THE UNIVERSITY OF
WARWICK

AUTHOR: **Ian McDonnell** DEGREE: **Ph.D.**

TITLE: **Object Segmentation from Low Depth of Field Images and Video Sequences**

DATE OF DEPOSIT: .....

I agree that this thesis shall be available in accordance with the regulations governing the University of Warwick theses.

I agree that the summary of this thesis may be submitted for publication.

I **agree** that the thesis may be photocopied (single copies for study purposes only).

Theses with no restriction on photocopying will also be made available to the British Library for microfilming. The British Library may supply copies to individuals or libraries. subject to a statement from them that the copy is supplied for non-publishing purposes. All copies supplied by the British Library will carry the following statement:

> "Attention is drawn to the fact that the copyright of this thesis rests with its author. This copy of the thesis has been supplied on the condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the thesis and no information derived from it may be published without the author's written consent."

AUTHOR'S SIGNATURE: .....

USER'S DECLARATION

1. I undertake not to quote or make use of any information from this thesis without making acknowledgement to the author.
2. I further undertake to allow no-one else to use this thesis while it is in my care.

DATE SIGNATURE ADDRESS

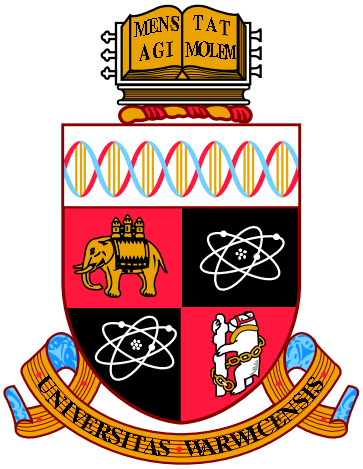
.....

.....

.....

.....

.....



Object Segmentation from Low Depth of Field Images and Video Sequences

by

Ian McDonnell

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

School of Engineering

June 2013

THE UNIVERSITY OF WARWICK

Contents

List of Tables

List of Figures

Acknowledgments

I would like to take this opportunity to thank my research supervisor Dr. Tardi Tjahjadi for his time, endless patience, and academic support towards this research. I am extremely grateful. I would also like to thank the UK Engineering and Physical Science Research Council for providing the studentship for this research and giving me the opportunity to work towards a PhD.

A PhD can often be a tough and isolating experience and so I extend my thanks to the fantastic groups of friends I have at Warwick and elsewhere. There are too many of you to name, but whether you provided a friendly ear, kept me active, made me laugh, danced with me, improved my mathematics knowledge, supported me or just shouted at me to 'get it done!', I appreciate you all so much!

Finally I would like to thank my parents and my brother for their ongoing support, both financially and emotionally during my time at Warwick - this thesis is dedicated to you.

Declarations

This thesis is submitted in partial fulfilment for the degree of Doctor of Philosophy under the regulations set out by the Graduate School at the University of Warwick. This thesis is solely composed of research undertaken by Ian McDonnell under the supervision of Dr. Tardi Tjahjadi. The research materials have not been submitted in any previous application for a higher degree. All sources of information are specifically acknowledged in the content.

Abstract

This thesis addresses the problem of autonomous object segmentation. To do so the proposed segementation method uses some prior information, namely that the image to be segmented will have a low depth of field and that the object of interest will be more in focus than the background. To differentiate the object from the background scene, a multiscale wavelet based assessment is proposed. The focus assessment is used to generate a focus intensity map, and a sparse fields level set implementation of active contours is used to segment the object of interest. The initial contour is generated using a grid based technique.

The method is extended to segment low depth of field video sequences with each successive initialisation for the active contours generated from the binary dilation of the previous frame's segmentation. Experimental results show good segmentations can be achieved with a variety of different images, video sequences, and objects, with no user interaction or input.

The method is applied to two different areas. In the first the segmentations are used to automatically generate trimaps for use with matting algorithms. In the second, the method is used as part of a shape from silhouettes 3D object reconstruction system, replacing the need for a constrained background when generating silhouettes. In addition, not using a thresholding to perform the silhouette segmentation allows for objects with dark components or areas to be segmented accurately. Some examples of 3D models generated using silhouettes are shown.

Abbreviations

2D Two-dimensional

3D Three-dimensional

BPT Binary partition tree

CG Conjugate Gradient

CoC Circle of confusion

DoF Depth of field

HOS Higher-order statistics

HR High resolution

IGC Interactive graph cuts

LR Low resolution

HVS Human vision system

ML Modified Laplacian

MVS Machine vision system

OoI Object of interest

SFM Sparse fields method

PSF Point spread function

SfS Shape from silhouettes

SML Sum modified Laplacian

Std Standard deviation

VH Visual Hull

Chapter 1

Introduction

Being able to extract an object of interest (OoI) from an image (referred to as object segmentation in this thesis) is important in a wide variety of computer vision applications, such as object recognition, but to do so without any user input is difficult due to wide varying scenes and image characteristics. Whilst an edge detection method can successfully extract contours from images, additional processing is required to determine which are object contours, which are background contours and which are caused by other image features such as textures or colour changes in an object or background.

Image segmentation methods aim to divide images into regions where pixels contain similar characteristics such as colour, intensity or texture. To aid this, several methods also utilise human annotations to the image. In the specific case of object segmentation, the objective is to produce a binary segmentation, i.e., the image is divided into two types of regions, background and object. Image segmentation is a popular field of research and numerous object segmentation algorithms have been proposed, many of which require some user input or a priori knowledge about the object to be segmented. It is widely accepted that it is difficult to produce a general autonomous algorithm suitable for all image types, but methods that require no user input can be applied to specific scenarios or scenes. For this thesis, the case of images with a low depth of field (DoF) is investigated, allowing the cue to be taken from the focus of pixels rather than relying on texture or colour.

DoF is the zone, or range of distances, in a given scene that appear to be in acceptably sharp focus. Although a camera lens will only be in critical focus for one point in the scene, a range of points behind and in front of this point will also appear sharp depending on the DoF. Thus in an image captured by a camera with a large DoF most of the scene will appear to be sharp (in focus), whereas in a low DoF

image parts of the scene closer to, or further away from, the lens than the point the camera has focused on will appear blurred (out of focus). Capturing images with a low DoF is a commonly used technique in photography as it emphasises the subject of a photograph, as well helping viewers understand the depth of particular objects in a scene.

To address the problem of autonomous object segmentation, this thesis proposes a method which combines a focus assessment of image pixels and an active contours algorithm to segment an OoI. The premise behind the proposed method is that the image background will not be as sharp as the OoI the camera has focused on. A focus assessment enables the method to differentiate between object and background contours, and thus extract the OoI.

Increasingly a low DoF is also being used in video sequences, in a range of situations from adverts and news broadcasting, to film and television programs. Using a low DoF emphasises the important part of a frame and prevents a cluttered background from detracting from the focus of a scene. The linked nature of video frames is utilised to expand the object segmentation method to provide fast and accurate segmentations for low DoF video sequences.

Digital matting addresses the problem of foreground estimation in images. Matting methods determine an opacity or alpha value for mixed or ambiguous pixels along an object's boundary. This allows for complex natural objects, the most difficult cases being those with hair or fur, to be composited onto new background. Typically matting methods make use of a user defined trimap - where the original scene is split into 3 segments; object, background and ambiguous. Object pixels are given an alpha value of 1 (opaque) and background pixels 0 (transparent). The matting method chosen then calculates the opacity within the ambiguous region based on a series of probabilities. The foreground element can then be composited into a new scene. The enveloping properties of the active contours algorithm used in the proposed object segmentation method mean that it can be adapted to automatically generate accurate trimaps for use in matting algorithms.

There are two general categories of 3-dimensional (3D) object reconstruction, active methods and passive methods. Active methods involve some form of interaction or scanning of the OoI whereas passive methods use only sensors (typically one or two image sensors in the visible range). Passive techniques have the advantage of being unintrusive to the OoI and the equipment involved is relatively inexpensive when compared to active methods. 3D object reconstruction from 2-dimensional (2D) images is intrinsically problematic as the information in one dimension is lost when a 3D scene is projected onto a 2D image. One passive 3D object reconstruction

technique is known as the shape from silhouettes (SfS) method. Images of the OoI are captured by a camera at numerous viewpoints. By segmenting the images and backprojecting the resulting silhouettes of the object, a visual hull representing the object's volume is created. The proposed object segmentation method is integrated into an existing SfS-based 3D object reconstruction system, removing the need for a bulky background in the image capture stage, and the need for user input to generate the silhouettes.

1.1 Contributions and Thesis Structure

The principal contributions of this thesis are as follows:

1. Evaluation and comparison of focus assessment methods;
2. Multiscale focus assessment of image pixels;
3. Unsupervised object segmentation from low depth of field images;
4. Unsupervised object segmentation from low depth of field video sequences;
5. Automatic trimap generation for matting algorithms and scene composition;
6. Automatic silhouette generation for 3D object reconstruction.

This thesis is concerned with an autonomous object segmentation algorithm and its applications, in particular to digital matting, and silhouette generation for an automatic 3D object reconstruction system. It focuses on the required image processing techniques of focus assessment and object segmentation. The thesis is organised into 8 Chapters. In each chapter, a review of related techniques are presented. The individual chapters of this thesis are structured as follows:

Chapter 2 provides an introduction to the concept of focus and its relationship with DoF. The problem of object segmentation is introduced and an overview of popular methods and techniques given.

Chapter 3 covers a range of existing techniques for assessing the focus values of image pixels. The performance of these techniques are evaluated and a modified multiscale focus assessment algorithm proposed.

Chapter 4 presents a new autonomous object segmentation method for use with low DoF images. Experimental results of this method are presented and the performance of this method is compared with other popular object segmentation algorithms.

Chapter 5 expands upon the algorithm presented in Chapter 4, and extracts the OoI from low DoF video sequences. Experimental results of this algorithm are presented, and the performance of this algorithm is compared with another related method.

Chapter 6 presents an application of the object segmentation method, applying it to automatically generate trimaps for use in matting algorithms to perform scene compositions, both in images and video sequences. Experimental results of the methods on a number of realistic scene superimpositions are presented.

Chapter 7 applies the object segmentation method to automatically generate silhouettes for use in an existing 3D object reconstruction system. A variety of 3D models generated using this method are shown. Finally, Chapter 8 concludes the thesis.

Chapter 2

Overview of Background Fundamentals

2.1 Introduction

The object segmentation method presented in this thesis is designed to work without user input on low depth of field (DoF) images, i.e., images where there is a focus differential between the object and background. This overview chapter looks at some of the fundamental topics involved in this area of research. The concept of focus and how it relates to low DoF images are explained. The fundamentals of image segmentation are also discussed and some of the main approaches and methods reviewed.

This overview chapter is organised as follows: Section 2.2 provides a definition of focus. The derivation of the Thin Lens Law and how this relates to defocused regions of an image are presented. The amount of image blurring is shown to be related to the distance in depth from the critical focus point. DoF is defined and the factors affecting it are discussed. Finally, some of the applications and uses of low DoF images are demonstrated. Section 2.3 provides a definition of image segmentation, whilst Section 2.4 discusses the specific case of object segmentation with Section 2.5 giving an overview of the popular methods and techniques.

2.2 Focus and Depth of Field

2.2.1 Focus

From a human perspective, focus is generally associated with sharpness - if an object or area is sharp it is considered to be in-focus, whilst if a region is blurred or fuzzy

it is considered to be out of focus, or defocused. In optics, focus is defined as the point at which light rays originating from a point on an object converge. Thus, if an image point is in-focus, light from the point will be well converged on the image plane, whereas light from defocused image points will not. This is illustrated in Figure 2.1, where the object point that is perfectly in focus is known as the point of critical focus (B).

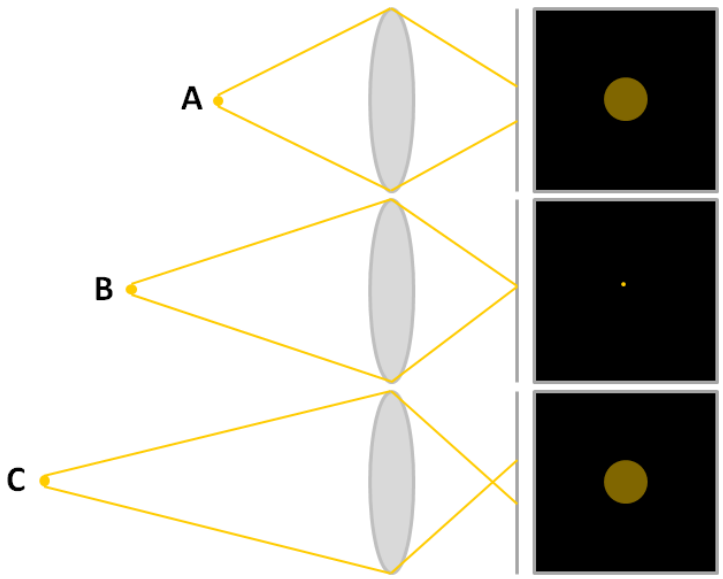


Figure 2.1: Light rays from a focused point in a given scene (B) converge on the image plane to form a point, whereas light rays from defocused points (A and C) do not converge and thus form a spot on the image plane.

2.2.2 Thin Lens Law

Modelling a camera as an image plane and a thin convex lens with a focal length f, the relationship between a focused point in a scene and the position of its focused point on the image plane can be derived by using the optical geometry shown in Figure 2.2. Two pairs triangles are identified as shown in Figure 2.3.

Using the law of similar triangles gives the following equations

$$\frac{I}{O} = \frac{v}{u} \,, \tag{2.1}$$

and

$$\frac{I}{O} = \frac{v - f}{f} \,, \tag{2.2}$$

where f is the focal length, I is the the image size, O is the object size and u and v

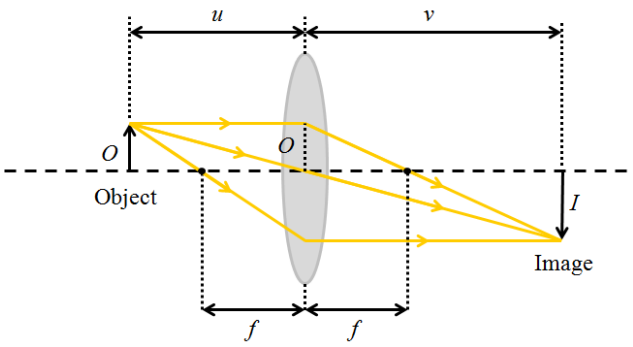
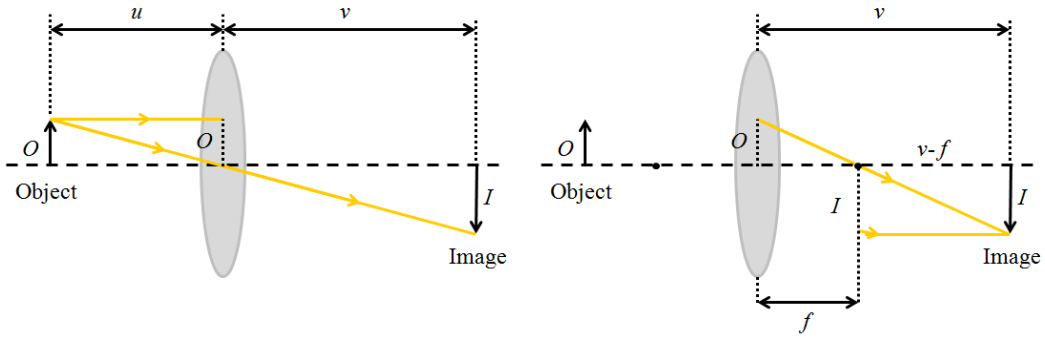


Figure 2.2: Lens system showing central, parallel and focal rays from focused object point and corresponding image, where f is the focal length, I is the the image size, O is the object size, u the distance from the lens to the object and v the distance from the lens to the image.



(a) (b)

Figure 2.3: Two similar triangles within the lens system shown in Figure 2.2.

are the distances from the lens to the object and image, respectively. Substituting Equation 2.1 in Equation 2.2 gives

$$\frac{u}{v} = \frac{f}{v-f} \ . \tag{2.3}$$

thus giving the well known thin lens formula for a focused point:

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \ . \tag{2.4}$$

2.2.3 Circle of Confusion

Consider the cases where an object point is either shallower or deeper than the distance u. As shown in Figure 2.1 the light will not converge to a single point on

the image plane, but instead upon an optical spot. This spot is known by a variety of names including 'circle of indistinctness', 'blur circle', 'blur spot' and 'circle of confusion'. For the purposes of this thesis it will be referred to as the the circle of confusion (CoC). The relationship between the diameter of the CoC and the depth is shown by [Pentland, 1987]. Rearranging the thin lens law (i.e., Equation 2.4) in terms of u gives

$$u = \frac{vf}{v - f} \ . \tag{2.5}$$

For a particular lens system, the focal length f is constant. Assuming that the distance between the lens and the image plane is fixed at $v = v_0$, and the distance at which a point will be in perfect focus is at $u = u_0$, gives

$$u_0 = \frac{fv_0}{v_0 - f} \ . \tag{2.6}$$

Figure 2.4 illustrates the case when the distance of the object from the lens, u, is greater than the critical point of focus, u_o with a lens of radius r. This gives a CoC with radius σ.

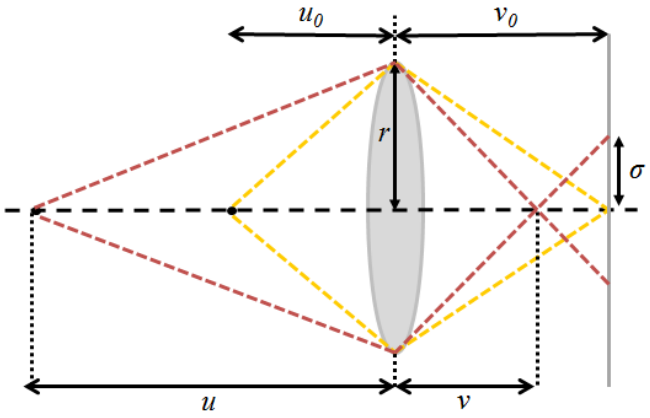


Figure 2.4: Lens system with fixed values v_0, u_0 and f, when the distance of the object from the lens, u, is greater than the critical point of focus, u_0. This gives the resultant circle of confusion of radius σ.

Using the law of similar triangles, it can be shown that

$$\frac{r}{v} = \frac{\sigma}{v_0 - v} \ . \tag{2.7}$$

Substituting $v = r(v_0 - v)/\sigma$ into Equation 2.5 gives

$$u = \frac{frv_0}{rv_0 - fr + f\sigma} \ . \tag{2.8}$$

It can equally be shown that for objects closer to the lens than the critical focus point

$$u = \frac{frv_0}{rv_0 - fr - f\sigma} \ , \tag{2.9}$$

which gives the general equation

$$u = \frac{frv_0}{rv_0 - fr \pm f\sigma} = \begin{cases} + & \text{if} \quad u > u_0 \\ - & \text{if} \quad u < u_0 \end{cases} \ , \tag{2.10}$$

or

$$u = \frac{fv_0}{v_0 - f \pm \sigma F_N} = \begin{cases} + & \text{if} \quad u > u_0 \\ - & \text{if} \quad u < u_0 \end{cases} \ , \tag{2.11}$$

where F_N is the f-number of the lens. This shows depth to be an indicator for defocus. Lai [Lai et al., 1992] rewrites the equation to more clearly show the relationship. Assuming that for a given lens system f, v_0 and F_N are all constant, then Equation 2.11 can be written as:

$$u = \frac{P}{Q \pm \sigma} \ . \tag{2.12}$$

where $P = fv_0/F_N$, $Q = (v_0 - f)/F_N$, and P and Q are constant for a given camera system. When a point is in perfect focus at the critical focus point, the amount of defocus, σ, will be zero ($u_0 = P/Q$). The formula shows that the CoC gradually increases as the object point moves either deeper (i.e., further away) or shallower (i.e., nearer) to the lens than the critical focus point u_0.

2.2.4 Depth of Field

It has been shown that the diameter of the CoC increases with the distance (either shallower or deeper) of the point from the point of critical focus. If the diameter of the CoC is less than the resolution of the human eye (or of the display medium), then the image point will still appear to be in focus. The region for which this holds true is known as the DoF. Note that the resolution of the human eye and display medium are likely to be different, thus an image might have a different DoF in a human visual system (HVS) as opposed to a machine vision system (MVS). An alternative qualitative definition for DoF is the distance between the deepest and shallowest points in a given scene that appears acceptably sharp in the image of the

scene. Figure 2.5 shows an example whereby an identical scene is captured twice by a camera. The DoF of image (b) is lower than that of (a), and thus the background and parts of the object further from the lens appear blurred.

(a) (b)

Figure 2.5: Images of an identical scene captured using lens systems with a different DoF: (a) with an aperture of $f/32$ and (b) with a relatively large aperture of $f/5$.

It can be seen from Equation 2.11 that a number of variables can be altered to produce an image with a low DoF, i.e. a larger CoC for a given distance from the critical focus point. Photographers will usually either use a larger aperture or lens with a longer focal length in order to achieve the effect. Figure 2.6 (a) and (b) respectively show the effect of using a large and a small aperture. It can be seen that a smaller aperture results in smaller CoCs, and thus a greater DoF, and vice versa.

2.2.5 Low Depth of Field Photography

Depending on the desired effect, a photographer or cameraman can use a low or high DoF. A low DoF is standard in many television or film productions as it directs the viewer's attention to the important part of the scene. Background objects are blurred and thus do not provide a distraction to the viewer. Whilst the HVS system is normally very good at determining depth in an image, using a low DoF also helps to convey this information more clearly. An example of a low DoF used in a film production is shown in Figure 2.7(a). Similarly a photographer will use a shallow DoF in fields such as portrait or flowers/animal photography to emphasise the subject of the photograph, such as in Figure 2.7(b). Using a low DoF is also common in scientific applications. For example in microscopy, by panning through different planes of focus, a user can get an idea of the relative depths and heights of very small structures.

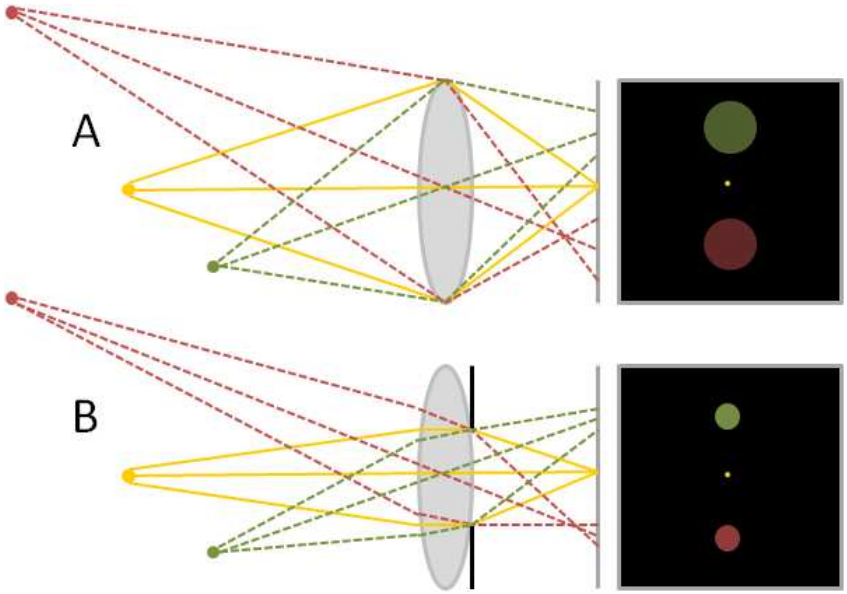


Figure 2.6: The effect of aperture size on depth of field. Lens system (A) shows the effect of a larger aperture on CoC size (and thus DoF) whilst (B) shows that of a smaller aperture.

2.3 Image Segmentation

In computer vision, image segmentation is the process of dividing a digital image into multiple parts or segments. This is typically to simplify or change the appearance of the image to allow meaningful data to be more more easily extracted or analysed. Pixels are grouped into non-overlapping segments which share a similar characteristic or property, such as colour, intensity or texture. The union of these segments forms the entire image, and no two adjacent segments will have the same property. Segmentation is more formally defined in [Pal and Pal, 1993].

For a segmentation method where F is the set of all pixels and $P(\ )$ is a characteristic value assigned to a group of similar connected pixels if they fulfil a logical criteria (known as the uniformity predicate), then segmentation is the partitioning of the set F into a set of connected subsets or regions $(S_1, S_2, S_3, ..., S_n)$ such that

$$\bigcup_{i=1}^{n} S_i = F \text{ with } S_i \cap S_j = \emptyset, i \neq j \ . \tag{2.13}$$

The uniformity predicate $P(S_i)$ = true for all regions S_i, and when S_i is adjacent to S_j then $P(S_i \cup S_j)$ = false. This remains true for all types of images, not just those representing the visible light domain.

(a) (b)

Figure 2.7: Use of low DoF: (a) in film production, and (b) in portrait photography.

2.4 Object Segmentation

Object segmentation is a specific case of image segmentation whereby the aim is to divide the image into two different areas, the object of interest (OoI) and the background. This is known as a binary segmentation. Some algorithms also generate a third ambiguous region along object boundaries. Object segmentation can be more problematic than image segmentation as a given object may have different properties such as colour or texture across its surface. It my also contain multiple internal contours (e.g., edges within an object) making object segmentation via edge detection a difficult task.

Segmentations can be divided into three different categories: those that are performed entirely by the user (i.e., manual segmentations); those that deal with a specific type of image or object, thus reducing the unknowns and allowing for autonomous segmentation methods (i.e., unsupervised segmentations); and those that use human input to guide or refine a segmentation (i.e., supervised segmentations) and thus can function with a wider or general range of objects and scenes.

2.4.1 Manual Segmentation

Manual segmentations involve the user identifying which regions have a similar characteristic and marking them manually to form a segmented image. Such segmentations can be very time consuming with large images and rely on a certain degree of skill from the user. Which areas belong to which segments can also be subjective, meaning two users could end up with significantly different segmentations. Other methods involve marking object contours and potentially making use of geometrical shapes such as ellipses to approximate the boundaries of objects, thus saving time but at the cost of accuracy.

2.4.2 Unsupervised Segmentation

Creating a general unsupervised segmentation method is notoriously difficult due to the sheer range and variation in image characteristics and is considered to be an unsolved problem. Autonomous segmentation methods therefore require some form of a priori knowledge about the image. This could be information about the background, or the object to be segmented, or even more general image properties. For example, the method this thesis presents in Chapter 4 uses the fact that input images will have a low DoF, and that the OoI will be in clear focus, in order to perform autonomous segmentations.

By limiting the potential variations from image to image, unsupervised segmentations can form part of larger autonomous systems. For example, one well documented application is in automated-picking robots.

2.4.3 Supervised Segmentation

Supervised segmentations combine the most efficient parts of manual and automatic segmentations. A HVS can identify very quickly which parts of an image are of interest and this additional information allows a segmentation algorithm to function quickly and accurately. The user input in supervised methods is generally given in one of the following three ways:

1. Specification of an initial boundary, or parts of a boundary. This initial contour then evolves to the desired object boundary and is used in segmentation methods based on the active contours algorithm [Kass et al., 1988].

2. Denotation of a small set or sets of pixels that belong to the object or segment of interest, this is sometimes known as a 'seed'. In some methods the user will also specify a set of pixels that belong to the background of the image. Well known techniques such as GraphCuts [Boykov and Jolly, 2001] and seeded region growing methods often use this kind of user input.

3. Specification of points along an OoI's boundary. These points are connected to form a contour which then 'snaps' to the desired object's boundary. This form of input is used in methods such as intelligent scissors [Mortensen and Barrett, 1995].

Feedback can also play a significant part in supervised segmentations, for example in the GrabCut method [Rother et al., 2004]. After an initial supervised segmentation is formed, the user is given the option to add in foreground or background seeds to refine the segmentation until satisfied. Such methods mean that

given enough time, a user can repeatedly refine the segmentation until a 'perfect' result is obtained.

2.5 Segmentation Methods

Image segmentation is a popular and well researched field, and thousands of segmentation methods have been presented in literature [McGuiness and O'Conor, 2010]. Aside from being categorised on levels of user input, methods can be further subdivided into two areas: edge based and region based methods.

Edge detection is an entire field of image processing in itself, but can form the basis of segmentation techniques. Edge-based segmentation methods generally use some form of edge operator or filter followed by a thresholding to obtain the contours in an image. Enclosed regions are considered to be separate segments as they lack continuity with adjacent regions and can be identified by simple 'fill' operations. Broken contour lines, for example caused by blurring, will result in failed segmentations and thus such methods tend to involve some form of line-linking operation.

For the task of binary object segmentation, region based techniques are more applicable. This is because edge detection methods cannot produce binary segmentations of objects with multiple internal contours. This section discusses some of the broad concepts that are the basis for many segmentation methods.

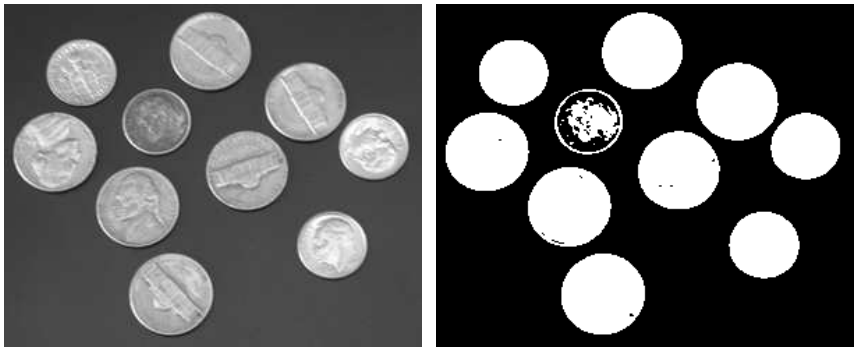
2.5.1 Thresholding

Thresholding is one of the simplest methods for image segmentation. In its most basic form it assigns all pixels with intensity values (F) above a certain level as object and those below this threshold, α, as background, i.e.,

$$\begin{array}{ll} \text{if } F(x,y) \geq \alpha & F(x,y) = 1 \text{ (object)} \\ \text{else} & F(x,y) = 0 \text{ (background)} \end{array} . \tag{2.14}$$

The threshold can be given manually or calculated automatically. For example Otsu's method [Otsu, 1979] automatically generates a threshold for a binary segmentation by minimising the intra-class variance, as shown in Figure 2.8.

Thresholds can be applied either globally (as in Equation 2.15) or locally. Using different thresholds for individual pixels in an image is known as adaptive thresholding. Adaptive methods tend to take one of two approaches: windowing, or local thresholding. A windowing thresholding method, such as that proposed by Chow [Chow and Kaneko, 1972], divides the image into a number of overlapping



(a) (b)

Figure 2.8: Segmentation of an initial image (a) using a threshold automatically generated via Otsu's Method to produce a binary segmentation (b).

subimages. These are considered separately and optimal thresholds for each subimage are found. These calculated thresholds are used to interpolate the threshold for each individual image pixel. More current approaches use a less computationally intensive local thresholding approach, where the threshold for a pixel is determined by the values of pixels in its neighbourhood. For a example, a simple local thresholding could use the mean of neighbouring pixels to calculate a threshold:

$$\begin{array}{ll} \text{if } F(i,j) \geq M - C & F(i,j) = 1 \text{ (object)} \\ \text{else} & F(i,j) = 0 \text{ (background)} \end{array} . \tag{2.15}$$

where C is a constant and M is the mean of pixels belonging to the neighbourhood of size $(2N+1) \times (2N+1)$ given by

$$M = \frac{1}{(2N+1)^2} \sum_{x=i-N}^{i+N} \sum_{y=j-N}^{j+N} F(x,y). \tag{2.16}$$

The size of the neighbourhood used greatly impacts on the performance of the segmentation method.

2.5.2 Histograms

Grey level histograms can also be used to determine thresholds for binary segmentations. Once peaks in a histogram have been identified, a threshold can be set at the minima between them, as shown in Figure 2.9.

A segmentation method can also apply many different level thresholds to an image, dividing it into multiple segments. This is known as multi-thresholding. A

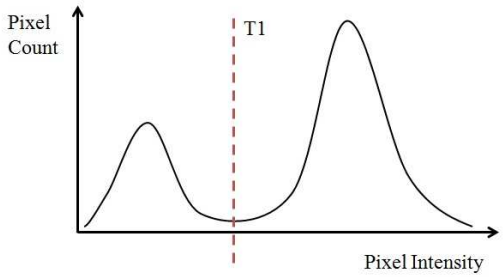


Figure 2.9: Histogram of the intensity/grey level in a image. The selected threshold, T1, corresponds to the minimum between the two peaks.

histogram is computed from all image pixels, and peaks and troughs are identified to group similar pixels, for example as shown in Figure 2.10. Such a method can easily be extended to grouping pixels with similar colours rather than grey levels.

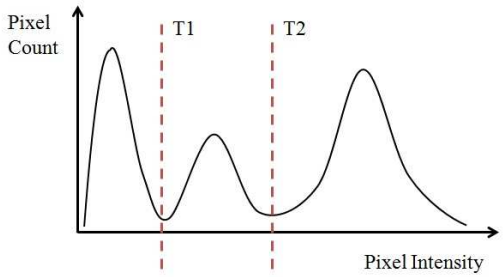


Figure 2.10: Histogram of the intensity level in a image with three groupings of pixels. Threshold levels correspond to the minima between peaks.

Compared to some other segmentation techniques, a histogram is computationally very efficient, requiring only a single pass of the image. If the groupings of pixels are not very distinctive then difficulties can arise in selecting appropriate thresholds. More complex methods use further histograms to recursively break down clusters of similar pixels into smaller groupings. Whilst effective at grouping similar colour pixels, a binary object segmentation via histogram may not be possible if an object is not of a uniform colour.

2.5.3 Clustering

The K-means algorithm is a popular cluster analysis method that is commonly applied to the problem of image segmentation. It was proposed concurrently by a number of scientists under various different guises [Bock, 2008]. In its basic form the algorithm segments an image into K clusters. The Initial cluster centres can either

be picked randomly or seeded. The algorithm then follows the following iterative process:

1. Each pixel is assigned to a cluster which minimises the distance between the cluster centre and the pixel.

2. Cluster centres are re-calculated by averaging all the pixels within the cluster.

These two steps are repeated until there are no further changes in the membership of the clusters. The distance is commonly defined as either the square or absolute difference between a pixel and the cluster centre, and can be based on colour, intensity, texture, or a weighted combination of these factors. Figure 2.11 shows an image segmented using a K-means algorithm with $K = 16$.

(a) (b)

Figure 2.11: Colour segmentation of an image (a) using a K-means algorithm with 16 clusters to produce a colour segmentation (b).

2.5.4 Region Growing

Region growing algorithms often make use of user defined seeds [Adams and Bischof, 1994]. In its simplest form, a seed is placed in each object or region to be segmented. From these starting points, regions are grown iteratively from all unallocated neighbouring pixels. The intensity or hue of the neighbouring pixel that is most similar to the mean of the region being grown is allocated to that region. This is repeated until all pixels are allocated to a region.

Seedless region growing methods can also produce successful segmentations. Starting from a random pixel, an initial region is created. The difference in intensity or hue of neighbouring pixels is calculated. If the difference is below a certain threshold then the pixel is added to a region. If it is greater, a new region is created with this pixel. The process is repeated until all pixels are assigned to a region.

Adjacent regions can be merged under some criteria, for example sharpness of region boundaries. A harsh criterion can create a fragmented segmentation whereas a lenient one could overlook blurred boundaries and oversimplify the segmentation.

2.5.5 Split and Merge Algorithms

Region merging methods address the problem of image segmentation from the bottom up. Every pixel is considered to be a seed. If two neighbouring pixels are the same, or similar enough according to some criterion, then they are merged into a single region. Likewise, if properties of two adjacent regions are similar enough to each other, they will be merged. This process continues until no further merging is possible. These kinds of algorithms are computationally very intense. Split and merge algorithms, proposed by Horowitz [Horowtiz and Pavlidis, 1974], are much more efficient and start from the top downwards.

A quadtree structure is generally used for the splitting process. The entire image is considered to be a single region. If this region is uniform (or if all pixels within the region have sufficient similarity) then the region is left as it is. If the pixels in the region are non-homogeneous (or outside of some range or threshold of conformity) it is then subdivided into four quadrants, i.e., the child regions. The process is then repeated for each of these child regions, i.e., the conformity of the pixels is checked which determines whether the region will be split again. These subdivisions are continued until no further splits occur or the resolution of the quadtree is reached. The merging algorithm can then proceed, merging regions from the bottom up. Starting with these small regions rather than single pixels means that split-merge methods are significantly more efficient than pure merge algorithms.

2.5.6 Watershed Transformation

The watershed transform, first proposed by Digabel [Digabel and Lantuejoul, 1977] is a popular segmentation method. It takes its inspiration from geography. Watershed transformation methods treat an image as a topographical map, where the intensity of a pixel is interpreted as its altitude, e.g., high value regions appear as peaks or 'mountains' and low values as troughs or 'valleys'. This topographical map is then flooded from local minima. 'Water basins' fill up from these minima and where two basins converge a 'dam' is formed. The flooding process is stopped once the water reaches the level of the highest peak. The resulting image is segmented into regions (or basins) separated by the dams known as watershed lines.

In practice watershed transform tends to be performed on the morphological gradient of the image, not the greyscale image. This generates watershed lines along the points of intensity discontinuity, most likely edges, meaning the regions or basins will correspond to objects, or object regions within an image.

2.5.7 Active Contours

The general principle behind an active contours algorithm is that an initial curve or snake evolves to try and minimise an energy function, drawing it towards an object boundary [Kass et al., 1988]. Representing the snake parametrically by $\mathbf{v}(s) = (x(s), y(s))$, the energy function can be written as

$$\begin{aligned} E^*_{snake} &= \int_0^1 E_{snake}(\mathbf{v}(s))\ ds \\ &= \int_0^1 [E_{internal}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s)) + E_{constraints}(\mathbf{v}(s))]\ ds \end{aligned} \quad (2.17)$$

where $E_{internal}$ is the internal energy of the curve due to bending, E_{image} is the force pulling the contour towards salient image features and $E_{constraints}$ are external constraints imposed upon the curve by the user.

Active contours models can either be parametric snakes or geometric snakes. Parametric snakes are represented by splines and the contour evolution is only performed on specific points along the contour. They have the disadvantage of not being able to split the contour to detect multiple objects without manual intervention and, unless the initial curve is close to the object boundary, can converge on non-object points [Hou and Han, 2005].

Geometric, or level set methods, represent the contour of an object as the zero-level set of a higher dimensional function. With an image, the contour of an object on the 2D image plane is updated when its 3D surface is evolved as illustrated in Figure 2.12. The advantages of level set methods are that they allow for complex curve behaviour, namely the merging and splitting of the contour is an easy process.

There are many different implementations for the energy function which can be used depending on the desired application of the active contours. These can either be implicit or explicit methods. Explicit or edge based active contours mainly use image gradient information to find object boundaries. Implicit or region based active contours utilise other information such as texture and localised grey level intensity. A popular implicit model is proposed by Chan [Chan and Vese, 2001]. Based on the Mumford-Shah functional [Mumford and Shah, 1989] for segmentation the method can detect objects within a given scene whose boundaries are not necessarily defined

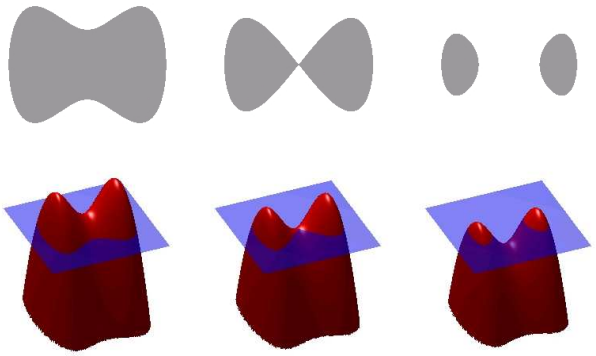


Figure 2.12: Level set method: (bottom row) the evolving level set function of a 3D dark grey object and a 2D light grey image plane; (top row) the corresponding contour curves of the object regions on a 2D image plane are the zero-level set values of the evolving object surface.

by a gradient and is very robust to noise.

One of the main criticisms of active contour algorithms is that they are computationally intensive, especially when dealing with large images. This is particularly true of level set methods. A number of implementations can be used to increase speed. The sparse field method [Whitaker, 1998] is a narrow band level-set implementation which substantially reduces the number of computations required per iteration by only performing calculations near the zero level set. Other criticisms of active contours are that methods tend to be very dependent on having a good initial contour, which is why they are commonly defined by the user. In some implementations there are also risks of the evolving boundary becoming stuck in local minima.

2.5.8 Graph Partitioning Methods

Graph partitioning methods model an image as a weighted undirected graph. Pixels form the nodes of the graph and edge weights represent the difference or similarity between neighbouring pixels. All the nodes in the graph are grouped into two or more partitions based on certain criteria. For example, in the graph cuts method [Boykov and Jolly, 2001] the user specifies some initial object and background seeds. A graph is created with two terminals. The edge weights or costs for t-links (node to node links) are defined by a regional term, and the costs for n-links (node to terminal) by a boundary term, with both taking the user defined seeds into account.

A min-cut algorithm (aiming to minimise the cost of links cut) categorises pixels as either object or background, thus segmenting the image. This process is illustrated in Figure 2.13.

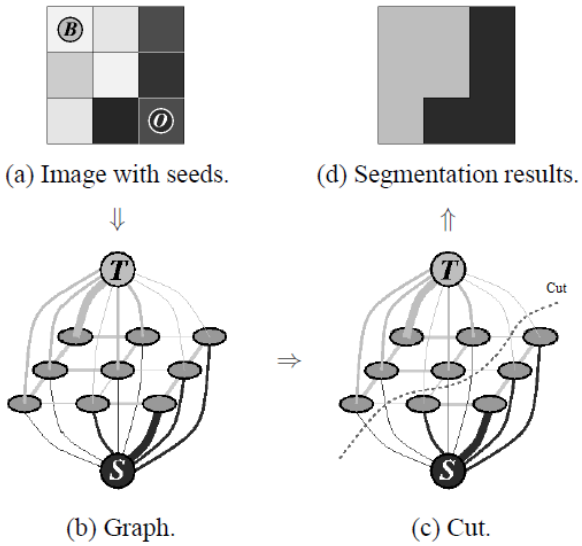


Figure 2.13: Example of a simple segmentation of a 3x3 image where T is the background terminal and S the object terminal. 'B' and 'O' denote background and object seeds, respectively. Figure is adapted from [Boykov and Jolly, 2001].

2.5.9 Conclusion

In this chapter the concept of focus is introduced and the thin lens formula is derived for a focused point. It is shown that light from shallower or deeper object points will not converge to a single point on the image plane, but instead upon an optical spot, known as the CoC. The size of the CoC is dependant on the distance from the point of critical focus, thus depth is shown to be an indicator for defocus. DoF and the use of a low DoF in the media is described. Image and object segmentation are defined and a number of popular methods and techniques covered. In order to produce an autonomous object segmentation method the number of unknowns must first be reduced. Thus, this thesis concerns itself with unsupervised object segmentation from low DoF images and video sequences.

Chapter 3

Focus Assessment

3.1 Introduction

Research into the focus of an image is not an uncommon theme. This can vary from assessing the focus of an entire image to identifying regions of different focus within a given image. However, the objective is the same, namely to determine whether the image, or parts of the image, have undergone some kind of blurring operation as a result of being outside of the DoF, as discussed in Chapter 2. Assessments of whole images can be used to determine whether images of a given scene are focused more or less than each other, for example when the distance of the image plane from the lens is changed. Focus assessment of the regions within an image can be used to obtain further information about a scene, e.g., to identify objects, or estimate the depth of regions within a calibrated image. The most common way of determining a focus value of a pixel is by determining how much it contrasts with its neighbouring pixels, i.e., how blurred the section of the image is.

Focus assessments are used in a wide variety of applications and research areas. They are an integral part of autofocusing algorithms, for example within the field of computer microscopy [Sun et al., 2004b], where autofocusing is used to automatically determine the best scope settings to view a slide. Autofocusing functions of some digital cameras also make use of a focus assessment in its simplest form to perform a contrast detection. A camera using a contrast detection method will pan through a range of lens settings to select the optimal. This is where the intensity difference between neighbouring pixels is at its maximum, i.e., where there is the least amount of blur. The use of a focus assessment is also common in iris recognition systems [Jang et al., 2008], where obtaining the clearest and most focused image is of the upmost importance if reliable results are to be consistently

produced.

Other applications include image fusion [Huang and Jing, 2007], where focus assessments are used to merge two images that focus on different parts of a scene in order to create one fully focused composite image. This can also be applied in microscopy so that structures at different levels are in focus in the composite image. Defocus can also be used to estimate the depth of pixels in an image [Pentland, 1987] which in turn can be used to perform 3D reconstruction of surfaces [Nayar and Nakagawa, 1994].

The premise behind the object segmentation method presented in this thesis is that by limiting the type of image to be processed to those with a low DoF, an autonomous method can be created. Assuming that an image generated from a camera system has focused on the OoI within a given scene, the background pixels of the image will be less sharp when compared to object pixels. A focus assessment allows the different regions, e.g., background and object, to be differentiated by some segmentation method. This chapter is primarily concerned with the selection of a suitable assessment to create a focus map, and is organised as follows: Section 3.2 briefly describes the most common way of modelling defocus. Section 3.3 provides an overview of the seminal focus assessment methods, dividing them into three different categories; statistical, derivative, and wavelet based methods. Example focus maps are shown for a number of different assessments. Section 3.4 evaluates the suitability of a range of different focus assessment methods for the problem of object segmentation from low DoF images. Section 3.5 considers the effects of image resolution on the performance of the focus assessment methods. Finally, Section 3.6 proposes a multiscale variation of a wavelet based focus assessment. Some focus maps generated using this method are shown in Section 3.7. In Section 3.8, the chapter is concluded.

3.2 Modelling Defocus

The point spread function (PSF) describes how an imaging system will respond to an object point, i.e., the blurring operation an object point undergoes when an image is formed. The relationship between the actual or original scene $f(x, y)$ and the corresponding captured image $g(x, y)$ can be described by the following convolution [Jain, 1998]:

$$g(x, y) = (f * h)(x, y), \tag{3.1}$$

where $h(x, y)$ is the PSF of blurring and has the characteristics of a low-pass filter. Autofocusing algorithms seek to minimise blur such that $g(x, y) \approx f(x, y)$. As the blurring caused by defocus is modelled as a low-pass filter, most focus assessments measure high-frequency components in an image, to detect areas less affected by blurring.

3.3 Focus Assessment Methods

The basic premise behind most focus assessment methods is that focused images will contain more information, or detail, than defocused or blurred images. Detecting the presence of this detail, shown as high frequency components, is key to assessing an image's level of focus. As such, it is common for methods to rely on edge information, where the contrast between neighbouring pixels will be greatest. As focused images will have sharp edges, they will contain more high frequency content than their equivalent in a defocused image.

This premise leads to a number of issues. If the focus differential between the background and foreground is small, then some background contours may have higher frequency components than internal parts of the focused OoI. This is particularly true of objects which are mostly homogeneous, or have weak textures. This is because despite being within the DoF, if there is no contrast between adjacent pixels (for example in a mono-colour plastic object where the illumination is equal on all parts) then the object will behave the same as a defocused region, having no high frequency components. Only the region boundaries can be differentiated.

Other potential problems in assessing the focus of an image include artefacts such as light glare or flash reflections. These can potentially create artificial high frequency components in the otherwise less sharp background.

The size or resolution of an image will also have an effect on focus assessment. Whilst in images of a small size (which is defined in this thesis as having height and width in the order of hundreds, not thousands of pixels), or low resolution, edges are likely to be the most prominent and useful factor in determining focus. For larger scales or higher resolution images, the internal contours and texture have significant effect.

This section presents some of the seminal focus assessment methods, many of which are evaluated in Section 3.4. The methods are grouped into the following areas: statistical methods, derivative and kernel based methods, and wavelet based methods.

3.3.1 Statisical Methods

Statistical methods are generally applied to automatic focusing problems. They work on the basis that focused images will have more information than defocused images. Rather than assessing focus on pixel level, they assess the whole image. Thus they are useful in determining the comparative focus between images of the same scene, hence their use in autofocusing algorithms. They tend to be more robust to image noise than other types of focus assessment methods.

Variance

The variance algorithm [Groenand et al., 1985; Yeo et al., 1993], sums the square in the difference in pixel intensities $i(x, y)$ from the mean intensity. The focus value is

$$F_{variance} = \frac{1}{H.W} \sum_{Height} \sum_{Width} (i(x, y) - \mu)^2 \ , \tag{3.2}$$

where H is the image height, W is the image width and μ is the mean pixel intensity. Squaring the difference amplifies larger difference in pixel intensities from the mean.

Normalised Variance

The normalised variance algorithm [Groenand et al., 1985; Yeo et al., 1993] factors the mean intensity into the final focus value. This allows the focus of images of different scenes to be compared as changes in average image intensity are compensated for. The focus value is

$$F_{variance} = \frac{1}{H.W.\mu} \sum_{Height} \sum_{Width} (i(x, y) - \mu)^2 \ , \tag{3.3}$$

where H is the image height, W is the image width and μ is the mean pixel intensity. As with the variance method, squaring the difference amplifies larger difference in pixel intensities from the mean.

Figure 3.1 shows an example of the normalised variance method being applied to (a), a low DoF image of a wizard against a forest background, and (b) the same image having undergone a Gaussian blurring operation. The focus values calculated using the normalised variance for the images are 0.0657 and 0.0594, respectively, showing the unblurred image to have a higher focus value as would be expected.

(a) (b)

Figure 3.1: Low DoF image taken from video footage (a), and the same image having undergone a blurring operation (b). The focus values calculated using the normalised variance for the images are 0.0657 and 0.0594, respectively.

Range Algorithm

Other statistical algorithms make use of histograms, $h(i)$, to analyse the distributions of intensities within an image. The range algorithm [Firestone et al., 1991] computes the difference between the highest and lowest intensity levels, i.e., the focus value is

$$F_{range} = max_i(h(i) > 0) - min_i(h(i) > 0) \ . \tag{3.4}$$

The premise being that blurring will attenuate any extreme values in pixel intensity.

Entropy

The entropy algorithm [Firestone et al., 1991], as with all the statistical methods, assumes that images that are in focus contain more information than those that are not. Utilising a histogram the focus value is

$$F_{entropy} = - \sum_{intensities} p_i.log_2(p_i) \ , \tag{3.5}$$

where $p_i = h(i)/(height \times width)$ is the probability that a pixel has an intensity of i, and $height$ and $width$ are the dimensions of the image.

3.3.2 Derivative Methods

Derivative methods assume that focused images have more high frequency components than blurred images. Based on this premise, neighbouring pixels will have larger differences in intensity in focused images. Derivative methods apply some

from of convolution mask to apply a high pass filter and obtain a map of the derivatives, giving an indication of the focused areas in the image.

The derivative methods in this section sum the results of the convolution to obtain a focus value for the image. In order to produce a pixel based focus assessment of the image, the summing is not performed and instead the results of the convolution are used as a focus intensity map. This makes such methods more suitable as an initial stage in a segmentation method than the statistical methods in Section 3.3.1.

Tenengrad

The Tenengrad focus operator [Tenenbaum, 1970] applies vertical and horizontal Sobel operators to the image. In order to compute the focus value of a pixel, the square of the results of the convolution are summed within an $(2N+1) \times (2N+1)$ window centred around the pixel to be assessed, i.e.,

$$F_{Tenengrad}(i,j) = \sum_{x=i-N}^{i+N} \sum_{y=j-N}^{j+N} S_x(x,y)^2 + S_y(x,y)^2, \tag{3.6}$$

where $S_x(x,y)$ and $S_y(x,y)$ are the results of the convolution of the image with the horizontal and vertical Sobel operators, respectively (i.e., along the x and y directions, respectively).

Figure 3.2 shows an example of the Tenengrad method being applied to a DoF image of a wizard against a forest background. The corresponding focus intensity map generated is shown in Figure 3.2(b).

(a) (b)

Figure 3.2: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using the Tenengrad method (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Modified Laplacian

Designed to cope with weakly textured images, the modified Laplacian (ML) operator [Nayar and Nakagawa, 1994] sums the absolute values of the convolution of the image with the Laplacian operators, giving the focus value

$$F_{ML}(i,j) = |L_x(x,y)| + |L_y(x,y)|, \tag{3.7}$$

where $L_x(x,y)$ and $L_y(x,y)$ are the results of the convolution of the image with the horizontal and vertical Laplacian operators (Lap_x and Lap_y), respectively (i.e., along the x and y directions, respectively), with

$$Lap_x = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}, Lap_y = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}.$$

The Laplacian methods were developed to measure focus at each image point in order to estimate depth as part of a shape from focus system. Figure 3.3 shows the ML method being applied to the image of the wizard.

(a) (b)

Figure 3.3: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using the ML method (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Sum Modified Laplacian

The sum modified Laplacian (SML) method sums the results of the ML method within a local window to obtain the focus value

$$F_{SML}(i,j) = \sum_{x=i-N}^{i+N} \sum_{y=j-N}^{j+N} F_{ML}(x,y), \tag{3.8}$$

where $F_{ML}(x, y)$ is the convolution of the image with the modified Laplacian operator.

(a) (b)

Figure 3.4: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using the SML method (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Energy Laplace

The Energy Laplace method [Subbarao et al., 1993] is another kernel based focus assessment. It was proposed as a recommended measure for camera autofocusing systems. It uses the operator

$$L = \begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix}$$

to give the focus value

$$F_{EL}(i, j) = \sum_{x=i-N}^{i+N} \sum_{y=j-N}^{j+N} C(x, y)^2, \tag{3.9}$$

where $C(x, y)$ is the convolution of the image with operator L. Again, results of the convolutions are summed within a $(2N + 1) \times (2N + 1)$ window to obtain the final focus value of the pixel. Figure 3.5 shows the Energy Laplace method being applied to the image of the wizard.

(a) (b)

Figure 3.5: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using the Energy Laplace method (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Daugman

Originally applied in the the field of iris recognition to select optimal images, Daugman's method [Daugman, 2004] uses the following 8×8 focus assessment kernel:

$$D = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 3 & 3 & 3 & 3 & -1 & -1 \\ -1 & -1 & 3 & 3 & 3 & 3 & -1 & -1 \\ -1 & -1 & 3 & 3 & 3 & 3 & -1 & -1 \\ -1 & -1 & 3 & 3 & 3 & 3 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}.$$

Its convolution with the image gives the focus value

$$F_{Daugman}(i, j) = C(x, y), \tag{3.10}$$

where $C(x, y)$ is the convolution of the image with kernel D. An example of Daugman's kernel being applied a low DoF image is shown in Figure 3.6.

(a) (b)

Figure 3.6: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using Daugman method (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Wei

Wei presents an improved focus assessment [Wei et al., 2006] over Daugman's method, using a smaller 5×5 kernel to improve computational efficiency, i.e.,

$$W = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & 2 & 2 & 2 & -1 \\ -1 & 2 & 0 & 2 & -1 \\ -1 & 2 & 2 & 2 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}.$$

Its convolution with the image gives the focus value

$$F_{Wei}(i,j) = C(x,y), \tag{3.11}$$

where $C(x,y)$ is the convolution of the image with kernel W. An example of Wei's kernel being convolved with a low DoF image of a wizard is shown by Figure 3.7.

(a) (b)

Figure 3.7: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using Wei's focus assessment kernel (b). The image has had its intensity enhanced by a factor of 3 for clarity.

Kang

Kang's focus assessment kernel [Kang and Park, 2005; Kang, 2006] is another 5×5 operator, i.e.,

$$K = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 4 & -1 & -1 \\ -1 & 4 & 4 & 4 & -1 \\ -1 & -1 & 4 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}.$$

It is again proposed for use within the field of iris recognition. Its convolution with the image gives the focus value

$$F_{Kang}(i,j) = C(x,y), \tag{3.12}$$

where $C(x,y)$ is the convolution of the image with operator K. Figure 3.8 shows the result of Kang's kernel being applied to a low DoF image.

3.3.3 Wavelet Methods

A series of focus measures utilising wavelets is proposed in [Yang and Nelson, 2003a,b]. The focus value is used as a cue for the segmentation of low DoF microscopic images via a graph partitioning method. The Daubechies 6 wavelet filter is used to divide the image into four subband images W_{LL}, W_{HL}, W_{LH} and W_{HH}, where L denotes lowpass filtered, H denotes highpass filtered, and their order denotes the order of the filtering applied, e.g., W_{HL} is a subband image obtained by

(a) (b)

Figure 3.8: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated using Kang's focus assessment kernel (b). The image has had its intensity enhanced by a factor of 3 for clarity.

highpass filtering followed by lowpass filtering. The focus measures are as follows:

Wavelet 1

$$F_{wavelet1} = |W_{HL}(x,y)| + |W_{LH}(x,y)| + |W_{HH}(x,y)| \tag{3.13}$$

Wavelet 2

$$F_{wavelet2} = (|W_{HL}(x,y)| - \mu_{HL})^2 + (|W_LH(x,y)| - \mu_{LH})^2 + (|W_{HH}(x,y)| - \mu_{HH})^2 \tag{3.14}$$

where μ is the mean of a subband image computed using absolute values.

Wavelet 3

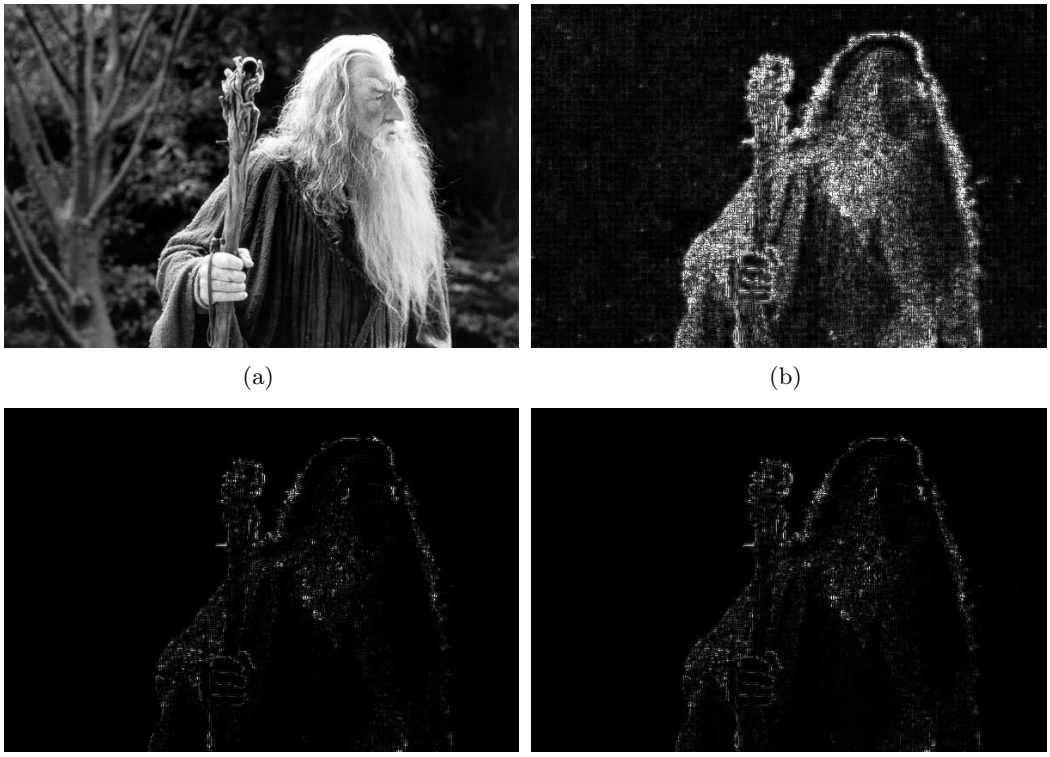
$$F_{wavelet3} = (W_{HL}(x,y) - \bar{\mu}_{HL})^2 + (W_LH(x,y) - \bar{\mu}_{LH})^2 + (W_{HH}(x,y) - \bar{\mu}_{HH})^2 \tag{3.15}$$

where $\bar{\mu}$ is the mean of a subband image computed without using absolute values.

Figure 3.9 shows the three methods being applied to a low DoF image of a wizard against a wooded background.

3.4 Evaluation of Focus Measures

A robust measure for the focus values of the image pixels is required for the first stage of the proposed segmentation method which incorporates the active contours algorithm. To determine which measure would be appropriate, the performance of the focus assessment methods presented in Section 3.3 are evaluated. Eight high



(a) (b)

(c) (d)

Figure 3.9: Low DoF image taken from video footage (a), and the corresponding focus intensity map generated by the wavelet 1 method (b), the wavelet 2 method (c) and the wavelet 3 method (d). The images have been contrast-enhanced by a factor of 3 for clarity.

resolution (HR) test images as shown in Figure 3.10 and eight lower resolution (LR) test images as shown in Figure 3.11 with clear focus differentials between the background and in-focus objects (i.e., the OoIs) are manually segmented to obtain their ground truth. The number of pixels along each of the dimensions of a HR image and a LR image is of the order of thousands and hundreds, respectively.

The focus assessment methods are applied to the images, and the properties of the resulting range of focus values for the object and background regions recorded. The following two criteria are used to determine the best measure:

$$\bar{F}_{background} \ll \bar{F}_{object} \tag{3.16}$$

$$\sigma_{background} \ll \sigma_{image} \tag{3.17}$$

where $\bar{F}_{background}$ and $\bar{F}_{object}$ are respectively the mean focus value of the background and in-focus object, and $\sigma_{background}$ and σ_{image} are respectively the standard devi-

Figure 3.10: High resolution test images with focus differentials between OoI and background.

Figure 3.11: Low resolution images with focus differentials between OoI and background.

ation of the background and image focus values. The value of $\bar{F}_{background}$ should be low compared to $\bar{F}_{object}$, thus the higher the ratio $\bar{F}_{object}/\bar{F}_{background}$ the better the measure. $\sigma_{background}$ should be as small as possible when compared to σ_{image}, as the background should be relatively homogeneous. The lower $\sigma_{background}$ is, the less likely the active contours algorithm will result in an incorrect segmentation due

to focus values of anomalous background pixels. The focus measures are ranked against each other for both criteria and given a score (the sum of the ranking for the two criteria) for each of the test images, i.e.,

$$Score = Ranking_{\bar{F}_{object}/\bar{F}_{bkgnd}} + Ranking_{\sigma_{bkgnd}/\sigma_{image}} \tag{3.18}$$

where a subscript denotes the criterion used for the ranking. The average score for each focus measure is obtained allowing the overall ranks to be calculated. This is performed separately for the HR and LR images, as summarised in Table 3.1.

Method	Score_{HR}	Rank_{HR}	Score_{LR}	Rank_{LR}
Daugman	7.500	4	19.625	10
Energy Laplace	6.500	3	3.000	1
Kang	13.375	7	15.000	8
ML	17.125	9	11.875	6
SML	16.375	8	12.375	7
Tenengrad	2.125	1	7.500	4
Wavelet 1	19.875	10	11.750	5
Wavelet 2	6.125	2	6.875	3
Wavelet 3	9.875	5	4.625	2
Wei	11.125	6	17.375	9

Table 3.1: Evaluation of focus assessment methods, where the subscripts denote the types of images processed.

Table 3.1 shows that the focus assessment method which gives the most suitable focus measure for the HR images is the Tenengrad method, followed by the wavelet 2 method and the Energy Laplace method. For the LR images the highest ranked focus assessment method is the Energy Laplace, followed by Wavelet 3 and Wavelet 2. Thus, there is no focus assessment method that is best for both sets of images.

The Tenengrad method being based on the Sobel edge detection kernels means that often background edges generate fairly high focus values and the small kernel size results in some discontinuities along object boundaries. The Energy Laplace method is also ranked highly for both sets of images, but again the small kernel size and squaring of the function places a very high emphasis on edges. All of the focus assessment methods that involve a squaring of values rank better than those that do not. This is because the squaring of focus values increases the ratio between the high and low focus regions, and reduces the standard deviation of the background pixel values. However this also creates a greater variance in the focus values of the object pixels that can result in discontinuities in the object boundary

as well as weaker values for the areas within an object boundary, i.e., object edges are amplified while most other areas are attenuated. This makes the segmentation of the OoI a more challenging prospect.

The Wavelet 3 method is ranked highly in the focus assessments for the LR images and the nature of the wavelet transform means that the method can be adapted for HR images. We therefore propose in Section 3.6 a focus assessment based on the Wavelet 3 method that provides a measure suitable for use with active contours on any image resolution.

3.5 Focus Assessment and Image Resolution

It is important to note the effect that image resolution, or image size, has on focus assessments. In Chapter 2, DoF is defined as the region for which the diameter of the CoC is less than the resolution of the display medium - in this case the resolution of the image. By reducing the resolution of the image, the diameter of the CoC becomes smaller when compared to pixel size. This means that lower resolution images are likely to return higher values when their focus is assessed.

The effect of reducing the resolution of the image on focus assessments is shown in Figure 3.12. As an example, Wei's focus assessment method is used on this low DoF image. The distance between the background and the OoI is relatively small. The focus assessment has been performed on the original 2816×2112 image, and then on a half scale (1408×1056), a quarter scale (704×528) and an eighth scale (352×264) version of the image. As the resolution of the image is decreased, it can be seen that the background foliage gradually becomes more prominent in the focus assessments, until it is impossible to distinguish the OoI from the background.

Magnifying a portion of the image allows us to see more clearly the effect that reducing resolution has on background contours. Figure 3.13 shows a portion of the image that has been magnified. One magnification is of the image at full scale (as shown in Figure 3.13(a)) whilst another is the same section, but magnified from an image an eighth the size of the original (as shown in Figure 3.13(b)). It can been seen that what was once a smooth part of the background has become a high frequency feature. The resolution has been reduced to such an extent that instead of the edge transitioning smoothly, there is now a significant contrast between adjacent pixels. Background edges will thus be picked up in the focus assessment and will make it difficult or impossible to segment the OoI. It should also be noted that too high a resolution can also be problematic as textures and object contours can also be very smooth, transitioning across many pixels. By selecting a wavelet based method, a

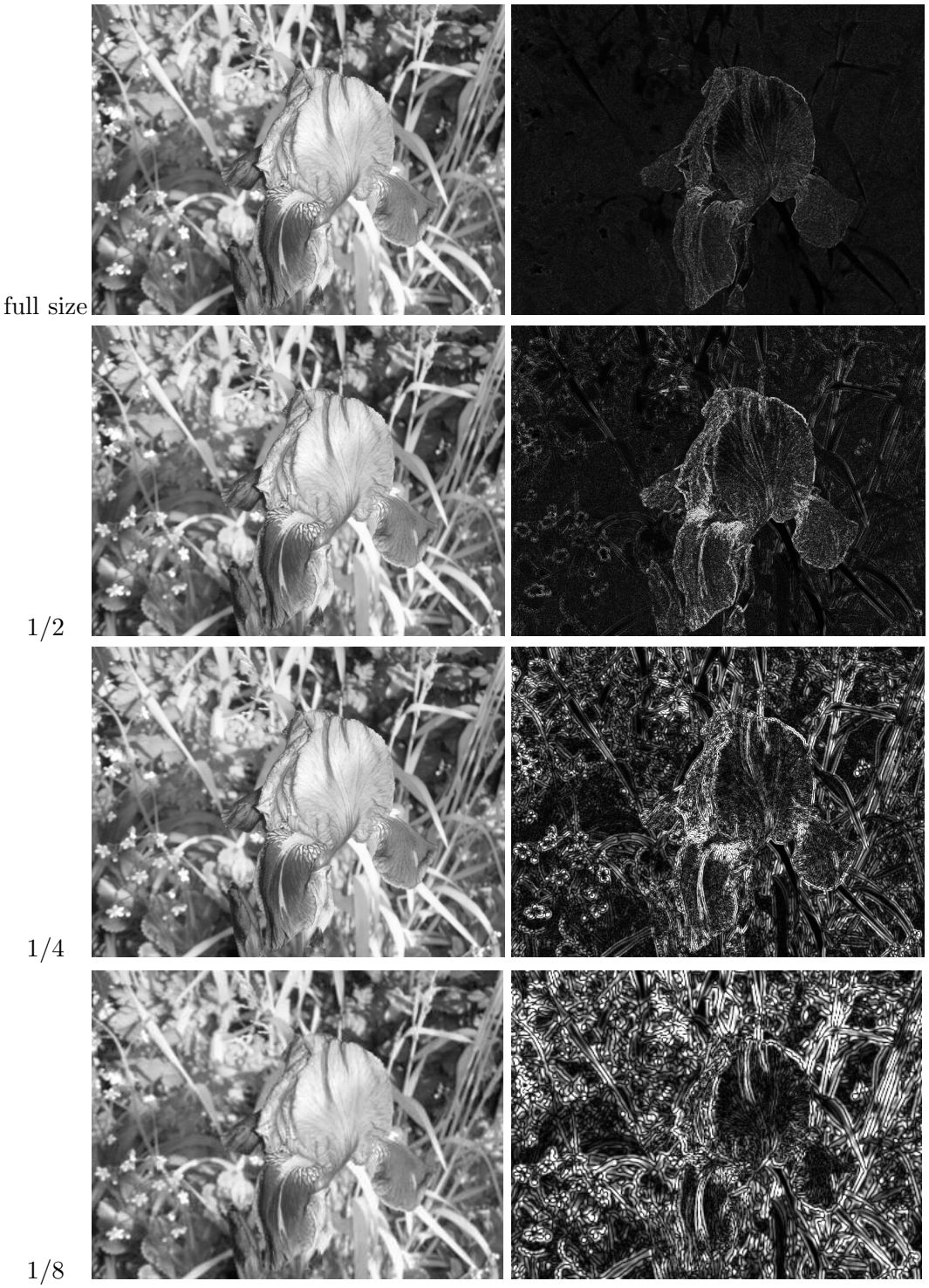


Figure 3.12: Effects of image resolution on focus assessments.

level of decomposition can be chosen that suits the image to be segmented.

(a) (b)

Figure 3.13: The effect of a reduction in image resolution on background contours.

3.6 Proposed Focus Assessment Method

A wavelet based focus measure which reflects the strength of the high frequency details is proposed in [Yang and Nelson, 2003a]. The method considers the first level of wavelet decomposition only and is given by (3.15). To extend the method for use with images of any resolution, (3.15) is modified to

$$F(x,y) = |DH(x,y) - \mu_{DH}| + |DV(x,y) - \mu_{DV}| + |DD(x,y) - \mu_{DD}|, \qquad (3.19)$$

where DH, DV, and DD are the reconstructed detail coefficients (horizontal, vertical and diagonal) for the wavelet decomposition of the image at a level N, and μ_{DH}, μ_{DV} and μ_{DD} are the mean values of each reconstructed subband. The modulus is taken as opposed to squaring the value as in Equation 3.15 to avoid object boundaries from becoming too dominant in the focus map.

For very high resolution images, taking the detail coefficients from further levels of decomposition aids in segmentation. This is because texture changes and contours of focused OoIs in high resolutions will transition over a number of pixels. Pixels will therefore contrast significantly less with their neighbours than if the image scale were smaller, leading to smaller values being returned in the focus assessment. By taking the detail coefficients from a lower level where this is the case, the performance of segmentation algorithms can be improved as it will be significantly easier to differentiate the object from the background.

The focus assessment method and the subsequent segmentation method are

intrinsically linked. The segmentation method was therefore developed simultaneously with the focus assessment method. This allowed the methods to be somewhat tailored to each other. An initial version of the object segmentation method presented in Chapter 4 based on the active contours algorithm was utilised to help determine the appropriate level of wavelet decomposition.

The focus assessment using (3.19) is first performed at level 1, i.e, $N = 1$. If the standard deviation (Std) of the image's focus values is below a threshold T, i.e., there is no significant difference between the background and object pixels, the process is then repeated at $N = 2$. If the Std is also below T at this level, the detail coefficients will be taken from the level 3 wavelet decomposition. This is unlikely to occur in images that are not of high resolution. If the standard deviation of the focus map is still beneath the threshold it is assumed that the OoI is either weakly textured or has a very small area and the values at $N = 3$ are used. Table 3.2 shows data used to determine the optimum value for the threshold, i.e., $T = 0.0105$. Comparing the Std of focus values with the percentage of correctly segmented pixels (found using manually generated ground truths) using the active contours algorithm enables the threshold to be chosen experimentally.

Image	Level	Image Std	Correctly Segmented
Flower	1	0.0069	35.7
	2	0.0119	98.7
	3	0.0165	97.3
	4	0.0249	91.9
Soft Toy	1	0.0074	95.5
	2	0.013	98.9
	3	0.0138	98.8
	4	0.0142	97.2
Watch	1	0.0047	34.6
	2	0.0100	91.2
	3	0.0150	99.3
	4	0.0209	97.4
Plant	1	0.0057	28.1
	2	0.0085	36.1
	3	0.0105	98.9
	4	0.0173	98.1

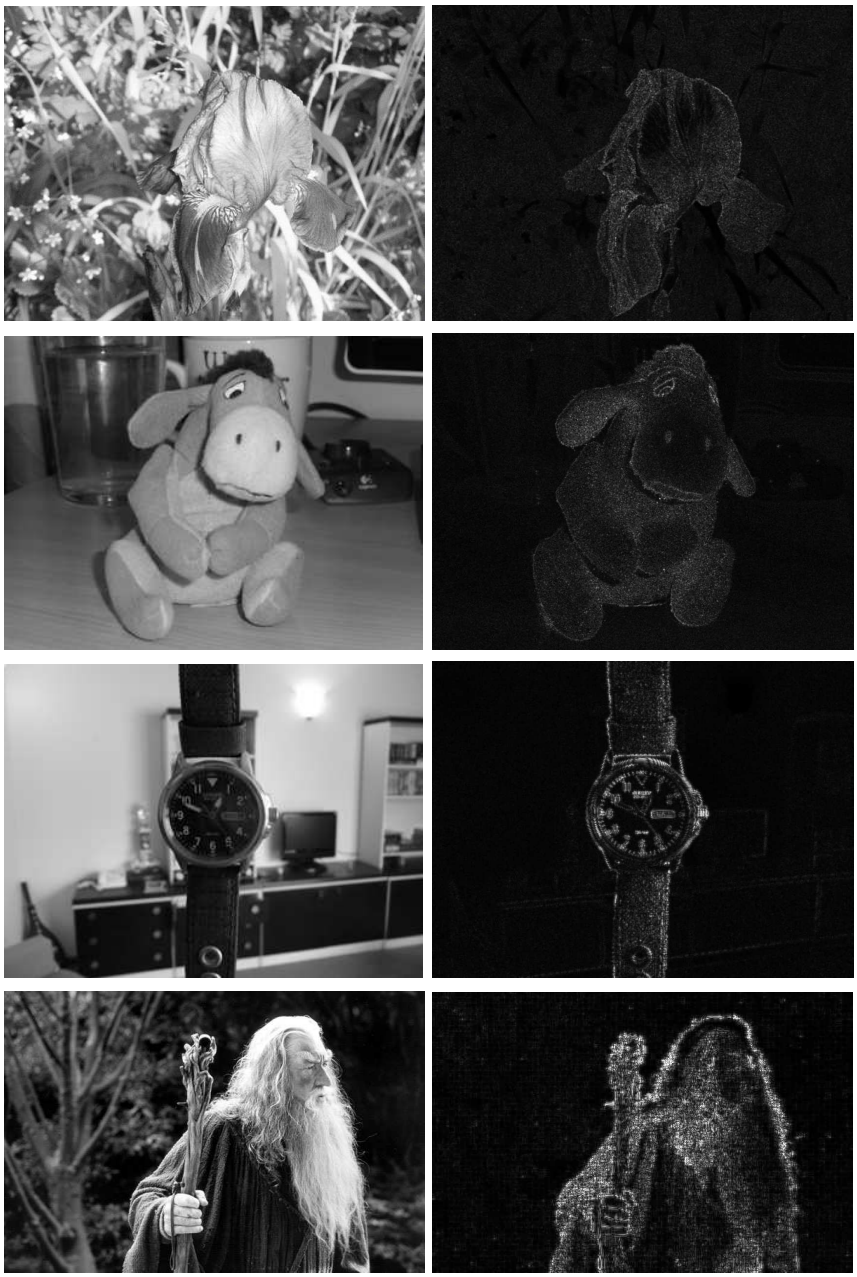
Table 3.2: Percentage of pixels correctly segmented compared to level of wavelet decomposition and Std of focus values for the four HR test images.

3.7 Results

In this section a few example focus assessments are performed using the proposed method. More detailed results and analysis of the overall object segmentation method are given in Chapter 4. Figure 3.14 shows four example low DoF images and the focus intensity maps generated using the multiscale wavelet based method. The high intensity parts of the image correspond to areas of the original image that are in focus, with the highest responses (corresponding to the brightest points) being due to the focused edges or strong textures. It can be seen from these examples that background objects have been attenuated and are not prominent in the focus intensity map. This creates a strong starting point for segmentation techniques.

3.8 Conclusion

In this chapter a number of focus assessment methods have been presented and their suitability for differentiating a focused OoI from a defocused background evaluated. They are compared on two measures. One, that the methods should return a higher average intensity for object regions than background regions, and two, that the background regions should be relatively homogeneous. It is shown that no single method is suitable for performing focus assessments for all image resolutions. Thus, a multiscale wavelet method is proposed which takes its cue for which level of wavelet decomposition to use from the Std of the pixels' focus intensities. The method is shown to be capable of differentiating focused and defocused regions from a variety of test images.



(a) (b)

Figure 3.14: Focus assessment of example images with a flower, a soft toy, a suspended watch and a wizard as the OoIs: (a) the image; and (b) focus values of individual pixels, i.e., focus energy map, where the brightness of a point in the map is proportional to its focus value (the images have their intensity enhanced by a factor of three for clarity in the display).

Chapter 4

Object Segmentation

4.1 Introduction

Object segmentation is the process of separating an image into two areas, the background and the OoI to form a binary segmentation. This is a difficult task as image properties can vary hugely from scene to scene. The previous chapter constrains the type of image to be segmented to those with a low DoF, a focus assessment method is used to generate a focus map, the goal being to provide some way to differentiate the focused OoI from the defocused background. This chapter presents the selected method for performing the binary segmentation on the focus map.

For this purpose an active contours algorithm is selected, and a narrow-band level sets implementation of the Active Contours without Edges [Chan and Vese, 2001] is implemented. The rationale being that given a suitable initialisation contour, the algorithm will be able to detect the focused regions without being reliant on sharp contrasts in edges and continuous boundaries. Adopting a narrow band method is not only more computationally efficient, but it also prevents interior contours from being generated spontaneously, thus allowing OoI with weak textures or homogeneous regions to still be segmented successfully. The focus map is used to generate a robust initial contour using a grid based method.

The remainder of this chapter is organised as follows. In Section 4.2 some of the related work in segmenting low DoF images is discussed. Three interactive segmentation algorithms that are used as the basis of a comparison in the results section of this chapter are described. Section 4.3 provides a detailed description of the classical active contours model and its level set representation. The Active Contours without Edges model is introduced and its level set representation and relation to the Mumford-Shah function presented. Also, regularisation, numerical

approximation, and discretization of the model is introduced. Section 4.4 presents the chosen implementation of active contours (the Sparse Field Method implementation of Active Contours without Edges). The implementation is applied to the segmentation of focus maps in Section 4.5 and finally results of the proposed segmentation method are shown in Section 4.6. The Chapter is concluded in Section 4.7.

4.2 Related Work

4.2.1 Low Depth of Field Methods

A number of methods for segmenting objects from defocused backgrounds in images with low DoF have been proposed and a brief overview of them is provided in this section.

Tsai's Method

An unsupervised method for segmenting in-focus objects in complex backgrounds is proposed in [Tsai and Wang, 1998]. The defocus of edge pixels is measured using a moment-preserving principle, and a three-stage edge linking process consisting of dilation, thinning and line linking, bounds the pixels corresponding to the object in focus. This is followed by a region filling procedure which eliminates all background pixels and retains the regions of in-focus objects defined by the boundaries. The method produces jagged segmentations due to the use of straight line segments to connect broken edges.

Wang's Method

A multiscale approach using high-frequency wavelet coefficients and their statistics to classify blocks in an image is proposed in [Wang et al., 2001]. This method has the advantages of fast processing time and does not rely on connecting object boundaries. However the segmentation results are not smooth due to the use of blocks and have a significant percentage of misclassified pixels.

Kim's Methods

A method which segments an image into two regions based on their higher order statistics is proposed in [Kim, 2005]. A higher order statistics map is first created and then simplified using a morphological filter. Image segmentation is obtained by region merging and thresholding. This method is fast but produces incorrect

segmentation if a large in-focus smooth region is present in the image. The method is improved in [Kim et al., 2007], where again a higher order statistics map is created. An approximate block based segmentation is then performed and used as a basis for a more accurate pixel-level segmentation. The method is also extended for the segmentation of an object in a video sequence. The method performs well on test images with blurred or relatively simple backgrounds.

Li's Method

The three-stage method in [Li and Ngan, 2007] first generates a saliency map using a reblurring model. The salient regions are then smoothed and accentuated using morphological filtering. This allows a trimap of object, background and ambiguous regions to be created. In the third stage the object boundaries are extracted using an adaptive error control matting scheme. The method can segment complex shapes such as text, but most of the backgrounds are relatively simple or have large focus differentials to the object.

Liu's Method

An automated segmentation method which uses a focus energy map estimated by measuring the differences in high frequency components between the focused object and un-focused background to create a region and boundary saliency map is proposed in [Liu et al., 2010]. A boundary linking algorithm is applied to obtain closed region and boundary masks. This is followed by image matting on the generated trimap to obtain the line segmented object. The method performs well with images that have a large focus differential or uniform background, but poorly with cluttered backgrounds.

4.2.2 Methods for Comparison

In this section, three interactive segmentation methods are discussed. These techniques are used to compare the results of the proposed algorithm with some of the most prominent image segmentation methods. The first two methods are selected as they have been shown to be the most robust and accurate in an evaluation of various different interactive segmentation techniques [McGuiness and O'Conor, 2010]. Finally, as the proposed method is completely autonomous, the Grabcut algorithm is chosen to provide a method that has as little user interaction as possible for comparison.

Binary Partition Trees

Interactive segmentation using binary partition trees (BPTs) was first proposed in [Salembier and Garrido, 2000] and further improved upon in [Adamek, 2006]. The method transforms a hierarchical region based segmentation into a segmentation of object and background by using user interaction to separate and merge regions in the tree. Any automatic algorithm which performs segmentation with the output in the from of a BPT can be used. The user assigns pixels in the image as object and background, which in turn labels the corresponding leaf nodes in the tree. These labels are propagated up the tree, assigning the same value to each node until a conflict occurs, i.e., a node with differently labelled children. The node is marked as conflicting and the algorithm moves on to the next leaf node. This is repeated for every user-marked leaf in the tree. In the next step, the algorithm passes every non-conflicting node and gives its label to its children. It is proposed in [Adamek, 2006] that unclassified regions are assigned the labels of adjacent regions. In cases where two or more regions are adjacent, the one with the shortest distance is chosen.

Interactive Graph Cuts

The graph cuts method considers an image to be a graph with each pixel being a node [Boykov and Jolly, 2001]. The user specifies some initial object and background seeds, and the method categorises the rest of the pixels as either object or background using max-flow/min-cut algorithms. The method has the advantage of providing robust segmentation even if the foreground and background colour distributions are not well separated.

GrabCut

GrabCut [Rother et al., 2004] extends the graph cuts method. With this method the user draws an initial area around the OoI to indicate which pixels are object and which are background. Colour information is then obtained, the graph reweighted and graph cuts is used to obtain a refined segmentation. This is repeated and after a specified number of iterations the user can re-define the foreground or background pixels to refine the segmentation. This method improves the results of the graph cuts method whilst reducing the amount of user input required.

4.3 Active Contours

In Chapter 2 the basics of active contours were described, introducing the concepts of an evolving curve and its level set representation. In this section the details of active contours are presented by first describing the classical snakes model and its level set implementation. The reliance on a clear gradient change on the object boundaries is noted and Chan-Vese's Active Contours without Edges [Chan and Vese, 2001] is introduced to counter this. Active contours' relation to a specific case of the Mumford-Shah functional [Mumford and Shah, 1989] is shown and then its level set implementation is given. This is followed by the regularisation, numerical implementation and discretization of the model and the framework of the algorithm. Finally in order to improve computational efficiency, Whitaker's Sparse Field implementation of active contours is utilised [Whitaker, 1998].

4.3.1 Basic Active Contour Model

The general principle behind the classical active contours model, often referred to as 'snakes', is to evolve a curve to detect objects within an image u_0 by minimising an energy function. The curve evolves based on both user imposed constraints and constraints imposed by u_0. An initial curve will move towards an object, stopping on the object boundary. In the traditional active contours model an edge detector is used as part of the model to prevent the curve from further evolving once it has reached the object boundary [Kass et al., 1988; Caselles et al., 1993, 1997].

Consider 2D space, $\mathbb{R}^2$, with Ω as a bounded open subset of $\mathbb{R}^2$, with a boundary denoted by $\partial\Omega$. The map $u_0 : \bar{\Omega} \to \mathbb{R}$ takes a point in the region $\bar{\Omega}$ (the region including the boundary) to a corresponding single value in $\mathbb{R}$. This is illustrated in Figure 4.1. The model is $\inf_C J_1(C)$, which is the minimum value of $J_1(C)$ considered for all possible curves C, i.e.,

$$J_1(C) = \alpha \int_0^1 |C'(s)|^2 ds + \beta \int_0^1 |C''(s)| ds - \lambda \int_0^1 |\nabla u_0(C(s))|^2 ds, \tag{4.1}$$

where α, β and λ are positive parameters. The first two terms of Equation 4.1 represent the internal energy of the contour and control its smoothness. The third represents the external energy and acts as the edge detector which draws the curve towards the object boundary/edge. The 2D gradient (i.e., in both x and y directions) of the image u_0 is denoted by ∇u_0. As we are trying to minimise the energy function $J_1(C)$, this will be when the negative term $-\lambda \int_0^1 |\nabla u_0(C(s))|^2 ds$ is large. This corresponds to the points with the largest gradient, i.e., at the object boundary. The

first two terms help maintain smoothness in the curve, preventing it from becoming too jagged.

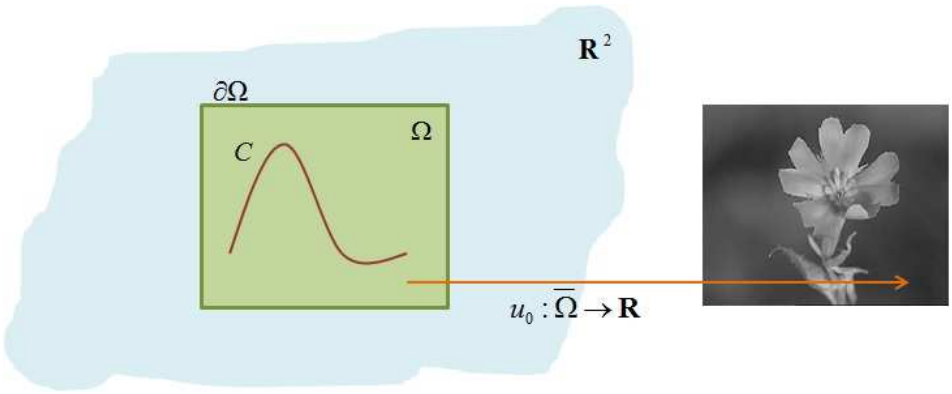


Figure 4.1: Framework of the classical snakes model.

A general edge detector is a positive decreasing function g, which depends on the gradient of the image, i.e., g returns increasingly small values as the gradient gets bigger, i.e.,

$$\lim_{z\to\infty} g(z) = 0 \tag{4.2}$$

One example of an edge detector is

$$g(|\nabla u_0(x,y)|) = \frac{1}{1+|\nabla G_\omega(x,y) * u_0(x,y)|^p},\ p \geq 1, \tag{4.3}$$

where $G_\omega * u_0$ is a smoothed version of u_0, obtained by convolving the image u_0 with the Gaussian $G_\omega(x,y) = \omega^{-1/2} e^{-|x^2+y^2|/4\omega}$. The edge detection function $g(|\nabla u_0|)$ is positive in homogeneous regions and zero at edges.

4.3.2 Level Set Formulation of the Active Contours Model

Instead of being represented as a parametrised curve, C can be represented as the zero level set of a Lipschitz (continuous) function ϕ, i.e., $C = \{(x,y)|\phi(x,y) = 0\}$, and the evolution of the curve is given by the zero-level set of the function $\phi(t,x,y)$, where t is time. The evolving curve can be found by solving the differential equation

$$\frac{\partial\phi}{\partial t} = |\nabla\phi|S \tag{4.4}$$

where S is the speed at which C is evolving in in the normal direction with the initial condition $\phi_0(x,y) = \phi(0,x,y)$. If the speed S and the initial contour are known, then the solution of the differential equation is $\phi(t,x,y)$ for all values of t.

An example of a particular case of curve evolution is motion by mean curvature [Osher and Sethian, 1988]. In this case the speed S is defined as the curvature of the zero level set curve ϕ passing through (x, y), i.e., $S = \text{div}\nabla\phi(x, y)/|\nabla\phi(x, y)|$. Thus Equation 4.4 becomes

$$\left.\begin{array}{ll} \frac{\partial\phi}{\partial t} = |\nabla\phi|\text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right), & t \in (0,\infty), x \in \mathbb{R}^2 \\ \phi(0, x, y) = \phi_0(x, y), & x \in \mathbb{R}^2 \end{array}\right\} \tag{4.5}$$

To form a geometric active contours model, an edge stopping term is introduced to the curve evolution by mean curvature so that the motion tends to 0 on an object's boundary [Caselles et al., 1993]. The model is therefore

$$\left.\begin{array}{l} \frac{\delta\phi}{\delta t} = g(|\nabla u_0|)|\nabla\phi\left(\text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) + \nu\right) \text{ in } (0,\infty)\times\mathbb{R}^2 \\ \phi(0, x, y) = \phi_0(x, y) \text{ in } \mathbb{R}^2 \end{array}\right\}, \tag{4.6}$$

where $g(|\nabla u_0|)$ is the edge function with $p = 2$, $\nu \geq 0$ is constant, and ϕ_0 is the initial level set function. The term ν is used so that $\partial\phi/\partial t$ is only 0 when the edge function $g(|\nabla u_0|)$ is 0.

4.3.3 Active Contours Without Edges

In this section the basis for the model used in this thesis is discussed. As stated in Sections 4.3.1 and 4.3.2 the classical snakes model depends on a edge function g to stop the curve from further evolving when it is on an object boundary, due to large image gradient $|\nabla u_0|$. These models can only detect objects with edges clearly defined by a gradient which can lead to the curve passing through the object boundary if it is relatively weak.

Chan and Vese propose a model which is not based on the gradient of the image [Chan and Vese, 2001], but instead is based on the Mumford-Shah Functional [Mumford and Shah, 1989]. This allows their method, called Active Contours without Edges, to detect objects with smooth boundaries or even discontinuities. This is a significant advantage when the algorithm is used in this thesis to segment focus maps.

Consider the framework as shown in Figure 4.2, where C is the evolving contour in Ω, defined as the boundary $\partial\omega$ of the open subset ω of Ω, i.e., $\omega \subset \Omega$ and $C = \partial\omega$. In the following notation, $inside(C)$, the region within the curve, corresponds to the region ω and $outside(C)$ represents the region outside the contour, denoted by $\Omega \setminus \bar{\omega}$.

A basic case is assumed whereby the image u_0 has two fairly homogeneous re-

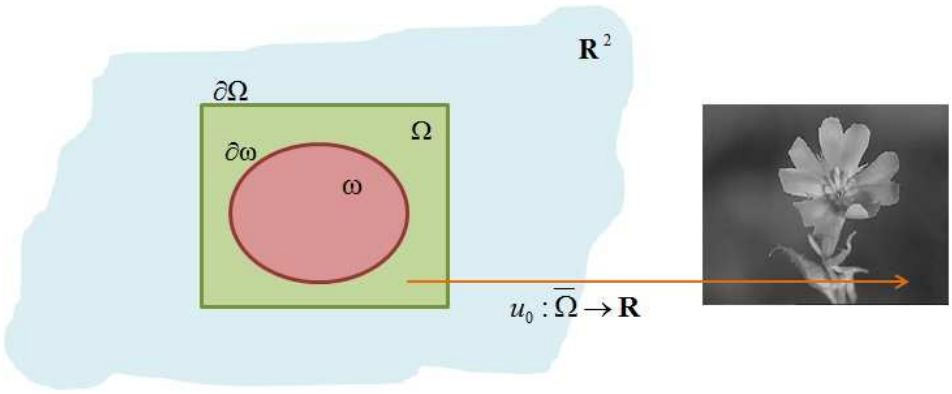


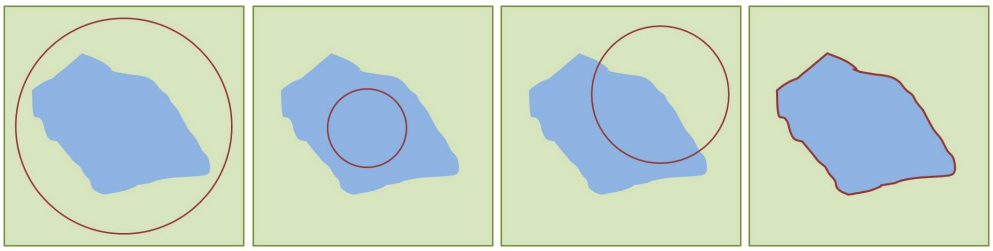
Figure 4.2: Framework for Active Contours without Edges.

gions of distinct values u_0^i and u_0^o. The region with value u_0^i is assumed to correspond to the object to be segmented and whose boundary is denoted by C_0. Therefore, $u_0 \approx u_0^i$ inside the object boundary C_0, and $u_0 \approx u_0^o$ outside of C_0. The fitting function is

$$F_1(C) + F_2(C) = \int_{inside(C)} |u_0(x,y) - c_1|^2 dxdy \; + \int_{outside(C)} |u_0(x,y) - c_2|^2 dxdy, \tag{4.7}$$

where C is the varying contour/curve, and c_1 and c_2 are the average values of u_0 inside and outside of the curve C, respectively. It can be seen that when $C = C_0$, i.e., when the curve is on the object boundary, then the fitting function is minimised. Figure 4.3 illustrates all the possible cases in fitting a curve onto an object. If the curve C is outside of the object then $F_1(C) > 0$ and $F_2(C) \approx 0$. If the curve is inside the object then $F_1(C) \approx 0$ and $F_2(C) > 0$. If the curve goes through both object and background then $F_1(C) > 0$ and $F_2(C) > 0$. Finally if the curve is on the object boundary then $C = C_0$. In this case $F_1(C) \approx 0$ and $F_2(C) \approx 0$, therefore $F_1(C) + F_2(C)$ is minimised, i.e., the fitting function is only minimised when the curve is on the object boundary.

In the Active Contours without Edges model, the goal is to minimise the fitting term given by 4.7 with some added regularising terms, which add weights to the importance of the length of the curve (C) and the area within the curve. The



(a) (b) (c) (d)

Figure 4.3: All possible cases in fitting a curve onto an object: (a) the curve is outside of the object; (b) the curve is inside the object; (c) the curve contains both object and background; (d) the curve is on the object boundary.

following energy functional is therefore introduced

$$\begin{aligned} F(c_1, c_2, C) &= +\mu \cdot \text{length}(C) + \nu \cdot \text{area}(inside(C)) \\ &\quad +\lambda_1 \int_{inside(C)} |u_0(x,y) - c_1|^2 dxdy \\ &\quad +\lambda_2 \int_{outside(C)} |u_0(x,y) - c_2|^2 dxdy, \end{aligned} \tag{4.8}$$

where μ, ν, λ_1 and λ_2 are fixed parameters with $\mu \geq 0$, $\nu \geq 0$, $\lambda_1 > 0$ and $\lambda_2 > 0$. The standard implementation of the model sets $\lambda_1 = \lambda_2 = 1$ and $\nu = 0$, thus removing the area component from the model. The minimisation problem considered is hence

$$\inf_{c_1, c_2, C} F(c_1, c_2, C). \tag{4.9}$$

4.3.4 Relation with the Mumford-Shah Functional

The active contours can be shown to be a reduced form of the Mumford-Shah functional, known as the minimal partition problem, assuming the same basic case as previously stated in Section 4.3.3. The Mumford-Shah function for segmentation is given by

$$\begin{aligned} F^{MS}(u, C) &= \mu \cdot \text{Length}(C) \\ &\quad +\lambda_1 \int_{\Omega} |u_0(x,y) - u(x,y)|^2 dxdy \\ &\quad + \int_{\Omega \setminus C} |\nabla u(x,y)|^2 dxdy, \end{aligned} \tag{4.10}$$

where, as with the active contour model, $u_0 : \bar{\Omega} \to \mathbb{R}$ and μ and λ are positive constants. The function u is a solution image, which is found by minimising the

Mumford-Shah Functional and is formed by smooth regions with distinct boundaries defined by C. The minimal partition problem is to restrict F^{MS} to the piecewise constant function u. This means that the regions inside and outside of C in the solution image u will be constant, i.e.,

$$u = \begin{cases} \text{average}(u_0) \text{ inside } C \\ \text{average}(u_0) \text{ outside } C \end{cases} \tag{4.11}$$

As the standard implementation of the Active Contours without Edges model removes the area term by setting $\nu = 0$ and sets $\lambda_1 = \lambda_2 = 1$, this shows the model to be a particular case of the minimum partition problem.

4.3.5 Level Set Formulation of the Method

As with classical snakes, Active Contours without Edges can be formulated as a level set method [Osher and Sethian, 1988]. The contour C is again represented by the zero level set of a Lipschitz function ϕ. Positive values above the zero level set correspond to points inside the curve C, whereas negative values are outside C, i.e.,

$$\left.\begin{array}{l} C = \partial\omega = \{(x,y) \in \Omega : \phi(x,y) = 0\} \\ \text{inside}(C) = \omega = \{(x,y) \in \Omega : \phi(x,y) > 0\} \\ \text{outside}(C) = \Omega \setminus \bar{\omega} = \{(x,y) \in \Omega : \phi(x,y) < 0\}. \end{array}\right\} \tag{4.12}$$

The Heavyside function $H(z)$ allows the reformulation of Equation 4.8 to perform the integral over the whole of Ω rather than the two seperate regions outside and inside the contour C [Zhao et al., 1996]. The Heavyside function H and its derivative the Dirac measure (equivalent to a perfect impulse) δ_0 are given by

$$H(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases} \quad , \quad \delta_0(z) = \frac{d}{dz} H(z). \tag{4.13}$$

The length and area are

$$\begin{aligned} \text{Length}\{\phi = 0\} &= \int_\Omega |\nabla H(\phi(x,y))| dxdy \\ &= \int_\Omega \delta_0(\phi(x,y)) |\nabla \phi(x,y) dxdy, \\ Area\{\phi \geq 0\} &= \int_\Omega H(\phi(x,y)) dxdy, \end{aligned} \tag{4.14}$$

and the other terms in Equation 4.8 are rewritten as

$$\begin{aligned}\int_{\phi\geq 0}|u_0(x,y)-c_1|^2dxdy &= \int_{\Omega}|u_0(x,y)-c_1|^2H(\phi(x,y))dxdy, \\ \int_{\phi<0}|u_0(x,y)-c_2|^2dxdy &= \int_{\Omega}|u_0(x,y)-c_2|^2(1-H(\phi(x,y)))dxdy.\end{aligned} \tag{4.15}$$

Note that

$$\int_{\phi\geq 0}|u_0(x,y)-c_1|^2dxdy = \int_{\phi>0}|u_0(x,y)-c_1|^2dxdy. \tag{4.16}$$

This is because

$$\int_{\phi\geq 0}|u_0(x,y)-c_1|^2dxdy = \int_{\phi>0}|u_0(x,y)-c_1|^2dxdy + \int_{\phi=0}|u_0(x,y)-c_1|^2dxdy \tag{4.17}$$

and the second term of the right hand side of the equation is 0 since the contour itself has a Lebesgue measure of 0. The energy $F(c_1,c_2,\phi)$ can therefore be expressed as

$$\begin{aligned}F(c_1,c_2,\phi) &= \mu\int_{\Omega}\delta(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ &\quad +\nu\int_{\Omega}H(\phi(x,y))dxdy \\ &\quad +\lambda_1\int_{\Omega}|u_0(x,y)-c_1|^2H(\phi(x,y))dxdy \\ &\quad +\lambda_2\int_{\Omega}|u_0(x,y)-c_2|^2(1-H(\phi(x,y)))dxdy.\end{aligned} \tag{4.18}$$

In the particular case of the minimum partition problem as described in Section 4.3.4, the solution image u can also be described using the Heavyside function and a level set formulation as

$$u(x,y) = c_1H(\phi(x,y)) + c_2(1-H(\phi(x,y))), (x,y)\in\bar{\Omega}. \tag{4.19}$$

For a given ϕ and assuming that both the contour interior and exterior are non-empty, the average values of u_0 inside and outside of the curve, denoted by c_1 and c_2 respectively, are computed using

$$c_1(\phi) = \frac{\int_{\Omega}u_0(x,y)H(\phi(x,y))dxdy}{\int_{\Omega}H(\phi(x,y))dxdy} \tag{4.20}$$

and

$$c_2(\phi) = \frac{\int_{\Omega}u_0(x,y)(1-H(\phi(x,y)))dxdy}{\int_{\Omega}(1-H(\phi(x,y)))dxdy} \tag{4.21}$$

I.e, the values c_1 and c_2 for a given ϕ are

$$\begin{cases} c_1(\phi) &= \text{average}(u_0) \text{ in } \{\phi \geq 0\} \\ c_2(\phi) &= \text{average}(u_0) \text{ in } \{\phi < 0\} \end{cases} \tag{4.22}$$

It should be noted that the existence of a solution to the minimisation of the energy $F(c_1, c_2, C)$ is expected, and has been proven to exist [Mumford and Shah, 1989; Maso et al., 1992].

4.3.6 Regularisation of the Model

In order to make the Euler-Lagrange equation for the unknown function ϕ computationally possible, regularised versions of the Heavyside Function and a Dirac measure must be used. The functions H and δ_0 are thus replaced by H_ϵ and δ_ϵ which respectively tend to H and δ_0 as ϵ tends to 0. Any regularisation of H must be differentiable as $\delta_\epsilon = H'_\epsilon$. The new energy, F_ϵ is therefore represented by the regularised functional

$$\begin{aligned} F_\epsilon(c_1, c_2, \phi) &= \mu \int_\Omega \delta_\epsilon(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ &\quad +\nu \int_\Omega H_\epsilon(\phi(x,y))dxdy \\ &\quad +\lambda_1 \int_\Omega |u_0(x,y) - c_1|^2 H_\epsilon(\phi(x,y))dxdy \\ &\quad +\lambda_2 \int_\Omega |u_0(x,y) - c_2|^2 (1 - H_\epsilon(\phi(x,y)))dxdy. \end{aligned} \tag{4.23}$$

Fixing c_1 and c_2, and minimising F_ϵ with respect to ϕ allows the Euler-Lagrange equation for ϕ to be found. The solution of the Euler-Lagrange equation is ϕ in terms of x, y *and* t, i.e., ϕ can be found for any given time t by solving

$$\begin{aligned} \frac{\partial\phi}{\partial t} &= \delta_\epsilon(\phi)\left[\mu \text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \nu - \lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2\right] = 0 \\ &\quad \text{in } (0,\infty)\times\Omega, \\ &\quad \phi(0,x,y) = \phi_0(x,y) \text{ in } \Omega, \\ &\quad \frac{\delta_\epsilon(\phi)}{|\nabla\phi|}\frac{\partial\phi}{\partial\vec{n}} = 0 \text{ on } \partial\Omega, \end{aligned} \tag{4.24}$$

where $\vec{n}$ is the exterior normal to the boundary $\partial\omega$ and $\partial\phi/\partial\vec{n}$ denotes the derivative of ϕ at the boundary in the normal direction.

4.3.7 Numerical Approximation of the Model

There are a number of possible regularisations for the Heavyside function H when implementing the model. One such function is [Zhao et al., 1996]

$$H_{1,\epsilon}(z) = \begin{cases} 1 \text{ if } z > \epsilon \\ 0 \text{ if } z < -\epsilon \\ \frac{1}{2}\left[1 + \frac{z}{\epsilon} + \frac{1}{\pi}\sin\left(\frac{\pi}{\epsilon}\right)\right] \text{ if } |z| \leq \epsilon. \end{cases} \tag{4.25}$$

Chan and Vese propose a different regularisation given by

$$H_{2,\epsilon}(z) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{z}{\epsilon}\right)\right). \tag{4.26}$$

Taking the Dirac measure as $\delta_\epsilon = H'_\epsilon$, using $H_{1,\epsilon}$ and $\delta_{1,\epsilon}$ can sometimes result in the method finding a local minimiser the of the energy whereas using $H_{2,\epsilon}$ and its corresponding Dirac measure $\delta_{2,\epsilon}$ obtains a global minimiser, regardless of the initial contour. This allows the method to automatically detect interior contours.

4.3.8 Discretization of the model

Having made a numerical approximation of the model, the next step is to perform a discretization of the model to make it suitable for implementation. A finite difference scheme is used to discretize the model. For an image of $M \times M$ pixels, the spatial step is denoted by h and the time step by Δt. Thus the grid points are $(x_i, y_j) = (ih, jh)$ for $1 \leq i, j \leq M$. The function $\phi(t, x, y)$ is approximated by its discrete equivalent $\phi^n_{i,j} = \phi(n\Delta t, x_i, y_j)$, with $n \geq 0$. The initial contours are the same on all grid points, i.e., $\phi^0 = \phi_0$. The finite differences are defined as the difference between a given pixel and its four direct neighbours. Thus

$$\begin{aligned} \Delta^x_- &= \phi_{i,j} - \phi_{i-1,j}, \\ \Delta^x_+ &= \phi_{i,j} - \phi_{i+1,j}, \\ \Delta^y_- &= \phi_{i,j} - \phi_{i,j-1}, \\ \Delta^y_+ &= \phi_{i,j} - \phi_{i,j+1}. \end{aligned} \tag{4.27}$$

The method adopts the discretization of the divergence operator utilised in [Rudin et al., 1992] and the iterative algorithm from [Aubert and Vese, 1997]. For a given ϕ^n the average values inside and outside of the contour (i.e., respectively $c_1(\phi^n)$ and $c_2(\phi^n)$) are first computed using Equations 4.20 and 4.21. The differential in

Equation 4.24 to compute ϕ^{n+1} from ϕ^n can then be solved using

$$\begin{aligned}\frac{\phi_{i,j}^{n+1}-\phi_{i,j}^{n}}{\Delta t} &= \delta_h(\phi_{i,j}^n)\left[\frac{\mu}{h^2}\Delta_-^x\cdot\left(\frac{\Delta_+^x\phi_{i,j}^{n+1}}{\sqrt{(\Delta_+^x\phi_{i,j}^n)^2/(h^2)+(\phi_{i,j+1}^n-\phi_{i,j-1}^n)^2/(2h)^2}}\right)\right. \\ &\quad +\frac{\mu}{h^2}\Delta_-^y\cdot\left(\frac{\Delta_+^y\phi_{i,j}^{n+1}}{\sqrt{(\phi_{i+1,j}^n-\phi_{i-1,j}^n)^2/(2h)^2+(\Delta_+^y\phi_{i,j}^n)^2/(h^2)}}\right) \\ &\quad \left.-\nu-\lambda_1(u_{0,i,j}-c_1(\phi^n))^2+-\lambda_2(u_{0,i,j}-c_2(\phi^n))^2\right] \qquad (4.28)\end{aligned}$$

Iteration

The model is implemented iteratively as follows:

1. Initialise ϕ^0 using the initial contour, ϕ_0, $n=0$.
2. Compute $c_1(\phi^n)$ and $c_2(\phi^n)$ using Equations 4.20 and 4.21.
3. Solve Equation 4.28 to get ϕ^{n+1}.
4. Check the solution to see if it is stationary. If not then $n=n+1$ and repeat the steps from 2.

4.4 Sparse Field Implementation

One of the main disadvantages of level set active contour methods is that they are computationally intensive - a large amount of computations are needed to maintain ϕ as the contour C changes. In order to improve the speed with which level set methods run, a variety of narrow band algorithms have been proposed (for example, [Shi and Karl, 2005]). These methods reduce the computational complexity by only performing calculations near the zero level set since only the area of ϕ where $\phi(x,y)\approx 0$ are required to maintain ϕ as the contour changes. One such narrow band method is Whitaker's sparse field method (SFM) [Whitaker, 1998]. The method is chosen as it allows for an efficient yet accurate representation of ϕ.

One of the main disadvantages of narrow band methods is that new contours are only searched for in the vicinity of the level set, and so it is not possible for interior contours, or indeed new contours, to spontaneously appear. This does not prevent the initial contour from the usual operations of splitting, growing and merging to obtain a final solution. Whilst for many applications this may be a strong incentive not to use a narrow band method, it is actually considered an

advantage for the implementation proposed as, assuming a good initialisation for the active contours, it allows focused objects which are weakly textured or have large homogeneous regions which would not normally return high focus values in the focus assessment to be segmented successfully. This is because new contours are unable to spontaneously appear within the object boundary. The implementation of the SFM is described in this section and follows that given by [Lankton, 2009].

The implementation makes use of five *doubly-linked-lists* which list all the points (by giving the x and y location) that belong to the zero level set and the two sets on either side, i.e.,

$$\begin{array}{lcrcrl} L_0 & \rightarrow & [& -0.5 & , & 0.5 \ ] \\ L_{-1} & \rightarrow & [& -1.5 & , & -0.5 \ ] \\ L_1 & \rightarrow & [& 0.5 & , & 1.5 \ ] \\ L_{-2} & \rightarrow & [& -2.5 & , & -1.5 \ ] \\ L_2 & \rightarrow & [& 1.5 & , & 2.5 \ ]. \end{array} \tag{4.29}$$

Two arrays are also used and set to the size of the image. One is the ϕ array and the other a label array. The label array can only take one of 7 values, $\{-3, -2, -1, 0, 1, 2, 3\}$, and provides information on the status of each point, i.e.,

$$\begin{array}{rl} -3 & \text{Object pixel, not in any level set lists.} \\ -2 & \text{Object pixel, } L_{-2} \text{ level set.} \\ -1 & \text{Object pixel, } L_{-1} \text{ level set.} \\ 0 & \text{0 level set, } L_0. \\ 1 & \text{Background pixel, } L_1 \text{ level set.} \\ 2 & \text{Background pixel, } L_2 \text{ level set.} \\ 3 & \text{Background pixel, not in any level set lists.} \end{array} \tag{4.30}$$

Note that in the implementation of the model the sign for ϕ has been swapped, i.e., on the interior of the contour $\phi < 0$ and on the exterior $\phi > 0$. This is to ensure the first and second derivatives can be computed about the contour more easily.

4.4.1 SFM Initialisation

The initialisation is the first stage of the SFM. Given a binary image the initialisation process returns full arrays for the label map ϕ and the five lists representing the zero level set and the two sets on either side. The binary input image has values of 0 to represent the background, and 1 to represent the foreground/object. The initialisation process is as follows:

1. Each point in the label map is given the value -3 if it is object and 3 if not, similarly the values of ϕ are also set to -3 or 3.

2. For each object point which has a background point as an immediate neighbour (i.e., adjacent horizontal or vertical pixel) with label 0, the corresponding point on the ϕ array is set to 0, and the point is added to the list representing the zero level set.

3. The +1 and -1 sets are then created. For each point in the zero level set, any immediate neighbours with label 3 are given the label 1, ϕ is set to 1, and these points are added to the +1 set. Any immediate neighbours with value -3 have the label replaced with -1, ϕ is set to -1, and the points are added to the -1 set list.

4. The same process is used with all the points in the -1 and +1 level sets to label the -2 and +2 level sets, set the corresponding points on the ϕ array and add the points to the +2 and -2 level set lists.

4.4.2 SFM Contour Evolution

Chan-Vese's energy proposed in the Active Contours without Edges method is then implemented (see Equation 4.18.) This gives the evolution equation

$$F = (u_0 - c_1)^2 - (u_0 - c_2)^2, \tag{4.31}$$

where u_0 is the image intensity at a specific point, and c_1 and c_2 are the average values inside and outside of the contour respectively. This is only computed along the zero level set (L_0) and is normalised such that $-0.5 < F < 0.5$ using

$$F = \frac{F}{|F_{max}| \times 0.4}. \tag{4.32}$$

The length constraint, determined by μ is then added to this to obtain F_2. As with Chan-Vese's implementation, an area constraint (ν) is not used. The array ϕ is then updated as follows. Five temporary lists are first defined which hold points with changing label, i.e.

$$\begin{aligned} S_0 &\rightarrow \text{points moving to } L_0 \\ S_{-1} &\rightarrow \text{points moving to } L_{-1} \\ S_1 &\rightarrow \text{points moving to } L_1 \\ S_{-2} &\rightarrow \text{points moving to } L_{-2} \\ S_2 &\rightarrow \text{points moving to } L_2 \end{aligned} \tag{4.33}$$

The following steps are then taken:

1. For each point in the zero level set, F_2 is added to the corresponding value in the ϕ array.

2. Any point whose value of ϕ is now outside of the range of the zero level set L_0 [-0.5 0.5] is moved from the set to one of two temporary lists, S_1 or S_{-1} to indicate whether the point will be changed to L_1 or L_{-1}.

3. The first level sets, L_1 and L_{-1}, are updated such that their ϕ values are 1 unit from their nearest neighbour in the zero level set. If no neighbour exists then the point is added to one of S_2 or S_{-2}, respectively.

4. Points with ϕ values that now fall outside of the ranges specified in Equation 4.29 for L_1 and L_{-1} are moved to either S_0, S_1 or S_{-1} depending on which range their new value falls into.

5. The above steps are repeated for the 2nd level sets. The values of ϕ are updated for points in L_2 and L_{-2} so that they are 1 unit from their nearest neighbour in the zero level set. If no neighbour exists then the point is removed from the list and ϕ, and the value in the label map is changed to 3 or -3, respectively.

6. Points with ϕ values that now fall outside of the ranges specified in Equation 4.29 for L_2 and L_{-2} are moved to either S_1 or S_{-1} if their value is too low. If their value is too high then the point is removed from the list and ϕ, and the value in the label map is changed to 3 or -3 accordingly.

The temporary lists are then used to change the status of the points within them as follows:

1. All the points in S_0 are added to L_0 and their corresponding values on the label map updated to 0.

2. All the points in S_1 and S_{-1} are respectively added to L_1 and L_{-1}, and their labels updated. The neighbours of these points are checked, and if any have neighbours with ϕ values of 3 or -3 then the neighbours are added to S_2 or S_{-2} as appropriate. Their ϕ values are changed to be one unit apart from that of the neighbouring point.

3. Finally all the points in S_2 and S_{-2} are respectively added to L_2 and L_{-2}, and their labels updated.

This completes one iteration of the algorithm. The process is repeated with F_2 values calculated on the new contour position. The method is iterated until a convergence is reached. The SFM implementation also has the benefit of being able to update the image statistics, i.e., the values c_1 and c_2 used in calculating F, efficiently. This is achieved by tracking when points cross the zero level set, either becoming part of the background or the object, and assigning them to two lists, 'in to out', or 'out to in', respectively. The two lists are processed in order to update the interior and exterior average values. This is significantly more efficient than calculating the new means from scratch for each iteration.

4.5 Proposed Method for Image Segmentation

The proposed object segmentation method comprises three stages. In the first stage the image is prepared and the focus value map generated. The second stage performs an initialisation for the active contours. In the third stage, the segmentation of the OoI is performed using Whitaker's SFM described in Section 4.4. The second and third stage of method are presented in this section. For the first stage, the focus assessment, the user is referred to the method described in Chapter 3.

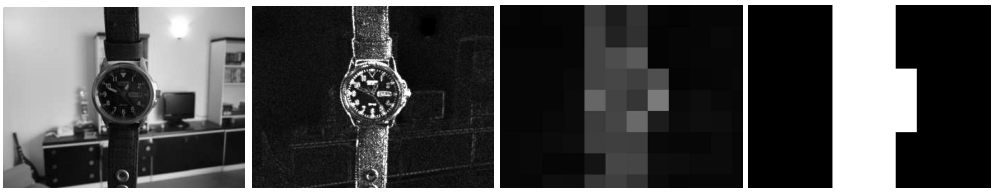
4.5.1 Contour Initialisation

A good initial boundary (i.e., an initialisation mask) for the active contours algorithm not only helps speed up the segmentation process, it is also of vital importance for the accuracy of the segmentation. As the SFM only looks for new contours in the vicinity of the previous iteration, the final segmentation is dependant on the initial one. Ideally an initialisation should encompass the entire OoI whilst excluding as much background as possible. A grid based approach is adopted to generate the binary initialisation mask, used in the SFM. The focus map is first split into boxes, the height and width of which are determined by the size of the image to determine the best initialisation mask, i.e.,

$$\text{box size (pixels)} = \begin{cases} 300 & \text{if} \quad & (\text{Height}_{Image} + \text{Width}_{Image}) & \geq 3000 \\ 200 & \text{if} \quad 3000 > & (\text{Height}_{Image} + \text{Width}_{Image}) & \geq 2000 \\ 100 & \text{if} \quad 2000 > & (\text{Height}_{Image} + \text{Width}_{Image}) & \geq 1000 \\ 30 & \text{if} \quad & (\text{Height}_{Image} + \text{Width}_{Image}) & < 1000. \end{cases} \tag{4.34}$$

The maximum focus value in each box of the grid is assigned to all pixels within that particular grid square. Otsu's thresholding method [Otsu, 1979] is then applied

to this grid, and blocks with a value above 0.5 of Otsu's threshold are assigned a value of 1 (denoted by white) and those otherwise assigned 0 (denoted by black). This is to ensure that that OoI lies within initial contour. The initialisation process is illustrated in Figure 4.4.



(a) (b) (c) (d)

Figure 4.4: Focus assessment and contour initialisation of an image with a watch as the OoI: (a) image; (b) focus energy map; (c) maximum values are assigned to each square in the grid; and (d) the corresponding initialisation mask after thresholding.

4.5.2 Object Segmentation

The automatically generated binary segmentation mask is used to create the initial condition for an active contours algorithm. An implicit level set active contours method is adopted, using the energy function defined by Chan-Vese [Chan and Vese, 2001]. The energy of the contour, C, is repeated here for clarity, i.e.,

$$\begin{aligned} E(C) &= \lambda_1 \int_{inside} |I(x,y) - c_1|^2 \, dxdy + \lambda_2 \int_{outside} |I(x,y) - c_2|^2 \, dxdy \\ &\quad + \mu \cdot \text{length}(C) + \nu \cdot \text{area}(inside(C)), \end{aligned} \tag{4.35}$$

where the contour C is represented by the zero-level set of a continuous Lipschitz function, I is the image, c_1 and c_2 are respectively the average pixel values inside and outside of contour C, length(.) and area(.) respectively impose length and area constraints on the contour, and λ_1 , λ_2, ν and μ are fixed parameters with $\lambda_1, \lambda_2 > 0$ and $v, \mu \geq 0$. In the proposed implementation $\lambda_1 = \lambda_2 = 1$, $\nu = 0$, as in [Chan and Vese, 2001], and $\mu = 0.4$ as the goal is to segment large objects and reduce the likelihood of the contour leaking into object boundaries. Using the Active Contours with Edges model means that areas in focus can be segmented, even if there is not a well defined boundary in the focus energy map. To speed up the active contours algorithm, Whitaker's sparse field method [Whitaker, 1998] (see Section 4.4) is used so that calculations are performed only around the zero level set, thus improving the algorithm efficiency.

A disadvantage of the method is that new contours cannot spontaneously

appear. This means that holes within an object, e.g., as with a donut, are considered parts of the object. However, this is an advantage when segmenting weakly textured objects since the object pixels only return small focus values, but as they are contained within more dominant object edges they are still segmented correctly.

The active contours algorithm is applied using the initialisation mask to a downsampled focus energy map for 200 iterations. The downsampling increases the segmentation speed for larger images and is performed with a factor of 2^n, where n is determined by the image size to obtain the best segmentation, i.e.,

$$\mathrm{n} = \begin{cases} 3 & \text{if} \quad\quad (\mathrm{Height}_{Image} + \mathrm{Width}_{Image}) \geq 3000 \\ 2 & \text{if} \quad 3000 > \quad (\mathrm{Height}_{Image} + \mathrm{Width}_{Image}) \geq 2000 \\ 1 & \text{if} \quad 2000 > \quad (\mathrm{Height}_{Image} + \mathrm{Width}_{Image}) \geq 1000 \\ 0 & \text{if} \quad\quad (\mathrm{Height}_{Image} + \mathrm{Width}_{Image}) < 1000. \end{cases} \tag{4.36}$$

A binary segmentation is obtained with interior pixels being assigned the value 1 and exterior 0. The binary segmentation is then used as the initialisation mask for a further 200 iterations. The reinitialisation prevents the level set function from becoming too flat as in [Chan and Vese, 2001]. This 2-stage process is repeated until the method converges on a solution to give an initial scaled down binary segmentation $S_i(x, y)$. This is then upscaled by interpolation by a factor of 2^n to the size of the original image with non-zero values being assigned the value 1, thus giving the final binary segmentation $S(x, y)$.

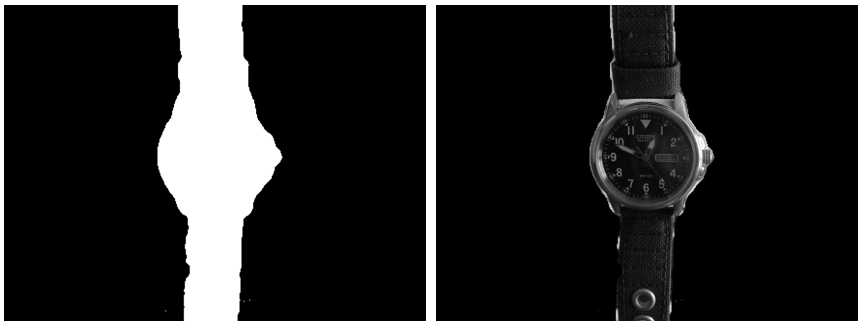
The final segmentation can then be used to obtain a view of the segmented object using a simple method. An object segmented image $I(x, y)$ is generated with pixels of value 0 being the background, i.e.,

$$I(x, y) = \begin{cases} G(x, y) & \text{if} \quad S(x, y) = 1 \\ 0 & \text{if} \quad S(x, y) = 0 \end{cases} \tag{4.37}$$

where G is the original greyscale image. This is illustrated in Figure 4.5.

4.6 Experimental Results and Discussion

The performance of the proposed object segmentation method is evaluated on a variety of test images as shown in Figures 4.6 and 4.7. Some of these images are generated for the purposes of this thesis or taken from personal collections, whilst others are taken from the Berkeley Segmentation Dataset [Martin et al., 2001] and screen captures of video footage. This gives a good variety of image resolutions,



(a) (b)

Figure 4.5: Object segmentation: (a) binary segmentation $S(x, y)$; and (b) object segmentation $I(x, y)$.

scene compositions and objects to be segmented. The test images are as follows: a flower with other foliage in the background; a soft toy on a desk; a watch suspended in a living room; a plant with other foliage and a house in the background; a boy model on a cluttered desk; a model house on a turntable in an office; a goose with a background of other geese and reeds; a wizard against a forest background; a lizard on a rock; a bird sitting on a branch against a clear sky; an ostrich against a distant background; a soldier with snow falling in a wood; a ranger against a wall of ice with various background features; and a mercenary against a textured wall.

The segmentation of the test images using the proposed method are shown in column (b) of Figures 4.6 and 4.7. The results show that the proposed method generates good segmentations with a variety of different objects and scenes where there is a focus differential between the in-focus object (i.e., the OoI) and the background. The segmentation of the soft toy illustrates a shortcoming of the method, namely it cannot exclude the background hole (the region under the ear).

The performance of the proposed method is compared with that of the interactive segmentation methods, Grabcut, BPT, and interactive graph cuts (IGC), respectively shown in columns (c), (d) and (e) of Figures 4.6 and 4.7. An overview of these methods is provided in Section 4.2.2. The Grabcut method is implemented with just the initial bounding box with no further user refinement, to enable it to be compared with a method that uses very little human input. BPT and IGC are chosen as they are ranked the best two methods by McGuiness and O'Conor [McGuiness and O'Conor, 2010]. They both involve a significant amount of user interaction and refinement. The implementations of these interactive segmentation methods are as in [McGuiness and O'Conor, 2010].

To enable a quantitative comparison of the performances of the four methods,

(a) (b) (c) (d) (e)

Figure 4.6: Segmentation of an OoI: (a) original images; and results using (b) proposed method; (c) GrabCut; (d) BPT and (e) IGC.

(a) (b) (c) (d) (e)

Figure 4.7: Segmentation of an OoI: (a) original images; and results using (b) proposed method; (c) GrabCut; (d) BPT and (e) IGC.

the test images were manually segmented to obtain their segmented ground truth of the OoI. This enables the segmentation error rate, as proposed in [Wollborn and Mech, 1998] to be calculated as

$$p(M_{seg}, M_{gt}) = \frac{\sum_{(height,width)} M_{seg}(x,y) \otimes M_{gt}(x,y)}{\sum_{(height,width)} M_{gt}(x,y)}, \tag{4.38}$$

where M_{seg} is the binary segmentation generated by the method being evaluated, M_{gt} is the manually segmented ground truth of the OoI and $\otimes$ is the exclusive OR logical operator. The error rates for the segmentations in Table 4.1 show that the proposed method performs very favourably when compared to the Grabcut algorithm and competes well with the interactive BPT and IGC algorithms, despite the proposed method being completely autonomous. It also has the additional advantage of being able to operate on greyscale images.

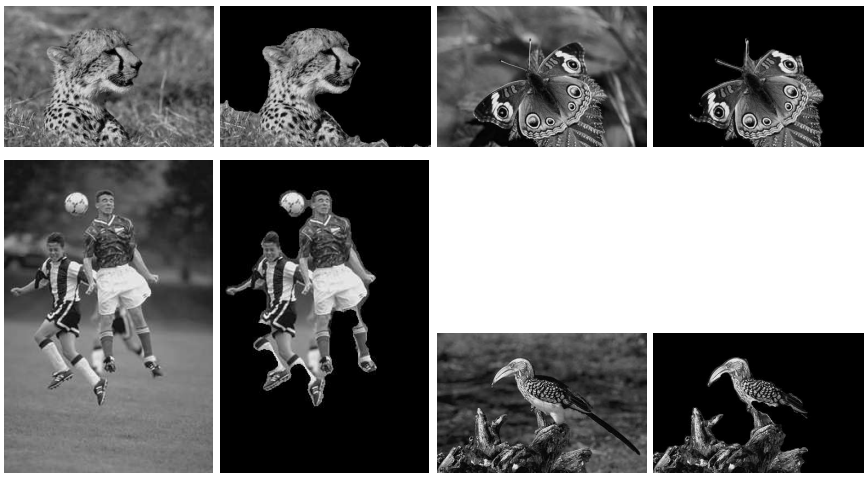
Image	Proposed Method	GrabCut	BPT	IGC
Flower 1	0.0585	0.162	0.0356	0.0272
Soft Toy	0.0323	0.085	0.0173	0.0188
Watch	0.0439	0.198	0.015	0.0128
Plant	0.1509	1.161	0.109	0.0904
Model	0.1842	0.632	0.0335	0.0191
House	0.0246	0.084	0.0214	0.0122
Goose	0.0184	0.448	0.0115	0.0245
Wizard	0.0219	0.096	0.0467	0.052
Lizard	0.0607	0.501	0.045	0.058
Bird 1	0.1502	1.466	0.193	0.194
Bird2	0.0765	0.263	0.094	0.131
Soldier	0.0373	0.1113	0.0118	0.0151
Ranger	0.0286	0.1420	0.0360	0.0465
Mercenary	0.0213	0.0468	0.0448	0.0394

Table 4.1: Segmentation error rates of the proposed method and three comparison segmentation methods.

Using test images from [Liu et al., 2010] the proposed method is also compared with the focus based segmentation methods of Kim [Kim, 2005] and Liu [Liu et al., 2010]. The results of the segmentation using the proposed method are shown in Figure 4.8. It should be noted that the ground truths were generated for the purpose of this thesis and thus will not be identical to the ground truth used by Kim and Liu. Therefore a direct comparison with these two results cannot be made.

The error rates in Table 4.2 show that the proposed method performs better

than Kim's Method for the images of the bird, butterfly and cheetah, and matches the performance of Liu's for the images of the cheetah, butterfly and bird, thus showing its performance to be comparable to these methods.



(a) (b) (c) (d)

Figure 4.8: Segmentation of test images from Liu et al. [2010] using the proposed method.

Image	Method	Kim2005	Liu2010
Cheetah	0.056	0.079	0.085
Butterfly	0.087	0.187	0.102
Footballers	0.230	0.133	0.111
Bird	0.194	0.219	0.177

Table 4.2: Segmentation error rates of test images from [Liu et al., 2010] using the proposed method and two low DoF comparison methods . The error rates for Kim's and Liu's Methods were obtained from [Kim, 2005] and [Liu et al., 2010], respectively.

4.7 Conclusion

This chapter addresses the problem of autonomous object segmentation by restricting input images to be those with a low DoF. The method takes focus maps generated as in Chapter 3 and uses a narrow band level sets implementation of Active Contours without Edges to segment the focused regions. The robust initial contour is provided through a grid based method. The proposed object segmentation method is shown to work on a variety of different test objects and background scenes, and produces favourable results when compared to other low DoF segmentation meth-

ods. It is limited by its inability segment internal holes if the initial contour does not include them, however this does enable the method to segment weakly textured objects, or those with largely homogeneous regions.

Chapter 5

Object Segmentation from Low DoF Video Footage

5.1 Introduction

An application of an image segmentation method is to see whether it can be adapted to perform object segmentation from a sequence of images, i.e., video footage. As with image segmentation, video segmentation is an increasingly popular area of research with a huge range of methods and applications being presented in the literature [Ngan and Li, 2011]. The increasing prevalence of mobile phones and other hand-held devices able to capture video footage makes this a very important and relevant area of research.

The applications of video segmentations are extensive and often form the first low level stage of more complex computer vision systems. For example, video segmentation algorithms can form part of a video monitoring system [Gentile et al., 2004] whereby a segmentation allows a system to detect the presence of an intruder or unusual occurrence and thus trigger the appropriate alert. As with image segmentation, video segmentations can also form part of object recognition [Todorovic and Ahuja, 2008] and content based retrieval systems [Ko and Byun, 2005].

Identifying the OoI in a scene via video segmentation also results in improved data compression and is important for various video coding formats [Meier and Ngan, 1999]. Segmenting each frame means that the areas of interest can be kept at a high quality whilst the background regions are compressed.

As videos and images are intrinsically linked to each other, video segmentation methods are often based on image segmentation methods. For example, the Interactive Video Cutout method [Wang et al., 2005] is based on the principles of

the Graph Cuts method [Boykov and Jolly, 2001]. Rather than painting seeds on each individual frame, which would be a laborious process, the method presents a novel interface which allows the user to paint on surfaces within the spatio-temporal video volume to indicate the background and foreground regions. Other methods, such as Active Graph Cuts [Juan and Boykov, 2006], can even operate in real time with live video feeds rather than recorded footage.

This chapter presents a low DoF video segmentation method based on the method presented in Chapter 4 and is organised as follows. Section 5.2 covers related work: two existing low DoF object segmentation methods that have been extended for video segmentation. Section 5.3 proposes changes to the initialisation process, both for the the first frame and then for subsequent frames, to the image segmentation method presented in Chapter 4 in order to efficiently segment video sequences. Finally, Section 5.4 presents the results of the proposed method, showing the segmentation of video sequences and making a comparison to related work. The chapter is concluded in Section 5.5

5.2 Related Work

Whilst other low DoF object segmentation methods could be applied to each image frame in a video sequence independently, there are currently only two main methods which make use of the relationship between frames in a video sequence. This section presents the two methods in detail.

5.2.1 Kim's Method

Kim et al. [Kim et al., 2007] propose a method for extracting objects from low DoF images which is extended and modified to extract an object from a sequence of images. The method is split into 3 parts. In the first a higher-order statistics (HOS) map is generated from the input image to find the high frequency (i.e., in focus) regions of the image. The fourth-order moments are calculated for all pixels within the red, green and blue channels of the input image. When generating the HOS map, only the maximum moment value across all 3 channels is used. The values in the map are then scaled to give each pixel a value between 0 and 255.

In the second stage a block-based OoI is extracted. Firstly the HOS map is partitioned into blocks. For each block the maximal value is found and assigned to all pixels within the block. The coordinate corresponding to the maximum HOS block value is used as the seed point. Starting at the seed point, all neighbouring blocks with values of 255 are extracted. This process is repeated for each connected

block with a value of 255 until no more are found. Finally a hole filling algorithm is used to obtain the final block based OoI.

In the third section the final segmentation is performed. The block based OoI serves as a mask which contains the final locations of the OoI inside it. A filling technique [Kim et al., 2001] obtains a vertically filled OoI and a horizontally filled OoI. An AND operation generates the final segmentation of the object. Cascaded opening and closing morphological operations can be used to smooth the object boundaries.

To extend the algorithm to video sequences the block based OoI from the previous frame is dilated and the HOS for the new frame is only calculated within this region, as due to correlation between video frames, the OoI is unlikely to have moved significantly. This reduces computational complexity and reduces processing time significantly.

5.2.2 Li's Method

Li and Ngan [Li and Ngan, 2007] adopt a different approach that consists of three stages. In the first a saliency map of a video frame is generated using a reblurring model to identify focused regions. The salient regions are then smoothed and accentuated using morphological filtering. This allows a trimap of object, background and ambiguous regions to be created. In the third stage the object boundaries are extracted using an adaptive error-control matting scheme. The method achieves good segmentation for a variety of different scenes where large focus differentials are present. For the remaining frames of the video sequence a motion estimation algorithm is used to identify the region containing the focused object in the current frame from information in the previous frame. Morphological erosion and dilation are performed on the projected region separately and used to create the trimap (the difference in the two regions being labelled ambiguous). The adaptive error-control matting method is then used to segment the image into object, background and mixed pixels.

5.3 Proposed Video Segmentation Method

Whilst an image segmentation method can be applied consecutively to each image frame of a video, and thus obtain a segmentation for the whole sequence, this ignores the correlation between frames and thus misses the opportunity of using additional information to either obtain a more accurate, or computationally efficient, segmentation. Two assumptions are made when addressing the problem of low DoF object

segmentation from video footage.

1. There is no change of scene in the video sequence, i.e., the OoI remains the subject of the entire video sequence.

2. The video frame rate is sufficiently high such that there are no large discrepancies in image composition from one frame to another.

The method proposed is nominally the same as that described in Chapter 4, with the exception of the contour initialisation process. For the initial contour of the first frame of the video sequence, a more robust initialisation is proposed to ensure that the the OoI is within the initial contour. For subsequent frames, the initialisation is based on the results of the segmentation of the previous frame. A block diagram is presented in Figure 5.1 to aid in the visualisation of the stages of the method.

5.3.1 First Frame Initialisation

If a method is utilising the previous frame's segmentation result as a factor in acquiring the current one, then this means that the first result, i.e., for frame $n = 1$, must be as accurate as possible, as all future segmentations will stem from it. The linked nature of frames (i.e., scene composition changes very little over a few frames) is exploited to obtain a more robust initial initialisation for the first frame of the video sequence.

The goal is to ensure that the OoI is entirely encompassed by the initial contour. Thus, focus intensity maps are generated from frames $n = 1, 3, 5$. The maximum focus value from across all three maps for each pixel is then used to generate a different focus intensity map, i.e.,

$$Map_{max}(x, y) = \max(Map_{n=1}(x, y), Map_{n=3}(x, y), Map_{n=5}(x, y)). \tag{5.1}$$

As the OoI interest will have moved little during only five frames, there will be a strong overlap in object pixels between the three frames. This focus map, Map_{max}, is then used in the grid process described in Section 4.5.1 to generate the initial contour for use with the active contours algorithm on $Map_{n=1}$ Taking the maximum values from across the three frames ensures that the initial contour will envelop the OoI, thus making the method more robust.

Figure 5.2 shows frames $n = 1, 3, 5$ and their corresponding focus assessments. The maximum focus values from across these three frames is then shown in

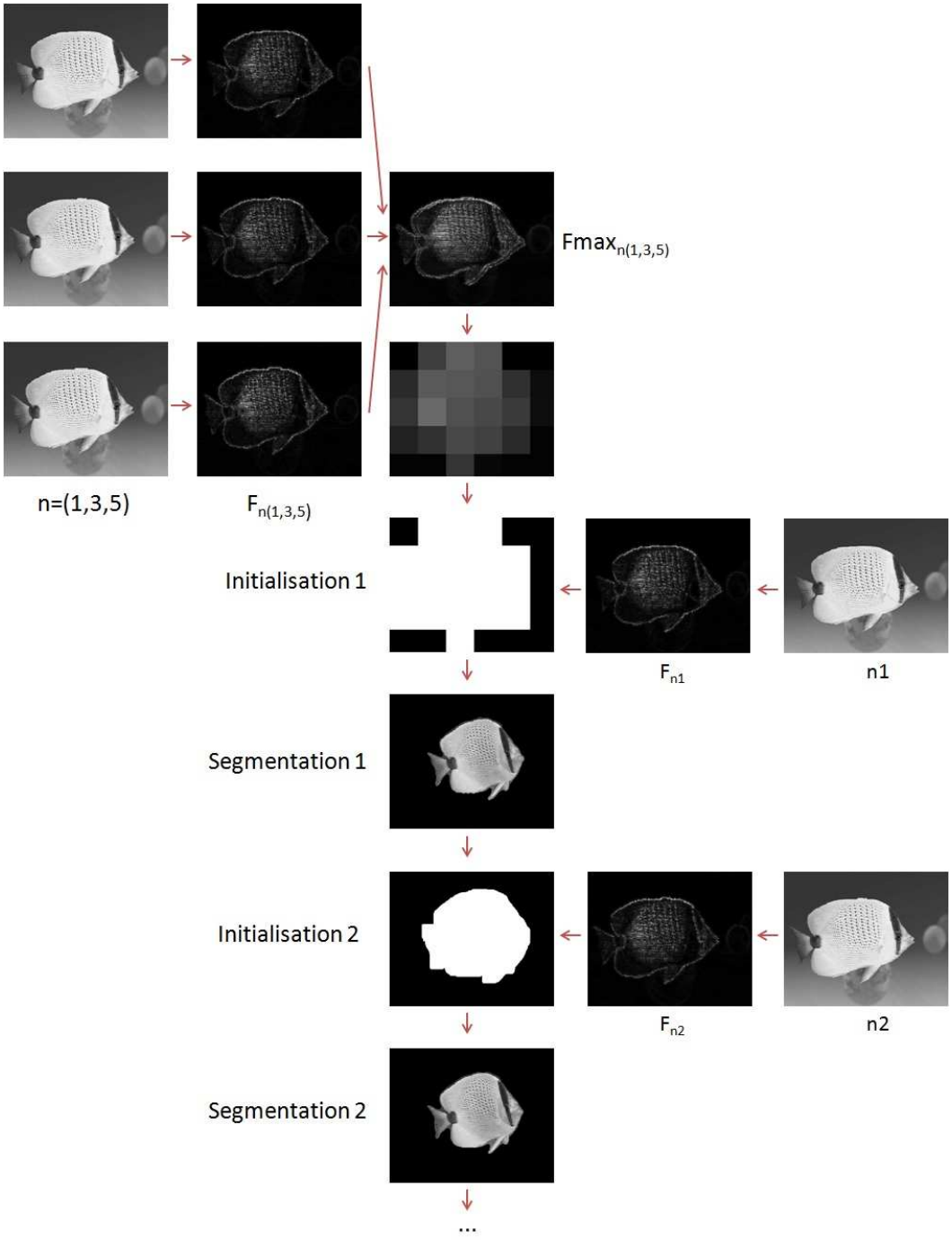


Figure 5.1: First, third and fifth frame of a video sequence of a swimming fish, and corresponding focus assessments used to produce a first initial contour. Subsequent initial contours are produced from the binary dilation of the previous frame's segmentation.

Figure 5.3 as well as the corresponding grid which is generated and thresholded in the usual manner as described in Section 4.5.1 to produce the initial contour, ready for the sparse fields active contour algorithm to produce the segmentation.

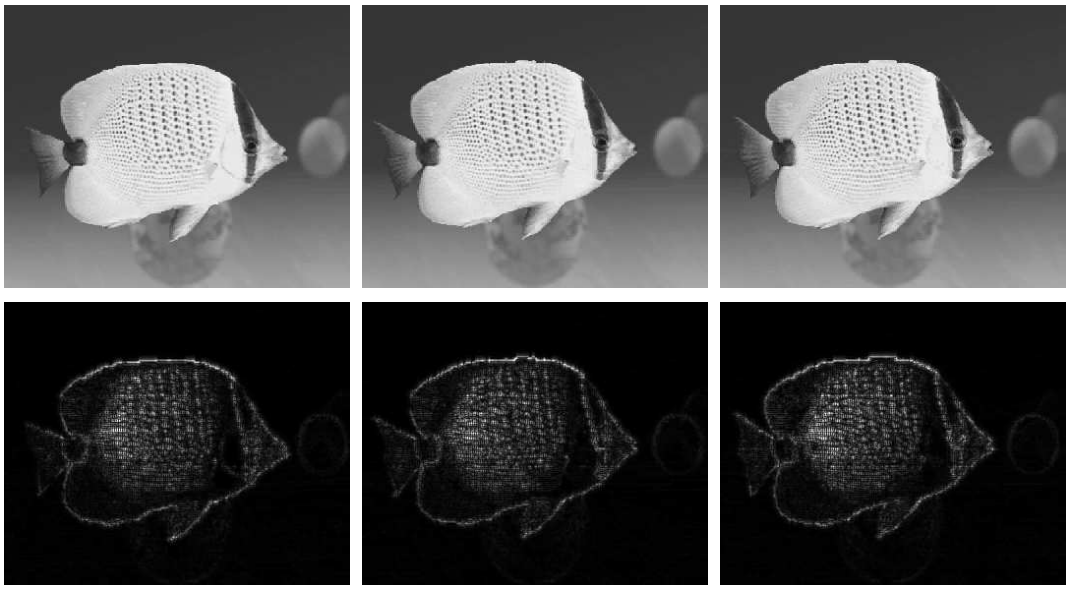
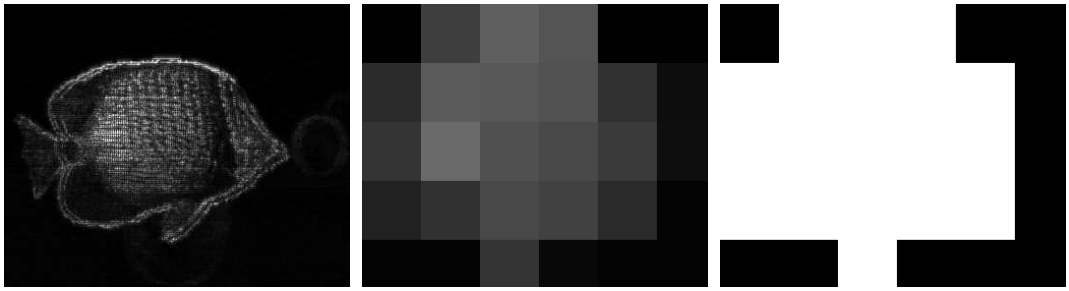


Figure 5.2: First, third and fifth frame of a video sequence of a swimming fish, and corresponding focus assessments.



(a) (b) (c)

Figure 5.3: Maximum focus values across frames $n = 1, 3, 5$ (a), block based focus assessment (b), and initialisation for active contours (c).

5.3.2 Further Frame Initialisations

To improve the computation efficiency of the segmentation method, further initialisation masks for the nth frame are generated from the binary dilation of the segmentation of the $n-1$th frame by a 50×50 square structuring element of ones. This can be scaled accordingly for higher resolution video squences. This allows for

all frames subsequent to the initial one to be segmented efficiently, with a suitable initial contour. As with the proposed image segmentation method, the proposed video segmentation method is fully automatic and requires no user input.

Figure 5.4 shows the result of a segmentation of a frame of the video sequence (a). The dilation of this segmentation to produce an initial contour is shown by (b). This is used to produce the segmentation (c).

(a) (b) (c)

Figure 5.4: Segmentation result of a frame $n-1$ of the image (a), dilation of segmentation result (b), segmentation of frame n using the dilation as an initialisation for the active contours (c).

5.4 Video Segmentation Results

The performance of the proposed video segmentation method is evaluated on several low DoF video sequences used in Kim et al. [2007]. The dimensions of a video frame in all video sequences is 352x288 pixels. The first sequence is that of a swimming fish. The second and third sequences are of blooming flowers. Experimental results from a selection of consecutive video frames are shown in Figures 5.5, 5.6 and 5.7. The segmentation error as proposed in Wollborn and Mech [1998] is again used as a measure of sucess of the video segmentations:

$$p(M_{seg}, M_{gt}) = \frac{\sum_{x,y} M_{seg}(x,y) \otimes M_{gt}(x,y)}{\sum_{(x,y)} M_{gt}(x,y)}, \tag{5.2}$$

where M_{seg} is the binary segmentation generated by the method being evaluated, M_{gt} is the manually segmented ground truth of the OoI and $\otimes$ is the exclusive OR logical operator.

It can be seen from Figure 5.5 that good segmentations of the fish are achieved, despite low video quality and non-rigid motion of the OoI. The mean segmentation error was calculated to be 0.0573, showing a good degree of accu-

racy. The segmentation accuracy of Kim's method for the swimming fish sequence is given in Kim et al. [2007] as 94.3%. Although a direct comparison cannot be made due to the use of a different frame range and potentially different manual segmentations, the mean accuracy of the segmentations obtained by the proposed video segmentation method is 98.9%.

The mean segmentation error for the first sequence of a blooming flower as shown in Figure 5.6 is 0.128. Relatively good segmentations are achieved but due to the low resolution of the videos, the algorithm had some difficulty in segmenting the areas between the flower's petals. The second video sequence of a flower shown in Figure 5.7 gives a mean segmentation error of 0.0593, again showing that an accurate segmentation has been achieved. Segmentation error rates for the percentage correctly segmented were not given by Kim for the flower sequences.

5.5 Conclusion

This chapter expands upon the low DoF object segmentation method presented in Chapter 4, enabling it to segment video sequences accurately and efficiently. For the first frame, a robust initialisation process is proposed, making use of the linked nature of video frames. Active contour initialisations for further frames are based on the dilation of the previous frame's binary segmentation. The method is tested on a number of video sequences, and is shown to compare favourably to another low DoF video segmentation method.

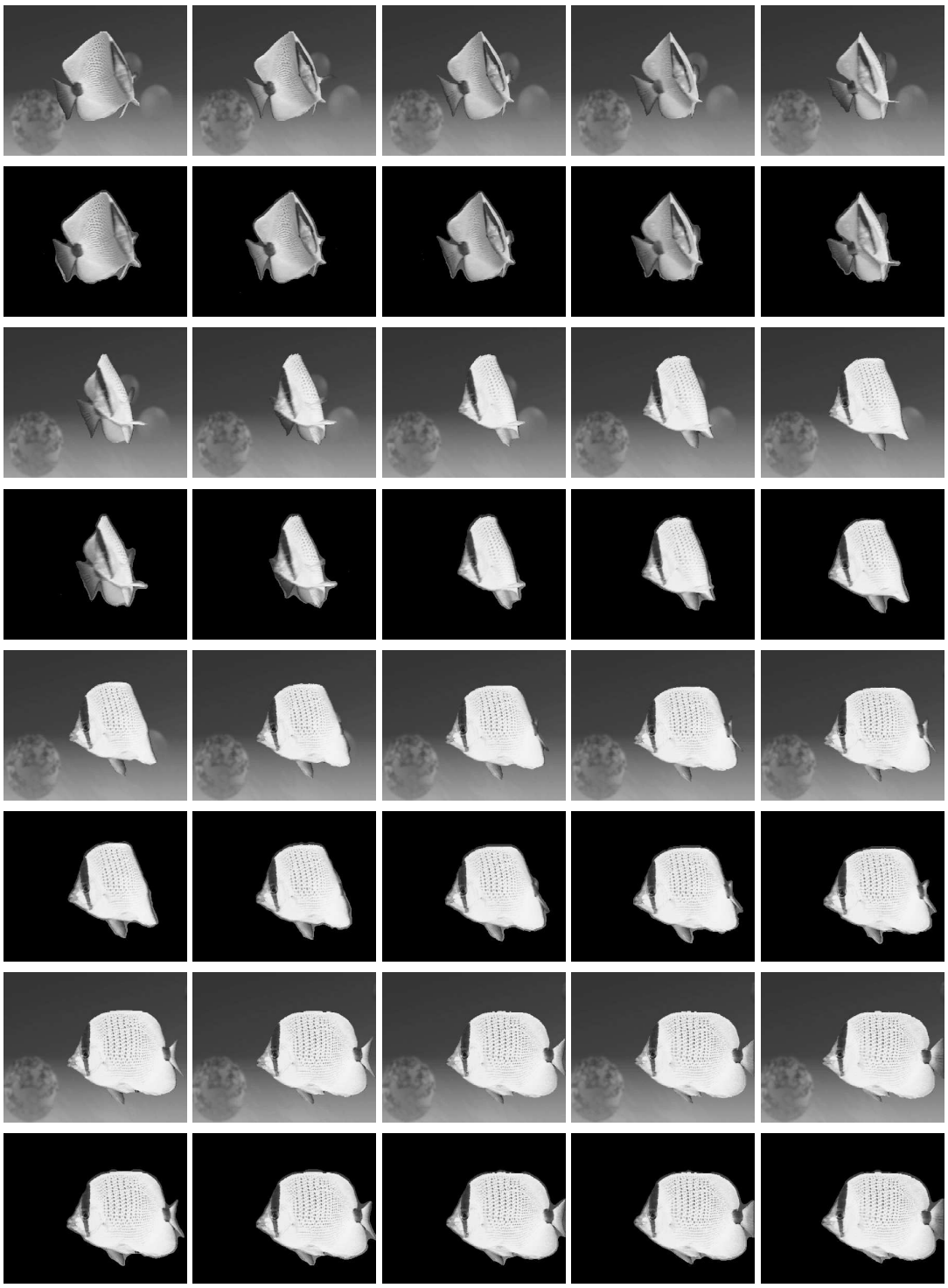


Figure 5.5: Segmentation of swimming fish video sequence: original image frames on odd rows and segmented OoI on even rows. Mean segmentation error = 0.0573.

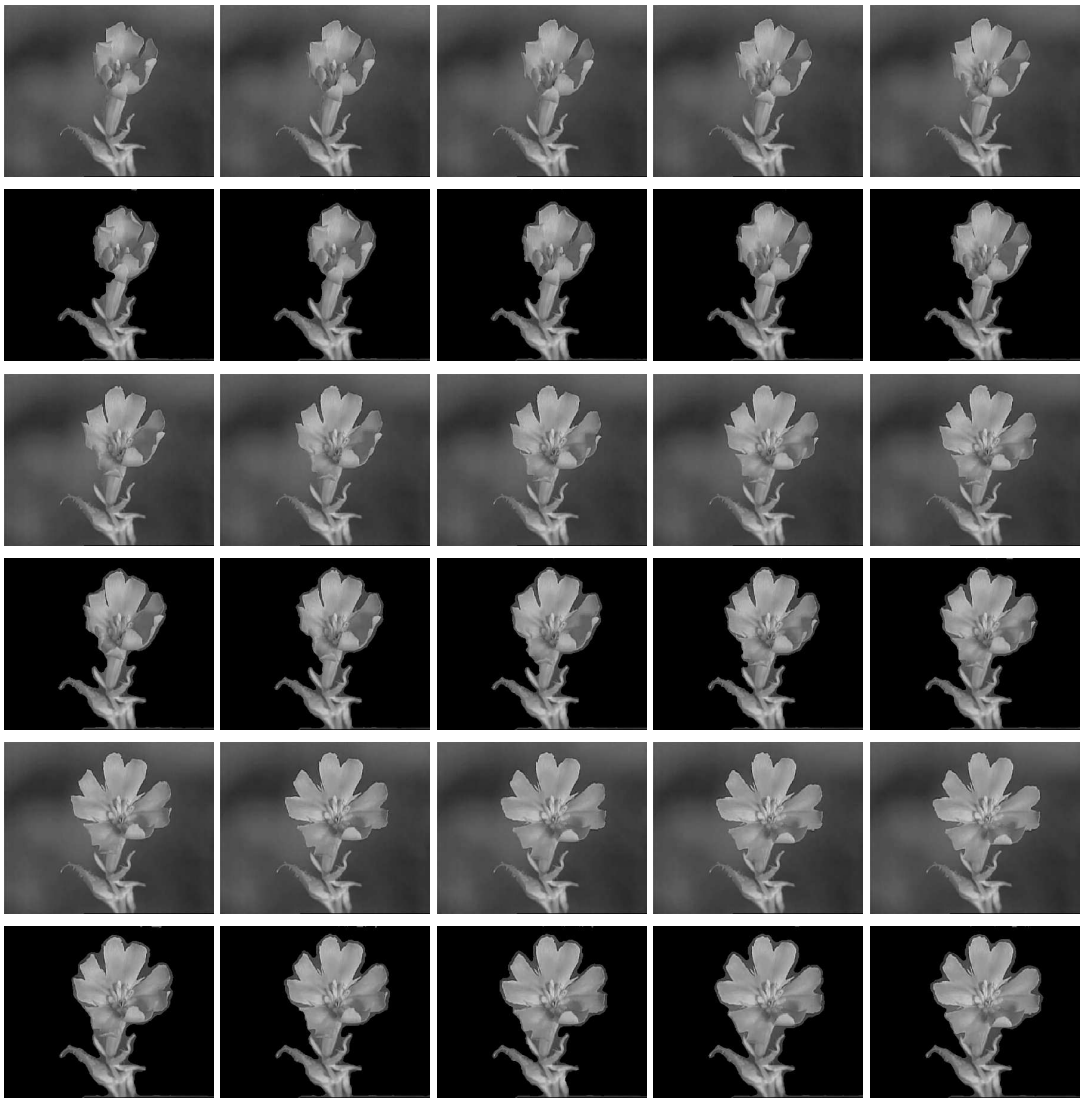


Figure 5.6: Segmentation of blooming flower 1 video sequence: original image frames on odd rows and segmented OoI on even rows. Mean segmentation error = 0.128.

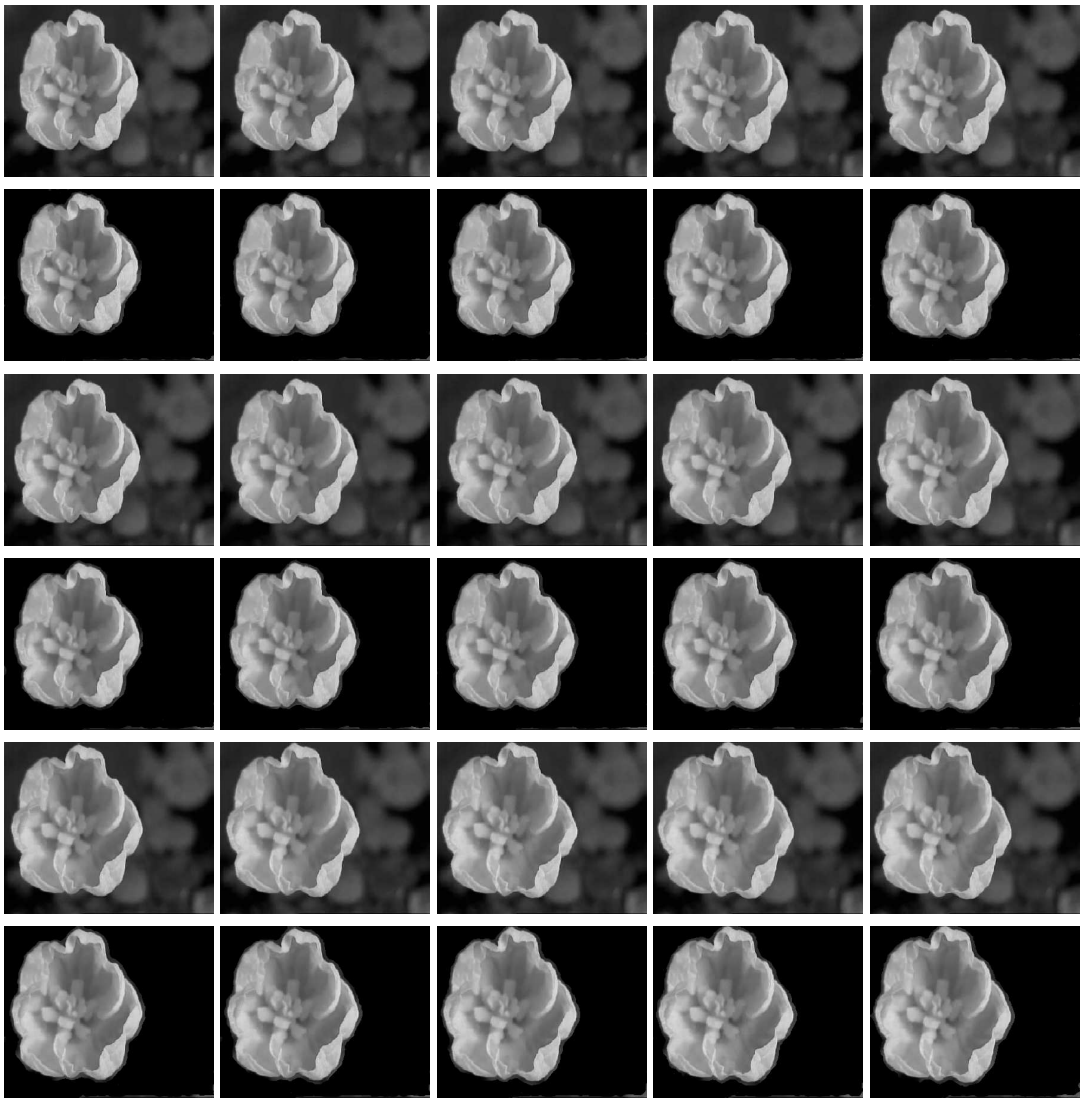


Figure 5.7: Segmentation of blooming flower 2 video sequence: original image frames on odd rows and segmented OoI on even rows. Mean segmentation error = 0.0593.

Chapter 6

Automatic Trimap Generation for Matting Algorithms

6.1 Introduction

Whilst a binary segmentation is a desirable result for many tasks, for applications in image and video editing such as image compositing (i.e., combining one image with another, usually a foreground onto a new background), a 'softer' approach is required. Image matting addresses the problem of foreground estimation in images - in the case of this thesis, the focused OoI. Matting determines which pixels are foreground (pixels with value 1), which are background (pixels with value 0), and also determines an alpha value between 0 and 1 for those that are a mixture of both. The alpha value is a measure of how much a mixed pixel belongs to the background or the foreground. It is used as a measure of the transparency of the pixel when compositing two images together. Matting is sometimes referred to as a soft segmentation - as opposed to a hard binary segmentation where pixels must only belong to one of two classes. It is particularly useful when applied to boundaries that have very fine details or textures such as hair or fur.

Most modern matting algorithms make use of a user defined trimap. This splits the image into areas of one of the following type: foreground, background and ambiguous/mixed. The ambiguous regions are then processed by a variety of methods to form a matte. These methods can generally be divided into two categories: sampling based and propagation based. In this chapter the image and video segmentation methods presented in Chapters 4 and 5 are applied to the problem of autonomous image matting. By adapting the segmentation method proposed in this thesis, trimaps can be automatically generated, thus removing the need for human

input in the matting process. The generated trimaps are used in conjunction with the Robust Matting method [Wang and Cohen, 2007] to produce alpha mattes and perform image and video compositions.

The rest of the chapter is organised as follows. Section 6.2 provides an overview of the basic fundamentals of image matting, whilst Section 6.3 covers some of the more prominent methods for extracting a matte. In Section 6.4 the Robust Matting method is described in detail. The automatic tripmap generation method is proposed in Section 6.5. This is used in conjunction with the Robust Matting method to generate the results in Sections 6.6 and 6.7. The chapter is concluded in Section 6.8.

6.2 Matting Fundamentals

Digital matting is the process of separating a foreground element or object from a background, determining which pixels both fully belong to the object group and which partially belong to the group. It is closely associated with image compositing - the process of rendering a foreground over a given background which was initially developed for use in film and video production [Fielding, 1972]. The problem was first described mathematically by Porter and Duff [Porter and Duff, 1984], where the alpha channel to control the linear interpolation of foreground and background colours was introduced. The alpha channel, or alpha matte, is an extra attribute assigned to each pixel, with a value between 0 and 1. An image I is modelled as the combination of the image background B, and the image foreground F using the alpha matte α. A pixel in I is given by

$$I_{(x,y)} = \alpha_{(x,y)} F_{(x,y)} + (1 - \alpha_{(x,y)}) B_{(x,y)}. \tag{6.1}$$

If $\alpha_{x,y} = 1$ then the pixel $I_{x,y}$ is definite object/foreground. Conversely if $\alpha_{x,y} = 0$ then the pixel is definite background, otherwise the pixel is a combination of the two. For example, Figure 6.1 shows an image of a ranger against an icy background (a), and the corresponding alpha matte (b). The black pixels (with value 0) represent the background of the image, the white pixels (with value 1) the foreground or object, and the pixels with grey values between 0 and 1 the mixed regions.

For image compositing, once the foreground F has been determined, the background B can simply be replaced with a new background, for example B', to form the new composite image J, i.e.,

$$J_{(x,y)} = \alpha_{(x,y)} F_{(x,y)} + (1 - \alpha_{(x,y)}) B'_{(x,y)}. \tag{6.2}$$

(a) (b)

Figure 6.1: Image of a ranger against an icy background (a) and the corresponding alpha matte (b). The matte is produced using a user generated trimap and with Wang's Robust Matting Method [Wang and Cohen, 2007].

An example of this image compositing, sometimes called scene superimposition, is shown in Figure 6.2. The extracted matte shown in Figure 6.1 is used to composite the image of the ranger onto a plain background (a), and a forest background (b). It can be seen that the character looks like a natural part of the image, despite the background having been changed.

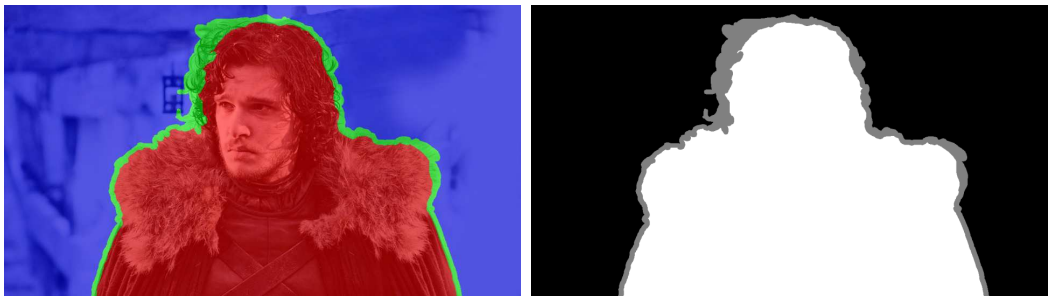
(a) (b)

Figure 6.2: Image of a ranger superimposed on a plain background (a) and a forest background (b). The original image and matte are taken from Figure 6.1.

Considering Equation 6.1, it can be see that for each pixel in a colour image there are three known variables, $I(r, g, b)$ but seven unknown variables, α, $F(r, g, b)$ and $B(r, g, b)$. This makes the the equation under constrained, i.e., more information is required in order to solve the problem or to estimate a solution. For this reason photos or videos are often traditionally taken in front of a blue or green scene if a composite image is desired. By using a background of a known colour and making assumptions about the foreground colours then there is enough information to estimate the matte and create a composite image [Smith and Blinn, 1996]

Extracting a matte from natural scenes with uncontrolled backgrounds is a

more difficult task. To do so, some form of prior information is required. This is usually some user input which takes the form of a trimap. In a trimap, the user segments the image into three areas: regions of the image that are definitely background, regions of the image that are definitely foreground, and unknown regions. This reduces the matting problem to only estimating F, B and α in this third ambiguous region of the image. These variables can be estimated based on known foreground and background pixels to produce the alpha matte. An example of a trimap painted using a user interface is shown by Figure 6.3, (a). The red regions correspond to the foreground or object, the blue the background, and the green the mixed or ambiguous regions. This produces the trimap used for the matting algorithm (b).

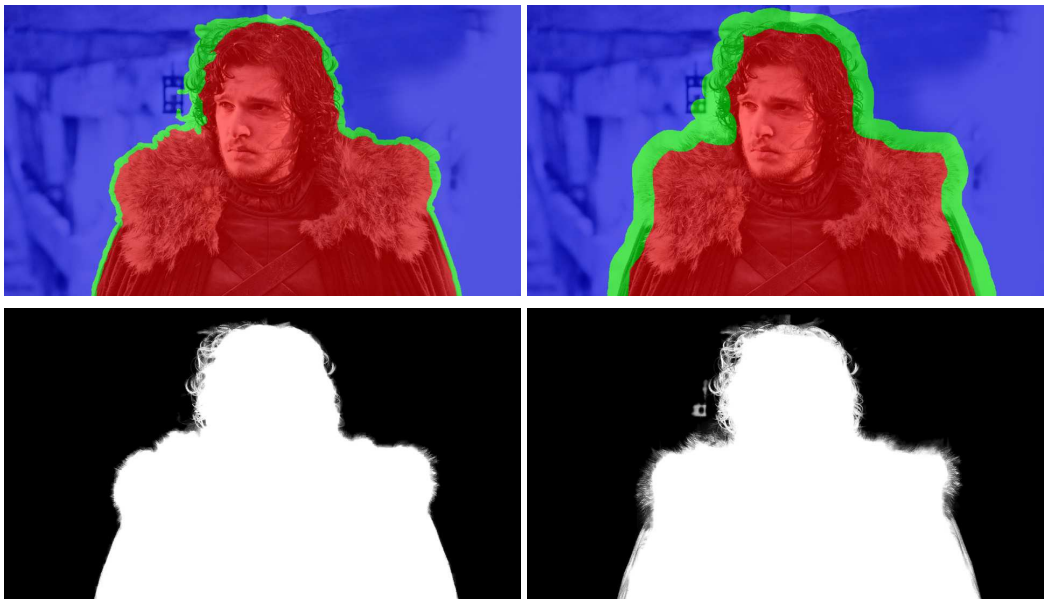


(a) (b)

Figure 6.3: Superimposed user painted trimap (a) and corresponding trimap used by matting algorithm (b).

The accuracy of the trimap is extremely important to the matting process. A perfectly accurate trimap's ambiguous region will only contain mixed pixels, thus reducing the number of unknown variables that need to be estimated. A small boundary region also increases the amount of background and foreground information that is available. Identifying the regions accurately requires a significant amount of user effort and skill, thus there is a trade off between the performance of the matting algorithm and the extent of the user interaction. Figure 6.4 illustrates this trade off. Column (a) shows an accurately painted trimap and its corresponding alpha matte, generated using the Robust Matting method. Whilst column (b) shows a trimap painted using a much bigger brush, thus saving time at the cost of accuracy. It can be seen that the matte generated from the roughly painted trimap assigns false transparencies to areas of the cloak, and includes some background objects.

The mattes are then used to form composite images with a plain background, allowing for the inaccuracies to been seen more clearly as illustrated in Figure 6.5.



(a) (b)

Figure 6.4: Trade off in accuracy between high level of user input (a) and faster user generation of tripmaps (b).

(a) (b)

Figure 6.5: Composite images of mattes generated in Figure 6.4. Inaccuracies in (b) can be seen along the edges of the cloak and in the background object appearing near the face.

6.3 Matting Techniques

Methods which extract (also known as pulling) alpha mattes can generally be divided into two approaches: sampling-based approaches and propagation-based approaches.

6.3.1 Sampling Methods

Sampling methods estimate the background and foreground components of a pixel by examining nearby pixels that have been specified as foreground or background by

the user. The colour of these sample pixels can be used directly to gain an estimate of the alpha value. Substituting z for (x, y) into Equation 6.1 for clarity, gives

$$I_z = \alpha_z F_z + (1 - \alpha_z) B_z. \tag{6.3}$$

For a pixel within the mixed region, I_z, it can be assumed that nearby background and foreground samples are close in colour space to the unknown foreground, F_z, and background, B_z, colours. The matting method will then obtain a good estimate of F_z and B_z from these nearby samples. These estimates can then be used in the matting equation, Equation 6.3, to determine the value of α_z. Colour sampling methods vary in the way in which samples are selected for the the estimation of F_z and B_z, and also in how these samples are used to obtain the estimates. A few of the more popular methods are reviewed in this section.

KnockOut

One example of a colour sampling method is the Knock Out technique [Berman et al., 2000a,b], one of the first successful matting methods for natural images. The method takes a user defined trimap and for a point I_z within the mixed region, F_z is calculated as the weighted sum of pixels along the boundary of the user defined foreground region. The weight of the nearest pixel is 1 which decreases linearly with distance (spatially) to 0 for pixels which are twice the distance from I_z as the nearest foreground pixel. The same process is used to estimate an initial value of the background component B'_z. This is refined to calculate B_z by considering the relative position of I_z and F_z. The alpha value is calculated for each different colour channel and the final value α_z is estimated via a weighted summation.

Parametric Sampling Methods

Some of the more successful colour sampling methods are based on statistical modelling and are sometimes grouped under the label of parametric sampling methods. Once samples have been collected for a particular pixel, I_z, these methods fit low order parametric statistical models to them, typically Gaussians. The distances between the unknown mixed pixel I_z and the foreground and background distributions are used to estimate the alpha value α_z.

In Ruzon and Tomasi's method [Ruzon and Tomasi, 2000] the boundary area is partitioned into subregions. For each subregion a local window is constructed that contains some of the user defined foreground and background regions. Pixels from these regions are treated as samples from a foreground and background colour

distribution respectively. Foreground and background pixels are split into clusters and unoriented Gaussians are fitted to each. All foreground clusters are linked to all background clusters and then some pairings are rejected based on a series of angle and intersection criteria. The observed colour of the pixel I_z is treated as a distribution in between the foreground and background distributions. This distribution is also defined as the sum of Gaussians, where each Gaussian is centred on the line between each linked foreground and background cluster. The fractional distance along which each Gaussian is centred is determined by the value of α. The optimal α is chosen such that the distribution for the observed colour of pixel I_z has maximum probability. F_z and B_z can be computed as a post process to satisfy Equation 6.3.

Baysian Matting [Chuang et al., 2001] is another statistical based method which improves upon Ruzon and Tomasi's method. The method also estimates the distributions for foreground and background regions in a similar fashion. However the windowing system used travels from the edge of the boundary region towards the region centre and incorporates calculated values for F, B and α for pixels in the neighbourhood when constructing oriented Gaussians, not just the user defined definite foreground and background regions. The matting problem is formulated in a well-defined Bayesian framework and is solved using the *maximum a posteriori* technique.

Parametric sampling methods do not generate satisfactory results if the regions are not smooth or their colour distributions are non-Gaussian. Trying to pull a matte from complex natural images without clear foreground and background colour distributions is problematic.

6.3.2 Propagation Methods

Propagation based methods make the assumption that in a given area, foreground or background pixels will not have sharp transitions, i.e., they are locally smooth. Colours can be modelled as either constant or transitioning in a linear fashion. Foreground and background colours an be eliminated from an optimisation process, allowing the matte problem to be solved.

Poisson Matting

The Poisson matting method [Sun et al., 2004a] assumes that intensity changes in the foreground and background are smooth in the immediate neighbourhood. This allows an approximation to show the matte gradient to be proportional to the

image gradient. F_z and B_z are chosen as the nearest (spatially) foreground and background pixels to I_z respectively. The final matte is produced by solving the Poisson equations on the image lattice. As with all propagation algorithms, the matte will tend to be smooth and not suffer from discontinuities.

Random Walks

The Random Walker algorithm for image matting [Grady et al., 2005] is based on the image segmentation method of the same name [Grady, 2006]. The image is modelled as a weighted graph with each pixel corresponding to a node. Edge weights are determined by the similarities between pixels. Taking user defined foreground and background pixels, for each unknown pixel the probability that a random walker will hit a foreground pixel is computed first, taking into account the weights between nodes. Based on these probabilities an alpha value is determined for each unclassified pixel.

6.4 Robust Matting

The Robust Matting method [Wang and Cohen, 2007] is selected for the purpose of evaluating the trimaps automatically generated by the segmentation algorithm. The method is shown by the authors to produce more accurate alpha mattes when compared to the other prominent matting algorithms. In the comparison, a variety of trimaps of differing quality are used to generate alpha mattes which are compared to a manually created ground truth to generate the mean squared error. Thus a quantitative evaluation is performed.

The matting method falls under the colour sampling area, which the authors argue is more robust than propagation based approaches for natural images, but also contains elements of a propagation method in the matte optimisation stage.

6.4.1 Limitations of Conventional Matting Algorithms

In order to understand the benefits of Robust Matting, two main problems with conventional matting extraction methods are first discussed. These are as follows:

1. Not fitting the linear model. Figure 6.6 illustrates this problem. Two clusters, a foreground and a background are shown in colour space, along with a horizontal interpolation line joining the cluster centres. Two pixels, A and B, defined as mixed by the user generated trimap are also shown. Point A is suitable for the linear model as it is close to the interpolation line. Point

B however is not, i.e., a linear combination of the two clusters is unlikely to estimate the correct foreground and background components, and such a pixel may not even be mixed but part of the background or foreground.

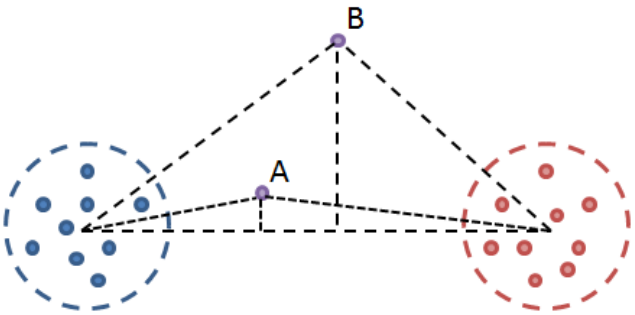


Figure 6.6: Illustration of an estimation problem involving two clusters corresponding to the blue dots and the red dots, and two pixels A and B. Pixel A fits the linear model represented by the horizontal line, whereas pixel B does not.

2. Non-uniformity of colour distribution. This problem occurs when the foreground and backgrounds are very complex and local regions do not have a uniformity of colour. As a result the assumption that propagation based algorithms make - that the background and foreground colours are smooth in the mixed/ambiguous band of the trimap - is not valid. Thus the resultant matte will not be accurate. Sampling based approaches will also produce an erroneous alpha matte if each sample is treated with equal weight.

The Robust Matting algorithm addresses these problems by calculating a confidence factor for each pair of samples. This is discussed in the next section.

6.4.2 Initial Matte Generation

The Robust Matting method makes the assumption that for an ambiguous pixel with colour C, I_z the true foreground (F_z) and background (B_z) colours will be close to (in colour space) some of the foreground and background samples taken. A good pair of samples will be able to describe a mixed pixel as a linear combination of the two samples. For a given pair of samples, F^i and B^j, the estimated alpha value is given by

$$\hat{\alpha} = \frac{(C - B^j)(F^i - B^j)}{\| F^i - B^j \|^2}. \tag{6.4}$$

The goodness of the sample pair is evaluated by a distance ratio, $R_d(F^i, B^j)$ which measures the ratio of the distances between the pixel colour, C, and the value it

would have been if predicated by the matte equation (Equation 6.3). This is given by

$$R_d(F^i, B^j) = \frac{\parallel C - (\hat{\alpha}F^i + (1-\hat{\alpha})B^j) \parallel}{\parallel F^i - B^j \parallel}. \tag{6.5}$$

This distance ratio favours samples pairs that have very different colours as $\parallel F^i - B^j \parallel$ will be large and thus the distance ratio smaller.

Since pixels that are close in colour space to pixels that are definite foreground or background are considered to be more likely to belong wholly to the background or foreground than be truly mixed. The following weights are also used:

$$w(F^i) = exp\{- \parallel F^i - C \parallel^2 / D_F^2\}, \tag{6.6}$$

and

$$w(B^j) = exp\{- \parallel B^j - C \parallel^2 / D_B^2\}, \tag{6.7}$$

where D_F and D_B are the minimum distance in colour space between the foreground or background sample and the current pixel, respectively $min_i(\parallel F^i - C \parallel)$ and $min_j(\parallel B^j - C \parallel)$. This allows a final confidence value, $f(F^i, B^j)$, for a given same pair to be calculated using the weights and the distance ratio, i.e.,

$$f(F^i, B^j) = exp\left\{-\frac{R_d(F^i, B^j)^2.w(F^i).w(B^j)}{\sigma^2}\right\}, \tag{6.8}$$

where σ is a constant and fixed at a value of 0.1. From every foreground and background sample pair, the three samples with the highest confidence are selected and the average of the estimated alpha values taken to generate an initial alpha matte. This average value, as well as the average of the confidence values are then taken into the next stage of the method.

6.4.3 Matte Optimisation

The previous stage of the algorithm obtains an initial estimate of alpha for each pixel, along with an associated confidence value. This initial estimation is taken into the next stage of the method where two assumptions are made to further improve the matte. The first assumption is that the matte is expected to be locally smooth. The second is that alpha values of 1 or 0 will be more common than truly mixed pixels, i.e., pixels in the ambiguous region are still more likely to be wholly background or wholly foreground than a mixture of the two. This type of pixel often have the lowest confidence scores.

Based on these assumptions, there are two constraints for the final matte -

a data constraint and a neighbourhood constraint. The data constraint is that the initial alpha generated should be respected, especially if the associated confidence value is high. The neighbourhood constraint is that the final matte should be relatively smooth and robust to image noise.

In order to generate an optimal matte subject to the constraints, the method considers the optimisation to be a graph labelling problem, which is later solved using a Random Walk. The alpha matte is treated as a graph, with each pixel being represented by a node and joined to its neighbours in the horizontal and vertical directions, i.e, a lattice. In addition, source and sink nodes, i.e., Ω_F and Ω_B, are virtual nodes respectively representing pure foreground and pure background. To satisfy the data constraint, a data weight is defined between each pixel and a virtual node. An edge weight defined between neighbouring pixels is used to satisfy the neighbourhood constraint.

Data Constraint

The relative probabilities that a node belongs to the background or the foreground are represented by the data weights. For nodes with high confidence values, $\hat{f}_i$, the initial alpha generated, $\hat{\alpha}$, is primarily used. Conversely, if the confidence is low, from the assumptions stated previously it is expected that the node is more likely to be fully background or foreground than a mixed pixel. Depending on the initial alpha estimate, the alpha of the node is biased towards foreground or background.

For a pixel in the ambiguous region i, the data weights, $W(i, F)$ and $W(i, B)$, between the pixel and respectively the two virtual nodes Ω_F and Ω_B are defined as

$$W(i, F) = \gamma[\hat{f}_i\hat{\alpha}_i + (1 - \hat{f}_i)\delta(\hat{\alpha}_i > 0.5)] \tag{6.9}$$

and

$$W(i, B) = \gamma[\hat{f}_i(1 - \hat{\alpha}_i) + (1 - \hat{f}_i)\delta(\hat{\alpha}_i < 0.5)] \tag{6.10}$$

where δ is a boolean function returning either 1 or 0, and γ is a parameter which balances the edge and data weights, set to be $\gamma = 0.1$. Setting γ too high gives a noisy matte whilst too low a setting will result in the matte being overly smooth.

Neighbourhood Constraint

The edge weight between node i and j, $W_(i, j)$, is specified to maintain the assumption that the alpha matte should be locally smooth. The weights between neighbouring pixels are based on the differences in their local colour distributions.

The same settings for the weights is used in the Closed Form Matting system [Levin et al., 2008].

The edge weight, $W_{(i,j)}$, is defined by a sum of all the 3×3 windows containing the nodes/pixels i and j, i.e.,

$$W_{ij} = \sum_{k}^{(i,j)\in w_k} \frac{1}{9}(1 + (C_i - \mu_k)(\Sigma_k + \frac{\epsilon}{9}I)^{-1}(C_j - \mu_k)), \tag{6.11}$$

where w_k represents the set of 3×3 pixels containing i and j, which k iterates over. μ_k and Σ_k are the colour mean and variance in each window, whilst ϵ is a regularisation coefficient, set at 10^{-5}, again using the same values and justification as Levin et al.'s method. I is the 3×3 identity matrix.

Solving the Problem for Optimal Alphas

The graph labelling problem is solved using a Random Walk as follows. For a given pixel i, the probability that a random walk starting at the pixel and reaching a foreground labelled pixel first is determined by solving a system of linear equation using the Conjugate Gradient (CG) method [Hestenes and Stiefel, 1952]. These linear equations are based on a Laplacian matrix containing the edge weights W between neighbouring pixels. The method is briefly outlined here.

A Laplacian matrix for the graph is first constructed as

$$L_{ij} = \begin{cases} W_{ii} & : \text{ if } i = j \\ -W_{ij} & : \text{ if i and j are neighbours,} \\ 0 & : \text{ otherwise,} \end{cases} \tag{6.12}$$

where $W_{ii} = \sum_j W_{ij}$. L is therefore a sparse, symmetric, positive-definite $N \times N$ matrix, where N is the total number of nodes comprising of all the image pixels and the virtual nodes Ω_B and Ω_F.

L is then decomposed into blocks corresponding to known nodes P_k (user labelled pixels and the virtual nodes) and unknown nodes P_u, i.e.,

$$L = \begin{bmatrix} L_k & R \\ R^T & L_u \end{bmatrix}, \tag{6.13}$$

where L_k is the matrix of the interactions between the known nodes, L_u between unknown nodes and R the mixed interactions, i.e., between known and unknown nodes.

It is shown by Grady [Grady, 2006] that the probability of an unknown pixel belonging to the foreground is the solution to

$$L_u A_u = -R^T A_k, \tag{6.14}$$

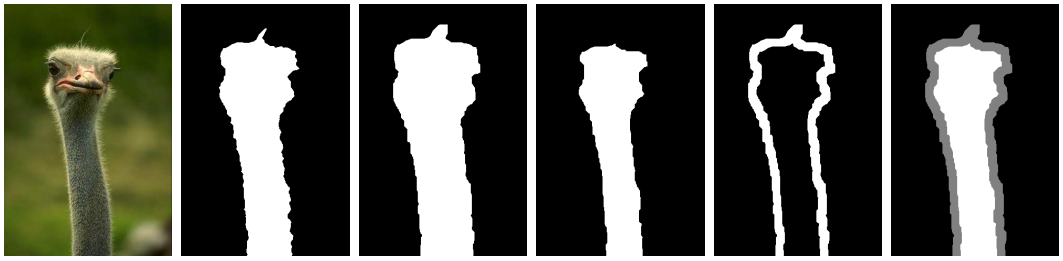
where A_u is the vector of unknown alphas to be solved and A_k is the vector encoding the boundary conditions (the known alphas and virtual nodes). The solution A_u is unique and guaranteed to exist, with entries of A_u each lying between 0 and 1. CG is used to solve the linear system.

6.5 Automatic Trimap Generation

In the previous chapters of this thesis an object segmentation method is proposed and extended to additionally segment video footage. In order to adapt the results to solve the matting problem the hard segmentations must be converted into trimaps. Given a binary segmentation S_b, the common approach [Li et al., 2005; Wang et al., 2005; Bai et al., 2009] is to perform an erosion and dilation of S_b by a fixed number of pixels to create the unknown region around the object's boundary. This band is given by the exclusive OR operation of the dilated image with the eroded image, i.e., the unknown band, S_u, corresponds to the difference between the eroded segmentation S_e and the dilated segmentation S_d, i.e.,

$$S_u = S_e \oplus S_d. \tag{6.15}$$

This process is illustrated **in** Figure 6.7.



(a) (b) (c) (d) (e) (f)

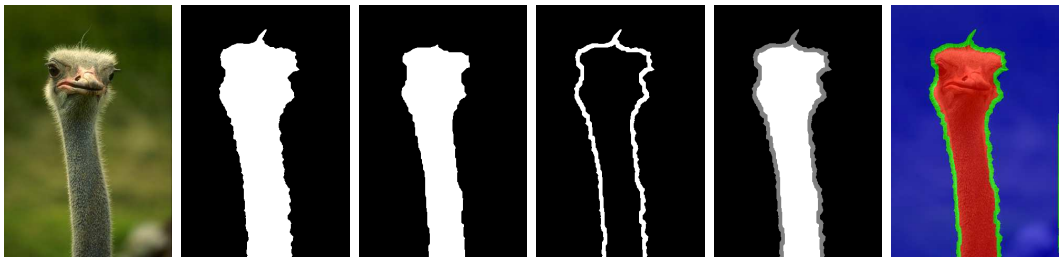
Figure 6.7: The process of automatically generating a trimap from a binary segmentation: (a) the original image; (b) the binary segmentation; (c) the dilation of the binary segmentation; (d) the erosion of the segmentation image; (e) the matting band or ambiguous region; and (f) the generated trimap (f).

As demonstrated earlier by Figure 6.4, the accuracy of the trimap is very

important if a good matte is to be obtained. An ideal trimap will only contain mixed pixels but in practice this is impossible. The nature of the active contours being an enveloping algorithm and stopping just before the object boundary allows for the removal of the dilation from the trimap generation process. Instead, in this thesis the matting band is generated from the difference in the eroded image and the original binary segmentation, i.e.,

$$S_u = S_b \oplus S_e. \tag{6.16}$$

This is illustrated in Figure 6.8.



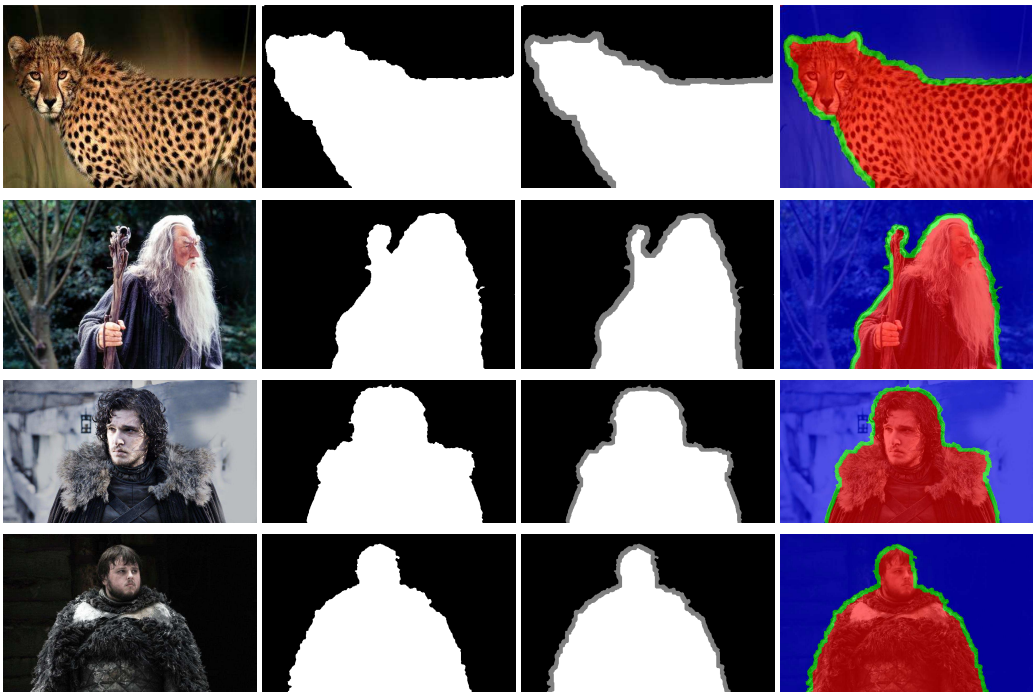
(a) (b) (c) (d) (e) (f)

Figure 6.8: The process of automatically generating a trimap from a binary segmentation, without a dilation operation: (a) the original image; (b) the binary segmentation; (c) the erosion of the segmentation image; (d) the matting band or ambiguous region; and (e) the generated trimap; and (f) the trimap superimposed on the original image.

Removing the dilation of the binary segmentation allows for a narrower, and thus more accurate matting band. A smaller ambiguous region also results in improvements in the computational efficiency of the algorithm as there are fewer unknown alphas to compute. The size of the $N \times N$ pixel structuring element used for the erosion is proportional to the size of the image and is determined by the following rule-of-thumb:

$$N = 2l_p + 10, \tag{6.17}$$

where l_p is 1% of the larger dimension (height or width) of the image. This prevents any discrimination between landscape and portrait photographs. N is then rounded to the nearest integer as the size of the structuring element must be in whole pixels. Figure 6.9 shows a series of images, and the trimaps generated for them.

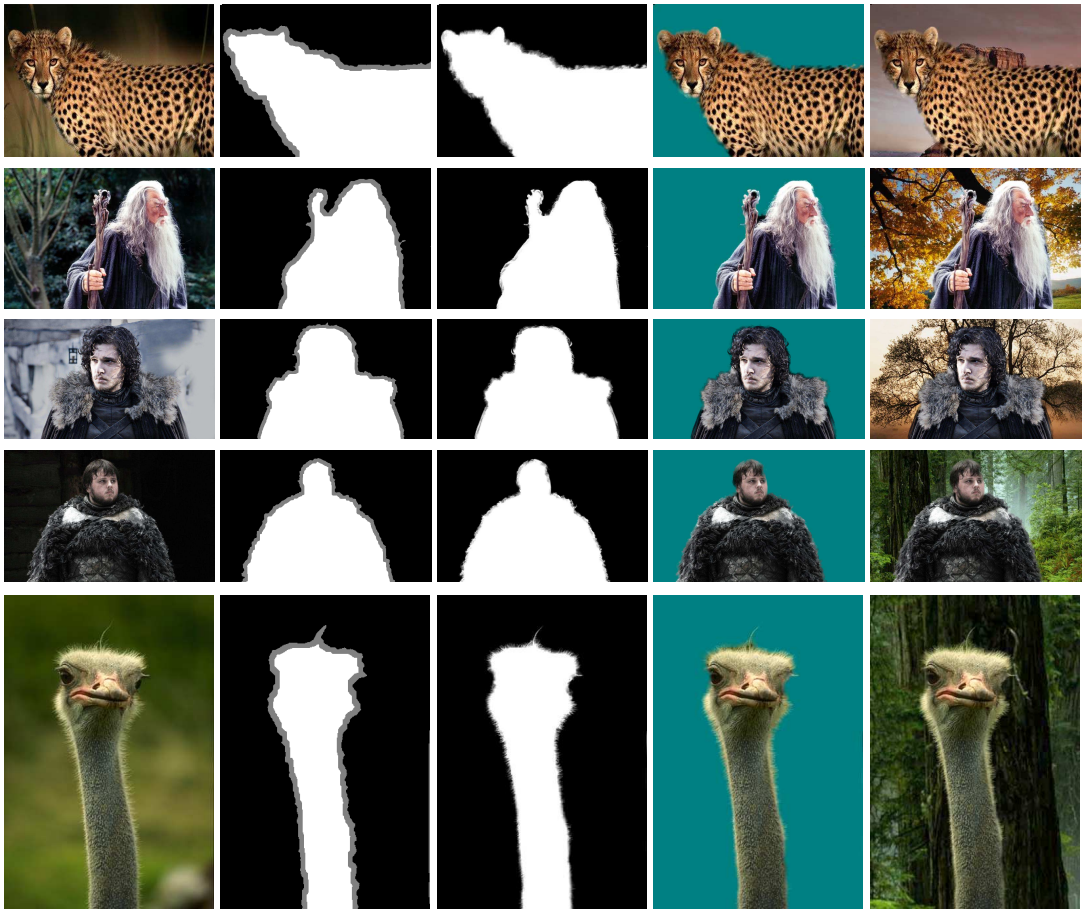


(a) (b) (c) (d)

Figure 6.9: Automatic generation of trimaps from binary segmentations: (a) original image; (b) segmented object; (c) automatically generated trimap; and (d) an overlay of the trimap onto the image, where green represents the matting band, red the object and blue the background.

6.6 Image Matting Results

The Robust Matting Algorithm is then applied to the automatically generated trimaps to produce corresponding alpha mattes and new image composites. A series of the results are shown in Figure 6.10. It can be seen that good alpha mattes are obtained, allowing for the objects to composite well with new backgrounds.



(a) (b) (c) (d) (e)

Figure 6.10: Automatic generation of trimaps from binary segmentations and corresponding alpha mattes and image composites: (a) original image; (b) automatically generated trimap; (c) alpha matte; (d) and (e) are respectively the object composited with a plain and detailed background.

6.6.1 Limitations

As with most automatic methods, there will be some cases for which the method of automatic trimap generation is not ideal. In this case, the method will not function

as intended for objects with long or varying lengths of mixed pixels. Additionally if the segmentation of the the OoI is not accurate, then it is unlikely that the resultant matte will be accurate. These two cases are illustrated in Figure 6.11. The first row shows an example where large portions of the character's hair are mixed or background, but are contained within the object region of the trimap. This results in composite images where part of the old background is still visible. The second row shows an example where the segmentation process failed to segment the hole in the soft toy (between the left ear and the body), and thus the trimap produced is inaccurate.

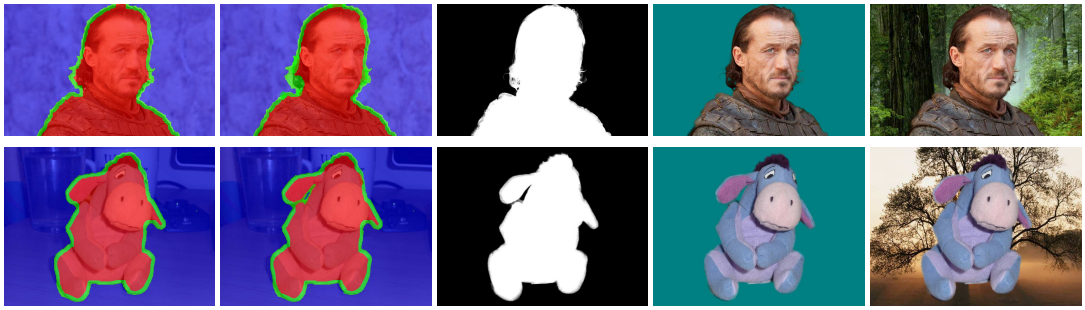
(a) (b) (c) (d) (e)

Figure 6.11: Limitations of automatically generating trimaps: (a) Original images; (b) the trimaps; (c) the resultant matt; and (d) and (e) are two composites.

The generation of such a trimap is not without merit as it provides a good foundation which can then be easily altered by the user. Most recent matting algorithms have some form of refinement procedure to enable user input to be performed if the user is unhappy with the initial matte. Figure 6.12 shows the effect of a limited amount of user input in modifying the initial matte by painting over some regions, and the resultant improvement in alpha mattes and composite images. These changes result in the improved mattes which in turn produce good composites shown in columns (d) and (e). Trimaps created in this fashion are still substantially quicker than if the user painted the different regions themselves.

6.7 Video Matting Results

For video sequence matting, the task of manually specifying a trimap for each individual frame becomes infeasible due the amount of effort and time required. Even a short video sequence may have hundreds of frames. Video matting methods typically require the user to generate trimaps for a few key frames and then automatically propagate these maps to other frames. In our case, having obtained binary segmen-



(a) (b) (c) (d) (e)

Figure 6.12: Limitations of automatic trimap generation overcome by a small amount of user input: (a) initial automatically generated trimap; (b) trimap modified by a user; (c) improved mattes; and (d) and (e) are the two resulting composites.

tations from each frame, the trimap generation method can simply be applied to generate trimaps automatically for each frame.

As an example, the automatic trimap generation process is applied to the video sequence of the swimming fish used in Chapter 5. Figure 6.13 shows the results of the method. This allows for the object, in this case a fish, to be composited onto a plain background in column (d) and then to be made to swim in outer space in column (e). It can be seen that good mattes are obtained from the automatically generated trimaps, and that the fish fits seamlessly in with the new background.

6.8 Conclusion

The method presented in this Chapter addresses the problem of unsupervised image matting. The methods presented in Chapters 4 and 5 are used to gain a binary segmentation of the OoI. The enveloping nature of the active contours algorithm is exploited, and erosions of the binary segmentations performed to automatically generate the matting band. This allows trimaps to be input into the Robust Matting algorithm to generate accurate alpha maps. A variety of mattes are extracted from natural images and video footage, and shown to produce good composites. The unsupervised method does not produce ideal trimaps for all cases and thus if a user is unhappy with a matte generated from a trimap, the trimap can be easily refined to produce a better result.

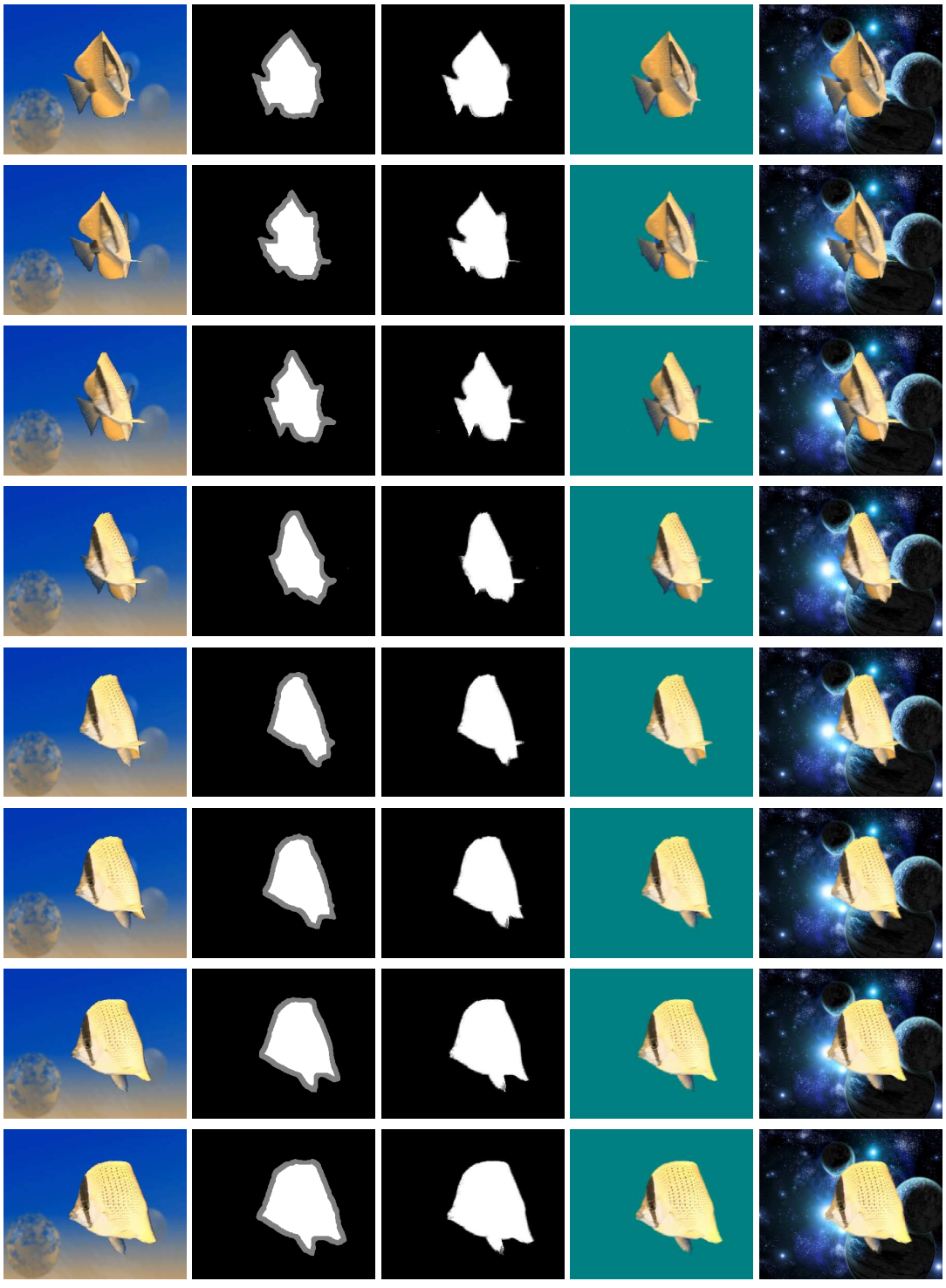


Figure 6.13: Automatic generation of trimaps for alpha matte generation to allow an object (i.e., a fish) to be composited onto a new background: (a) original video frame; (b) trimap; (c) alpha matte; and (d) and (e) are respectively the object composited onto a plain background and outer space.

Chapter 7

Silhouette Generation for 3D Object Reconstruction

7.1 Introduction

3D object reconstruction is a scientific discipline concerned with the generation of 3D models from sensor data. Whilst humans can very easily gain a lot of information from what they see, whether it be recognising a face or perceiving distance, it is significantly more difficult for machines to do so. Reconstructing depth information from images is intrinsically problematic as information in one dimension is lost when a 3D scene is projected onto a 2D image. Recovering this information has been a popular theme of research in the field of computer vision.

There are many different approaches to reconstructing a 3D object. Methods can generally be divided into one of two areas: active methods and passive methods. Active methods involve some form of interaction with the object to be reconstructed. This usually either involves a laser or the projection of a light pattern onto the object together with a sensor. The high accuracy required for the laser or projector often means that such methods are prohibitively expensive, and thus tend to be only practical for industrial applications where the accuracy and detail of the 3D reconstructions is highly desirable.

Passive methods do not interact with the object and typically comprise of one or two cameras, sensitive to visible light, which are used to capture images of the object and infer its 3D volume via some method. Techniques involving two cameras are known as stereo methods and use a triangulation approach to determine the depth of 2D points [Seitz et al., 2006]. Other reconstruction techniques use visual cues such as shading or texture to perform the reconstruction [Trucco and Verri,

1998]. The shape from silhouettes method (SfS) [Laurentini, 1994] is another passive method which captures silhouettes of an object from multiple different camera view points and backprojects them to create a visual hull (VH) to represent the object's volume. It is a particularly good approach if only simple 3D representation is required.

An existing system [Shin, 2008] consists of a black velvet turntable surrounded by a black velvet background, to prevent any cluttering of the background. A fixed camera, aimed at the turntable, is then calibrated using a calibration pattern before an object is placed on the turntable and a series of images captured at different angles. These are segmented using interactive thresholding techniques to obtain the silhouettes from each angle. The SfS method is then used to obtain a VH whose information is stored in an octree format. A surface can then extracted using a variety of methods.

This chapter is concerned with adapting the object segmentation method presented in this thesis to automatically generate silhouettes for use with an existing 3D object reconstruction system, rather than rely on manual thresholding and the use of a bulky backdrop as its background. The remainder of the chapter is organised as follows. In Section 7.2 the 3D object reconstruction system is briefly described. Section 7.3 presents the modifications to the system in the data acquisition and silhouette generation stages of the process. Finally in Section 7.4 a sample of automatically generated silhouettes and the subsequent 3D models created are shown. The Chapter is concluded in Section 7.5.

7.2 Silhouette Based 3D Object Reconstruction System

The SfS-based 3D object reconstruction system [Shin, 2008] consists of five phases which are described briefly in this section to provide the context for which the proposed automatic silhouette generation method is operating in:

- Data Acquisition;
- Camera Calibration;
- Silhouette Extraction;
- Octree Generation;
- Surface Generation.

The SfS method requires images to be captured from multiple camera views around an object. Silhouettes (i.e., binary segmentations of the object) can then

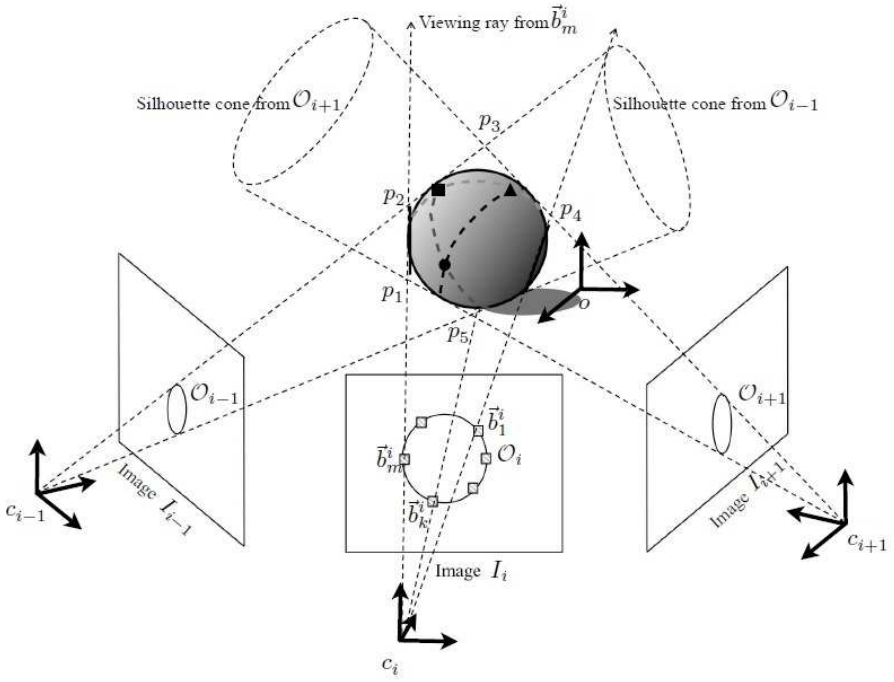


Figure 7.1: SfS-based 3D object reconstruction method. VH is created via the intersection of cones generated from several camera views. This figure has been adapted from [Shin, 2008].

be extracted from each of these view points. The premise is that the volume of an object can be approximated by the intersection of cones from the back projection of rays along the silhouette boundary. This is illustrated in Figure 7.1.

To obtain the images the object is placed on a turntable. To simplify the task of silhouette extraction, the turntable and the backdrop of the scene are covered with black velvet cloth as shown in Figure 7.2. The camera is fixed in place and the turntable is rotated 6 degrees between images, to make datasets of 60 views over a complete 360 degree view of the object.

A camera calibration pattern, consisting of a known geometric entity of two grids of black and white squares, orthogonal to each other (as shown in Figure 7.2), is used to obtain the camera parameters. A calibration matrix is computed for one view and extrapolated to produce calibration estimates for the other 59 view points.

The next stage of the reconstruction method is to extract the silhouettes from the captured data. This is the same as obtaining the binary OoI segmentation for each image. To achieve this an interactive thresholding program is used. The background surrounding the object is entirely black which makes this is a relatively

(a) (b)

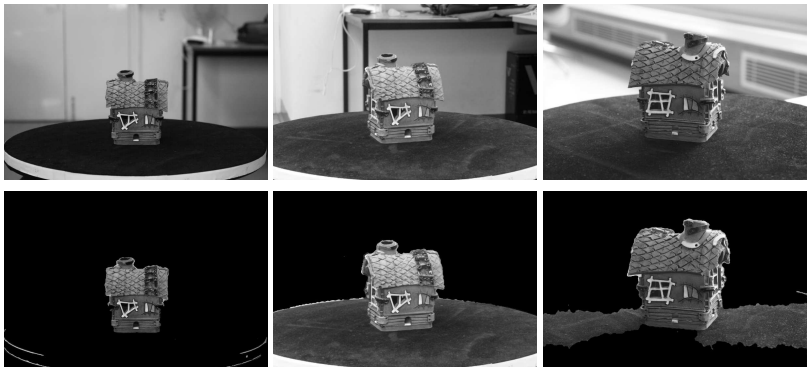
Figure 7.2: The 3D object reconstruction system: (a) the camera and background setup; and (b) the camera calibration pattern.

straight forward process. A variety of thresholding techniques can be used based on the judgement of the user, and applied to all of the image dataset. Operations include various thresholds, and pre and post processes including blurring, hole filling, histogram equalisation, adaptive thresholding and the ability to allow for internal holes. As such, the quality of the resulting silhouettes is partially determined by the skill of the operator, and one threshold setting might not be suitable for all images in the dataset. Additionally, if parts of the object are very dark then a thresholding process might produce erroneous silhouettes.

Having obtained the silhouettes and calibrations for each view point, the SfS method can be used to construct the VH of the OoI. The 3D object is stored in the form of an octree, a hierarchical tree data structure used to partition 3D space by repeatedly subdividing it into 8 child octants. Using the octree representation of the object, various algorithms can be used to extract a surface. For the purposes of this thesis an implementation of the marching cubes algorithm [Lorensen and Cline, 1987] is used to generate the example surfaces in the results section.

7.3 Automatic Silhouette Generation

The 3D object reconstruction system is modified in two ways to create a more accessible and faster method for reconstructing 3D objects. Firstly in the data acquisition phase the need for a bulky black velvet background is removed. Secondly, the silhouette extraction process is automated. These changes are presented in this section.



(a) (b) (c)

Figure 7.3: Difficulties associated with segmenting a focused OoI from a focused turntable: (a) Presence of strong contrasting edges; (b) both the turntable and OoI are sufficiently in focus; and (c) texture on turntable.

7.3.1 Image Acquisition

The image acquisition process is the same as described in Section 7.2, with the exception that the need for the bulky black velvet background is removed. The camera is set up pointing at an object placed on the centre of the turntable. In our examples the camera used is a DSLR Cannon 450D and is set to aperture priority with an F value of 5.6. This enables images with a relatively low DoF to be captured. The camera is autofocused to the object on the turntable and then the autofocus is turned off, i.e., the camera parameters are fixed. An image of the calibration pattern is then captured using the fixed camera parameters to calibrate the camera. This is followed by capturing the 60 images of the object to generate the dataset for one object, with the turntable being rotated by approximately 6 degrees between two image captures. The direction of the rotation is noted for calibration purposes.

The main difficulty with using the low DoF segmentation method to generate the silhouettes is that regions of the turntable will also be within the DoF. Figure 7.3 illustrates this problem. In column (a) part of the strongly contrasting turntable edge is included in the segmentation. In column (b) both the turntable and the OoI have been segmented because they are both sufficiently in focus when compared to the background. Finally in column (c) dust on the turntable has given it sufficient texture that it returns a focus value within the DoF and thus is segmented along with the model house.

In order to address the above mentioned difficulty, the system is set up such that the turntable is considered out of focus. This is relatively easy to achieve as the turntable is fairly homogeneous, being made entirely of black velvet. Making

sure the DoF is sufficiently low, a relatively flat (in relation to the OoI and the horizontal place) perspective is used during the image capture. This counters the dust effect. To counter the other two effects, the camera is positioned and zoomed in sufficiently such that the front edge of the turntable is not included in the image. The flat angle and the low DoF ensure that the homogeneous turntable does not return high enough focus values to be segmented and ensures that the rear edge of the turntable is out of focus and not sharp enough to return a high focus value.

7.3.2 Image Segmentation

For the low DoF video object segmentation method presented in Chapter 5, two assumptions were made; that there was no change of scene within the video sequence; and that there were no large discrepancies in image composition from one frame to another. Both these assumptions are valid for the sequence of images captured every 6 degrees of the OoI on the turntable. Thus, the 60 sequential images of the OoI are treated as a video sequence of 60 frames and segmented in exactly the same way as described in Chapter 5, namely the active contour for the first frame (or in this case, first image in the sequence) is initialised using the focus intensity maps generated from the first, third and fifth image, and subsequent initial contours are generated from the binary dilation of the previous frame's final segmentation. This allows for a fast and robust segmentation of the dataset of 60 images to give the silhouettes of the object. The remainder of the 3D object reconstruction process is followed as described in Section 7.2.

7.4 Results

Some examples of automatically generated silhouettes and the reconstructed 3D models are presented in this section. The first object for which a 3D object reconstruction is performed is that of a model house. Figure 7.4 shows the dataset from which the 3D reconstruction is performed. Figure 7.5 shows the automatically generated binary segmentations for each of the 60 view points. To provide some context for the binary segmentations, the segmented object from each of the different view points is shown in Figure 7.6.

Figure 7.4: Greyscale images of a model house, taken every 6 degrees of rotation of the turntable.

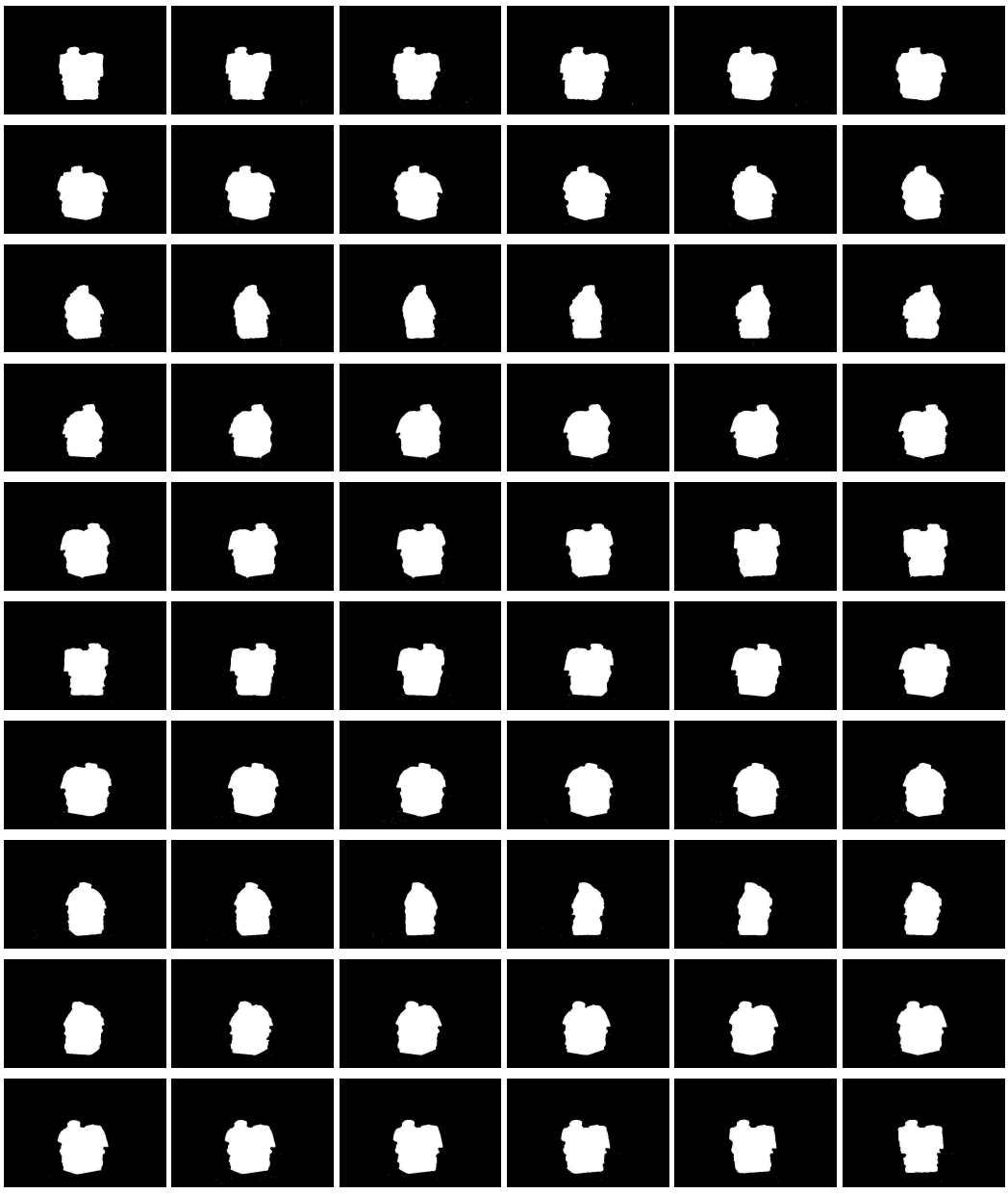


Figure 7.5: Binary segmentations of a model house generated for every 6 degrees of rotation of the turntable.

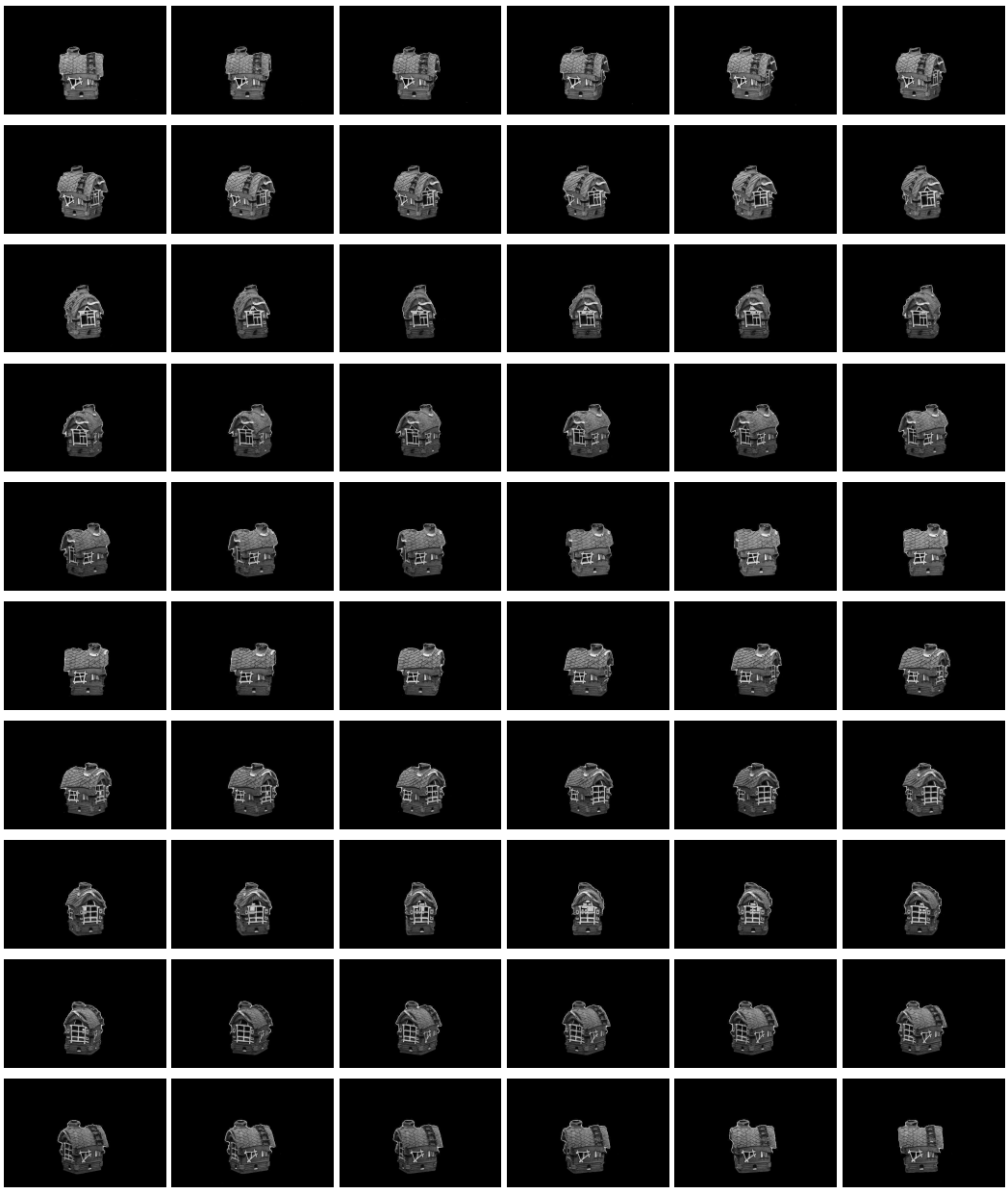
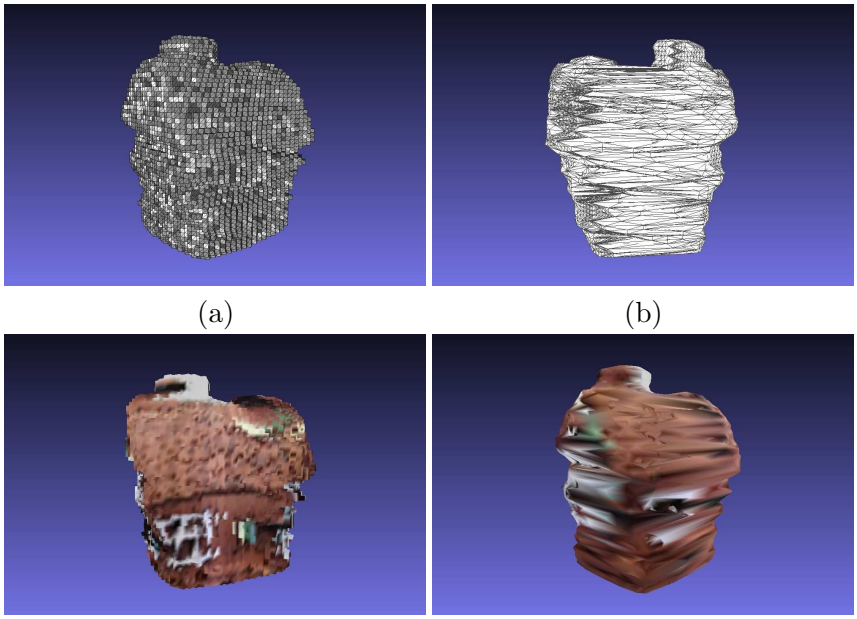


Figure 7.6: Segmentations of a model house generated for every 6 degrees of rotation of the turntable.



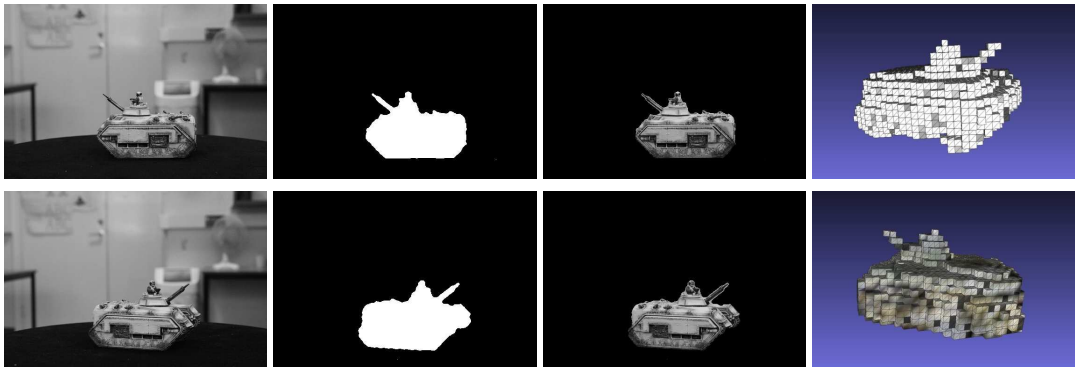
(a) (b)

(c) (d)

Figure 7.7: 3D reconstruction of a model house: (a) the octree representation; (b) the reconstructed 3D surface; (c) the octree representation with the estimated object surface colour (d); and the 3D surface model with added colour.

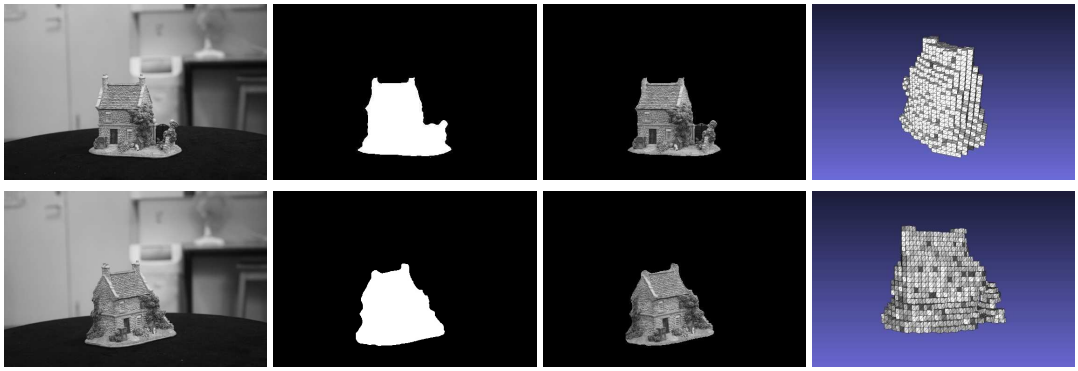
The binary segmentations shown in Figure 7.5 are then used in the SfS method to generate a 3D model of the house. This is illustrated in Figure 7.7, where (a) is the octree representation of the object, (b) shows the 3D surface extracted from the octree representation, and (c) and (d) respectively show the octree and surface representations with colours from the original image projected onto them.

Two more examples of object reconstructions are presented, with a sample of original images and segmentations. Figure 7.8 shows the reconstruction of a model tank, whilst Figure 7.9 shows that of another model house. The reconstructed object models are represented in octree format.



(a) (b) (c) (d)

Figure 7.8: Segmentation and 3D reconstruction of a model tank: (a) the acquired data; (b) the binary segmentations; (c) the segmented objects,;and (d) the resultant octree and coloured octree.



(a) (b) (c) (d)

Figure 7.9: Segmentation and 3D reconstruction of a model house: (a) the acquired data; (b) the binary segmentations; (c) the segmented objects; and (d) the resultant octree representation.

7.5 Conclusion

In this chapter an existing SfS based 3D object reconstruction system has been modified to remove the need for human input. In addition a bulky black backdrop is no longer needed for the silhouette generation process. The method treats the dataset of 60 images, taken every 6 degree rotation of the turntable, as frames in a video sequence and applies the unsupervised video segmentation method presented in Chapter 5 to automatically generate a binary segmentation for each view point. The silhouettes generated are shown to be accurate and are used to construct 3D models of a variety of objects.

Chapter 8

Conclusion and Further Work

8.1 Conclusions

This thesis addresses the problem of autonomous object segmentation and presents a method which allows for the automatic extraction of an OoI from a background scene in low DoF images. To achieve this goal, various focus assessment algorithms are studied and compared (Chapter 3). A multiscale wavelet based focus assessment method is proposed and used to generate focus intensity maps which allow the focused foreground and defocused background to be differentiated. In order to segment the regions of the focus map with a high intensity (i.e., the foreground) an active contours model is adopted (Chapter 4). A grid based method is proposed to generate the initial contour, and Whitaker's narrow band level sets implementation of Active Contours without Edges is used to separate the OoI from the background. This provides a relatively fast and robust method of segmenting objects from low DoF images. In Chapter 5 the image segmentation method is expanded to work with low DoF video sequences. A robust initial initialisation for the first frame is proposed, and subsequent initial contours are generated from the dilation of the previous frame's binary segmentation, thus improving the computational efficiency of the video segmentation method. The thesis then looks at two potential applications for the image and video segmentation methods. The first (Chapter 6) applies the segmentation method to the problem of autonomous image matting. The method generates trimaps automatically using the enveloping nature of the active contours algorithm and an erosion operation to automatically generate trimaps for use with the Robust Matting method. This allows for OoIs from both image and video sequences to be composited onto new backgrounds. The second application (Chapter 7) adapts the video segmentation method to segment a sequence of images of objects

rotated on a turntable. These extracted binary segmentations are then input into an existing 3D reconstruction system and used to reconstruct 3D models of objects via the SfS method.

In Chapter 2 the concepts of focus and DoF are introduced to provide an overview of the themes explored by this thesis. Image segmentation and the specific case of object segmentation are also explored, and several popular techniques and methods described.

A series of focus assessment methods are evaluated in Chapter 3 for their suitability in differentiating a focused OoI from a cluttered background. They can be classified under three categories, derivative and kernel based methods, statistical methods, and wavelet based methods. They are compared based on two desirable qualities, namely that the average intensity of the OoI should be higher when compared to the background, and the background region should return relatively homogeneous focus intensities. It is found that no particular method is suitable for all image resolutions and thus a multiscale wavelet based method is proposed to identify high frequency regions. This is based on a method proposed by Yang and Nelson [Yang and Nelson, 2003b] and uses the standard deviation of the focus values as an indication of whether the level of wavelet decomposition is suitable to be able to differentiate the focused OoI from the defocused background. The method produces a focus intensity map of the image which enables to the focused OoI to be differentiated from the background.

The method used to segment the focused regions from the focus intensity map is introduced in Chapter 4. A level set representation of the Active Contours without Edges model is chosen as it enables segmentations even when there is not a clearly defined gradient or when there are discontinuities in the object's boundaries. This is a desirable feature for segmenting the focus map. The SFM (a narrow band implementation) is used to implement the active contours. This dramatically improves computational efficiency by only performing calculations near the zero level set. It also has the added benefit of internal contours not being able to spontaneously appear, meaning that given a good initial contour the segmentation method can still segment OoIs with weak textures and largely homogeneous regions. To generate the initial contour, a grid based approach is proposed. The intensity map is divided into squares and each square assigned the value of its pixels' maximum focus value. Thresholding is then used to generate the initial contour. The method compares favourably to other low DoF segmentation methods and also produces results of a similar quality to some of the most accurate interactive general segmentation methods. The method is limited by its inability to segment holes within objects,

unless the hole is not part of the 'object' of the initial contour. There are also a number of limitations associated with all low DoF methods - namely that there must be a focus differential present within the image, and that with a low DoF the OoI may encompass regions both inside and outside of the DoF and is thus difficult to be wholly segmented.

Chapter 5 of this thesis expands the image segmentation method to work with low DoF video footage. For the first frame in a video sequence, the initial contour is given using the grid based method, but taking the maximum from across the $n = 1, 3, 5$ frames, not just the first frame. This increases the robustness of the method, making sure that the OoI is encompassed by the initial contour. Subsequent initialisations are generated from the dilation of the previous frame's binary segmentation, thus providing a fast and robust way of segmenting video footage.

In Chapter 6 the segmentation method is applied to the problem of image and video matting. Due to the enveloping nature of the active contours algorithm the matting band is obtained through the difference between the binary segmentation of the object and an eroded version of the segmentation. Pixels in this region are considered to be ambiguous and may contain mixed elements of both the foreground and the background. The Robust Matting algorithm is used to estimate the alphas in this region and a variety of accurately produced alpha mattes are shown, even for the difficult cases of hair and fur along an object's boundary. The mattes are used to perform several realistic looking scene composites. For cases where an initial matte has some inaccuracies, the user can refine the trimap which still saves time when compared with a user manually painting the entire trimap. The method is also shown to be successful when automatically applied to a video sequence and is used to perform a scene composition from video frames.

Finally in Chapter 7 the video segmentation method is expanded to automatically extract silhouettes as part of an existing 3D object reconstruction system, replacing the need for a bulky black velvet backdrop and an interactive thresholding process. Silhouettes (binary segmentations of the OoI) are extracted from datasets of 60 images of an object which is rotated on a turntable in 6 degree increments. As the dataset will be sequential in nature, this is exploited and rather than segmenting each image as an isolated case, the video segmentation method is applied to improve speed and accuracy. A 3D model is reconstructed using the SfS method and represented as an octree. The object surface can subsequently be extracted and its colour estimated. Several examples are shown for 3D models successfully reconstructed using the automatic silhouette generation process.

8.2 Further Work

In this section some potential avenues for further research and refinement of the methods presented in this thesis are discussed.

8.2.1 Focus Assessment using Colour Channels

The focus assessment method in Chapter 3 takes a greyscale image and performs the wavelet based focus assessment on it. It is considered an advantage being able to operate on greyscale images as they require less storage space. However, it also means that the algorithm is not making full use of the information available if colour images are used as its inputs. The method could be potentially improved by making use of all three colour channels instead of just the greyscale image.

This could be investigated by performing the focus assessment on each of the colour channels (r,g,b) separately, and either taking the mean,

$$F_{(x,y)} = \frac{1}{3}(F_{r(x,y)} + F_{g(x,y)} + F_{b(x,y)}) \tag{8.1}$$

or the maximum,

$$F_{(x,y)} = \max(F_{r(x,y)} + F_{g(x,y)} + F_{b(x,y)}) \tag{8.2}$$

from across the colour channels. This could result in improved and more robust focus assessments, although it could increase the prominence of focus values of edges (as this is where the most pronounced colour change will be).

8.2.2 Multi-channel Active Contours

The use of active contours for multi-channel segmentations has been proposed previously [Sandberg and Chan, 2005; Estellers et al., 2011] and could be particularly applicable to the segmentation method proposed in this thesis. The basic premise is that active contours are run concurrently on a variety of different image channels and some logical framework used to determine the final segmentation. Not only could all the focus assessments of the different colour channels be used in the segmentation process, but the use of the different levels of wavelet decomposition in the focus assessment could be investigated, as well as the use of various logical frameworks. This could potentially improve the robustness and quality of the segmentations obtained.

8.2.3 Adaptive Trimap Creation for Image Matting

One of the weaknesses of the automatic trimap generation method proposed in Chapter 6 is that currently it uses a 'one size fits all' approach to erode the binary segmentation to form the matting band of the trimap from the difference between the original binary segmentation and the eroded segmentation. Obviously this will not be suited to all images and whilst the user can refine the trimap afterwards, a fully autonomous method is desirable.

There is the potential to develop some form of adaptive trimap generation method. Strong textures such as hair and fur are often the most important areas where image matting is concerned and thus it might be possible to identify these regions as they will return the highest focus values due to their high frequency components. A different approach could look at the boundary regions which do have a clearly defined boundary. The matting band could be made narrow in these regions and wider elsewhere in the more 'busy' regions where pixels are more likely to be a combination of both foreground and background.

8.2.4 Matting for 3D Object Reconstruction

Due to the focus assessment returning high values for focus where there are large changes in image intensity, this means that the pixels on either side of an object boundary will return high focus values. Thus the active contour is likely to stop just before the object boundary. Whilst this makes sure the entire object is enveloped by the contour, and is indeed useful in the automatic trimap generation, it does mean the silhouettes generated for 3D object reconstruction are not as accurate as they could be.

There is the potential to investigate the use of a matting algorithm (followed by a thresholding) to generate silhouettes that are more accurate for 3D object reconstruction purposes. In addition the Robust Matting algorithm used in this thesis generates a confidence value associated with each alpha value in the alpha map. Both the confidence value and alpha value could be utilised when constructing surfaces for 3D models.

References

T. Adamek. Using contour information and segmentation for object registration, modelling and retrieval. *Ph.D.Dissertation, Dublin City University*, 2006.

R. Adams and L. Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–647, 1994.

G. Aubert and L. Vese. A variational method in image recovery. *SIAM Journal on Numerical Analysis*, 34(5):1948–1979, 1997.

X. Bai, J. Wang, D. Simons, and G. Sapiro. Video snapcut: robust video object cutout using localized classifiers. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, 28(70):1–11, 2009.

A. Berman, A. Dadourian, and P. Vlahos. Method for removing from an image the background surrounding a selected object. *U.S. Patent 6,134,346*, 2000a.

A. Berman, P. Vlahos, and A. Dadourian. Comprehensive method for removing from an image the background surrounding a selected object. *U.S Patent 6,134,345*, 2000b.

H. Bock. Origins and extensions of the k-means algorithm in cluster analysis. *Electronic Journal for History and Probability of Statistics*, 4(2):1–18, 2008.

Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *Proceedings of the IEEE international conference on Computer Vision*, 1:105–112, 2001.

V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numerical Mathematics*, 66:1–31, 1993.

V. Caselles, R. Kimmel, and G. Sapire. On geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.

C. Chow and T. Kaneko. Automatic boundary detection of the left ventricle from cineangiograms. *Computers and Biomedical Research*, 5(4):388–410, 1972.

Y. Chuang, B. Curless, D. Salesin, and R. Szeliski. A bayesian approach to digital matting. *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 264–271, 2001.

J. Daugman. How iris recognition works. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004.

H. Digabel and C. Lantuejoul. Interactive algorithms. *Actes du Second Symposium Européen d'Analyse Quantitative des Microstructures en Sciences des Matériaux, Biologie et Médecine*, pages 85–99, 1977.

V. Estellers, D. Zosso, X. Bresson, and J. Thiran. Harmonic active contours for multichannel image segmentation. *IEEE International Conference on Image Processing*, pages 3141–3144, 2011.

R. Fielding. *The Technique of Special Effects Cinematography.* Focal/Hastings House, London, 1972.

L. Firestone, K. Cook, K. Culp, N. Talsania, and K. Preston. Comparison of autofocus methods for automated microscopy. *Cytometry*, 12(3):195–206, 1991.

C. Gentile, O. Camps, and M. Sznaier. Segmentation for robust tracking in the presence of severe occlusion. *IEEE Transaction on Image Processing*, 13(2):166–178, 2004.

L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1–17, 2006.

L. Grady, T. Schiwietz, S. Aharon, and R. Westermann. Random walks for interactive alpha-matting. *Proceedings on Visualisation, Imaging and Image Processing*, pages 423–429, 2005.

F. Groenand, I. Young, and G. Ligthart. A comparison of different focus functions for use in autofocus algorithms. *Cytometry*, 12:81–91, 1985.

M. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):406–436, 1952.

S. Horowtiz and T. Pavlidis. Picture segmentation by a directed split and merge procedure. *Proceedings of the International Conference on Pattern Recognition*, pages 424 – 433, 1974.

Z. Hou and C. Han. Force field analysis snake: an improved parametric active contours model. *Pattern Recognition Letters*, 26:513–526, 2005.

W. Huang and Z. Jing. Evaluation of focus measures in multi-focus image fusion. *Pattern Recognition Letters*, 28(4):493–500, 2007.

A. Jain. *Fundamentals of Digital Image Processing.* Prentice Hall, 1998. ISBN 978-0133361650.

J. Jang, K. Park, J. Kim, and Y. Lee. New focus assessment method for iris recognition systems. *Pattern Recognition Letters*, 29(13):1759–1767, 2008.

O. Juan and Y. Boykov. Active graph cuts. *IEEE conference on Computer Vision and Pattern Recognition*, 1:1023–1029, 2006.

B. Kang. A study on fast iris restoration based on focus checking. *Proceedings of the 4th international conference on Articulated Motion and Deformable Objects*, pages 19–28, 2006.

B. Kang and K. Park. A study on iris image restoration. *Proceedings of the 5th international conference on Audio- and Video-Based Biometric Person Authentication*, pages 31–40, 2005.

K. Kass, A. Witkin, and D. Terzopoulos. Snakes: active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.

C. Kim. Segmenting a low-depth-of-field image using morphological filters and region merging. *IEEE Transactions on Image Processing*, 14(10):1503–1511, 2005.

C. Kim, J. Park, and J. Hwang. Video object extraction for object-orientated applications. *Journal of VLSI Signal Processing*, 29:7–21, 2001.

C. Kim, J. Park, J. Lee, and J. Hwang. Fast extraction of objects of interest from images with low depth of field. *ETRI Journal*, 29(3):353–362, 2007.

B. Ko and H. Byun. Frip: a region-based image retrieval tool using automatic image segmentation and stepwise boolean and matching. *IEEE Transactions on Multimedia*, 7(1):105–113, 2005.

S. Lai, C. Fu, and S. Chang. A generalized depth estimation algorithm with a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(4): 405–411, 1992.

S. Lankton. Sparse fields method technical report. *Shawn Lankton Online, www.shawnlankton.com*, 2009.

A. Laurentini. The visual hull concept for silhouettes-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):150–162, 1994.

A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (2):228–242, 2008.

H. Li and K. Ngan. Unsupervised video segmentation with low depth of field. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(12):1742–1751, 2007.

Y. Li, J. Sun, and H. Shum. Video object cut and paste. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, 24(3):595–600, 2005.

Z. Liu, W. Li, L. Shen, Z. Han, and Z. Zhang. Automatic segmentation of focused objects from images with low depth of field. *Pattern Recognition Letters*, 31(7): 572–581, 2010.

W. Lorensen and H. Cline. Marching cubes: A high resolution 3d surface reconstruction algorithm. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, 21(4):163–169, 1987.

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. *Proceedings of the 8th International Conference on Computer Vision*, 2:416–423, 2001.

G. Dal Maso, J. Morel, and S. Solimini. A variational method in image segmentation: existence and approximation results. *Acta Matematica*, 168:89–151, 1992.

K. McGuiness and N. O'Conor. A comparative evaluation of interactive segmentation algorithms. *Pattern Recognition*, 43(2):434 – 444, 2010.

T. Meier and K. Ngan. Video segmentation for contents-based coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(8):1190–1203, 1999.

E. Mortensen and W. Barrett. Intelligent scissors for image composition. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, pages 191–198, 1995.

D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

S. Nayar and Y. Nakagawa. Shape from focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):824–831, 1994.

N. Ngan and H. Li. *Video segmentation and its applications*. SpringerLink : Bücher. Springer New York, 2011. ISBN 9781441994820.

S. Osher and J. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulation. *Journal of Computational Physics*, 79:12–49, 1988.

N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

N. Pal and S. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

A. Pentland. A new sense for depth of field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(4):523–531, 1987.

T. Porter and T. Duff. Compositing digital images. *Computer Graphics*, 18(3): 253–259, 1984.

C. Rother, V. Kolmogorov, and A. Blake. Grabcut: interactive foreground extraction using iterated graph cuts. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, 23(3):309–314, 2004.

L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60:259–268, 1992.

M. Ruzon and C. Tomasi. Alpha estimation in natural images. *Computer Vision and Pattern Recognition*, pages 18–25, 2000.

P. Salembier and L. Garrido. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Transactions on Image Processing*, 9(4):561–576, 2000.

B. Sandberg and T. Chan. A logic framework for active contours on multi-channel images. *Journal of Visual Communication and Image Representation*, 16(3):333–358, 2005.

S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:519 – 528, 2006.

Y. Shi and W. Karl. A fast level set method without solving pdes. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2:97–100, 2005.

D. Shin. Robust surface modelling of visual hull from multiple silhouettes. *PhD Thesis, University of Warwick*, 2008.

A. Smith and J. Blinn. Blue screen matting. *23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 259–268, 1996.

M. Subbarao, T. Choi, and A. Nikzad. Focusing techniques. *Journal of Optical Engineering*, 32:2824–2836, 1993.

J. Sun, J. Jia, C. Tang, and H. Shum. Poisson matting. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, pages 315–321, 2004a.

Y. Sun, S. Duthaler, and B. Nelson. Autofocusing in computer microscopy: selecting the optimal focus algorithm. *Microscopy Research and Technique*, 65(3):139–149, 2004b.

J. Tenenbaum. Accommodation in computer vision. *Ph.D. Thesis, Stanford University*, 1970.

S. Todorovic and N. Ahuja. Unsupervised category modelling, recognition, and segmentation in images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(12):2158–2174, 2008.

E. Trucco and A. Verri. *Introductory techniques for 3-D computer vision.* Prentice Hall PTR, 1998. ISBN 0132611082.

D. Tsai and H. Wang. Segmenting focused objects in complex visual images. *Pattern Recognition Letters*, 19(10):929–940, 1998.

J. Wang and M. Cohen. Optimized color sampling for robust matting. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

J. Wang, J. Li, R. Gray, and G. Wiederhold. Unsupervised multiresolution segmentation for images with low depth of field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(1):85–90, 2001.

J. Wang, P. Bhat, R. Colburn, M. Agrawala, and M. Cohen. Interactive video cutout. *Proceedings of ACM SIGGRAPH Transactions on Graphics*, 24(3):585–594, 2005.

Z. Wei, T. Tan, Z. Sun, and J. Cui. Robust and fast assessment of iris image quality. *Proceedings of the 2006 international conference on Advances in Biometrics*, pages 464–471, 2006.

R. Whitaker. A level-set approach to 3d reconstruction from range data. *International Journal of Computer Vision*, 29(3):203–231, 1998.

M. Wollborn and M. Mech. Refined procedure for objective evaluation of video object generation algorithms. *In ISO/IECJTCI/SC29/WG11 M3448, 43rd MPEG Meeting*, 1998.

G. Yang and B. Nelson. Wavelet-based auto-focusing and unsupervised segmentation of microscopic images. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3:2143–2148, 2003a.

G. Yang and B. Nelson. Micromanipulation contact transition control by selective focusing and microforce control. *Proceedings of IEEE International Conference on on Robotics and Automation*, 3:3200–3206, 2003b.

T. Yeo, S. Jayasooriah, and R. Sinniah. Autofocusing for tissue microscopy. *Image and Visual Computing*, 11:629–639, 1993.

H. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of Computational Physics*, 119:179–195, 1996.