Μεταπτυχιακή Διατριβή
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΑΣ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ
ΠΡΟΓΡΑΜΜΑ ΙΑΤΡΙΚΗΣ ΦΥΣΙΚΗΣ

ΚΑΤΑΤΜΗΣΗ ΙΑΤΡΙΚΩΝ ΕΙΚΟΝΩΝ ΜΕ ΤΗ
ΜΕΘΟΔΟ ΣΥΝΟΛΩΝ ΕΠΙΠΕ∆ΟΥ
(LEVEL SETS)

Εξεταστική Επιτροπή
κ. Παναγιωτάκης Γεώργιος
κ. Κωσταρίδου Ελένη
κ. Οικονόμου Νικόλαος

Αμπατζής Δημήτρης
Μηχανικών Η/Υ και Πληροφορικής
MEDICAL IMAGE SEGMENTATION BY USE
OF THE LEVEL SET FRAMEWORK

Medical School 2009

Abatzis Dimitris
Computer Engineering & Informatics
And no-one knows the wheres or why
But something stirs and something tries
And starts to climb towards the light

“Echoes”
Pink Floyd
Contents

Chapter 1 ......................................................................................................................5
  Introduction ..............................................................................................................5
    1.1 Motivation........................................................................................................5
    1.2 Clinical significance.........................................................................................7
    1.3 Thesis Layout...................................................................................................8

Chapter 2 ....................................................................................................................10
  Methodological Background and Literature Review .........................................10
    2.1 Anatomy of carotids.......................................................................................10
    2.2 Overview ..........................................................................................................12
    2.3 Deformable models........................................................................................21
      2.3.1 Geometric Deformable Model...............................................................23
      2.3.2 Curve Evolution Theory.................................................................23
    2.4 Level Set Method...........................................................................................25
      2.4.1 Basic Derivation....................................................................................25
      2.4.2 Speed Functions....................................................................................31

Chapter 3 ....................................................................................................................34
  Materials and Methods ..........................................................................................34
    3.1 CT protocol and Programming Interface .......................................................35
    3.2 Segmentation methods...................................................................................38
      3.2.1 Level Set Methodology in ITK...............................................................38
      3.2.2 Preparing the Input................................................................................39
      3.2.3 Shape Detection Segmentation............................................................40
      3.2.4 Geodesic Active Contour Level lSet...............................................57

Chapter 4 ....................................................................................................................64
  Quantification of Results and Conclusions ..........................................................64
4.1 Evaluation Metrics .................................................................64
4.2 Results, conclusions and future work ....................................67

Appendix ..........................................................................................71
1. The Fast Marching Method (FMM), mathematic formulation analysis and its algorithmic implementation .................................................................71
2. Narrow-band level set method .......................................................76

References ..........................................................................................79
Acknowledgements ...............................................................................82
Summary (English) ................................................................................85
Summary (Greek) ...................................................................................86
Chapter 1

Introduction

1.1 Motivation

The so-called segmentation problem constitutes a significant fraction of the literature in the fields of image processing, computer vision and medical image analysis. Moreover, image registration which is of equal importance in the aforementioned fields, often heavily relies on segmentation since it is less error prone to segment objects in multiple images prior to registration, especially when the images to be registered, are derived from different modalities such as CT and MRI.
The automatic or semiautomatic\(^1\) segmentation of structures from 2D and 3D images, acquired from any medical modality, is an important first step in analyzing medical data. Since scanning methods became faster, more accurate and less artifacted, the shape recovery of anatomical structures has been increasing tremendously, because it contributes largely in medical treatment planning and therapy.

The segmentation or recovery of shapes of anatomical structures of the human body is of greater difficulty, compared to other fields, primarily due to the large variability in shapes, complexity of medical structures, several kinds of artifacts and restrictive body scanning methods.

The automatic and semiautomatic segmentation methods are therefore a very critical task in medical imaging. On the other hand, manual segmentation is not only a tedious and time consuming process, but also inaccurate, a fact that is verified by the variability of the results produced by segmentation experts [1]. In this way, it became an absolute must the development of algorithms that are accurate and require as little user interaction as possible.

In this direction targets the work carried out under the frame of this Msc thesis, which investigates the segmentation of the human carotids by deployment of two level set methods. In the next sections introductory information is provided, concerning the clinical significance of this task and the layout of this thesis.

---

1. The differentiation between automatic and semiautomatic segmentation methods is quite subtle. Primarily the distinction is related upon the degree that the user is involved in the procedure. For example the methods used in the present work are characterized as semiautomatic methods, since the user is required to place seed points to the structure under investigation, in order to initialize the procedure.
1.2 Clinical significance

Functionally, carotids are the brain’s main blood suppliers. In this sense, they are closely related to the brain’s pathologies, the severest of which is the stroke. Stroke is the third leading cause of death in the western world, after heart disease and cancer and causes 10% of deaths worldwide [2]. The carotid’s causality to stroke events is mainly due to atherosclerotic plaque, which is stacked onto the carotid’s walls and gradually blocks the vein (figure 1). In this way, even if the carotids are not totally clogged, the blood supply to the brain gradually changes decreasingly, thus often causing respective brain dysfunctions. Particularly, thrombus (blood clot, figure 1) is formed around the atherosclerotic plaque and when breaks off, at which
point it is called an embolus, can cause an embolic stroke as it is called. At last, a particle originated from the atherosclerotic plaque can cause an embolic stroke, that is simpler, a coarctation of a vessel.

In this way accurate, repeatable and efficient segmentation of carotids by extracting quantitative information can help in a prognostic, diagnostic and treatment-based way to the risk assessment of patients. In particular, this procedure is able to support clinical activities such as disease diagnosis and monitoring, intra-operative navigation, surgical and therapy planning. The latter include surgical procedures that remove atherosclerotic narrowing, known as carotid endarterectomy or carotid angioplasty. Thus, judging from its manifold contribution to clinical practice, it is conceivable that carotid segmentation is of tremendous significance in the treatment of stroke.

1.3 Thesis Layouts

The present thesis outlines the methods we have developed for segmenting both normal and pathological carotid images, acquired with the Computed Tomography (CT) protocol. The layout of the thesis is the following:

Chapter 2 analyses the methodological background of the current study. At first, section 2.1 provides an overview to the anatomy of carotids. Section 2.2 reviews the literature of segmentation methods based on level sets for medical images and at last reviews the level set methods developed for segmenting carotids. In addition, section 2.3 presents the conceptual model deployed in the current study, following with the analysis of the particular
class we used. Next, section 2.4 treats of the level set method, presenting its basic derivation and furthermore discriminating between the two algorithms used according to their speed function.

Chapter 3 refers to the materials and methods. It begins in section 3.1 with a description of the data provided for the experimental demonstration, and the programming interface by deployment of which the experimental procedure took place. Later on, in section 3.2 the implementation of the deployed methods in the programming interface used is presented with an analysis of their components. At last, all intermediate outputs and the final results of each method are illustrated.

Chapter 4 presents the evaluation of the results of each method by comparison with a corresponding manual segmentation result on the basis of appropriate metrics. At last, refers to the conclusions occurred and to future work that can be carried out based on the current Msc thesis.

In Appendix A some subsidiary methods, for the sake of a coherent flow are stated and analyzed independently.
Chapter 2

Methodological Background and Literature Review

2.1 Anatomy of carotids

Having already emphasized the importance of automatic and semiautomatic segmentation, it is imperative to further examine the special weight of such a procedure, in order to segment the human carotid arteries. In respect of human anatomy, the common carotid artery is an artery that supplies the head and neck with oxygenated blood; it divides in the neck to form the external and internal carotid arteries, as illustrated in figure 1 in the
previous chapter. The carotid artery is a paired structure, meaning that there are two in the body, one for each half. The left and right common carotid arteries follow the same course with the exception of their origin. The right common carotid artery originates in the neck while the left arises from the thoracic region.

Regarding its topology, at the lower part of the neck the common carotid artery is deeply seated, being covered by fibrous layers, muscles and glands. This part of the artery is crossed obliquely by a branch of the superior thyroid artery, middle and superior thyroid veins. As ascending its route, the common carotid artery is contained in a sheath, which is fibrous connective tissue that surrounds the internal carotid artery, the internal jugular vein and vagus nerve (figure 2.1). As seen in figure below, the carotid artery lies medial to the internal jugular vein, and the vagus nerve is situated posteriorly to these two vessels.

![Figure 2.1: The carotid sheath (www.wikipedia.org)](image)

From the above brief description of the carotid’s anatomy and due to the placement amongst the neighbor osculating structures, it is conclusive that the identification of the carotid arteries is a complex task. Additionally this artery presents a large variability in shape from person to person and at the
same person over time. On this basis, it is inferable that accurate segmentation is a challenging task for an automated or semi-automated system.

The next section presents an overview of medical image segmentation based on level set methods, which qualifies as the cornerstone of the current thesis. Finally, the overview converges to carotid’s segmentation methods based on the level set methodology, so as to provide adequate information for what has already been done so far in this field.

2.2 Overview

Below in figure 2.2 the taxonomy of the segmentation methods for 2D and 3D medical imagery is illustrated. This figure shows the classification of segmentation techniques for 2D and 3D images, divided into three core classes: 1) region-based, 2) boundary/surface-based and 3) fusion of boundary/region-based. The second core class of segmentation is also known as deformable models and is comprehensively analyzed right after the presentation of the overview.

The research in deformation methods started in the late 1980s, when the paper called “snakes” (the first class of deformable models) was published by Terzopoulos et al [3]. The second class of deformable models is level sets. These deformable models were first introduced by Osher and Sethian [4], which started from Sethian’s PhD dissertation [5].
Geometric deformable models or level-set techniques are broadly classified into two classes (see fig. 2.2 top, shown in dotted line area): 1) without regularizers and 2) with regularizers. Level sets without regularizers are techniques where the propagation force, that is the force that drives the navigation of the propagation front inwards or outwards, does not utilize the region-based strategy, like statistical means for its computation. These forces are constant and these methods are sometimes called level set stoppers or leakage prevention techniques, because they try to prevent any leaking out of the boundaries during propagation.

The first class is further classified into five categories, depending upon the design of the stopping force: 1) gradient-based stopping force, 2) edge-based
stopping force, 3) area-minimization stopping force, 4) curvature-based stopping force and 5) application-driven level sets. In each case the stopping force is designed so as to satisfy a criterion imposed by gradient calculation, edge identification, area minimization, curvature calculation or a combinational, application-specific, hybrid estimation of the above respectively. In the same way, as can be seen in figure 2.2, curvature-dependent stopping force is in turn subdivided into the below four subclasses: 1) plain curvature-based, 2) mean curvature flow-based with directionality, 3) bubbles and 4) morphing. So as to highlight their central concept, we can mention that plain curvature techniques are those which are driven solely by the curvature that is computed using differential geometry. Mean curvature flow with directionality xy-based techniques, is a set of techniques which use the combination of Euclidean curvature and direction together to achieve the deformation process. Bubbles are a set of seeds which grow, shrink and merge under the influence of image information such as edges and gradients to segment objects in images and volumes. At last, morphing techniques are those which undergo shape deformation from one initial shape to the target shape, by the combination of the signed distance and the gradient of the signed distance functions. This transformation catches the similarity between user-defined shape and target shape.

The second core class of level sets uses regularizers or level sets that derive the propagation force using statistical means, such as region-based strategy. This class is further classified into four types, depending upon the design of the propagation force, in accordance to the first class mentioned above. They are: 1) clustering-based, 2) classification based on Bayesian statistics, 3) shape-based and 4) constrained coupled level sets where the propagation force is derived from Bayesian strategies. According to the first approach, the speed function is estimated based on clustering the image
intensities of the region of interest and the background. In the second type, statistical tests are performed to yield an initial estimate of high-confidence subsets of the image regions. Furthermore, the velocities for the propagation of the region contours are defined in accordance with the a posteriori probability of the respective regions, leading to the Bayesian Level Set methodology. The shaped-based type integrates prior shape knowledge into the level set segmentation method. In particular, dissimilarity measures for shapes are encoded by the signed distance function, a function that stands as principal concept in the level set methodology and is analysed below in section 2.4. Last, the latter type of the regularized level set approaches optimally separates the values of certain image statistics over a known number of region types. Multiple sets of contours deform according to a coupled set of curve evolution equations, derived from a global cost functional. In this way the evolution of each curve depends at all times upon every pixel in the image and is directly coupled to the evolution of every other curve regardless their mutual proximity.

Having roughly defined the taxonomy of level sets in medical image segmentation, it is imperative at this point to draw attention to the level set methods that have been deployed till the present time, for the segmentation of human carotids. This investigation reveals that several papers have been published so far and are stated as follows in chronological order, starting from the most recently published.

To start with, the paper written by Scherl et al [7] entitled “Semi-automatic level-set based segmentation and stenosis quantification of the internal carotid artery in 3D CTA data sets“, presents a level-set method to segment and quantify stenosed internal carotid arteries in 3D contrast-enhanced computed tomography angiography (CTA). With this method different kind of plaque were almost completely excluded from the segmented regions. For an objective evaluation of the method’s performance
was studied with four different phantom data sets for which the ground truth of the degree of stenosis was known a priori and also was applied to 10 ICAs and compared the obtained segmentations with manual measurements of three physicians. According to this method, the definition of two points roughly in the middle of the vessel to be analyzed in order to solve the vessel/plaque segmentation problem is required. In particular the level set method used, conforms to the active contour model without edges of Chan and Vese [8], with the addition of more regularizing coefficients. Unfortunately, the introduction of the various regularization terms increases the dimension of the parameter space, thus making it more difficult to select a suitable parameter configuration. This model looks for a particular partition of the given volume into two regions, one representing the objects to be detected and the second one representing the background. The active contour is given as the boundary between the two regions and does not necessarily have to be defined by gradient. This is achieved by the minimization of an energy based functional using the intensity statistics (mean values) of the volume data set inside and outside the evolving implicit curve. At last, the algorithmic results obtained present an average error of 7% and a maximum error of 9.8% compared to the manual measurements of the observers.

Another related work in this domain carried out by T. Deschamps et al [9]. It presents a level set method for vessel segmentation and computational fluid dynamics simulations (CFD) of flows in complex structures. The method is demonstrated on a three-dimensional carotid artery MR dataset but since the goal of this work was the blood flow simulation, no quantitative results are presented for the level set segmentation technique. By deployment of the level set method presented in [19], which stands as one of the methods used as well in the current thesis, the purpose of this study is to transform the result of an accurate surface extraction method into
a CFD mesh with no loss of information. In this way, an accurate computation of measures such as fluid velocity, pressure and wall shear stress in realistic arterial geometries is performed.

In the same way contributed the work by Y. Jin and H. Ladak [10] named “Software for interactive segmentation of the carotid artery from 3D black blood magnetic resonance images”. The algorithm is based on a deformable model, and allows the user to initialize the model by combining and molding primitive shapes such as cylinders and spheres to form an initial approximate model on the organ surface. In particular, the algorithm used consists of three major steps: a) interactive placement of the initial balloon model inside the lumen, b) automatic inflation of the model toward the arterial wall and c) automatic localization of the arterial wall. The balloon model is represented by a closed mesh of triangles, with the initial mesh being an icosahedron. After the initial model placed inside the artery, it is rapidly inflated toward the arterial wall by means of a uniform force applied to the vertices of the mesh. When equilibrium under the influence of the inflation forces is reached, the model represents the appropriate shape of the artery. This mesh is then further deformed by means of image-based forces to localize the wall of the artery, as at the third step of the algorithm the inflation force is replaced with an image-based term. This term is a 3d potential function constructed from the image data which attracts the model to 3d intensity edges. The algorithm was applied to segment the carotid bifurcation from 3D black blood magnetic resonance of 5 subjects. The resulting surfaces produced were compared to surfaces segmented manually by an experienced user. The average distance between corresponding points on the manually and algorithmic-segmented surfaces was 0.37 mm whereas the average maximum distance was 2.03 mm.

Even older, equally important work is presented in the paper named “Level-Set Based Carotid Artery Segmentation for Stenosis Grading”,...
produced by C. M. van Bemmel et al [11]. In this work a semi-automated method is proposed for the determination of the degree of stenosis of the internal carotid artery (ICA) in 3D contrast enhanced MR angiograms. Hereto the central vessel axis is determined (CA), which subsequently is used as an initialization for a level set based segmentation of the stenosed carotid artery. The degree of stenosis is determined by calculating the average diameters of cross sectional planes along the CA. In this study four kinds of speed functions are embedded to the level set model, which are correspondingly tested: a) grey-level based speed function, b) gradient-level based speed function, c) vesselness-based speed function and d) combined speed function of the aforementioned functions. For twelve cases the degree of stenosis was determined and correlated with the scores of two experts (NASCET criterion). The Spearman’s correlation coefficient for the proposed method was 0.96 (p<0.001), versus 0.89 and 0.88 for the manual scores; a smaller bias and tighter confidence bounds for the automated method were found.

The utilization of the level set method was extended to process data obtained from imagery modalities which such as ultrasound images. The more characteristic sample was the paper entitled “An automated segmentation method for three-dimensional carotid ultrasound images”, proposed by Abir Zahalka and Aaron Fenster [12]. They developed an automated segmentation method for three-dimensional vascular ultrasound images. This method consists of two steps, an automated initial contour identification, followed by application of a Geometrical Deformable Model (GDM). This GDM works by imposing a contour energy minimization criterion to the speed function, thus standing as a regularized level set technique according to the distinction presented above. Specifically, the total force acting on the initialized contour is the weighted sum of internal and external forces. The internal force is computed as a function of the
curvature, which is defined as a second order derivative of the contour, while the external force is computed as a function of the gradient of each vertex of the contour and is calculated as a first order derivative of each vertex of the contour. In this way the user-specified parameters are the internal and external forces weightings and the desired model resolution, which is nothing more than the vertex spacing in pixels. Moreover a single seed point in the centre of any slice of the vessel is required to initiate the segmentation. This algorithm was tested on stenosed vessel phantoms with known geometry, and the segmentation of the cross-sectional areas was found to be within 3% of the true area. It was also applied to two sets of patient carotid images, one acquired with a mechanical scanner and the other with a freehand scanning tool, with results on the basis of mean and standard deviation concerning the difference in area ($mm^2$) between initial and final contours of each branch of the stenosed vessel as follows. For the 30% stenosed carotid artery the mean deviation is included in the range 0.23-0.36, the standard deviation is included in the range 0.20-0.31. Correspondingly, for the 60% stenosed carotid artery the ranges are 0.23-0.41 and 0.23-0.58 and at last for the 70% stenosed carotid artery the ranges are 0.20-0.38 and 0.21-0.27.

In the same route is directed the work performed by H. M Ladak et al. [13] named “Rapid Three-dimensional Segmentation of the Carotid Bifurcation from Serial MR Images”. In this study a virtual balloon model is deployed that combines two-dimensional segmentation and serial reconstruction, so as to produce the three-dimensional surface geometry. This three-dimensional technique is demonstrated in application to finite element meshing and computational fluid dynamics (CFD) modeling in the carotid bifurcation of a normal three-dimensional MR dataset. In this approach, surface reconstruction is particularly challenging due to the branching geometry. To handle such bifurcations the analyst must fit
individual spline surfaces to the outlines of the common carotid artery (CCA), the internal carotid artery (ICA) and the external carotid artery (ECA). Then the three separate surfaces must be joined together seamlessly, which is often difficult and labor intensive. Once the user places the balloon inside the lumen, it is automatically inflated toward the arterial wall, under the influence of an image-based force field and a simulated surface tension. The first one deforms the balloon toward the arterial wall while the second keeps the balloon smooth in the presence of image noise. As far as the results of this study are concerned, a time-averaged wall shear stress magnitude is graphically illustrated, for a deeper insight of which the reader is directed to [13], since it is out of the scope of the present overview.

Another paper also presenting “Accuracy and variability assessment of a semiautomatic technique for segmentation of the carotid arteries from three-dimensional ultrasound images” was published by J. D. Gill et al [14]. The technical specs of the method were described above in [10]. The comparison of this method with a fully manual segmentation method showed mean separation between the average segmented surface and the gold standard of 0.35 mm. The two surfaces were determined to agree with each other, within uncertainty, at 65% of the mesh points comprising the semiautomatic average surface.

In the next sections, the mathematical and conceptual background formulations regarding the level set methods deployed in the current study, are firstly introduced so as to establish and verify the model used. Right after, follows the analysis of the methodology, upon which the level set methods were utilized, assessed and their results visualized. Finally, the results and conclusions produced are presented and assessed comparatively for the methods addressed in this thesis, on the basis of a ground truth manual segmentation result, produced by an experienced user.
2.3 Deformable models

Deformable models are curves or surfaces defined within an image domain, which can move under the influence of internal forces, which are defined within the curve or surface itself and external forces, which are computed from the image data. The internal forces are designed to keep the model smooth during deformation. The external forces are defined to move the model toward an object boundary or other desired features within an image (fig. 2.3).

By constraining extracted boundaries to be smooth and incorporating other prior information about the object shape, deformable models offer robustness to both image noise and boundary gaps and allow integrating boundary elements into a coherent mathematical description. Such a boundary description can then be readily used by subsequent application.
Moreover, since deformable models are implemented on the continuum, the resulting boundary representation can achieve subpixel accuracy, a highly desirable property for medical image analysis applications.

The popularity of deformable models is due to the paper “Snakes: Active contour models” [3] written by Kass and Terzopoulos. Since its publication, deformable models have grown to be one of the most active and successful research areas in image segmentation. Various names such as snakes, active contours or surfaces, balloons and deformable curves or surfaces, have been used in the literature to refer to deformable models.

There are basically two types of deformable models: Parametric Deformable Models (PDM) and Geometric Deformable Models (GDM). Parametric Deformable Models represent curves and surfaces explicitly in their parametric forms during deformation. This representation allows direct interaction with the model and can lead to a compact representation for fast, real time implementation. Adaptation of the model topology, however, such as splitting or merging parts during the deformation, can be difficult using parametric models. Geometric deformable models, on the other hand can handle topological changes naturally. That is because their parameterizations are computed after complete deformation, thereby allowing topological adaptivity to be easily accommodated.

Despite this fundamental difference, the underlying principles of both methods are very similar. To argument this, Caselles et al. [15] formulated the relationship between parametric and geometric models and showed through an energy minimization formulation, that a geometric deformable contour is equivalent to a parametric deformable contour without the rigidity term.

The remainder of this chapter runs on the Geometric Deformable Contour Model, focusing right after to the level-set method which is the cornerstone of the current thesis and was deployed for the three-dimensional
segmentation of both normal and pathological carotid arteries. In this way, for further reading pertaining to the parametric deformable model, the reader is referenced to [16-19].

Last but not least, we have to clarify that the terms contour and surface can be used interchangeably wherever met into the text, under the condition that contour pertains to two-dimensional space while surface to three-dimensional respectively.

2.3.1 Geometric Deformable Model

Geometric deformable models proposed independently by Caselles et al. [20] and Malladi et al. [21], provide an elegant solution to address the primary limitation of parametric deformable models. These models are based on curve evolution theory [22-24] and the level set method [4, 26]. In particular, curves and surfaces are evolved using only geometric measures, resulting in an evolution that is independent of the parameterization. Since the evolution is independent of the parameterization, the evolving curves and surfaces can be represented implicitly, as a level set of a higher dimensional function. As a result, topology changes such as merging, splitting and/or developing holes can be handled implicitly.

In the next subsections, we first review the fundamental concepts in curve evolution derivation, which constitutes the fundamental equation of level sets, and afterwards the level set method for implementing curve evolution is presented.
2.3.2 Curve Evolution Theory

The purpose of curve evolution theory is to study the deformation of curves using only geometric measures such as unit normal and curvature. Let us consider a closed interface or front \( \Gamma(t) : [0, +\infty) \rightarrow \mathbb{R}^n \), propagating along its normal direction and denote its inward unit normal as \( N \) defined in Eq. 2.8 and its curvature as \( \kappa \) defined in Eq. 2.10, respectively. If \( \Gamma(t) = X(s,t) = [X(s,t), Y(s,t)] \), where \( s \) is any parameterization and \( t \) is the time, then the evolution of the curve along its normal direction, can be described by the following partial differential equation:

\[
\frac{\partial X}{\partial t} = V(s)N
\]  

where \( V(s) \) is called the speed function, since it determines the speed of the curve evolution. We note that a curve moving in some arbitrary direction, can always be reparameterized so as to have the same form as equation (2.1).

The most extensively studied deformations in curve evolution theory are the curvature deformation and the constant deformation. Curvature deformation is given by:

\[
\frac{\partial X}{\partial t} = \alpha \kappa N
\]

where \( \alpha \) is a positive constant. This equation will smooth a curve, eventually shrinking it to a circular point. The use of curvature deformation
has an effect similar to the use of an elastic internal force, which discourages bending and stretching.

Hereupon constant deformation is given by:

\[
\frac{\partial X}{\partial t} = V_o N \quad (2.3)
\]

where \( V_o \) is a coefficient determining the speed and direction of deformation. Constant deformation plays the role of a pressure force which pulls the contour towards the desired object boundaries. The properties of curvature deformation and constant deformation are complementary to each other. Curvature deformation removes singularities by smoothing the curve, while constant deformation creates singularities from an initial smooth curve.

The basic idea of the geometric deformable model is to couple the speed deformation (using constant and/or curvature deformation) with the image data, so that the evolution of the curve stops at object boundaries. The evolution is implemented using the level set method which is analyzed in the next section.

2.4 Level Set Method

2.4.1 Basic Derivation

The level set methodology is used to account for automatic topology adaptation, and it also provides the basis for a numerical scheme that is used
by geometric deformable models. The level set method is due to Osher and Sethian [4, 26 and 27].

According to formal description, given a closed \((N-1)\) hypersurface \(\Gamma(t=0)\), we now produce an Eulerian formulation for the motion of the hypersurface \(\Gamma(t)\) propagating along its normal direction with speed \(V\), where \(V\) can be a function of arguments such as the curvature and normal direction. The main idea of the level set methodology is to embed this propagating interface as the zero level set of a higher dimensional function. In other words, the moving interface is modelled to be the set of points that have the same function value, which is 0.

In this way, instead of tracking a curve through time, the level set method evolves a curve by updating the level set function at fixed coordinates through time. A useful property of this approach is that the level set function remains a valid function while the embedded curve can change topology.

The construction of the level set function \(\phi\) is as follows. Let \(\phi(x,t=0)\), where \(x\) is a point in \(\mathbb{R}^N\), be defined by:

\[
\phi(x,t=0) = \pm d
\]  

(2.4)

where \(d\) is the distance from \(x\) to \(\Gamma\), and the plus (minus) sign is chosen if the point \(x\) is situated outside (inside) the initial hypersurface \(\Gamma(t=0)\), as depicted in figure 2.4 below. Thus we have an initial function \(\phi(x,t=0) : \mathbb{R}^N \to \mathbb{R}\) with the property that:

\[
\Gamma(t=0) = \{ x | \phi(x,t=0) = 0 \}. 
\]  

(2.5)
The goal is to produce an equation for the evolving function $\phi(x,t)$ that contains the embedded motion of $\Gamma(t)$ as the level set $\phi(x,t) = 0, \forall t$. To do so, let $x(t)$ be the path of a point on the propagating front. That is $x(t = 0)$ is a point on the initial front $\Gamma(t = 0)$, and $(x_t) \cdot n = V(x(t))$ with the vector $(x_t)$ normal to the front at $x(t)$. The stipulation that the zero level set of the evolving function $\phi$ always matches the propagating hypersurface means that:

$$\phi(x(t), t) = 0, x \in R^N \quad (2.6)$$

Differentiating the above equation with respect to $t$ and using the chain rule, we obtain:

$$\frac{\partial \phi}{\partial t} + \nabla \phi \cdot \frac{\partial x}{\partial t} = 0 \quad (2.7)$$
where \( \nabla \phi \) denotes the gradient of \( \phi \). Assuming that \( \phi \) is negative inside the zero level set and positive outside, the inward unit normal to the level set curve is given by:

\[
N = -\frac{\nabla \phi}{|\nabla \phi|} \tag{2.8}
\]

Using this fact and equation (2.1), we can rewrite (2.7) as

\[
\frac{\partial \phi}{\partial t} = V(\kappa)|\nabla \phi| \tag{2.9}
\]

where the curvature \( \kappa \) at the zero level set is given by

\[
\kappa = \nabla \frac{\nabla \phi}{|\nabla \phi|} = \frac{\phi_{xx} \phi_y^2 - 2 \phi_x \phi_y \phi_{xy} + \phi_y \phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}} \tag{2.10}
\]

The relationship between Eq. (2.1) and Eq. (2.9) provides the basis for performing curve evolution, using the level set method. The figures (2.5) and (2.6) depict schematically what has been stated mathematically above.
Figure 2.5: An example of embedding a curve as a level set of function $\phi$. A single curve (left), the height map of the level set function with its zero level set depicted in red (right).
Some notable observations concerning the analysis that came before need to be considered:

- First, the evolving $\phi(x,y,t)$ always remains a function as long as $V$ is smooth. However, the level surface $\phi = 0$, and hence the propagating hypersurface $\Gamma(t)$, may change topology, break, merge and form sharp corners as the function $\phi$ evolves, as depicted in figure 2.7 below.
• Since the evolution equation (2.9) is derived from the level set only, the speed function $V(\kappa)$ in general is not defined on the other level sets. Hence we need a method to extend the speed function to all of the level sets. However, as one can note, the expressions for the unit normal and the curvature hold for all level sets. Many approaches for such extensions have been proposed but the level set function that evolves using these extended speed function can lose its property of being a signed distance function, causing inaccuracy in curvature and normal calculations. As a result, reinitialization of the level set function to a distance function is often required to these schemes. Recently, a method that does not suffer from this problem was proposed by Adalsteinsson and Sethian [29], which casts the speed extension problem as a boundary value problem and can be solved efficiently using the Fast Marching Method (FMM) (see Appendix A).

• The numerical methods used to solve the equation of evolution of the level set function, deploy a finite difference scheme that makes use of the Hamilton-Jacobi method and hyperbolic conservation laws.
2.4.2 Speed functions

In this section, we provide a brief overview of three examples of speed function used by geometric deformable contours. The analysis of these functions is of outmost importance. Its importance abuts not on the fact that aids further the insight into the level set method and discriminates between its different types, but also offers coherence as two of the speed functions have been deployed for the experimentation section, which will extensively presented in the next chapter.

The geometric deformable contour formulation, proposed by Caselles et al. [20] and Malladi et al. [21], takes the following form:

\[
\frac{\partial \phi}{\partial t} = c(\kappa + V_0)|\nabla \phi| \tag{2.11}
\]

where

\[
c = \frac{1}{1 + |\nabla(G_\sigma * I)|} \tag{2.12}
\]

where \(G_\sigma\) is a Gaussian kernel and I the image intensity, while the * symbol signifies the operation of convolution.

Positive initial speed function \(V_0\) shrinks the curve, and negative \(V_0\) expands the curve. The curve evolution is coupled with the image data through a multiplicative stopping term c. This scheme can work well for objects that have good contrast. However, when the object boundary is
indistinct or has gaps, the geometric deformable contour may leak because the multiplicative term only slows down the curve near the boundary, rather than completely stopping the curve. Once the curve passes the boundary, it will not be pulled back to recover the correct boundary.

To remedy the latter problem, Caselles et al. [20] and Kichenassamy et al. [30, 31] proposed an energy minimization formulation to design the speed function. This leads to the following geometric deformable contour formulation:

\[
\frac{\partial \phi}{\partial t} = c(\kappa + V_o) \nabla \phi + \nabla c \cdot \nabla \phi
\] (2.13)

It is worth noting that the resulting speed function has an additional stopping term \( \nabla c \cdot \nabla \phi \), which can pull back the contour/surface if it passes the boundary. Extensive results concerning these two forms of the level set method are depicted in the next chapter.

The latter formulation can still generate curves that pass through the boundary gaps. Siddiqi et al. [32] partially address this problem by altering the constant speed term through energy minimization, leading to the following geometric deformable contour:

\[
\frac{\partial \phi}{\partial t} = \lambda(c \kappa \nabla \phi + \nabla c \cdot \nabla \phi) + (c + \frac{1}{2} X \cdot \nabla c) \nabla \phi
\] (2.14)

In this case, the constant speed term \( V_0 \) in Eq. (2.13) is replaced by the second term, and the term \( \frac{1}{2} X \cdot \nabla c \) provides additional stopping power that can prevent the geometrical contour from leaking through small boundary gaps. The second term can be used alone as the speed function for shape
recovery as well. Although this model is robust to small gaps, large boundary gaps can still cause problems. Finally we have to emphasize that there is no geometric deformable contour model possessing the property of convergence to both perceptual boundaries (large boundary gaps) and boundary concavities, so some kind of compromise must be performed according to the application.

In the next chapter, the application of the first two speed functions analyzed above take place, in order to segment both a normal and a pathological dataset of carotid arteries. In this way, their performance and differences will be further investigated in a practical level.

Chapter 3

Materials and Methods
3.1 CT protocol and Programming Interface

Two CT volume datasets acquired on a 16 slice Multi-Detector CT were used for the experimentation section. The first dataset provided is regarded as normal by inspection of a neuroradiologist, since it no abnormalities due to atherosclerotic plaque appear inside the volume. To the contrary, the second dataset provided presents notable constriction of the carotid vessel and is characterized by inspection of the neuroradiologist as pathological. The resolution of the images is $512 \times 512$, and the volume datasets comprise both of 81 slices, with a voxel resolution of $0.488281 \times 0.488281$ and slice thickness $0.599998\text{mm}$. These data obtained with the aid of the neuroradiographical department of the university hospital at Rio Patras.

The programming interface on which the demonstration of the level set methods took place is the Insight Toolkit (ITK) [33]. The ITK is an open-source software toolkit for performing segmentation and registration. The processing data are found in a digitally sampled representation. Typically the sampled representation is an image acquired from such medical instrumentation as CT or MRI scanners. In short, and according to the philosophy of the ITK segmentation is the process of identifying and classifying data, while registration is the task of aligning or developing correspondence between data.

ITK is implemented in C++. It is cross-platform, using a build environment known as CMake to manage the compilation process in a platform independent way. Two worthy note characteristics of ITK are
generic programming and extreme programming. The first is an implementation style used to build ITK’s infrastructure, which uses templates so that the same code can be applied generically to any class or type that happens to support the operations used. As a result the code is highly efficient and many software problems are discovered during compile time, rather than run-time during program execution. Moreover, extreme programming is a software development model that collapses the usual software creation methodology into a simultaneous and iterative process of design – implement – test - release. Its key features are communication and testing. Communication amongst the members of the ITK community is what helps manage the rapid evolution of the software, while testing keeps the software stable.

The ITK does not provide any interface for visualization operations of the results. For this purpose, an extension of the Visualization Toolkit (VTK) [34] named vtkINRIA3D [35] was used. This interface is straight compatible and connectible to ITK and consists of a versatile library, providing numerous sophisticated functionalities with minimal development efforts, such as spatiotemporal visualization, management and synchronization of data.

This particular component that was deployed offers numerous important capabilities. Firstly, a synchronized view of the three 2d orientations (axial, sagittal and coronal) that comprise the volume is provided. This means that when clicking the image in the window of any one view, the other two are synchronized so as to point in the same voxel. The pointing system used is the intersection of two axes defining the axial plane of the image. Furthermore, easy navigation through slices of any view is performed with the scroll mouse button and the intensity and count of slice of the pointed voxel is projected. These features are essentially important so as navigate through the volume of interest in any orientation and thus choose the seed
points in order to initialize the procedure. In the next section ITK’s and vtkINRIA3D’s features are demonstrated in the analysis of the segmentation methods deployed in this study.

Moreover, except for this visualization module assembled with the core segmentation framework, an open source visualization toolkit named 3D Slicer version 3.0 [36] was deployed for the illustration of the final results, because of its embedded plug-in ability, which offers maximum resolution exploitation of the machine on which it runs. 3D Slicer is a free, open source software package for visualization and image analysis. This package is natively designed to be available on multiple platforms, including Windows, Linux and Mac Os X. Its features include sophisticated complex visualization capabilities, scene snapshots allow capture of all visualization parameters of a scene, extensive support for diffusion tensor imaging, advanced registration and data fusion capabilities, as well as comprehensive I/O capabilities.

Last but not least, a tool based on Insight Toolkit named ITK-SNAP 1.6.0 [37], was used for the manual segmentation of the datasets provided, so as to stand as the ground truth with which the two methods’s results were compared. This tool provides numerous functionalities such as image magnification and contrast enhancement and is considered ideal for such a task. The manual segmentation was performed in the axial orientation, slice by slice as illustrated in the picture below.
3.2 Segmentation methods

3.2.1 Level set methodology in ITK

In consistency with the mathematical model analysis that came before, we rewrite the general equation of motion of the level set function. Each methodology used, makes use of a generic level-set equation to compute the updated solution $\phi$ of the partial differential equation:

$$\frac{\partial \phi}{\partial t} = -a \cdot A(x) \cdot \nabla \phi - \beta \cdot P(x) \cdot |\nabla \phi| + \gamma \cdot Z(x) \cdot \kappa \cdot |\nabla \phi|$$  \hspace{1cm} (3.1)

where $A$ is the advection term, $P$ is the propagation term, and $Z$ is a spatial modifier for the mean curvature $\kappa$. The scalar constants $\alpha, \beta$ and $\gamma$ weight the relative influence of each of the terms on the movement of the interface.
In a level set segmentation method one or more terms may be omitted. In the methods demonstrated for the segmentation of the carotids, the first one uses the propagation and curvature terms while the second moreover uses the advection term, as described in the previous chapter.

Level sets can be used for image segmentation by using image-based features such as mean intensity, gradient and edges in the above equation. In a typical approach, a contour is initialized by the user and is then evolved until it fits the form of an anatomical structure in the image. In the next sections a detailed description of the methods is presented.

3.2.2 Preparing the input

Firstly, because of the unstructured format of the raw data provided, it was necessary to transform it to a suitable, supported format so as to feed them to the level set segmentation algorithms. By use of the corresponding module of the ITK, the raw slices were transformed to three-dimensional metadata (.mha) volume files, a flexible format, easily transformable to any other three-dimensional or two-dimensional file format.
3.2.3 Shape Detection Level-Set method

The first method is based on the paper by Malladi et al. [21], makes use of the standard propagation term and also the curvature-based term in the governing differential equation. This term acts as a smoothing force, where areas of high curvature, assumed to be due to noise, are smoothed out. Scaling parameters are used to control the tradeoff between the expansion term and the smoothing term.

The procedure is comprised of a cluster of filters, a pipeline as is called, connected one after the other. These filters operate on data fed from the
foregoing in the pipeline filter, and their output qualifies as input to the next in the pipeline filter. At the extremes, there are two special filters, a file reader and a file writer which import and export the data into and out of the pipeline respectively. Below, the pipeline used for the shape detection method is depicted and right afterwards an examination of the method is presented.

Figure 3.3: Collaboration diagram of the pipeline for the shape detection segmentation method.
As seen in the image 3.3 the Shape Detection Level Set Image Filter expects two inputs, in the current application two three-dimensional images. The first is an initial level set in the form of an image and the second is a feature image, which is nothing more than an image with a chosen feature such as edges, highlighted with special preprocessing. The first image defines an initial model \( \phi(x, t = 0) \), while the second defines the speed function. So as to generate the first one, the fast marching method is used. This filter implements a fast marching solution to a simple level set evolution problem, where the propagation takes place only outwards or only inwards (see Appendix A). Thus, since the procedure includes the selection of seed points (starting points) inside the structure of interest, an initial level set image is generated by forming an initial contour from the seed points and its outward evolution. The output of the Fast Marching Image Filter is a time-crossing map that indicates, for each pixel, how much time it would take for the front to arrive at the pixel location. Below in figures 3.4 and 3.5 we can see the effect of this filter on our volume datasets.

![Visualization Tool](image)
Figure 3.4: Output of the Fast Marching Filter for the normal dataset. 2-d navigation of, (a) axial, (b) coronal and (c) sagittal views screenshot.
Figure 3.5: Output of the Fast Marching Filter for the pathological dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views screenshot.

For the algorithm examined, the feature image is an edge potential image. As observed in figure 3.1 above, this image is computed as a function of the gradient magnitude. But since differentiation is an ill-defined method over digital data, it is convenient to define a scale in which the differentiation should be performed. It has been shown that a Gaussian kernel is the most suitable choice for performing such smoothing. By choosing a particular value for the standard deviation (σ) of the Gaussian, an associated scale is selected that ignores high frequency content, commonly considered image noise. The filter named Gradient Magnitude Recursive Gaussian Image Filter was used for this purpose, computing the magnitude of the image gradient at each pixel location. The computational process is equivalent to first smoothing the image by convolving it with a Gaussian kernel and then applying a differential operator. The user selects the value of σ. The results for a standard deviation of 0.5 used in the current cases are depicted in figure 3.6 and 3.7 for the normal and pathological datasets respectively.
Figure 3.6: Output of the Gradient Magnitude Filter for the normal dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views screenshot.
Next and right after the calculation of the gradient magnitude of the image, so as to shape a nice speed image, we have to apply a mapping in such a way that the propagation speed of the front, will be very low close to high image gradients, while it will move rather fast in low gradient areas. This arrangement will make the contour propagate until it reaches the edges.
of anatomical structures in the image and then slow down in front of those edges. So as to achieve this goal, a sigmoid function is applied over the gradient magnitude image, implemented by the Sigmoid Image Filter in ITK. The minimum and maximum values desired in the output, are defined with two corresponding methods and in the current study were chosen to be 0.0 and 1.0 respectively so as to get a nice, linear speed in the image. A crucial point in the correct application of this particular filter is the selection of the coefficients $\alpha$ and $\beta$. The former defines the width of the intensity window, while the latter defines the center of the intensity window as depicted in the figure below:

Figure 3.8: Effects of the various parameters in the SigmoidImageFilter. The alpha coefficient defines the width of the intensity window (left). The beta parameter defines the center of the intensity window.

Below in figures 3.9 and 3.10 the output of the Sigmoid Image Filter for the normal and pathological datasets is respectively displayed, which was fed with the output of the Gradient Magnitude Image filter according to the analysis that came before. In this way, the speed images of the two datasets were defined.
Figure 3.9: Output of the Sigmoid Image Filter for the normal dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views screenshot.
Figure 3.10: Output of the Sigmoid Image Filter for the pathological dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views screenshot.
Before the aforementioned operation, the input image has been filtered with a smoothing, edge preserving image filter so as to reduce noise and simultaneously keep the useful image information (edges) as intact as possible. This smoothing filter can be tuned by the user, who defines the number of iterations performed by the smoothing mask, the time step used in the computation of the level set evolution and the value of conductance. Typical values for the time step are 0.125 in 2D images and 0.0625 in 3D images that is the case in the current application. The number of iterations can be usually around 5, more iterations will result in further smoothing and will increase linearly the computing time.

These two images are fed to the Shape Detection Level Set Image Filter which outputs the final level set in the form of an image. From an initial, distributed in image space, volume that is comprised of the seed points and all neighboring pixels in distance 3 pixels away in the current study, the evolution procedure has as follows. For every point on the interface of the volume, a calculation of its velocity is performed according to the forces acting on it. For the present filter and methodology, two forces are considered and configured by the user as parameters before execution, provoking the corresponding velocities:

- The propagation velocity which is proportional to the feature image. The constant of proportionality can be set to be positive, in which case positive values of the feature image cause this velocity to point outwards or negative, in which case positive values of the feature image cause this velocity to point inwards. In a homogenous region of the feature image the propagation velocity is constant, causing the snake to expand (or contract) at a unit speed.

- The curvature velocity is used to control the shape of the evolving snake, and it can sometimes prevent the snake from leaking into adjacent structures. The curvature velocity acts inwards and is
approximately proportional to the curvature of the snake at the point. Sharp corners in the snake's boundary have high curvature, while points where the snake is straight have low curvature. The effect of curvature velocity is to slow down the snake evolution at places of high curvature, effectively smoothing out the sharp corners that may otherwise be formed.

As has been pinpointed earlier, the Fast Marching Image Filter output which is also fed as input to the Shape Detection Level Set Image Filter, provides a time crossing map for every point in the image, that is how much time it will take for the interface to get there. In this way, the distance of neighboring to the interface points is reinitialized before the evolution of the level set function. At last, the time the evolution takes place, or in other words the algorithm converges to a solution, is defined either by the maximum number of iterations performed, that is the number the differential equation is updated or the mean squared error between two successive estimations of the differential equation. The output of the ShapeDetectionLevelSetImageFilter is depicted below in figures 3.12- 3.15 for the left and right carotids in each of the normal and pathological datasets.
Figure 3.12: Output of the Shape Detection Level Set Image Filter for the normal dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views of the right carotid artery screenshot.

Figure 3.13: Output of the Shape Detection Level Set Image Filter for the normal dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views of the left carotid artery screenshot.

Figure 3.14: Output of the Shape Detection Level Set Image Filter for the pathological dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views of the right carotid artery screenshot.
Figure 3.15: Output of the Shape Detection Level Set Image Filter for the pathological dataset. 2-d navigation of (a) axial, (b) coronal and (c) sagittal views of the left carotid artery screenshot.

Then the application of a threshold in the output image is equivalent to taking a snapshot of the contour at a particular time during its evolution. Practically speaking, this time range is chosen to be that, in which the contour was contained for a long time in a region of the image space. The Binary Threshold Image Filter which is applied to serve this purpose, thresholds the resulting level set produced from the Shape Detection Level Set Image Filter in order to get a binary image representing the segmented object. The upper threshold of the Binary Threshold Image Filter is set to 0.0 in order to display the zero set of the resulting level set. The lower threshold is set to a large negative number, so as to ensure that the interior of the segmented object will appear inside the binary region. Below the figure schematically describes the response of the Binary Threshold Image Filter:
As seen in the figure above the user defines two thresholds, upper and lower, and two intensity values, inside and outside. For each pixel in the input image, the value of the pixel is compared with the lower and upper threshold. If the pixel value is inside the range \([\text{Lower}, \text{Upper}]\) the output pixel is assigned the inside value. Otherwise the output pixels are assigned to the outside value. Thresholding, as commonly used, is applied as the last operation of the segmentation pipeline, the result of which for the shape detection method is shown below in figures, as well as some intermediate stages of evolution.
Figure 3.12: Intermediate stages of surface evolution for the segmentation of the left carotid artery by use of the Shape Detection Level Set method.

Figure 3.12: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Shape Detection Level Set Method for the segmentation of the left carotid artery for normal dataset.
Figure 3.13: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Shape Detection Level Set Method for the segmentation of the right carotid artery for normal dataset.

Figure 3.14: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Shape Detection Level Set Method for the segmentation of the left carotid artery for pathological dataset.
3.2.4 Geodesic Active Contour Level-Set method

The filter that performs this kind of segmentation is based on the paper by Caselles et al [20]. This filter named Geodesic Active Contour Level Set Image Filter extends the functionality of the Shape Detection Level Set Image Filter, by the addition of a third advection term. Specifically this filter makes use of all three velocities defined by the level set model:

- The propagation velocity analyzed in the previous section
- The curvature velocity also analyzed in the previous section
- The advection velocity which attracts the level set to the object boundaries. In particular, it causes the evolving surface to slow down or stop as it approaches edges in the grayscale image and in some cases pulls the evolving interface back when leaking out of the structure that is to be segmented. In quantitative terms, the advection
velocity is defined by the dot product of the unit vector perpendicular to the evolving surface and the gradient vector of the feature image. That means that when the evolving surface is parallel to an image edges and close to it, the advection force acting inwards on the evolving surface gets maximized.

The Geodesic Active Contour Level Set Image Filter expects two inputs, analogously to the Shape Detection Level Set Image Filter (figure 3.4). The first is an initial level set, in the form of an image. The second input is a feature image. For this algorithm, the feature image is an edge potential image that basically follows the same processing steps described in the previous section, so as to be produced.

As can be noticed in figure 3.4, the pipeline components involved in the application of the GeodesicActiveContourLevelSetImageFilter are exactly the same with those used for the application of the shape detection method pictured in figure 3.1. In this way, the analysis of the pipeline components will be quite restricted, since a discussion of those components has already been done in the previous section.
The pipeline involves a first stage of smoothing using the Curvature Anisotropic Diffusion Image Filter, so as to reduce noise as much as possible, without however neither lose valuable information about edges of the structures, nor burden the algorithmic time complexity. Next, as we have already described in detail in the previous section, the gradient magnitude of the smoothed image is computed, in order to produce the edge potential image. At last, a speed mapping is performed for every pixel by use of a sigmoid function, so as to finally produce the speed image. The choice of the parameters alpha and beta are crucial for the achievement of the desired result but there is not a standard procedure to accomplish this. Instead it is more of a trial and error process and an empirical method is often used.
On the other branch of the pipeline, an identical cluster of filters is used compared to the previous level set method. A set of user provided seeds are passed to the FastMarchingImageFilter, which outputs a time crossing map in the form of an image. A constant value is subtracted from this map in order to obtain a level set, in which the zero set represents the initial contour. This level set is also passed as input to the Geodesic Active Contour Level Set Image Filter which serves as an initial reference model. Since the filter “knows” how the position of the contour when propagating outwards with constant speed 1.0, is able to estimate its position when moving under the influence of the sum of propagation, curvature and advection velocities.

Finally, the final level set generated by the core filter of the pipeline is passed to the BinaryThresholdImageFilter in order to produce a binary mask representing the segmented object. For the Geodesic Active Contour Image Filter, scaling parameters are used to trade off between the propagation (inflation), the curvature (smoothing) and the advection term, by invocation of the corresponding methods. In the current study, the advection scale is set to one and the other two parameters are set as command-line arguments. The intermediate outputs of the pipeline are the same to the ones presented in the previous section. Thus, only the final output of the pipeline is depicted below in figures, as well as intermediate results during surface evolution by use of the Geodesic Active Contour Level Set Image Filter.
Figure 3.17: Intermediate stages of surface evolution for the segmentation of the left carotid artery by use of the Geodesic Active Contour Level Set method.

Figure 3.18: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Geodesic Active Contour Level Set method for the segmentation of the left carotid artery for normal dataset.
Figure 3.19: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Geodesic Active Contour Level Set method for the segmentation of the right carotid artery for normal dataset.

Figure 3.20: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Geodesic Active Contour Level Set method for the segmentation of the left carotid artery for pathological dataset.
Figure 3.21: From left to right 3D, 2D axial, sagittal and coronal views screenshot of the Geodesic Active Contour Level Set method for the segmentation of the right carotid artery for pathological dataset.
Chapter 4

Quantification of Results and Conclusions

4.1 Evaluation metrics

The metrics used to assess the results produced by the methods analyzed in the previous chapter, are presented and explained in this section. These
metrics were utilized in the programming environment of Insight Toolkit by use of the corresponding filters, since have been normalized before for this task, as can been noticed in the figure 4.1 below. In this way the segmentation results obtained, were compared with an expert’s manual segmentation results which served as a gold standard, as described in the previous chapter, on the basis of the following metrics:

1. **Kappa Statistic Metric**

The Kappa Statistic Metric computes the spatial intersection of two binary images. The metric here is designed for matching pixels in two images with the same exact value. Given two images A and B, the $\kappa$ coefficient is computed as

$$\kappa = \frac{|A \cap I|}{|A| + |B|}$$  \hspace{1cm} (4.2)
where $|A|$ is the number of foreground pixels in image A. This computes the fraction of area in the two images that is common to both the images. In the computation of the metric only foreground pixels are considered.

2. Match Cardinality Metric
The Match Cardinality Metric computes cardinality of the set of pixels that match exactly between two images A, B. In other words, it computes the number of pixels matches and mismatches between the two images.

3. Hausdorff Distance Metric
The Hausdorff Distance Metric computes the distance between the set non-zero pixels of two images according to the following formula:

$$H(A,B) = \max(h(A,B), h(B,A)), \quad (4.3)$$

where

$$h(A,B) = \max \min_{a \in A, b \in B} \|a - b\|$$

is the directed Hausdorff distance and A, B are respectively the set of non-zero pixels in the first and second input images. The Hausdorff distance measures the degree of mismatch between two sets and behaves like a metric over the set of all closed bounded sets with properties of identity, symmetry and triangle inequality.

As long as the application of the aforementioned metrics is concerned, we can say that operate on two input 3-d images, the first being the manual segmentation result and the second each one method’s corresponding result. These two are compared according to the aforementioned metrics in a registration process. That is, the fit between the segmented structures is
assessed by placing the images one into the other. Ideally speaking, the result of this registration process would be:

- Kappa Statistic Metric = 1 (normalized to one)
- Match Cardinality Metric = 1 (normalized to one)
- Hausdorff Distance Metric = 0

4.2 Results, conclusions and future work

It is imperative to emphasize that apart from the errors that come in the automatic segmentation procedure, the manual segmentation of the pipe of the carotid vessel presents many challenges and is beyond any doubt prone to significant errors. The metrics presented in the previous section in conjunction with the manually segmented results, provide a stable and overall qualitative as long as quantitative basis for the assessment of the two methods’s segmentation results. The evaluation of the results produced by the two level set methods for each carotid artery, in each dataset, is presented comparatively in the tables below.

<table>
<thead>
<tr>
<th>Normal Left Carotid</th>
<th>Shape Detection Level Set Method</th>
<th>Geodesic Active Contour Level Set Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kappa Statistic Metric</td>
<td>0.910916</td>
<td>0.912157</td>
</tr>
<tr>
<td>Match cardinality Metric</td>
<td>0.999923</td>
<td>0.999924</td>
</tr>
<tr>
<td>Hausdorff Distance Metric</td>
<td>4,12311 voxels</td>
<td>4,24264 voxels</td>
</tr>
</tbody>
</table>

Table 4.1: Quantification of results produced by the two level set methods, for the segmentation of the normal left carotid.
Table 4.2: Quantification of results produced by the two level set methods, for the segmentation of the normal right carotid.

<table>
<thead>
<tr>
<th></th>
<th>Shape Detection Level Set Method</th>
<th>Geodesic Active Contour Level Set Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kappa Statistic Metric</td>
<td>0.92384</td>
<td>0.925063</td>
</tr>
<tr>
<td>Match cardinality Metric</td>
<td>0.999928</td>
<td>0.999929</td>
</tr>
<tr>
<td>Hausdorff Distance Metric</td>
<td>4,47214 voxels</td>
<td>5,38516 voxels</td>
</tr>
</tbody>
</table>

Table 4.3: Quantification of results produced by the two level set methods, for the segmentation of the pathological left carotid.

<table>
<thead>
<tr>
<th></th>
<th>Shape Detection Level Set Method</th>
<th>Geodesic Active Contour Level Set Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kappa Statistic Metric</td>
<td>0.873556</td>
<td>0.874704</td>
</tr>
<tr>
<td>Match cardinality Metric</td>
<td>0.999904</td>
<td>0.999905</td>
</tr>
<tr>
<td>Hausdorff Distance Metric</td>
<td>7,07107 voxels</td>
<td>6,08276 voxels</td>
</tr>
</tbody>
</table>

Table 4.4: Quantification of results produced by the two level set methods, for the segmentation of the pathological right carotid.

<table>
<thead>
<tr>
<th></th>
<th>Shape Detection Level Set Method</th>
<th>Geodesic Active Contour Level Set Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kappa Statistic Metric</td>
<td>0.885937</td>
<td>0.885054</td>
</tr>
<tr>
<td>Match cardinality Metric</td>
<td>0.999899</td>
<td>0.999897</td>
</tr>
<tr>
<td>Hausdorff Distance Metric</td>
<td>3 voxels</td>
<td>2,82843 voxels</td>
</tr>
</tbody>
</table>

A first conclusion that occurs by the inspection of the above tables is that the two algorithms yield fairly acceptable results. This assumption is strengthened by the factors and conditions under which the experimentation took place. To begin with, the range of results in total is included in a high level for all four metrics used, considering the fact that inter and intra variation of segmentation results amongst experts varies significantly. It is imperative to note that the experimentation process was established in a base according to which results are obtained in a short time. That means that the mean squared error, used as criterion for the algorithmic convergence of the methods, can be set lower to 0.2, thus producing even less arithmetic error to
the final results, but no significant qualitative conclusions, at the expense of lower velocity and stability.

In the same way, the distribution of the seed points was denser at volumetric areas where the structure of interest is rather small and tubular and thus more artifacted because of the inevitable image blurring. In such areas the contrast between edges (carotid walls) and the interior of the structure is degraded and thus more initialization points are needed in order the interface to reveal the whole structure. In its turn edge blurring was balanced between adequate contrast and the prevention of leakage out of the structure. The decisions made during experimentation are always balanced with the need for acceptable performance in terms of computing resources. Therefore by use of 20-25 seed points the same initial configuration was provided for the two methods, a procedure though which is inherently variational in the level sets methodology. In this way the above results were produced in a relatively small amount of time, that is 2.5 min for the shape detection level set method and 3.0 min for the geodesic active contour level set method in a personal computer (2x1.73 GHz CPU, 2 GB Ram).

Under the frame described and by considering the tables above, it becomes perceptible that between the level set methods, the Geodesic Active Contour Level Set method produces slightly better results than the shape detection in three out of four cases, in the latter of which they yield almost the same results. By separate inspection of the Kappa Statistic Metric it is inferable that a large amount of voxels is common in the automated and manual segmentation results. Specifically ranging from 0.8735 (worst case) to 0.925 (best case), the kappa statistic metric justifies the good potential of the method, since proves that the automated results are from 87.35% to 92.5% identical with the corresponding ground-truth results produced manually. Another important remark straightly arising is that the common volume fraction between the automated and manual segmentation decreases more in
the pathological cases, where the uncertainty involved in the manual segmentation procedure is bigger as illustrated in the picture below.

Figure 4.2: Manual delineation of a slice of the left carotid artery constricted from atherosclerotic plaque.

The Match Cardinality Metric further validates the above assumptions, since it assesses the cardinality as described before, of the common volume fraction calculated by the Kappa Statistic Metric. Moreover the Hausdorff Distance Metric estimates the highest distance difference in voxels, between the corresponding segmented volumes. It is very interesting that in three out of four cases, as observed by tables 4.1 to 4.4, that its values don’t follow expected values, in accordance to the previous metrics, although are close enough. This fact reveals random errors introduced into the manual segmentation procedure, which are absent by the use of automated methods which preserve a stable behavior in the segmentation of the volume in total as opposed to the corresponding human manual segmentation.

Then it becomes evident that the results produced need to be cross-verified with other validation procedures such as atlas-based procedures and
averaged manual segmentation results, so as to obtain a wider view of the methods’s potential, both comparatively and separately. As well, the programming infrastructure deployed under the frame of the current thesis is easily extendable, so as to embed more automated segmentation methods. Moreover the open source nature of the software, makes it flexible enough to be intergraded into a friendly user interface, since the crucial parameters that must be used in such a graphic interface have been discovered.

Appendix

In the next sections, we itemize some additional methods used in the level set framework, in order to solve subsidiary algorithmic problems within the general level set methodology. In this way, it is inferable in order to achieve a complete overall analysis to state these methods’s concept in a separate section.
1. The Fast Marching Method (FMM), mathematic formulation analysis and algorithmic implementation.

An interesting method related to the level set method is the fast marching method, which was introduced by Sethian [38, 39]. The fast marching method solves a subclass of the problems normally solved with the level set method, but it does so much more quickly.

Like the level set method, the FMM also uses an implicit representation for an evolving interface, but for the fast marching method, the entire evolution of the interface is encoded in the embedding motion, not just a single time slice. In other words, the location of the interface in a 2-d image space at time $t$ is given by the set:

$$\Gamma(t) = \{(x, y) : \phi(x, y) = t\} \quad (A1)$$

As a result, in the fast marching method, the embedding function $\phi$ has no time dependency.

The embedding function $\phi$, is constructed by solving a static Hamilton-Jacobi equation of the form

$$F\|\nabla \phi\| = 1 \quad (A2)$$
where $F$ is the speed of the interface. What makes the FMM fast is the fact that equation (A2) can be solved with one pass over the mesh. This contrasts with the level set method, where each time step requires an additional pass over the mesh to evolve the level set function in time.

So as to gain the insight of the method, it is inferable to say that in essence, the FMM produces a time crossing map, which for every grid point $(x, y)$, determines a function $T(x, y)$ which gives the time at which the front crosses the point $(x, y)$. As an example, suppose the initial disturbance is a circle propagating outwards as depicted in figure 1 below. The original region (the blue on the left below) propagates outwards, crossing over each of the timing spots. The aforementioned function $T(x, y)$ (for two dimensional space) gives a cone-based surface, which is shown on the right. This surface has the property of intersecting the xy-plane exactly where the curve is initially. Better yet, at any height $T$ the surface gives the set of points reached at time $t$. 
The key to solving equation A2 in one pass is to traverse the mesh in the proper order. The grid points must be evaluated in the order of increasing $t$. This is accomplished by using a sorted heap which always keeps track of which grid point is to be evaluated next. To begin, the set of grid points is divided into three disjoint subsets, the far away points A, the alive points B and the active points C as depicted in figure 2 below. The far away points in A are the points $x_{i,j}$ for which the computed value is already determined. The active points in B are the points $x_{i,j}$ for which a tentative value of $\phi_{i,j}$ is computed. The remainder of the points is in the set C. One by one points in
B are taken, in order of increasing value of $\phi_{i,j}$ from the set B into A. Each time, points $\phi_{i,j}$ in C which become adjacent to points in the set A are moved into the set B and a tentative value for $\phi_{i,j}$ is computed by use of equation A2. The algorithm successfully terminates when all points have migrated into the set A. See fig.2 for an illustration of the sets A, B and C.

The full algorithm for the fast marching method becomes:

1. Initialize all the points adjacent to the initial interface with an initial value, put those points in A. All points $x_{i,j} \notin A$, adjacent to a point in A, are given initial estimates for $\phi_{i,j}$ by solving equation 2. These points are active points and put in the set B. All remaining points unaccounted for are placed in C an given initial value of $\phi_{i,j} = +\infty$. 

![Figure 2: Algorithmic analysis of the FMM](image-url)
2. Choose the point \( x_{i,j} \in B \) which has the smallest value of \( \phi_{i,j} \) and move it into A.

3. Any point which is adjacent to \( x_{i,j} \) (i.e. the points \( x_{i-1,j}, x_{i,j-1}, x_{i,j+1} \) and \( x_{i+1,j} \)) which is in B has each value recalculated using equation 2.

Any point adjacent to \( x_{i,j} \) and in C has its value computed using equation 2 and is moved into the set B.

4. If \( B \neq 0 \), go to step 2.

The Fast Marching method presented above sweeps through a grid of \( N \) total points in \( N \log N \) steps to obtain the evolving time position of the front, as it propagates through the grid.

2. Narrow-band level set method

There is another technique frequently used in the level set methods that deserves attention. While it is not an essential part of the level set method, it is useful in that it can significantly reduce the computational cost.

As noted earlier, switching from a parametric representation to an implicit representation used in the level set method also increased the computational
cost. For example, if an evolving curve in the plane can be modeled with $O(N)$ points, then the corresponding level set representation would require $O(N^2)$ points, due to the higher dimension of the level set function. However, most of that increased computational cost is spent computing the evolution of $\phi$ in regions far from the $\phi = 0$ interface of interest.

It was observed in [40] that it is not necessary to compute the evolution of $\phi$ everywhere, but only in the neighborhood of $\phi = 0$ interface. This observation effectively reduces the computation back to $O(N)$. This technique is called a narrow-band level set method and was significantly refined in [41]. Basically, the evolution equation of $\phi$ is computed on a dynamically determined set of grid points where $\phi$ is small, as demonstrated below.

![Figure 3: Narrow-band level set method](image)
The narrow band method in respect to figure 3 has as follows:

1. Extract the latest position of the curve for which it is \( \phi(x, y, t) = 0 \).

2. Define a band within a certain distance \( d \) from the curve. Its boundaries are located at position \( \phi(s) = -d \) (inward band) and \( \phi(s) = +d \) (outward band).

3. Update the level set function, so as to evolve the contour.

4. Check new position with respect to the limits of the band.

5. Update the position of the band regularly, and re-initialize the implicit function, so as to remain a valid distance function.

It is imperative to note that not all applications will benefit from a narrow band implementation; it depends heavily on the cost of computing the speed function \( F \), which can easily overwhelm the cost of the rest of the level set method.
References


Imagery: A State-of-the-Art Review”, *IEEE Transactions on Information technology in Biomedicine*, vol. 6, no 1, Mach 2002


[33] www.itk.org
[34] www.vtk.org
Acknowledgements

First of all I’d like to thank Mrs Costaridou Helen for the confidence showed me by assigning to me this thesis. In the same way, my sincere thanks go to Mr Pratikakis Ioannis engineer at the National Research foundation Demokritos, who served as my coordinator and co-supervisor, contributing considerably with his technical advice and persistent guidance. Last but not least, I want to express my deepest grateful feeling for Mr Panagiotakis George, president of department of Radiology in Medical School of Patras for all he has done for me, regarding the elaboration of this work in a psychologically supporting way.
THESIS SUMMARY

The present thesis outlines the methods we have developed for segmenting both normal and pathological carotid images, acquired with the Computed Tomography (CT) protocol. The layout of the thesis is the following:

Chapter 2 analyses the methodological background of the current study. At first, section 2.1 provides an overview to the anatomy of carotids. Section 2.2 reviews the literature of segmentation methods based on level sets for medical images and at last reviews the level set methods developed for segmenting carotids. In addition, section 2.3 presents the conceptual model deployed in the current study, following with the analysis of the particular class we used. Next, section 2.4 treats of the level set method, presenting its basic derivation and furthermore discriminating between the two algorithms used according to their speed function.

Chapter 3 refers to the materials and methods. It begins in section 3.1 with a description of the data provided for the experimental demonstration, and the programming interface by deployment of which the experimental procedure took place. Later on, in section 3.2 the implementation of the deployed methods in the programming interface used is presented with an analysis of their components. At last, all intermediate outputs and the final results of each method are illustrated.

Chapter 4 presents the evaluation of the results of each method by comparison with a corresponding manual segmentation result on the basis of appropriate metrics. At last, refers to the conclusions occurred and to future work that can be carried out based on the current Msc thesis.

In Appendix A some subsidiary methods, for the sake of a coherent flow are stated and analyzed independently.
ΠΕΡΙΛΗΨΗ

Στα πλαίσια της παρούσης εργασίας πραγματοποιήθηκε μελέτη της μεθόδου Συνόλων Επιπέδου για την κατάτμηση καροτίδων από τρισδιάστατες εικόνες.

Ειδικότερα πραγματοποιήθηκε μελέτη των παθολογιών που συνδέονται με αυτές προκειμένου να καταστούν εμφανή τα κίνητρα της παρούσης εργασίας, όσον αφορά στη συμβολή της στην κλινική σημασία και ιατρική πρακτική. Κατ’αυτόν τον τρόπο, αφού παρουσιάστηκε η ανατομία των καροτίδων και οι δυσκολίες που ενέχει το εγχείρημα της κατάτμησης τους καθώς και μια ανασκόπηση των μεθόδων Συνόλων Επιπέδου (Level-Sets) για κατάτμηση ιατρικής εικόνας και δή καροτίδων, παρουσιάστηκε το γενικό μοντέλο και ο μαθηματικός φορμαλισμός της μεθόδου που χρησιμοποιήθηκε.

Εν συνεχεία παρουσιάστηκαν τα τρισδιάστατα δεδομένα και η διαχείρισή τους, οι προγραμματιστικές διαπαρές και υποδομές με τις οποίες υλοποιήθηκαν δύο παραλλαγές της μεθόδου. Επίσης παρουσιάζονται τα αποτελέσματα της μεθόδου οπτικοποιημένα και τέλος συγκρίνονται με αντίστοιχα αποτελέσματα ενός ειδικού ακτινολόγου στη βάση κάποιων κατάλληλων μετρικών. Τέλος παρουσιάζονται τα συμπεράσματα που προέκυψαν καθώς και κάποιες ιδέες για μελλοντική δουλειά που μπορεί να γίνει στη βάση αυτής που έγινε στα πλαίσια της εν λόγω μετατυπωχικής διατριβής.