https://github.com/cwkx/IGAC (*p.* 59).

139. Meuer, S. M., Myers, C. E., Klein, B. E., Swift, M. K., Huang, Y., Gangaputra, S., Pak, J. W., Danis, R. P. & Klein, R. The epidemiology of vitreoretinal interface abnormalities as detected by spectral-domain optical coherence tomography: the beaver dam eye study. *Ophthalmology* **122,** 787–795 (2015) (*p.* 61).

140. Neffendorf, J. E., Simpson, A. R., Steel, D. H., Desai, R., McHugh, D. A., Pringle, E. & Jackson, T. L. Intravitreal gas for symptomatic vitreomacular adhesion: a synthesis of the literature. *Acta Ophthalmologica* **96,** 685–691 (2018) (*p.* 61).

141. Matsumiya, W., Kusuhara, S., Shimoyama, T., Honda, S., Tsukahara, Y. & Negi, A. Predictive value of preoperative optical coherence tomography for visual outcome following macular hole surgery: effects of imaging alignment. *Japanese Journal of Ophthalmology* **57,** 308–315 (2013) (*pp.* 61, 74).

142. Team, R. C. A language and environment for statistical computing. R foundation for statistical computing, Vienna, Austria. *URL:(https://www. R-project. org)* (2018) (*p.* 64).

143. Wickham, H. *ggplot2: elegant graphics for data analysis* (Springer, 2016) (*p.* 64).

144. Philippakis, E., Amouyal, F., Couturier, A., Boulanger-Scemama, E., Gaudric, A. & Tadayoni, R. Size and vitreomacular attachment of primary full-thickness macular holes. *British Journal of Ophthalmology* **101,** 951–954 (2017) (*pp.* 75, 78–79).

145. Liu, L., Enkh-Amgalan, I., Wang, N.-K., Chuang, L.-H., Chen, Y.-P., Hwang, Y.-s., Chang, C.-J., Chen, K.-J., Wu, W.-c., Chen, T.-L., *et al.* Results of macular hole surgery: evaluation based on the international Vitreomacular traction study classification. *Retina* **38,** 900–906 (2018) (*p.* 75).

146. Chui, T. Y., VanNasdale, D. A., Elsner, A. E. & Burns, S. A. The association between the foveal avascular zone and retinal thickness. *Investigative Ophthalmology & Visual Science* **55,** 6870–6877 (2014) (*p.* 77).

147. Tick, S., Rossant, F., Ghorbel, I., Gaudric, A., Sahel, J.-A., Chaumet-Riffaud, P. & Paques, M. Foveal shape and structure in a normal population. *Investigative Ophthalmology & Visual Science* **52,** 5105–5110 (2011) (*p.* 77).

148. Shin, J., Chu, Y., Hong, Y., Kwon, O. & Byeon, S. Determination of macular hole size in relation to individual variabilities of fovea morphology. *Eye* **29,** 1051 (2015) (*p.* 77).

149. Banerjee, P., Elgohary, M., Wickham, L. & Charteris, D. Measuring macula hole size on optical coherence tomography; inter and intra observer agreement. *Investigative Ophthalmology & Visual Science* **54,** 3612–3612 (2013) (*p.* 77).

150. Geenen, C., Murphy, D., Sandinha, M., Rees, J. & Steel, D. The significance of preoperative external limiting membrane height on visual prognosis in patients undergoing macular hole surgery. *Retina.* (In press) (2018) (*p.* 77).

151. Tornambe, P. E. Macular hole genesis: the hydration theory. *Retina* **23,** 421–424 (2003) (*p.* 78).

152. Woon, W. H., Greig, D., Savage, M. D., Wilson, M. C., Grant, C. A., Mokete, B. & Bishop, F. Movement of the inner retina complex during the development of primary full-thickness macular holes: implications for hypotheses of pathogenesis. *Graefe's Archive for Clinical and Experimental Ophthalmology* **253,** 2103–2109 (2015) (*p.* 78).

153. Madi, H. A., Dinah, C., Rees, J. & Steel, D. H. The case mix of patients presenting with full-thickness macular holes and progression before surgery: implications for optimum management. *Ophthalmologica* **233,** 216–221 (2015) (*p.* 78).

154. Salter, A. B., Folgar, F. A., Weissbrot, J. & Wald, K. J. Macular hole surgery prognostic success rates based on macular hole size. *Ophthalmic Surgery, Lasers and Imaging Retina* **43,** 184–189 (2012) (*p.* 78).

155. Gupta, B., Laidlaw, D., Williamson, T., Shah, S., Wong, R. & Wren, S. Predicting visual success in macular hole surgery. *British Journal of Ophthalmology* **93,** 1488–1491 (2009) (*p.* 78).

156. Nasrulloh, A. V. *3D multiscale segmentation and measurements of macular holes* GitHub. https://github.com/amarvijai/macularhole_segmentation3d (*p.* 79).

157. O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D. & Walsh, J. *Deep learning vs. traditional computer vision* in *Science and Information Conference* (2019), 128–144 (*p.* 83).