Multi-resolution Active Models for Image Segmentation

by

Ahmed Gawish

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Image segmentation refers to the process of subdividing an image into a set of non-overlapping regions. Image segmentation is a critical and essential step to almost all higher level image processing and pattern recognition approaches, where a good segmentation relieves higher level applications from considering irrelevant and noise data in the image. Image segmentation is also considered as the most challenging image processing step due to several reasons including spatial discontinuity of the region of interest and the absence of a universally accepted criteria for image segmentation.

Among the huge number of segmentation approaches, active contour models or simply snakes receive a great attention in the literature. Where the contour/boundary of the region of interest is defined as the set of pixels at which the active contour reaches its equilibrium state. In general, two forces control the movement of the snake inside the image, internal force that prevents the snake from stretching and bending and external force that pulls the snake towards the desired object boundaries. One main limitation of active contour models is their sensitivity to image noise. Specifically, noise sensitivity leads the active contour to fail to properly converge, getting caught on spurious image features, preventing the iterative solver from taking large steps towards the final contour. Additionally, active contour initialization forms another type of limitation. Where, especially in noisy images, the active contour needs to be initialized relatively close to the object of interest, otherwise the active contour will be pulled by other non-real/spurious image features.

This dissertation, aiming to improve the active model-based segmentation, introduces two models for building up the external force of the active contour. The first model builds up a scale-based-weighted gradient map from all resolutions of the undecimated wavelet transform, with preference given to coarse gradients over fine gradients. The undecimated wavelet transform, due to its near shift-invariance and the absence of down-sampling properties, produces well-localized gradient maps at all resolutions of the transform. Hence, the proposed final weighted gradient map is able to better drive the snake towards its final equilibrium state. Unlike other multiscale active contour algorithms that define a snake at each level of the hierarchy, our model defines a single snake with the external force field is simultaneously built based on gradient maps from all scales.

The second model proposes the incorporation of the directional information, revealed by the dual tree complex wavelet transform (DT $\mathbb{C}$WT), into the external force field of the active contour. At each resolution of the transform, a steerable set of convolution kernels is created and used for external force generation. In the proposed model, the size and the orientation of the kernels depend on the scale of the DT $\mathbb{C}$WT and the local orientation

statistics of each pixel. Experimental results using nature, synthetic and Optical Coherent Tomography (OCT) images reflect the superiority of the proposed models over the classical and the state-of-the-art models.

Acknowledgements

All praise and gratitude to God almighty who gave me the strength and the knowledge to bring forth this thesis.

I would like to thank my supervisor, Prof. Paul Fieguth, for his enormously beneficial guidance, advice, comments, support and discussions over more than four years. I would like also to thank my doctoral committee members, prof. Kostadinka Bizheva, prof. Hamid Tizhoosh, prof. Alexander Wong and prof. Mahmoud El-Sakka, for their insightful comments and suggestions.

Finally, I wish to express my extreme gratitude and love to my parents, my beloved wife, my children, my brothers and my sisters for their understanding, encouragement and continuous support.

Dedication

To my beloved parents, my dear wife, and my children.

To the soul of my brother and teacher Osama Azmy.

To ALL whom I love.

Table of Contents

List of Figures ix

1 Introduction 1

2 Background 4

2.1 Image Segmentation 4

2.1.1 Similarity Based Segmentation 6

2.1.2 Discontinuity Based Segmentation 10

2.2 Edge Detection 11

2.2.1 First Derivative Edge Detectors 13

2.2.2 Second Derivative Edge Detectors 16

2.2.3 Edge Linking 17

2.3 Multiresolution Image Processing 20

2.3.1 Wavelet Transform 21

2.4 Overcomplete Wavelet Transforms 25

2.5 Snakes: Active Contour Models 27

2.5.1 Parametric Active contours 28

2.5.2 Non Parametric Active contours 31

3 Multi-resolution Undecimated Active Contours **34**
3.1 Noise Sensitivity of Active Contour Models 34
3.2 Hierarchical Active Contours 35
3.3 Undecimated Active Contours 37
3.4 Multi-resolution Gradient Maps 40
3.5 Segmentation Results and Discussion 45

4 Multi-resolution Directional Active Contours **52**
4.1 The Dual Tree Complex Wavelet Transform 53
4.2 Multi-resolution and Directionality 54
4.3 Multi-resolution Directional Vector Field Convolution 56
4.4 Local Statistics and Dominant Orientation Calculation 61
4.5 Segmentation Results 64

5 Optical Coherent Tomography (OCT) Corneal Image Segmentation **73**
5.1 Human Cornea 74
5.2 Optical Coherent Tomography 76
5.3 OCT image segmentation 78
5.3.1 OCT Signal Enhancement 79
5.3.2 Active Contour Initialization 81
5.3.3 Epithelium Layer Thickness 84
5.3.4 Motion Correction 87
5.4 Segmentation Results 90

6 Conclusions **94**

References **96**

List of Figures

2.1 Segmentation by thresholding 7
2.2 Segmentation by split and merging 9
2.3 Over-segmentation phenomenon of watershed method 9
2.4 A point detection mask 10
2.5 Line detection masks for Horizontal, 45°, Vertical and −45° directions . . . 11
2.6 Line detection using 45° mask 12
2.7 Edge models 13
2.8 Examples of partial derivative approximations 14
2.9 Gradient based edge detection 14
2.10 Edge detection using Marr-Hildrith approach 17
2.11 The Hough transform 19
2.12 Building image pyramid using approximation and interpolation filters . . . 20
2.13 Standard and scaled-translated Haar functions 23
2.14 Standard and scaled-translated Haar wavelets 24
2.15 Wavelet decomposition and reconstruction 26
2.16 The undecimated wavelet transforms 27
2.17 A parametric vs. geometric active contour methods 32

3.1 A parametric vs. geometric active contour at different levels of noise 36
3.2 Near shift invariant property of non-decimated wavelet transform 39
3.3 A schematic diagram of the proposed model 40

3.4 An average ASD convergence for different values of α 41
3.5 Traditional external force fields of noisy images (PSNR=20) 43
3.6 UWAC external force fields of noisy images (PSNR=20) at 1, 3, 5 and 8 layer of the UDWT 44
3.7 Images used for testing 45
3.8 Proposed model using 1, 3, 5, and 8 resolutions vs. classical GVF and VFC at different noise levels 46
3.9 Average performance of the proposed model using 1, 3, 5, and 8 resolutions vs. classical GVF and VFC at different noise levels 47
3.10 A comparison of the average performance of the proposed UWAC and other methods 48
3.11 Comparisons between converged contours of the UWAC and other methods 49
3.12 Segmentation accuracy versus number of wavelet scales 50

4.1 DT $\mathbb{C}$WT real and imaginary parts of the impulse responses 53
4.2 Direction selectivity property of DT $\mathbb{C}$WT. 55
4.3 Combining all wavelet data of the $\mathbb{C}$WT 57
4.4 The classical VFC kernel versus the proposed kernels $K_{l,\theta}$ 60
4.5 External force fields of VFC and MDVFC using a toy example 61
4.6 Synthetic example of dominant direction assignment 63
4.7 A comparison of converged contours for MVFC, MTVF and MDVFC. . . . 65
4.8 Enlarged external force fields of MDVFC and MTVF at different resolutions 66
4.9 Comparisons of the final segmentation results of MDVFC and UWAC modesl. 67
4.10 Segmentation results of the MDVFC under different noise models. 68
4.11 A comparison of average converged contours for MVFC, MTVF and MDVFC. 68
4.12 Iterative solver steps towards the final segmentation 70
4.13 Segmentation accuracy versus number of wavelet scales 71

5.1 Human cornea anatomy [121]. 75

5.2 Normal versus keratoconus cornea. 76
5.3 Schematic diagram of the swept-source setup 77
5.4 A typical OCT corneal image and segmentation challenges 78
5.5 Active contour leakage due to weak signal at cornea peripherals 80
5.6 The Generalized Hough Transform (GHT) search space 81
5.7 Example of active contour initialization using the GHT 82
5.8 Active contour initialization using GHT for different types of KC corneas . 85
5.9 Epithelium layer segmentation 86
5.10 Spatial locations of the reference frames on one OCT slice 89
5.11 Elevation map before and after applying the motion correction algorithm . 89
5.12 Example of final segmentation comparison between MDVFC, IS, MVFC and MTVF 91
5.13 Examples of anterior, posterior and epithelium layers segmentation for different KC eyes 93

Chapter 1

Introduction

Image segmentation refers to the process of subdividing an image into a set of non-overlapping regions. Image segmentation is a critical and essential step to almost all higher level image processing and pattern recognition approaches, where a good segmentation relieves higher level applications from considering irrelevant and noise data in the image. Image segmentation is also considered as one of the most challenging image processing steps due to several reasons including the spatial discontinuity of the region of interest and the absence of a universally accepted criteria for image segmentation.

Among the huge number of segmentation approaches, active contour models or simply snakes receive a great attention in the literature [1] [2]. Where the contour/boundary of the region of interest is defined as the set of pixels at which the active contour reaches its equilibrium state. In general, two forces control the movement of the snake inside the image, internal force that prevents the snake from stretching and bending and external force that pulls the snake towards the desired object boundaries.

One main limitation of active contour models is their sensitivity to image noise. Specifically, noise sensitivity leads the active contour to fail to properly converge, getting caught on spurious image features, preventing the iterative solver from taking large steps towards the final contour. Additionally, active contour initialization forms another type of limitation. Where, especially in noisy images, the model needs to be initialized relatively close to the object of interest, otherwise it will be pulled by other non-real/spurious image features.

recently, the multi-resolution wavelet representation and analysis of images and curves have been the subject of significant research attention [3] [4] [5]. Certainly wavelets are widely used for compression and denoising, and the wavelet transform maxima indicate the locations of image edges, however for orthogonal wavelets the decimation at coarser scales

means that the transform is not invariant to translations. This translation sensitivity leads to undesirable spatial variations in how an image gradient would be transformed into the wavelet domain, similar to the ringing artifacts around image singularities or edges when an image is reconstructed after modifying its wavelet coefficients [6].

In this thesis, aiming to improve accuracy of the active contour convergence, two models for building up the external force field of the active contour are proposed. The first model (Chapter 3), builds on the gradient vector flow GVF [7] and the vector field convolution (VFC) [8], and uses the undecimated wavelet transform [9] to form a resolution-based weighted gradient map for external force generation. The undecimated wavelet transform, due to its near shift-invariance and the absence of down-sampling properties, produces well-localized gradient maps at all resolutions of the transform. So, salient features of an image tend to survive at coarse scales of the transform where noise contribution becomes minimum. Also, fine structures and details of the image are localized and maintained at fine scales of the transform, however, noise contribution becomes more significant. Hence, in order to better drive the active contour towards its final equilibrium state, the first model builds the external force map of the snake using a resolution-based weighted gradient map with preference given to coarse gradients over fine gradients. Unlike other multiscale active contour algorithms [10] [11] that define a snake at each level of the hierarchy with the previous/coarser scale being the initialization of the snake in the current/finer scale, the model proposed in Chapter 3 defines a single snake with the external force field is simultaneously built based on the gradient maps from all scales.

The second model (Chapter 4) incorporates the directional information, revealed by the dual tree complex wavelet transform (DT $\mathbb{C}$WT) [12], into the external force field of the active contour. At each resolution of the transform, a steerable set of convolution kernels is created and used for external force generation of the active contour. The size of the kernels depends on the scale of the DT $\mathbb{C}$WT, with bigger kernels used at coarse scales and smaller kernels at fine scales. Moreover, the orientation of the kernel to be used for generating the force vector at each pixel is determined based on the local orientation statistics of the pixel and its neighbors. With big oriented kernels at coarse scales, the model is capable of taking larger steps towards the dominant/strong features of the object of interest, and is less sensitive to noise. At fine scales, the small oriented kernel allows the model to capture the fine details of the underlying object.

Finally, because Optical Coherent Tomography (OCT) technology is based on visible light, media opacity highly affects the quality of the images. Moreover, the light reflections during the imaging process make speckle noise an inherent feature of OCT images, which makes OCT image segmentation a very challenging problem. Hence, Chapter 5 address corneal OCT image segmentation and proposes solutions to four OCT-related problems.

First, low SNR and the lack of light reflections at the peripherals of the cornea causes the classical active models to give inaccurate segmentation results. We address this problem through the usage of a spatially varying kernel that is able to restore the signal at the peripherals. Second, the problem of automatic initialization of the active contour in the noisy OCT images is addressed using the generalized Hough transform (GHT). Finally, we propose solutions to the problems of calculating the epithelium layer thickness and compensating for the eye motion.

Chapter 2

Background

2.1 Image Segmentation

Image segmentation refers to the process of subdividing an image into its constituent meaningful regions or objects. In other words, image segmentation is the process of labeling image pixels, such that pixels that share certain predefined characteristics are assigned the same label [13]. If I denotes the input image, then image segmentation aims to partition I into N subregions $R_1, R_2, \ldots, R_N$ such that

$$\begin{aligned} &I = \cup_{i=1}^{N} R_i \\ &R_i \cap R_j = \phi, \text{for } i \neq j \\ &H(R_i) = \text{true, for } i = 1, 2. \ldots, N \\ &H(R_i \cup R_j) = \text{false, for } i \neq j \end{aligned} \tag{2.1}$$

where H() is a logical predicate that measures the homogeneity of a certain region against a predefined set of features.

Image segmentation is a critical and essential step in most higher level image processing and computer vision applications such as pattern analysis and recognition. Proper segmentation of an image ignores noise and irrelevant data introduced by other objects/background and reduces processing complexity that higher levels need to carry out to produce an interpretation of the image. It is likely that successful image segmentation leads to successful higher level results. In pattern recognition for example, a correct segmentation of the object of interest helps constructing a good representative feature vector of that object, which in turn eases the recognition process. Due to its importance, image segmentation algorithms

are found in a very wide range of applications including medical image processing [14], content-based image retrieval [15] and remote sensing [16].

Image segmentation is considered as one of the most difficult tasks in image processing [17]. The difficulty arises from a variety of reasons including, and not restricted to, non-stationary environment, spatial discontinuity of the objects and determination of the level to which the subdivision or segmentation is carried.

Generally speaking, image segmentation approaches can be divided into two categories: similarity/homogeneity based segmentation and discontinuity based segmentation [18]. In similarity based segmentation, pixels having similar predefined features are grouped in the same partition. Discontinuity based segmentation, on the other hand, partitions the image based on abrupt changes in image features such as color. Both similarity based and discontinuity based segmentation try to answer the same question of "what is the best labeling of image pixels?", and hence should ideally give the same segmentation results. Sections 2.1.1 and 2.1.2 give more details about similarity and discontinuity based segmentation respectively.

Related to similarity-discontinuity categorization, segmentation techniques can also be divided to Bottom-Up (BU) and Top-Down (TD) techniques. BU approaches mainly depend on the continuity principle and group pixels according to their similarity to each other with no prior information used during the segmentation process. Discarding image priors aims to make BU algorithms capable of automatically segmenting images under all environments and for all applications [19]. However, for a variety of reasons, this aim is to some extent not achievable. One reason is that most real images suffer from irregular illumination, shadows, reflections and partially occluded objects. Additionally, the presence of different types of noise, with each noise type treated in different way. Finally and more importantly, the non-fixed definition of "object", the level of details to which segmentation process should continue differs from one application to another. In some applications the purpose of segmentation might be to find the outer boundary of the object, while in other applications finding different textures that constitute the object may be the goal of segmentation [20]. the aforementioned reasons cause a BU algorithm to accidentally split a single object to different objects or merge parts of the object with background due to the unavoidable lake of information that can only be resolved by incorporating prior information.

In Top-Down (TD) approaches, as opposed to BU approaches, segmentation is guided by prior object information such as outer shape or orientation [21]. One limitation of TD approaches is that prior information in most cases cannot capture all variability that a certain object can have. Using higher volume of prior information could overcome this

limitation with the side effect of restricting the approach to a very specific type of image and increasing the processing time.

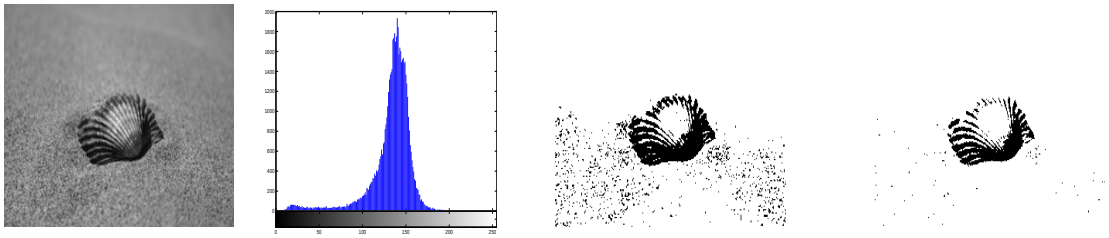
Recently, hybrid solutions of both BU and TD have been proposed that use both image data and general prior information [22] [23] [24]. Using image data makes hybrid approaches as general as BU while using general prior information helps improving and customizing segmentation results [25].

Although a great number of segmentation techniques have been developed, there is still no universally accepted evaluation criteria for segmentation accuracy, which explains the relatively little research done on the segmentation assessment field. Subjective and objective evaluation are two main approaches for segmentation quality assessment [26]. Subjective approaches use human observers to evaluate segmentation results [27]. The two main disadvantages of this approach are: 1) evaluation may significantly differ from one person to another. 2) it is a time consuming process and hence cannot be used for real time application. Objective approaches on the other hand use quantitative criteria for segmentation evaluation [28], and can be further divided to supervised and unsupervised [29] [30]. Supervised approaches require manually segmented images (ground truth) to compare against [29] [31], while unsupervised approaches evaluate segmentation results based only on the segmented image. Examples of used metrics in the later approach could be measuring the intra-object features such as shape regularity or inter-object features such as contrast [32].

In summary, image segmentation is a critical and essential image processing task in which the underlying image is partitioned into meaningful segments. Similarity and discontinuity properties are the most common used features that judge whether a certain pixel belongs to one class or the other. Both approaches are complementary and can be combined to improve segmentation quality. The two following subsections are discussing, in more details, the similarity and discontinuity based segmentation. Segmentation assessment is not an easy task due to the lack of standard quantitative criteria that judge the effectiveness of a segmentation technique. Most segmentation assessment approaches use ground truth databases to evaluate segmentation results.

2.1.1 Similarity Based Segmentation

Similarity based segmentation approaches subdivide the image into regions with each region containing pixels that have similar predefined features. Pixels may be grouped based on one or more features such as intensity, texture and orientation. Similarity based approaches are suitable for segmenting images where objects are homogeneous and have strong contrast



Original I $Hist(I)$ $Thresh(I)$ using [36] Alternative threshold

Figure 2.1: For a non-bi-modal histogram, histogram thresholding does not produce good segmentation results

to each other. For textured objects or objects that have poor contrast relative to other objects, similarity based approaches fail to give good results. Thresholding, region growing, splitting and merging and watershed are considered the main similarity based approaches [33].

Thresholding is the simplest form of similarity based segmentation where a threshold value T is used to segment the image according to

$$f_t(x,y) = \begin{cases} c_1 \text{ if } f(x,y) > T \\ c_2 \text{ if } f(x,y) \leq T \end{cases}$$

where $f(x,y)$ is the gray level of the image I at the spatial coordinate (x,y), $f_t(x,y)$ is the thresholded image and c_1 and c_2 are two different labels (e.g. object and background). Due to its simplicity and intuitive properties, image thresholding received a great research attention in the literature [34] [35]. The key issue in thresholding approaches is the proper selection of the T value that optimally separates image objects. In Otsu method [36], the value of T is selected as the intensity value that maximizes the between-classes variance and minimizes the in-class variance through iterating the dynamic intensity range of the image. Relying only on the intensity value as a measure of discrimination and segmentation restricting the Otsu method to images where objects have high contrast relative to each other and hence restricting the method to images having bi-modal histogram with deep valley between them. In situations where objects in the same image share common intensity values this method fails to properly segment the image (see figure 2.1).

In order to improve the ability of thresholding methods to discriminate objects having common gray values, image can be thresholded based on both gray level information and the spatial correlation between image pixels. A 2-D histogram was developed in [37] with the first dimension represents pixel gray values and the second dimension represents the

average gray values of each pixel and its neighbors. The final threshold T is then selected as the average value of the optimal thresholds on both dimensions T_1 and T_2. The main advantage of this method is its ability of properly separating objects that share common gray values. On the other hand, this method adds the burden of considering both gray level dimension and average gray level dimension when selecting the optimal threshold, which in turn increases the computational complexity of the method. Addressing the computational cost of image thresholding using more than one feature, both pixel gray value and pixel neighborhood gradient were used in [38] to build the 2-D histogram. Pixel neighborhood gradient is the difference between the pixel gray value and its neighbors average gray value. Using pixel neighborhood gradient instead of neighborhood average decreases the dynamic range of this dimension and hence improves the computation time.

Region growing is another similarity based segmentation approach in which each pixel is checked against its neighbors, if the pixel has similar attributes as its neighbor then it is considered as belonging to the same object. Otherwise it forms its own class and this repeats until all pixels are labeled. Region growing assumes spatially continuous objects, which means that all the object pixels are spatially adjacent which is not true in many real images. If the object is scattered along the image, then region growing approaches will consider each group of adjacent object pixels as a separate object (a phenomenon known as over-segmentation). Recognizing the fact that pixels that belong to the same object often appear in spatially scattered location in the image, a method called hierarchical segmentation or simply HSeg was proposed in [39]. HSeg allows spatially disjoint regions to have the same label if they share similar features. In this method, merging spatially disjoint regions is controlled by a parameter S_{wght} which vary from 0.0 to 1.0. This parameter reflects the importance of merging spatially disjoint regions (0-not important and 1-highly important). A dissimilarity criterion value d is calculated between all pairs of regions. A merging threshold T_{merge} is set as the minimum value of d between adjacent regions. Finally, two spatially disjoint regions i and j are given the same label if $d_{i,j} \leq S_{wght} \times T_{merge}$. The obvious benefit of this method is the solution of the over-segmentation problem. However, the intercomparison between each pair of regions in order to compute the dissimilarity value d comes on the expense of computation time. To address this problem, a recursive method called RHSeg was introduced in [40]. In RHSeg the image is first divided into subsection and then the HSeg is applied to each subsection individually. Finally, HSeg is applied to merge subsections.

In region splitting and merging approaches [41] [42], a single region/object that covers all the image is initially assumed. This region is then split into parts, typically four equal parts. For each part R_i, $H(R_i)$ (see (2.1)) is evaluated and only non-homogeneous regions are considered for further splitting. While regions where $H()$ evaluates true are considered

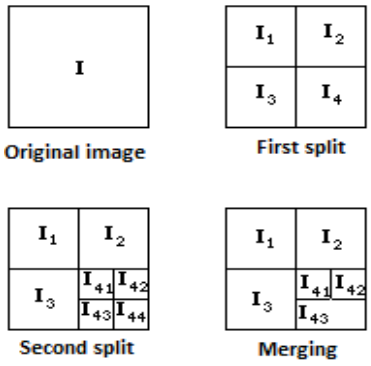
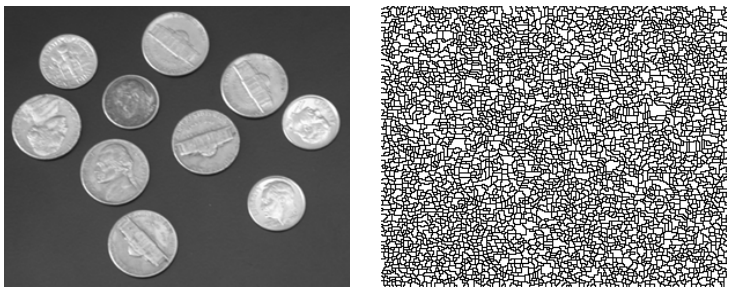


Figure 2.2: split and merge image segmentation



Original I $Watershed(I)$

Figure 2.3: Watershed over-segmentation phenomenon

for merging with their homogeneous neighbors. Figure 2.2 gives an explanatory example for split and merge approaches.

Region split and merge approaches suffer major difficulties such as (i) computation complexity, (ii) parallel implementation is difficult due to dependencies between splitting and merging processes, (iii) Highly depended on selected features and thresholds (merging and splitting), (iv) boundaries between regions are always not smooth. Addressing problems (iii) and (iv), authors of [42] dynamically selected splitting and merging thresholds based on inter- and intra-variances of image regions. At the final step of their algorithm, any region that is less than 6% of its neighbor region size is considered as a part of the larger region.

Finally, watershed segmentation [43] is a similarity based segmentation with its intuitive idea comes from geography. The image can be seen as a topographic relief with pixels

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

(a)

-1	-1	-1
-1	8	-1
-1	-1	-1

(b)

Figure 2.4: A 3×3 (a) general mask, (b) point detection mask

intensity represent their altitude. Each local minimum in the image intensity or gradient represents a catchment basins. The basic idea of watershed segmentation is to place a water source in each regional minimum of the image, and to build barriers at places where the water from different catchment basins meet. Those barriers are the boundaries between image regions. Watershed approaches have been successfully integrated with other discontinuity based segmentation approaches (e.g. active contours) for better segmentation results. However, one main issue regarding watershed in its original definition is that it produces severe over-segmentation of the image due to the large number of local minimas in the image or its gradient [44] (see Figure 2.3).One solution to that problem is to use a set of markers that refer to the objects to be segmented [45]. This solution can be used only in cases where these markers can be defined a head before watershed method runs, which in many cases is not possible. Another solution to the over-segmentation problem is to use a hierarchical watershed approach [46] which aims at merging the catchment basins of the watershed image that belong to similar regions.

2.1.2 Discontinuity Based Segmentation

Discontinuity based approaches for image segmentation aim to locate positions where abrupt changes in the image features such as intensity or gradient happen. Those positions are then recognized as the boundaries between different image regions. Basically, there are three types of discontinuities in a digital image [17]: point, line and edge. In this section detecting the first two types of discontinuities (point and line) is covered, while edge detection is covered in more details in the following section. In the literature, the most common approach to locate different types of discontinuities is to run a mask through the image and evaluate the response of image pixels to that mask [17]. Figure 2.4 shows a 3×3 mask that could be used to locate point discontinuities.

The center of mask is located at each pixel in the image and the response R is calculated

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal

-1	-1	2
-1	2	-1
2	-1	-1

45°

-1	2	-1
-1	2	-1
-1	2	-1

Vertical

2	-1	-1
-1	2	-1
-1	-1	2

−45°

Figure 2.5: Line detection masks for Horizontal, 45°, Vertical and −45° directions

according to

$$\begin{aligned} R &= w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 \\ &= \sum_{i=1}^{9} w_i z_i. \end{aligned} \tag{2.2}$$

where z_i is the gray level of the pixel corresponding to the mask coefficient w_i.

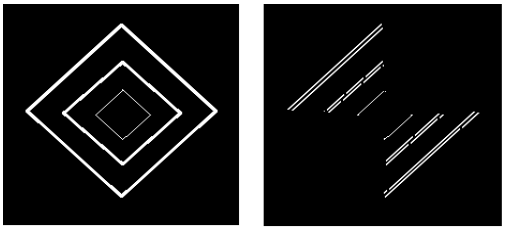
Point discontinuity is the first type of abrupt change that could happen in an image and can easily be detected by running a mask as the one shown in Figure 2.4. If there is a pixel that is highly different in color form its neighbors then locating the mask center at that pixel and computing R will give a result that is much higher in magnitude than computing R for any other pixel in the neighborhood. A simple thresholding process $|R| > T$ will detect all point discontinuities in the image, where T is a non-negative threshold and R is given by (2.2).

Line discontinuity is the second type of abrupt change that happen in an image. Now the problem is no longer to detect an abrupt change at a single pixel, but instead to detect abrupt changes for a series of pixels along a specific direction. Modifying the mask of Figure 2.4 to accommodate this new challenge gives the set of masks in Figure 2.5

To detect a line in an image, four responses R_1, R_2, R_3 and R_4 are computed for each pixel in the image, and the larger response (in magnitude) gives an indication that this pixel most likely lies on a line in the direction of the mask that produced the highest response. Figure 2.6 shows the result of applying the 45° mask to the original image and threshold the responses.

2.2 Edge Detection

Edge Detection is the process of locating the third type of discontinuities that happen in an image, namely edges. An edge is defined as a set of connected pixels that lie on



(a) (b)

Figure 2.6: Line detection using 45° mask (a) original image, (b) thresholded 45° mask responses

the boundary between two regions [17]. An edge can also be defined as locations where significant local changes of image intensity occur [47]. There are several sources that cause sharp changes in image intensity such as inherent image structures (textures), lighting (shadows) and noise. Noise is considered as the main challenge in the process of edge detection since noise has the same definition as edges "an abrupt change in image intensity". Edge detection and boundary extraction are generally ill-posed image processing problems [48] [49]. It is impossible to design an edge detector that only responds to true edges and ignores all other non-real edges and noise. Reducing image noise affects edges (e.g. blurring) and strengthen edges also strengthen noise.

According to their intensity profiles, there are three types of edges (see Figure 2.7):

- Step edge where the abrupt change in the intensity occurs in a very short distance, ideally one pixel. This type of edge usually found in synthesis images and rarely found in real life images.
- Ramp edge where the abrupt changes in the intensity takes a distance longer than one pixel which means that the transition crossing the ramp edge is gradual as opposed to sharp transition in step edges. This gradual transition happens as a result of many factors such as noise and imaging system deficiencies.
- Roof edge, this type of edge happens when a line cut a smooth region of the image. The width of the roof basically depends on the width of the passing line.

First and second derivatives of the image are the common approaches for edge detection. First derivative detectors are covered in the following section, while second derivative detectors are covered in Section 2.2.2.

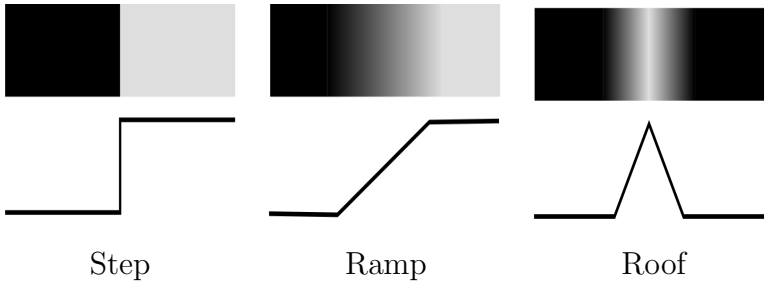


Figure 2.7: Edge models

2.2.1 First Derivative Edge Detectors

One of the most important tools for locating edges in an image I is to compute the first derivative, or simply gradient as follows

$$\nabla I = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix}$$

where $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ are the partial derivatives of the image in x and y directions respectively. In digital image processing, partial derivatives are approximated by forward, backward or symmetric difference. Forward difference in the x direction for example is achieved by: $g_x = I(x+1, y) - I(x, y)$ and in the y direction $g_y = I(x, y+1) - I(x, y)$. Image gradient provide us with two important pieces of information. First, the edge strength which is equal to the magnitude value of ∇I and is calculated as $M(x, y) = \sqrt{g_x^2 + g_y^2}$. Gradient magnitude is used to locate edges in the image by selecting pixels that have high $M(x, y)$. Second, the edge direction which is the direction perpendicular to the gradient direction. The vector ∇I at certain point (x, y) points at the direction of the greatest change of I at that point. The direction is given by

$$\theta(x, y) = tan^{-1} \begin{bmatrix} \frac{g_y}{g_x} \end{bmatrix}$$

In the literature there are many symmetric filters that approximate derivatives in the x and y directions such as Prewitt and Sobel filters [50] (Figure 2.8). All masks coefficients in Figure 2.8 sum to zero which gives a response of zero in constant intensity regions of the image. Building symmetric 3×3 approximation filters involves data from the opposite sides of the center pixel and hence better extracts the edge direction information. The only variation between Prewitt and Sobel masks is the use weight 2 in the center coefficient,

-1	-1	-1
0	0	0
1	1	1

g_x

-1	0	1
-1	0	1
-1	0	1

g_y

Prewitt

-1	-2	-1
0	0	0
1	2	1

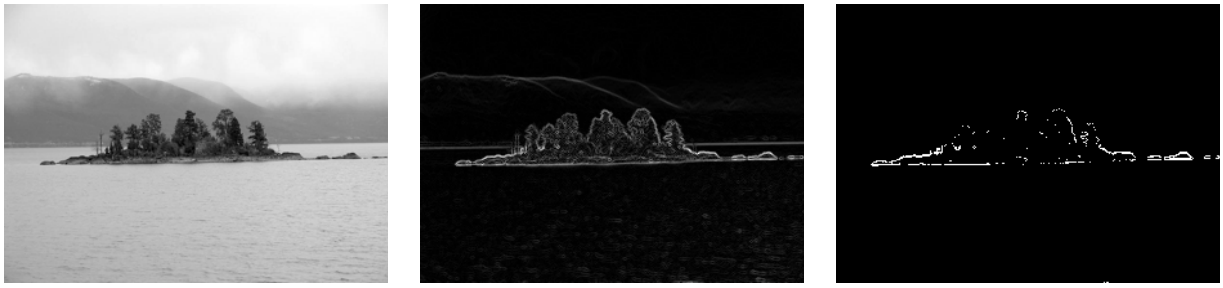
g_x

-1	0	1
-2	0	2
-1	0	1

g_y

Sobel

Figure 2.8: Examples of partial derivative approximations



Original I $|\nabla(I)|$ $Thresh(|\nabla(I)|)$

Figure 2.9: Gradient image usually has thick edges that need to be thresholded

which has the effect of noise suppression. As a direct result of having ramp edges in the image, first derivative filters produce thick edges that need to be thresholded. Figure 2.9 illustrates the problem of thick edges and how to solve this problem using thresholding. Threshold selection is a key issue when using gradient information for edge detection. Selecting one global threshold for the whole gradient image almost produces irrelevant edges that are of no concern to human visual system. Addressing this issue, in [51] [52] different set of local thresholds were used in order to produce perceptually better edge maps. The gradient image is firstly thresholded using a low threshold value to find all potential edges in the image. Secondly, for each potential edge pixel, weak edges in the 5×5 local region of this pixel are removed based on the activity masking property of human visual system. In brief, the activity masking property states that edges near loose edges or in smooth regions are more noticeable to our eyes than edges in textured regions.

In his pioneering work [53], Canny defined three objectives to be satisfied by the edge detector

1. Low error rate, which means that only true edges should be detected,
2. Edge localization, which means that detected edges should be as close as possible to

the actual edges and

3. Single edge response, which means that only a single response to a true edge point should be recorded.

Canny's approach can be summarized as follows:

- Smooth the image using a Gaussian filter
- Compute the magnitude and angle of the gradient image.
- Apply non-maxima suppression to the gradient magnitude image
- Use hysteresis thresholding and connectivity analysis to detect and connect edges

The Canny method first convolves the image with a Gaussian filter to reduce noise effect. After calculating the gradient magnitude M and the direction θ, For each edge point (x, y) in θ image, quantize the edge normal at this point to one direction of a predefined set of directions (for example, four directions horizontal, vertical, $-45°$ and $+45°$). If the magnitude value of the edge pixel (x, y) is less than one of its two neighbors along the quantized direction, then this pixel is suppressed. Which means that the pixel was not a true edge pixel and hence removed from the final edge map. After noise suppression step, the final image is then thresholded in order to remove any weak edges. Canny used two threshold values T_{low} and T_{high}. All pixels having gradient magnitude $\geq T_{high}$ are marked as true edges and all pixels having values $< T_{low}$ are removed from the final edge map. Finally, All pixels having values in the interval $[T_{low}, T_{high}[$ are considered edge pixels only if connected to an edge pixel through neighbors of value $\geq T_{low}$.

Despite the fact that the Canny detector is one of the most used methods for edge detection, Canny's method has some issues regarding its performance. One issue is related to the selection of two thresholds T_{high} and T_{low} which is difficult to select and usually produces broken and false edges. Another issue is related to the Gaussian blurring step which in fact weaken both noise and edge pixels. Addressing these issues, [54] uses a self-adaptive smoothing filter that changes its weights according to the features of the underlying region. In smooth regions large weights are used and in regions where high intensity changes are detected smaller weights are used. Regarding the broken and false edges problem of the Canny method, an edge linking step was added in [54] to the maximum suppression step of the Canny algorithm. The condition states that after calculating the quantized direction of the normal to the edge pixel in the θ image, at least one neighbor pixel to the current pixel should have the same quantized direction in order to label the current pixel as an edge pixel. This condition excludes noise pixels which by nature have

random directions and preserves edge pixels that have consistent directions. Which in turn improves edge connectivity and error rate.

2.2.2 Second Derivative Edge Detectors

Image second derivative ∇^2 is another tool that is highly used for edge detection in the literature [55] [56]. The second derivative of an image I in the x direction is approximated as

$$\frac{\partial^2 I}{\partial x^2} = I(x+1, y) - 2 \times I(x, y) + I(x-1, y)$$

and in the y direction as

$$\frac{\partial^2 I}{\partial y^2} = I(x, y+1) - 2 \times I(x, y) + I(x, y-1)$$

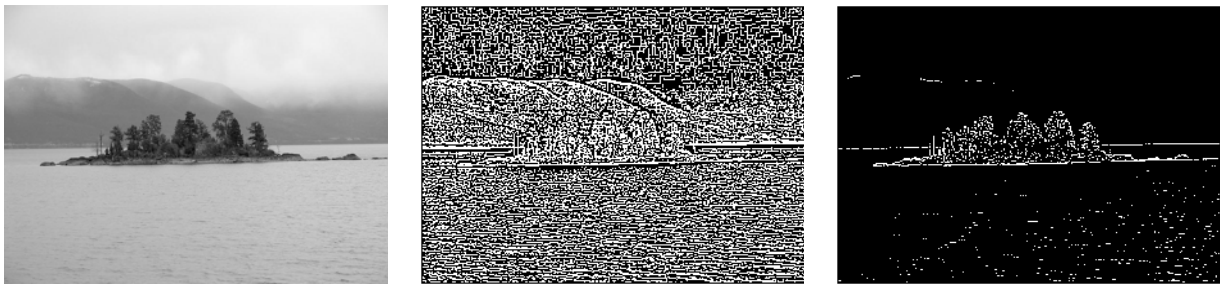
Ideally, the second derivative of an image should have zero crossing at edge pixels as opposed to low or high values for the first derivative. The second derivative operator ∇^2 has the advantage of being isotropic as opposed to the directional first order operator ∇, which means that there is no need to run multiple masks in order to compute the strongest edge response. Using second derivatives for edge detection facing two limitations: (1) it has a strong response to fine detail, such as thin lines, isolated points, and noise, and (2) it produces a double-edge response to image edges (the positive pixel on one side and the negative one on the other). Addressing those limitations, Marr-Hildreth [57] proposed their second derivative based edge detector known as Marr-Hildreth detector.

In their method, Marr and Hildreth addressed the noisy sensitivity problem of the second derivative by initially smoothing the image with a Gaussian blurring filter. Using Gaussian blurring mask reduces image noise, and more importantly because Gaussian function is smooth in both spatial and frequency domains, it is less likely to introduce ringing artifacts compared to any other averaging masks. After blurring the image, the second step is to compute the second derivative of the smoothed image. The two steps can be written together as

$$g(x, y) = \nabla^2 \left[G(x, y) * I(x, y) \right]$$

where $*$ is the convolution operator and $G(x, y)$ is a 2-D Gaussian function given by

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$



Original I $LoG(I)$ $Thresh(LoG(I))$

Figure 2.10: Noise sensitivity and double edge response of Marr-Hildrith approach

Because ∇^2 and $*$ are linear processes, the above equation can be rewritten as

$$g(x,y) = \left[\nabla^2 G(x,y)\right] * I(x,y)$$

where $\nabla^2 G(x,y)$ is called Laplacian of Gaussian (LoG) and is given by

$$\begin{aligned} \nabla^2 G(x,y) &= \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2} \\ &= \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} \end{aligned} \tag{2.3}$$

LoG can be approximated using Difference of Gaussian (DoG) as follows

$$DoG(x,y) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2+y^2}{2\sigma_1^2}} - \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}}$$

the last step of Marr-Hildreth approach is to find all zero crossing locations in the resulted image $g(x,y)$ and mark them as edge pixels. In practice, a thresholding step is still needed in order to remove all false and double response edges (see Figure 2.10).

2.2.3 Edge Linking

Edge linking is an essential step that usually comes after edge detection with the aim of producing a consistent and connected set of edge pixels. Local and Global processing are the two main approaches for edge linking. In local edge linking, the neighborhood of each edge pixel is checked for similarity according to a set of predefined criteria. If a certain

level of similarity between them exists, they are connected. the gradient magnitude M and the direction θ are the most used criteria for connecting edge pixels [58] [17]. For each pixel (x_n, y_n) in the neighborhood of an edge pixel (x, y) let:

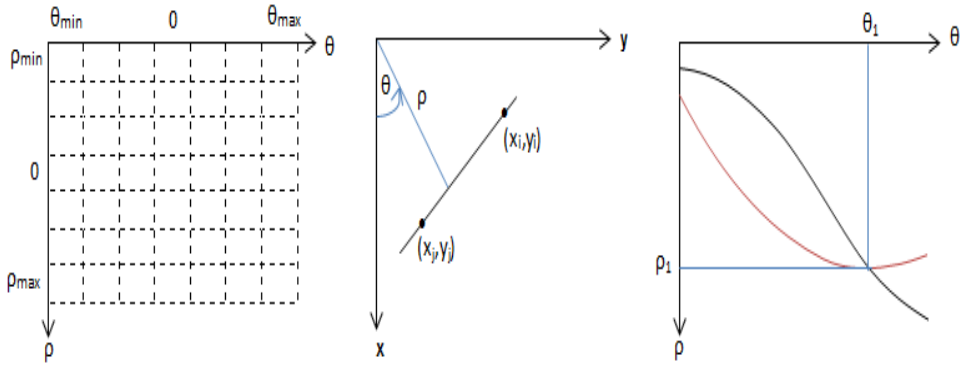
$$e(x_n, y_n) = \begin{cases} 1 & \text{if } |M(x,y) - M(x_n, y_n)| \leq E \text{ and } |\theta(x,y) - \theta(x_n, y_n)| \leq A \\ 0 & \text{otherwise} \end{cases}$$

where E and A are positive thresholds and $e()$ is the final edge map. Finally, a last step of filling (set to 1) all gaps that do not exceed a specified length K between edge pixels is carried out.

Global approaches for edge linking are applicable in situations where there is no prior information about the location or the shape of the region of interest in the edge map image. The Hough transform is considered as one of the most popular global methods for edge detection and linking [59]. The Hough transform is usually used to connect edge pixels that lie on a line, a circle or any other parametric curves. Assuming linking edge pixels that lie on lines for simplicity, the simple idea behind the Hough method is to convert the slope-intercept equation of all lines $y_i = mx_i + b$ passing through an edge point (x_i, y_i) which is an infinite number of (m, b) pairs to another plane ($m - b$ plane or parameter plane) where all solutions to that equation form a line in the new plane. The new equation becomes $b = -x_i m + y_i$. The new representation has the limitation of being unable to represent vertical or semi vertical lines (slope approaches infinity). A simple solution to this problem is to use the polar representation of a line $x \cos\theta + y \sin\theta = \rho$. In the new $\rho - \theta$ space all solutions to the equation will from a sinusoidal curve. To link edges using the Hough transform, an accumulator H that divides the $\rho - \theta$ space into cells or bins of a predefined resolution is set and initialized to zero (see Figure 2.11). The algorithm goes as follows

for all image pixels (x_i, y_i) **do**
  if (x_i, y_i) is an edge point **then**
    for $-90° \leq \theta \leq 90°$ **do**
      compute $\rho = xcos\theta + ysin\theta$
      increment the cell corresponding to (ρ, θ) in the accumulator H by 1
    end for
  end if
end for

after scanning all edge pixels in the image, a value v in an accumulator cell $H(i, j)$ means that v edge points encourage an edge line to be exist between them. This line is defined by the pair (ρ, θ) corresponding to that cell in the accumulator. Although the Hough transform



(a) (b) (d)

Figure 2.11: The Hough transform: (a) The accumulator H. (b) Two points connected by an edge in xy plane (i.e. the image). (c) Sinusoidal curves which represent all lines passing by (x_i, y_i) and (x_j, y_j) intersect at (ρ_1, θ_1) which identifying the line passing by the two points, this increments the corresponding cell of (ρ_1, θ_1) in H by 1.

is a robust edge linking method that is extensively used in the literature [60] [61] [62], it still suffers the major bottlenecks of computational complexity and high storage requirements especially when the Hough transform is used to link edges that have complex shapes like circles and parabolas.

In [63], both problems of the traditional Hough transform were addressed using a new probabilistic version of the transform. The new method called Random Hough Transform (RHT) with the basic idea of randomly sampling points from the edge map and incrementing the corresponding bins in the accumulator H instead of visiting every single pixel in the image. Authors of the RHT stated that if the random sampling process continued long enough, the local maximas locations in H will be the same for the both traditional Hough transform and the RHT. The problem arises again, running RHT "long enough" would be computationally and storage expensive. A hierarchical version of the Hough transform was presented in [64] where the parameter space (ρ-θ in the simple case of line edges) is divided recursively from coarse to fine scale. An initial run of the Hough transform is performed on a low resolution parameter space where a single bin in the accumulator H represents relatively big area of the space (low resolution accumulator). Then a higher resolution H is generated that only considers bins which exceed a threshold value T from the previous lower resolution H. The process continues until a predefined resolution is reached or there are no cells that exceed the threshold T.

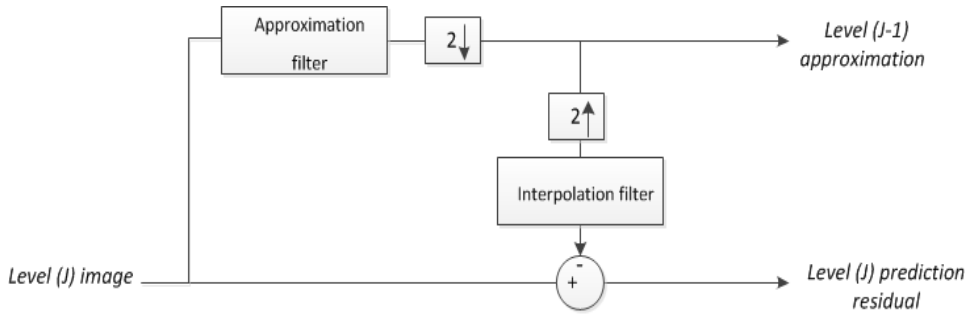


Figure 2.12: Building image pyramid using approximation and interpolation filters. Approximation is a lower image resolution while prediction residual encodes differences between predicted and real image.

2.3 Multiresolution Image Processing

Multiresolution approaches provide a powerful tool in the field of image processing where the underlying image features/structures can be handled at more than one resolution. The simple idea behind multiresolution theory [65] is that features that are undetectable at one resolution might be easily detectable at another one. Objects that are large in size or high in contrast are normally examined at low resolutions while objects that are small in size or low in contrast are examined at high resolutions.

Image pyramid [66], as the simplest multiresolution method, is a collection of the original image at decreasing resolutions arranged in a pyramid shape. The pyramid base contains a high resolution approximation of the image (usually the original image itself) and the apex contains a low resolution approximation. Usually both size and resolution decrease as you move up the pyramid. The size of the base level J is $2^J \times 2^J$ and the apex level 0 is of size 1×1. Most multiresolution approaches stop away before the level 0, as an image of size 1×1 is indeed of a little value. Figure 2.12 gives a simple system for generating two intimately related layers of the pyramid. The $2\uparrow$ and $2\downarrow$ in the figure refer to upsampling and downsampling respectively. The approximation filter could be an averaging filter which produces a mean pyramid [67], a lowpass Gaussian filter which produces a Gaussian pyramid [68] or other filters. The interpolation filter could be as simple as nearest neighbor, bilinear or cubic filter. In order to build a P layers pyramid, the following three steps are repeated P times, assume starting from level j

1. A low resolution version of the approximation image at level j is generated through filtering followed by downsampling. The resulted approximation image is the level

$j-1$ approximation image.

2. An estimate of the level j image is generated through upsampling followed by filtering the image generated in the previous step.
3. The difference between the generated image from step 2 and the input image from step 1 is stored as the prediction residual in level j of the pyramid

The importance of the last step is that it both enables the full reconstruction of the original image and more importantly, unlike approximation image, residual images tend to have histograms that are mostly centered at zero. This gives the chance of highly compressing those images by assigning fewer bits to non-frequent intensity values in the image. Which in turn makes signal/image compression using pyramid technology extensively used in the literature [69] [70].

A more generalized multiresolution representation that uses any function as opposed to Gaussian function in the Gaussian pyramid is the wavelet transform which is covered in the following section.

2.3.1 Wavelet Transform

Wavelet transform is a multiresolution transform based on small functions (called wavelets) that are localized in both time and frequency [71]. Talking about wavelet inherently means talking about a pair of functions namely, the scaling φ and the wavelet ψ functions. The scaling function φ is used to generate the image approximations while wavelet function ψ is used to encode the difference in information between successive approximations (i.e. detail images). According to series expansion, a function $f(x)$ can be represented as a weighted sum of φ as follows

$$f(x) = \sum_{k} \alpha_k \varphi_k(x) \tag{2.4}$$

where α_k are the expansion coefficients and φ_k are the expansion functions. Functions that can be expressed by the set $\{\varphi_k(x)\}$ forms a function space that is referred to as the closed span of the expansion set, denoted as

$$V = \overline{\text{Span}_k\{\varphi(x)\}}$$

the set of expansion functions that are composed of translations and scalings of the original expansion function is

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k) \tag{2.5}$$

where j and $k \in \mathbf{Z}$. The position of $\varphi_{j,k}(x)$ along x-axis is controlled by k while its width is controlled by j. The amplitude of the function is controlled by the term $2^{j/2}$. Restricting the set $\{\varphi_{j,k}(x)\}$ to a specific value j_0 produces a subset $\{\varphi_{j_0,k}(x)\}$ that spans a subspace of $L^2(\mathbf{R})$. This new subspace is defined as

$$V_{j_0} = \overline{\text{Span}_k\{\varphi_{j_0,k}(x)\}} \tag{2.6}$$

hence, if $f(x) \in V_{j_0}$ then it can be decomposed as

$$f(x) = \sum_k \alpha_k \varphi_{j_0,k}(x) \tag{2.7}$$

as j increases the subspace V_j increases which allows scaling functions with small variations and fine details to be included. One fundamental requirement of multiresolution analysis [65] is that subspaces spanned by low resolution scaling functions must be nested within those subspaces spanned by higher resolution functions, which can be written as

$$V_{-\infty} \subset \ldots \subset V_{-1} \subset V_0 \subset V_1 \subset \ldots \subset V_{\infty} \tag{2.8}$$

hence, expansion functions in arbitrary subspace V_j can be represented as a weighted sum of expansion functions in a higher resolution subspace V_{j+1} using (2.4) as follows:

$$\varphi_{j,k} = \sum_n \alpha_n \varphi_{j+1,n}(x) \tag{2.9}$$

where the summation index k is replaced with n to avoid confusion. Using (2.5), the above equation can be rewritten as

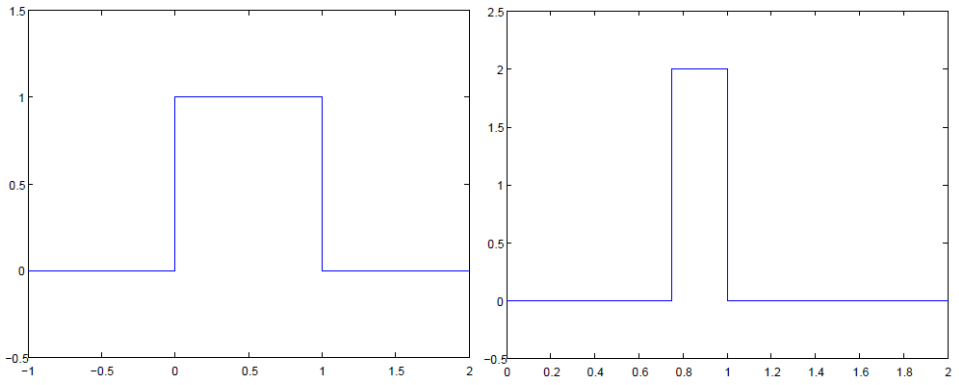
$$\varphi_{j,k} = \sum_n h_\varphi(n) 2^{(j+1)/2} \varphi(2^{j+1}x - n) \tag{2.10}$$

where $h_\varphi(n)$ are called the scaling function coefficients. Because $\varphi_{0,0}(x) = \varphi(x)$, both j and k can be eliminated from the previous equation to have the simpler equation of

$$\varphi(x) = \sum_n h_\varphi(n) \sqrt{2} \varphi(2x - n) \tag{2.11}$$

Equation (2.11)is called the refinement equation and simply means that expansion functions in any subspace can be built from expansion function in the next higher resolution subspace. The most simple scaling function is the Haar function [72] shown in Figure 2.16. The Haar scaling function φ is given by

$$\varphi(x) = \begin{cases} 1 \text{ if } x \in [0,1) \\ 0 \text{ otherwise} \end{cases}$$



(a) Standard Harr function $\varphi_{0,0}(x)$

(b) Scaled-translated version $\varphi_{2,3}(x)$

Figure 2.13: Haar functions, higher resolution scaling functions (e.g. (b)) can be used to construct lower resolution functions (e.g. (a))

Details are encoded using the wavelet function ψ. The set of wavelets that are composed of translations and scalings of the original wavelet can be written as:

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k) \tag{2.12}$$

as with the scaling function, a subspace W_j is defined as

$$W_{j_0} = \overline{\text{Span}_k\{\psi_{j_0,k}(x)\}} \tag{2.13}$$

the scaling and wavelet subspaces are related by

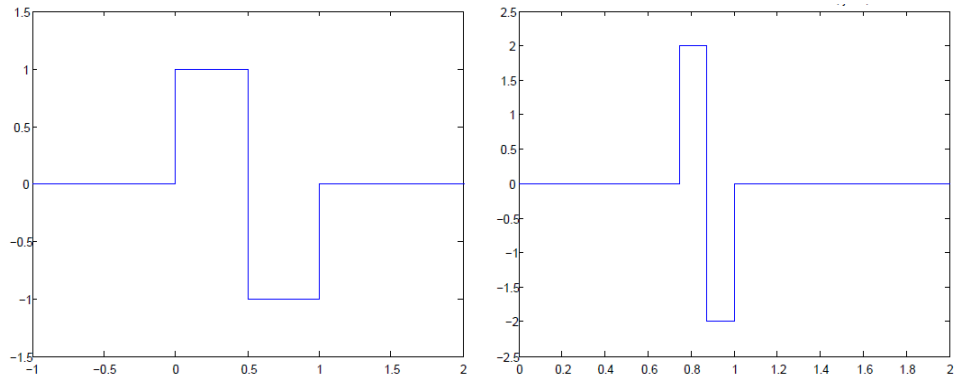
$$V_{j+1} = V_j \oplus W_j \tag{2.14}$$

where $\oplus$ denotes the union of spaces. Using (2.11) a wavelet function can be expressed as

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\varphi(2x - n) \tag{2.15}$$

where $h_\psi(n)$ are called the wavelet function coefficients, and are related to scaling coefficients by

$$h_\psi(n) = (-1)^n h_\varphi(1 - n)$$



(a) Standard Harr wavelet $\psi_{0,0}(x)$

(b) Scaled-translated version $\psi_{2,3}(x)$

Figure 2.14: Haar wavelets, low resolution wavelets (e.g. (a)) are used to encode details in lower resolution layers, while details in higher resolution layers are encoded with high resolution wavelets (e.g. (b))

Haar wavelet is a well known wavelet that is used in the literature [73] [74] and is given by

$$\psi(x) = \begin{cases} 1 \text{ if } x \in [0, 0.5) \\ -1 \text{ if } x \in [0.5, 1) \\ 0 \text{ otherwise} \end{cases}$$

Figure 2.14 shows the original Haar wavelet and one of its scaled-translated versions. Putting things together, a function $f(x) \in L^2(\mathbf{R})$ can now be represented by a scaling function expansion at certain subspace V_{j_0} and wavelet expansions at subspaces $W_{j_0}, W_{j_0+1} \ldots$ as follows

$$f(x) = \sum_k c_{j_0}(k)\varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty}\sum_k d_j(k)\psi_{j,k}(x) \tag{2.16}$$

where $c_{j_0}(k)$ are approximation coefficients and $d_j(k)$ are detail coefficients. The one-dimension wavelet transform is easily extended to the two-dimension space, where two dimensions scaling function is defined as follows

$$\varphi(x, y) = \varphi(x)\varphi(y)$$

and three wavelet functions are defined as

$$\psi^H(x, y) = \psi(x)\varphi(y)$$

$$\psi^V(x,y) = \varphi(x)\psi(y)$$
$$\psi^D(x,y) = \psi(x)\psi(y)$$

where ψ^H, ψ^V and ψ^D measure the image variations along horizontal, vertical and diagonal directions respectively. Now, for an image $f(x,y)$ of size $M \times N$, the wavelet transform is computed as follows

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y) \tag{2.17}$$

$$W_\psi^i(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\psi_{j_0,m,n}^i(x,y), \quad i = \{H,V,D\} \tag{2.18}$$

where $W_\varphi(j_0,m,n)$ is the image approximation at certain scale j_0 and $W_\psi^i(j_0,m,n)$ is the wavelet coefficients for the different i values. The inverse wavelet transform can be obtained using:

$$\begin{aligned} f(x,y) =& \frac{1}{\sqrt{MN}} \sum_m \sum_n W_\varphi(j_0,m,n)\varphi_{j_0,m,n}(x,y) \\ &+ \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{\infty} \sum_m \sum_n W_\psi^i(j,m,n)\psi_{j,m,n}^i(x,y) \end{aligned} \tag{2.19}$$

Discrete Wavelet Transform (DWT) can be implemented using successive lowpass, highpass filtering and downsampler, where the lowpass filters produce the approximation images at different scales and the highpass filters produce the detail images. Both the approximations $W_\varphi(j,m,n)$ and the details $W_\psi^i(j,m,n)$ of scale j can be computed by convolving $W_\varphi(j+1,m,n)$ with the filters $h_\varphi(-n)$, $h_\psi(-n)$, $h_\varphi(-m)$ and $h_\psi(-m)$ as shown in Figure 2.15. As can be seen from the figure, the separable filters are applied first in one dimension (e.g. vertically) then to the other dimension.

2.4 Overcomplete Wavelet Transforms

The basic idea behind the overcomplete wavelet transforms is very simple. After applying the lowpass and highpass filters to the data, the downsampling step is skipped. Hence, the output sequences (after the filtering step) have the same dimensions as the input data. However, the used filters are upsampled by 2 each time. Since more output information

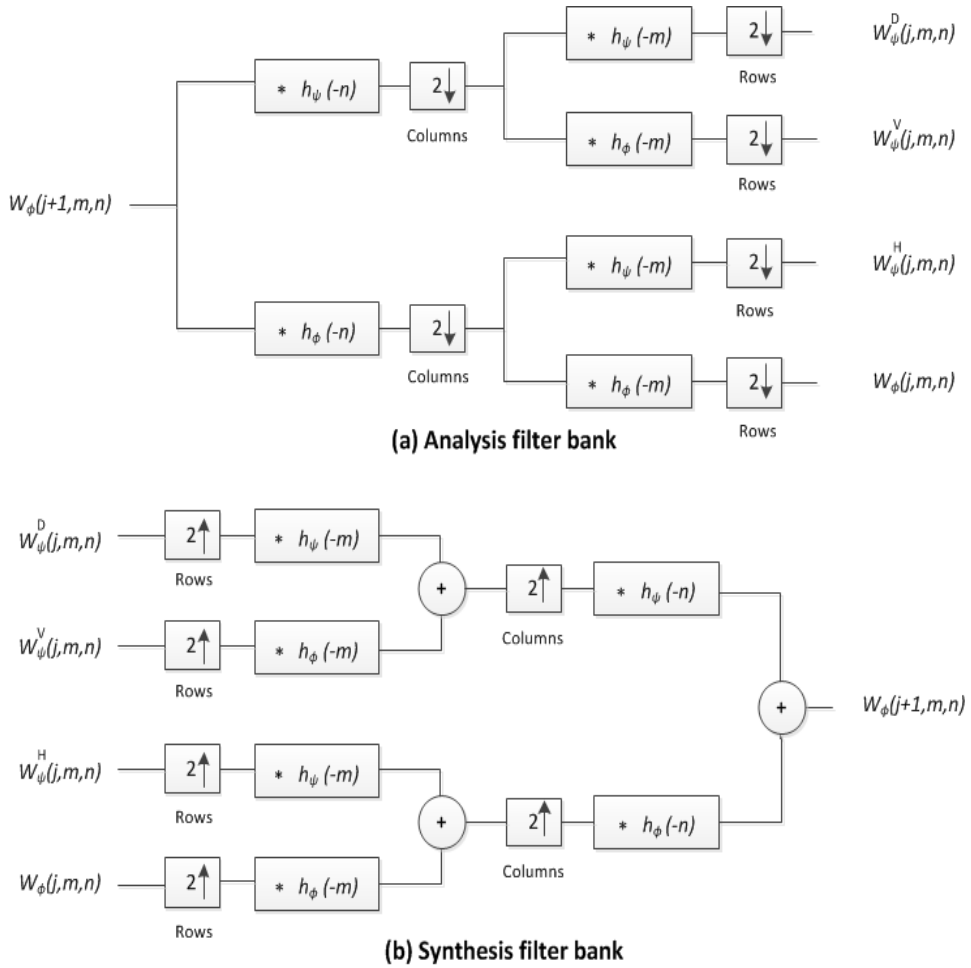


Figure 2.15: Wavelet decomposition (a) and reconstruction (b) of the $j+1$ layer approximations. The wavelet transform and its inverse is implemented using lowpass, highpass filters and down/up-sampler

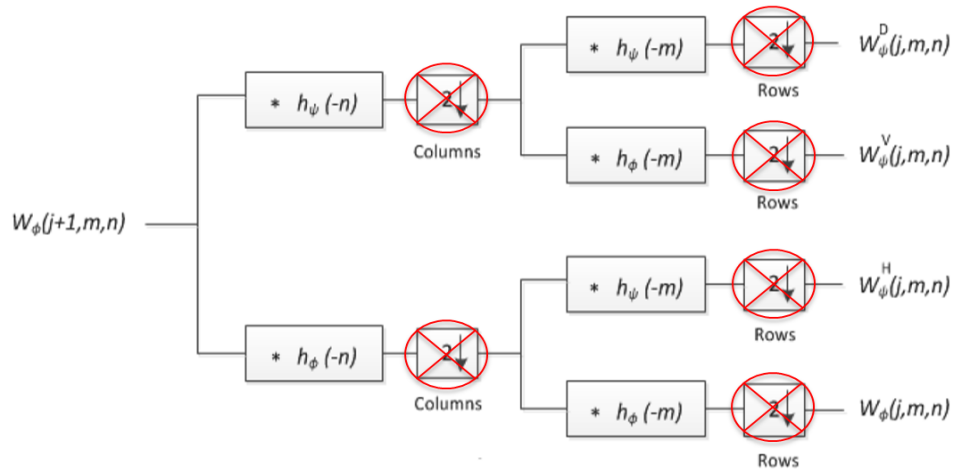


Figure 2.16: The undecimated wavelet transforms. The basic idea is to cancel the downsampling step of the classical wavelet transform. The direct result is a redundant, overcomplete, and shift invariant transform. The used filters in each level are up-sampled versions of the previous filters.

is kept at each level of the transform, overcomplete wavelet transforms have been called redundant transforms in the literature [6].

The lose of the translation invariance property, due to the downsampling step, in the classical wavelet transform makes the results of applying this transform to problems such as deconvolution, detection and data analysis, far from optimal [75]. These non-optimal results are basically because of the large number of artifacts happening when reconstructing the image after modifying the wavelet coefficients. On the other hand, the absence of the downsmapling in the oversomplete transforms makes them translation invariance and eliminates the reconstruction artifacts. Figure refUDWTFiltrers illustrates the idea of the undecimated wavelets where, as mentioned earlier, the basic idea is to cancel the downsampling step and upsample the filters themselves by 2 each time.

2.5 Snakes: Active Contour Models

Active contour models, or simply snakes, are deformable curves used to contour and edge detection. The process of locating image boundary through active contour models is controlled by two types of forces, a force coming form the active contour itself (internal force)

that controls the prior behavior or curvature of the snake, and the other force coming from the image data (external force) that pulls the snake towards features of interest.

Active contours models can generally be classified as either parametric or geometric models. In the first type, contours are explicitly represented as parameterized curves, while in the second category contours are implicitly represented as level sets of two dimensional scalar functions. Sections 2.5.1 and 2.5.2 give more details about the two models.

2.5.1 Parametric Active contours

Parametric Active Contour (PAC), introduced by Kass et al. [76], is an energy minimizing curve that deforms over a series of time steps. The parametric snake curve is defined as

$$\upsilon(s) = (x(s), y(s)) \text{ and } s \in [0, 1] \tag{2.20}$$

where $x(s)$ and $y(s)$ are the active contour coordinates in terms of a normalized arc length s. There are three ways for representing curves in the literature [77]

- Explicit $y = f(x)$, where one variable is defined in terms of the other.
- Implicit $f(x, y) = 0$, where the curve is represented as the zeros of a specific formula. The formula implicitly defines the relation between its arguments and its value.
- Parametric $\upsilon(s) = (x(s), y(s))$, where the curve parameters are represented as functions of another parameter.

The explicit representation of curves prevents one parameter (e.g. y) form having multiple values for a single value of the other parameter (x). Closed curves like circles can not be represented with the explicit representation. Another limitation with this type of representation is the difficulty of representing curves with infinite slope. With implicit curve representation the equation may have more solutions than we want, for example in the circle equation $x^2 + y^2 = 1$, a half of that circle can not be represented with out adding an extra constraint such as $x \geq 0$. The parametric curve representation overcomes the limitations of both other representation methods where multiple values of one term y can be computed with respect to another term x since both of them are computed with respect to another term s. Additionally, the infinite slope problem is solved by replacing geometric slope with parametric tangent which never has infinite slope.

Active contour model tries to segment the image by minimizing the snake energy

$$E_{snake} = E_{internal} + E_{external} \tag{2.21}$$

where $E_{internal}$ is the snake internal energy that aims to keep the curve smooth by preventing stretching and bending, and $E_{external}$ is external energy that pulls the snake towards the desired object boundaries. More formally the above equation can be written as

$$E = \int_0^1 \frac{1}{2}[\alpha(s)|\upsilon^{'}(s)|^2 + \beta(s)|\upsilon^{''}(s)|^2] + E_{external}(\upsilon(s))ds \tag{2.22}$$

where α and β are weighting parameters, $\upsilon^{'}$ and $\upsilon^{''}$ are the first and second derivative of $\upsilon(s)$ with respect to s. The external energy $E_{external}$ is derived from the image data such that it has a minimum value at the features of interest. Typically, the contour should locate edges, so $E_{external}$ could be assigned the negative value of image gradient $-|\nabla I(x,y)|$ or its smoothed version $-|\nabla[G_\sigma(x,y) * I(x,y)]|$. For numerical implementation purposes, the energy function E needs to be discretized.

The traditional active contour method [76] has some limitations [10] [78] [24] including the inability of capturing complex-curvature shapes, inability to detect multiple objects, noise sensitivity, initialization sensitivity, small capture range, slow convergence and parameter dependency.

Addressing small capture range, noise sensitivity and slow convergence problems, most developed approaches altered the external energy term. In [79] the external energy term was modified by considering the curve as a balloon and adding a new inflation force term (called pressure force) defined as

$$E_{pressure} = k_1 n(s), \tag{2.23}$$

where $n(s)$ is the normal unitary to the curve at $\upsilon(s)$ and k_1 is the amplitude of the force. The sign of k_1 determines wether the balloon will be inflated or deflated. The pressure force prevents the active contour from being trapped by the local minimas (spurious edges and noise). Ideally, the pressure force dynamically can either inflate or deflate the balloon to reach its final destination, hence eliminates the need to initialize the model near the object boundaries. The inflation force also prevents the curve from turning to a point or a line when is not submitting to external force. One limitation of the balloon model is that the model needs to be initialized either inside or outside the object to be detected. Another limitation of this model is that if the pressure force is not set properly, a leakage problem occurs at broken edges.

The Gradient vector flow (GVF) [7] is another variation of the traditional active contour mode,l which addresses the initialization and the complex-curvature problems. The main idea of the GVF is to extend the gradient force away from edges and into homogenous regions via computational diffusion process. By doing this, homogeneous regions no longer

have no/weak external force, which is the main reason behind the slow and the misconvergence problems. A gradient vector flow field $g(x,y) = [u(x,y), v(x,y)]$ is defined to minimize the following energy function

$$E = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |g - \nabla f|^2 dxdy \tag{2.24}$$

where μ is a regularization parameter and f is the image edge map. According to (2.24), when $|\nabla f|$ is large (near edges), setting g to large value minimizes E and when $|\nabla f|$ is small (homogeneous regions), squares of the partial derivatives of g dominate the energy function which gives the effect of slowly-varying in homogeneous regions as opposed to no-force in the traditional active contour model. In [80] a Generalized version of the GVF called GGVF is proposed to increase the ability of the GVF to capture narrow boundary concavities by dynamically controlling the regularization parameter μ to decrease smoothing effect near strong gradients (edges). Although the GVF and the GGVF have been used extensively in the literature [81] [82] [83], they suffer some limitations such as small capture range, noise sensitivity, parameter sensitivity and high computational cost [10] [78].

Li and Acton [8] addressed the aforementioned GVF problems via replacing the standard external force term by a Vector Field Convolution (VFC) force term f_{vfc}. A vector field kernel $k(x,y) = [u(x,y), v(x,y)]$ is defined where all the vectors point to the kernel origin. The VFC external force f_{vfc} is then calculated as the convolution of the vector field kernel k and the image edge map $f(x,y)$ as follows

$$\begin{aligned} f_{vfc}(x,y) =& f(x,y) * k(x,y) \\ & [f(x,y) * u(x,y), f(x,y) * v(x,y)] \end{aligned} \tag{2.25}$$

where $k(x,y)$ is the product of a scalar function $m(x,y)$ that assigns magnitudes to the forces and a unit vector field $n(x,y)$ that controls the vectors orientation. m and n are defined as

$$n(x,y) = [-x/r, -y/r] \tag{2.26}$$

$$m_1(x,y) = (r+\varepsilon)^{-\gamma} \text{ or } m_2(x,y) = \exp(-\frac{r^2}{\varsigma^2}) \tag{2.27}$$

where the kernel radius $r = \sqrt{x^2 + y^2}$ and ε is a small constant that prevents division by zero. The range of VFC is governed by r in case of m_1 is used or by ς in case of the Gaussian decay m_2 is used.

Because the edge map $f(x,y)$ is non negative and has large values near image edges, the edges contribution to the new external force f_{vfc} is larger than homogeneous regions which

in turn attracts the snake to the object boundary. The convolution in (2.25) simulates the diffusion process carried in [7] via one step convolution that can be computed in the Fourier domain which in turn reduces the computation time of the method. Nevertheless that VFC has a large capture range and a low computation cost, VFC has its shortcomings. The kernel size is one limitation of the method. Large kernels improve the capture range of the model, but at the same time, overwhelm fine details in the image, while small kernels improve fine details preservation but decrease the capture range.

All parametric based active contour models discussed so far have the limitation of being parameter dependent, which in most cases prevents the model from giving good results under different circumstances. Non parametric active contour models provide a solution to this limitation via processing the active contour in the context of level set method. Non parametric models are discussed in the following section.

2.5.2 Non Parametric Active contours

One limitation of parametric active contour models is the lack of parametrization independence. Level set methods [84] provided an alternative way for evolving the active contour by considering the active contour as a zero level set of a scalar function in a higher dimension which in turn allows curve splitting and merging. The first non parametric active contour (also called Geometric Active Contour GAC) model was introduced by Caselles in [85] and Malladi [86] where the curve evolves using only geometric measures and hence independently of the curve parameters. If $\phi(x,t)$ is a 2-D scalar function with its zero level set defines the active contour such that the function is positive inside the contour, zero on it and negative outside, then ϕ can evolve according

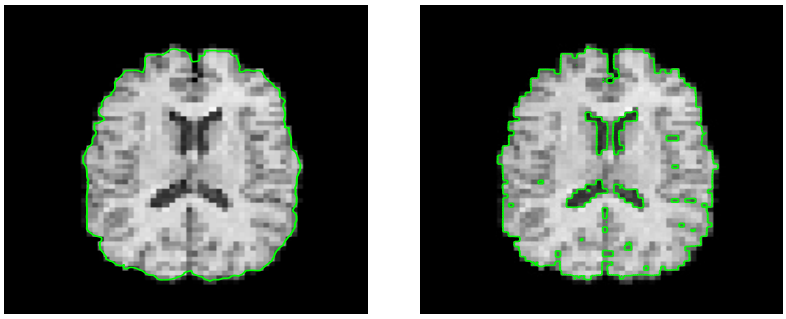
$$\phi_t = c(k + V_0)|\nabla\phi| \tag{2.28}$$

where k is the curvature, V_0 is a constant and c is the edge potential derived from the image as follows

$$c \equiv c(x) = \frac{1}{1 + |\nabla(G_\sigma(x) * I(x))|} \tag{2.29}$$

the term $c(k + V_0)$ in (2.28) controls the evolution speed of the level sets of $\phi(x,t)$, while the curvature term k controls the smoothness of the contour.

The above mentioned geometric model works well when objects are highly distinguishable and have high contrast comparing to the background. But in cases when object boundaries are indistinct or broken, a leakage problem occurs. An extra stopping term



A PAC method [7] A GAC method [87]

Figure 2.17: A parametric vs. a geometric active contour method. The parametric model is capable of locating only one object while the geometric model, due to the thresholding step in the level set method, can definitely accommodate multiple objects

was added to (2.28) in [88], the new term pulls back the contour in case it passes the object boundary. The new equation becomes

$$\phi_t = c(k + V_0)|\nabla\phi| + \nabla c.\nabla\phi \tag{2.30}$$

Although (2.30) has the new stopping terms, employing this equation does not prevent GAC models suffering the edge leakage problem. The direct reason behind the edge leakage problem is that, except for edge gradient, no local edge information is considered in most conventional GAC models. A new framework of GAC that uses local features of the image was proposed in [89].

Despite the fact that GAC models overcome some of the limitations of classical parametric models such as curve splitting/merging and parameter independence (see Figure 2.17), GAC models have their own limitations such as noise sensitivity and computation complexity. Defining the active contour model in the context of level set method makes the models sensitive to noise since the level set method generates local minima of the energy function. Also, the level set method is known to by computationally expensive.

Addressing the noise sensitivity and the computation complexity problems, graph-cut based GAC models were developed in [90] [91]. Graph-cut based GAC methods use the graph cut method to evolve the active contour instead of the level set method. The graph cut method is known to produce global minima and is less computationally expensive than the level set method [91]. Multi-resolution GAC [92] is another way to tackle the aforementioned problems of traditional GAC models. The simple idea behind the approach

is to use the image at lower resolutions in order to get a rough estimate of object(s) boundaries and then use those estimate to better localize the final active contour at the highest resolution of the image.

Chapter 3

Multi-resolution Undecimated Active Contours

3.1 Noise Sensitivity of Active Contour Models

Sensitivity of both active contour models (parametric and geometric) to noise is considered one of their main limitations. Many solutions to noise sensitivity have been proposed in the literature [94] [95]. As mentioned in Section 2.5.1, the balloon model, the GVF, the GGVF, the TVF, the VFC and many others are all aiming to address the noise sensitivity problem. The aforementioned methods still poorly behave under high levels of noise. Figure 3.1 illustrates the behavior of the GVF method [7] and the Active Contour Without Edges [87] under different noise levels. The GVF belongs to the PAC model and the Active Contour Without Edges belongs to the GAC model. Under a high PSNR value (50 in the figure), both methods locate the object of interest easily with the GAC method having the privilege of being able to split to locate the fine details inside the object (e.g. choroid plexus). At a lower PSNR value of 20, the split property of the GAC method turned out to be a disadvantage. The GAC method –being highly affected by the noise– identified too many objects that almost all of them are not true. Both methods misconverge, with the PAC method showing some resistance to the noise. At a PSNR value of 10, both methods give final segmentation results that are far away from the correct ones.

Alternatively, the multiresolution processing of the active contour is considered as the current state-of-the-art in approach for addressing the noise sensitivity issue of the active

Some parts of this chapter are excerpted verbatim from [93], where the thesis author is one of the co-authors.

contours [96] [97]. The dominant features of the object of interest (e.g. edges and strong textures) are used to guide the active contour at low resolutions, where these features are most likely to survive. At fine resolutions, the active contour is permitted to take small movements in order to capture the fine details of the underlying image. Hierarchical methods for active contours are discussed in the following section.

3.2 Hierarchical Active Contours

Addressing the noise sensitivity of active contour models in the context of the multiresolution framework helps to overcome this problem and is considered as the state-of-the-art in the literature. The first multiresolution active contour model was proposed in [98] as a hierarchical version of the balloon model proposed in [79]. The basic idea was to initialize the active contour in a low resolution image to roughly locate object boundaries. Intuitively, at lower resolutions of the image, only strong features (e.g edges) are survivable while other spurious features (e.g noise) have less contribution. After locating the image boundaries at the lowest resolution, which most likely are the true edges, these boundaries are then used to initialize the active contour at a higher resolution image. The process continues until the active contour is initialized in the finest version of the image (original image). In the same paper [98], another multiresolution active contour model that uses the Fourier transform was proposed. The model defines the scale as the number of harmonics used to describe the curve. The elliptic Fourier representation of a closed curve is

$$\upsilon(\theta) = \begin{pmatrix} x(\theta) \\ y(\theta) \end{pmatrix} = \sum_{k=0}^{N} A_k \begin{pmatrix} cos(\theta k) \\ sin(\theta k) \end{pmatrix} \tag{3.1}$$

where A_k is a 2×2 matrix, N is the number of used harmonics and θ is the angular parametrization index. The algorithm uses an initialization curve υ_1 to find another curve $\upsilon_1^\star$ that could be described by one harmonic and minimizes the energy function of the model. Iteratively, the number of harmonics p increases from 2 to N and each time the previous curve $\upsilon_{p-1}^\star$ is used to find a finer curve $\upsilon_p^\star$ described by p harmonics. One limitation of the Fourier approach happens when the underlying object has a complex curvature, in such situation a large number of harmonics are needed to accurately locate the object.

The Multiscale Tensor Vector Field (MTVF) was introduced in [10] as a noise robust active contour model, where the VFC kernel is adaptively modified using the image tensor Γ that is defined as:

$$\Gamma_{x,y} = \begin{pmatrix} \sigma_{x,x} & \sigma_{x,y} \\ \sigma_{y,x} & \sigma_{y,y} \end{pmatrix} \tag{3.2}$$

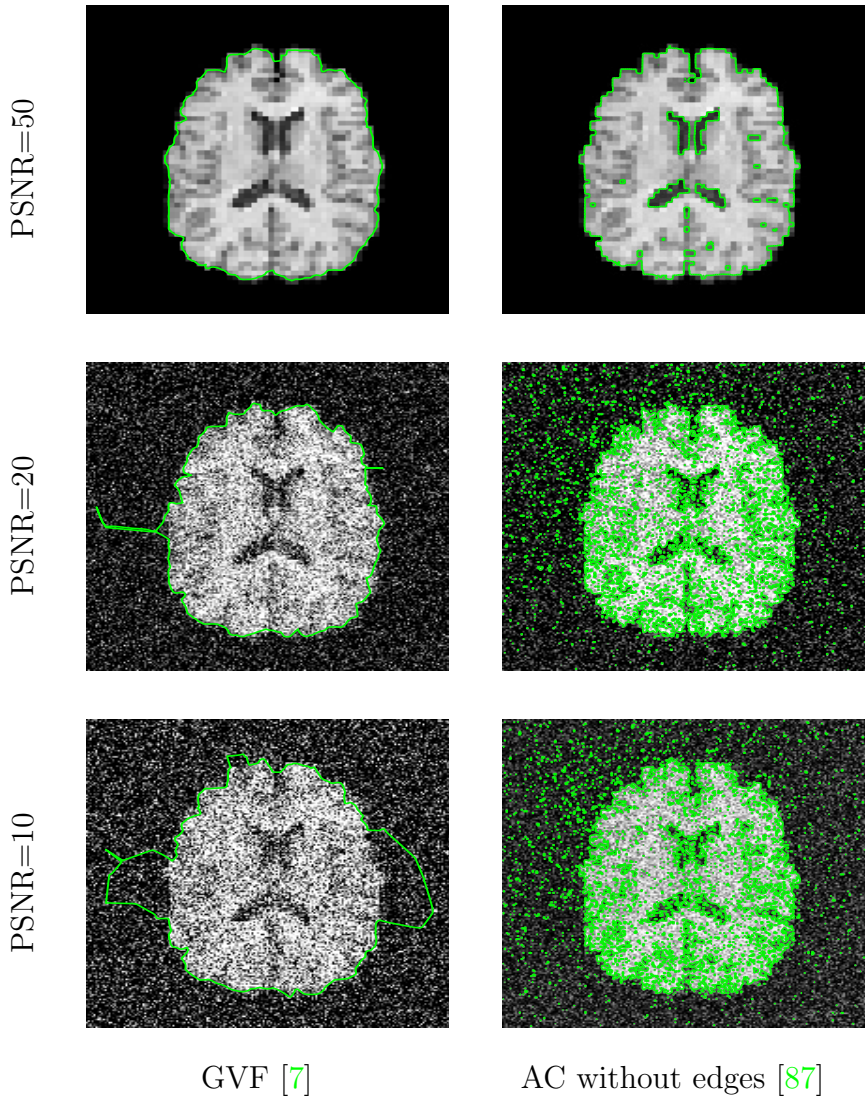


Figure 3.1: The segmentation results of the GVF [7] vs. The AC without Edges [87] active contour methods under different levels of noise. It is clear that both contours misconverge to incorrect object boundaries in noisy images

where $\sigma_{x,x}, \sigma_{y,y}$ are weighted variances and $\sigma_{x,y}$ is the weighted covariance of the image gradient. The coarsest image is used to initialize the first snake then finer resolutions make use of the snake position in their predecessor coarser resolutions in a manner similar to [98].

Due to the kernel size dependency of [8] that introduces a trade off between fine details preservation and large capture range, a Multiscale Vector Field Convolution (MVFC) [99] is introduced. The method convolves the image edge map with a bank of vector kernels as follows

$$F_v(x, y; t) = f(x, y) * k_t(x, y) \tag{3.3}$$

where $k_t(x, y) = m(x, y; t).n(x, y)$ (see 2.26 and 2.27) and $t \in [t_{min}, t_{max}]$. At each location of the image, F_v for all t values are computed and stored. For each pixel, the optimal scale t_{opt} is then determined as the scale at which the computed force is maximum:

$$t_{opt} = \arg \max_{t_{min} \leq t \leq t_{max}} |F_v(x, y; t)| \tag{3.4}$$

the final multiscale force field is created by combining the selected forces at each pixel into a single force field. However, MVFC has some limitations including the method dependency on the edge map, which is not usually available especially in noisy images. Moreover, the kernel size are arbitrary and do not depend on actual image features (e.g. edge strength), which in most cases requires a different bank for each image.

3.3 Undecimated Active Contours

Using the wavelet transform as a multiresolution framework to improve active contour models is extensively proposed in the literature [100] [101]. The zero-crossing of the wavelet coefficients is used in [101] to build a vector $ZCS_{i,j} = (ZCS_x, ZCS_y)$ whose components are the wavelet zero-crossing of x and y directions. The vector $ZCS_{j,j}$ represents the orientation and strength of a zero-crossing point. The vector field ZCS is then used to build a new image force that better drives the active contour. Statistical proprieties of the wavelet transform were analyzed in [100] using Principle Component Analysis (PCA) and used as priors during the contour deformation process. The global shape priors are obtained from the low resolution approximations of the transform, while the local shape priors are obtained from the detail images of the transform. One main limitation of the classical wavelet transform when used in image processing application is its down sampling property. At each resolution of the transform, the image is downsampled by 2, which, in

most cases, throws out important image information. Edge delocaliztion is one direct side effect of the downsampling step.

A key strength of undecimated wavelet transforms is their edge preservation property at coarse resolutions, made possible by their near shift-invariance and the absence of down-sampling [6] [102]. Figure 3.2 demonstrates the near-shift invariant property of the undecimated discrete wavelet transform as opposed to the traditional discrete wavelet transform. In the figure, no matter that both curves are just shifted versions of each other (shifted by 21), the first resolution detail coefficients of the traditional wavelet transform, due to the shift variant property, are unable to discriminate the two signals. While the first resolution detail coefficients of the non-decimated wavelet transform, due to the shift-invariant property, can easily be used to discriminate the two signals. The important properties of being shift-invariant and the absence of down sampling make the undecimated wavelet transforms a perfect choice for image segmentation applications. Using the undecimated wavelet transform, important image features such as edges and textures are guaranteed to have the same exact location, no matter which resolution of the transform is used for processing. Such phenomena (known as localization) makes the undecimated wavelet transform a perfect choice for multiresolution based active contours. Of course, being an over-complete transform, data redundancy and the need for more storage space can be considered as down sides of the undecimated wavelet transform. However, the great advances in the storage and processing capacities of the modern computing devices reduces the worries about time complexity issues behind the usage of the undecimated wavelet transforms.

As illustrated in Section 2.5.1, the movement of the active contour is controlled by two types of forces (internal and external). To build up the external force of the active contour, a force vector that measures the strength and the orientation of the image feature (e.g. edge) at each pixel is calculated. Collectively, the force vectors of the image pixels form the external force field of this image. The force field drives the active contour towards the object of interest. Since salient features of an image are most likely to survive at coarse resolutions of the UDWT, our proposed undecimated wavelet active contour (UWAC) model aims to build the external force map of the active contour using the gradient information from all resolutions of the transform, with preference given to coarser over finer resolutions. Since image noise is not expected to survive at coarse resolutions, favoring coarse resolutions over fine resolutions ensures that noise contribution in the final external force is suppressed. Moreover, this favoring ensures that the final external force is primarily influenced by the dominant features in the image, which hopefully, are the true edges of the object of interest. The new map accurately drives the active contour and improves edge localization. The proposed UWAC builds on both Gradient Vector Flow and Vector Field Convolution

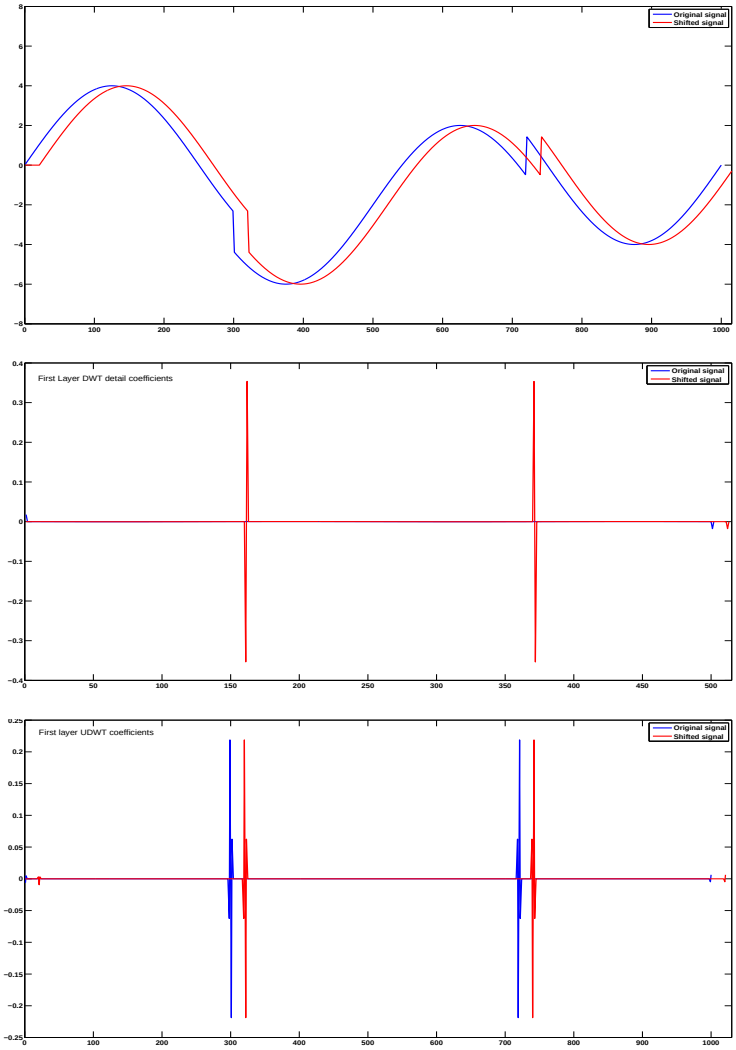


Figure 3.2: The shifted two signals in the first row are not distinguishable using the traditional DWT first resolution coefficients (second row), while the two signals are easily distinguishable using UDWT first resolution coefficients (third row) due to shift invariant property.

active contours. Comparisons to classical and state-of-the-art methods show a dramatic improvement in active contour convergence for all levels of noise.

Unlike other multi-resolution active contour algorithms [10] [11] that define a snake

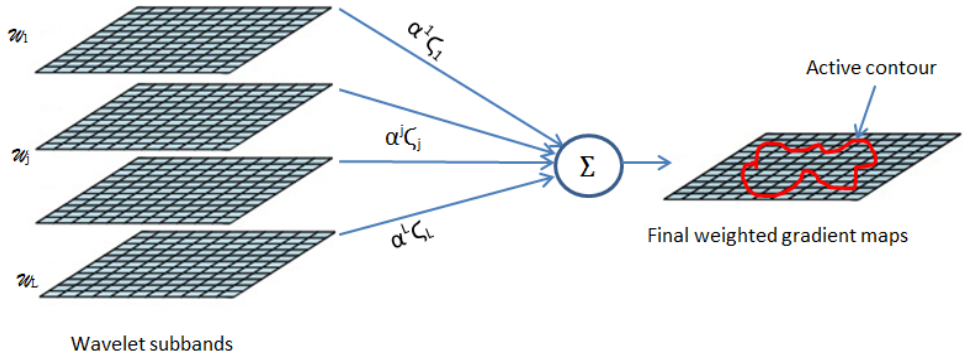


Figure 3.3: A schematic diagram of the proposed model (UWAC). The final gradient map is simultaneously built up by combining the resolution-based weighted wavelet gradients, $\alpha^j \varsigma_j$, of the UDWT. The external force field of the UWAC is generated using the final gradient map.

at each level of the hierarchy, with the previous/coarser resolution being the initialization of the snake in the current/finer resolution, the UWAC drives a single snake with an external force field based on the gradient information from all resolutions of the transform. Figure 3.3 shows a schematic diagram of the proposed model, where the weighted gradient information from all UDWT resolutions are combined to build up a final gradient map. This final gradient map is then used to generate the external force field of the active contour. The proposed model builds on both the Gradient Vector Field (GVF) [7] and the Vector Field Convolution (VFC) [8]. Experimental results show superior improvement of the UWAC, in terms of capture range and noise sensitivity, over the GVF and VFC methods.

3.4 Multi-resolution Gradient Maps

To address the problem of noise sensitivity of the active contour, our model uses the UDWT coefficients of all resolutions to build a weighted gradient map. The external force of the snake is then developed based on this map. Aiming to decrease noise participation in the developed map, gradients at coarser resolutions are weighted more highly than gradients at finer resolutions. However, the fact that UDWTs filters do not need to be orthogonal or bi-orthogonal [9] imposes no relation between the amplitude of the wavelet coefficients in different resolutions. Hence, the direct favoring of coarser resolution gradients without this

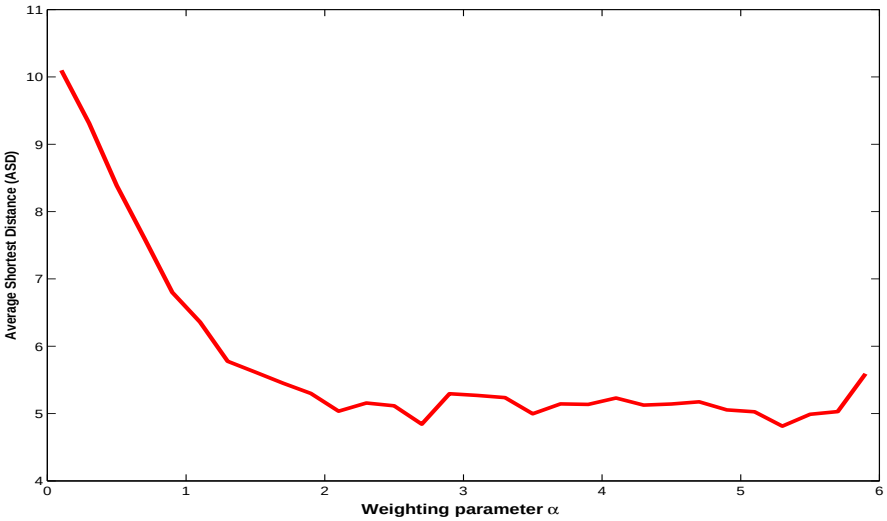


Figure 3.4: An average ASD [24] convergence using the test image set for different values of α in (3.7). Favoring fine resolutions ($\alpha < 1$) leads to worse ASD values. While favoring coarse resolution gives better ASD results.

relation between wavelet coefficients in different resolutions becomes of almost no sense. Therefore, as a first step, the UWAC normalizes the wavelet coefficients based on their local moment statistics.

Assuming a high correlation between the variances of the wavelet coefficients in a local neighborhood $\aleph(k)$, then for each $w(k) \in \aleph(k) \quad \sigma^2(k) \approx \sigma^2(\aleph)$. An estimate of $\sigma^2(k)$ for each coefficient $w(k)$ is computed using maximum likelihood (ML)

$$\begin{aligned} \hat{\sigma}^2(k) &= \arg\max_{\sigma^2 \geq 0} \Pi_{k \in \aleph} P(w(k)|\sigma^2) \\ &= \max\left(0, \frac{1}{K}\sum_{k \in \aleph} w^2(k) - \sigma_n^2\right) \end{aligned} \tag{3.5}$$

where σ_n^2 is the noise variance and K is the number of coefficients in $\aleph$. $\hat{\sigma}_n^2$, an estimate of the noise variance, is calculated by applying the robust median estimator on the coefficients of the finest resolution [103] [104]. To suppress the participation of noisy coefficients in forming the final force map, the wavelet coefficients are non-linearly normalized using an inverse logarithmic function $\mathcal{F}$ such that $\hat{w} = \mathcal{F}(w)$. $\mathcal{F}$ is designed such that noisy coefficients are mapped into a smaller range of the the new dynamic range.

At each resolution j, the normalized wavelet sub-bands of the x-details w_j^1 and the y-details w_j^2 are used to calculate the gradient map

$$\varsigma_j = \sqrt{(\hat{w}_j^1)^2 + (\hat{w}_j^2)^2} \tag{3.6}$$

The idea of non-linear and resolution based weighting of wavelet coefficients has been subject to researchers' investigation as a tool for noise reduction [105] [106]. The idea is to give high weights to low resolution bands of the wavelet transform, where salient features of the image are dominant. While high resolution bands, where noise contribution is expected to be high, are given lower weights. In our proposed UWAC, to further suppress the noise participation in the final external force map, an exponential weighting map is used to give resolution-based preference to coarse gradient maps over fine gradient maps as follows

$$\varsigma_t = \frac{1}{Z} \sum_{j=1}^{L} \alpha^j \varsigma_j \tag{3.7}$$

where Z is a normalization parameter. Figure 3.4 shows the empirical results of using different α values in (3.7). It can be seen from the figure that for α values less than one, which basically means giving higher weights to fine resolutions gradient maps, the resulted external force misleads the active contour model to incorrect segmentation boundaries. Which is expected and intuitive due to the high contribution of the noise in fine resolutions compared to coarse resolutions. On the other hand, Favoring coarser resolutions using α vales grater than 1 leads to more reliable final external force fields that correctly lead the active contour model to the object boundaries, and hence low segmentation metric (in this case, the Average Shortest Distance – ASD [24]) values are achieved. Again, these low ASD values are because of the higher contribution of image salient features (compared to noise contribution) in coarse resolutions. Section 3.5 gives more details about the ASD metric. In this chapter experiments, α is set to 2.

To illustrate the noise effect on the external force of the active contour, Figure 3.5 shows the external forces of the GVF and the VFC for three different images. No doubt the external forces of the first two images (Starfish and Leaf) will lead the active contour to a very poor segmentation result. While the external force of the Ushape image, even with the high contrast between the object and background, is still not consistent, especially in the concave part of the image.

Figure 3.6, on the other hand, shows the UWAC external forces of the same three images. Using only one resolution of the transform (second row), our model is able to produce a consistent gradient maps that correctly point to the object boundaries, which

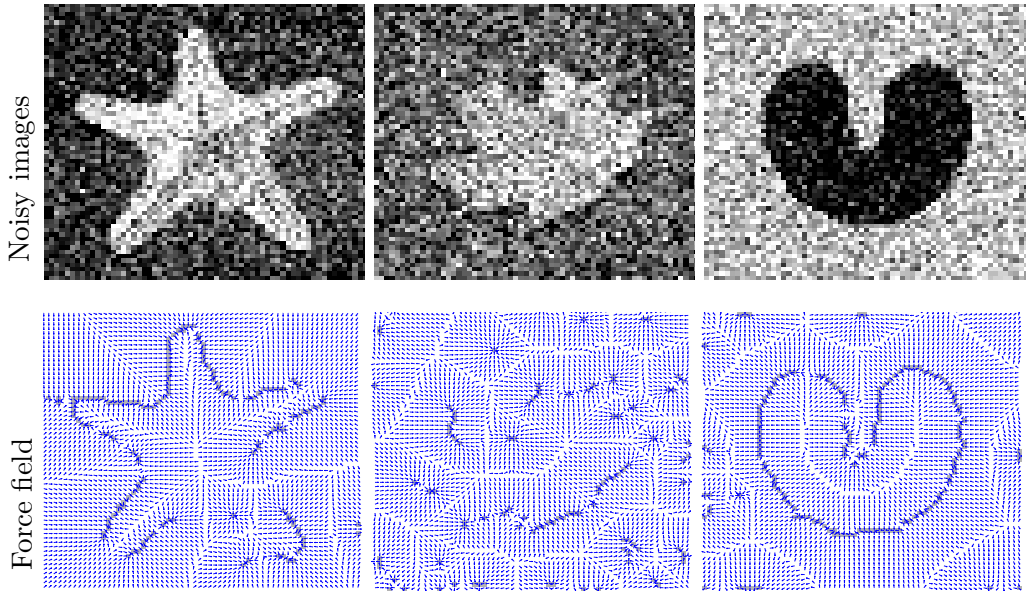


Figure 3.5: The external force fields of the VFC at PSNR=20. The noise causes the force field (blue arrows) to mistakenly point to spurious features even in the relatively high contrast U-shaped image, right. Relying only on the edge map of the image to generate the force field could be problematic, especially in case of noisy images.

in turn improves the resulted force fields compared to Figure 3.5. However, the influence of the noise on the 1-resolution gradient maps is still noticeable, especially in the Leaf and Starfish images, where the gradient maps misleadingly guide the active contour model towards non-real edges. The reason why the noise influence is still noticeable in these force fields is that using only one UDWT resolution is not enough to suppress noise contribution since there is not enough weighting of coarse resolution features in this case. As the number of the used resolutions increases and due to the property of edge preservation at coarser resolutions of the UDWT and the resolution-based weighting property (3.7) of our model, the consistency of the force maps increases. It can be seen from the figure that in case of using 3 resolutions the final maps of all images are, to a great extent, improved. For 5 resolutions, the noise effect on the final maps starts to disappear and the gradient maps are still pointing towards the actual edges. At 8 resolutions and due to the absence of UDWT down-sampling, the edges are still localized and the noise effect is almost not there. The final maps correctly point towards the edges of the objects of interest.

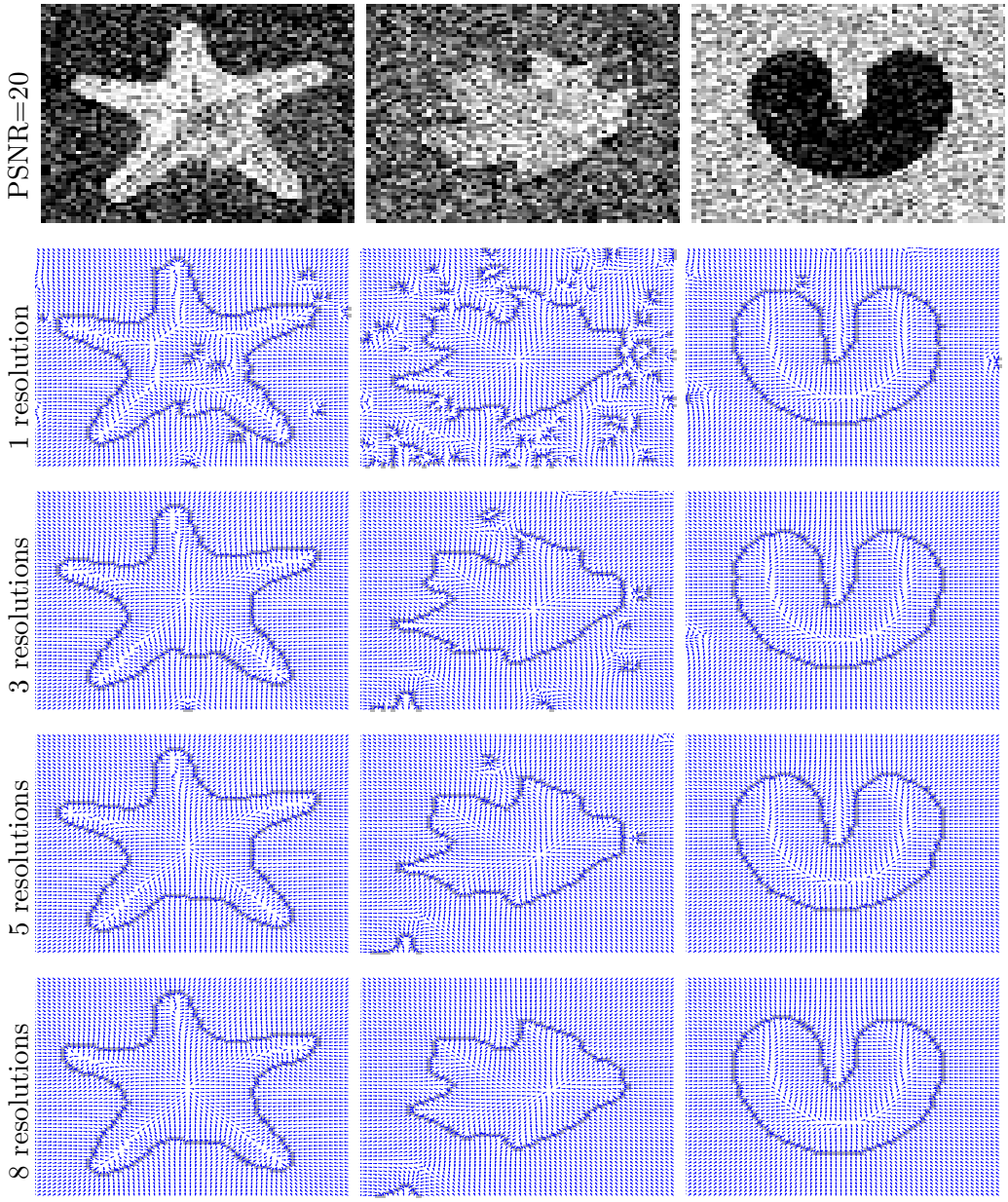


Figure 3.6: The external force fields of the UWAC model using one, three, five and eight resolutions of the UDWT at PSNR=20. The edge preservation property of the UDWT reduces the noise impact at coarse resolutions (compared to Figure 3.5). The consistency of the force field improves and the edges are not de-localized as the number of resolutions increases.

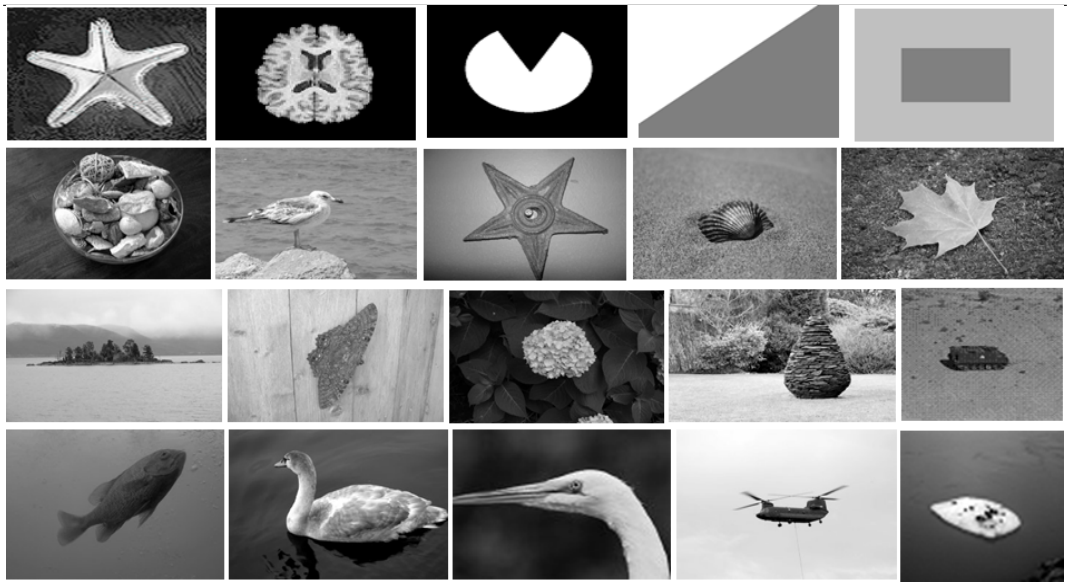


Figure 3.7: The image set used for testing. Images are selected to impose different challenges to the proposed model (e.g. weak edges, high curvature and high texture.)

3.5 Segmentation Results and Discussion

The UWAC model is tested using the Weizmann single object database [107]. A set of 20 images has been selected and used to demonstrate the effectiveness of the model. Figure 3.7 shows the images used for testing our proposed models. The first two images in the figure (namely the Starfish and the Brain images) are well known images in the active contours literature, while the remaining images in the first line of the figure are synthetic images created to illustrate different aspects our work. The remaining images of the figure are from the Weizmann database, and are selected such that they contain different types of challenges (e.g. high texture, curvature, weak and incomplete edges). Each image is contaminated with different amounts of zero-mean Gaussian noise. To ensure fair comparisons, the same noisy image is passed to all of the tested methods — UWAC, GVF, VFC, MVFC, and MTVF — with the active contour initialized at the same location for all methods. The average shortest distance (ASD) [24] is used to measure the dissimilarity between the converged contour $\underline{v}^c$ and the true (ground truth) contour $\underline{v}^t$, where the ASD is defined as the average shortest distance of the converged contour from

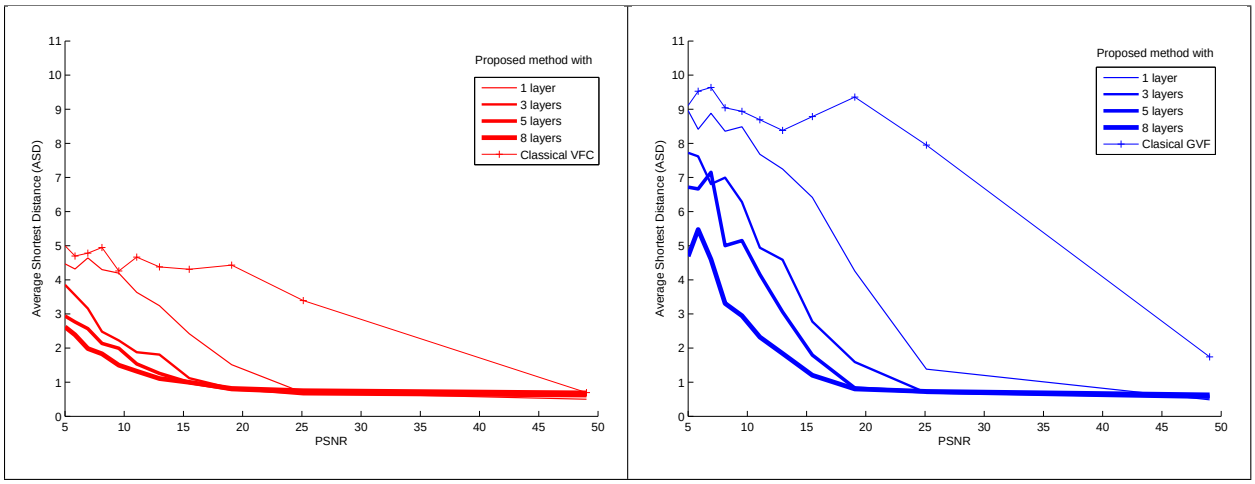


Figure 3.8: Our proposed UWAC model, building on GVF (blue) and VFC (red), compared to the classical GVF [7] and VFC [8] using one image (Leaf image). UWAC achieves better ASD values as the number of used resolutions increases. The other methods give poor segmentation results, even under relatively high PSNR values.

the true contour and the true contour from the converged contour:

$$ASD = \frac{1}{2q}\sqrt{\sum_{i=1}^{q} min_{j=1,2...n}(\|v^t(s_j) - v^c(s_i)\|)} + \frac{1}{2n}\sqrt{\sum_{j=1}^{n} min_{i=1,2...q}(\|v^t(s_j) - v^c(s_i)\|)} \tag{3.8}$$

where q and n are the numbers of points in the converged and true contours, respectively. Since ASD computes average shortest distance, q and n need not to be equal.

Figure 3.8 shows the performance of the UWAC, using different number of resolutions, compared to classical GVF and VFC. The figure shows, quantitatively, the segmentation results of the UWAC compared to other methods. A total of twenty runs of the algorithms using only one image (the Leaf image – textured with cluttered background) are used to generate this figure. For each run, the image is subjected to randomly induced Gaussian noise and passed to all methods. Also, the active contour is initialized at the same location for all methods. It can be seen from the figure that all methods give a good performance (with VFC better than GVF) at very high PSNR values (50), where there is almost no noise

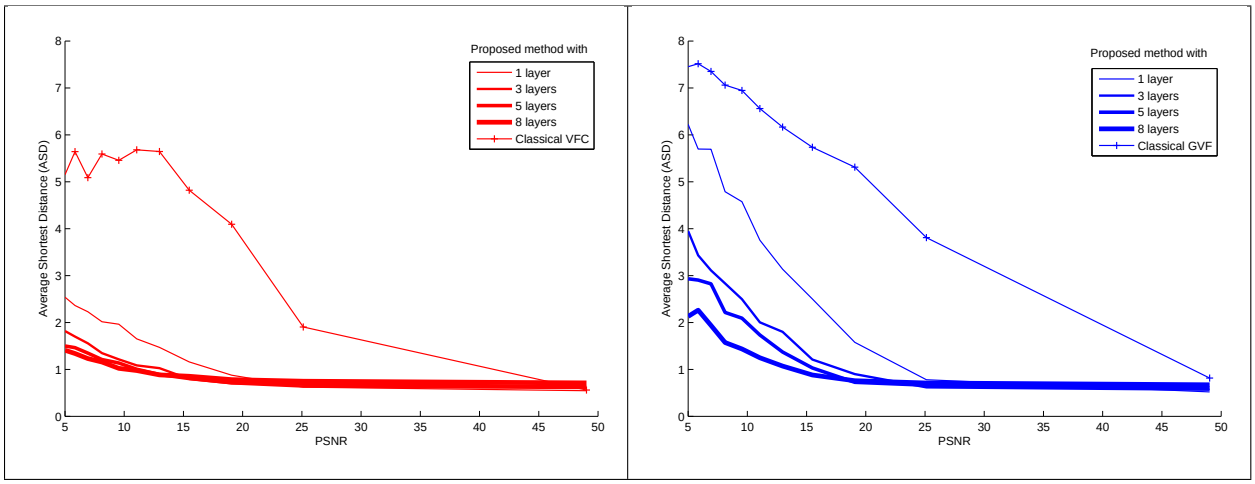


Figure 3.9: The average performance over 20 images of UWAC, building on GVF (blue) and VFC (red), compared to the classical GVF and VFC. Due to the resolution based weighting nature of the UWAC, the resulting segmentation metric outperforms the other methods under all levels of noise.

influence. As the noise contribution starts to increase, both GVF and VFC are misguided by the noise, and the final segmentation metric starts to increase. As the PSNR goes lower, the performance of the other methods gets worse. Our model on the other hand, gives better results compared to the other methods even using only one resolution of the UDWT. Thanks to the resolution-based weighting nature of the UWAC, as the number of used resolution increases, the final segmentation metric improves. The figure shows that the UWAC gives considerably consistent and accurate segmentation results even under high levels of noise.

To generalize the performance of the proposed model, Figures 3.9 shows the average performance of the UWAC, over the entire test set using different numbers of UDWT resolutions, compared to the classical GVF and VFC. Again, each image is contaminated with 60 different amounts of noise, and for each image a total of twenty runs are averaged. Even at high PSNR values, the cluttered backgrounds of the images mislead the active contour in the GVF and VFC, a problem which gets even worse at lower PSNR. UWAC, on the other hand, overcomes the noise sensitivity problem and outperforms its competitors at all levels of noise.

Figure 3.10 shows the performance of the UWAC compared to other state-of-the-art

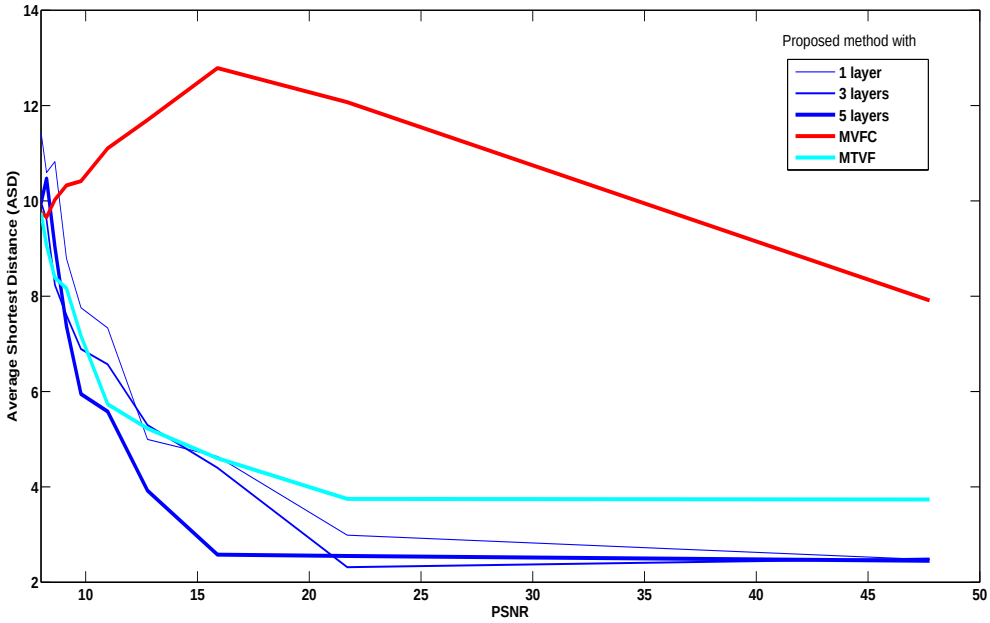


Figure 3.10: A comparison of the average performance of our proposed UWAC (building on VFC), MVFC [99], and MTVF [10] using 20 images. The proposed model outperforms both other multiscale methods under all levels of noise.

multiresolution methods (namely MVFC and MTVF). The weak behavior of the MVFC is due to two reasons. First, the parameters of the smoothing kernel set depend on the features of the underlying image and hence the method gives poor results in generalization tests. Second, the MVFC uses Canny edge map [53] instead of the gradient map to build up the external force field of the active contour. Canny edge detector is not a good tool to use for noisy images. The performance of the MTVF, on the other hand, can be related to the edge delocalization nature of the classical wavelet transform due to the down sampling step. The UWAC gives the best performance over its competitors for all levels of noise. The two figures, 3.9 and 3.10, show that our method outperforms both classical and multiresolution methods under all levels of noise and for all types of images. The results in this figure are based on the Matlab code provided by the authors of the competing methods.

To qualitatively illustrate the superiority of the UWAC model over other models, Figure 3.11 illustrates the ability of the UWAC to accommodate fine details and complex curvature under all levels of noise compared to GV, VFC, MVFC and MTVF. Eight resolutions of the UDWT transform were used to generate the external force field of the UWAC.

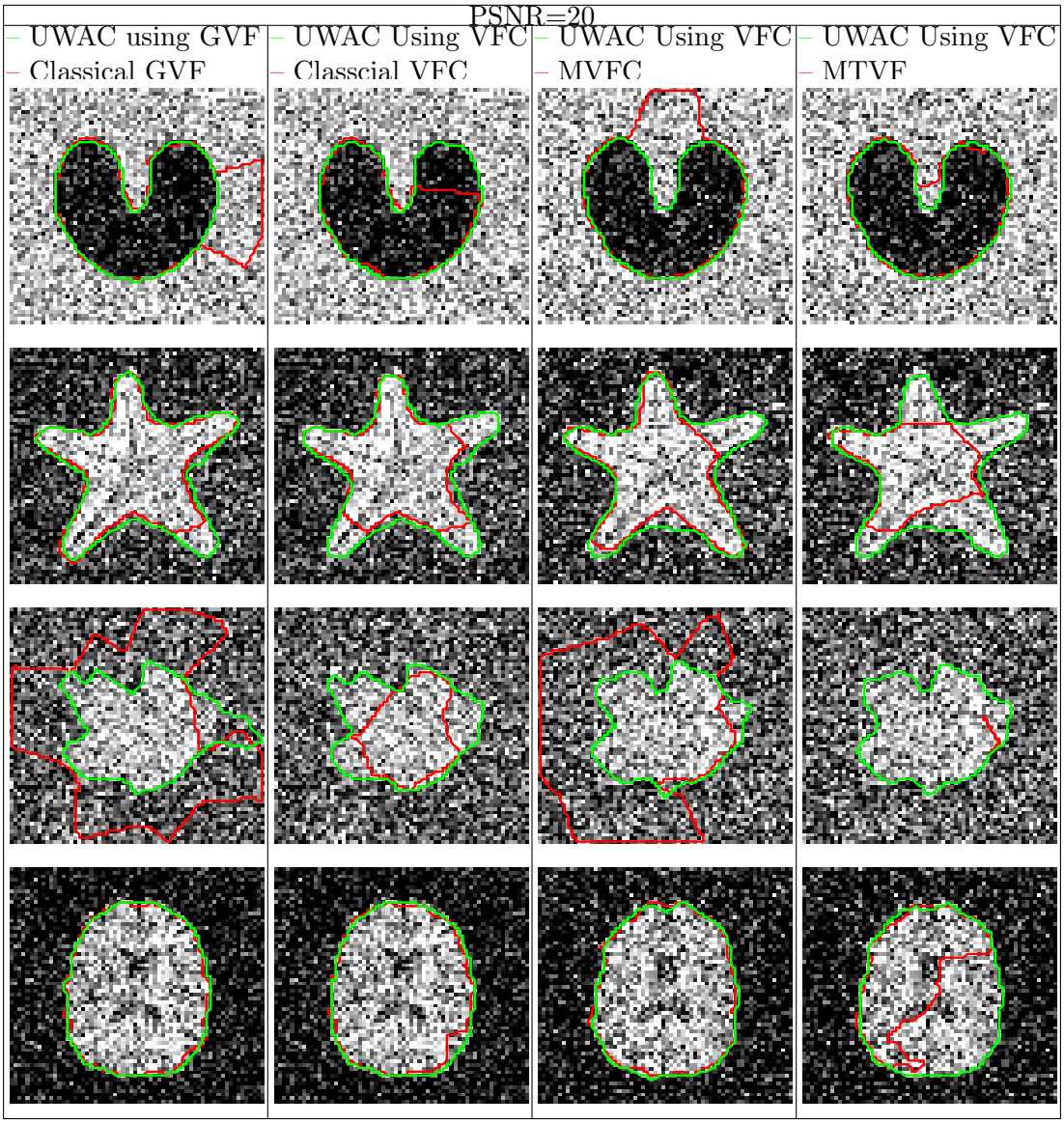


Figure 3.11: Comparisons between converged contours of the UWAC using GVF and VFC (green) against the classical GVF [7], VFC [8], MVFC [99] and MTVF [10] (red). Converged contours of the our model (using 8 resolutions) accommodate fine details and complex curvatures even under low PSNR. Other methods are not able to correctly segment the underlying objects

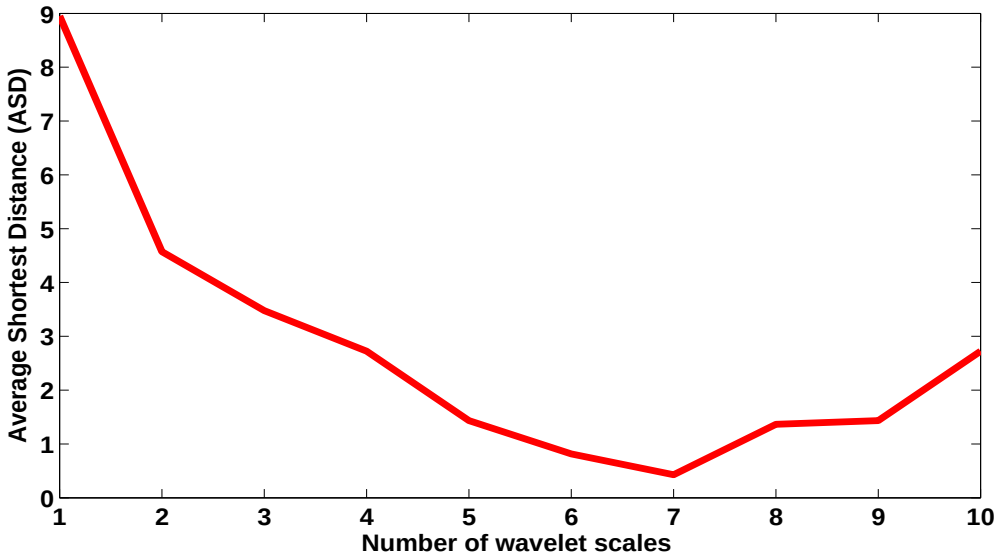


Figure 3.12: Segmentation accuracy versus number of wavelet scales. Though strong features (e.g edges) are most likely to survive at coarse scales, a significant increase in the number of scales negatively affects the accuracy of the final segmentation.

In the figure, even under high noise level (PSNR=20), UWAC is able to correctly force the active contour properly into the concave parts of the U-shaped and the Starfish images, while GVF, VFC, MVFC and MTVF contours failed. Fine details in both the Leaf and the Brain images were captured by the proposed model, for which the other methods failed.

The last question to address in this section is about the number of wavelet resolutions that should be used to generate the gradient map. As mentioned before, salient features are most likely to survive at coarse scales, and fine features contributes in fine scales. So, does this mean that we can use as many scales as we want? The answer of this question is revealed in Figure 3.12. Though salient features of the image (e.g. strong edges) survive and are well localized at coarse scales of the undecimated wavelet transform, as the number of scales significantly increases these features start to get weaker and weaker. The figure shows that, at least for the images used in this study, using 5-9 scales of the undecimated wavelet resolutions give the best ASD values. For number of scales less than 5, noise contribution prevents the active contour from accurately locating the object of interest. Also, for number of scales greater than 9, the salient features contribution in the final force map starts to get weaker resulting in higher ASD values.

In summary, this chapter proposed a multi-resolution active contour model that builds on the GVF and the VFC, and uses the undecimated wavelet transform to simultaneously build the external force map from all resolutions of the transform. Our model favors coarse scale features over fine scale features based on the fact that salient image features are most likely to survive at coarse scales. Experimental results showed the superiority of our model over the sate-of-the-art and classical models. However, in noisy images, favoring features of coarse scales might not be enough for good segmentation results. Hence, in the following chapter, we introduce our second multi-resolution active contour model which incorporates the directional information revealed by the dual tree complex wavelet transform for better guiding the active contour towards the correct object boundaries.

Chapter 4

Multi-resolution Directional Active Contours

A common limitation of active contour models is their poor convergence in case of complex object shapes and noisy backgrounds. Though Chapter 3 proposed a model that addresses the noise sensitivity problem through favoring coarse scale features of the undecimated wavelet transform, when object complexity increase, the favored gradient maps will not be able to correctly drive the active contour to the correct segmentation, especially in complex areas of the image (e.g. intersecting edges). For such circumstances, many solutions have been proposed in the literature, with the current state-of-the-art based on incorporating additional information into the active contour model such as texture cues [108] and gradient directions [109]. However, these solutions become not suitable when the complexity of the object is combined with a high noise level. The high noise level makes both texture information and gradient directions less reliable and misleading to the active contour. Hence, in this chapter, we aim to incorporate a more reliable and rich directional information source into the active contour model so that it can be correctly guided to the correct boundaries of noisy and complex shape objects.

A key strength of the dual tree complex wavelet transform (DT $\mathbb{C}$WT) [12] is its rich directional selectivity compared to other transforms. Moreover, the near shift invariant property of the transform preserves image feature from being delocalized or distorted at coarse resolutions of the transform, which makes the DT $\mathbb{C}$WT an attractive candidate for image segmentation problems. Therefore, the work in this thesis is extended to incorporate the multi-resolution directional information of the dual tree complex wavelet transform into the external force of the active contour in order to achieve better segmentation results for noisy backgrounds and complex object shapes. Unlike other directional active contour

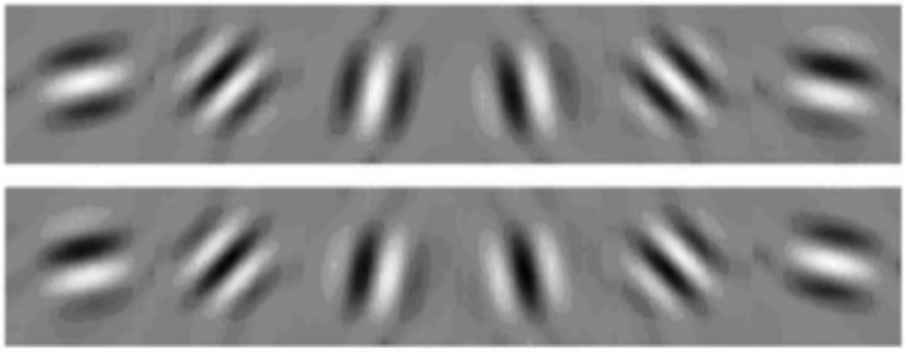


Figure 4.1: The real (top row) and imaginary parts of the DT ℂWT. The six directionally selective filters of the DT ℂWT (compared to 3 filters in case of real wavelets) makes orientation analysis more accurate using the transform.

models that add an extra directional term to the snake energy, our method implicitly incorporates the orientation information into the external force term of the energy function. The proposed method builds on the Vector Field Convolution active contour and the results show a dramatic improvement in active contour convergence for both real and synthetic images with different objects and backgrounds complexity. In order to make this chapter a self-contained one, the following section gives a brief introduction to the DT ℂWT, then the remaining of the chapter will be describing our second proposed model.

4.1 The Dual Tree Complex Wavelet Transform

Since the introduction of the dual tree complex wavelet transform in 1998 by Kingsbury [12], it has been successfully used in a wide range of image processing and computer vision applications including segmentation [110], compression [111], classification [112], watermarking [113] and object tracking [114]. Two different paralleling wavelet filter banks compose the DT ℂWT, with the two filters orthogonal, approximate half sample delay and linear phase. The DT ℂWT is nearly shift invariant and has good directional selectivity of six directions for each subband as opposed to only three directions in the classical discrete wavelet transform. The DT ℂWT decomposes a 2-D image $I(x, y)$ using a complex scaling function $\phi(x, y)$ and a complex wavelet function $\psi(x, y)$ oriented at different six

orientations, namely $\{15°, 45°, 75°105°135°165°\}$, as follows

$$I(x,y) = \sum_{l\in\mathbb{Z}^2} s_{j_0,l}\phi_{j_0,l}(x,y) + \sum_{\theta\in\Theta}\sum_{j\geq j_0}\sum_{l\in\mathbb{Z}^2} c^{\theta}_{j,l}\psi^{\theta}_{j,l}(x,y). \tag{4.1}$$

where $\mathbb{Z}$ is the set of integer numbers, j and l are the indices of shifts and dilations respectively, $s_{j_0,l}$ are the scaling coefficients, $c_{j,l}$ are the wavelet coefficients and Θ is the six directions set.

The inherent features of being nearly shift invariant and having good directional selectively make the DT $\mathbb{C}$WT a very attractive choice for image segmentation applications. Additionally, the perfect construction using short linear phase filters, the limited redundancy ($2^m : 1$ for $m-$D data) and the efficient order-N computation of the DT $\mathbb{C}$WT make it a good choice for real time applications. Hence, in this study, the directional information provided by the DT $\mathbb{C}$WT is incorporated into the active contour model in order to build more reliable external force fields with the aim to give better directions to the active contour in noisy images and complex shape objects. The experimental results in this chapter show a dramatic improvement in the final convergence accuracy of the proposed active contour model over classical and state-of-the-art methods.

4.2 Multi-resolution and Directionality

This section analyzes the directional selectivity property of the DT $\mathbb{C}$WT, and how directional information at different scales of the transform can be combined, while the next section is describing the second proposed model, namely the multi-resolution directional vector field convolution (MDVFC) model, that incorporates the directional information revealed by the DT $\mathbb{C}$WT into the active contour model.

To demonstrate the direction selectivity property of the DT $\mathbb{C}$WT and the benefits of combing the information from different resolutions of the transform, let $I_{l,\theta}$ be the wavelet coefficients at resolution $l \in L$ and direction $\theta \in \Theta$ of the original image I. And let T be a transformation function that takes a set of coefficient images $I_{l,\theta}$ and returns $T(I)$. Choosing T to be simply the summation over resolutions of the maximum responses of wavelet subbands, we get:

$$T(I) = \sum_{l\in L} \max_{\theta\in\Theta} |I_{l,\theta}| \tag{4.2}$$

Figure 4.2 shows the $T(I)$ for each individual orientation $\theta = \{15, 45, 75, 105, 135, 165\}$ over three resolutions of the transform. It can be seen that edges along the same orientation

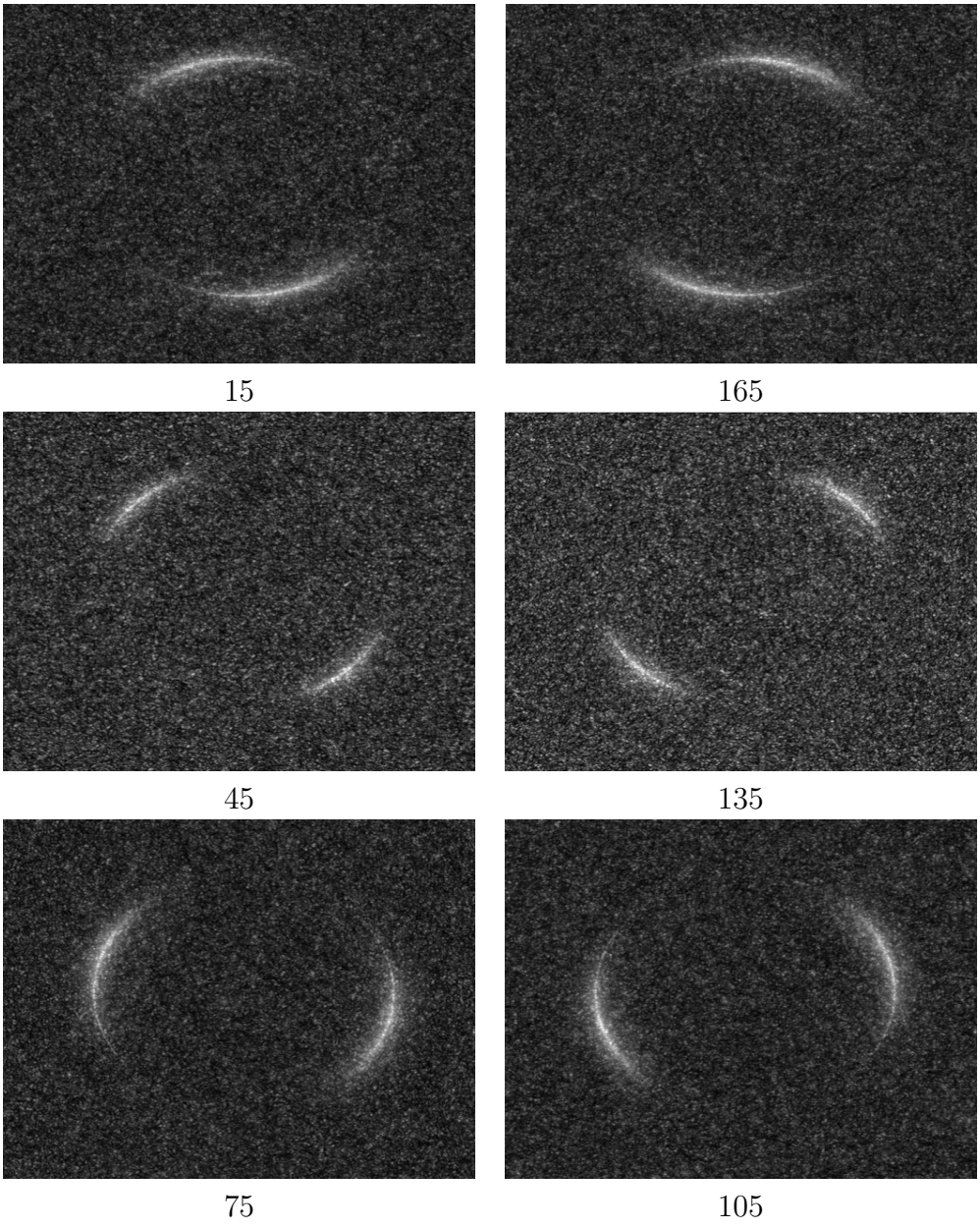


Figure 4.2: Summation over individual orientations of DT ℂWT three resolutions. Features along the same direction of the underlying wavelet band have noticeably higher responses than other image features.

of the DT ℂWT wavelet have noticeably higher responses than other features. By simply summing the wavelet coefficients of maximum response orientations at all resolutions we get Figure 4.3. The figures shows that after applying T in (4.2), the disc boundary in the resulted image $T(I)$, even under a high noise level, still has a noticeably higher response than other features, which make it easily distinguishable and traceable by the active contour.

At coarse resolutions of the DT ℂWT, dominant features of the underlying image are most likely to survive. Consequently and due the lower contribution of noise, the orientation information generated by the DT ℂWT is more accurate and consistent at coarse resolutions. As we move towards fine resolutions, noise contribution becomes stronger and the orientation information revealed by DT ℂWT becomes less accurate. Hence, coarse scales directional information of the DT ℂWT can be used, with high confidence, in connection with other information (e.g. edge data) to locate the dominant features of the underlying object of interest. On the other hand, DT ℂWT fine scales can be used, with caution, to fine tune the detection through incorporating the directional information of the object fine structures.

4.3 Multi-resolution Directional Vector Field Convolution

In this section, we propose the second model for active contour external force generation. The proposed Multi-resolution Directional Vector Field Convolution (MDVFC) model, utilizes the directional information revealed by the DT ℂWT to guide the active contour in noisy and complex shape images. Starting from the coarsest resolution, the model incorporates directional information into the external force field using a resolution-dependent convolution kernel $K_{l,\theta}$. At coarse resolutions, where reliable orientation information and wavelet responses exist, large size kernels are used. At finer resolutions, where orientation information and wavelet responses are less reliable, small size kernels are used. Moreover, the profile of the kernel changes according to the local orientation statistics of the image. At each resolution l, a set of six kernels $K_{l,\theta}$ steered at the same orientation of the DT ℂWT is generated, where $\theta \in \Theta$. Compared to the classical VFC kernel, the MDVFC kernel has the following unique attributes:

1. Size: the size of the kernel $K_{l,\theta}$ adaptively changes based on the DT ℂWT resolution. At coarser resolutions of the transform, where salient features of the image dominate,

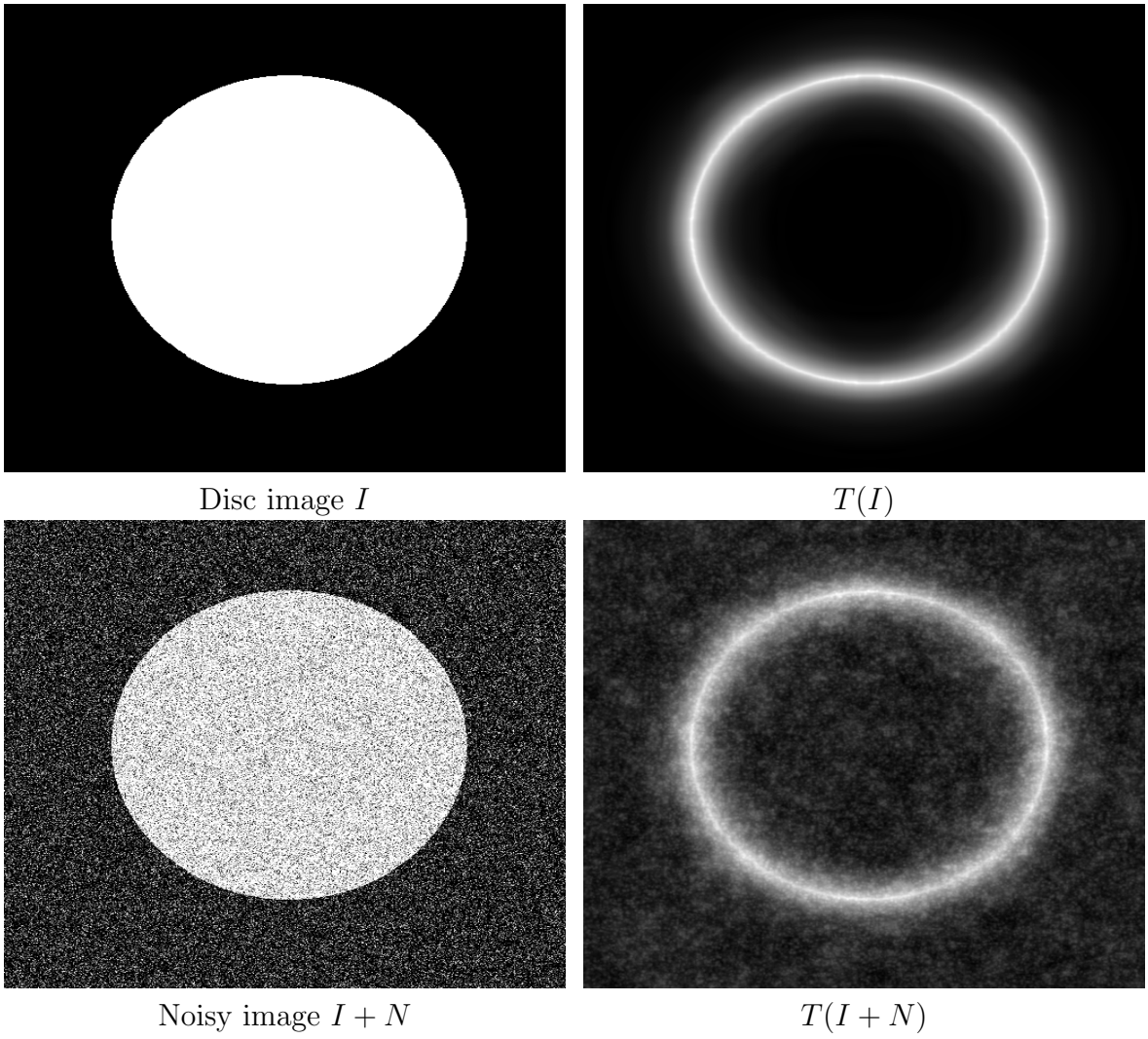


Figure 4.3: Applying T in (4.2) for the disc image. By simply summing the wavelet coefficients of the maximum responses over the three resolutions, the real object boundary has noticeably higher responses than noise pixels, which makes $T(I)$ a good image to build a more robust external force field for active contour segmentation.

large size kernels are used to allow the iterative solver to take larger steps towards the final solution. At finer resolutions, where spurious features contribution increases, small size kernels are used to accommodate fine details of the underlying object. Hence, the kernel size is positively correlated to the resolution l as moving from the

finest resolution $l = 1$ to the coarsest resolutions $l = L$ as follows:

$$\begin{aligned} K_{l,\theta} = &\{K_l(x,y); R = e^l; 1 \leq l \leq L; \\ &x, y = -R, \ldots, -1, 0, 1, \ldots, R\} \end{aligned} \tag{4.3}$$

2. Directionality: at each pixel, the directional information revealed by the DT $\mathbb{C}$WT is used to steer the kernel to a direction that best drives the active contour to the nearby edges. Hence, $K_{l,\theta}$ can adapt its profile to local structures of the image, which increases the robustness of the active contour model and decreases its initialization sensitivity.

3. Neighborhood: as the directionality can not be attributed to noise, considering the neighborhood directionality adds more robustness to the proposed method. For each pixel, a dominant direction is calculated and used to select the best kernel, $K_{l,\theta}$, to calculate the force vector at this pixel. The calculation of the dominant direction at each pixel depends on the local directional statistics of the pixel neighborhood. Again, because directionality can not be attributed to noise, considering the neighborhood directional information makes the proposed method less sensitive to noise.

At each resolution l of the DT $\mathbb{C}$WT, two images are generated, 1) a direction image $\vartheta_l^\star$, defined as follows

$$\vartheta_l^\star = \underset{\theta}{\operatorname{argmax}} \, |I_{l,\theta}| \tag{4.4}$$

where the value of each pixel in $\vartheta_l^\star$ equals to the the orientation of the wavelet subband that has the maximum response in l. 2) a maximum response image $I_{l,\vartheta^\star}$, defined as

$$I_{l,\vartheta^\star} = \max_{\theta \in \Theta} |I_{l,\theta}| \tag{4.5}$$

Hence, $\vartheta_l^\star$ describes the best orientation of each pixel in the image at resolution l, and $I_{l,\vartheta^\star}$ records the wavelet responses of the orientations in $\vartheta_l^\star$. Also, at each resolution l, a set of six oriented convolution kernels $K_{l,\theta}$ are generated to be used for external force field calculations. Unlike other methods (e.g. VFC and MVFC) where the convolution kernels do not reveal anything about directional information of the underlying pixel, the proposed kernels are steered based on the DT $\mathbb{C}$WT information to incorporate the orientation information of each pixel.

The proposed method works as follows: for each pixel in the response image $I_{l,\vartheta^\star}(x,y)$, a single convolution kernel is selected for force vector calculation. To calculate the force

vector associated with each pixel in $I_{l,\vartheta^\star}$, the directional information of $\vartheta_l^\star$ is used to select the best convolution kernel amongst the six kernels in l. The kernel selection depends on the orientation information of the underlying pixel and its neighborhood. One main characteristic of the convolution kernels is that their sizes are resolution dependent. Big kernels are used at coarse resolutions of the DT $\mathbb{C}$WT, and smaller kernels are used as we move towards fine resolutions. This dynamic nature (in both size and orientation) of the used convolution kernels dramatically decreases the noise sensitivity of the active contour, and increases convergence accuracy. The proposed model works scale by scale, meaning that the active contour is initialized at the coarsest resolution and then moves to the next finer layer with the convergence coordinates of the previous layer taken as the initialization coordinates in the current layer. The process continues until the convolution kernel set with the smallest size is used to build up the force field of the finest resolution. Moreover, unlike other directional active contour methods [89] [108] that add an extra directional term to the snake energy model proposed in (2.22), so that the model energy function becomes:

$$E = \int_0^1 \frac{1}{2}[\alpha(s)|\upsilon^{'}(s)|^2 + \beta(s)|\upsilon^{''}(s)|^2] + E_{external}(\upsilon(s) + E_{directional}(\upsilon(s))ds \tag{4.6}$$

our method keeps the model in (2.22) unchanged. Alternatively, the directional information is implicitly embedded into the $E_{external}(\upsilon(s))$ term of the energy function through convolving each pixel with the kernel that best describes the directional information of that pixel and its neighborhood. Keeping the model unchanged allows our method to build on the already exist active contour models such as the VFC, also it keeps the model complexity and computational time low.

Figure 4.4 illustrates the differences between the classical VFC kernel and the proposed kernels. The figure gives only three examples out of the six kernels $K_{l,\theta}$. Using the classical VFC kernel, all pixels at the same distance from the origin of the kernel will have equal influence on the force vector at the origin. Hence, if two pixels P_e (edge pixel) and P_n (noise pixel) are at the same distance from the kernel center P_c and both points have the same intensity, then no matter a free particle is placed over the true edge pixel P_e or the noise pixel P_n, the particle will be affected by the same force. Which means that both points, P_e and P_n, will have the same effect on the final force vector formed at P_c.

On the other hand, using the proposed kernels and since P_e will most likely have the same orientation as P_c, the corresponding kernel $K_{l,\theta}$ will give P_e higher weight than P_n when forming the force vector at P_c. Now, using the proposed kernels, a free particle placed over P_e will be strongly pulled towards P_c more than if the particle were to be placed over P_n. Hence, noise and spurious features of the image will have less effect on the final external force field than real edges. By doing this, the proposed method adds another

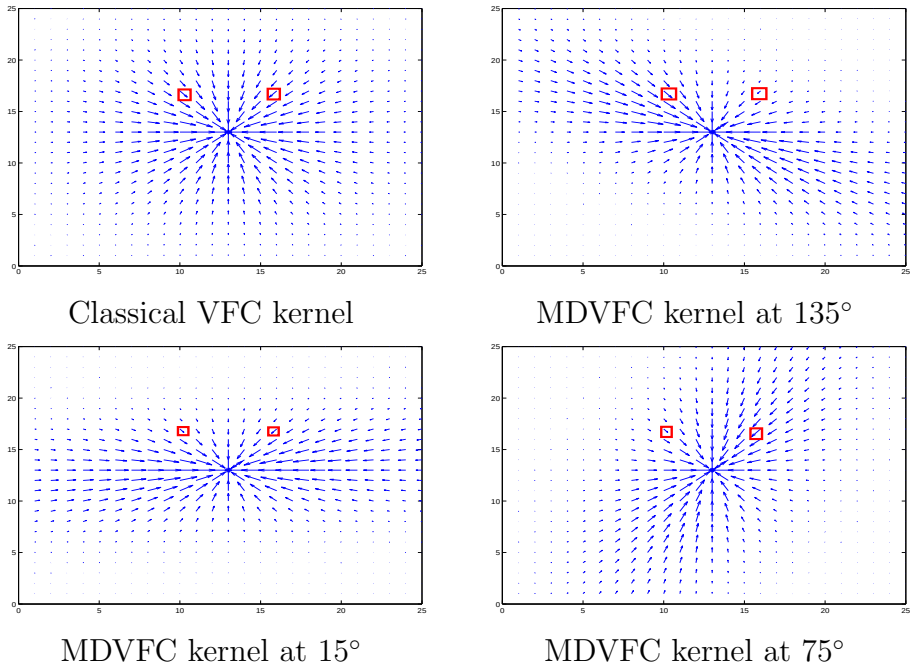
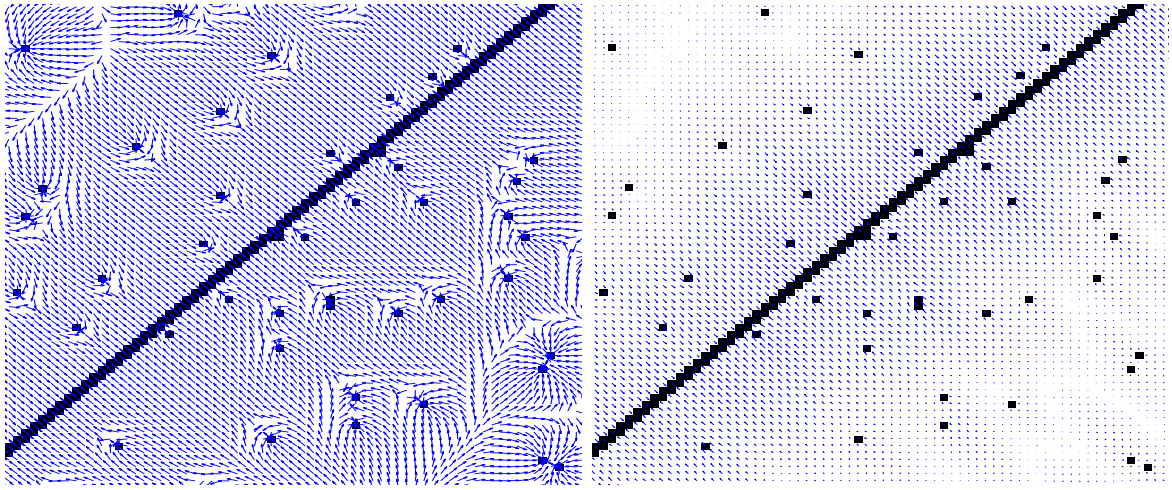


Figure 4.4: The classical VFC kernel versus the proposed kernels $K_{l,\theta}$. Using the classical VFC kernel, all pixels at the same distance from the center of the kernel (e.g. two red rectangles) are affected be equal amount of force, and hence contribute equally in the final force vector. While using the proposed kernels, features along the same direction of the used kernel are affected by stronger forces, and hence contribute more in forming the final force vector.

level of robustness to the active contour model. Figure 4.5 illustrates the idea of giving less weights to noise pixels compared to real edge pixels. In the figure, since the VFC kernel gives equal weights to all points at the same distance from the kernel center, the resulted force filed is highly affected by the noise pixels. The force vectors formed at the noise pixels, even those that are relatively close to the real edge, are pointing towards incorrect directions. The force field of our method, on the other hand, correctly points towards the diagonal edge. The kernel steerability property of the proposed method gives higher weights to the pixels along the same direction of the used kernel (in this case $K_{1,135^\circ}$), and less weight to other pixels. Hence, the resulted force vectors correctly point towards the real edges and the noise effect is minimal.



VFC external force MDVFC external force

Figure 4.5: External force fields of VFC and our MDVFC method. Noise pixels affect the resulted force vectors, and hence the vectors are not consistently pointing to the real edge. The force field of our method, due to kernel steerability, gives less weights to noise pixel, and hence the resulted force vectors correctly point to the real edge.

4.4 Local Statistics and Dominant Orientation Calculation

To add another level of robustness to our proposed method, the local orientation statistics is taken into account when building the force vector at the underlying pixel P_c. The process of selecting which kernel to be used for the external force calculation does not depend only on the orientation of P_c, it also depends on the orientation of the pixels in P_c neighborhood. The dominant direction of P_c and its neighbors is calculated and used to select the appropriate kernel $K_{l,\theta}$. Taking the local directional information into account makes the final force field less sensitive to noise. For example, during the convolution of the kernel with $I_{l,\vartheta^\star}(x, y)$, if the origin of the kernel P_c is placed over a noise pixel P_n, then based on the dominant orientation of the neighborhood, a kernel that has the correct orientation will be selected and the effect of having a noisy pixel inside the kernel will be eliminated. Figure 4.5 shows that incorporating the neighborhood orientation information in the process of selecting which kernel to be used for the external force calculation produces more robust force fields compared to the classical VFC kernel

It is important to make it clear that the calculation of the dominant orientation is carried out using the directional information stored in $\vartheta_l^\star$. Hence, $\vartheta_l^\star$ is used primarily to select the most appropriate convolution kernel. But, to calculate the actual value of the force vector at a certain pixel, the selected kernel is convolved with $I_{l,\vartheta^\star}(x, y)$.

Assuming a high correlation between pixel orientations in a local neighborhood $\aleph$ centered at the pixel $\vartheta_l^\star(x, y)$. When convolving $K_{l,\theta}$ with the image, three scenarios could happen

1. there is no dominant edge information in the area of $I_{l,\vartheta^\star}(x, y)$ covered by the kernel (smooth area or just noise).
2. there is only one dominant edge information in the area covered by the kernel
3. the kernel covers more than one edge (i.e. two or more dominate edges).

In the experiments of this work, edge information inside the kernel is defined as dominate if the number of pixels having the same orientation hits a threshold D, which is empirically set to 20% of the kernel size. Intuitively, in the first scenario, the dominate orientation of the neighborhood will be the same as $\vartheta_l^\star(x, y)$. In the second scenario, the dominate orientation will be equal to the direction of the covered edge, and kernel that has a perpendicular orientation to the dominate orientation will be used to calculate the force vector. In the last scenario, since the neighborhood size is relatively small compared to the image size, we assume that there will be at most two true edges in $\aleph$. Hence, the two orientations that have the highest occurrences in $\aleph$ will be considered during the kernel selection process. In this case, the dominate orientation $\bar{\theta}$ of the neighborhood is calculated as follows

$$\bar{\theta} = Q\Big(\theta_i + \Big(\frac{\theta_j - \theta_i}{M + N}\Big) \times N\Big) \tag{4.7}$$

where θ_i and θ_j are the orientations of the two edges, with M and N counts respectively and $M \geq N$. Q is a quantization function that maps its parameter to the nearest orientation in Θ. The dominate orientation $\bar{\theta}$ is then used to select the best kernel $K_{l,\theta}$. Since there are two edges competing to label the center point as belonging to one of them, a kernel $K_{l,\theta}$ that has the closest direction (not perpendicular, as in the second scenario) to $\bar{\theta}$ will be used for the force vector generation.

Figure 4.6 illustrates two different synthetic examples of using the dominate orientation to select the best kernel in the cases where there is one and two edges in the neighborhood. In the first row of the figure, since there is only one main edge in the neighborhood $\aleph$, $K_{l,\theta}$

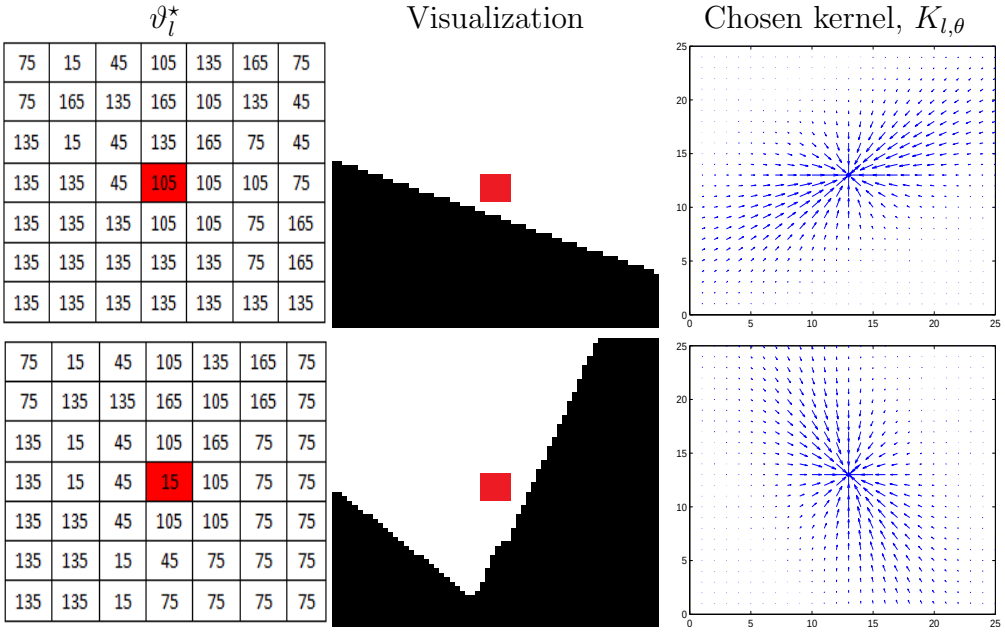


Figure 4.6: Synthetic examples of dominant orientation calculation and kernel selection (for visualization: all pixels having dominant orientation are drawn in black and other pixels in white). In the top row, since there is only one edge, the kernel $K_{l,\theta}$ is chosen to be perpendicular to kernel center (red) orienation. In the bottom row, the kernel orientation is chosen to be the closest to $\bar{\theta}$, in this case $K_{l,\theta} = Q(98.57°) = 105$ according to (4.7)

is chosen so that it is perpendicular to the calculated dominant orientation. The dominant orientation in this case is equal to the edge orientation, namely 135°. So, the kernel $K_{l,45}$ is used to calculate the external force at the kernel center (red point). In this scenario, even though the orientation of the pixel at the kernel center is a noisy (105 instead of 135), our method is able to pick the correct kernel, thanks to the calculation of the local orientation statistics. In the second row of the figure, the kernel center is located between to main edges at orientations 75° and 135°. $\aleph$ has 17 and 11 occurrences for each direction respectively. After applying (4.7), the dominant orientation becomes $Q(98.57) = 105°$. Based on that, the corresponding kernel that is equal to the dominant orientation, $K_{l,105}$,is used to compute the force vector. Again, with a noisy orientation of the pixel at the kernel center (15 instead of 75 or 135), our method is able to select the best kernel for this scenario. Choosing the kernel orientation to be the same as the dominant orientation in the case of competing edges helps the active contour to go easily through concave/convex parts of the image. Using the kernel $K_{l,105}$ to generate the force vector of the second row

example of Figure 4.6 will result in a vector that points towards the 75° edge but slightly to the south (the concave part), which enables the active contour to go through this area. On the other hand, choosing the orientation of the kernel to be perpendicular to $\bar{\theta}$ in case of a single edge, helps the active contour to be guided to the true edges even in noisy images.

4.5 Segmentation Results

In this section, the proposed model is tested using the same set of images in Section 3.5. The images are carefully selected to proposed different challenges to our model such as cluttered backgrounds, strong textures and sharp and complex edges. Additionally, our proposed model is compared to the state-of-the-art multi-resolution based methods, namely MVFC and MTVF as well as our UWAC model (Chapter 3).

Figure 4.7 illustrates the ability of the proposed model to segment complex shape objects in noisy images. The figure compares the proposed model to the state-of-the-art methods. In order to reflect the superiority of the proposed method over the other methods, Gaussian noise as well as a hand made noise (big speckles in the concave part) were added to the original image. It can be seen from the figure that, due to the complexity of the underlying object and the high level of noise, the other methods are not able to locate the correct boundaries of the object. The problem gets worse as the level of the noise increases. Our approach, on the other hand, is able to segment the object under all levels of noise. Due to the directional information incorporation and the local neighborhood consideration, our proposed method is able to pass over the hand made noise in the concave part to the correct boundaries, while other methods got caught by the noise. Moreover, as the noise level increases, our method is still able to correctly segment the object due to the use of adaptive kernel sizes. In such noisy images, the big kernels drive the active contour safely to near the true boundaries, while small kernels are allows the active contour to capture the fine details of the object.

Figure 4.8 shows an enlarged part of the external forces of both the MDVFC and MTVF at resolutions 5, 3 and 1 of the DT $\mathbb{C}$WT. At resolution 5, both methods perform well due to the less impact of the noise. However, at resolution 3 and due to the decreased kernel size and the directional selectivity of the proposed method, the force field vector (blue arrows) still consistently points towards the correct edge. The noise impact starts to be clear on MTVF at this resolution. At resolution 1, the kernel small size makes the noise impact on our method very less compared to the other method. It is very clear that the force vectors of the proposed method near the edge are still consistently pointing to the correct

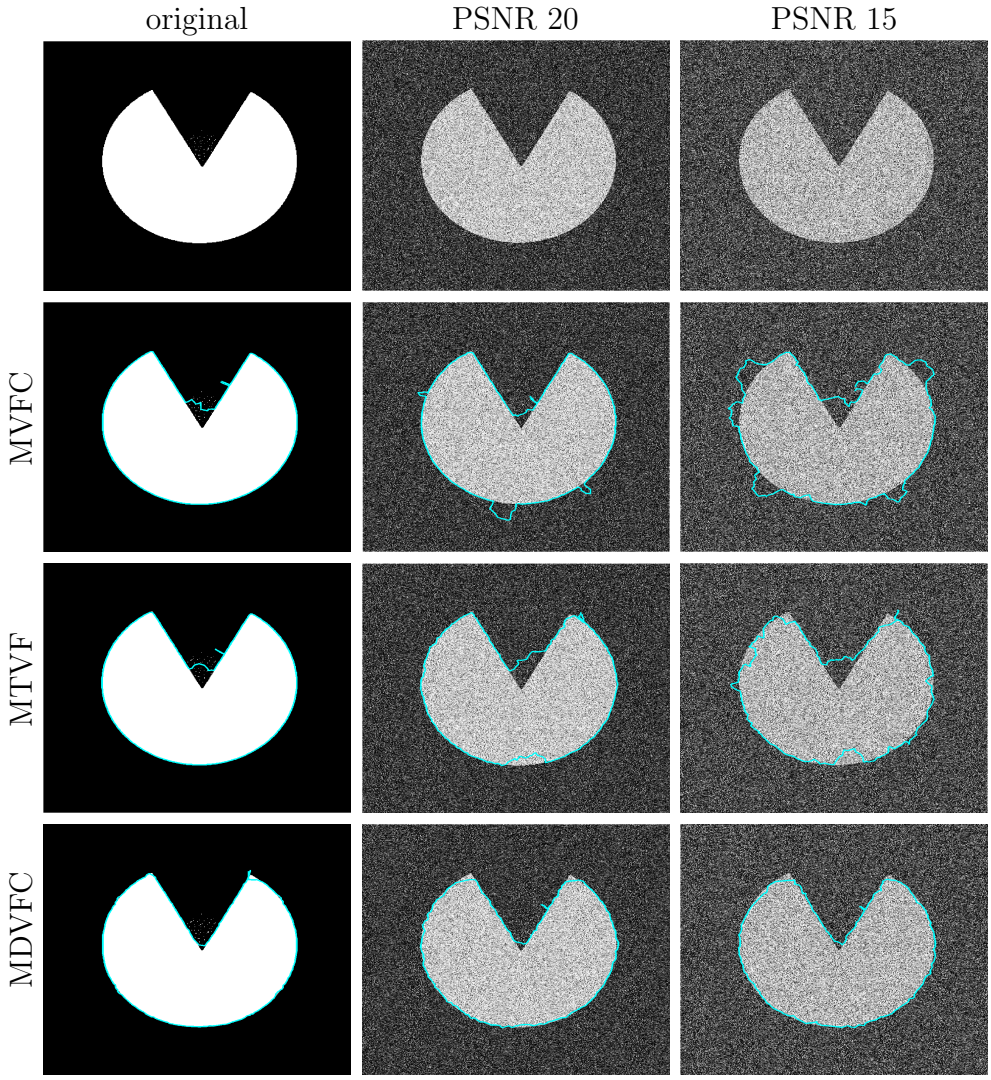


Figure 4.7: A comparison of converged contours for MVFC, MTVF and MDVFC. Incorporating the directional information of the DT ℂWT and the usage of steerable, adaptive kernels make the proposed MDVFC capable of locating correct edges in the concave part of the image even under high noise levels.

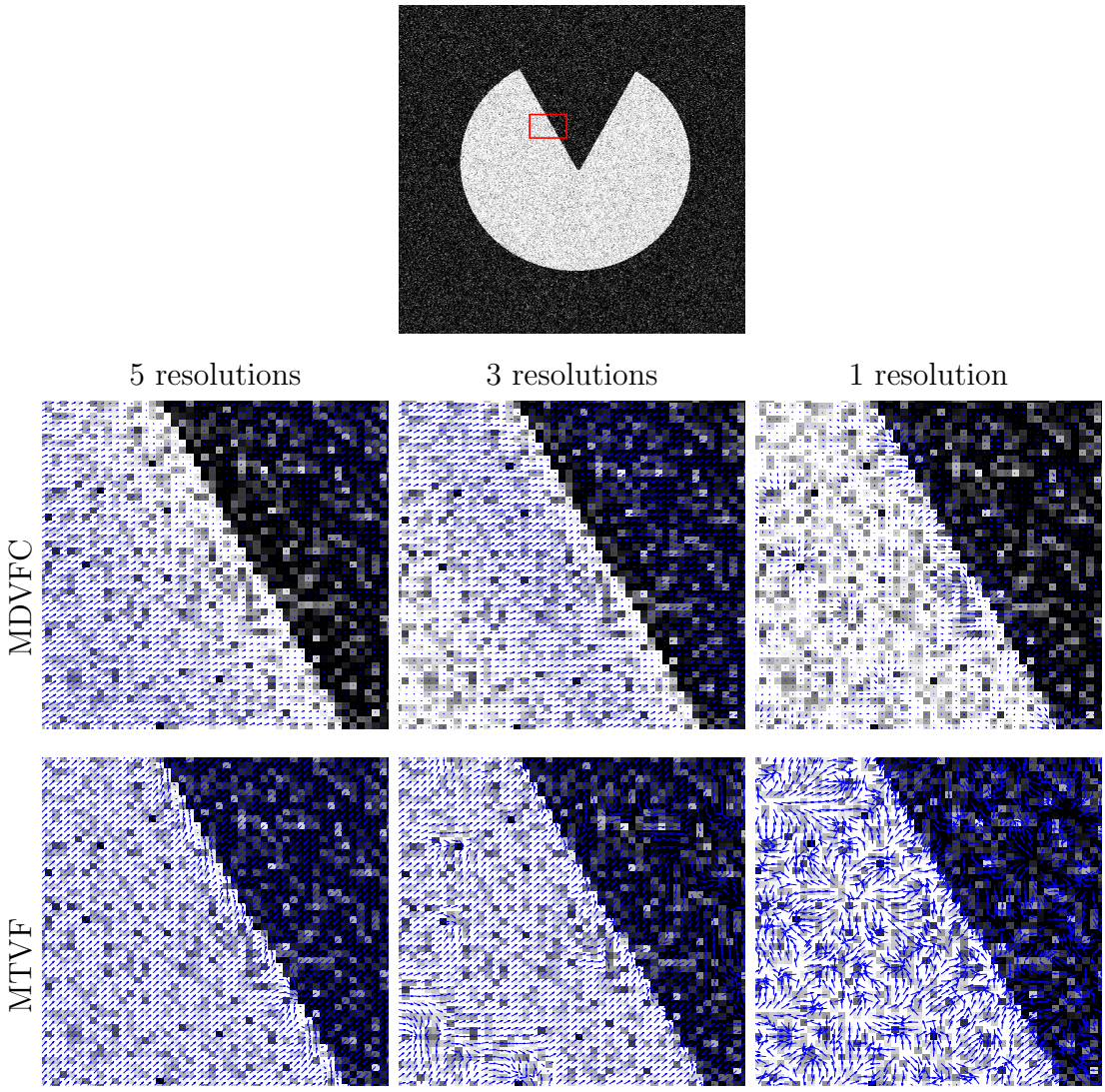


Figure 4.8: Enlarged external force fields (of the red rectangle) of MDVFC and MTVF at resolutions 5, 3 and 1. At resolution 5, both MDVFC and MTVF force fields point correctly to the exact edge location. However, at resolution 3 and 1 the, the directional selectivity and the adaptive kernel size make the noise impact on the MDVFC much less than the impact on the MTVF. Notice also that the force magnitudes (the blue arrow size) at resolution 1 in the MDVFC are less than the MTVF, which prevents the snake from taking large steps in such noisy environment

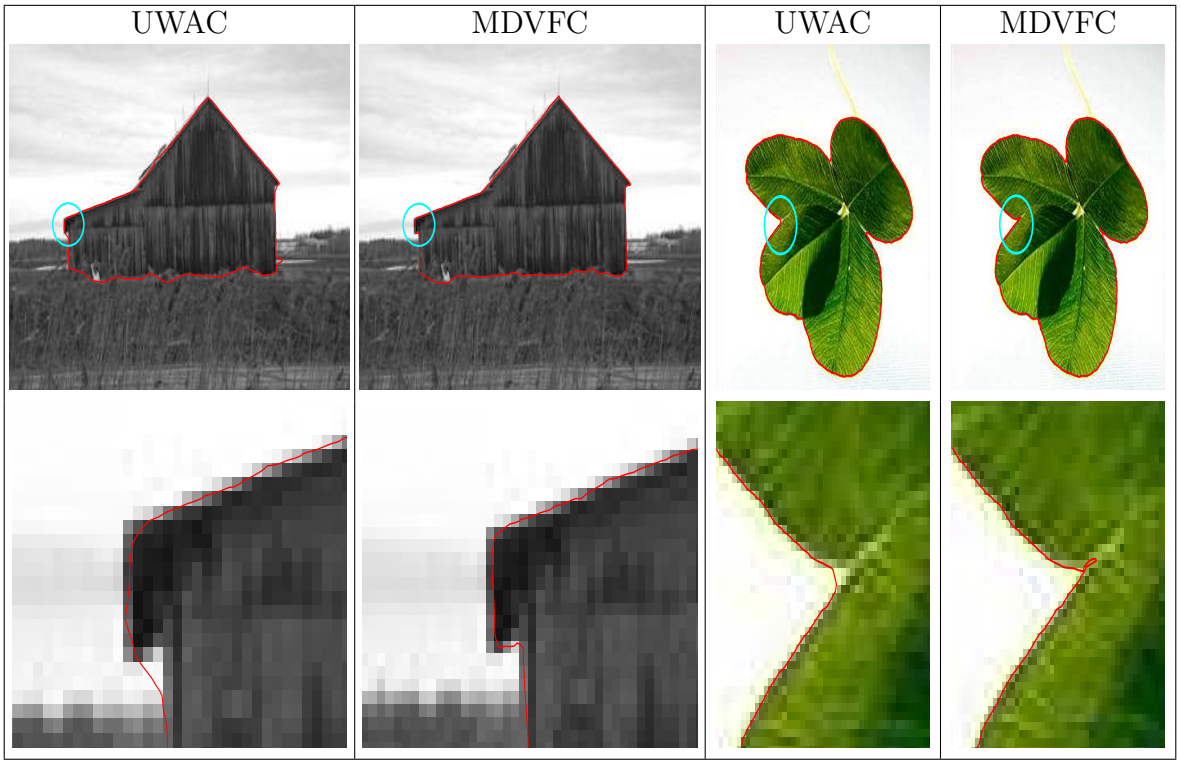


Figure 4.9: Comparisons of the final segmentation results of MDVFC and UWAC(Chapter 3). The figure shows the ability of the MDVFC, due to its oriented convolution kernels, to capture the fine details and sharp edges of the two images compared to UWAC.

direction, while the other method force vectors are pointing inconsistently everywhere. Another point to be noted from the figure is that the sizes of the blue arrows (force magnitude) are decreasing from resolution 5 to 1 in the proposed method. This allows the iterative solver of the active contour to take larger steps towards the final solution in the coarse resolution (5 and 3), while at finer resolution where the small details of the object need to be segmented, the small force magnitudes allow the model to take small steps to fine tune the final segmentation in order to capture these details. The other method have the force magnitudes almost the same across the entire resolutions which make the final segmentation results inaccurate as shown in Figure 4.7.

Figure 4.9 compares the final segmentation results of the MDVFC and the UWAC model developed in the previous chapter. Using two images from the Microsoft Research

Speckle noise Poisson noise Gaussian noise

Figure 4.10: Segmentation results of the MDVFC under different noise models. MDVFC, using the adaptive oriented kernels, is able to correctly detect the object boundaries under different noise models.

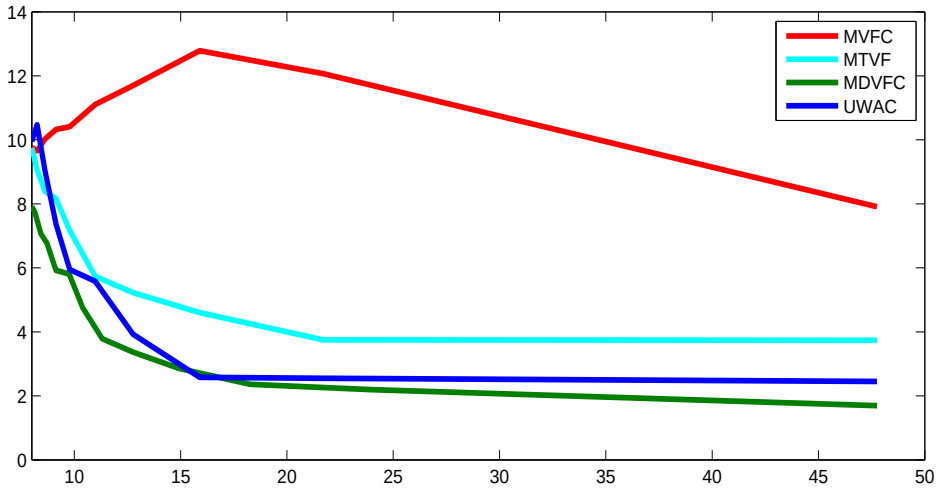


Figure 4.11: A comparison of the average performance of our proposed model (MDVFC) with MVFC [99], MTVF [10] and UWAC of the previous chapter, using 20 images. The proposed model outperforms all other methods under all levels of noise.

Asia (MSRA) Database [115], the figure qualitatively illustrates the ability of the MDVFC to capture the fine details and sharp edges of the underlying object due to the incorporation of the directional information revealed by the DT $\mathbb{C}$WT. Compared to UWAC, the small

size of the kernels used at the fine resolutions of transform as well as the selection the convolution kernel based on the dominant orientation and local statistics allow the active contour to get into the complex structures of the images, especially when two edges intersect (as can be seen in the enlarged parts of the images – bottom row). Figure 4.10 illustrates the performance of the MDVFC under different noise models (namely: Poisson, Speckle and Gaussian). The figure shows that our model is capable of correctly locating the object of interest under the three models of noise. At coarse scales, the noise noise contribution is limited and hence the big oriented kernels enable the active contour to take big steps towards the edges of the object of interest. Additionally, the over-complete nature of the transform ensures that the object features are well localized and that small structures are preserved, which in turn drive the active contour to the correct boundaries. At fine scales, the fine structures of the object of interest are detected using the small oriented kernel of our model.

Figure 4.11 shows the average segmentation results using the same image set of the previous chapter, It can be seen from the figure that incorporating the directional information revealed by the DT $\mathbb{C}$WT enables the MDVFC model to outperform all of the other methods including UWAC. The difference in the performance of the MDVFC and UWAC can be related to the usage of the oriented convolution kernels which increases the ability of the MDVFC to capture the fine details of the underlying object as discussed in Figure 4.9.

Figure 4.12 illustrates the idea of allowing the iterative solver to take large steps at the coarse resolutions while taking smaller steps at fine resolutions. The idea in the proposed method is to use the coarsest resolution image $I_{L,\vartheta^\star}$, where image strong features survive and noise contribution is minimum, to initialize and deform the model using the largest kernel $K_{L,\theta}$. Using bigger kernels at coarse resolutions produces stronger force fields, which allow big steps towards the edges. Again, because presumably strong features (like edges) will be dominant in coarse resolutions, taking bigger steps in such safe environments does make sense. After reaching its equilibrium state, the snake is then initialized in the next finer resolution using the previous resolution coordinates. Now, using smaller kernel will produce weaker external forces that allow the active model to take less aggressive steps towards the object of interest. At fine resolutions, noise becomes more influential, and hence using smaller kernels makes a good sense. The process continues until the finest resolution image $I_{1,\vartheta^\star}$ is convolved with the smallest kernel $K_{1,\theta}$, which allows the accommodation of the object fine details as well as ensuring that the snake is not pulled towards spurious noise. It can be seen form the figure that, at coarse resolutions (5 and 3) the iterative solver takes large steps (the distance between the blue lines) towards the object. While at fine resolutions (resolution 1), only small steps are allowed to accommodate the fine details of the underlaying objects. Examples of these fine details are the corner of the concave

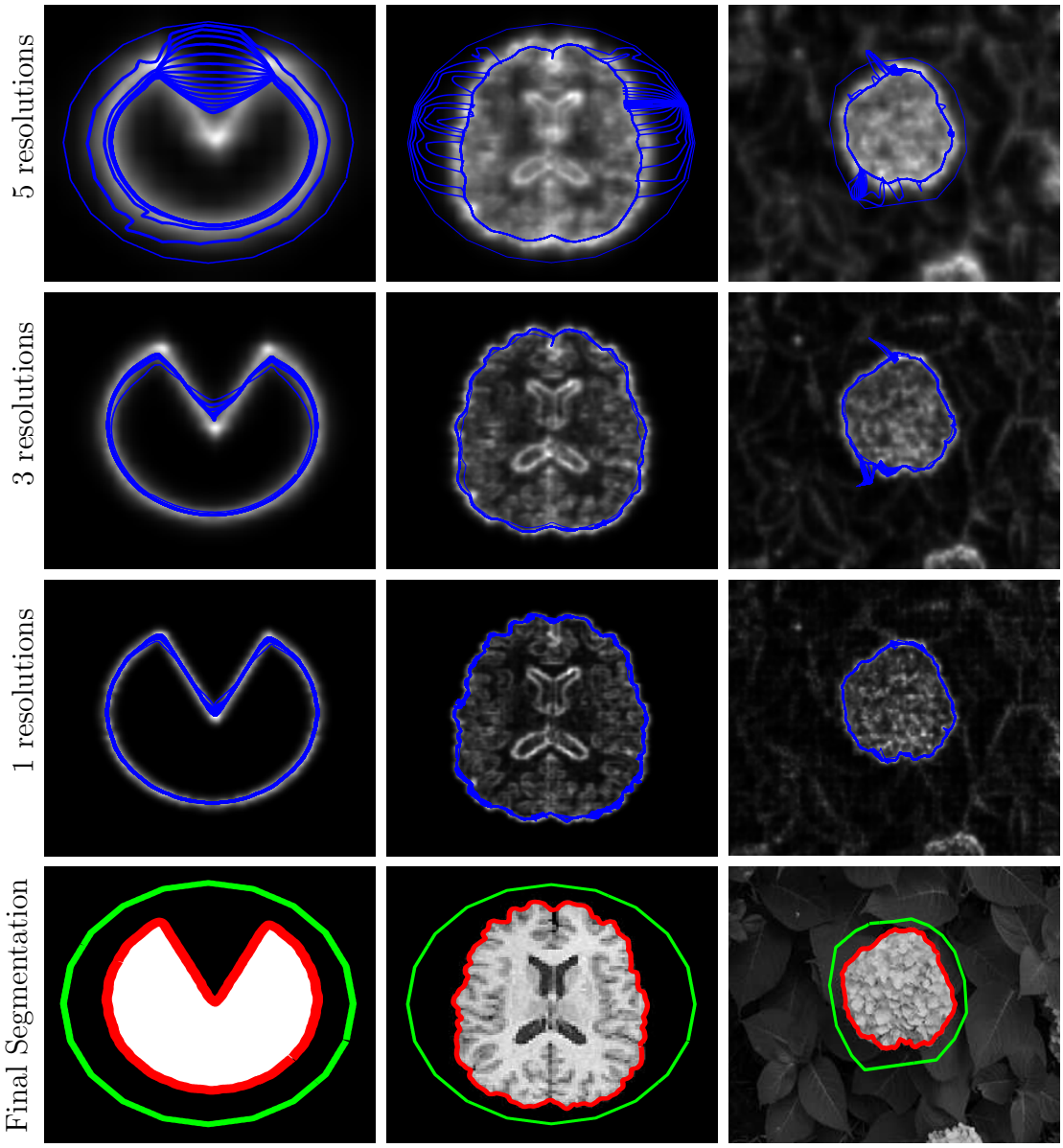


Figure 4.12: Iterative solver steps towards the final segmentation. At coarse resolutions, the model can take big steps towards the final solution, where noise contribution is minimum. At fine resolutions, the small size kernels allow the model to take smaller and less aggressive steps, and hence the proposed method is able to capture the fine details of the three images.

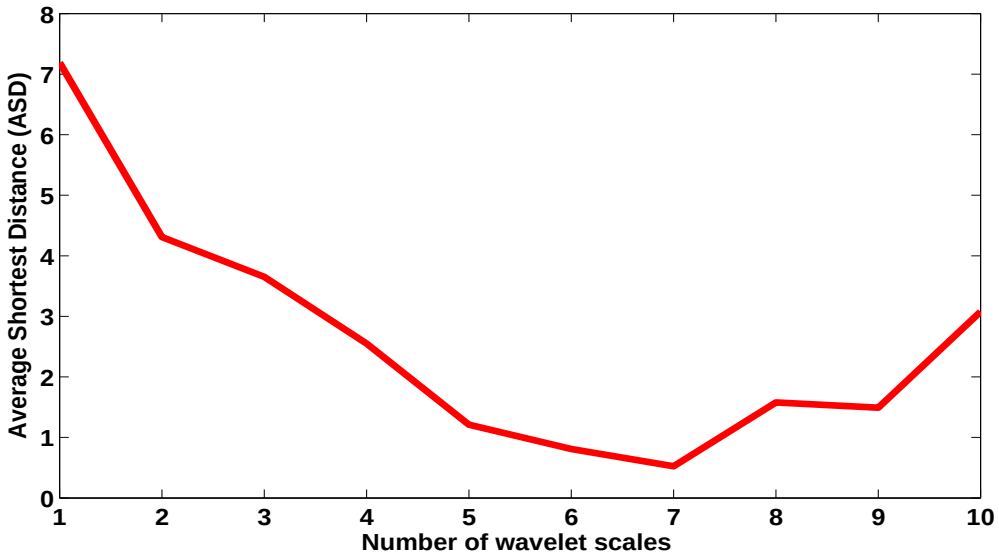


Figure 4.13: Segmentation accuracy versus number of wavelet scales. Though strong features (e.g edges) are most likely to survive at coarse scales, a significant increase in the number of scales negatively affects the accuracy of the final segmentation.

part of the disk image and the high irregularities of the outer boundaries of the brain and the flower images.

Finally Figure 4.13 addresses the question of how many resolutions of the DT $\mathbb{C}$WT should be used. The figure shows that using 5-9 scales of the transform gives the best ASD metric. As we go higher in the number of scales, salient features become weaker which consequently influences the consistency of the directional information in these scales. This explains the high ASD values when a high number of scales is used. On the other hand, when a low number of scales is used, the noise contribution in the force maps of these scales increases which results in higher ASD values.

Chapter 5

Optical Coherent Tomography (OCT) Corneal Image Segmentation

With the modern advances in medical imaging technologies, contactless modalities, that require no physical contact with the sample, have become more convenient for patients. However, the contactless property of these modalities adds new challenges that affect the quality of the images, such as the low signal to noise ration (SNR). Recently, OCT imaging modalities have become very useful and widely used for imaging different human organs. However, OCT images suffer from different challenges that affect their quality, including speckle noise cause be ligh reflections and signal roll-off. Particularly, in this chapter, the automatic segmentation of corneal optical coherent tomography (OCT) images is proposed. The chapter tackles a number of corneal OCT-related problems such as active contour initialization and motion correction. Since the work of Chapters 3 and 4 addressed the problems of noise sensitivity and complex object segmentation, the OCT segmentation problem could be seen as a good domain for the direct implementation of our work.

To make this chapter self-contained, Sections 5.1 and 5.2 give brief introductions to the human cornea anatomy and the optical coherent tomography technology, respectively. The rest of the chapter is organized as follows: Section 5.3 illustrates our proposed method and gives solutions to a number of OCT-related problems such as low SNR, epithelium thickness and motion correction. Finally, Section 5.4 illustrates the segmentation results of our proposed method and gives comparisons to other methods.

Some parts of this chapter are excerpted verbatim from [116], [117] and [118], where the thesis author is on of the co-authors.

5.1 Human Cornea

The cornea is the transparent front part of the eye that covers the iris, pupil and anterior chamber [119]. Both the cornea and the lens refract light of approximately 2/3 of the optical power of the eye. Since transparency is of key importance, cornea does not receive its nutrients via blood vessels. Instead, tear fluid, aqueous humour and the neurotrophins supplied by nerve fibres are the main sources of cornea nutrients. Normally, a human cornea has a diameter of 11.5 mm and is thicker at the periphery 600-800 μm than the center 500-600 μm.

Figure 5.1 illustrates the human cornea anatomy. Typically, a human cornea is a dome-shaped membrane with five layers [120]:

1. **Epithelium**: a layer, with thickness of 5-6 cells, that covers the cornea. The epithelium is a self-healing layer in the sense that its cells are fast-growing and easily regenerated. The typical thickness of the epithelium layer is 60 μm.

2. **Bowman's membrane**: is a tough layer mainly composed of randomly organized type I collagen fibrils, and lies just under the epithelium. It's typical thickness is $10 - 15\mu m$.

3. **Stroma**: is the thickest layer of a human cornea with a typical thickness $500\mu m$, and consists of regularly arranged collagen fibers which gives the cornea its transparent feature. Stroma lies just under Bowman's membrane.

4. **Descement's membrane**: a thin, elastic, and perfectly transparent cellular layer composed mainly of collagen type IV fibrils (less rigid than type I). Descement's membrane lies between the stroma and the endothelium and has a typical thickness of $5 - 20\mu m$.

5. **Endothelium**: a single cellular layer, with a typical thickness of $5\mu m$, that maintains the dehydration level of the cornea required for optical transparency. If exposed to physical damage, the endothelium will not be regenerated.

Recently, a new layer, "Dua's layer" [122], was discovered to be exist between the corneal stroma and Descemets membrane. Dua's layer is composed primarily of collagen and is so strong to the extent that certain types of corneal surgeries that inject air bubbles into the cornea could rely more on that layer rather than the other fragile layers.

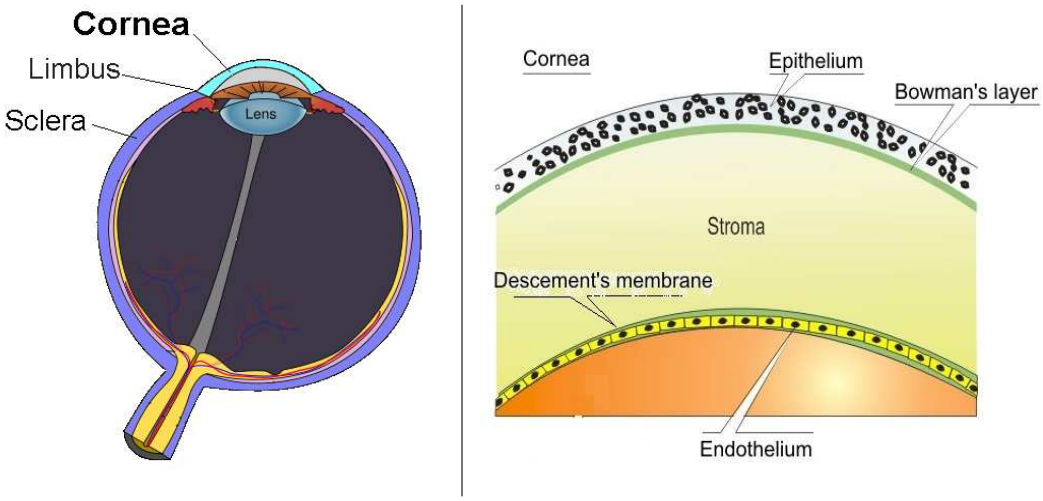


Figure 5.1: Human cornea anatomy [121].

Human cornea can be affected by a number of diseases [123] including allergies, conjunctivitis, dystrophies and keratoconus. Among the different corneal diseases, Keratoconous cornea (KC) is of our main interest in this research. KC is a corneal disease that affects around 0.1% of the population causing breakage of the collagen fibrils that control the normal rounded shape of the cornea, which in turn changes the cornea curvature to be to a cone-shaped. In early stages of KC, rigid contact lenses are used for treatment, while in its severe stages, KC can lead to distorted or blurred vision and a surgical treatment is usually needed. Figure 5.2 shows the differences between a normal and a Keratoconus cornea. Surgical treatment of KC requires precise calculations of the thickness and the curvature of both the donor/healthy and the patient/diseased corneas [116]. Clinical examination of KC is currently accomplished using instruments that are unable to visualize the internal structure of the cornea, such as slit lamp biomicroscopes and corneal topographers. Other instruments that have the ability to visualize the internal structure of the cornea are suffering from the disadvantage of imaging a very small area of the cornea (typically 1mm x 1mm) such as confocal microscopy and specular microscopy. Recently, OCT technology started to take its place in imaging human eyes with the two major advantages of being non-contact technology (which increases patients comfortability) and producing higher resolution and speed images (which enables more investigation of the eye structures).

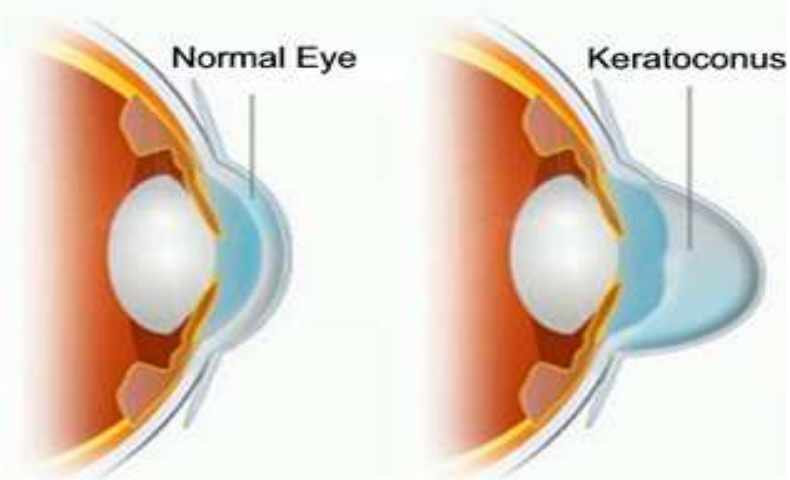


Figure 5.2: Normal versus keratoconus cornea. Due to the breakage of the collagen fibers, the dome-shape healthy cornea changes to a cone-shape KC cornea [124].

5.2 Optical Coherent Tomography

Optical Coherent Tomography (OCT) [125] is a non-invasive imaging technology that uses light waves to provide real-time and high-resolution (micron scale) cross sectional images of tissue structure. Unlike other modalities, such as ultrasound, X-ray and MRI, that are relatively slow, use ionising radiation, or require a physical contact with the sample, OCT has the attractive attributes of a short acquisition time, non-ionising radiation, and contactless imaging, making OCT more tolerant to sample motion, safer for long exposure times, and generally more convenient for patients [126].

Currently used instruments for clinical examination of KC corneas prior to and after surgery such as slit lamp biomicroscopes or corneal topographers lack the ability to visualize the internal layer structure of the cornea [127] [128]. High resolution microscopy techniques such as confocal microscopy and specular microscopy can visualize individual corneal cells such as endothelial, epithelium and keratocyte cells, as well as the fibrous structure of the Bowmans and Descement's layers [129]. However, because of the high magnification requirements, the imaged area of the cornea is very small (typically 1 mm x 1 mm), and physical contact with the cornea is often required which causes considerable discomfort for patients.

OCT has proven to be very useful for clinical evaluation of KC corneas [130], as it provides

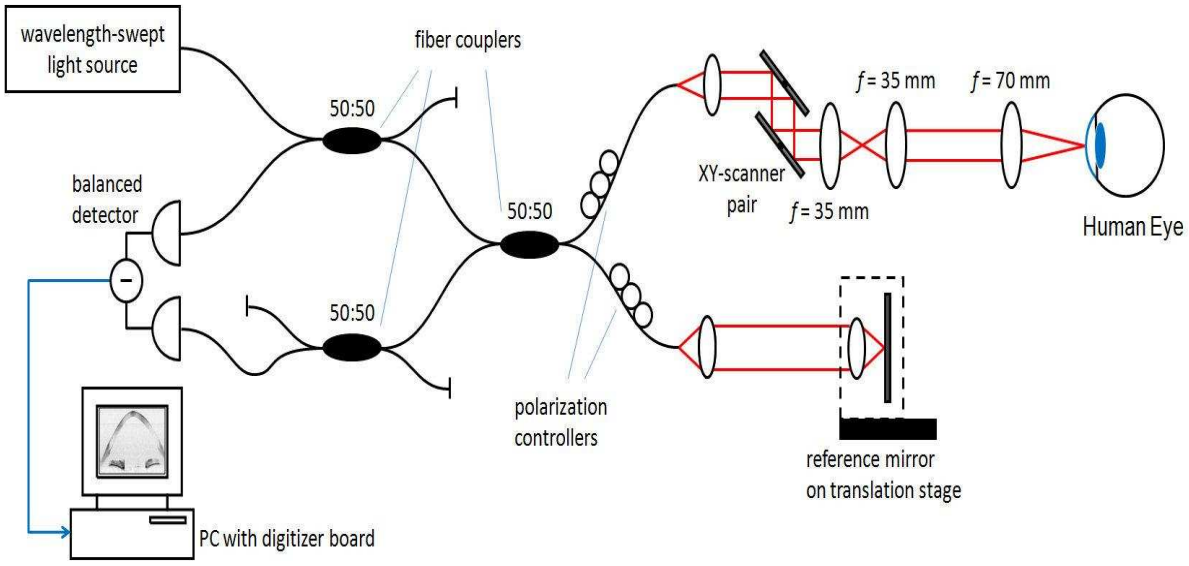
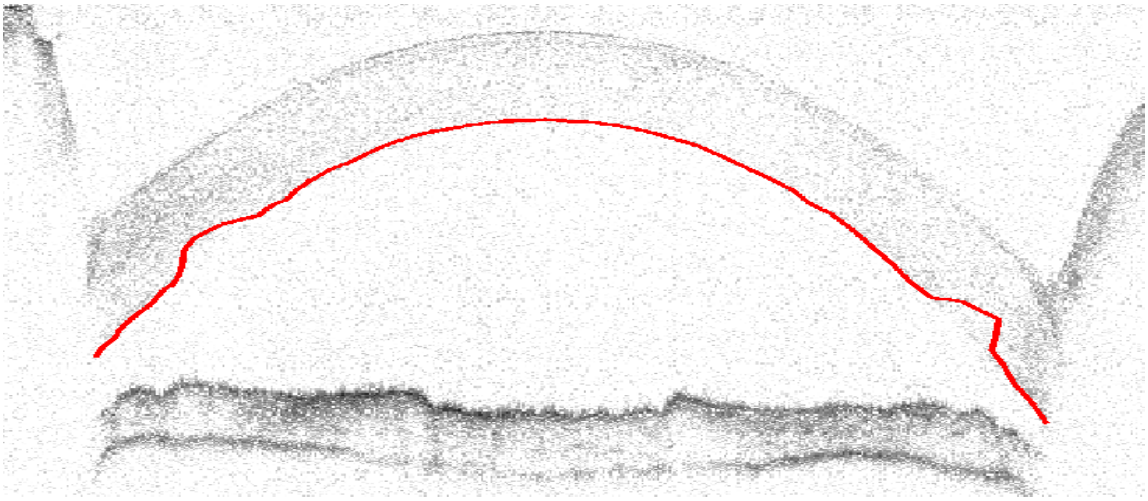


Figure 5.3: The swept-source setup is based on a fiber-optic Michelson interferometer. This configuration consisting of three 50:50 fiber couplers allows dual balanced detection while minimizing the residual DC-signal caused by the wavelength-dependent coupling ratio. The long focal length objective provides a long depth of focus and reduces the geometric distortion of the field of view [116].

cross-sectional images of the cornea with a contact-free measurement. Research-grade ultrahigh-resolution optical coherence tomography (UHR-OCT) systems, that have been developed over the past few years [131] [132] enable visualization of thin corneal layers, precise measurement of the layer thickness [133], and in some cases, visualization and counting of individual corneal cells [134]. Until now, state-of-the-art UHR-OCT systems based on spectrometers could only provide a very limited imaging depth range (1 mm) due to the low pixel number of available detector arrays. This is insufficient for covering the full volume of a cornea, especially in the case of keratoconus where the curvature can be significantly steeper than normal. Recently, high-speed imaging of the entire anterior segment of the eye using a spectral-domain OCT (SD-OCT) system equipped with a new prototype line camera was demonstrated [135]. However, the long imaging range of 6 mm ($\times 2$ after removal of the complex conjugate mirror image) is only possible with moderate axial resolution ($10\mu m$ in air), and there is a strong signal roll-off for longer probing depths. In contrast, swept-source OCT (SS-OCT) systems relying on rapid wavelength-swept laser light sources enable high-speed imaging with a long depth range [136]. Commercial swept



GVF segmentation – leakage at peripherals

Figure 5.4: In a typical corneal OCT image, the inherent speckle noise as well as the signal roll-off at the peripherals of the cornea mislead the GVF active contour [7] to spurious image features causing segmentation leakage.

sources have become available that provide high axial resolution and support depth ranges longer than 5 mm with little signal roll-off [137], which is sufficient for imaging the entire cornea.

During this research, our team members, at the Physics and Astronomy department of the University of Waterloo, have developed a new SS-OCT modality for examining KC corneas (Figure 5.3). The proposed modality is a high-speed, high-axial-resolution, long-scanning-range SS-OCT system. A commercially available swept source operating at a sweep rate of 100 kHz, which enables fast acquisition of large datasets, has been applied in the proposed modality. With 106 nm total bandwidth centered at 1040 nm, it provides $\sim 7\mu m$ axial resolution in air ($5.1\mu m$ in corneal tissue). The scanning range of $5.6mm$ is sufficient to image the entire cornea down to the iris plane, enabling a comprehensive assessment of the cornea shape.

5.3 OCT image segmentation

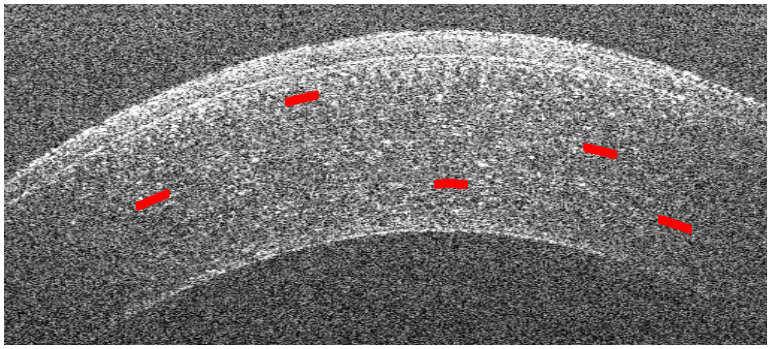
Because OCT is based on visible light, media opacity highly affects the quality of the images. Moreover, the light reflections during the imaging process make speckle noise an

inherent feature of OCT images. Corneal OCT images, in particular, have rather weak signals towards the sides of the cornea due to the reduced amount of reflected light. Additionally, corneal OCT images usually have poor signal-to-noise ratio, particularly because images are sampled quickly, at high resolutions, and in-vivo (with no physical contact with the cornea). With such challenges, classical segmentation methods such as histogram thresholding [36], region growing [40] and watershed [33], give very poor segmentation results. Hence, active contour based-methods emerge as potential solutions to OCT image segmentation [118], due to their ability to accommodate some of the OCT aforementioned problems, such as signal absence and low SNR. However, even with the usage of classical active contour methods, such as GVF and VFC, the results are still not good enough. Figure 5.4 illustrates some OCT image segmentation challenges. It can be seen from the figure that both the speckle noise and the signal roll-off at cornea peripherals make the GVF [7] unable to properly segment the image. Hence, a call for more advanced active contour-based segmentation techniques, that can handle such challenging circumstances with a high level of accuracy, is important. In this chapter, our developed method in the previous chapter (Chapter 4) will be used to tackle the high noise problem of OCT images. Moreover, the problems of signal roll-off and active contour initialization, specifically in OCT images, are subject to more investigation and study in the remaining of the chapter.

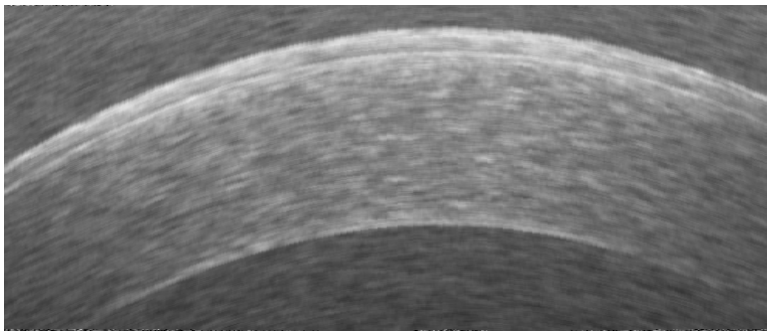
The following sections are organized as follows: Section 5.3.1 addresses the problems of low SNR and signal roll-off, while Section 5.3.2 addresses the problem of active contour initialization. Epithelium layer thickness and motion correction are addressed in Sections 5.3.3 and 5.3.4, respectively. Finally, Section 5.4 shows different segmentation results of human corneas.

5.3.1 OCT Signal Enhancement

Speckle noise [138] and the signal roll-off at corneal peripherals [118] are the two main challenges for OCT segmentation methods. Figure 5.4 shows the segmentation result of the bottom layer of a human cornea using the classical GVF method [7]. It can be seen that, due to the signal roll-off and the speckle noise, the active contour is misguided towards the spurious features of the image, which in turn leads to incorrect final results. In addressing this problem, the authors of [139] estimated the locations of low SNR using the second derivative of the converged active contour. Linear interpolation is then used to fit the contour in these regions of low SNR. Admittedly, the use of curve interpolation to estimate the missing data in OCT could be misleading, especially in KC where the shape of the cornea changes significantly, compared to healthy corneas, from one spatial point to



Examples of $h(x)$ (red patches)



Enhanced image $\hat{I}$

Figure 5.5: The spatially varying kernel, h, used to enhance the received signal, especially at cornea peripherals. The direction of the kernel changes to be aligned with the cornea layers. Prior information about the cornea structure is used to steer h.

another. Alternatively and to avoid linear interpolation problems, our proposed method uses a spatially varying kernel, $h(x)$, to enhance the received signal as follows:

$$\hat{I}(x) = h(x) \star I \tag{5.1}$$

where $\hat{I}$ is the enhanced image, $\star$ is the convolution operator and I is the original image. Based on the prior knowledge of the cornea shape, the orientation of the kernel h changes according to its location inside I. The width and the height of h are modality dependent, and are empirically selected of to be 5 and 15 pixels respectively. The prior information about the cornea curvature (whether it is healthy or KC) [140] [141] is used to change the orientation of h on a pixel basis to accommodate the underlying cornea structure. Figure 5.5 shows the spatially varying kernel h and how its orientation changes to be aligned

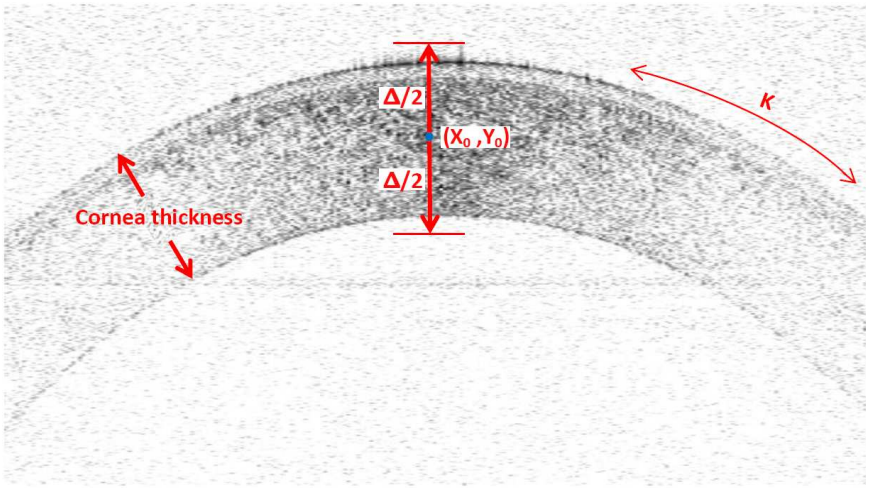


Figure 5.6: The Generalized Hough Transform (GHT) search space. GHT tries to find the best (x_0, y_0) through which a dilated parabola with curvature k and thickness Δ passes. The thickness of the dilated parabola should be slightly wider than the actual cornea thickness, so that active contour is not pulled towards the inner layers of the cornea (e.g. Bowman's membrane).

with the real edges in the image. Having $h(x)$ steered towards real edge orientations, the received signal in the enhanced image $\hat{I}$ is dramatically improved, especially at cornea peripherals where the amount of the reflected light is very limited. Looking at the peripherals of the cornea in the enhanced image, it can be seen how the edge information is present, which eases the task of the active contour and prevents the leakage problem (see Figure 5.4).

5.3.2 Active Contour Initialization

Initializing the active contour near the object of interest is considered one main issue of active contour-based methods [24] [142], especially in noisy images (such as OCT) where the risk of the active contour being trapped by spurious image features is high. Since KC corneas, in most, cases have complex shapes, and since this work aims to segment both healthy and non healthy OCT cornea images, our proposed method uses the generalized Hough transform (GHT) [143] to find the best initialization of the active contour inside the

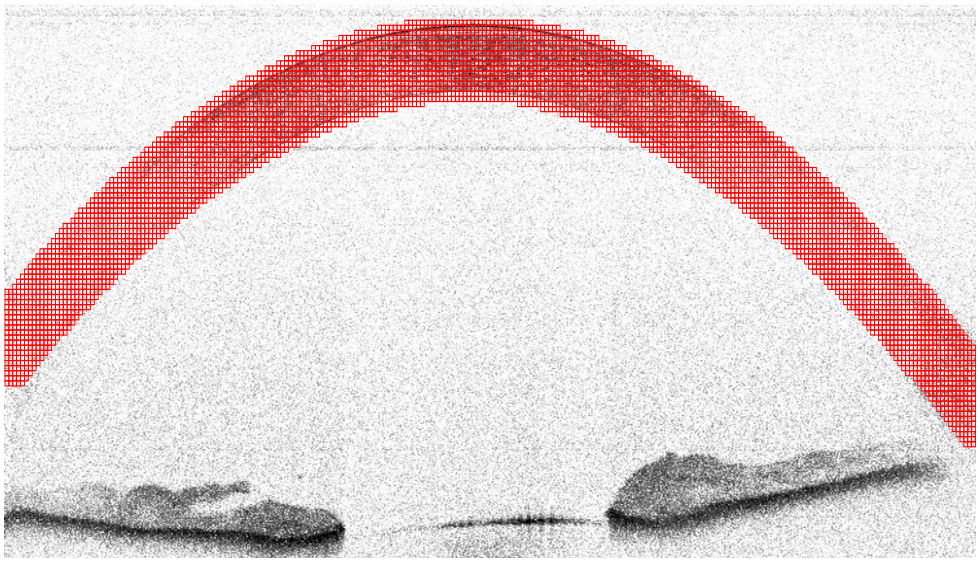


Figure 5.7: Active contour initialization using the GHT. The active contour is initialized at the boundary of the best dilated parabola $\mathfrak{R}_i^*$ (red mesh) found by the GHT.

images. GHT is a modification of the original Hough transform [59], that is able detect not only analytically defined shapes (e.g., line, circle), but also any arbitrary object described with its model. In order to find a good estimation of the cornea in the image, GHT scans image pixels (x_i, y_i) to find the best pixel (x_0, y_0) through which a set of dilated parabolas $\mathfrak{R}_n$ passes. Figure 5.6 illustrates the idea of using the GHT to estimate the cornea location inside the image and uses this information to initialize the active contour. The best parabola $\mathfrak{R}_i^*$ with curvature k and thickness Δ should cover an area that is slightly wider than the actual cornea thickness, which in turn protects the active contour from being pulled towards the inner layers if the cornea (e.g. Bowman's membrane), and reduces the risk of the active contour being trapped by the speckle noise.

To that end, the image blurring step carried out in the previous subsection using the kernel h helps achieve two goals. First, to reduce the impact of the speckle noise and any artifacts that might happen during the image acquisition process. Second, and more importantly, to diffuse the highly bright pixels lying on the anterior and posterior layers of the cornea. This diffusion increases the brightness of the set of pixels right above the anterior and right below the posterior. By increasing the brightness of these pixels, most likely they will be included in the search space of the GHT, and hence guarantees that $\mathfrak{R}_i^*$

has a thickness that is slightly wider than the actual cornea thickness.

Before evaluating the pixels covered by each parabola using the GHT, the parabola is dilated in order make it thicker, which ensures that it covers the whole cornea. The amount of dilatation Δ is determined from the thickness space $[\Delta_{min}\ \Delta_{max}]$ [141] [144]. Also, image pixels are penalized to give preference to corneal data over non corneal data. Assuming that the cornea pixels are brighter than other pixels in the image (if this is not the case, the image is flipped $\hat{I} = 255 - \hat{I}$), the penalization process happens as follows:

$$\hat{I}_p(x_i, y_j) = \begin{cases} \hat{I}(x_i, y_j) & \text{if } \hat{I}(x_i, y_j) \geq T \\ \mathfrak{P} & \text{otherwise} \end{cases} \tag{5.2}$$

where $\hat{I}_p$ is the penalized image, T is the penalization threshold and $\mathfrak{P}$ is the penalization parameter. The value of T is empirically calculated using the image data at the upper right and left corners, where there is a guarantee that no cornea is present. The GHT scans $\hat{I}_p$ to find the pixel (x_0, y_0) that makes the parabola

$$y - y_0 = k(x - x_0)^2 \tag{5.3}$$

with a curvature k_i and thickness Δ_j covers the largest set of pixels having intensity values higher than T. Figure 5.7 shows the best dilated parabola $\mathfrak{R}_i^*$ found by the GHT. It can be seen that the found parabola covers a slightly wider area than the actual cornea in the image. This ensures that the active contour will not be misguided by any other layers inside the cornea (e.g. Bowman's membrane) or by any spurious features such as speckle noise. Now, the problem turns out to be an optimization of the following evaluation function E

$$E = GHT(I_p, x, y, k, \Delta) \tag{5.4}$$

such that

$$|(y - y_0) - k(x - x_0)^2| \leq \Delta/2 \tag{5.5}$$

In order to speed up the above optimization, the size of the search space needs to be reduced. Hence, the prior information about the thickness, curvature and the apex of normal and keratoconus corneas [140] [141] are used as the boundaries of the GHT search space. Figure 5.8 shows the initialization results for different KC corneas. The figure shows three different cases of KC corneas, 1) severe KC cornea, 2) intacs cornea, where two small crescent-shaped plastic polymer pieces are inserted into the cornea to reshape it, 3) Penetrating Keratoplasty (PKP), where the damaged cornea is replaced with a donor cornea. In the figure, regardless of the big shape deformation in the three images, it can be seen that our proposed method accurately accommodates the different shapes of the

cornea, and correctly finds the best dilated parabola $\mathfrak{R}_i^*$ to initialize the active contour close to the actual upper and lower boundaries of the cornea.

After estimating the best initialization of the active contour using the above method, the MDVFC, developed in Chapter 4, is fed with this estimation and the upper and the lower layers segmentation of the cornea is calculated. With the segmentation coordination of both layers in hand, the following sections address the problem of epithelium thickness and motion correction.

5.3.3 Epithelium Layer Thickness

It has been well documented that Stromal thinning is a KC characteristic, particularly at the apex of the cone [145]. However, the limitations of imaging modalities make the effect of the disease on other corneal layers unclear [146]. Recently, a number of researchers have shown that epithelium thinning is another characteristic of KC eyes. In [147], the epithelium layer profile is generated by manually selecting 11 points on the interface of the layer. Then, a spline interpolation is carried out to get the first estimate of the layer profile. Subsequently, around the initial estimate, a ± 5 pixel region is searched for the highest signal intensity. Finally, the highest intensity pixels are fitted using a fifth order polynomial to generate the final epithelium. The method in [147] has a number of issues. First, the manual selection of the 11 points could be time consuming and inaccurate, especially in the scenarios where a large amount of data is to be processed. Second, searching for the brightest pixels in a ± 5 area certainly will be misleading, especially with the high amount of speckle noise due to the light reflections. Third, the polynomial fitting using a fifth order function could be, in most KC eyes, inaccurate since the curvature of the epithelium layer changes significantly from one spatial point to another. Hence, in this section, a novel method for accurately and automatically measuring the epithelium layer thickness is developed.

Given the converged MDVFC active contours of the anterior and posterior layers of the cornea, $y^a(x)$ and $y^p(x)$, the maximum thickness of the cornea is calculated as:

$$\Delta = \max_x |y^a(x_i) - y^p(x_i)| \tag{5.6}$$

It is well known from the literature that out of the approximately $540\mu m$ thickness of the healthy cornea, the epithelium thickness is $53.4 \pm 4.6\mu m$ [148]. Also, for KC cornea, it is known that the epithelium thickness is $45.7 \pm 5.9\mu m$ [149]. This prior information is used

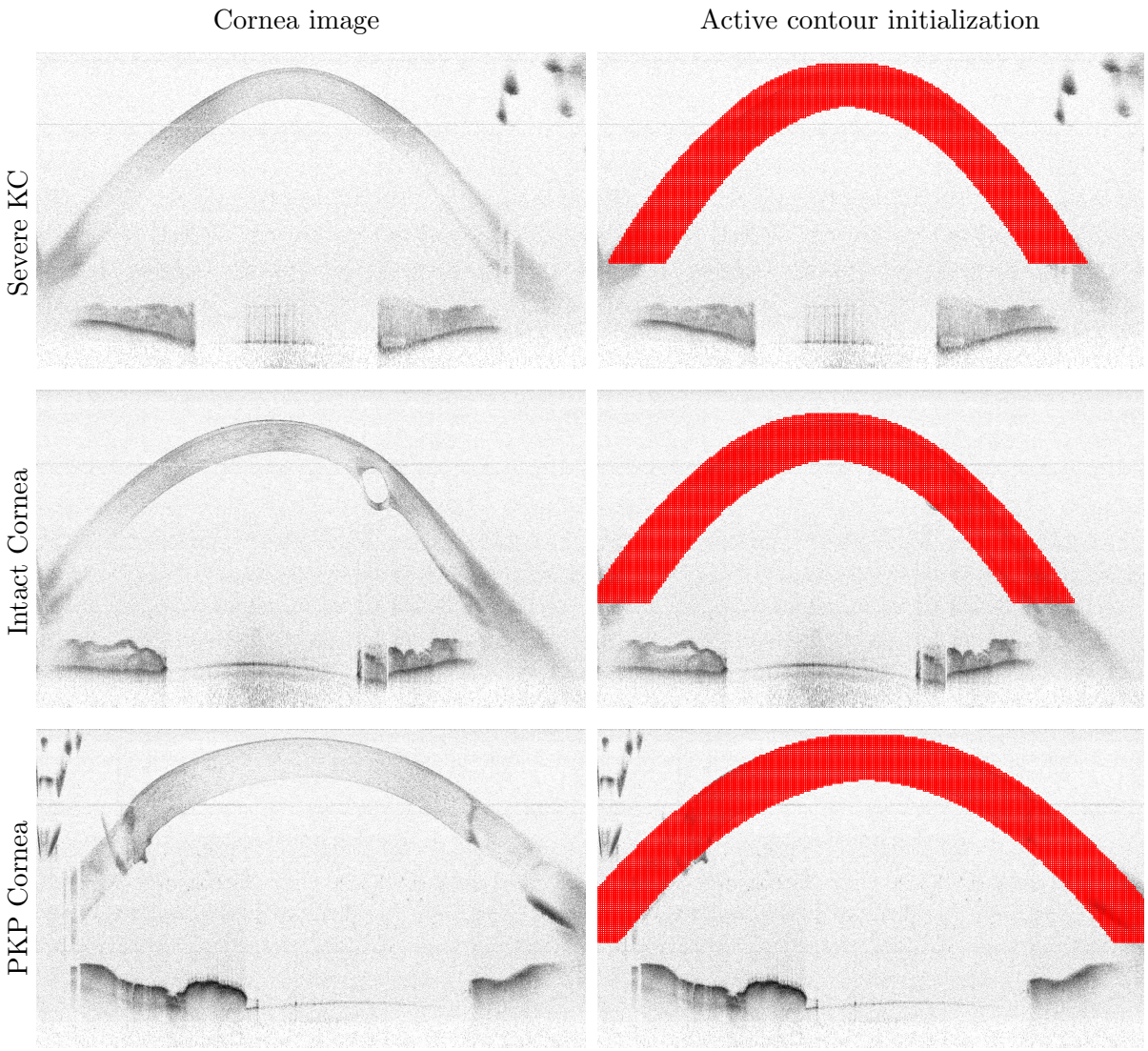
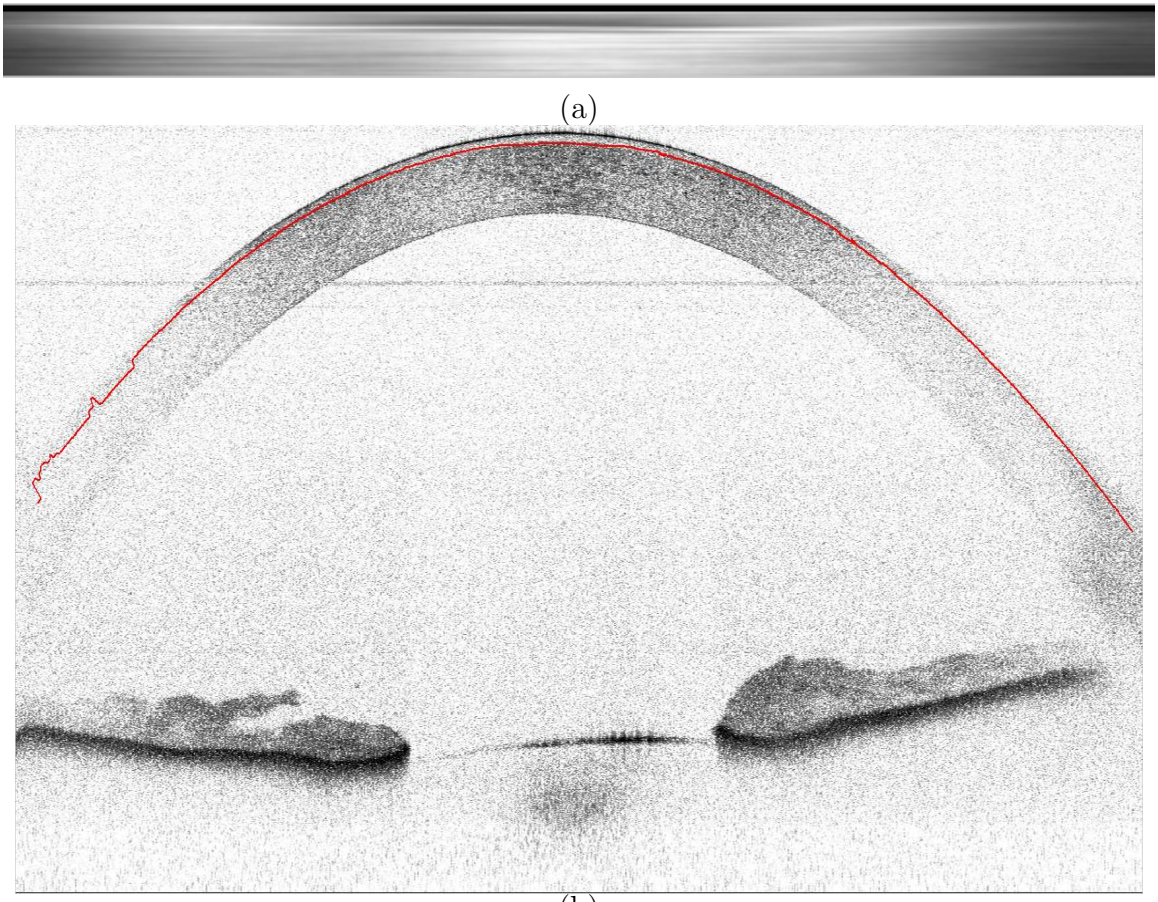


Figure 5.8: Active contour initialization using the GHT for different types of KC corneas. It can be seen that our proposed method is able to accommodate the different curvatures of the human cornea for all different cases.

to limit the search space of our method to only the upper part, $\mathfrak{U}$, of the cornea, where the epithelium layer is guaranteed to be found. $\mathfrak{U}$ is a modality dependent parameter and is proportional to the calculated Δ. Empirically, $\mathfrak{U}$ is set to 1/8 of the maximum cornea thickness Δ during our experiments.



(a)

(b)

Figure 5.9: Epithelium Layer segmentation. (a) The curvature of the converged MDVFC contour $y^a(x)$ is used flatten $\ddot{I}$ in Equation 5.8. It can be seen that epithelium pixels have high intensity values compared to other non-epithelium pixels. (b) The final segmentation of the epithelium layer

.

To detect the epithelium layer, a kernel $\aleph(x)$ with height 2α and width 2γ is defined. Each snake point (snaxel) in $y^a(x_i)$ is perpendicularly shifted by a distance d which guarantees that the new snaxel is not lying on the bright upper boundary (anterior layer) of

the cornea. d is a modality dependent parameter and is empirically set to 8 pixels in our experiments. The new contour $y^{new}(x)$ is iteratively perpendicularly shifted by a distance ϵ to generate another contour as follows

$$y^{new}(x) = y^{old}(x) - \epsilon \quad \forall \quad y^{old}(x) \in \mathfrak{U} \tag{5.7}$$

where the subtraction happens along the perpendicular direction of each snaxel. For each pixel $I(x, \alpha)$ lying on y^{new}, a new value $\ddot{I}(x, \alpha)$ is calculated as follows:

$$\ddot{I}(x, \alpha) = \frac{1}{N} \sum_{i=-\delta}^{\delta} \sum_{j=-\gamma}^{\gamma} I(x - i, \alpha - \gamma) \tag{5.8}$$

where N is the number of pixels covered by $\aleph$. The parameters of the kernel, α and γ, are modality dependent, and based on the empirical experiments of this study they are set to 5 and 15 pixels respectively. For efficient calculation of the epithelium thickness, the curvature of the converged MDVFC contour, $y^a(x)$, is used to generate a flattened version of $\ddot{I}$. Figure 5.9a shows the flattened version of $\ddot{I}$ with inverted brightness. Due to the fact that the new contours, y^{new}, are perpendicular shifted versions of the converged anterior contour, $y^a(x)$, and that the kernel is elongated in the direction parallel to the anterior layer, the epithelium layer pixels have higher intensity values compared to non-epithelium pixels. The image in 5.9a is searched vertically (column by column) to locate the pixel with the maximum intensity for each column. Figure 5.9b shows an example of the final segmentation of the epithelium layer. It can be seen from the final result that our method is capable of locating the correct epithelium layer even at the peripherals where the SNR is very low. The averaging that $\aleph$ does over the vertical direction (5 pixels) reduces the impact of the speckle noise in $\ddot{I}$. Additionally, this averaging bridges the gaps in the areas where the epithelium layer data is missing due to the signal roll-off.

5.3.4 Motion Correction

Constructing a 3-D cornea from the series of OCT 2-D slices is essential for many applications including cornea thickness calculation for KC diagnosis [150]. Since it is very difficult to keep the patient's eye looking at the same exact spot during the imaging session, the constructed 3-D cornea is usually ragged and has a rough surface. Hence, a motion correction algorithm to fix for eye movement is needed. To that end, during patients imaging sessions in our experiments, a set of reference frames/images are captured. A reference frame is an image slice that is captured perpendicular to the scanning direction. So, if

the eye is being scanned by the laser beam vertically, a number of N reference frames I_n^R where $n = 1 \dots N$ are captured horizontally. These reference frames are basically used to align the whole captured vertical slices to compensate for the eye motion.

To correct for the eye motion, let ℓ be the spatial distance between successive slices. Let $y_i(x)$ be the final segmentation of the anterior layer of slice I_i. Also, let $y_n^R(x)$ be the final segmentation of the anterior layer of the reference frame I_n^R and let X be the frame width. Then, a motion corrected frame $\bar{I}_i$ could be constructed using

$$\bar{I}_i(x, y) = I_i(x - \frac{X}{2} + x_i^M, y - y_i(x_i^M) + y_n^R(x_i^M + \ell \cdot (i - \frac{U}{2}))) \tag{5.9}$$

where x_i^M is the x coordinate of the apex point in $y_i(x)$ and U is the total number of slices. Figure 5.10 shows the spatial locations (black bars) of the five reference frames, $I_{n=1\dots5}^R$, in one OCT image. The frames were, perpendicular to the scanning direction, captured at the spatial locations $x = 176, 264, 351, 439$ and 526. Figure 5.11 shows the surface elevation map of a human cornea before and after the proposed motion correction algorithm. It can be seen that the elevation map in (a), due to the eye motion, is ragged and irregular. After applying the motion correction algorithm, one can qualitatively see that resulted elevation map is more consistent and smooth. It is worth to mention here that only 400 out of 1000 frames were used to generate these elevation maps.

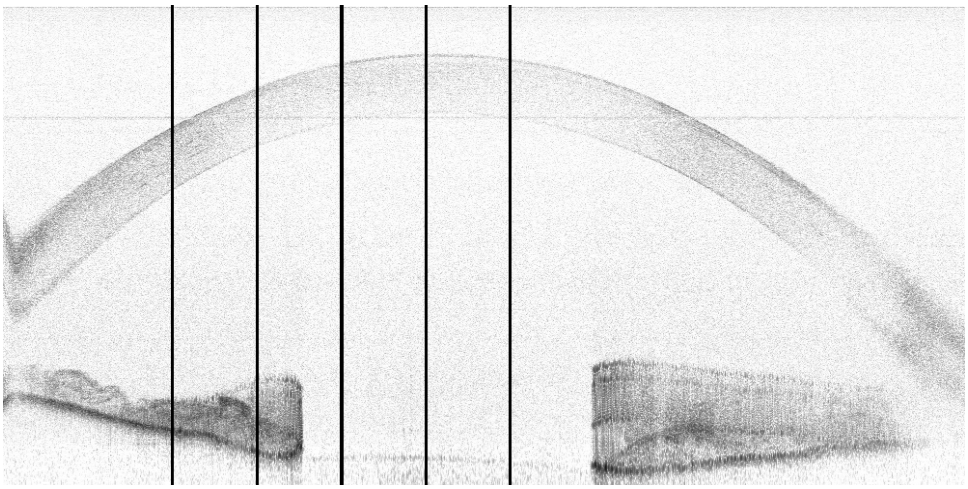
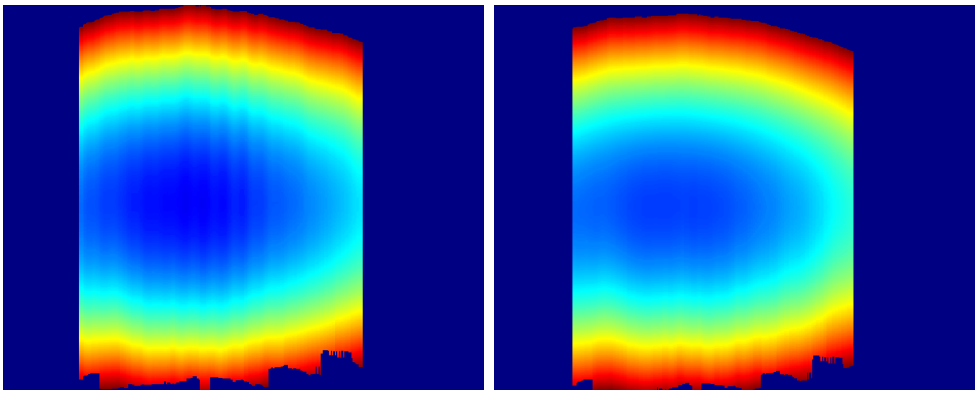


Figure 5.10: Spatial locations of the reference frames in one OCT slice. Reference frames are used to align OCT slices for motion correction.



Before motion correction After motion correction

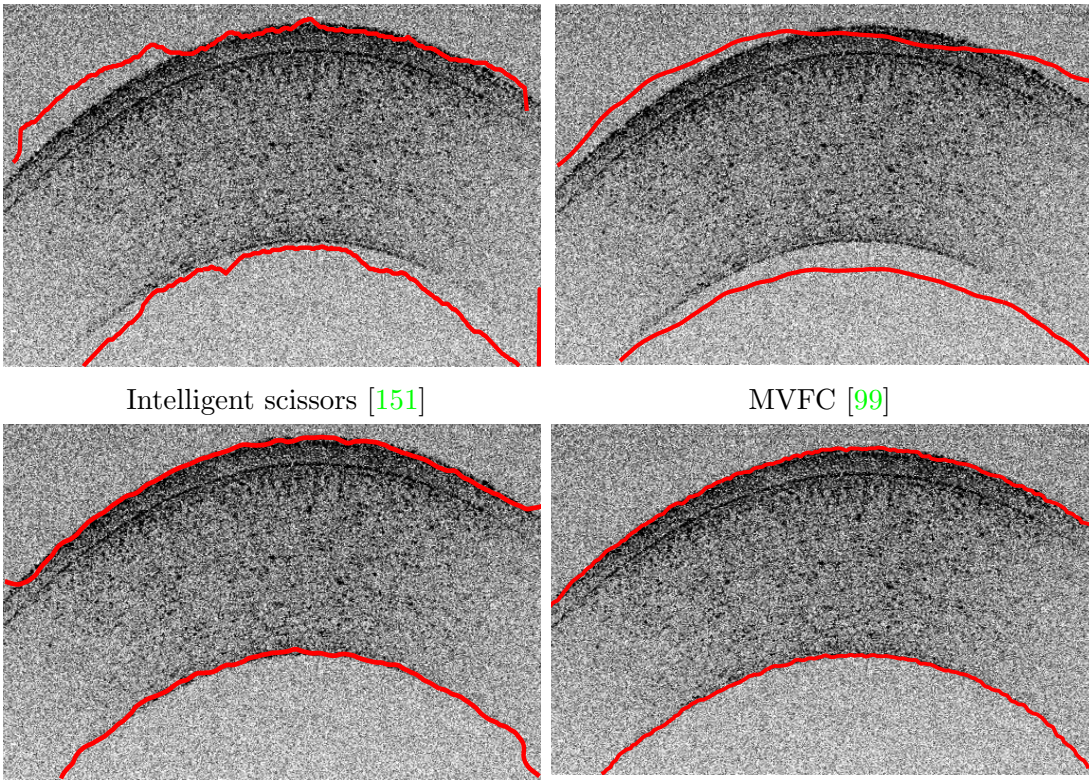
Figure 5.11: Cornea elevation map before and after applying the motion correction algorithm. It can be seen that the ragged surface of the uncorrected map, due to eye movement, is corrected after applying the motion correction method. Correcting for eye motion provides accurate measurements for calculating the cornea thickness.

5.4 Segmentation Results

During the course of this study, our proposed work, in this chapter and in Chapter 4, was applied to a wide range of healthy and diseased human corneas imaged using the new SS-OCT modality. Moreover, 10 sets (each set consists of 1000 images) of different contact lenses, with known thickness and curvature parameters, were used to verify the accuracy of the proposed model (MDVFC) compared to other imaging modalities segmentation (e.g. pentacam) [117]. Our model was also compared to the state-of-the-art multi-scale active contour based methods such as Multi-scale Tensor Vector Field (MTVF) [10] and Multi-scale Vector Field Convolution (MVFC) [99], as well as a classical method used for medical image segmentation, namely Intelligent Scissors(IS) [151]. The results we got, in terms of initialization and segmentation accuracy, prove the superiority of the proposed model compared to the state-of-the-art and classical methods.

To illustrate the effectiveness of the MDVFC model, Figure 5.12 shows the performance of the model, compared to the other methods, without the preprocessing steps of sections 5.3.1 and 5.3.2. Hence, the same initialization is given to all methods without and signal enhancement or noise reduction. The reason why the figure was intentionally restricted to only the upper part of the cornea is to have a relatively ideal environment (strong signal where upper and lower cornea boundaries are distinguishable) for all methods to work. Later in this section, our proposed model will be used to segment the whole cornea with the big challenge of the signal roll-off at cornea peripherals. Regardless of the fact that both upper and lower edges have strong signals, the competitor methods fail to segment this small part of the cornea correctly. Our proposed model, on the other hand, demonstrates a higher level of accuracy compared to aforementioned methods. Regarding the cornea peripherals and, again, with the relatively strong signal, the figure shows also that all other methods suffer from leakage at the peripherals, while the proposed method achieves a high accuracy result for the whole cornea.

For quantitative assessment and to ensure accurate comparisons, two segmentation metrics are used; namely the Jaccard index [152] and the average shortest distance measure (ASD) [24]. For Jaccard index calculation, ground truth images of the middle 400 frames of human cornea and contact lens sets were manually segmented to compare our results against. Table 5.1 quantitatively compares our proposed method to the same methods in Figure 5.12 using the aforementioned measures, and convincingly demonstrates the superiority of the proposed method over the others under high noise levels. It can be seen from the table that our method achieves the lowest ASD measure (the ideal case happens when ASD=0) and the highest Jaccard measure(the ideal case happens when Jaccard=1). The



Intelligent scissors [151] MVFC [99]

MTVF [10] MDVFC

Figure 5.12: Final segmentation example of MDVFC, IS, MVFC and MTVF. The MDVFC method outperforms all other methods.

Measurement	IS	MVFC	MTVF	Proposed
ASD	17.609 ± 3.68	18.176 ± 5.31	10.515 ± 2.31	**5.910** ± 1.02
Jaccard	0.732 ± 0.21	0.698 ± 0.29	0.8673 ± 0.15	**0.909** ± 0.09

Table 5.1: Quantitative performance comparisons. The proposed method outperforms all others using both the ASD metric, having an ideal value of zero, and the Jaccard metric, having an ideal value of one.

other methods provide lower levels of accuracy for the two metrics. It worth mentioning here that the comparison results of our method to any other method is based on the code provided by the authors of the competing method.

Since being able to image the entire cornea with high resolution is a key characteristic of the developed SS-OCT modality, our segmentation method is tested using image sets scanned by this modality. More than 15 data sets for healthy and KC corneas, plus 10 sets of contact lenses are segmented using the proposed method for elevation map and thickness calculations, with each set having 1000 images. However, because cornea/lens data appears only in the middle of each set, only the middle 400 images of each set are segmented. The segmentation results for the all sets reflect the superiority of our proposed method over the other methods. Figure 5.13 shows the segmentation results of our method using two entire KC corneas from different sets. Our method is capable of segmenting the anterior, posterior and the epithelium layers accurately. It can be seen from the figure that the high noise level does not prevent our method form correctly segment the 3 layers. Another point to note in this figure is that the GHT initialization module of the proposed method is not affected by the non corneal data on both sides of the image and is able to start the model relatively close to the actual cornea, and hence the final results are accurate. Moreover, the figure demonstrates the ability of our method to correctly detect the anterior and posterior layers of the cornea near the peripherals, thanks to the spatial variant kernel h. Also, the figure reflects the ability of our proposed method to correctly detect and accommodate the deformation happened in the epithelium layer due to the illness. Finally, one can note that due to the hierarchical nature of the proposed method and the adaptive kernel size discussed in Chapter 4, our method is able to correctly and accurately segment the three layers in the two images without being caught by the speckle noise.

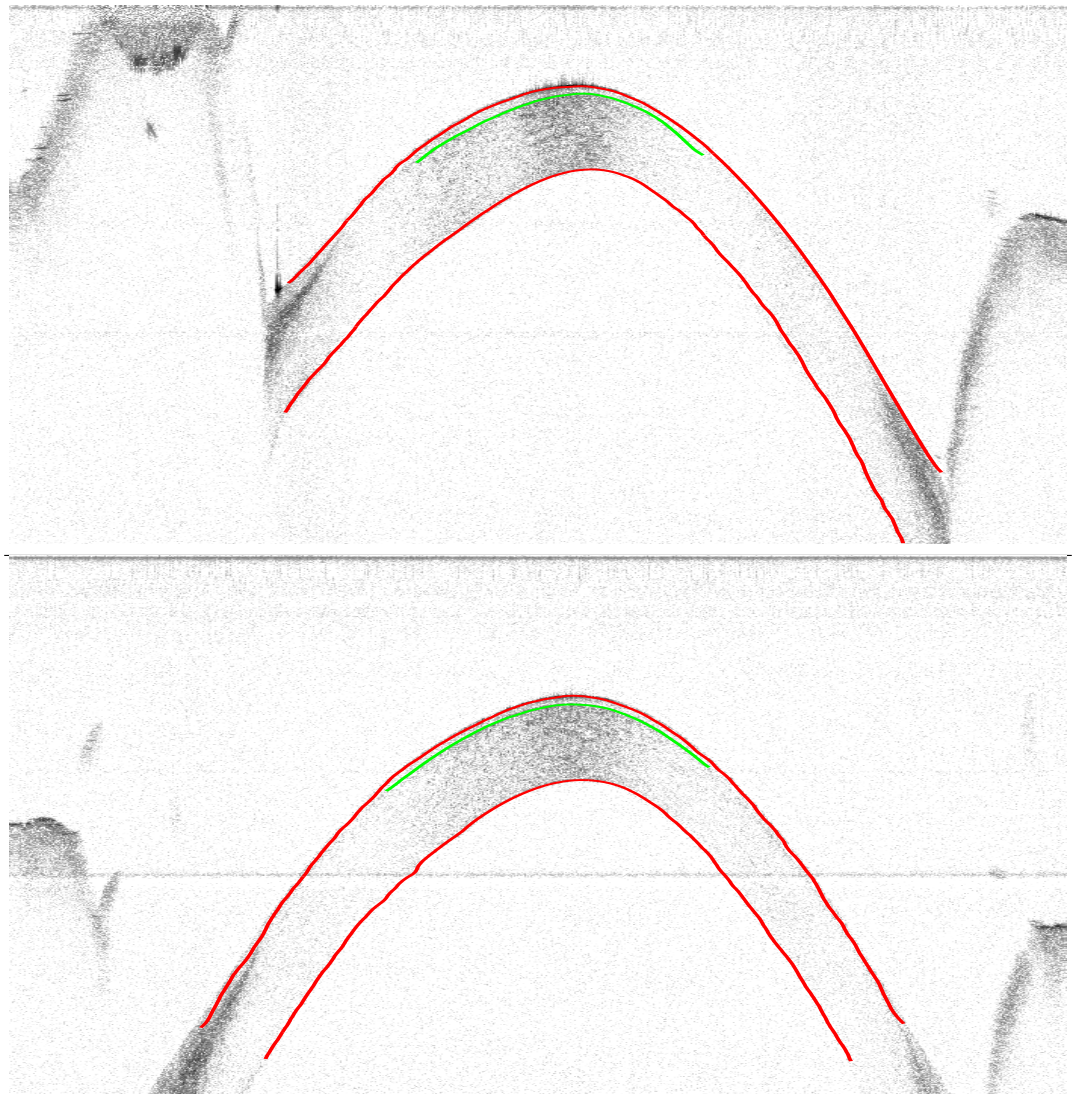


Figure 5.13: Illustrative examples of anterior, posterior (red) and epithelium layers (green) segmentation for different KC eyes. The proposed method is able to accurately segment the 3 layers under the high noise condition of the underlying image sets.

Chapter 6

Conclusions

In my thesis research, two models for improving the the external force field of multi-resolution active contours are proposed. After highlighting the limitations of classical active contour segmentation approaches, Chapter 3 proposed the usage of the over-complete multiresolution transforms (namely, the undecimated wavelet transform) to build up the external force field of the active contour. The wavelet coefficients are first normalized to find a direct relation between the amplitude of the wavelet coefficients in different resolutions. Then, a resolution-based favoring scheme of the gradient maps of each layer is used to simultaneously build up the external force of the active contour. The key features of noise robustness and edge preserving of the UDWT in addition to coarser gradient favoring make the proposed method outperforms classical and state-of-the-art methods.

Incorporating directional information in the external force of the active contour is addressed in Chapter 4. The directional information revealed by the (DT $\mathbb{C}$WT) is utilized to generate a steerable set of kernels to be used for external force generation. The directional information of each pixel as well as the local directional statistics are used pick up the appropriate kernel to calculate the force vector at each pixel. Incorporating the directional information during the process of external force generation makes the proposed model able to segment objects with complex shapes in noisy images. Experimental results using synthetic and real images show the superiority of the proposed method over classical and stat-of-the-art methods.

OCT corneal segmentation is proposed in Chapter 5. Basically, the proposed method of Chapter 4 is used to segment the upper and lower layers of the cornea. Moreover, Chapter 5 proposed a signal enhancement approach, based on a spatially varying kernel, to overcome the signal roll-off and the low SNR problems of OCT modalities. Also, the

chapter proposed a solution to the problem of automatic active contour initialization in corneal OCT images using the generalized Hough transform. Chapter 5 also addressed two other OCT segmentation-related problems. First, the calculation of the epithelium layer thickness. Second, the motion correction. The proposed model is tested using healthy and KC corneas as well as contact lenses. The comparisons with other methods show that our proposed method is able to give better results for all images.

The work in this thesis results a number of general outcomes and directions of significance. One main outcome is the advances achieved by using overcomplete wavelet transforms for generating the external force of the active contour. Having the image salient features localized at the coarse scales of the transform gives the active contour the well-desired feature of converging to the correct boundaries in noisy environments. Moreover, favoring the image features at coarse scales gives the active contour more robustness against noise. The question about the best way to favor the coarse scales is touched during this research, however, a more detailed research about the nature of this favoring is one of the main future extensions of this work. Also, All experimental results in this thesis use up to 8 resolutions of the wavelet transform. Though the improvement in the segmentation metrics after the fifth resolutions is not big, the questions of, up to how many scales of the transform should we use? and what is the relation between the number of scales and the noise level of the underlying image? remain as future extensions to the work of this thesis.

Convergence improvement of the active contour due the the incorporation of directional information into the external force field is another main outcome of this thesis. The incorporation of the directional information revealed by the dual tree complex wavelet transform into the external force field gives the active contour more resistance to noise and ability to capture fine details in the image. The selection of the dual tree complex wavelet transform as the tool to learn the directional information of the image meant to keep the calculation complexity low. Also, the near shift invariance property of the dual tree complex transform is another reason why we use this specific transform. However, there are other redundant steerable wavelets, which give a richer directional information such as Gabor wavelets, could be used to learn the directional information. One main future direction to the work in this thesis is to look deeply at the aspect of learning the directional information for better active contour segmentation results.

Finally, the work in this thesis is mainly applied to the corneal OCT images. However, with the wide spread of using different medical imaging modalities, we are aiming to expand the application of this work to other imaging technologies such as X-rays, ultrasounds and magnetic resonance imaging (MRI).

References

[1] H. Wu, V. Appia, and A. Yezzi, "Numerical conditioning problems and solutions for nonparametric i.i.d. statistical active contours," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, pp. 1298–1311, June 2013.

[2] E. Mylona, M. Savelonas, and D. Maroulis, "Automated parameterization of active contours: A brief survey," in *Signal Processing and Information Technology(ISSPIT), 2013 IEEE International Symposium on*, pp. 344–349, Dec 2013.

[3] L. Brechet, M.-F. Lucas, C. Doncarli, and D. Farina, "Compression of Biomedical Signals With Mother Wavelet Optimization and Best-Basis Wavelet Packet Selection," *IEEE Transactions on Biomedical Engineering*, vol. 54, pp. 2186–2192, Dec. 2007.

[4] G. Quellec, M. Lamard, G. Cazuguel, B. Cochener, and C. Roux, "Adaptive Nonseparable Wavelet Transform via Lifting and its Application to Content-Based Image Retrieval," *IEEE Transactions on Image Processing,*, vol. 19, pp. 25–35, Jan. 2010.

[5] L. Zhang, P. Bao, and X. Wu, "Multiscale LMMSE-based Image Denoising with Optimal Wavelet Selection," *IEEE Transactions on Circuits and Systems for Video Technology,*, vol. 15, pp. 469–481, Apr. 2005.

[6] J.-L. Starck, J. Fadili, and F. Murtagh, "The Undecimated Wavelet Decomposition and its Reconstruction," *IEEE Transactions on Image Processing*, vol. 16, pp. 297–309, Feb. 2007.

[7] C. Xu and J. Prince, "Snakes, Shapes, and Gradient Vector Flow," *IEEE Transactions on Image Processing*, vol. 7, pp. 359–369, Mar. 1998.

[8] B. Li and S. Acton, "Active Contour External Force Using Vector Field Convolution for Image Segmentation," *IEEE Transactions on Image Processing*, vol. 16, pp. 2096–2106, Aug. 2007.

[9] J.-L. Starck, J. Fadili, and F. Murtagh, "The undecimated wavelet decomposition and its reconstruction," *Image Processing, IEEE Transactions on*, vol. 16, pp. 297 –309, Feb. 2007.

[10] A. Kumar, A. Wong, P. Fieguth, and D. Clausi, "Multi-scale Tensor Vector Field Active Contours," in *19th IEEE International Conference on Image Processing (ICIP)*, pp. 2833–2836, Sept. 2011.

[11] G. Sundaramoorthi, A. Yezzi, and A. Mennucci, "Coarse-to-Fine Segmentation and Tracking Using Sobolev Active Contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 851–864, May 2008.

[12] N. Kingsbury, "The dual-tree complex wavelet transform: A new technique for shift invariance and directional filters," in *IEEE Digital Signal Processing Workshop*, pp. 319–322, 1998.

[13] B. Peng, L. Zhang, and D. Zhang, "Automatic Image Segmentation by Dynamic Region Merging," *IEEE Transactions on Image Processing*, vol. 20, pp. 3592–3605, Dec. 2011.

[14] C. Li, R. Huang, Z. Ding, J. Gatenby, D. Metaxas, and J. Gore, "A Level Set Method for Image Segmentation in the Presence of Intensity Inhomogeneities With Application to MRI," *IEEE Transactions on Image Processing*, vol. 20, pp. 2007–2016, Jul. 2011.

[15] J. Da-Rugna, G. Chareyron, and H. Konik, "About Segmentation Step in Content-based Image Retrieval Systems," in *World Congress on Engineering and Computer Science*, pp. 550–554, IAENG, 2011.

[16] L. Yi, G. Zhang, and Z. Wu, "A Scale-Synthesis Method for High Spatial Resolution Remote Sensing Image Segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, pp. 4062–4070, Oct. 2012.

[17] G. RC and W. RE, *Digital Image Processing*. Pearson Prentice Hall, 2002.

[18] J. Huart and P. Bertolino, "Similarity-based and Perception-based Image Segmentation," in *IEEE International Conference on Image Processing (ICIP)*, vol. 3, pp. 1148–1151, Sept. 2005.

[19] J. du Buf, M. Kardan, and M. Spann, "Texture Feature Performance for Image Segmentation," *Pattern Recognition*, vol. 23, no. 3, pp. 291–309, 1990.

[20] X. Liu and D. Wang, "Image and Texture Segmentation Using Local Spectral Histograms," *IEEE Transactions on Image Processing*, vol. 15, pp. 3066–3077, Oct. 2006.

[21] E. Borenstein and S. Ullman, "Class-specific, Top-down Segmentation," *7th European Conference on Computer Vision (ECCV)*, pp. 639–641, 2002.

[22] E. Borenstein and S. Ullman, "Combined Top-Down/Bottom-Up Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 2109–2125, Dec. 2008.

[23] L. Liu and S. Sclaroff, "Region Segmentation via Deformable Model-guided Split and Merge," in *Eighth IEEE International Conference on Computer Vision (ICCV)*, vol. 1, pp. 98–104, 2001.

[24] A. Mishra, P. Fieguth, and D. Clausi, "Decoupled Active Contour (DAC) for Boundary Detection," *IEEE Transactions onPattern Analysis and Machine Intelligence*, vol. 33, pp. 310–324, Feb. 2011.

[25] J. Winn and N. Jojic, "Locus: Learning Object Classes With Unsupervised Segmentation," in *Tenth IEEE International Conference on Computer Vision (ICCV)*, vol. 1, pp. 756–763, IEEE, 2005.

[26] H. Zhang, J. E. Fritts, and S. A. Goldman, "Image Segmentation Evaluation: A survey of Unsupervised Methods," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 260–280, 2008.

[27] E. D. Gelasca, T. Ebrahimi, M. C. Q. Farias, M. Carli, and S. K. Mitra, "Towards Perceptually Driven Segmentation Evaluation Metrics," in *Computer Vision And Pattern Recognition Conference*, 2004.

[28] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward Objective Evaluation of Image Segmentation Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 929–944, Jun. 2007.

[29] P. Correia and F. Pereira, "Objective Evaluation of Relative Segmentation Quality," in *International Conference on Image Processing*, vol. 1, pp. 308–311, 2000.

[30] P. L. Correia and F. Pereira, "Stand-alone Objective Segmentation Quality Evaluation," *EURASIP J. Appl. Signal Process.*, vol. 2002, pp. 389–400, Jan. 2002.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Natural Images and Its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," in *8th International Conference on Computer Vision*, pp. 416–423, 2001.

[32] P. Correia and F. Pereira, "Objective Evaluation of Video Segmentation Quality," *IEEE Transactions on Image Processing*, vol. 12, pp. 186–200, Feb. 2003.

[33] K. Fu and J. Mui, "A survey on Image Segmentation," *Pattern recognition*, vol. 13, no. 1, pp. 3–16, 1981.

[34] L. Cao, Z. Shi, and E. Cheng, "Fast Automatic Multilevel Thresholding Method," *Electronics Letters*, vol. 38, pp. 868–870, Aug. 2002.

[35] C.-I. Chang, Y. Du, J. Wang, S.-M. Guo, and P. Thouin, "Survey and Comparative Analysis of Entropy and Relative Entropy Thresholding Techniques," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 153, pp. 837–850, Dec. 2006.

[36] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man. and Cybernetics,*, vol. 9, pp. 62–66, Jan. 1979.

[37] L. Jianzhuang, L. Wenqing, and T. Yupeng, "Automatic Thresholding of Gray-level Pictures Using Two-dimension Otsu Method," in *International Conference on Circuits and Systems*, vol. 1, pp. 325–327, Jun. 1991.

[38] C. Yu, C. Dian-ren, Y. Xu, and L. Yang, "Fast Two-Dimensional Otsu's Thresholding Method Based on Integral Image," in *International Conference on Multimedia Technology (ICMT)*, pp. 1–4, Oct. 2010.

[39] J. Tilton, "Analysis of Hierarchically Related Image Segmentations," in *IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, pp. 60–69, Oct. 2003.

[40] J. Tilton, Y. Tarabalka, P. Montesano, and E. Gofman, "Best Merge Region-Growing Segmentation With Integrated Nonadjacent Region Object Aggregation," *IEEE Transactions on Geoscience and Remote Sensing*.

[41] S. L. Horowitz and T. Pavlidis, "Picture Segmentation by a Directed Split-and-Merge Procedure," *Proceedings of the 2nd International Joint Conference on Pattern Recognition, Copenhagen, Denmark*, pp. 424–433, 1974.

[42] A. Faruquzzaman, N. Paiker, J. Arafat, Z. Karim, and M. Ameer Ali, "Object Segmentation Based on Split and Merge Algorithm," in *IEEE Region 10 Conference (TENCON)*, pp. 1–4, Nov. 2008.

[43] H. Digabel and C. Lantuejoul, "Iterative Algorithms," in *Actes du Second Symposium Europeen d'Analyse Quantitative des Microstructures en Sciences des Mateeriaux, Biologie et Meedecine*, pp. 85–99, Oct. 1977.

[44] J. B. T. M. Roerdink and A. Meijster, "The Watershed Transform: Definitions, Algorithms and Parallelization Strategies," *Fundamenta Informaticae*, vol. 41, pp. 187–228, Jan. 2000.

[45] S. Beucher and F. Meyer, "The Morphological Approach to Segmentation: The Watershed Transformation. Mathematical Morphology in Image Processing," *Optical Engineering*, vol. 34, pp. 433–481, 1993.

[46] S. Beucher, "Watershed, Hierarchical Segmentation and Waterfall Algorithm," *Mathematical morphology and its applications to image processing*, pp. 69–76, 1994.

[47] B. Kurt and M. Gokmen, "Goal Oriented Edge Detection," in *23rd International Symposium on Computer and Information Sciences (ISCIS)*, pp. 1–5, Oct. 2008.

[48] P. Fieguth, *Statistical Image Processing and Multidimensional Modeling*. Springer, 2010.

[49] S. Konishi, A. L. Yuille, J. Coughlan, and S. C. Zhu, "Fundamental Bounds on Edge Detection: An Information Theoretic Evaluation of Different Edge Cues," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 573–579.

[50] S. Joshi and R. Koju, "Study and comparison of edge detection algorithms," in *Internet (AH-ICI), 2012 Third Asian Himalayas International Conference on*, pp. 1–5, Nov 2012.

[51] S. Wan, F. Yang, and M. He, "Gradient-threshold Edge Detection Based on Perceptually Adaptive Threshold Selection," in *3rd IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 999–1002, Jun. 2008.

[52] F. Yang, S. Wan, and Y. Chang, "Improved Method for Gradient-Threshold Edge Detector Based on HVS," in *Computational Intelligence and Security* (Y. Hao, J. Liu, Y. Wang, Y.-m. Cheung, H. Yin, L. Jiao, J. Ma, and Y.-C. Jiao, eds.), vol. 3801 of *Lecture Notes in Computer Science*, pp. 1051–1056, Springer Berlin Heidelberg, 2005.

[53] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679–698, Nov. 1986.

[54] B. Wang and S. Fan, "An Improved CANNY Edge Detection Algorithm," in *Second International Workshop on Computer Science and Engineering (WCSE)*, vol. 1, pp. 497–500, Oct. 2009.

[55] S. Guillon and M. Donias, "Directional second order derivatives: Application to edge and corner detection," in *Signal Processing Conference (EUSIPCO 1998), 9th European*, pp. 1–4, Sept 1998.

[56] S. Coleman, B. Scotney, and M. Herron, "An empirical performance evaluation technique for discrete second derivative edge detectors," in *Image Analysis and Processing, 2003.Proceedings. 12th International Conference on*, pp. 594–599, Sept 2003.

[57] D. Marr and E. Hildreth, "Theory of Edge Detection," *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 207, pp. 187–217, Feb. 1980.

[58] G. Cook and E. Delp, "Multiresolution Sequential Edge Linking," in *International Conference on Image Processing*, vol. 1, pp. 41–44, Oct. 1995.

[59] P. Hough, "Method and Means for Recognizing Complex Patterns." U.S. Patent 3.069.654, Dec. 1962.

[60] O. Barinova, V. Lempitsky, and P. Kohli, "On Detection of Multiple Object Instances Using Hough Transforms," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2233–2240, Jun. 2010.

[61] W.-C. Li and D.-M. Tsai, "Defect Inspection in Low-Contrast LCD Images Using Hough Transform-Based Nonstationary Line Detection," *IEEE Transactions on Industrial Informatics*, vol. 7, pp. 136–147, Feb. 2011.

[62] Q. Zhou, L. Zeng, and S. Zhou, "Statistical Detection of Defect Patterns Using Hough Transform," *IEEE Transactions on Semiconductor Manufacturing*, vol. 23, pp. 370–380, Aug. 2010.

[63] P. Kultanen, L. Xu, and E. Oja, "Randomized Hough transform (RHT)," in *10th International Conference on Pattern Recognition*, vol. 1, pp. 631–635, Jun. 1990.

[64] H. Li, M. A. Lavin, and R. J. L. Master, "Fast Hough Transform: A Hierarchical Approach," *Computer Vision, Graphics, and Image Processing*, vol. 36, no. 2, pp. 139–161, 1986.

[65] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.

[66] P. Burt and E. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Transactions on Communications*, vol. 31, pp. 532–540, Apr. 1983.

[67] A. Swilem, M. Girgis, E. Zanaty, and R. Ramadan, "An Algorithm for Mean Pyramids Vector Quantization Using Hadamard Transform and Tree Structure," in *16th IEEE International Conference on Image Processing (ICIP)*, pp. 1929–1932, Nov. 2009.

[68] M. Suplata and R. Ravas, "Gaussian Pyramid Based Acceleration of Optical Flow Reconstruction," in *19th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pp. 169–171, Apr. 2009.

[69] S. Dube and L. Hong, "Adaptive image pyramid based compression algorithm," in *Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1045–1048, Nov. 2000.

[70] M. Unser, "An Improved Least Squares Laplacian Pyramid for Image Compression," *Signal Process.*, vol. 27, pp. 187–203, May 1992.

[71] J. Morlet, G. Arens, E. Fourgeau, and D. Giard, "Wave Propagation and Sampling Theory : Part I : Complex Signal and Scattering Multilayered - Part II : Sampling Theory and Complex Waves," *Geophysics*, vol. 47, pp. 203–236, 1982.

[72] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," *Mathematische Annalen*, vol. 69, pp. 331–371, 1910.

[73] A. Ohkubo, S. Moro, and T. Matsumoto, "A Method for Circuit Analysis Using Haar Wavelet Transform," in *47th Midwest Symposium on Circuits and Systems (MWSCAS)*, vol. 3, pp. 399–402, Jul. 2004.

[74] X. Wang, "Moving Window-based Double Haar Wavelet Transform for Image Processing," *IEEE Transactions on Image Processing*, vol. 15, pp. 2771–2779, Sept. 2006.

[75] D. D. lL. Starck, M. Elad, "Redundant Multiscale Transforms and Their Application for Morphological Component Analysis," *Advances in Imaging and Electron Physics Series*, vol. 132, 2004.

[76] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *International Journal of Computer Vision*, vol. 1, pp. 321–331, 1988.

[77] C. Ünsalan and A. Erçil, "Conversions Between Parametric and Implicit Forms Using Polar/Spherical Coordinate Representations," *Computer Vision and Image Understanding*, vol. 81, no. 1, pp. 1–25, 2001.

[78] A. Kumar, A. Wong, A. Mishra, D. Clausi, and P. Fieguth, "Tensor Vector Field Based Active Contours," in *18th IEEE International Conference on Image Processing (ICIP)*, pp. 2833–2836, Sept. 2011.

[79] L. D. Cohen, "On Active Contour Models and Balloons," *CVGIP: Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991.

[80] C. Xu and J. L. Prince, "Generalized Gradient Vector Flow External Forces for Active Contours," *Signal Processing*, vol. 71, pp. 131–139, 1998.

[81] M. Charfi, "Using The GGVF for Automatic Initialization and Splitting Snake Model," in *5th International Symposium on I/V Communications and Mobile Network (ISVC)*, pp. 1–4, Oct. 2010.

[82] H. Kim, H. Kim, C. Ahn, G. Jeong, M. Kim, and K. Sun, "Vessel Boundary Detection for its 3D Reconstruction by Using a Deformable Model (GVF Snake)," in *27th Annual International Conference of the Engineering in Medicine and Biology Society (IEEE-EMBS)*, pp. 3440–3443, Jan. 2005.

[83] C.-C. Liu, C.-Y. Tsai, and S.-S. Yu, "Breast Extraction Scheme for Mammograms Using GVF-Snake," in *International Symposium on Computer Communication Control and Automation (3CA)*, vol. 1, pp. 248–251, May 2010.

[84] S. Osher and J. A. Sethian, "Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *JOURNAL OF COMPUTATIONAL PHYSICS*, vol. 79, no. 1, pp. 12–49, 1988.

[85] V. Caselles, "Geometric Models for Active Contours," in *International Conference on Image Processing*, vol. 3, pp. 9–12, Oct. 1995.

[86] R. Malladi, J. Sethian, and B. Vemuri, "Shape Modeling With Front Propagation: a Level Set Approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 158–175, Feb. 1995.

[87] T. Chan and L. Vese, "Active Contours Without Edges," *IEEE Transactions on Image Processing*, vol. 10, pp. 266–277, Feb 2001.

[88] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum, "A Geometric Snake Model for Segmentation of Medical Imagery," *IEEE Transactions on Medical Imaging*, pp. 199–209, 1997.

[89] Z. Li, Q. Liu, and H. Lu, "A Geometric Active Contour Framework using Multi-Cue and Local Feature," in *18th International Conference on Pattern Recognition (ICPR)*, vol. 2, pp. 113–116, 2006.

[90] Y. Boykov and V. Kolmogorov, "Computing Geodesics and Minimal Surfaces via Graph Cuts," in *Ninth IEEE International Conference on Computer Vision*, vol. 1, pp. 26–33, Oct. 2003.

[91] N. El-Zehiry and A. Elmaghraby, "A Graph Cut Based Active Contour for Multiphase Image Segmentation," in *15th IEEE International Conference on Image Processing (ICIP)*, pp. 3188–3191, Oct. 2008.

[92] P. yan Tsang, "Multi-resolution Image Segmentation using Geometric Active Contours," 2004.

[93] A. Gawish and P. Fieguth, "External forces for active contours using the undecimated wavelet transform," in *Accepted at: Image Processing (ICIP), 2015 IEEE International Conference on*, Sept 2015.

[94] H. Li and A. Yezzi, "Local or global minima: Flexible dual-front active contours," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 1–14, Jan 2007.

[95] A. Hamou and M. El-Sakka, "Srad and optical flow based external energy for echocardiograms with primitive shape priors," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 2613–2616, Nov 2009.

[96] G. Zhu and S.-X. Guo, "Image-fusion-based multi-resolution active contour model," *Optik - International Journal for Light and Electron Optics*, vol. 125, no. 17, pp. 4955 – 4957, 2014.

[97] W. Wang, J. Qin, Y.-P. Chui, and P.-A. Heng, "A multiresolution framework for ultrasound image segmentation by combinative active contours," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pp. 1144–1147, July 2013.

[98] B. Leroy, I. Herlin, and L. Cohen, "Multi-resolution Algorithms for Active Contour Models," in *ICAOS '96* (M.-O. Berger, R. Deriche, I. Herlin, J. Jaffr, and J.-M. Morel, eds.), vol. 219 of *Lecture Notes in Control and Information Sciences*, pp. 58–65, Springer Berlin Heidelberg, 1996.

[99] C. Yang and S. Acton, " External Forces for Active Contours via Multi-scale Vector Field Convolution," in *19th IEEE International Conference on Image Processing (ICIP)*, pp. 2569–2572, Sept. 2012.

[100] C. Davatzikos, X. Tao, and D. Shen, "Hierarchical Active Shape Models, Using The Wavelet Transform," *IEEE Transactions on Medical Imaging*, vol. 22, pp. 414–423, Mar. 2003.

[101] H.-H. Wu, J.-C. Liu, and C. Chui, "A Wavelet-frame Based Image Force Model for Active Contouring Algorithms," *IEEE Transactions on Image Processing*, vol. 9, pp. 1983–1988, Nov. 2000.

[102] A. Ellmauthaler, C. L. Pagliari, and E. A. B. da Silva, "Multiscale Image Fusion Using the Undecimated Wavelet Transform With Spectral Factorization and Nonorthogonal Filter Banks," *IEEE Transactions on Image Processing,*, vol. 22, pp. 1005–1017, Mar. 2013.

[103] D. L. DONOHO and J. M. JOHNSTONE, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[104] L. Sendur and I. Selesnick, "Bivariate shrinkage with local variance estimation," *Signal Processing Letters, IEEE*, vol. 9, pp. 438–441, Dec 2002.

[105] L. Grigoriu and R. Zaciu, "The scale-based pyramid weighting of the wavelet coefficients in zerotree coding techniques," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 3, pp. III–213–16 vol.2, Sept 2003.

[106] V. Sethu, E. Ambikairajah, and L. Ge, "Selective weighting of undecimated wavelet coefficients for noise reduction in sar interferograms."

[107] S. Alpert, M. Galun, R. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.

[108] E. Mylona, M. Savelonas, and D. Maroulis, "Self-adjusted active contours using multi-directional texture cues," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pp. 3026–3030, Sept 2013.

[109] Z. Luo and J. Wu, "The integration of directional information and local region information for accurate image segmentation," *Pattern Recognition Letters*, vol. 32, no. 15, pp. 1990 – 1997, 2011.

[110] T. Celik and K.-K. Ma, "Multitemporal image change detection using undecimated discrete wavelet transform and active contours," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 49, pp. 706 –716, Feb. 2011.

[111] I. Voicu and M. Borda, "Image compression using complex wavelet transform," in *Computer as a Tool, 2005. EUROCON 2005.The International Conference on*, vol. 2, pp. 919–922, Nov 2005.

[112] T. Celik and T. Tjahjadi, "Multiscale texture classification using dual-tree complex wavelet transform," *Pattern Recognition Letters*, vol. Vol.30, pp. 331–339, February 2009.

[113] J. Liu and K. She, "Robust image watermarking using dual tree complex wavelet transform based on human visual system," in *Image Analysis and Signal Processing (IASP), 2010 International Conference on*, pp. 675–679, April 2010.

[114] R. Singh, R. Purwar, and N. Rajpal, "A better approach for object tracking using dual-tree complex wavelet transform," in *Image Information Processing (ICIIP), 2011 International Conference on*, pp. 1–5, Nov 2011.

[115] http://research.microsoft.com/en-us/um/people/jiansun/SalientObject/salient_object.htm.

[116] S. Marschall, A. Gawish, Y. Feng, L. Sorbara, P. Fieguth, and K. Bizheva, "In-vivo imaging of keratoconic corneas using high-speed high-resolution swept-source oct," in *Proc. SPIE 8802, Optical Coherence Tomography and Coherence Techniques VI*, vol. 8802, 2013.

[117] K. Carter, S. Marschall, A. Gawish, P. Fieguth, L. Sorbara, and K. Bizheva, "Accuracy evaluation of scleral lens thickness and radius of curvature using high-resolution sd- and ss-oct," in *Proc. SPIE*, vol. 8930, 2014.

[118] A. Gawish, P. Fieguth, S. Marschall, and K. Bizheva, "Undecimated hierarchical active contours for oct image segmentation," in *Image Processing (ICIP), 2014 IEEE International Conference on*, pp. 882–886, Oct 2014.

[119] E. B. Goldstein, *Sensation & Perception (7th ed.)*. Thompson Wadsworth, 2007.

[120] https://en.wikipedia.org/wiki/Cornea.

[121] http://www.medicatravel.co.uk/index.htm.

[122] H. S. Dua, L. A. Faraj, D. G. Said, T. Gray, and J. Lowe, "Human corneal anatomy redefined: A novel pre-descemet's layer (dua's layer)," *Ophthalmology*, vol. 120, no. 9, pp. 1778 – 1785, 2013.

[123] https://nei.nih.gov/health/cornealdisease.

[124] http://www.cxlusa.com/treatment-options.aspx.

[125] F. A. Fercher, C. K. Hitzenberger, W. Drexler, G. Kamp, and H. Sattmann, "In vivo optical coherence tomography," *Am. J. Ophthalmol*, vol. 116, pp. 113–114, July 1993.

[126] B. V. Ventura, H. V. M. Jr., N. Kara-Junior, and M. R. Santhiago, "Role of optical coherence tomography on corneal surface laser ablation," *Journal of Ophthalmology*, 2012.

[127] Y. S. Rabinowitz, "Tangential vs sagittal videokeratographs in the "early" detection of keratoconus.," *Journal of Ophthalmol*, vol. 122, no. 6, pp. 887–889, 1996.

[128] S. N. Rao, T. Raviv, P. A. Majmudar, and R. J. Epstein, "Role of orbscan ii in screening keratoconus suspects before refractive corneal surgery.," *Journal of Ophthalmology*, vol. 109, no. 9, pp. 1642–1646, 2002.

[129] C. Mazzotta, A. Balestrazzi, C. Traversi, S. Baiocchi, T. Caporossi, C. Tommasi, and A. Caporossi, "Treatment of progressive keratoconus by riboflavin-uva-induced cross-linking of corneal collagen: ultrastructural analysis by heidelberg retinal tomograph ii in vivo confocal microscopy in humans.," *Cornea*, vol. 26, no. 4, pp. 390–397, 2007.

[130] D. P. Pinero, A. B. Plaza, and J. L. Alio, "Anterior segment biometry with 2 imaging technologies: very-high-frequency ultrasound scanning versus optical coherence tomography.," *Journal of Cataract Refract Surgery.*, vol. 34, no. 1, pp. 95–102, 2008.

[131] V. Christopoulos, L. Kagemann, G. Wollstein, H. Ishikawa, M. L. Gabriele, M. Wojtkowski, V. Srinivasan, J. G. Fujimoto, J. S. Duker, D. K. Dhaliwal, and J. S. Schuman, "In vivo corneal high-speed, ultrahigh-resolution optical coherence tomography," *Arch Ophthalmol.*, vol. 125, no. 8, pp. 1027–1035, 2007.

[132] I. Grulkowski, M. Gora, M. Szkulmowski, I. Gorczynska, D. Szlag, S. Marcos, A. Kowalczyk, and M. Wojtkowski, "Anterior segment imaging with spectral oct system using a high-speed cmos camera," *Journal of Optics Express.*, vol. 17, no. 6, pp. 4842–4858, 2009.

[133] N. Hutchings, T. L. Simpson, C. Hyun, A. A. Moayed, S. Hariri, L. Sorbara, and K. Bizheva, "Swelling of the human cornea revealed by high speed, ultrahigh resolution optical coherence tomography," *Journal of Investigative Ophthalmology & Visual Science.*, vol. 51, no. 9, pp. 4579–4584, 2010.

[134] A. Karimi, A. Wong, and K. Bizheva, "Automated detection and cell density assessment of keratocytes in the human corneal stroma from ultrahigh resolution optical coherence tomograms," *Journal of Biomedical Optics Express.*, vol. 2, no. 10, pp. 2905–2916, 2011.

[135] P. Li, L. An, G. Lan, M. Johnstone, D. Malchow, and R. K. Wang, "Extended imaging depth to 12 mm for 1050-nm spectral domain optical coherence tomography for imaging the whole anterior segment of the human eye at 120-khz a-scan rate," *Journal of Biomedical Optics*, vol. 18, no. 1, pp. 16012–16012, 2013.

[136] M. Gora, K. Karnowski, M. Szkulmowski, B. J. Kaluzny, R. Huber, A. Kowalczyk, and M. Wojtkowski, "Ultra high-speed swept source oct imaging of the anterior segment of human eye at 200 khz with adjustable imaging range," *Journal of Optics Express.*, vol. 17, no. 17, pp. 14880–14894, 2009.

[137] B. Potsaid, B. Baumann, D. Huang, S. Barry, A. E. Cable, J. S. Schuman, J. S. Duker, and J. G. Fujimoto, "Ultrahigh speed 1050nm swept source /fourier domain oct retinal and anterior segment imaging at 100,000 to 400,000 axial scans per second," *Journal of Optics Express.*, vol. 18, no. 18, pp. 20029–20048, 2010.

[138] J. J. Gomez-Valverde, J. E. Ortuno, P. Guerra, B. Hermann, B. Zabihian, J. L. Rubio-Guivernau, A. Santos, W. Drexler, and M. J. Ledesma-Carbayo, "Evaluation of speckle reduction with denoising filtering in optical coherence tomography for dermatology," in *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*, pp. 494–497, April 2015.

[139] F. LaRocca, S. J. Chiu, R. P. McNabb, A. N. Kuo, J. A. Izatt, and S. Farsiu, "Robust automatic segmentation of corneal layer boundaries in sdoct images using graph theory and dynamic programming," *Biomed. Opt. Express*, vol. 2, pp. 1524–1538, Jun. 2011.

[140] M.-J. Chen, Y.-T. Liu, C.-C. Tsai, Y.-C. Chen, C.-K. Chou, and S.-M. Lee, "Relationship between central corneal thickness, refractive error, corneal curvature, anterior chamber depth and axial length," *Journal of the Chinese Medical Association*, vol. 72, no. 3, pp. 133 – 137, 2009.

[141] Z. Schlegel, T. Hoang-Xuan, and D. Gatinel, "Comparison of and correlation between anterior and posterior corneal elevation maps in normal eyes and keratoconus-suspect eyes," *Journal of Cataract and Refractive Surgery*, vol. 34, no. 5, pp. 789–795, 2008.

[142] J. Felix, P. Cortez, T. Cavalcante, A. Alexandria, and M. Holanda, "Autoin: Method of automatic initialization of active contours applied to lungs in ct images," *Latin America Transactions, IEEE (Revista IEEE America Latina)*, vol. 10, pp. 1954–1960, June 2012.

[143] D. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[144] N. Q. Ali, D. V. Patel, and C. N. J. McGhee, "Biomechanical responses of healthy and keratoconic corneas measured using a noncontact scheimpflug-based tonometer," *Investigative Ophthalmology & Visual Science.*, vol. 55, no. 6, pp. 3651–3659, 2014.

[145] R. L. Brautaset, M. Nilsson, W. L. Miller, N. E. Leach, J. H. Tukler, and J. P. G. Bergmanson, "Central and peripheral corneal thinning in keratoconus.," *Journal of Cornea and External Disease*, vol. 32, no. 3, pp. 257–261, 2013.

[146] W. Zhou and A. Stojanovic, "Comparison of corneal epithelial and stromal thickness distributions between eyes with keratoconus and healthy eyes with corneal astigmatism 2.0 d," *PLoS ONE*, vol. 9, no. 1, 2014.

[147] R. Yadav, R. Kottaiyan, K. Ahmad, and G. Yoon, "Epithelium and bowmans layer thickness and light scatter in keratoconic cornea evaluated using ultrahigh resolution optical coherence tomography," *Journal of Biomedical Optics*, vol. 17, no. 11, pp. 116010–116017, 2012.

[148] Y. Li, O. Tan, R. Brass, J. L. Weiss, and D. Huang, "Corneal epithelial thickness mapping by fourier-domain optical coherence tomography in normal and keratoconic eyes.," *Journal of Ophthalmology*, vol. 119, no. 12, pp. 2425 – 2433, 2010.

[149] D. Reinstein, T. Archer, M. Gobbe, R. Silverman, and D. Coleman, "Epithelial, stromal and corneal thickness in the keratoconic cornea: three-dimensional display

with artemis very high-frequency digital ultrasound.," *Journal of Refract Surgery*, 2010.

[150] "Automated 3d reconstruction and segmentation from optical coherence tomography," in *Computer Vision ECCV 2010* (K. Daniilidis, P. Maragos, and N. Paragios, eds.), vol. 6313 of *Lecture Notes in Computer Science*, 2010.

[151] E. N. Mortensen and W. A. Barrett, "Intelligent scissors for image composition," in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 191–198, 1995.

[152] P. Jaccard, "The distribution of the flora in the alpine zone.1," *New Phytologist*, vol. 11, no. 2, pp. 37–50, 1912.