

**UNIVERSITY OF SOUTHAMPTON**  
Faculty of Engineering and Applied Science  
Department of Electronics and Computer Science

A project report submitted for the award of  
BEng Electronic Engineering

Supervisor: Dr. Steve Gunn  
Examiner: Dr. Sheng Chen

**Exploration of Methods  
for Automatic Whale Identification**

by Heike Heusel

10th May 2001



UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND APPLIED SCIENCE  
DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE

A project report submitted for the award of BEng Electronic Engineering

by Heike Heusel

This paper describes research into techniques for automating the classification and identification of humpback whales from tail fin images. The need for a system to aid in the visual identification process is becoming greater as the number of whales encountered during wildlife studies increases. Currently manual handling of large numbers of photographs, including matching the whales in each new image of whales photographed with those already cataloged is required and is time consuming and expensive. The location of the fluke in the image is a critical part of an automatic identification system. Although, the overall goal is to create a complete system useful to whale researchers, this report focuses on investigation of techniques to locate the whale fluke as the first step towards this goal. Fluke location makes use of knowledge of whale anatomy and image processing methods to build an active shape model to locate the fluke in the image. A prototype fluke location system was implemented and results are presented.



# Contents

<b>List of Symbols</b>	<b>ix</b>
<b>Acknowledgements</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem . . . . .	2
1.3 Whale Anatomy . . . . .	3
<b>2 Previous approaches to identification systems</b>	<b>7</b>
2.1 Mizroch's Whale Identification System . . . . .	7
2.1.1 Pigment pattern . . . . .	7
2.1.2 Marks . . . . .	8
2.1.3 Notch shape . . . . .	9
2.1.4 Conclusions . . . . .	9
2.2 Computer aided matching system for grey seals . . . . .	10
2.2.1 Conclusions . . . . .	10
<b>3 Potential approaches to automatic fluke location</b>	<b>11</b>
3.1 Image database . . . . .	11
3.2 Edge based techniques to locate the fluke in the image . . . . .	12
3.2.1 Snakes . . . . .	13
3.2.2 Deformable Templates . . . . .	13
3.2.3 Active Shape Models . . . . .	13
3.3 Region-based techniques to locate the fluke in the image . . . . .	14
3.3.1 Image histogram . . . . .	15
3.3.2 Global Threshholding . . . . .	15
3.3.3 Local Adaptive Thresholding . . . . .	16
<b>4 Chosen approach to automatic fluke location</b>	<b>19</b>
4.1 Implementation of active shape model . . . . .	19
4.1.1 Introduction . . . . .	19
4.1.2 Placing model points . . . . .	20
4.1.3 Sampling . . . . .	20
4.1.4 Aligning the training shapes . . . . .	21
4.2 The fluke model . . . . .	21

4.3	Model initialisation . . . . .	22
4.4	Image functionals . . . . .	23
4.4.1	Template convolution . . . . .	23
4.4.2	Image smoothing . . . . .	24
4.4.3	Edge detection . . . . .	26
4.4.3.1	First order edge detection operator . . . . .	26
4.4.3.2	Second order edge detection operator . . . . .	27
4.4.4	Bilinear Image Interpolation . . . . .	28
4.5	Chamfer Distance Transformation . . . . .	29
4.5.1	Chamfer Distance Minimisation . . . . .	30
4.6	Fluke model optimisation . . . . .	30
<b>5</b>	<b>Results</b>	<b>33</b>
5.1	Tools used . . . . .	33
5.2	The fluke model . . . . .	33
5.3	Whale fluke location . . . . .	35
<b>6</b>	<b>Conclusions and Future Work</b>	<b>41</b>
6.1	Complete System Requirements . . . . .	42
	<b>Bibliography</b>	<b>43</b>

# List of Figures

1.1	Examples of different fluke shapes. . . . .	3
1.2	One fluke blade showing orientation of fiber bundles when viewed from directly above the fluke. . . . .	4
1.3	Flexibility of single fluke as seen from behind:(top) configuration at maximum upstroke. (middle) fluke in glide position.(bottom) configuration at maximum downstroke. . . . .	4
2.1	Generic fluke patterns. . . . .	8
2.2	Fluke map. . . . .	9
3.1	A whale fluke image, the corresponding grey level distribution histogram and thresholding results . . . . .	16
4.1	104 landmark points on the boundary of the fluke with the green circles representing the major landmark points. . . . .	20
4.2	The fluke model as placed onto the image. . . . .	23
4.3	Surface plot of Gaussian function with $\sigma$ varied . . . . .	25
4.4	Applying Gaussian averaging . . . . .	26
4.5	Applying the sobel edge detection operator . . . . .	27
4.6	Applying different levels of chamfering. . . . .	30
5.1	Fluke image database used for building model . . . . .	34
5.2	Variation $-2\sqrt{\lambda_1} \leftarrow b_1 \rightarrow 2\sqrt{\lambda_1}$ . . . . .	34
5.3	Variation $-2\sqrt{\lambda_2} \leftarrow b_2 \rightarrow 2\sqrt{\lambda_2}$ . . . . .	34
5.4	Variation $-2\sqrt{\lambda_3} \leftarrow b_3 \rightarrow 2\sqrt{\lambda_3}$ . . . . .	34
5.5	Variation $-2\sqrt{\lambda_4} \leftarrow b_4 \rightarrow 2\sqrt{\lambda_4}$ . . . . .	36
5.6	Variation $-2\sqrt{\lambda_5} \leftarrow b_5 \rightarrow 2\sqrt{\lambda_5}$ . . . . .	36
5.7	Process of fluke location with applied direct averaging once, laplace operator and chamfering at level 6. . . . .	36
5.8	Smoothing performance shown on chamfer image . . . . .	37
5.9	Fluke location performance on test image. . . . .	37
5.10	Fluke image with the fluke model superimposed, showing the location after different iterations. . . . .	39
5.11	Different fluke images with the fluke model superimposed, showing the final location after different iterations. . . . .	40



# List of Symbols

$x, y$	Image co-ordinates
$I(x, y)$	Image
$M(x, y)$	Edge magnitude of image
$\psi(x, y)$	Phase angle of edge in image
$\mathbf{v}(s)$	Parametrical snake representation
$\varepsilon(x, y)$	Thresholded binary image
$\tau$	Threshold
$T$	Transpose
$N$	Number of co-ordinate pairs
$A$	Number of mutually aligned boundaries
$\bar{\mathbf{v}}$	Mean shape vector of fluke model
$\mathbf{d}$	Variation of model
$\mathbf{S}$	Covariance matrix
$\mathbf{p}$	Eigenvector
$\lambda$	Eigenvalue
$\mathbf{P}$	Matrix of eigenvectors
$\mathbf{b}$	Vector indicating variation
$t$	Number of Eigenvectors
$\mathbf{v}$	Model shape parameter vector
$t_1$	Convolution template
$t_2$	Direct averaging template
$t_3$	Gaussian template
$t_4$	Laplace template
$I_{New}$	New image after convolution
$g(x, y)$	Gaussian function
$\sigma$	Gaussian variance
$Mx, My$	Sobel templates
$\nabla^2$	Laplace operator
$a, b, c, d$	Bilinear image interpolation coefficients

$z_{00}$	Known image point in image interpolation
$C_1, C_2$	Constants for chamfer distance
$I'(x, y)$	Chamfer distance transformed image
$D_{chamfer}(I(x, y))$	Chamfer distance transformed image
$E$	Energy
$\theta_j$	jth point of model $\theta$
<b>u</b>	x,y co-ordinates of image
$E_{image}$	Image force
$E_{ActiveShapeModel}$	Energy functional of active shape model
$E_{Snake}$	Snake energy
$E_{int}$	Internal snake energy
$E_{con}$	External constraint force

## **Acknowledgements**

I am very grateful to my supervisor Dr. Steve Gunn for his support and guidance. Thanks to Dr. Mark Nixon who was a great tutor throughout my time at University. I am grateful to Dr. Howard Rosenbaum and Peter Ersts at the American Museum of Natural History in New York for supplying the whale images and valuable information. I would like to thank my family for their understanding and support. Thanks to Blake for his encouragement. Finally many thanks to all my friends without whom these years would not have been as enjoyable.



*In memory of my father*



# Chapter 1

## Introduction

### 1.1 Background

Research at the American Museum of Natural History (AMNH), which is led by Dr. Howard Rosenbaum, is aimed at expanding the understanding of humpback whales and aiding the development of improved conservation programs. To achieve these goals, it is necessary to track the movements and activities of individual whales over long periods of time. At the moment, this is done by photographing the whales at the surface, manually identifying the individual whales in the photographs, and building a database of sightings. Every year, hundreds of photographs are collected, digitally scanned, and then matched manually against a large set of whale images previously collected ([Yakutchik, 1998](#)).

Over the years, thousands of images have been collected. Manual identification of each new set of images is increasingly time consuming and subject to human error. To make the identification process more efficient, the feasibility of using a computer system to automate some or all of the classification and identification of whales from surface images is investigated. It is expected that such a system would greatly aid the AMNH Whale conservation research as well as being useful in other conservation programs.

This investigation of automatic whale identification methods is being conducted at the University of Southampton. Dr. Howard Rosenbaum and Peter Ersts at the AMNH have provided a definition of the problem as well as a sample set of whale images.

## 1.2 Problem

Individual whales are identified from sighting to sighting, forming a record of each whale's activities. Identification of individuals observed in the field has permitted detailed studies of social behaviour as described in [Shane and McSweeney \(1990\)](#). Similarly [Kaufmann et al. \(1990\)](#), [Katona and Beard \(1990\)](#) and [Perry et al. \(1990\)](#) researched the migration and residency of humpback whales. [Glockner-Ferrari and Ferrari \(1990\)](#) and [Clapham and Mayo \(1990\)](#) have produced detailed life histories of Humpback Whales in Hawaiian waters and in the Gulf of Maine. This information is tracked over years and is added to the database based on identification of individual whales in photographs taken during the field observation. This information is used to understand the whales and their conservation requirements. The key to entering new information into the database is always identification of the individual whales from surface photographs.

Whale researchers mainly use the whale fluke photographed during the whales natural movements while swimming near the surface. Observers photograph the fluke when it breaks the surface. Every humpback whale has a unique edge and pigment pattern on it's fluke and these features can be used for identification like the details of a human fingerprint. See Figure [1.1](#) .

When identification is done manually, the image is first classified depending on the pigment pattern on the fluke, as described in [Rosenbaum and Clapham \(1995\)](#) and [Burton \(1989\)](#). This classification is then used to select a set of likely candidates and a human then tries to visually match the new whale image to one of those among the selected class. This means that the human must compare each image to on average half of the candidates in the class before a match is found. It also means that if the classification is wrong, the whale will not be found in that class at all. With an automatic system it would be best to not classify too early, but rather develop a score for as many reasonable candidates as possible.

The whale researchers at AMNH would like to partially or fully automate the process of classifying and perhaps fully identifying new images to be added to the database. The goal of this project is to investigate possible techniques to automate the process of classification and identification. This research includes prototyping techniques in Matlab to isolate a whale fluke in a photograph. The results of the prototyped techniques are then evaluated for suitability for being integrated into a complete system for use by whale researchers.

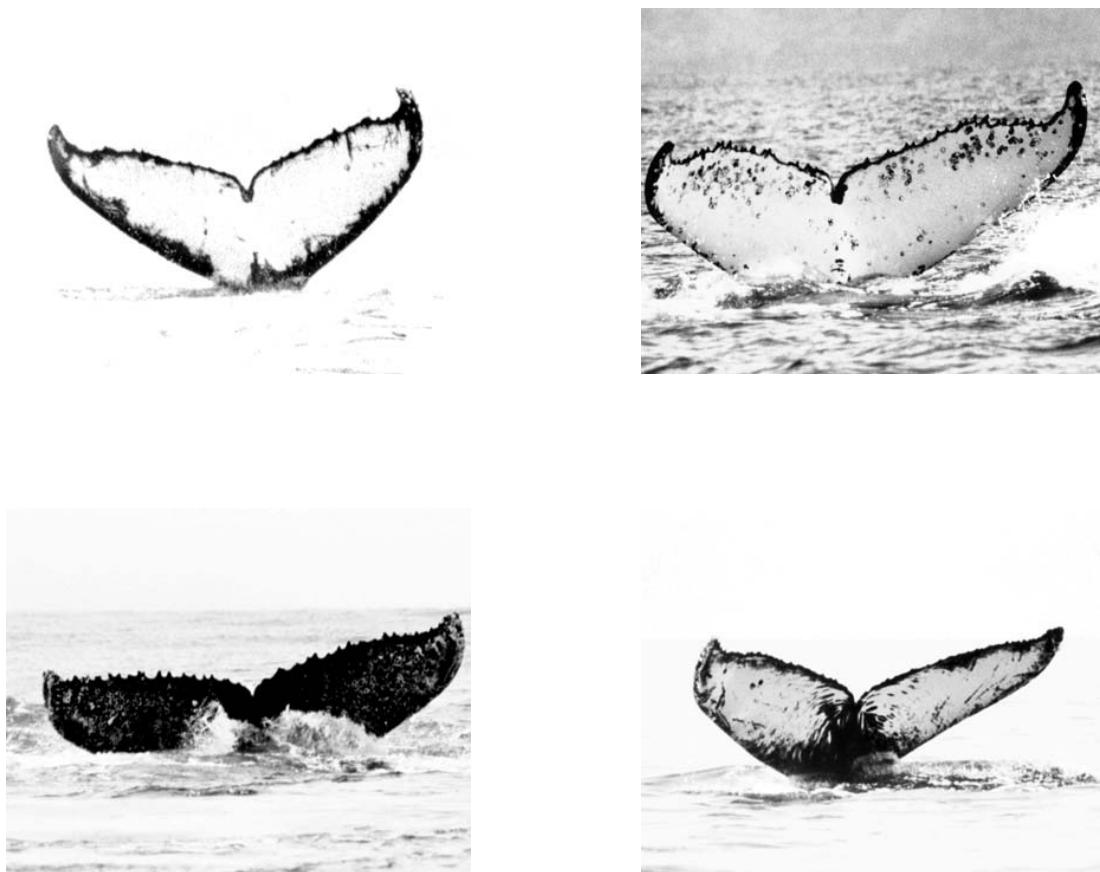


FIGURE 1.1: Examples of different fluke shapes.

### 1.3 Whale Anatomy

A basic understanding of whale anatomy is required to choose the best methods for automatically finding them in photographs with natural background. [Felts \(1966\)](#) describes the very simple structure of the fluke. It is composed of nonskeletal connective tissue. It has five basic components:

- Cutaneous layer(skin)
- Blubber layer(fat)
- Ligamentous layer
- Superficial veins
- Arterial system within core

The fibrous mass consists of horizontal, vertical and oblique bundles of collagen fibers (connective tissue) interspersed with fat cells. The oblique fibers lie in the

plane of the long axis of the fluke and make it very strong. Figure 1.2 shows the

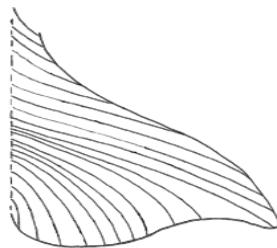


FIGURE 1.2: One fluke blade showing orientation of fiber bundles when viewed from directly above the fluke.

orientation of the fiber bundles, which is similar to the longitudinal ridges of the skin diver's foot flipper.

The core tissue extends into the free trailing edge of the fluke and becomes thinner and narrower toward the tips of the fluke. The structure remains unchanged. The ligamentous layer keeps the shape and resists dorsal and ventral bending of the organ. Variation of the shape can be expected at the tips of the fluke.

The core material is solidly attached to the caudal vertebrae (backbone) as well as the intervertebral disk and ligaments. This arrangement unites the caudal vertebral column, behind the tail-fluke hinge, into one tough element. Because of the nature of the tail fluke hinge, the fluke can only be moved up and down.

The difference in shape at downstroke and upstroke can be explained by differential power output by the dorsal and ventral tail musculature. Since most images capture the fluke at maximum upstroke, the first blade configuration in Figure 1.3 can be expected with a small degree of bending.



FIGURE 1.3: Flexibility of single fluke as seen from behind:(top) configuration at maximum upstroke. (middle) fluke in glide position.(bottom) configuration at maximum downstroke.

To summarize, the anatomy of the whale fluke shows that an individual fluke is constrained in its degrees of freedom. This is because the shape of the fluke will only change at the tips and only minimally at the upstroke. This knowledge is the key to developing a strategy to deal with the difficult inverse problem of fluke location in photographs with a naturally noisy background.



# Chapter 2

## Previous approaches to identification systems

### 2.1 Mizroch's Whale Identification System

Mizroch et al. (1990) developed a system based on manual classification of a fluke to help identify humpback whales in photographs.

The user of the Mizroch system describes the pigment pattern, notch shape and location of marks and scars by entering the most appropriate generic code for each feature manually. Other data fields used are photograph quality, filing numbers, contributor information and location. The system then uses a matching algorithm to compare the characteristics of the unknown whale to the entire database and ranks whales with similar patterns and marks together. Possible matches are displayed on the screen and evaluated by eye. This system has reduced the time necessary for identification of new whale sightings.

#### 2.1.1 Pigment pattern

The pigment pattern can be represented by one of the thirty-eight generic pattern codes, as can be seen in Figure 2.1.

The types of features are:

- Black or white on the trailing edge.

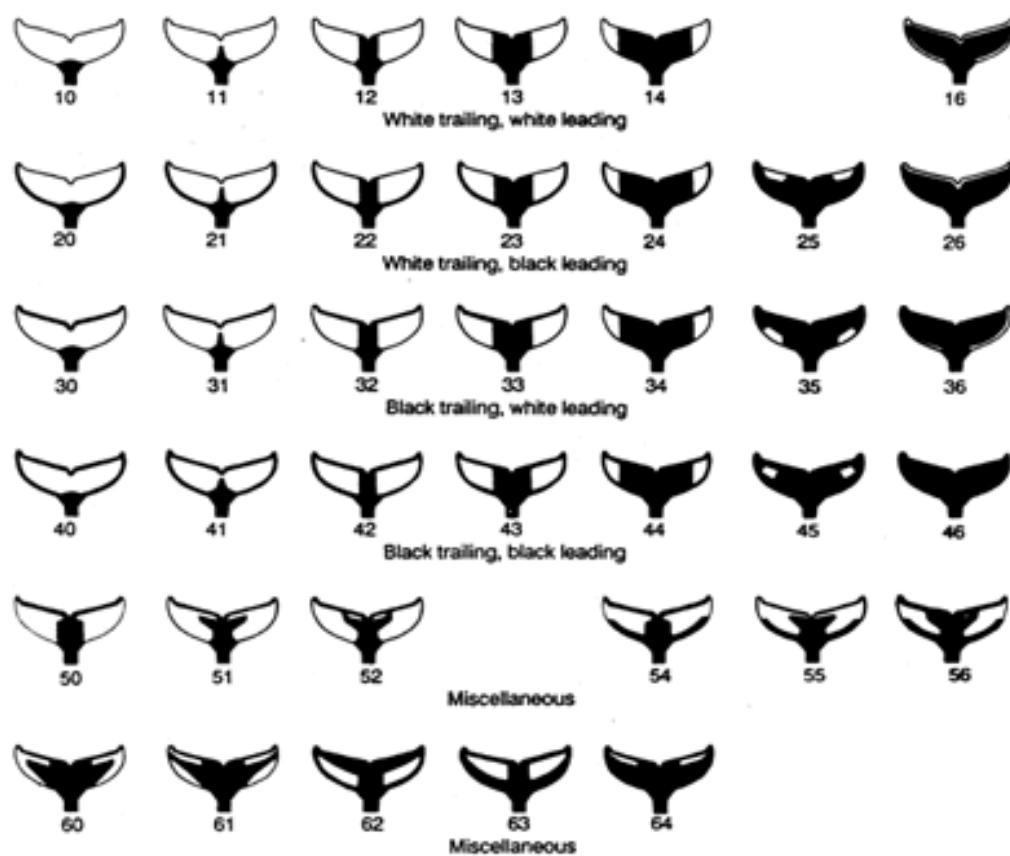


FIGURE 2.1: Generic fluke patterns.

- Characteristics of the medial line.
- Amount of black pigmentation within the fluke.

In the case of an asymmetric pattern, the fluke is coded based on the pattern on the right side of the fluke.

### 2.1.2 Marks

The 14-sector fluke map (Figure 2.2) developed by Balcomb and Katona in the late 1970s, is used to locate and code natural markings and scars. Mark codes refer to the presence of a mark, not the frequency. Uppercase letters refer to black marks, lowercase to white marks.

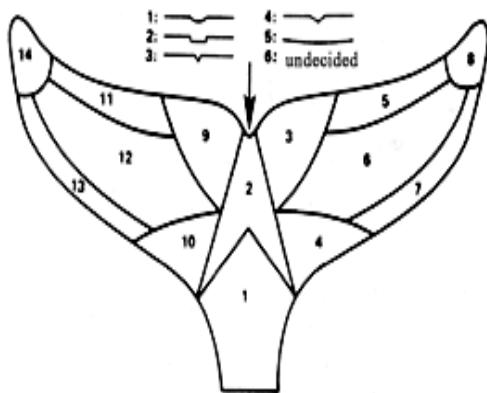


FIGURE 2.2: Fluke map.

### 2.1.3 Notch shape

The fluke notch can either be rounded, square, narrow or wide (see Figure 2.2) and the appropriate code for the shape of the notch is entered into the system.

### 2.1.4 Conclusions

The advantage of Mizroch's approach is that it has the ability to handle big photograph collections. The system has increased the speed of matching new photographs greatly. The sort criteria is very flexible and it is possible to run multiple iterations through the database in order to reduce missed matches. This technique can also have general application to a wide variety of other photo-identification studies, such as in an automatic identification system.

The disadvantage of this approach is that an experienced user is necessary to enter the generic code into the database. Also, the data entry takes about two minutes per photograph which results in a time consuming procedure when many new photographs have to be processed. The performance of the system is greatly dependent on the quality of the photographs. The results are poor when the test photograph or the photograph in the database is poorly printed or only shows partial flukes.

## **2.2 Computer aided matching system for grey seals**

Hiby and Lovell (1990) describe the use of a three-dimensional model to locate a particular region of the body surface of a grey seal. The system can be used for species where the individuals have distinctive patterns. In the case of grey seals it is the pelage pattern on the head and neck. The model describes the pattern such that the resulting description is not affected by the orientation or the posture of the animal at the time the photograph was taken. A mathematical model of the surface of the head and neck is constructed by interpolation and contouring over a set of three-dimensional coordinates scattered over head and neck. The model is fitted by using the cursor to identify three features (eyes, ears, nostrils). These features have known coordinates in the model. The system then rotates and translates the model, so that the points and features coincide. The user can then change the surface model to match the outline of the model to the outline of the animal in the photograph exactly. The operator has to check if any of the pattern cells are under water. In this case, the image must not be included in the library. The system then uses identifier arrays to compare the pattern cells. The system then scans a database for potential matches retrieving a set of candidate images which are compared with the new photograph by eye.

### **2.2.1 Conclusions**

The advantage of this technique is that the number of photographs compared by eye is reduced and therefore larger populations may be monitored. Grey seals have markings which are inconsistent in shape and location and so it is not possible to locate the pattern cell with reference to markings themselves. This system makes it possible to locate the pattern cell in relation to morphological features. The grey seals have a non-planar neck surface. The system handles the appearance of the pattern changes with viewpoint and the change in shape as the posture of the seal varies very well. The disadvantage of this approach is, that the fitting of the model to each new photograph takes a few minutes. Further, it is time consuming to compare the candidate images with the new photo by eye.

# Chapter 3

## Potential approaches to automatic fluke location

### 3.1 Image database

Automatically locating the fluke in the image will be an important part of any automatic whale identification system. As described above, although the fluke itself provides a promising basis for image processing, extracting it from the surrounding water in an image is difficult.

The whale fluke images have the following characteristics which must be dealt with in any automatic identification system:

- The images have a turbulent water background.
- The images may be photographed from any perspective.
- The angle of the fluke varies in every image, which changes with the view of the pigment pattern and the contour.
- The image can be out of focus.
- Only part of the fluke may be visible.

The fluke photographs used in this research were scanned and filed by Peter Ersts at the American Museum of Natural History. The images chosen for this project have the following characteristics:

- Grey level tif images with slightly varying dimensions of about 1500x2300 pixels and the size of 3.45MB.
- The images are in focus.
- The whole fluke is clearly visible.
- The flukes are in profile to avoid having to consider rotation and perspective initially.

## 3.2 Edge based techniques to locate the fluke in the image

Edge detection plays a very important role in image segmentation as described in ([Lei et al., 1999](#)). Edges in an image occur, where there is a discontinuity in the intensity function or a very steep intensity gradient in the image. Common types of edges are ramp and step edges. Edges can be described by direction, position and strength. The edge magnitude of the the image is given by:

$$M(x, y) = \sqrt{Mx(x, y)^2 + My(x, y)^2}, \quad (3.1)$$

where  $M$  is the length of the vector.

The edge direction is the vector's orientation and the angle is defined as:

$$\psi(x, y) = \tan^{-1} \frac{My(x, y)}{Mx(x, y)}. \quad (3.2)$$

This section describes possible edge based techniques to extract the fluke from a digitized photograph. Some of the options are:

- Flexible contour models such as snakes
- Deformable templates
- Active shape models

### 3.2.1 Snakes

Various authors have described different flexible model techniques used to locate objects in images whose shape can vary. [Kass et al. \(1988\)](#) describe flexible contour models, called snakes, which are attracted toward lines and edges. These snakes are energy-minimizing splines guided by external constraints and image forces. The external forces place the snake near the desired local minimum. The internal forces account for tension and curvature. The image forces push the snake towards lines and edges. Snakes have few constraints on their shape and may take almost any smooth boundary. They contain no knowledge of the expected shape, in this case a fluke, to be segmented. The snake can be represented parametrically by  $\mathbf{v}(s) = (x(s), y(s))$ .

The snake energy is given by

$$\begin{aligned} E_{\text{Snake}} &= \int_0^1 E_{\text{snake}}(\mathbf{v}(s))ds \\ &= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s)) + E_{\text{con}}(\mathbf{v}(s))ds, \end{aligned} \quad (3.3)$$

where  $s \in [0, 1]$ .  $E_{\text{int}}$  represents the internal energy of the spline due to bending,  $E_{\text{image}}$  stands for image force, and  $E_{\text{con}}$  gives rise to external constraint forces.

### 3.2.2 Deformable Templates

[Yuille et al. \(1992\)](#) perform feature extraction from faces using deformable templates. Features in the image are described by parameterized templates. Defined energy functions are then used to link drastic changes in the image intensity to the corresponding properties of the template. The template then alters its parameter values to minimize the energy function and deforms itself to the best fit. Yuille et al. built deformable eye and mouth templates. Deformable templates incorporate user specified parametric models to increase the prior shape knowledge, offering potential for greater robustness in the contour extraction process. However, they can suffer from problems of model mismatch if the shape model is incorrect.

### 3.2.3 Active Shape Models

[Cootes et al. \(1995\)](#) applies global shape constraints to Active Contour Models

in order to restrict the deformation of a model. They have similar advantages to deformable models, and they reduce the problem of model mismatch by learning the form of the parametric model from examples.

A Point Distribution Model represents the objects as a set of labeled points. The mean position is obtained as well as a small set of modes of variation describing how the object's shape can change. Limits to the range of the parameters are applied and hence only certain deformations are allowed to occur.

These flexible models are called Active Shape Models or Smart Snakes. At each point in the model, the changes to the overall position, orientation, and scale of the model are calculated to best fit the image evidence. [Cootes et al. \(1995\)](#) successfully created models of resistors, hands, and heart ventricles and located these objects successfully in complex, noisy and cluttered images.

Active shape models can be extended further by considering enhanced use of image information. [Lanitis et al. \(1993\)](#) used a grey level modeling technique for recognizing human faces. To fit the Point Distribution Models, a grey level model for each model point was generated. It was necessary to use the grey level modeling technique, since not all model points lay on strong edges and hence the calculation of the new position for each model point was complicated. [Cootes and Taylor \(1992\)](#) describes how to model the grey levels at each point of the shape model and how to use these grey level models for an active shape model search. The grey level environment is modeled by a mean and a number of modes of variation. When locating the best position for each model point, the position has to be found where the image best matches the grey level environment. When using active shape models to locate the fluke in the image, the grey level model might be incorporated, leading to improved reliability and accuracy.

### **3.3 Region-based techniques to locate the fluke in the image**

An image can be segmented according to its texture by examining the texture in a chosen region and then classifying it. Region based segmentation can be implemented by using the thresholding-based approach, as described by [Lei et al. \(1999\)](#) and [Milstein \(1998\)](#). This approach uses global or adaptive thresholds to segment an image and to detect whole shapes. These shapes tend to have distinctive properties as opposed to the background they are aligned against.

### 3.3.1 Image histogram

The grey level distribution histogram plots the number of pixels with a particular grey level against the grey level. Figure 3.1(a) shows an image of a fluke and the corresponding grey level histogram can be observed in Figure 3.1(b). The histogram shows that all grey levels are used, but the majority of the pixels have a grey level of or near 255, which stands for white. The whiter points relate to the parts of the fluke surface, the sea and the sky. The region between grey level zero and 50 contains the dark parts of the image, such as edges of the fluke and some areas of the sea. The histogram can be used to understand and improve the quality and the appearance of images.

### 3.3.2 Global Thresholding

Global thresholding divides the pixels in the image into two dominant groups according to their grey-level. The appropriate threshold  $\tau$  has to be chosen manually to distinguish between the two dominant levels. The original image  $I(x, y)$  is scanned pixel by pixel and each pixel is tested against the selected threshold. In the case where  $I(x, y) > \tau$ , the pixel is classified as a background pixel, otherwise the pixel is classified as an object pixel. The thresholded binary image  $\varepsilon(x, y)$  can then be described by:

$$\varepsilon(x, y) = \begin{cases} 255 & \text{if } I(x, y) > \tau \\ 0 & \text{if } I(x, y) \leq \tau \end{cases} \quad (3.4)$$

Global thresholding works well if the grey-level distribution histogram contains distinctively separated peaks corresponding to the objects and background, which is the case in fluke images. The chosen threshold according to the histogram in Figure 3.1(b) is  $\tau = 140$  and the result is shown in Figure 3.1(c). Since the fluke edge has a low grey level, this approach works well, even though the image has a turbulent background. The difficulty with this approach is that it is not possible to use a standard threshold  $\tau$  for all images and it is tedious to manually choose these thresholds.

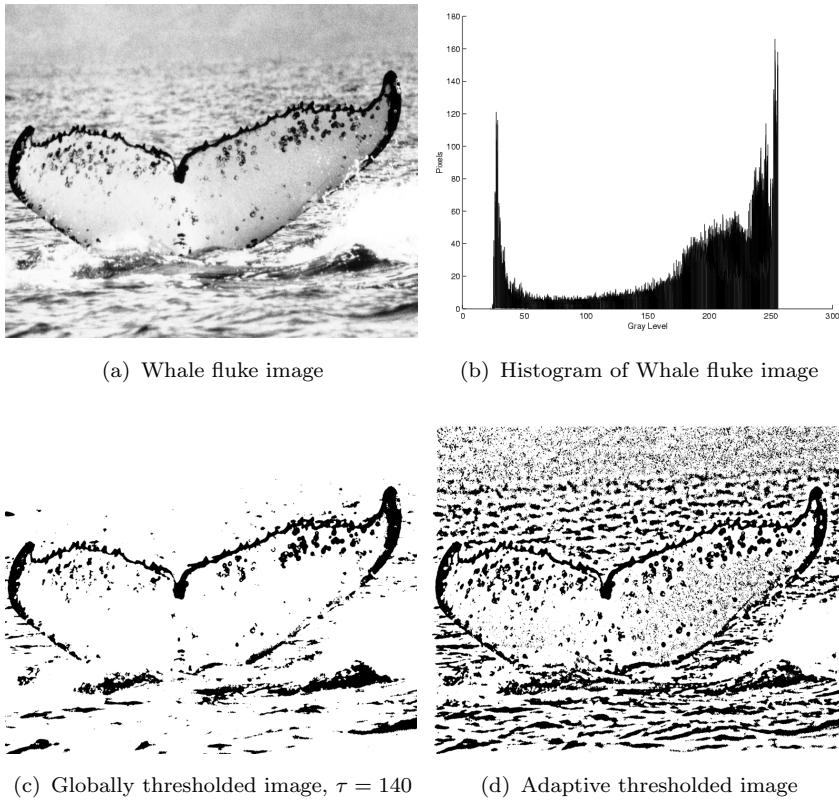


FIGURE 3.1: A whale fluke image, the corresponding grey level distribution histogram and thresholding results

### 3.3.3 Local Adaptive Thresholding

Local adaptive thresholding can be used for images whose histograms do not contain distinctive peaks. This method selects an individual threshold for each pixel based on the range of intensity values in its local neighbourhood. The intensity values of the local neighbourhood of each pixel can be statistically examined by the average, the median, or the average between the minimal and the maximal grey level in the neighbourhood. The neighbourhood size has to be chosen accordingly. Milstein (1998) states, that the result becomes poorer with increasing the neighbourhood size, since it is then more influenced by the illumination gradient. If the environment chosen is too small, it may lead to poor results since pixels in a noisy environment will be extremely influenced by the noise.

Figure 3.1(d) shows the result when local adaptive thresholding with a neighbourhood size of 61x61 is attempted. The result shows, that local adaptive thresholding is not really suitable in this case. The adaptive thresholding method leads to a noisy background. When pixels are centered in a background environment only the range of grey levels within this environment is very small. An unsuitable

threshold can be the result. This problem can be avoided by combining global and adaptive thresholding, where the threshold of a pixel as the average of its environment minus a predefined fixed threshold is computed.

In conclusion, the global thresholding yields good results, since the fluke histograms have distinctive peaks. The method of segmentation by thresholding should be investigated further and included into the system of automatic fluke location.



# Chapter 4

## Chosen approach to automatic fluke location

### 4.1 Implementation of active shape model

#### 4.1.1 Introduction

The active shape model technique was chosen for the prototyping work on this project. It is a powerful shape description technique. It can represent the typical shapes and variability of the fluke in the training set. Automatically locating the fluke in the image is an ill posed problem and so any domain information available should be captured to help the process. Hence a Point Distribution Model was implemented.

To build a Point Distribution Model, points were placed around the boundary as described by [Cootes et al. \(1995\)](#). [Bookstein \(1991\)](#) labeled significant points by describing their meaning. Hence he called them landmark points. In the case of whale identification, the fluke was divided into four parts and the first and last point of each part labeled as major landmark points (see Figure 4.1). These points mark easily identifiable features of the fluke and have an application-dependent significance.

### 4.1.2 Placing model points

To help build the model, a prototype system was constructed in Matlab in which the user can enter as many points as necessary along the boundary for each part of the fluke by using the left mouse button. The right mouse button may be used to specify the endpoint of each part. The five major landmark points can be observed in Figure 4.1, where they are represented by green, round circles. Placing the points manually is very time consuming. Note however that the points are only placed manually during the training phase when building a point distribution model and would not be required later. For capturing the shape variability reliably, it is important that the labeling is done correctly along the boundary on each training shape.

### 4.1.3 Sampling

To ensure that every image is described by the same number of points, the system can equally space the rest of the points along the connecting boundaries. The length of each boundary part is calculated and divided by the number of points to sample. The location for each point is calculated. When entering the points and when sampling, a line is drawn around the fluke and the new points are represented by red x's for easy observation as shown in Figure 4.1. Due to the division of the fluke into parts, major landmark points remain unchanged.



FIGURE 4.1: 104 landmark points on the boundary of the fluke with the green circles representing the major landmark points.

An early approach required entry of six points per part, but then the system was

improved to allow the user to enter as many points as necessary in order to model distinctive shapes without loosing important information. The system allows for the number of sampling points to be adjusted. Each structure is now represented by 104 points, which seems to describe the fluke with sufficient accuracy.

#### 4.1.4 Aligning the training shapes

[Cootes et al. \(1995\)](#) compare equivalent points from different shapes by scaling, rotating and translating the training shapes so that they correspond as closely as possible. As a computational compromise for aligning the training shapes, the fluke notch was set to zero in all training shapes as a new origin. Therefore all training shapes are translated to the same origin, without computing scale and rotation. The scale and rotation could be included later on if required.

## 4.2 The fluke model

As described in [Cootes et al. \(1995\)](#), the outcome of the alignment will be  $A$  (mutually aligned) boundaries  $\mathbf{v}^1, \mathbf{v}^2, \dots, \mathbf{v}^A$ . Each shape is given by  $N$  co-ordinate pairs

$$\mathbf{v}^i = (x_1^i, y_1^i, \dots, x_N^i, y_N^i)^T. \quad (4.1)$$

The mean shape is given by

$$\bar{\mathbf{v}} = (\bar{x}_1, \bar{y}_1, \bar{x}_2, \bar{y}_2, \dots, \bar{x}_N, \bar{y}_N)^T. \quad (4.2)$$

Where

$$\bar{x}_j = \frac{1}{A} \sum_{i=1}^A x_j^i, \quad (4.3)$$

and

$$\bar{y}_j = \frac{1}{A} \sum_{i=1}^A y_j^i. \quad (4.4)$$

The variation may be explicitly measured by

$$\mathbf{d}^i = \mathbf{v}^i - \bar{\mathbf{v}}. \quad (4.5)$$

This calculation is done for each training vector. Then the  $2N \times 2N$  covariance

matrix  $\mathbf{S}$  can be calculated as

$$\mathbf{S} = \frac{1}{A} \sum_{i=1}^A \mathbf{d}^i \mathbf{d}^{iT}. \quad (4.6)$$

Solving the equation

$$\mathbf{S}\mathbf{p}_i = \lambda_i \mathbf{p}_i \quad (4.7)$$

provides the eigenvectors  $\mathbf{p}_i$  and eigenvalues  $\lambda_i$  of  $\mathbf{S}$ .

The eigenvalues have to be sorted with the largest eigenvalue first, assuming  $\lambda_i \geq \lambda_{i+1}$ . The eigenvectors with large eigenvalues provide the important modes of variation. Finding the highest eigenvalues gives the indication where the largest variation in the model will occur. A matrix  $\mathbf{P}$  for the first  $t$  eigenvectors must be built

$$\mathbf{P}_t = \mathbf{p}^1 \mathbf{p}^2 \mathbf{p}^3 \dots \mathbf{p}^t. \quad (4.8)$$

For any vector  $\mathbf{v}$ , a vector  $\mathbf{b}$  exists such that

$$\mathbf{v} = \bar{\mathbf{v}} + \mathbf{P}_t \mathbf{b}_t, \quad (4.9)$$

where  $\mathbf{b}_t$  indicates how much variation is exhibited with respect to the eigenvectors

$$\mathbf{b}_t = (b_1, b_2, \dots, b_t)^T. \quad (4.10)$$

The parameter  $\mathbf{b}_i$  may be varied to create similar shapes to those in the training set. The suitable limits for variation can be expected to lie in the range,

$$-3\sqrt{\lambda_i} \leq \mathbf{b}_i \leq 3\sqrt{\lambda_i}, \quad (4.11)$$

since most of the population is within three standard deviations of the mean. But using the factor of 3 results in extreme shapes. Hence a smaller factor of 2 is used.

$$-2\sqrt{\lambda_i} \leq \mathbf{b}_i \leq 2\sqrt{\lambda_i}, \quad (4.12)$$

### 4.3 Model initialisation

The end user of the system defines the location of the fluke notch in the image by marking the fluke notch with the left mouse button. The system places the notch of the model onto the notch of the fluke in the image as shown in Figure 4.2.

This method gives a good initial position for the model to move towards the outer



FIGURE 4.2: The fluke model as placed onto the image.

boundary/fluke edge, instead of locking onto marks on the fluke or onto the inner boundary of the leading edge.

## 4.4 Image functionals

The image functionals are used to improve the performance of the system by producing minima at the edges of the fluke. Several different image-preprocessing techniques may be applied to suppress irrelevant information and to enhance important features. The fluke location can be tested with and without preprocessing to see which image preprocessing technique is the most suitable. The methods of template convolution, image smoothing, edge detection and bilinear image interpolation are detailed in this section.

### 4.4.1 Template convolution

A template can be used to calculate a new image from an original one when smoothing an image or performing edge detection (Nixon and Aguado, 2001). The template is a set of weighting functions with a 3x3 template having the size of three pixels wide and three pixels long. The template is usually chosen to be square and odd to ensure that it can be positioned correctly. The template is

passed over the image and convolution calculates new pixel values from a pixel's neighbourhood. The 3x3 template in Equation 4.13 is given by nine weighting coefficients  $w_e$ .

$$t_1 = \begin{bmatrix} w_0 & w_1 & w_2 \\ w_3 & w_4 & w_5 \\ w_6 & w_7 & w_8 \end{bmatrix} \quad (4.13)$$

The template is centered on the pixel to evaluate. A new value for this pixel is calculated, by multiplying the pixel values with the corresponding weighting coefficient and adding it to an overall sum. The resulting value becomes the pixel in a new image. This operation occurs for every single pixel in the image by moving the template horizontally until the end of the line and then by positioning it at the beginning of the next line. A new Image  $I_{New}$  with the coordinates  $x, y$  can be calculated by convoluting the template  $t_1$  in Equation 4.13 with the original image  $I$  according to:

$$\begin{aligned} I_{New(x,y)} = & w_0 \times I_{x-1,y-1} + w_1 \times I_{x,y-1} + w_2 \times I_{x+1,y-1} + \\ & w_3 \times I_{x-1,y} + w_4 \times I_{x,y} + w_5 \times I_{x+1,y} + \\ & w_6 \times I_{x-1,y+1} + w_7 \times I_{x,y+1} + w_8 \times I_{x+1,y+1}, \end{aligned} \quad (4.14)$$

where  $x, y \in [2]$ . When placing the template at the border, parts of the template fall outside the image hence it is not possible to calculate a new pixel value. Since the border, which is half the size of the template, does in this case not contain important information, it was decided to set the border pixels to black.

#### 4.4.2 Image smoothing

As described in Sonka et al. (1999) and Nixon and Aguado (2001), image smoothing may be used to reduce noise in the image. The disadvantage is that smoothing also blurs all sharp edges which may contain important information about the fluke edge. The goal is to smooth the image, but to preserve the edges.

One image smoothing technique is the direct averaging operator, where templates are used to reduce noise. When using a 3x3 convolution template  $t_2$ ,

$$t_2 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4.15)$$

the noise is reduced and the image is slightly blurred. The template convolution method is a low pass filter since it suppresses high frequencies (noise) and allows low frequencies to be retained. But higher frequencies also exist at sharp feature edges, so removing these high frequencies causes blurring. A larger 7x7 template removes more noise, but reduces the detail in the image and the blurring is greater. Since larger templates also impose high computational cost, the 3x3 template  $t_2$  in Equation 4.15 is suitable for smoothing the whale fluke images.

Another possible method for smoothing the image is the gaussian averaging operator (Nixon and Aguado, 2001). The coefficients for the gaussian template can be calculated according to:

$$g(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (4.16)$$

where the function  $g$  at coordinates  $x, y$  is controlled by the variance  $\sigma^2$ . In order for the template's coefficients near the template's edge to drop to near zero, the variance is chosen accordingly. If setting the variance to  $\sigma = 1.0$  and choosing a template size of 5x5, then the corresponding template is given by:

$$t_3 = \begin{bmatrix} 0.02 & 0.08 & 0.14 & 0.08 & 0.02 \\ 0.08 & 0.37 & 0.61 & 0.37 & 0.08 \\ 0.14 & 0.61 & 1.0 & 0.61 & 0.14 \\ 0.08 & 0.37 & 0.61 & 0.37 & 0.08 \\ 0.02 & 0.08 & 0.14 & 0.08 & 0.02 \end{bmatrix} \quad (4.17)$$

The template  $t_3$  is then normalized by the sum of the template values and convolved with the image  $I(x, y)$ . The surface plot corresponding to  $\sigma = 1.0$  is shown in Figure 4.3(a). Comparing Figure 4.3(a) with Figure 4.3(b) where  $\sigma = 0.5$ , it

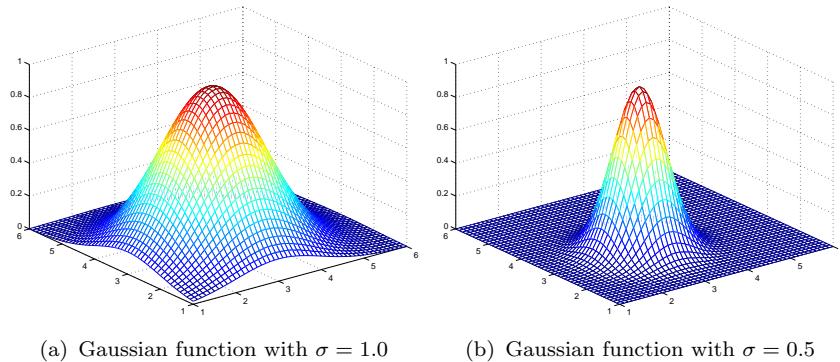


FIGURE 4.3: Surface plot of Gaussian function with  $\sigma$  varied

can be observed, that  $\sigma$  controls the width of the bell shaped surface plot, hence changing the performance of the edge smoothing method. A larger  $\sigma$  leads to

greater smoothing of both noise and image sharpness. As a result, if  $\sigma$  is too small, it will provide insufficient noise filtration while if  $\sigma$  is too large, it will smooth out distinct transitions. This creates a problem of finding the optimal  $\sigma$  for a given image. The advantage of this approach is, that compared to direct averaging, the noise is removed without removing important features. The effect of smoothing an image with an  $5 \times 5$  gaussian operator ( $\sigma = 1.0$ ) is illustrated in Figure 4.4

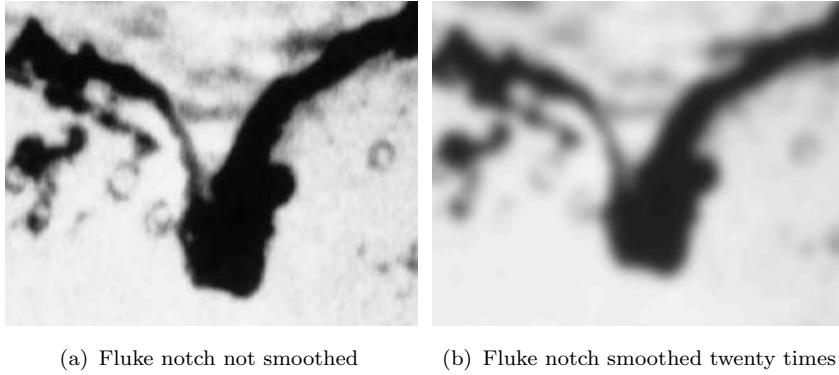


FIGURE 4.4: Applying Gaussian averaging

### 4.4.3 Edge detection

The two major edge detection algorithms can be categorized as:

- First order edge detection operator.
- Second order edge detection operator.

Examples of these edge detection operators are described below ([Nixon and Aguado, 2001](#)).

#### 4.4.3.1 First order edge detection operator

First order edge detection operators detect the difference in intensity in an image. Here, first order differentiation is used to find the edge position. Where the rate of change of the original signal is greatest, a peak can be observed. When differencing horizontally adjacent points, vertical edges can be detected. Whereas a vertical operator detects horizontal edges.

A common operator with good performance is the Sobel edge detection operator. The Sobel templates are given by

$$Mx = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (4.18)$$

and

$$My = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (4.19)$$

The image  $I(x, y)$  is convolved with  $Mx$  and with  $My$  separately. Then the edge magnitude can be calculated by using Equation 3.1. Areas with constant brightness reflect in low edge magnitude. The result of applying the Sobel operator can be seen in Figure 4.5. The advantage of this operator is, that it has smoothing

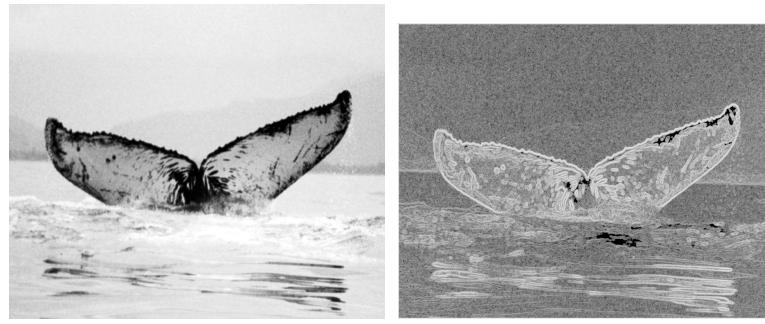


FIGURE 4.5: Applying the sobel edge detection operator

properties included, which saves computational costs.

#### 4.4.3.2 Second order edge detection operator

One example of a second order edge detection operator is the Laplace operator. This operator is used to locate changes in the intensity function. The linear differential Laplace operator  $\nabla^2$  approximates the second derivative which gives the gradient magnitude. This is based on the observation, that the maximum of the first order derivative occurs at a zero crossing of the second derivative.

The 3x3 template  $t_4$  is defined as

$$t_4 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4.20)$$

and can be used for the edge detection of the fluke images. This operator is invariant to rotation in the image. The advantage of the Laplacian operator is, that it imposes low computational cost. The disadvantage is, that it does not contain smoothing and that it responds to noise more than the first order operator since the differentiation is of higher order.

#### 4.4.4 Bilinear Image Interpolation

The bilinear image interpolation technique is used to eliminate discontinuities at pixel boundaries since this imposes difficulties when using the gradient descent method as described by [Gunn \(1996\)](#). This method fits a bilinear surface through existing data points. The value of an interpolated point is a combination of the values of the four closest points. Information from neighboring pixels is used to define pixel intensities in between pixel address boundaries. A minimal surface provides a continuous function for a discrete image.

The equation of the surface is given by

$$I(x, y) = a + bx + cy + dxy, \quad (4.21)$$

where the known image points are

$$\begin{aligned} z_{00} &= I(0, 0), \\ z_{01} &= I(0, 1), \\ z_{10} &= I(1, 0), \\ z_{11} &= I(1, 1), \end{aligned} \quad (4.22)$$

and the coefficients  $a, b, c, d$  are given by

$$\begin{aligned} a &= z_{00}, \\ b &= z_{10} - z_{00}, \\ c &= z_{01} - z_{00}, \\ c &= z_{11} - z_{10} - z_{01} + z_{00}. \end{aligned} \quad (4.23)$$

## 4.5 Chamfer Distance Transformation

The chamfer distance transformation was first proposed in 1977 by Barrow et al. (1977), and then further developed by Borgefors (1988). Shi (2001) used the chamfer distance transform method for recognizing handwritten Chinese characters and showed very good results. The most important property of this method is, that it is insensitive to noise and other disturbances. Global distances in the image are approximated by propagating local distances. Sequential distance transformation is known as chamfer distance. In the binary image, each edge pixel is set to zero and each non-edge pixel is set to infinity. A 3x3 pixel window is used to scan through the image forwards (from left to right and top to bottom) and backwards (from right to left and bottom to top). For each pixel valued  $I(x, y)$  a new value can be calculated as follows:

$$\text{Forwards: } I(x, y) = \min \begin{cases} I(x - 1, y - 1) + C2 \\ I(x - 1, y) + C1 \\ I(x - 1, y + 1) + C2 \\ I(x, y - 1) + C1 \\ I(x, y) \end{cases} \quad (4.24)$$

$$\text{Backwards: } I(x, y) = \min \begin{cases} I(x, y) \\ I(x, y + 1) + C1 \\ I(x + 1, y - 1) + C2 \\ I(x + 1, y) + C1 \\ I(x + 1, y + 1) + C2 \end{cases} \quad (4.25)$$

Borgefors (1988) assigned the values 3 and 4 to the constants  $C1$  and  $C2$ . The iterations are carried out until no value changes.

Applying the Chamfer distance method, a blurred image is created, which can tolerate noise and distorted data. The level of chamfering can be adjusted and the results can be seen in Figure 4.6. Here, the image has been smoothed and the Sobel edge detection operator applied, before the Chamfer distance method was carried out. A satisfactory basin of attraction is created at a level of around 4, which is needed to lead the Active Shape Model to the edge of the fluke.

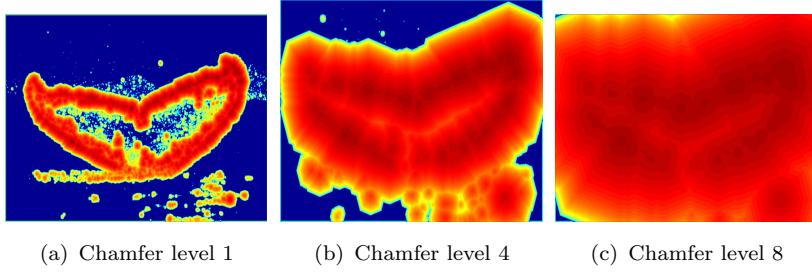


FIGURE 4.6: Applying different levels of chamfering.

#### 4.5.1 Chamfer Distance Minimisation

The chamfer distance transformed image of  $I$  is  $I'$  and can be expressed as

$$I'(x, y) = D_{chamfer}(I(x, y)) \quad (4.26)$$

Therefore the energy of a model  $\theta$  is given by:

$$E = \sum_{j=1}^N D_{chamfer}(I(\theta_j(\mathbf{b}))), \quad (4.27)$$

where  $\theta_j$  is the  $j$ th point of  $\theta$ , which is located in a 2-dimensional position  $(x_j, y_j)$ .

We seek to choose  $\mathbf{b}$  such that

$$\frac{dE}{d\mathbf{b}} = \sum_{j=1}^N \frac{d}{d\mathbf{b}} D_{chamfer}(I(\theta_j(\mathbf{b}))) = 0 \quad (4.28)$$

The energy functional in Equation 4.28 can be minimized by using a gradient descent based approach. Since the initial position of the model  $\theta$  is given, the solution of Equation 4.28 can be obtained by adjusting the model variation coefficients  $\mathbf{b}$ , using the gradient information. This theory is applied below when optimizing the fluke model.

## 4.6 Fluke model optimisation

After placing the model, Snakes may be used to find the edge of the fluke as stated in Cootes et al. (1995). The edge energy attracts the snake to contours with large image gradients and therefore locates strong edges.

The image is defined by  $I = I(x, y) = I(\mathbf{u})$ , where  $x$  and  $y$  are  $x, y$  co-ordinates of

the image.

A simple energy functional can be used to find the edges in the image. If we set

$$E_{image}(\mathbf{v}) = -|\nabla I(\mathbf{v})|^2, \quad (4.29)$$

then the snake is attracted to contours with large image gradients.

The external force of the snake, usually imposed by the user, is calculated with  $\mathbf{b}$ , which indicates how much variation is allowed with respect to each of the eigenvectors. The internal spline energy specifies the elasticity and stiffness of the snake and is not required in this case since this information has already been included with building the Point Distribution Model.  $E_{con}$  gives rise to external constraint forces. The external constraint forces place the snake near the desired local minimum. These forces can come from an automatic attentional mechanism or user interface. It can be defined as

$$E_{con}(\mathbf{v}) = \|\mathbf{b}\|^2, \quad (4.30)$$

but for simplicity  $E_{con}$  has not been included in the calculations here. Hence for a set of active shape model points  $\mathbf{v}$ , the energy functional for each active shape model point is given by:

$$\begin{aligned} E_{ActiveShapeModel}(\mathbf{v}) &= E_{image}(\mathbf{v}) \\ &= E_{image}(\bar{\mathbf{v}} + \mathbf{P}_t \mathbf{b}_t). \end{aligned} \quad (4.31)$$

Therefore the energy functional of the active shape model to be minimized is given by:

$$\begin{aligned} E_{ActiveShapeModel}(\mathbf{v}) &= \sum_{i=1}^N I(\mathbf{v}_i) \\ &= \sum_{i=1}^N I(x_i, y_i) \\ &= \sum_{i=1}^N I(\mathbf{v}_{x_i} + \mathbf{P}(i-1, :) \mathbf{b}, \mathbf{v}_{y_i} + \mathbf{P}(i, :) \mathbf{b}), \end{aligned} \quad (4.32)$$

where the active shape model is represented by  $N$  coordinate pairs and where  $P(i, :)$  is a Matlab expression standing for row  $i$  and the whole column of the eigenvector matrix  $P$ .

The derivative of the energy functional can be defined by:

$$\begin{aligned}
 \frac{dE_{ActiveShapeModel}}{d\mathbf{b}} &= \sum_{i=1}^N \frac{dI}{d\mathbf{b}}(x_i, y_i) \\
 &= \sum_{i=1}^N \frac{dI}{d\mathbf{b}}(\mathbf{v}_{x_i} + \mathbf{P}(i-1,:) \mathbf{b}, \mathbf{v}_{y_i} + \mathbf{P}(i,:) \mathbf{b}) \\
 &= \sum_{i=1}^N \left( \frac{dI}{dx_i} \frac{dx_i}{d\mathbf{b}} + \frac{dI}{dy_i} \frac{dy_i}{d\mathbf{b}} \right) \\
 &= \frac{d\mathbf{x}}{d\mathbf{b}} \frac{dI}{d\mathbf{x}} + \frac{d\mathbf{y}}{d\mathbf{b}} \frac{dI}{d\mathbf{y}}
 \end{aligned} \tag{4.33}$$

To find the local energy minimum, a gradient descent based approach is used. We seek to choose  $b$  such that

$$\frac{dE_{ActiveShapeModel}}{d\mathbf{b}} = \sum_{i=1}^N \frac{dI}{d\mathbf{b}}(\mathbf{v}_{x_i} + \mathbf{P}(i-1,:) \mathbf{b}, \mathbf{v}_{y_i} + \mathbf{P}(i,:) \mathbf{b}) = 0 \tag{4.34}$$

Adjusting the model variation coefficients  $\mathbf{b}$  is done by using a function which uses a scaled conjugate gradient algorithm to find a local minimum of the function 4.32. The bilinear image interpolation method as described above is used here to eliminate discontinuities at pixel boundaries. In order for the model to lay within the modes of variation, the minimizing function has been modified and prevents exceeding the allowed modes of variation.

# Chapter 5

## Results

### 5.1 Tools used

The experiments were conducted on a PC (Pentium III 700MHZ) using Matlab. Matlab was chosen for the implementation of this prototype system, because the project was time limited and it was more efficient to write the code in Matlab than in the C or C++ programming languages. The disadvantage of using Matlab is slower execution. Since Matlab is a interpreted language, it needs to be interpreted when the program is run. A compiled language such as C or C++ is compiled and optimized once generating more efficient code. Converting it into C or C++ would greatly improve the speed of the system.

### 5.2 The fluke model

The point distribution model technique was chosen to implement the fluke model because the model can only deform in ways characteristic for the object it represents. Twenty fluke images, as shown in Figure 5.1 , were used for constructing a fluke model. This model represents the fluke as a set of labeled points, giving the mean position and a small set of modes of variation which describe how the fluke's shape can change. It was found that the variance could be explained by the first seven modes of variation. The figures (5.2, 5.3, 5.4, 5.5, 5.6) show the modes of

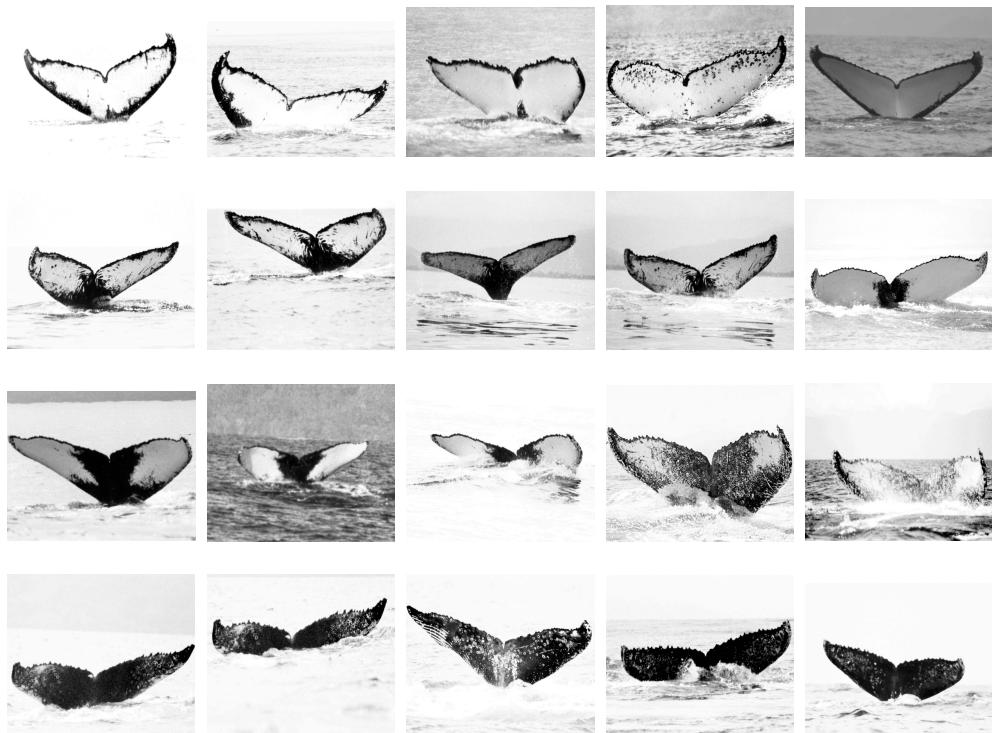


FIGURE 5.1: Fluke image database used for building model

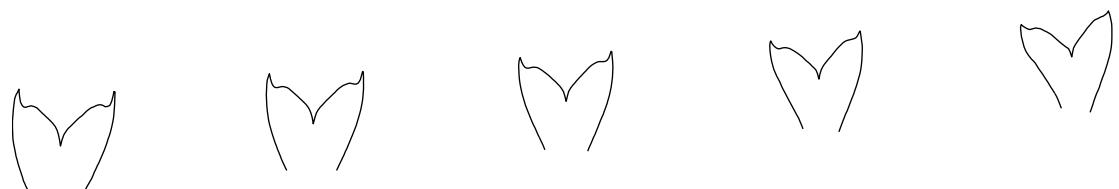


FIGURE 5.2: Variation  $-2\sqrt{\lambda_1} \leftarrow b_1 \rightarrow 2\sqrt{\lambda_1}$



FIGURE 5.3: Variation  $-2\sqrt{\lambda_2} \leftarrow b_2 \rightarrow 2\sqrt{\lambda_2}$



FIGURE 5.4: Variation  $-2\sqrt{\lambda_3} \leftarrow b_3 \rightarrow 2\sqrt{\lambda_3}$

variation obtained by varying the first five model parameters by using

$$-2\sqrt{\lambda_i} \leq \mathbf{b}_i \leq 2\sqrt{\lambda_i} \quad (5.1)$$

individually. The first parameter varies the height and width of the shape. The second and third parameters vary the width of the fluke and the shape and length of the fluke tips. The fourth parameter varies the overall size of the fluke. The fifth parameter varies the shape and length of the left and right side of the fluke. The mean model can be seen in the center of each figure, where the mode of variation is zero. Observing the different variations of the model and comparing them with the images in Figure 5.1, the shapes variations appear to be correct, hence the result is very satisfactory. The choice of using 104 points for representing the fluke boundary is a compromise between accuracy and processing time. It proved to be a good choice providing a reasonable approximation of the fluke boundary.

This compact parameterized fluke model provides a good basis for a successful location of whale flukes in images.

### 5.3 Whale fluke location

A whale fluke location process has been implemented and the performance and results are described below. Using the direct averaging operator (Equation 4.15) together with the second order laplace operator ( 4.20) for edge detection showed reasonable results when locating the fluke. The image (see Figure 5.7(b)) has not been squared after second order edge detection. Hence the chamfered image does not show an even chamfering flow, but many small local minima as can be observed in Figure 5.7(c). The final location of the fluke after 47 iterations is shown in Figure 5.7(d) .

When squaring the edge detected image, the chamfer method failed to work. Hence, the gaussian smoothing operator together with the sobel edge detection was tested. Since the sobel operator has smoothing properties, additional smoothing was not necessary. In any event, the performance of adding the gaussian operator with  $\sigma = 1$  was tested. Comparing the performance (see Figure 5.8) when smoothing is omitting versus smoothing the image with the gaussian operator where  $\sigma = 1$  shows that smoothing with the gaussian operator leads to a good chamfer image, because more noise is suppressed. Hence the images were smoothed once with  $\sigma = 1$  before using the sobel edge detection operator. In con-

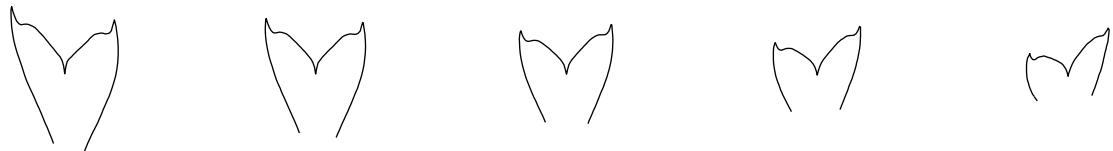
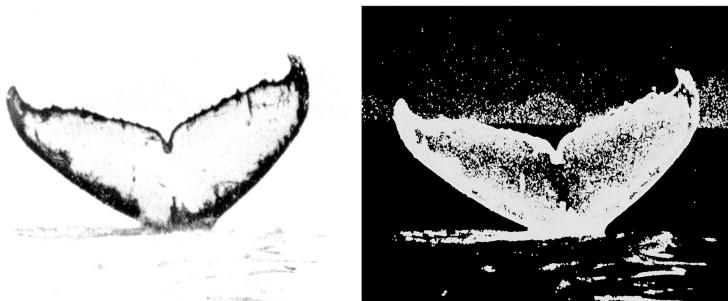


FIGURE 5.5: Variation $-2\sqrt{\lambda_4} \leftarrow b_4 \rightarrow 2\sqrt{\lambda_4}$

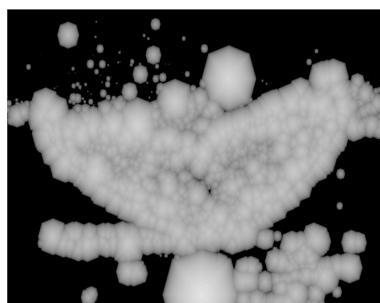


FIGURE 5.6: Variation $-2\sqrt{\lambda_5} \leftarrow b_5 \rightarrow 2\sqrt{\lambda_5}$



(a) Original image.

(b) After smoothing and laplace edge detection.

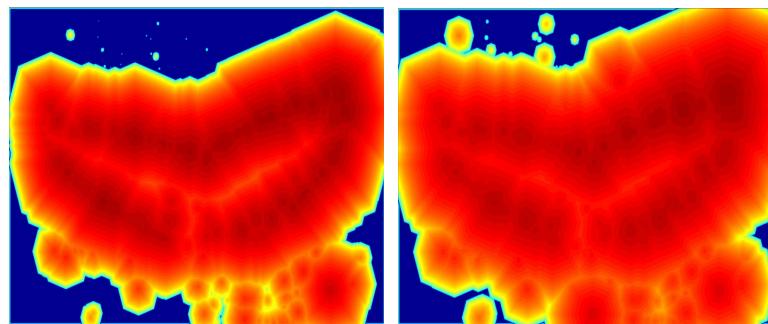


(c) Chamfer distance transformed image.



(d) Fluke located.

FIGURE 5.7: Process of fluke location with applied direct averaging once, laplace operator and chamfering at level 6.

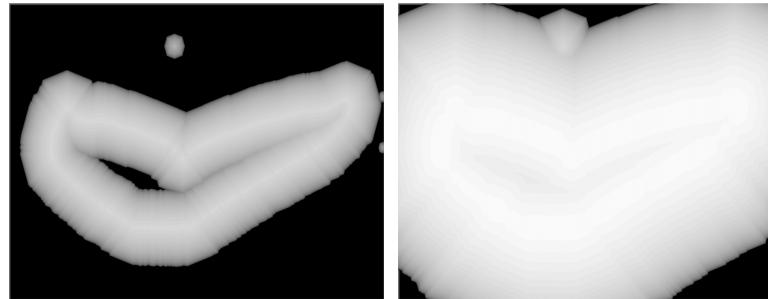


(a) Image smoothed once with template  
 (b) Image not smoothed

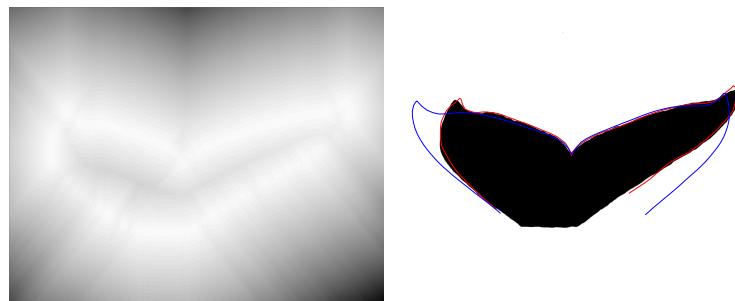
FIGURE 5.8: Smoothing performance shown on chamfer image

clusion, this method of preprocessing resulted in very satisfactory chamfer distance transformed images, creating a good basin of attraction.

A test image was created which shows a black fluke on a white background (see Figure 5.9(d)). Figure 5.9 shows the testing of the system using the test



(a) Chamfer distance image with level 1.  
 (b) Chamfer distance image with level 3



(c) Chamfer distance image with level 6  
 (d) Fluke location using test image at  
 chamfer level 6, mean model showing.

FIGURE 5.9: Fluke location performance on test image.

image. The effect on changing the chamfer level can be observed in the Figures 5.9(a) 5.9(b) 5.9(c). In Figure 5.9(d) the initial position of the mean fluke

model is shown. For the fluke location, level 6 was used. The fluke model located the fluke edge, where the energy is the lowest, accurately. It can be observed however, that the model does not reach the bottom of the right half of the fluke. This is because the lower boundary of the model has not been closed when labeling the model, since there is no edge information in that area. It might be possible to solve this problem of locking on to the whole fluke, by closing the boundary when labeling the model.

The movement of the model towards the fluke edge of a whale image with each iteration is illustrated in Figure 5.10 . The image has been smoothed once and the chamfer level has been set to 6 before using the gradient descent method. The system used 43 iterations, but the eye cannot differentiate after about 20 iterations. The minimizing function could be modified to iterate only until it reaches a point of diminishing returns. Hence, iterations up to 20 are displayed. The result is satisfactory, since the model was able to find the fluke edge.

Further fluke location testing has been conducted on different whale fluke images as displayed in Figure 5.11 . The system worked well for different fluke shapes and flukes with different patterns. On average it took the system 40 iterations to rest in the final position. The model in Figure 5.11(j) rests partly on the inner boundary. Also the right fluke tip could not be located correctly. Possibly other image preprocessing techniques or an improved model could be used to increase the accuracy in locating such features.

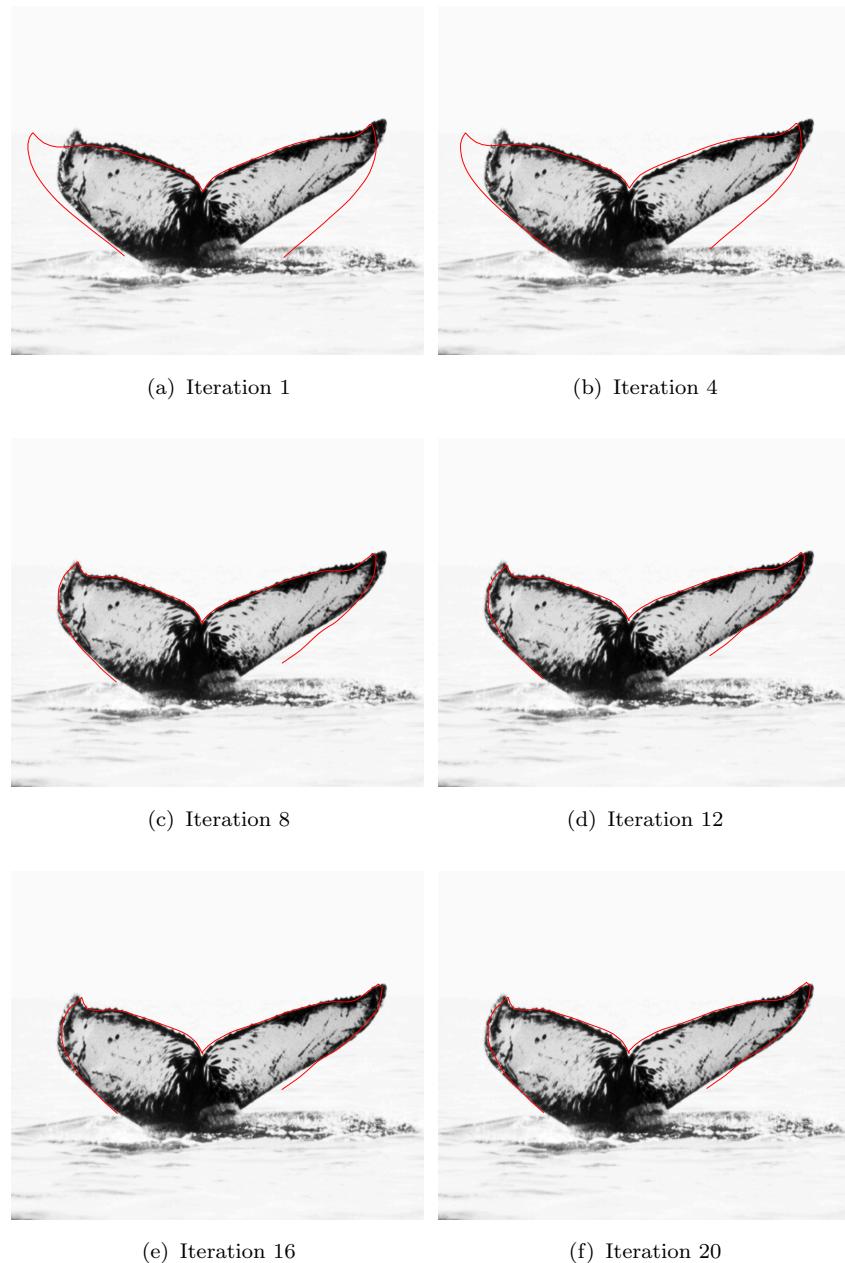


FIGURE 5.10: Fluke image with the fluke model superimposed, showing the location after different iterations.

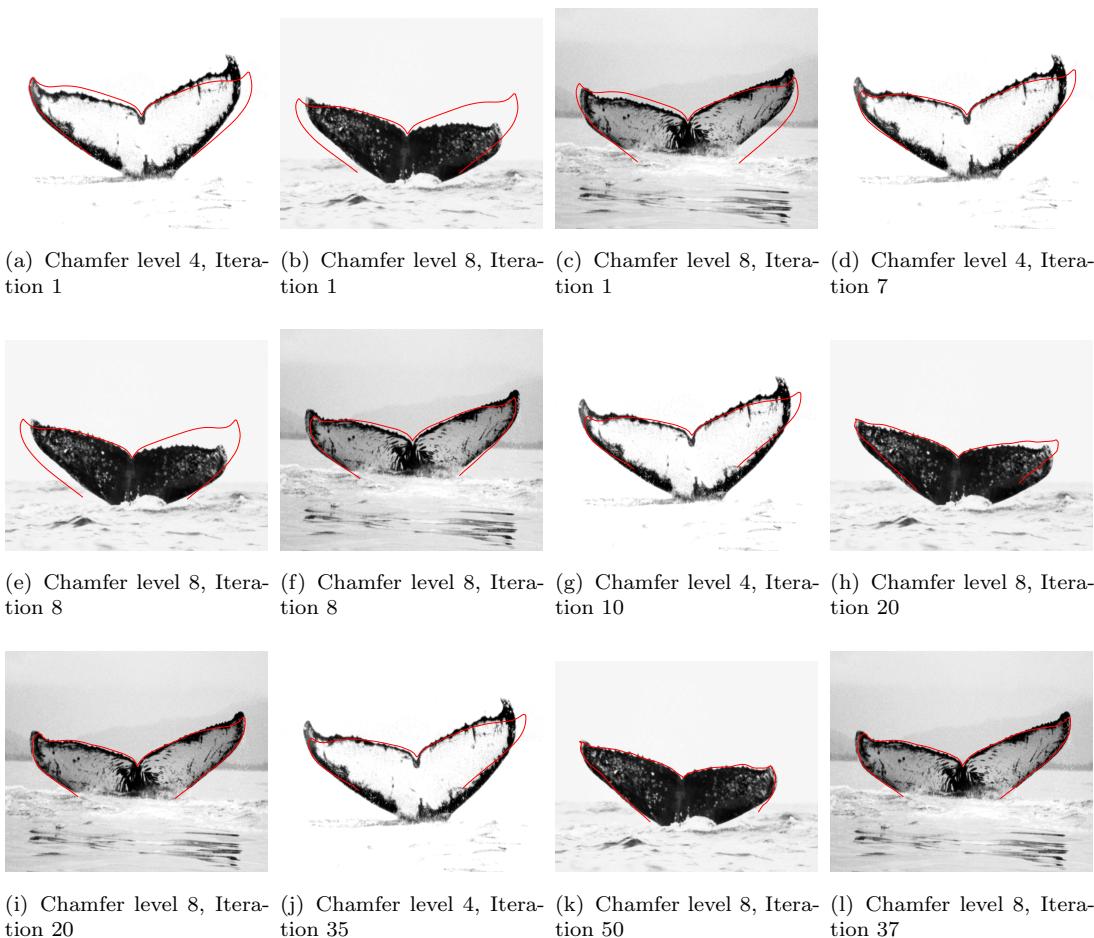


FIGURE 5.11: Different fluke images with the fluke model superimposed, showing the final location after different iterations.

# Chapter 6

## Conclusions and Future Work

This work has lead to several conclusions. The automatic location of the fluke in an image is a critical component of any automatic whale identification system based on photographs taken during field work. The anatomy of the whale fluke with limited degrees of freedom is a reasonable target for automatic image processing methods which will yield the fluke's location in a noisy image with a natural background. The ability to build a fluke model based on active shape models was prototyped and the results demonstrate that this process locates the fluke in the image with a high rate of success. The fluke location process did require image preprocessing such as image smoothing, edge detection and especially chamfer distance transformation to function well. The results of this research are very promising and suggest that the goal of providing a useful system to whale researchers would be possible with more time and resources. In particular there are several avenues of future exploration which have been identified below which would help to improve the system.

Increasing the size of the training set used to build the model would enable greater generalisation of subtle data. A more accurate model could be generated by placing more points on the boundary of the fluke when labeling the training set images. Further, an automatic method for placing the model at the right place in the image could be researched and developed in order to further automate the location of the fluke.

To overcome the problem of the model not encircling the whole fluke, growing snake properties could be included in the system, as described by Neuenschwander (1996). Also different image preprocessing techniques such as Canny edge detection (Nixon and Aguado, 2001) could be implemented and tested for bet-

ter performance of the system. The region-based thresholding technique could be developed as part of the system for improved performance.

The system could be converted from Matlab into C or C++ to improve the processing speed. The image database should be increased and intensive performance testing should be carried out in order to investigate how reliable and stable the system is when locating the fluke. Grey level images could be replaced by colour images since the following provide more information.

After testing and improving the system performance of locating the fluke in the image, pattern recognition techniques have to be implemented for fluke matching which is the goal of any automatic identification system.

## **6.1 Complete System Requirements**

A complete system useful to whale researchers would require the following features:

- Digitize the images.
- Locate fluke in image.
- Scale and rotate fluke to standardized plane.
- Perform pattern recognition
- Classify the images.
- Build database of images.
- Web-base the database for better sharing of information.
- Search through database for possible matches.
- Display the matches on screen.
- Exclude low quality photographs.
- Allow interaction with user.

# Bibliography

- H.G. Barrow, J.M. Tenenbaum, R.C. Bolles, and H.C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. *In Proceedings of 5th International Joint Conference on Artificial Intelligence*, pages 659–663, 1977.
- F.L. Bookstein. *Morphometric Tools for Landmark Data*. Cambridge Univ. Press. London/New York, 1991.
- G. Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10: 849–865, 1988.
- C.L.K. Burton. Sighting analysis and photo-identification of humpback whales off western australia. *Memoirs of the Queensland Museum*, 30:259–270, 1989.
- P.J. Clapham and C.A. Mayo. Reproduction of humpback whales (megaptera novaeangliae) observed in the gulf of maine. *International Whaling Commission, Special Issue* 12:171–175, 1990.
- T.F. Cootes and C.J. Taylor. Active shape model search using local grey-level models: A quantitative evaluation. *British Machine Vision Conference*, pages 639–648, 1992.
- T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models—their training and applications. *Computer Vision and Image Understanding*, 61:38–59, 1995.
- W.J.L. Felts. *Some Functional and Structural Characteristics of Cetacean Flippers and Flukes*. University of California Press, Berkeley and Los Angeles, 1966.
- D.A. Glockner-Ferrari and M.J. Ferrari. Reproduction in the humpback whale (megaptera novaeangliae) in hawaiian waters, 1975-1988: the life history, reproductive rates and behaviour of known individuals identified through surface and

- underwater photography. *International Whaling Commission*, Special Issue 12: 161–169, 1990.
- S.R. Gunn. *Dual Active Contour Models for Image Feature Extraction, PhD Thesis*. University of Southampton, 1996.
- L. Hiby and P. Lovell. Computer aided matching of natural markings: A prototype system for grey seals. *International Whaling Commission*, Special Issue 12:57–61, 1990.
- M. Kass, A. Witkin, and D. Terzopoulos. Snakes:active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.
- S.K. Katona and J.A. Beard. Population size, migrations and feeding aggregations of the humpback whale(megaptera novaeangliae)in the western north atlantic ocean. *International Whaling Commission*, Special Issue 12:295–305, 1990.
- G.D. Kaufmann, M.G. Osmond, A.J. Ward, and P.H. Forestell. Photographic documentation of the migratory movement of a humpback whale (megaptera novaeangliae) between east australia and antarctic area v. *International Whaling Commission*, Special Issue 12:265–267, 1990.
- A. Lanitis, C.J. Taylor, and T.F. Cootes. A generic system for classifying variable objects using flexible template matching. *Procs. of the 4th British Machine Vision Conference*, pages 329–338, 1993.
- B.J. Lei, A. Hendriks, and M.J.T. Reinders. On feature extraction from images. *Technical report on "inventory properties" for MCCWS, TU Delft*, pages 1–57, 1999.
- N. Milstein. Image segmentation by adaptive thresholding. *Technion-Israel Institute of Technology, The Faculty for Computer Sciences*, pages 1–32, 1998.
- S.A. Mizroch, J.A. Beard, and M. Lynde. Computer assisted photo-identification of humpback whales. *International Whaling Commission*, Special Issue 12:63–70, 1990.
- W.M. Neuenschwander. *Elastic Deformable Contour and Surface Models for 2-D and 3-D Image Segmentation*. Hartung-Gorre Verlag Konstanz, 1996.
- M.S. Nixon and A.S. Aguado. *Feature Extraction in Image Processing and Computer Vision*. Butterworth Heinmann, 2001.

- A. Perry, C.S. Baker, and L.M. Herman. Population characteristics of individually identified humpback whales in the central and eastern north pacific: A summary and critique. *International Whaling Commission, Special Issue* 12:307–317, 1990.
- H.C. Rosenbaum and P.J. Clapham. Geographic variation in ventral fluke pigmentation of humpback whale megaptera novaeangliae populations worldwide. *Marine Ecology Progress Series*, 124:1–7, 1995.
- S.H. Shane and D. McSweeney. Using photo-identification to study pilot whale social organization. *International Whaling Commission, Special Issue* 12:259–263, 1990.
- D. Shi. *An Active Radical Approach to Handwritten Chinese Character Recognition, MPhil Transfer Report*. Department of Electronics and Computer Science, University of Southampton, 2001.
- M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. PWS Publishing, 1999.
- M. Yakutchik. *Humpbacks of Madagascar* <http://www.discovery.com>. Discovery Communications Inc., 1998.
- A.L. Yuille, P.W. Hallinan, and D.S. Cohen. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8:99–111, 1992.