Imperial College London

Department of Computing

Final year project
FINAL REPORT

Machine learning for the automatic extraction and classification of foetal features in-utero

---

Author:
Chris Bowles

Supervisor:
Dr. Bernhard Kainz

June 17, 2014

Abstract

This project looks to find a solution to the task of automatically locating different parts of a foetus in-utero from cinematic MRI scans using Machine Learning techniques. The solution has a large computer vision component which processes the scan and removes unwanted information. Areas which correspond to foetal body parts are located using Optical Flow and clustering, after which Machine Learning is used to decide what part of the foetus these areas belong to.

In addition, tools to track a manually defined feature, suppress maternal movement present in a scan, and to extract just the foetal tissue from the scan with different body parts identified have also been produced.

Results are promising suggesting the system produced could form the foundation of further research and help improve our understanding of foetal development. The work done may also prove to have an impact on other areas of medical image analysis, as techniques developed are potentially transferable to other applications. In particular, the work done to suppress the periodic maternal movement present in scans could be used in other areas of imaging where the presence of breathing or heartbeat is an issue.

Contents

Chapter 1

Introduction

1.1 Motivation

Prenatal diagnosis is a rapidly changing field. Ensuring the well-being of our children is one of the most fundamental desires across the entire animal kingdom. There were approximately 4 million births in the US in 2011, of which roughly 3% will have been born with a major structural or genetic birth defect, a major contributor to infant mortality [1, 2]. As technology advances, more and more information becomes available to parents about their unborn child throughout pregnancy. From the gender of the baby, all the way to the entire sequencing of a foetus' DNA [3], information which could potentially indicate a range of possible abnormalities. Whatever the ethical implications of knowing this information, it is undoubtedly true that the health and well-being of both the mother and the foetus should be of paramount importance if this information is to be acquired. There are currently many ways to perform prenatal screening and diagnosis (See Section 2 for a review of the most common techniques), ranging from totally non-invasive, to highly invasive procedures, some of which carry significant risks to both the foetus and the mother, including spontaneous miscarriage.

Amongst the most recent advances in the field of medical diagnostics is the introduction of in-utero MRI techniques (See Section 2 for technical details). These new techniques allow us to capture and view high quality images of inside a womb. This is an invaluable addition to the field of prenatal diagnostics and research, because it means that for the first time, clear 3D volumetric or 2D moving images of a foetus can be viewed and analysed. Temporal sequences are an especially useful tool for prenatal diagnostics, as there is evidence that the kind and extent of foetal movements can be used to assess the healthiness of a foetus [4].

The aim of this project is to provide a tool which, using images taken from a temporally resolved MRI scan, can detect different foetal movements and identify what part of the foetus they belong to. Tools derived from

3

this information could then be used by a physician to help identify, diagnose and treat any foetal abnormalities, potentially reducing the number of foetuses and mothers who put at risk by undertaking invasive procedures. Additionally, such a tool could be used to research the early development of humans by giving us a greater understanding of foetal behavior throughout pregnancy.

1.2 Objectives

Effective real-time MRI scanning is a very new technique, it is therefore unknown exactly what information is and is not possible to derive from these scans, and how easy it is to extract this information. As a result, it is hard to define exactly what can be achieved over the course of a project such as this. We therefore decided at the start of the project to divide it into a number of separate objectives. We fully expected these to change over time as the project developed and as we get more information about what it is realistic to achieve. Below are the initial objectives along with the some of the techniques we expected might have been used to reach that goal. Descriptions of the *italicised* techniques can be found in Chapter 2, along with the motivation for the objectives of this project. Detailed descriptions on how we worked to achieve these goals can be found in the Chapter 3.

1.2.1 Main objective

The main objective of this project has been to perform an in depth investigation into the applicability of machine learning to classify foetal movement derived from real-time MRI scans with a view of ultimately designing and developing a tool which can be used to automatically identify types of foetal movement. Such a tool has applications in a wide variety of research areas, which could potential lead to an improvement in diagnostic practices and a better understanding of human development.

Techniques: *Optical Flow* 2.3.1, *Motion Histograms* 2.3.2, *Machine Learning* 2.4

1.2.2 Side objectives

Feature Tracker

To create a tool and corresponding user interface to allow a user to select a feature (such as a foetus' foot) and then to automatically track the movement of the feature over the course of the scan. Such a tool could be used to automatically measure the forces of these movements, which will help us better understand foetal development.

Techniques: *Optical Flow* 2.3.1

Maternal movement filter

To create a tool which will remove maternal movement (such as breathing) from an input scan, so that the only movements present are that of the foetus. Removing maternal movement will make foetal movement clearer, thereby allowing more accurate motion estimates to be calculated.

Techniques: *Fourier Transform* 2.5.1

1.2.3 Bonus objective

Once the main objective is achieved, it becomes possible investigate the possibility of not only extracting and classifying foetal features, but also of using this information to discriminate between scans containing healthy and unhealthy foetuses.

Techniques: *Optical Flow* 2.3.1, *Motion Histograms* 2.3.2, *Machine Learning* 2.4

Chapter 2

Related Work

2.1 Prenatal Diagnostics

The face of prenatal diagnostics has changed rapidly over the last century. The first reported prenatal diagnosis of a foetal defect came in 1916 when, 20 years after its invention, an X-Ray machine was used to examine the development of a 7 month old foetus when physical examination had proved to be inconclusive [5]. Despite this, prenatal screening and diagnostics did not become common practice until the mid 1960s, when expectant mothers undertaking high risk pregnancies were offered the chance to undergo *Amniocentesis*, an invasive procedure which carried a risk of spontaneous miscarriage [5][6]. Since the 1960s, the number of prenatal diagnostic techniques has increased enormously, providing doctors a wide range of options for all types of pregnancies. Below is a brief description of some of the most common techniques.

2.1.1 Amniocentesis and Chorionic Villus Sampling

This procedure is generally offered to mothers undertaking a high risk pregnancy, typically this means those over a certain age, or with a history of complicated pregnancies. In general, it is offered when the risks associated with performing the procedure are outweighed by the risk of not performing the procedure [6].

Both Amniocentesis and Chorionic Villus Sampling (CVS) are invasive procedures which involve using a needle to extract either amniotic fluid, in the case of Amniocentesis, or cells from the placenta, in the case of CVS. The risks involved with CVS are slightly higher, with a risk of miscarriage being 1-2%, as opposed to 1% from Amniocentesis. However CVS can be performed much earlier, usually from 10 weeks, as opposed to 15 weeks for Amniocentesis [6].

Even if the benefits outweigh the risks, it can be a hard decision for the mother to take such risks. Non-invasive procedures tend to carry little to

no risk, and are therefore preferred. However, it is currently not possible to get some diagnostic information using non-invasive methods. If cinematic MRI scans hold diagnostic information which could be used to reduce the number of Amniocentesis and CVS procedures, then it would be invaluable for physicians.

2.1.2 Ultrasound

This is by far the most common and routine prenatal procedure. Where the facilities are available, almost every mother will undergo at least one ultrasound during the pregnancy, often at both 3 months and 6 months. The first is usually just to confirm the pregnancy and to determine the due date. The second scan is to check on the health of the baby, by looking for structural abnormalities [6]. Despite its advantages over ultrasound techniques (see the section on MRI for more details), it is unlikely that cinematic MRI will replace ultrasound for these routine examinations in the near future. This is partly because the technology is not very prevalent, but also because an ultrasound scan is currently much cheaper and more convenient, requiring comparatively less training and equipment. In fact, an ordinary MRI scan costs roughly 5 times as much as an ultrasound [7][8]. However foetal MRI is a valuable tool for further investigation, should an ultrasound scan suggest abnormalities. Volumetric sequences are already used for this purpose, but cinematic sequences will provide even more information.

2.1.3 Cell-free Foetal DNA

This is a relatively new technology that involves examining the foetal DNA which is naturally present in the mother's bloodstream. A huge amount of information can be reconstructed from this DNA, providing the ability to test for a large number of genetic disorders, as well to be able to gather other information about the foetus from a very early age (10 weeks) [9].

It is a practice full of ethical issues due to its ability to ascertain foetal characteristics at such an early stage of the pregnancy. In fact, this technique can determine the sex of a child as early as 5 weeks into the pregnancy [10]. This is of particular importance in countries where there is a preference for one sex over the other. In fact, the Indian government has banned prenatal sex determination to control the practice of terminating pregnancies which are thought to be of the wrong sex [11].

Despite the ethical complications, it remains a very effective form of risk free prenatal diagnosis. However there are many potential abnormalities which cannot be identified by examining the foetal DNA. Drugs, infections and other environmental stimuli can all effect the development of a foetus in ways which cannot be identified by DNA analysis, but can be seen using foetal MRI.

2.1.4 Ethics

As touched upon perviously, the field of prenatal diagnostics is fraught with ethical controversy. It is a sensitive subject for many due to its association with miscarriage and termination. It can be a highly difficult decision for families to make for the mother to undergo an invasive diagnostic procedure due the risks involved. Professional counselling is often available to offer advice to those who may be at risk of passing on an inherited condition to their child, including recommendations of prenatal tests, and an analysis of the risks to both the foetus and mother carried by such procedures [6].

As well as the immediate health risks, there is also the question of what course of action is taken given the information derived from testing, and the impact that this will have on both the foetus and the family. Undoubtedly the most ethically complex situation occurs when a prenatal test suggests a serious foetal abnormality, and the mother is offered the option of aborting the pregnancy. Different countries have different laws regarding termination, however in the situations when the mother is given the option, it can be a very stressful choice. Studies have shown that mothers who undergo termination in the second or third trimester are at risk of psychological disorders including post traumatic stress disorder, with a reported 16.7% suffering from some form of affective or anxiety disorder after 14 months [12].

A person's Religious belief can also influence their opinion on prenatal diagnosis, with many religions directly addressing the issue. A complete analysis of these opinions is beyond the scope of this report, however, as an example, the Catholic position is that: *"... [Prenatal diagnosis] is permissible...if the methods employed safeguard the life and integrity of the embryo and the mother, without subjecting them to disproportionate risks. But this diagnosis is gravely opposed to the moral law when it is done with the thought of possibly inducing an abortion depending upon the results...."* [13].

However it is hard to argue that prenatal diagnostics does not carry a lot of benefits. Consider the situation when, say, an infection is identified through a diagnostic procedure. This infection can now be treated with antibiotics, potentially curing the foetus of a disease which could have had serious implications.

2.2 Foetal MRI

Magnetic Resonance Imaging (MRI) is a diagnostic imaging technique used to visualise the anatomy of a body. It has uses across the entire spectrum of diagnostics. The process involves exposing the subject to a strong oscillating magnetic field, which causes hydrogen atoms present in water molecules in the tissues to become excited. When the field is off, and hydrogen atom relaxes, it emits a small burst of electromagnetic radiation in the form of a radio wave which is picked up by sensors. The time delay between

the field switching off and relaxation will be different in different tissues, therefore providing a means to differentiate between different parts of the anatomy. If additional contrast is required to get a more precise image, a contrast agent can be given to the patient, which effects the relaxation delay. All of this information can then be reconstructed to produce an image like the one seen in Figure 2.1. For purposes of comparison, Figure 2.2 shows an image taken using ultrasound. There is little difference in detail and clarity between to two, however, the ultrasound has a much smaller field of view. This means that more developed foetuses cannot be fully captured using ultrasound. It is also important to note that there is no significant evidence of harm or injury to the foetus from exposure to the magnetic field used in MRI [14].
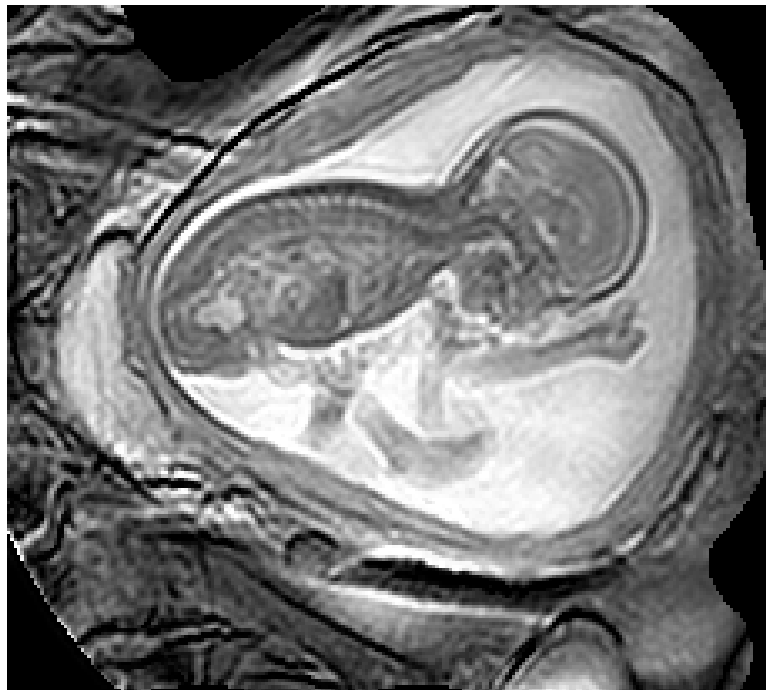
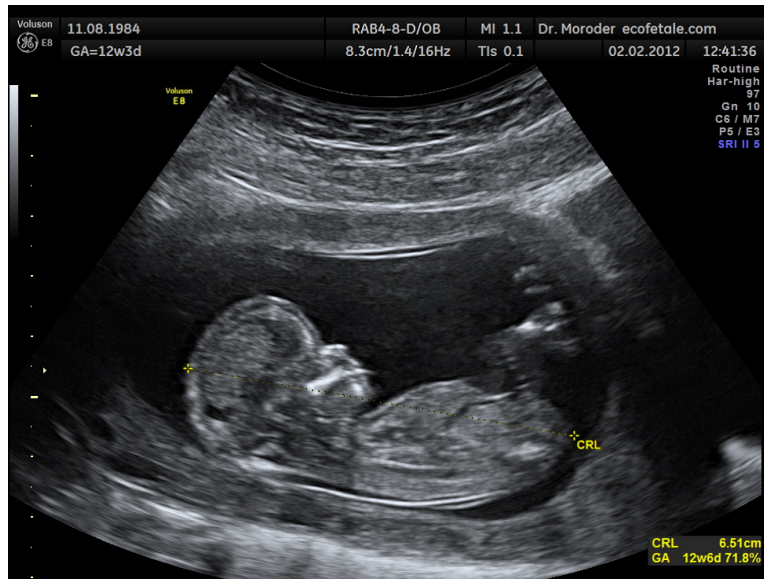Figure 2.1: An example of an MRI scan showing a foetus. A single frame from a cinematic MRI scan.

Figure 2.2: An example of an ultrasound scan. *Source Dr. Wolfgang Moroder*

2.2.1 Cinematic MRI

Cinematic MRIs build on conventional MRIs by taking snapshots at a high rate. The exact processes involved in achieving this requires an understanding of the physics involved which is beyond the scope of this report, however the results are very impressive with temporal resolutions as low as 2ms (50 frames-per-second) [15]. Unfortunately, this high frame rate comes with trade offs regarding the image size and quality. The scans used in this project are therefore taken at three frames per second. Figure 2.3 shows an example 5 seconds from one of the scans.

Some features to note from these images:

- There is maternal movement (breathing, heartbeat and digestive movements) which moves the foetus around through the frames. This can be seen most clearly by looking at the width of the right hand wall of the uterus, which moves between frames as the mother breaths.

- The foetus can display a wide variety of movements. For example, one can see the frames start with a kick of the foetus' left leg, after which it tucks its head in.

- Limbs cannot be clearly seen at all times. The left leg is very clear in the first few frames, but then fades.

- There can be large spatial differences between frames. Since each frame covers 1/3s, there can be big differences between successive frames. This can be seen most clearly during fast movement such as the kick at the start.
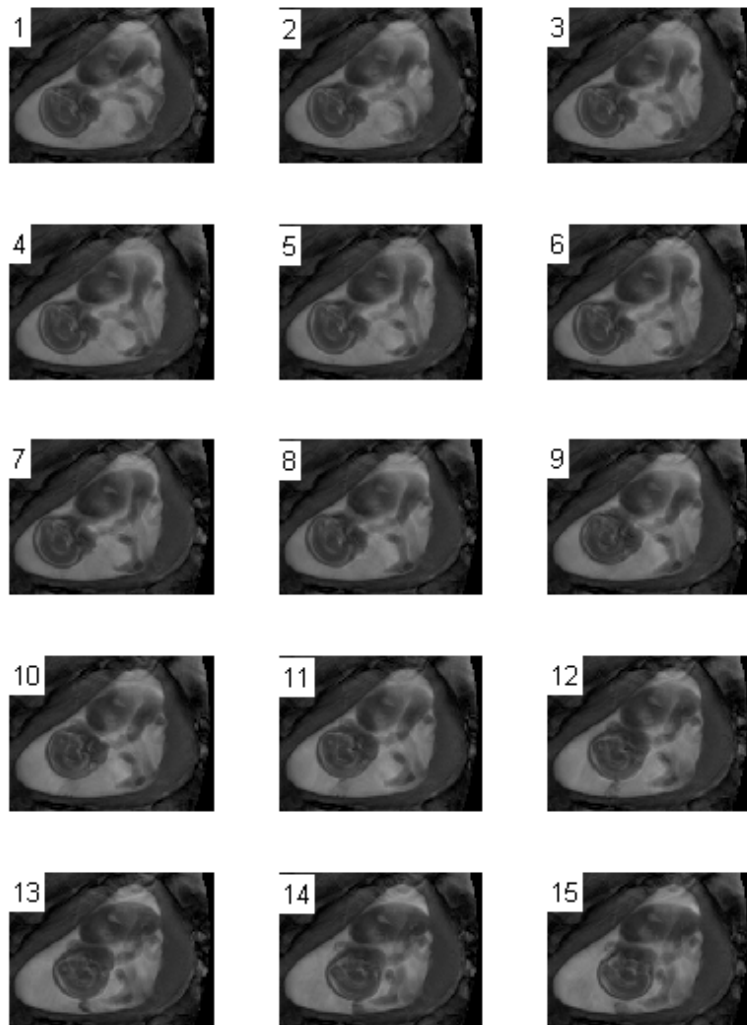

Figure 2.3: 15 frames from a cinematic MRI scan corresponding to 5 seconds of foetal movement. Frames cropped to highlight movement.

Cinematic MRI scans are not always clear however, for example in Figure 2.4. The head of the foetus appears to be obscured by shadow. This is a result of magnetic susceptibility artefacts due to local magnetic field inhomogeneities introduced by quick foetal movements. The magnetized areas used to measure the density at a particular location are moved outside of the plane of view.
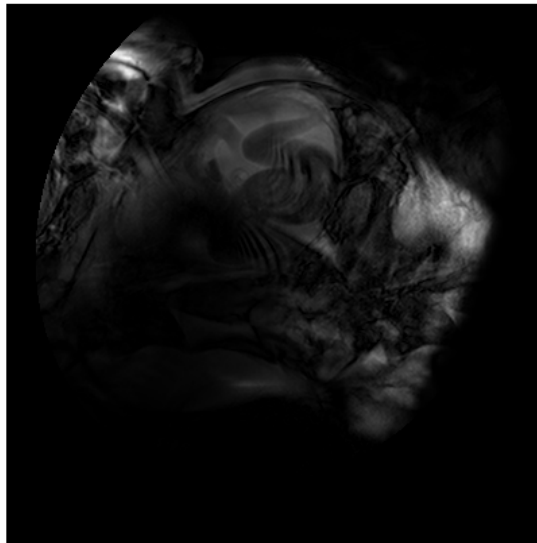
Figure 2.4: First frame from an unclear scan. Note the shadows appearing to obscure the foetus' head.

2.2.2 Current Application of Foetal Cine MRI

Given the novelty of foetal cinematic MRI sequences and the fact that it is currently not yet very prevalent in hospitals, there has not been a great deal of research into the area. Currently, the technique has been used primarily in cardiology, to provide detailed views of a moving heart [16]. As of yet, there has been very little research into its use for prenatal diagnostics. The biggest advance in the application of cinematic MRI sequences in the area of foetal development has been the capturing of a birth [17], which, whilst fascinating, is not relevant to this project. Instead, foetal movement is usually viewed using ultrasound techniques. However the quality of images created using ultrasound tend to be poor and hard to extract anything but the most basic movements from, especially at later stages of foetal development. Cinematic MRI provides an excellent opportunity to view and analyse movements which would previously have been very hard to detect.

In [18] Nowlan et al. discuss the importance of foetal movement to the development of healthy skeletons, demonstrated in chick embryos by Hosseini et al. [19]. Being able to accurately identify the types and extent of movement of a foetus in-utero will allow us to measure and better understand the forces exerted on the foetal skeleton. This in turn will help us understand the role of muscular contractions in the development of a healthy foetus. A tool which can automate this process will therefore contribute significantly to this area of research.

2.3 Computer Vision Background

2.3.1 Optical Flow

Optical Flow is a technique used to extract movement from an image sequence. An optical flow field can be created by comparing successive frames and calculating estimates for motion between them. Each motion estimate will take the form of a vector, which describes the displacement between an object's location in the first and the second image. Optical flow fields can be either dense (Figure 2.5) or sparse (Figure 2.6). A dense field will include motion estimates for each pixel in the image, whereas a sparse field will only include motion estimates for certain pixels corresponding to detectable features in the image. Dense fields have the advantage of being guaranteed to cover all of the motion which is present in an image sequence. The disadvantages are that, by covering all the motion, dense fields are guaranteed to pick up spurious movements caused by noise, and are likely to take longer to calculate than sparse fields. Sparse fields tend to be preferred when tracking a small area through a sequence as opposed to looking at global motion.
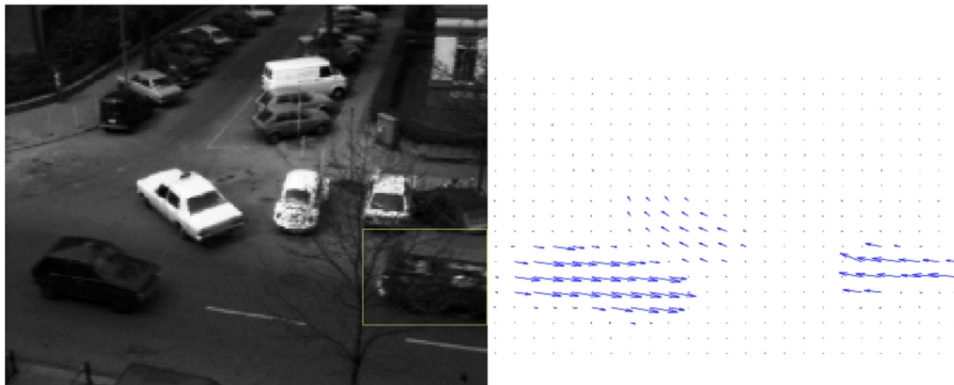

Figure 2.5: An example of a dense optical flow field. Arrows in the right hand image correspond to the estimated movement of the pixels in the left hand image. The field has recognised the three moving vehicles, and provides an estimate of their velocity. [20]
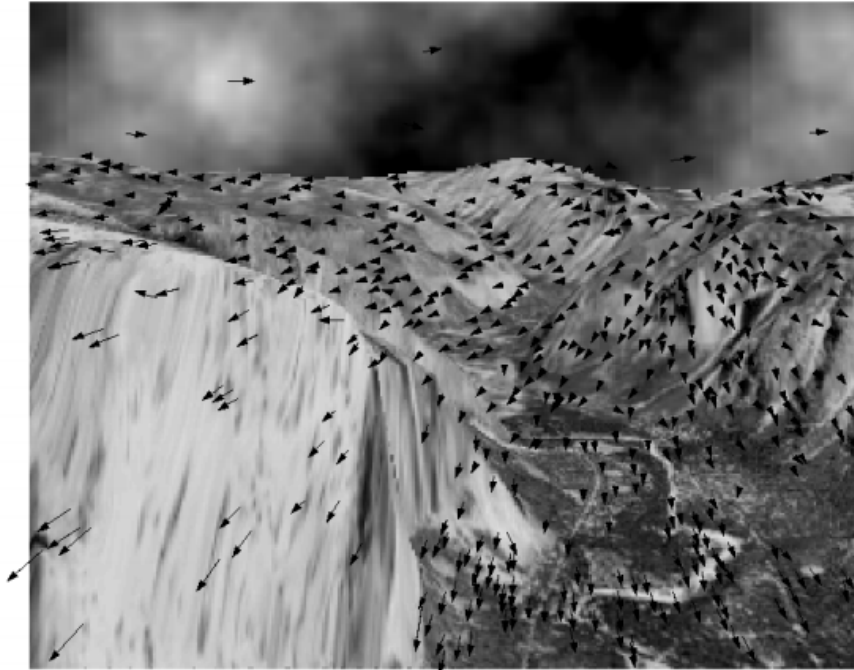
Figure 2.6: This sparse optical flow field has been created by finding matching features between two frames and thus finding their displacement [21].

There are a lot of different techniques to calculate the optical flow of an image. However two popular techniques which are included in Matlab's Computer Vision System toolbox [22] form the initial basis of the investigation. These are the *Horn-Schunck Method* [23] and the *Lucas-Kanade Method* [24].

2.3.2 Motion Histograms

A motion histogram is a method which allows us to aggregate the information from the changes of individual pixels in such a way as to provide a more high level interpretation of the movement in a scene. It is an effective way of producing a signature describing a movement, which can let us differentiate between different actions by providing a summary of all the movement in a scene.

A Histogram of Optical Flow (HOOF) is a motion histogram created using optical flow information. It will have a number of bins corresponding to a segment of 360°. For example, each bin in a motion histogram with 8 bins will correspond to 45°. Bin 1 will correspond to $0° - 45°$, bin 2 to $45° - 90°$ and so on. Each frame in the sequence is then analysed using optical flow, with each vector voting for a bin in the motion histogram, with

votes weighted by magnitude.

The basic structure outlined above is not necessarily always the most useful for a particular application. For example, one may want a measure which is invariant to whether something is moving left or right. In this case, it would be better to use histograms with bins divided as in Figure 2.7. In another case it might be important for the measure to be scale invariant, so that an action which happens far away produces the same histogram as one which is closer in the scene. This can be done by normalising the histogram so that all the bins sum to 1. Both of these techniques are used by *Rizwan Chaudhry et al.* in [25].
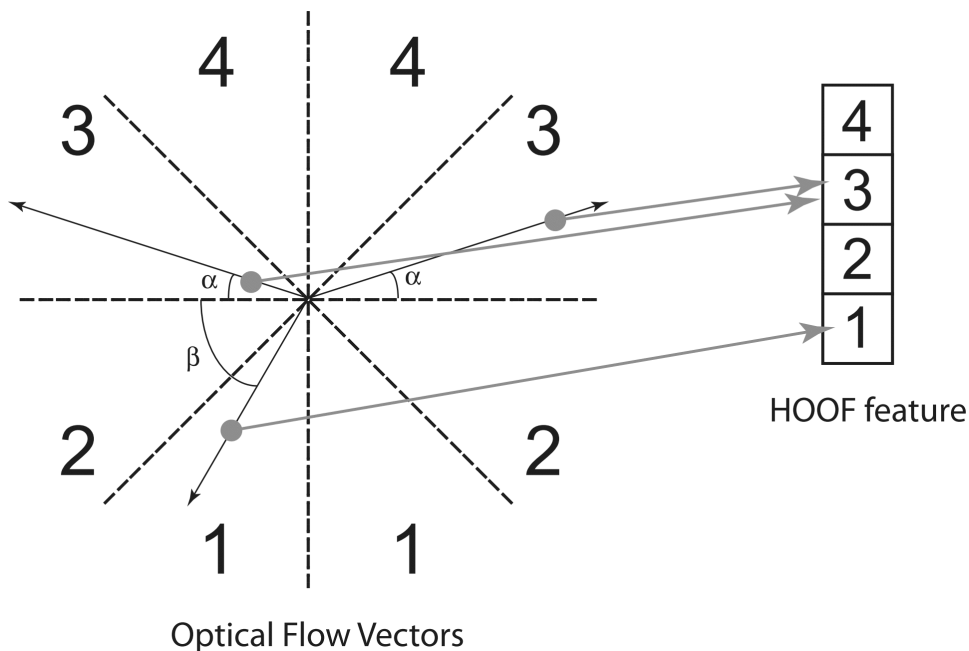
Figure 2.7: An example of a 4 bin motion histogram set up which is horizontally invariant [25].

An alternative approach is to first create a Motion History Image (MHI), and get the movement information from the gradient at each point in the image, a technique used by *James Davis* in [26]. An MHI is an image generated by repeated differencing of the silhouette of the object you wish to track. Each frame is compared with the next and the difference between the silhouettes is calculated. These differences are then layered on top of each other to create the MHI (Figure 2.8). The intensity with which each frame difference is added is proportional to its position in the sequence. This means that areas corresponding to movements which happened towards the start of the sequence will be darker than those towards the end. In this way, the silhouette can be thought to have been smeared through time. As a

result of this, the gradient of the motion histogram at a particular point will correspond to the speed and direction of movement through that location. These gradients can be calculated by convolving with gradient masks, and the resulting directions and magnitudes can be used to populate a motion histogram.
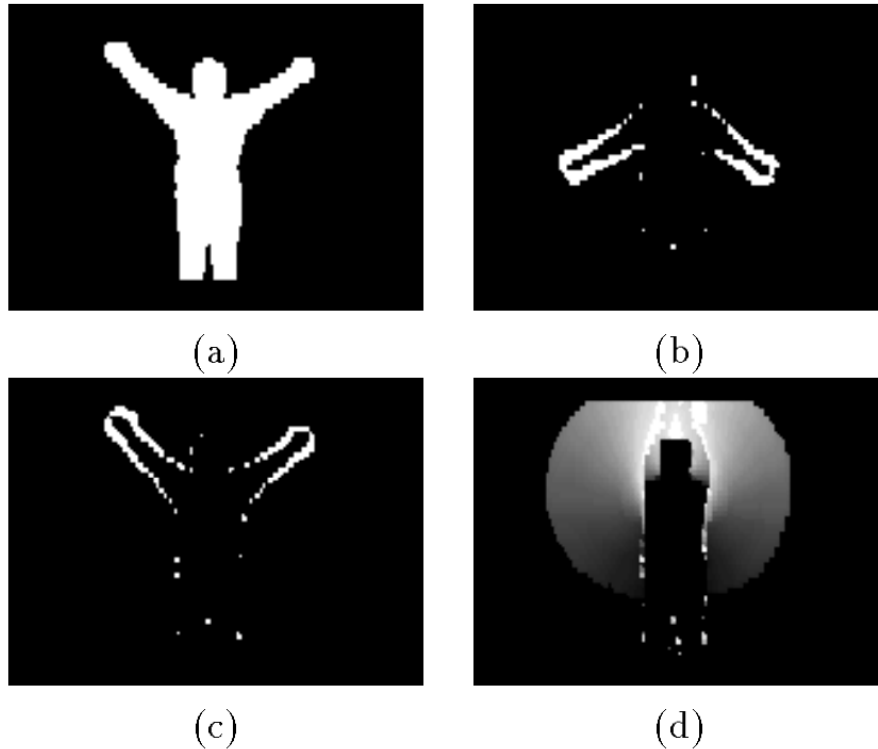

(a) (b)

(c) (d)

Figure 2.8: Generation of MHI for raising arms movement. a) Sample silhouette of the person with their arms raised near the end of the movement. b) A difference of silhouettes early in the sequence. c) A difference of silhouettes later in the sequence. d) Resulting MHI of layered silhouette differences normalized for display [26].

This technique may not be applicable in this project due to the low frame rate of the cinematic MRI scans. This will cause the MHI to be choppy, and therefore hard to extract meaningful gradient information from. Another challenge when using an MHI approach is that it must be possible to segment the image so as to have a silhouette of the moving object. This could be hard to do accurately and reliably in the cinematic MRI scans.

The general idea is that every action which can happen in a scene will produce a different motion histogram. This will allow us to differentiate between different types of foetal movement.

It is important that movements are segmented into basic atomic movements. For example, if you have a long sequence including a lot of movements, the information from the first movements will be overwritten by later movements. This can be done by segmenting an sequence based upon when an object moves back in the direction it came from. For example, an sequence containing an object moving left then right again will be segmented into a left movement and a right movement.

A single motion histogram lets us see what directions objects in the scene appear to be moving in. While useful, this would not be enough to differentiate between, say, a foetus moving its leg up or moving its hand up. In order to get this extra information, we can create a number of motion histograms, each created using information from a different part of the image (Figure 2.9) [25]. This will tell us information such as that there is a lot of upward motion in the left hand side of the image. If we know the foetus' orientation, it may be possible to work out if it corresponds to a leg or hand movement.
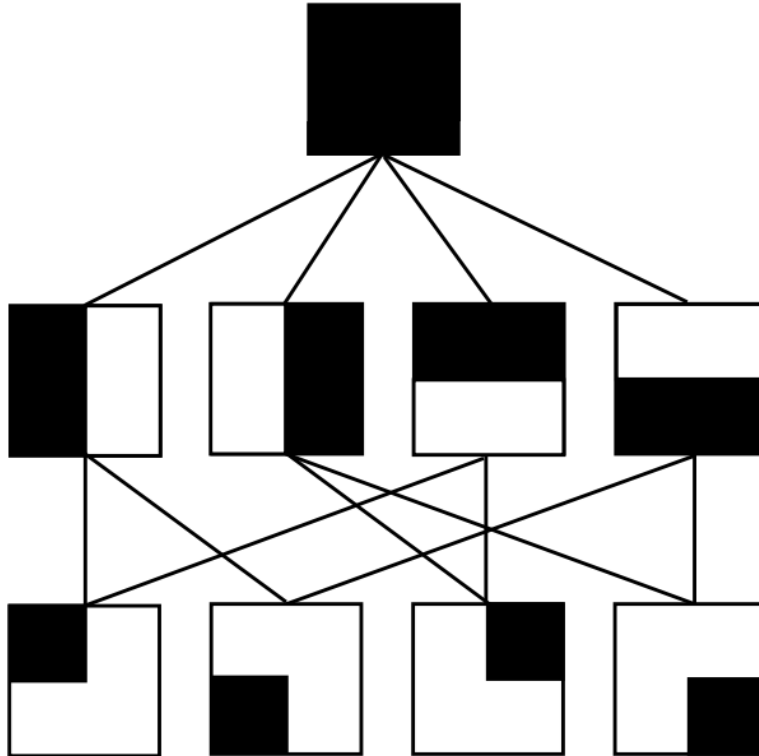
Figure 2.9: Overlapping windows for generating motion histograms. The dark areas represent areas which are included in that window's histogram; white areas are ignored. The first window (top window) covers the entire motion region within the MHI. The windows below cover progressively smaller regions of the motion [25].

2.4 Machine Learning Background

Machine Learning is an exceptionally broad term which covers a huge number of techniques and applications. What exactly constitutes machine learning is difficult to define, however Tom Mitchell, an American computer scientist who is widely regarded as one of the most important contributors to the field of machine learning, gives the following definition:

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." [27]

Most of the information in this section has been taken from Tom Mitchell's

book, *Machine Learning* [27], with additional information from Imperial College London Department of Computing course notes [28, 29, 30, 31].

Machine learning algorithms can mostly be categorised as belonging to one of two broad families, *supervised* and *unsupervised.*

Supervised learning algorithms are those which learn from sets of what are known as labelled examples. A labelled example is a piece of data which we know what it describes. For example, the a labelled example could be a description of a tree (width, height, bark colour...) along with a label which tells us what type of tree it is. The supervised learning algorithm will learn from many of these labelled examples, and then aim to classify previously unseen inputs. In other words, it will aim to decide what kind of tree a particular input describes, using what it has learnt about different types of trees from the training data.

Unsupervised learning, on the other hand, works with unlabelled examples. The aim here is to look for patterns and structure in the input data. For example, it could perform clustering to suggest how many distinct types of tree are present in a set of data. This is closely linked to the field of data mining.

2.4.1   Supervised Learning Algorithms

Naive Bayes

Naive Bayes classifiers use Bayes' theorem (Equation 2.1) to create a probabilistic model from training data in order to classify an novel case. Naive Bayes classifiers make the strong assumption that features are independent. That is to say, that in the case when we are classifying trees using *height*, *bark colour* and *leaf size* as descriptive features, each feature contributes independently to the probability a tree is of a particular type, despite the values of the other features.

$$P(D|S) = \frac{P(D)P(S|D)}{P(S)} \tag{2.1}$$

The probability of a tree being of a particular type (T), known as the *posterior* probability can therefore be expressed as,

$$P(T|\Theta) = \alpha P(T)p(h|T)p(c|T)p(s|T) \tag{2.2}$$

Where,

- Θ is a set of observations $\{height = h, barkcolour = c, leaf size = s\}$

- $P(T)$ is the proportion of all the training examples which are classified as T. Known as the *prior* probability.

- The value of $p(x|T)$ can be found by calculating the proportion of training examples classified as T which have a value of x for that particular feature.

Note it is necessary to quantise continuous features. For example if $0 \leq height < 100$, it can be quantised into 4 discrete values, $\{0 \leq height < 25, 25 \leq height < 50, 50 \leq height < 75, 75 \leq height < 100\}$. In reality it may be better to quantise into a unequal sections, depending on the properties of the feature.

Support Vector Machine

A Support Vector Machines (SVM) are primarily binary classifiers. This means they can take training examples corresponding to two different labels. It can then take an input and classify it as one of the two labels. It does this by determining what is called the maximum margin hyperplane separating the two classes of training examples. This is a hyperplane which completely separates the two sets of training examples, whilst maximising the distance between the plane and the closest points. It can then check which side of this hyperplane a novel input lies on, and therefore what to classify it as. The problems with SVMs come when the data is not linearly separable, as can be see in Figure 2.10. However, if the examples are mapped into a higher dimensional space, it does become possible to separate them, see Figure 2.11. In fact, a technique called the *Kernel trick* can allow us to treat the data as being in a high dimensional space, yet still let us perform the necessary calculations in a low dimensional space. This makes it possible to find a separating hyperplane in cases in when it was previously impossible. There are extensions to SVMs which allow them to be used for multi-class classification, and a more noise resistant version which allows a number of points to lie on the incorrect side of the hyperplane.
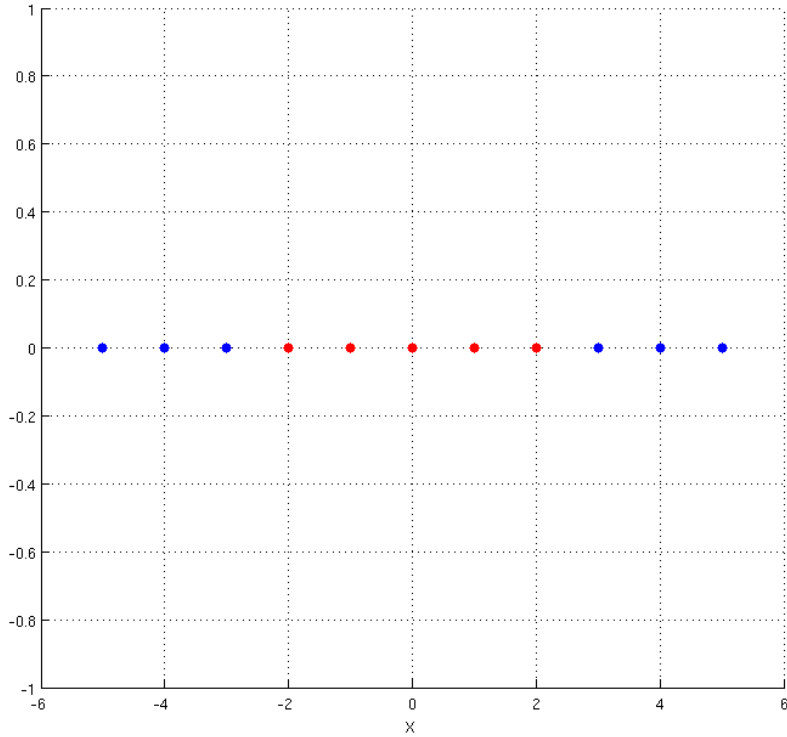
Figure 2.10: A set of 1 dimensional data where a separating hyperplane cannot be found. If we treat red and blue dots as examples of different classes, we cannot find a hyperplane which separates the two colours.
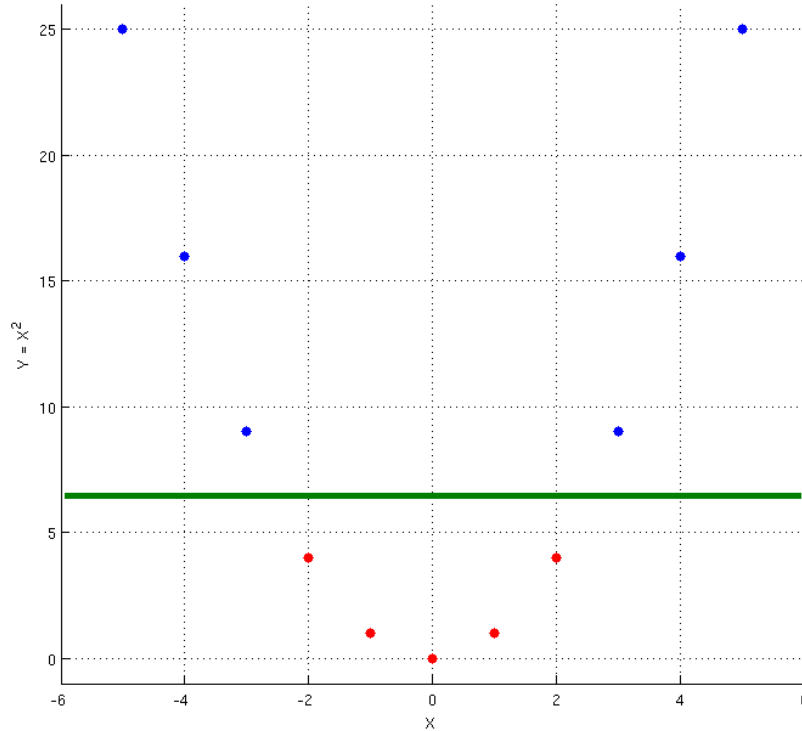
Figure 2.11: By mapping the examples into a higher dimensional space, it is possible to find a hyperplane (green line) which separates the two colours.

K-Nearest-Neighbour

K-Nearest-Neighbour (KNN) algorithms store labelled training examples and then classify future inputs by finding which stored examples most closely resemble it. The similarity between two examples can be calculated in a number of ways, however it commonly uses the euclidean distance between the two examples. The system will look to find the k most similar entries to the input, and typically classify it according to the most common label among those k entries. See Figure 2.12. If the labels are continuous, the label given to the input could be an average (perhaps weighted by similarity) of the k labels. The choice of k can have a significant impact on the performance of the classifier. It is common to evaluate the classifier over a range of k values, and choose the value which gave the best performance. It is important to use cross validation during this procedure (see the Section 2.6 for further details of cross validation). KNN works best in low dimensional cases and when each dimension is known to be relevant (see Section 2.4.2). High dimensions or irrelevant features will lead to poor simi-

22

larity measures. Preprocessing data by performing dimensionality reduction before using KNN will tend to improve the performance of the classifier.
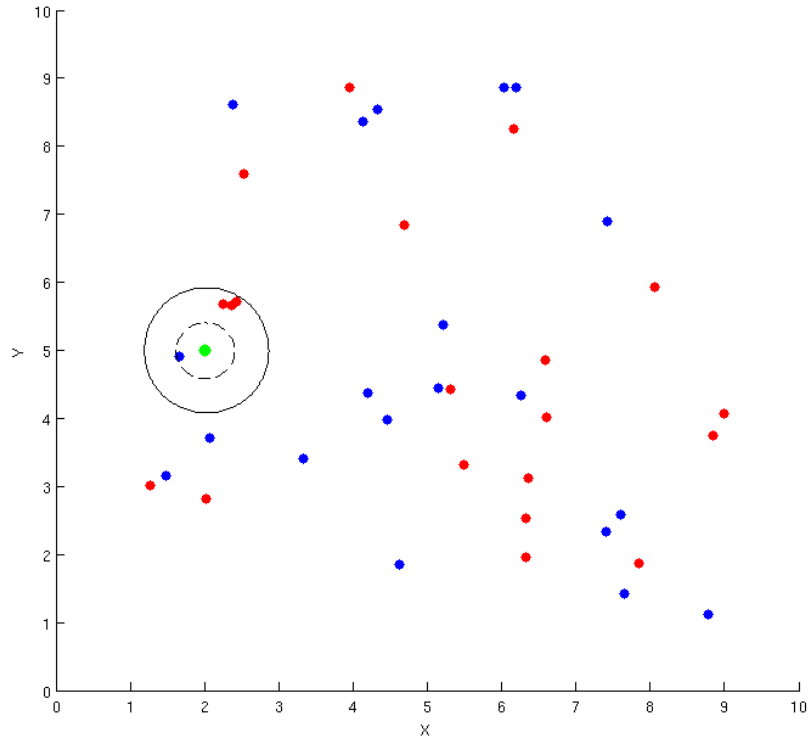
Figure 2.12: A simple K-Nearest-Neighbour representation. Red and blue dots represent different classes of trained examples. The green dot represents an input we would like to classify. If the value of k is 1, then the input would be classified as blue as its most similar entry is blue, shown by the dashed circle. If the value of k is 4, the input would be classified as red, as the most common label among the 4 most similar entries is red, shown by the solid circle. This demonstrates the importance of the value of k.

Decision Tree

A decision tree uses information theory to generate a tree comprising of nodes corresponding to the features of the training examples, and edges from the nodes corresponding to the values which that feature can take. Note that if a feature is continuous, it must first be quantised, for example if $0 \leq x < 100$, the feature can be quantised into 4 discrete values, $\{0 \leq x < 25, 25 \leq x < 50, 50 \leq x < 75, 75 \leq x < 100\}$.

The tree is built from the root node to the leaves. The root node is

selected by using a measure of how much information any given feature tells us about the data. One algorithm called the *ID3 algorithm*, uses a measure called the information gain defined in Equation 2.3. The data is then split into two or more sub sets corresponding to the different values the chosen attribute can take. The attribute with the highest information gain is chosen as the next node in the tree, and the process is repeated on the remaining subsets.

$$IG(A) = H(S) - \sum_{t \in T} p(t)H(t) \qquad (2.3)$$

Where,

- A is the feature which we are using to split the data set

- $H(S)$ is the entropy of set S using Equation 2.4

- T are the subsets of S created by splitting the data using A

- $p(t) = \frac{|t|}{|S|}$

- $H(t)$ is the entropy of subset t using Equation 2.4

$$H(S) = - \sum_{x \in X} p(x) \log_2 p(x) \qquad (2.4)$$

Where,

- S is the current data set we wish to split

- X is the set of labels present in S

- $p(t) = \frac{|x|}{|S|}$

The tree can then be post processed to improve performance. For example, generality can be improved by trimming long branches. Novel cases can then be classified by working through the decision tree like a flow chart.

2.4.2 Dimensionality Reduction

However, it doesn't matter how clever your algorithm is if you don't train it on the right information. In many situations, the choice of features which comprise the data the classifier has been trained on is much more important than the choice of algorithm itself. Consider the tree classifier from before. A good selection of features might be *height*, *trunk width*, *bark colour* and *leaf size*. These are all features which can be used to differentiate one type of tree from another. However, if your features are *time of year*, *weather* and *if I was wearing a hat when I saw it*, then even the most sophisticated algorithm won't be able to correctly differentiate between different types of tree, because the information it has been trained on is completely irrelevant.

The main challenge often faced when creating classifiers is not what algorithm will be best, but how to extract the most relevant and informative features from the data. We will be attempting to extract the most relevant information possible, and then choose an algorithm which will suite the type and dimensionality of the data gathered.

Dimensionality is an important characteristic which must be considered. Different algorithms perform better with examples of different dimensionality. It is often the case that the examples which are used to train a classifier contain features which are of varying relevance. For example, the colour of a tree's bark may be a much more important feature than the width of the trunk when determining the type of a tree. It may therefore be desirable to ignore features which are not especially relevant, or combine two less relevant features into one more relevant feature, in order to reduce the dimensionality of the data to a value which works well for a particular algorithm. There are several techniques to achieve this.

Principle Component Analysis

Principle Component Analysis (PCA) involves calculating a new basis through which to represent the available data. The dimensionality of this new basis is equal or less than the dimensionality of the original data. The components which make up the new basis are chosen to reflect the directions of the most variance in the data. The components are orthogonal. The first principle component will points along the direction of most variance, the second will then point along the direction of most variance, orthogonal of the first (so as not to contain the same information) and so on. This can be seen more clearly in Figure 2.13 Therefore, to reduce an n dimensional data set to an $n' < n$ dimensional set, you need only select the first n' principle
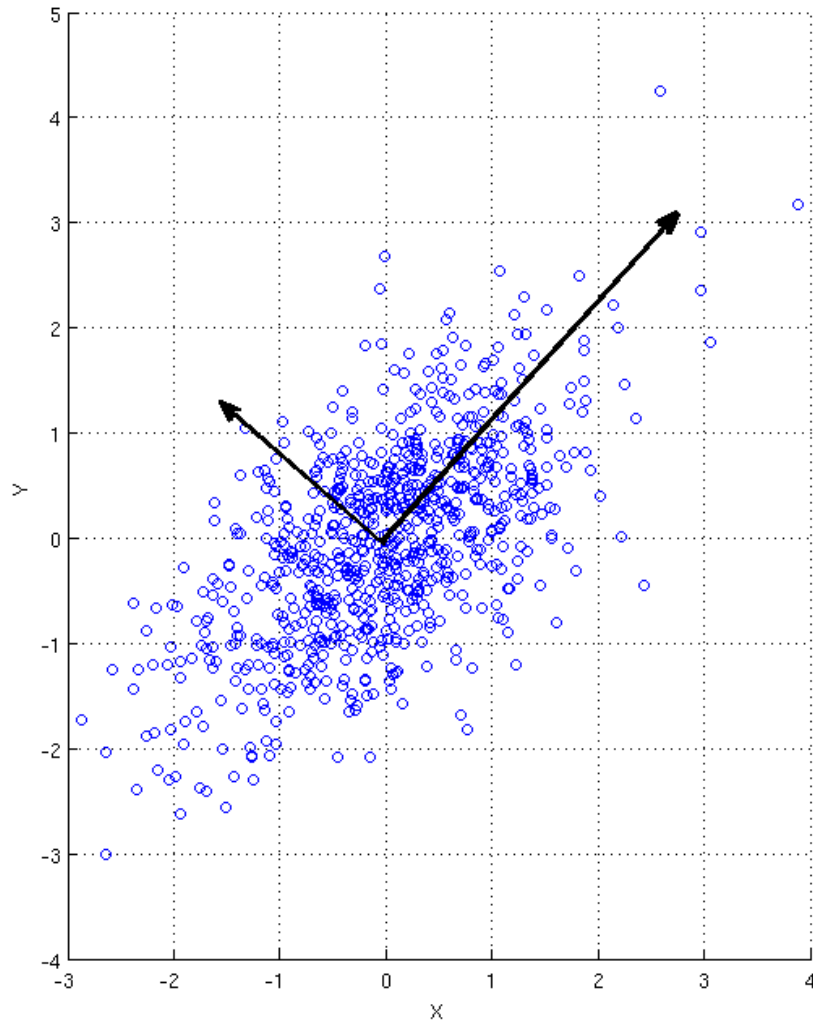
components.

Figure 2.13: A scatter plot showing a multivariate Gaussian distribution. The arrows correspond to the directions of the principle components. The larger arrow points in the direction of greatest variance and so describes the first principle component. The smaller arrow corresponds to the second principle component.

The process of calculating the principle components of a data set is quite straightforward. First we must calculate the covariance matrix of the data.

$$\Sigma = E[(\mathbf{X} - E[\mathbf{X}])(\mathbf{X} - E[\mathbf{X}])^T]$$

Where $E(x)$ is the expected value of x.

The basis described by the eigenvectors of this covariance matrix will then correspond to the directions of most variance. The corresponding eigenvalues will describe the variance along each eigenvector. Therefore, the eigenvector corresponding to the highest eigenvalue, point along the direction of most variance - the first principle component. Therefore, to choose an n'-dimensional basis to represent the data, we simply select the n' largest eigenvalues, and form the basis using the corresponding eigenvectors. In fact, the eigenvalues directly represent the proportion of the information which is contained along the corresponding eigenvectors. We therefore know exactly how much information we are losing from dropping particular components.

PCA can be an extremely useful tool. A 65536 feature dataset describing human faces can be reduced to 25 dimensions, whilst still retaining 56% of the information [32]. One feature of PCA it is important to note it that each dimension must be of the same units, otherwise comparing variances is irrelevant. PCA also relies heavily on the data being Gaussian.

2.5 Signal Processing Background

2.5.1 Fourier Transform

A Fourier transform is a mathematical method which converts a signal into a frequency domain from, for example, a spatial or temporal domain, and vice versa. Fourier transforms have a huger number of practical uses across mathematics. Two uses in particular might prove useful in this project.

Firstly, Fourier transforms can be used when searching for a particular object or pattern in an image. An image of the feature you are searching for can be transformed into the frequency domain, and a signature can be derived from the results. This signature can be scale and/or rotationally invariant, depending on the what features of the transformed image are used to to create the signature [33].

Applying a Fourier transform to a temporal signal can be used to analyse what frequencies are present in the signal. For example, components of the signal which change slowly will correspond to low frequencies, whereas those which change quickly will correspond to high frequencies. It is a common technique in signal processing to use a Fourier transform to filter out noise from a signal. The signal is transformed into the frequency domain, unwanted frequencies can then be identified and filtered out, and the signal can be reconstructed.

2.6 Evaluation Techniques

Being able to evaluate the performance of a system is arguably just as important as the implementation, for both individual components and the system as a whole. It is important to know how to judge the success of a project. The following sections discuss some methodologies which will prove useful when evaluating the contributions of this project.

Classifiers

There are a wide range of measurements which are traditionally used to evaluate a classifier (Table 2.2). These measurements can be derived from a confusion matrix, which tells us the number of True Positives (TP), False Positives (TP), False Negatives (FN) and True Negatives (TN) which are present in the classification. How these values are calculated is summarized in Table 2.1. A combination of these measures provide a quantitative way of evaluating the performance of any classifier.

		Predicted	
		Positive	Negative
Known	Positive	TP	FN
	Negative	FP	TN

Table 2.1: Structure of a confusion matrix

Measure	Formula	Meaning
Precision	$\frac{TP}{TP+FP}$	The percentage of examples which are predicted to be positive that are known to be positive
Sensitivity	$\frac{TP}{TP+FN}$	The percentage of examples which are known to be positive that are predicted to be positive
Specificity	$\frac{TN}{TN+FP}$	The percentage of examples which are known to be negative that are predicted to be negative
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	The percentage of examples which are predicted correctly

Table 2.2: Summary of classifier evaluation measures

Another technique which is particularly useful if the amount of available data is low is known as cross validation. This involves randomly splitting

the data into n equal partitions, then for each partition (N), you evaluate the classifier by training on all the data which isn't in N, and testing its performance on N. This stops the performance measurements from being biased by over-fitting, which occurs when the model fits too tightly to the training data, thus losing the ability to classify novel examples accurately.

Cross validation can also be used in the training process. Learning algorithms often have a number of parameters which must be set (the value of k in K-Nearest-Neighbour (Section 2.4.1) for example). These parameters can have a huge impact on the performance of the resulting classifier. One way of selecting these parameters is to create a number of classifiers with different parameters. Each classifier can then be evaluated using cross validation and their performance compared. The parameters which create the classifier which performs best, can then be chosen when training the final classifier.

Filtering

A simple and subjective method of evaluating the performance of a movement filter would be to survey a number of people and ask them to give a numerical score within a certain range, describing how well they think the movement has been filtered from the scan.

Another measure of how effective a movement filter is would be to calculate an average image of a filtered and unfiltered scan by simply averaging the pixel intensities for each pixel over the duration of the video. The more movement present in a video, the more blurred an averaged image will be. The images from the filtered and unfiltered scans can then be compared subjectively, with significantly less blurring in a filtered image being indicative of effective filtering. This is a qualitative measure and therefore is quite subjective. Whilst it can be used during the development stage to get a rough idea of how well different algorithms are performing, a quantitative procedure would also be useful.

One quantitative procedure would be to calculate the Mean Squared Error of pixel intensities between consecutive frames. This will give a measurement of how much a video changes between frames. A lot of change between frames would mean there is a lot of movement in the scene, therefore a lower MSE between the frames of a filtered video would be indicative of an effective filtering algorithm.

$$MSE(I_t, I_{t-1}) = \frac{1}{M \times N} \sum_{M \times N} (I_t(m,n) - I_{t-1}(m,n))^2$$

Where M and N are the width and height of a frame respectively.

Techniques taken from [34].

Possible measurements

- Comparisons of mean image (subjective)

- Opinion survey score (subjective)

- Mean squared deviation (objective)

Chapter 3

Method

.

3.1 Overview

The approach taken in this project to the problem of segmentation and classification can be thought of as a pipeline. A raw cinematic MRI scan is input, and undergoes a series of processing steps, resulting in segmented foetal movements which form the output. Figure 3.1 illustrates the processing stages, after which is a brief description of each stage.
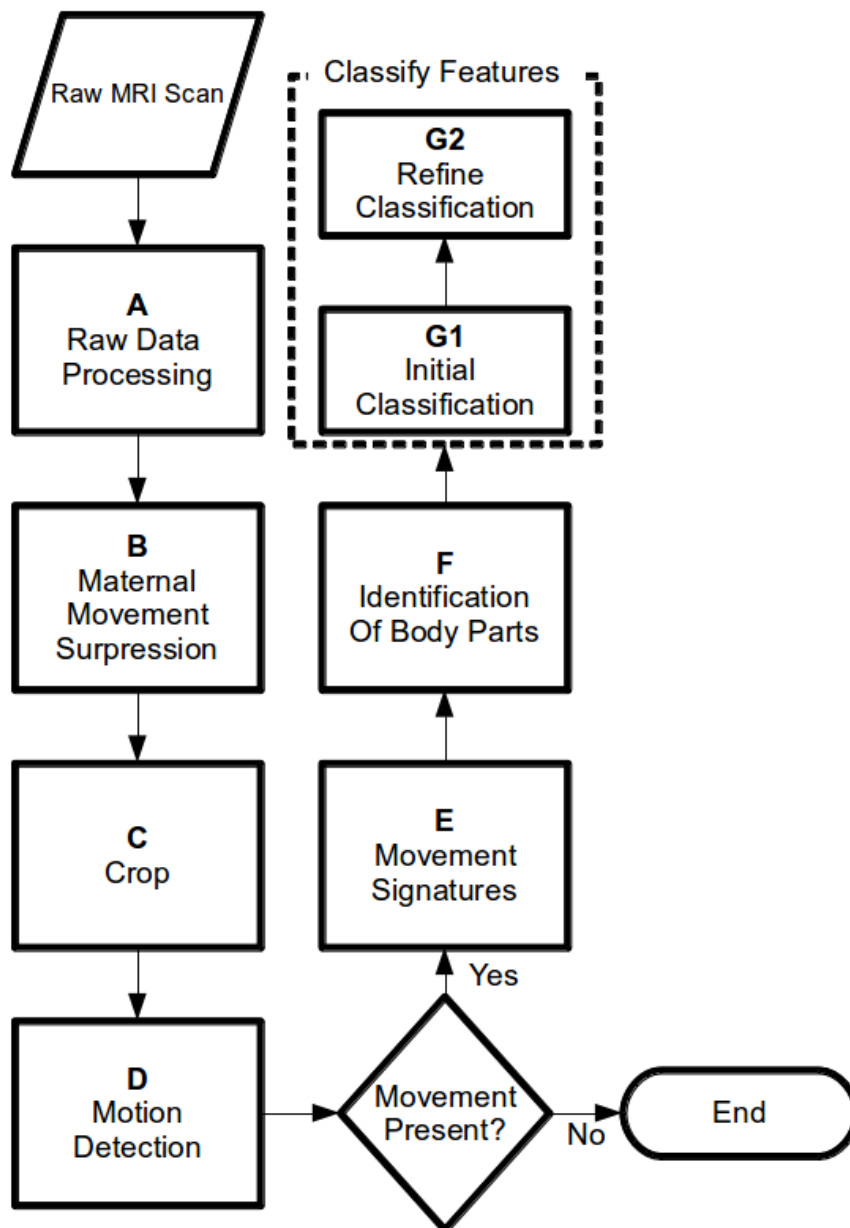
Figure 3.1: Illustration of the stages of the pipeline which make up the system developed in this project.

Figure 3.1.A - Raw Data Processing

The first stage of the pipeline involves performing some image enhancement on each individual frame of the scan so as to improve the performance of the subsequence stages.

Figure 3.1.B - Maternal Movement Suppression

At this stage it is beneficial to try and remove as much maternal movement from the scans as possible, primarily breathing. This is done by analysing the scan using a Fourier Transform to identify the frequency of the breathing. Movement at this frequency is then filtered out. This results in a scan with significantly less visible periodic movement, but a certain degree of 'ghosting'.

Figure 3.1.C - Crop

The next stage is to try and crop the scan down to, ideally, just the uterus. The benefits of this is to firstly reduce the computational load of the later stages by having them work in a smaller area, and secondly to eliminate as much of the maternal tissue as possible, in order to prevent it from being mistaken for foetal movement.

This is a two stage process. The first involves locating the areas which exhibit a lot high frequency movements, which is an indicator of foetal movement. A series of morphological operations then attempts to locate the area with the highest probability of containing the foetus. The second stage involves running a foreground detection algorithm in this area, which again works by attempting to locate areas of movement. A further series of morphological operations locate the final estimate for the foetal area. The more foetal movement exhibited by the scan, the more accurate the cropping algorithm tends to be. Scans with little or no foetal movement are left relatively uncropped, ones with lots of foetal movement are cropped very well.

Figure 3.1.D - Motion Detection

At this point it is possible to discard scans with no significant foetal movement. This is done by divining the scan into short sections and creating an average image corresponding to the frames in these sections. In cases where there is little or no foetal movement, these images are likely to be very similar in each section, despite maternal movement. If there is foetal movement, some of the images will be significantly different to the others. Comparing all images with each other and summing the total differences, we can reach an accurate measure of the amount of foetal movement in a scan. This method has an accuracy of 81%.

Figure 3.1.E - Movement Signatures

This next step involves segmenting the scan into periods of significant foetal movement. By calculating optical flow fields for each frame, it is possible to

locate periods which contain a higher than average amount of movement. These are likely to correspond to periods of foetal movement.

Figure 3.1.F - Identification of Body Parts

This process is performed on each of the movement signatures identified in the previous stage. It uses the observation that areas which move at the same time and in the same direction are likely to correspond to the same part of the foetus. By using K-means clustering it is possible to extract these areas and therefore identify separate features. Areas which are often identified by this process include: the head, the leg, the upper back, the lower back and the arms. Occasionally, left over maternal tissue is identified. The scan can then be cropped around these areas resulting in a large collection of short movements of individual parts of the foetus.

Figure 3.1.G - Classify Features

The final stage has the aim of classifying each of the features identified by the previous stage as one of: Head, body, leg, arm or maternal. This is a lengthy process which starts by dividing the features once more into periods of movement a few frames long. Signatures are then derived from each of these. These signatures are designed to extract the most relevant information from the features in order to be able to discriminate between the different classes, reflecting their shape, texture and how they move.

Using a number of manually pre-labelled scans as training data, a K-Nearest-Neighbour (2.4.1) classifier can be used to classify the features as being representative of a particular part of a foetus. However, this classification alone is not enough, resulting in only a 47.8% accuracy, but it can give us an idea of roughly where the individual parts of the foetus are. By repeatedly fixing the location of a certain part, and reclassifying, it is possible to achieve a higher level of accuracy in the majority of scans.

Additional tools

While not necessary for the main system, two additional tools which could aid future work with cinematic MRI foetal scans were developed. Firstly, a feature tracker using optical flow, which allows the user to select a point at the start of the scan, and for it to be tracked throughout. As well as this a method of visualising the segmentation of the foetus was created. Using the results of classification, and various computer vision techniques we were able to identify and colour code different parts of the foetus. These tools provide a proof of concept and demonstrate the practical uses for the achievements of this project.

3.2 Data Analysis

At the start of this project, a number of data sets containing cinematic MRI scans were made available. A number were selected as representative scans to work with when developing and evaluating the individual components of the system during development. However, in order to evaluate the system as a whole, it is important to use as much data as is available. We therefore manually viewed each scan to assess its use. Despite being a very time consuming process, it was necessary in order to fully assess the system. Figures 3.1 and 3.2 show a full breakdown of the data set. Some files were formatted in such a way as to contain a number of separate scans in sequence, in these cases, each scan was assessed and counted separately.

Assessment	Quantity	Explanation
Incompatible	16	The file contained 4-dimensional data or was otherwise corrupted.
Still	33	There was no / very little discernible foetal movement.
Short	58	The scan was too short to be considered useful. Typically containing fewer than 100 frames.
Twins	17	The scan contained twins. Figure 3.3
Unclear	64	The scan was unclear, due to significant shadows or other artefacts, or depth of scan. Figure 3.4
Non-temporal	54	The data contained in the scan included no temporal component.
Acceptable	56	The scans which did not fall into the other categories.
Total	347	The total number of scans in the data set.

Table 3.1: Summary of classifier evaluation measures
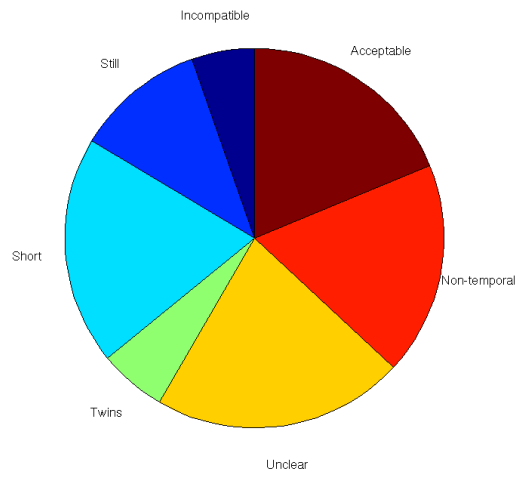
Figure 3.2: The composition of the data set.
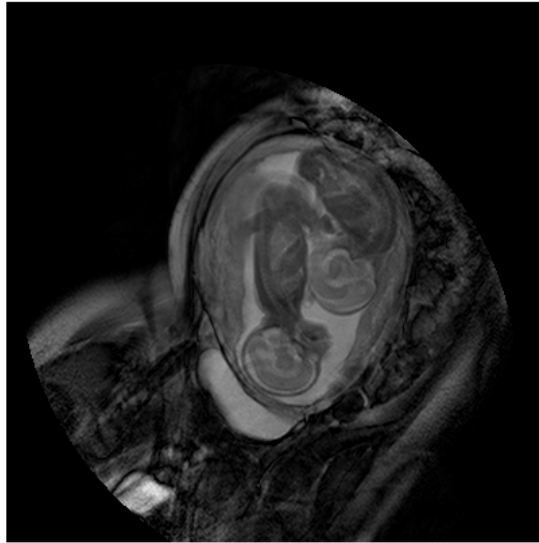
Figure 3.3: First frame from a scan containing twins.

Figure 3.4: First frame from an unclear scan. Note the shadows obscuring the foetus' head.

Due to the methods employed in the system, the decision was made to leave out the short scans and those containing twins. Twins would confuse the segmentation process, and short scans rarely contained enough information to perform segmentation. Incompatible and non-temporal scans were also discarded. As can be seen, only 26% of the dataset was considered usable from this point forward. This suggest a serious limitation to the system, and therefore requires some explanation.

Firstly, it is important to keep in mind that the aim of this project is to perform segmentation on a scan, so as to aid in the analysis of the individual foetal movements. Non-temporal, incompatible and unclear scans are clearly going to be useless in this regard, hence they can be safely discarded without reflecting poorly on the performance of the system. However the need to exclude short scans and those containing twins is one of the more significant limitations of the system, which will be discussed in Chapter 6

3.3 Raw Data Processing

As soon as the scan is loaded, it will undergo two operations designed to enhance the clarity of each frame.

The first step is to sharpen the image. This is done by subtracting a blurred version of the scan from itself, see Equation 3.1. This is repeated 3

times, the results of which can be seen in Figure 3.5.

Variable definitions:

- $S =$ The scan being processed.

- $S(x, y, n) =$ The intensity of the pixel at location (x, y) at frame n in scan S.

- $W =$ The width of the scan S.

- $H =$ The height of the scan S.

- $N =$ The number of frames in S.

- $f \in S =$ An H by W frame from S.

$$\forall f \in S, \ \forall f' \in S' \ : \ f' = f - (f * g) \tag{3.1}$$

Where g is an m-by-m averaging function defined as:

$$\forall x, y \ : \ g(x, y) = 1/m^2 \tag{3.2}$$

The second step is to adjust the grey levels and increase the contrast of the image such that 1% of the image is saturated at the highest and lowest intensities, in this case, black and white. The effect this has on the image can be seen in Figure 3.6.
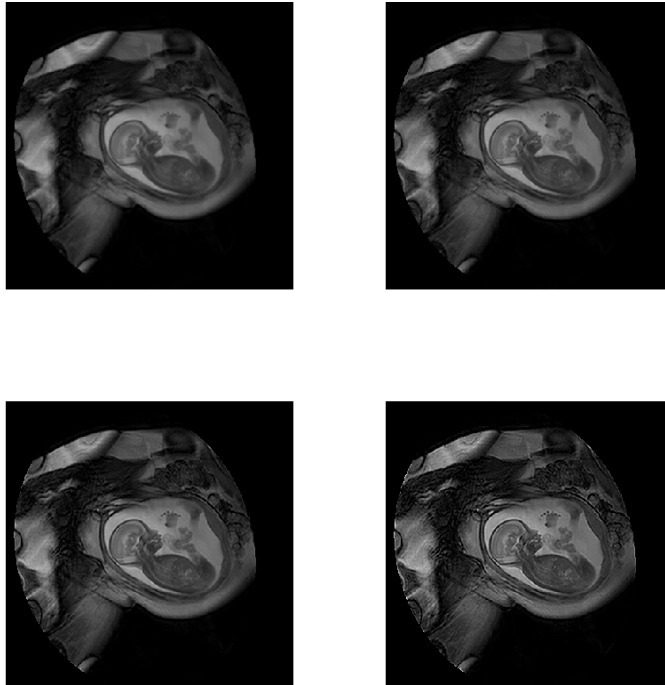
Figure 3.5: Results of repeated applications of sharpening, shown left-right, top-bottom. The first image being the raw unprocessed scan.
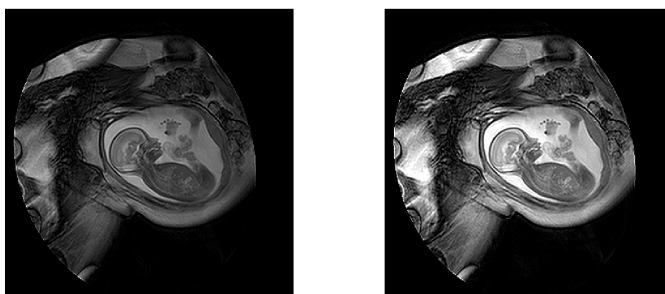
Figure 3.6: Results of enhancing contrast. The unprocessed image on the left, and adjusted image on the right.

3.4 Maternal Movement Suppression

Since most of the maternal movement is down to breathing, it is very periodic. Reasonably promising results have therefore been found using Fourier transforms to filter out maternal movement from the scans. If we consider the value of each pixel in every frame as a time series, we can visualise (Figure 3.7) and examine its properties. Contained somewhere in this time series is the effect of the maternal movement. It can be seen that there does appear to be a relatively constant oscillation. If we take the Fourier transform using Equation 3.3 of this time series we get Figure 3.8).

$$F(x, y, k) = \sum_{n=0}^{N-1} S(x, y, n)e^{-i2\pi k \frac{n}{N}} \qquad (3.3)$$
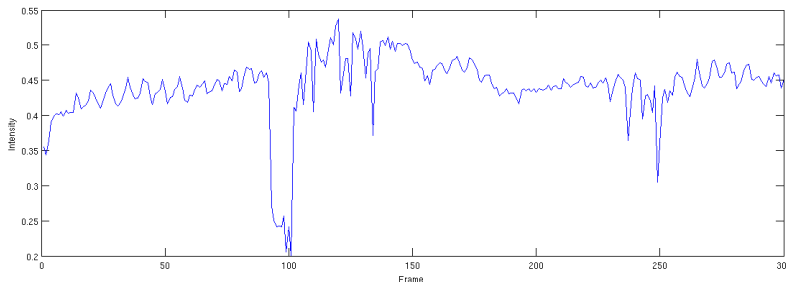
Figure 3.7: The intensity value of a single pixel taken over the course of a 300 frame scan.
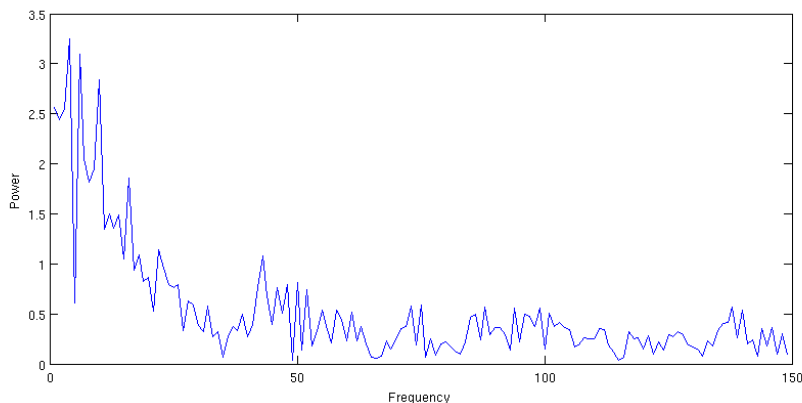
Figure 3.8: The result of a Fourier transform on the time series in Figure 3.7. Only frequencies greater than 0 have been plotted as the power of the 0 frequency is extremely high. Including the 0 frequency would make the properties of the rest of the graph unclear.

40

However, if we take the Fourier transform of a large number of pixels and average them, we get a clearer picture of what is happening. Figure 3.9 shows this calculated using a 10-by-10 grid of pixels using Equation 3.4.

$$M(k) = \frac{\sum_{x=1}^{n} \sum_{y=1}^{n} F(\frac{xW}{n}, \frac{yH}{n}, k)}{n^2} \qquad (3.4)$$

Where $F(x, y, k)$ is found using Equation 3.3 and n is the size of the grid.
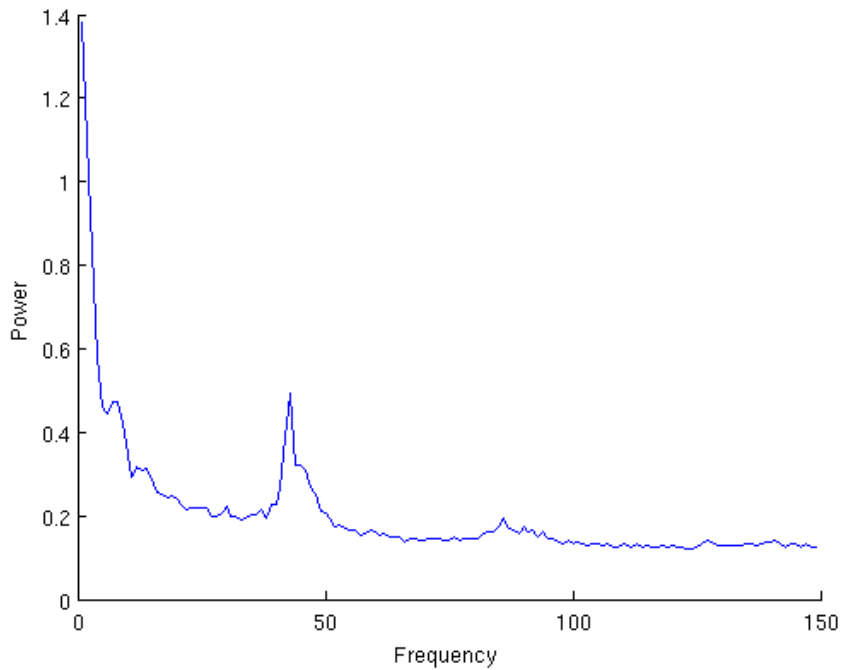
Figure 3.9: The result of averaging the Fourier transform taken from a 10x10 grid of pixels covering a frame. The 0th frequency has not been included, as the power is very high, which makes the properties of the rest of the graph unclear.

There are two clear peaks, with frequencies of approximately 45 and 90. Given the scan was taken at 3 frames per second, these frequencies match up well to the frequency of breathing (just over 2 seconds per breath), and a harmonic of this. This harmonic will be because the breathing isn't exactly sinusoidal. Every scan analysed in this way shows similar results. We can now filter out these two frequencies using Equation 3.5 (Figure 3.10) and apply an inverse Fourier transform (Equation 3.7) to rebuild the video. The peaks can be identified reasonably accurately by finding the peaks of a smoothed data series. The width of these peaks can be determined by

41

searching either side until the data flattens out. Filtering is done by multiplying by one minus a Gaussian window.

$$F'(x, y, k) = F(x, y, k) \times G(p_1, w_1, k) \times G(p_2, w_2, k) \tag{3.5}$$

$$G(p, w, k) = \begin{cases} 1 - e^{-0.5(\frac{k-p}{\frac{w}{2}})^2} & \text{if } (p - \frac{w}{2}) < k < (p + \frac{w}{2}) \\ 1 & otherwise \end{cases} \tag{3.6}$$

Where $F(x, y, k)$ is the Fourier transform of the pixel at location (x, y), p is the locations of the peaks located, w is the widths of these peaks and $F'(x, y, k)$ is the result of reducing the values around the peak frequencies to 0.

$$S'(x, y, n) = \frac{1}{n} \sum_{k=0}^{N-1} F'(x, y, k) e^{i2\pi n \frac{k}{N}} \tag{3.7}$$

Where $S'(x, y, n)$ is the new intensity value of the pixel located at position (x, y) at frame n and $F'(x, y, k)$ is found using Equation 3.5.
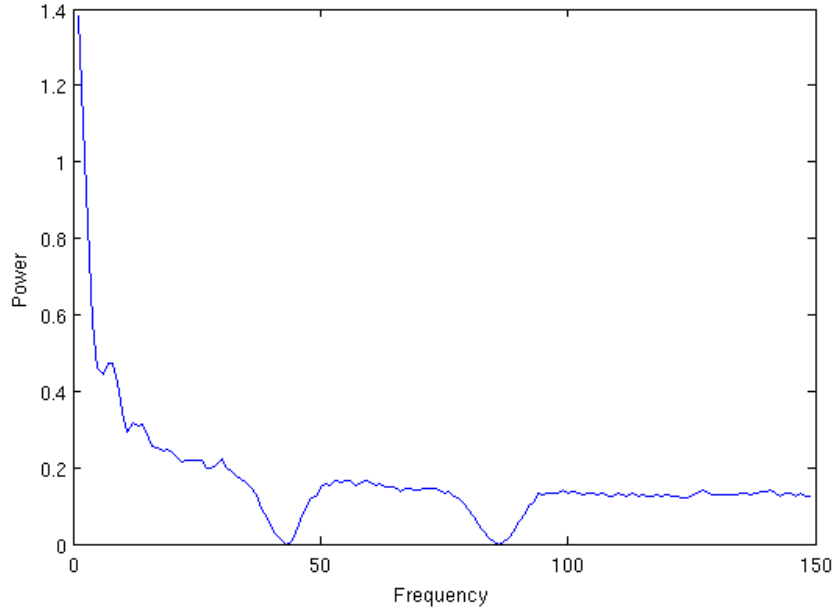
Figure 3.10: The result of filtering the the average Fourier transform in Figure 3.9. The 0th frequency has not been included, as the power is very high, which makes the properties of the rest of the graph unclear.

3.5 Crop

The cropping stage has two distinct parts. The first part tries to narrow down the area to a rectangle which roughly contains the uterus. The second stage narrows this area down to an arbitrary shape which attempts to match as closely as possible to the uterus. Whilst automatically cropping a scan may not necessarily help with the extraction of any useful information for the purposes of feature tracking or machine learning, it will help when it comes to the computational speed and memory requirements of such operations, as well as reduce the chance maternal tissue will be mistaken for the foetus.

Having used Fourier transforms to good effect to remove maternal breathing, it was possible to reuse the Fourier transform information already calculated to aid in cropping. By looking for areas of high frequency movement, it is possible to find an area which is likely to contain the foetus. This is because locations which exhibit high frequency movement generally correspond to the location of the foetus. By finding the pixels which contain high frequencies we can get a good idea as to where the foetus is. Figure 3.11 shows these areas.
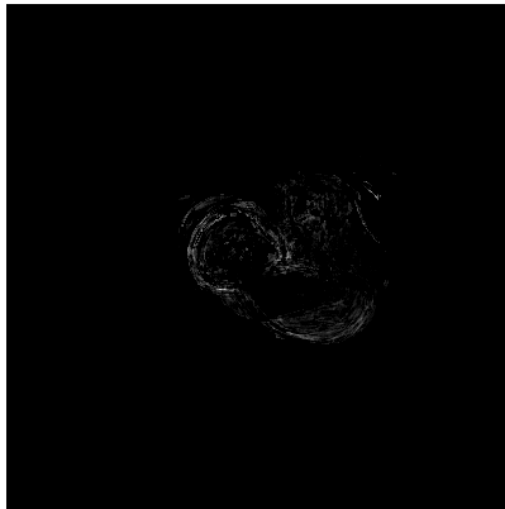
Figure 3.11: An image showing which areas exhibit high frequency movement, an indicator of foetal movement.

The area around the foetus is included in this area, however there are often other areas which also exhibit this kind of movement. By using the heuristic that the foetus tends to be away from the edges of the scan, we can reduce the search area further. Finally, a series of morphological operations

are performed to give us an estimation of the foetal area (Figure 3.12). The final result of this stage of cropping can be seen in Figure 3.13.
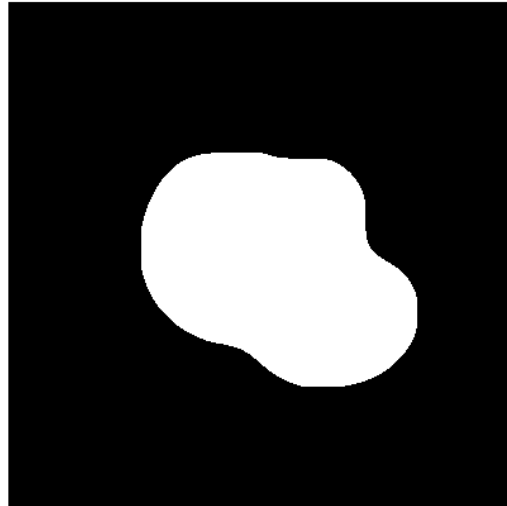
Figure 3.12: The result of a series of morphological operations on Figure 3.11.
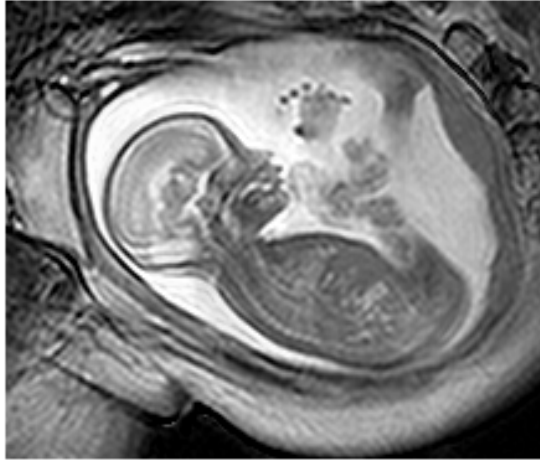
Figure 3.13: An example of the first stage of a successful cropping process.

The next stage aims to crop tighter around the uterus using foreground detection. This attempts to learn the background of a scene, and subtract it, to leave only the foreground. Whilst this method does not perform well enough to give us a complete segmentation for every frame, it does allow us to locate areas which are classified as foreground in a significant number of frames, and are therefore likely to belong to the foetus. Figure 3.14 shows an initial estimation for the location of the foetus. A series of morphological operations similar to those used in the previous cropping stage, give us a more robust estimate, which will form the mask to crop the scan around. The results of this can be seen in Figure 3.15.

Figure 3.14: An initial estimation for the location of the foetus using foreground detection.
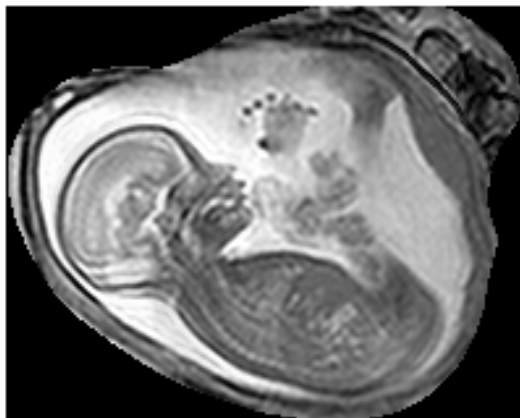
Figure 3.15: Cropping as a result of a series of morphological operations on Figure 3.14.

In most cases, this provides a reasonably good cropping. However, certain scans, particularly those with long periods of little to no movement, are sometimes cropped too zealously, resulting in a cropping around particular foetal features, see Figure 3.16.
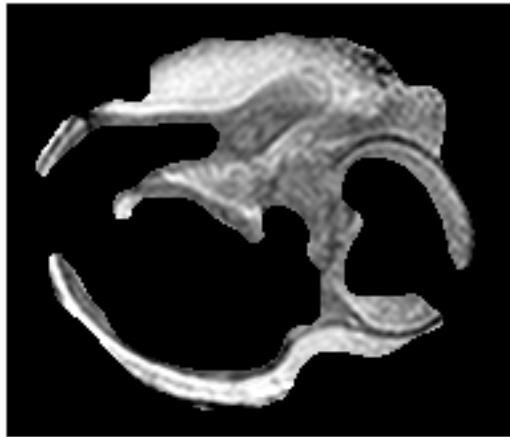
Figure 3.16: A poor initial estimation for the location of the foetus.

To overcome this issue, a function primarily used later in the pipeline is utilised, which attempts to segment the scan into periods of foetal activity, see section 3.7 for details on this algorithm. Creating a mask for each period of activity, and subsequently combining them allows us to eliminate the effects of long periods of foetal idleness and increase the probabilities of an accurate cropping.

Scans which do not exhibit enough movement to be cropped satisfactorily using these methods are left uncropped. This is detected by having too little high frequency movement to reach an initial estimate of the location of the foetus.

3.6   Motion Detection

Detecting the presence of foetal movement is a surprisingly difficult task. Even with an optical flow field for every frame, it is difficult to decide what corresponds to foetal movement and what is a result of maternal movement. The maternal breathing we attempted to suppress previously is not the only

maternal movement present in scans. There is occasionally additional movement detected as a result of movement in the mother's organs, or small shifts in the mother's position. The method used to try and identify the presence of foetal movement is to divide the scan into short periods and create an image by averaging the frames in each period. An example set of these images taken from a scan with foetal movement is shown in Figure 3.17, with an example set from a scan without foetal movement shown in Figure 3.18.
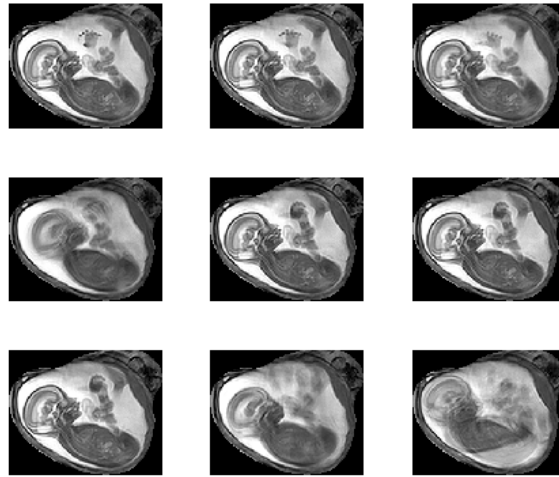
Figure 3.17: A set of images created by averaging short sections of a scan which exhibits movement.
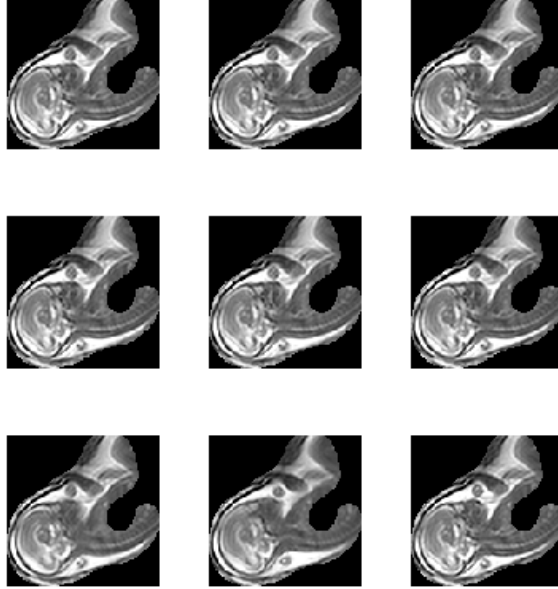
Figure 3.18: A set of images created by averaging short sections of a scan which does not exhibit movement.

As can be seen, in the case of the scan with no foetal movement, there is very little difference between each image, even though there is still maternal movement. Whereas, in the scan with foetal movement, there are noticeable differences between images. By identifying the presence of these differences, it is possible to decide if the scan contains foetal movement or not.

The technique used to identify these differences is to compare each image with every other image and calculate the total mean squared differences between the grey-scale intensity values between each pair of images. By taking the average of these differences and comparing them to a threshold found using trial and error, we can classify the scan as containing significant foetal movement or not - Equation 3.8

$$
has\_movement(S) = \begin{cases} True & \text{if } movement(S) > t \text{ where } q = sqd(S) \\ False & otherwise \end{cases}
$$

$$(3.8)$$

$$
movement(S) = \sum_{x=1}^{W} \sum_{y=1}^{H} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{f=1}^{\frac{N}{n}} (S(x, y\frac{iN}{n} + f) - \sum_{f=1}^{\frac{N}{n}} S(x, y, \frac{jN}{n} + f))^2
$$

$$(3.9)$$

49

Where n is the number of periods to divide the scan into.

3.7   Movement Signatures

The aim of this stage is to segment the scan into a number of shorter sections containing periods of foetal movement, called movement signatures. An observation from viewing a large number of scans is that foetal movement tends to be characterised by bursts of high activity, between periods of little or no activity. These periods can therefore be found by looking for unusually high bursts of movement. The first step to achieve this is to calculate a dense optical flow field at each frame, see section 2.3.1. It is then possible to work out how much movement there is between each frame by summing the magnitudes of all the vectors in the optical flow field. By plotting the amount of movement present in each frame over time, smoothing out local fluctuations using a moving average, and then zeroing every point beneath the mean, we arrive at a number of distinct periods. Another moving average connects nearby sections to each other, before the start and end point of each period is extracted and used to segment the scan.

Figures 3.19, 3.20, and 3.21 illustrate the full process on three separate scans.
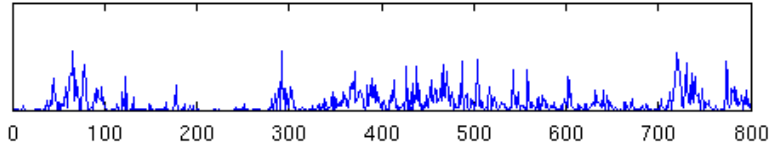
(a) Time series showing the sum of the magnitudes of all the movement located in each frame of a scan.
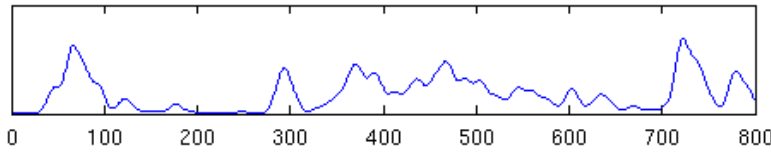
(b) The result of locally averaging the time series to reduce noise.
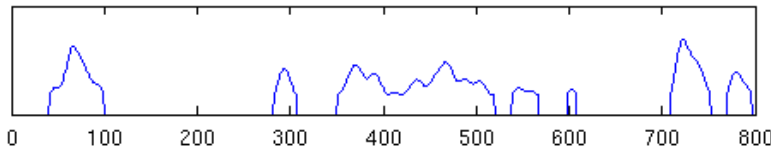
(c) The result of zeroing every point less than the mean in order to find periods of unusually high movement.
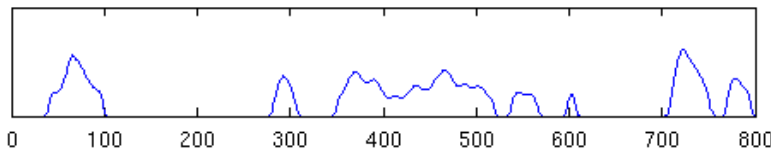
(d) The result of applying another local averaging, connecting nearby segments.

Figure 3.19: An illustration of the full process of segmenting a scan into foetal movements.
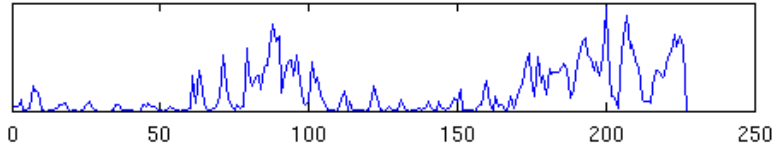
(a) Time series showing the sum of the magnitudes of all the movement located in each frame of a scan.
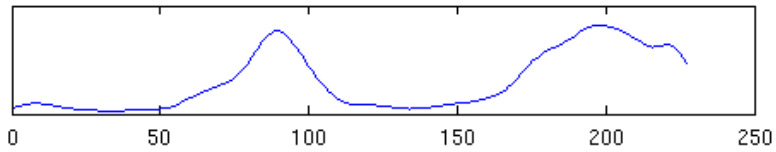
(b) The result of locally averaging the time series to reduce noise.
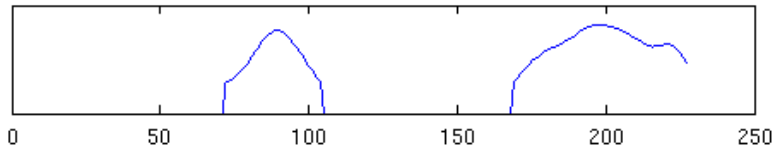
(c) The result of zeroing every point less than the mean in order to find periods of unusually high movement.
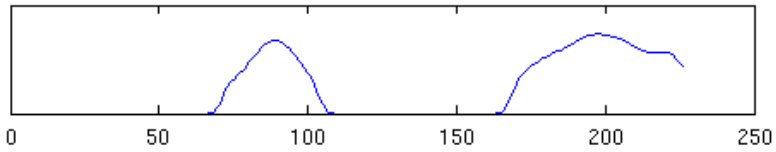
(d) The result of applying another local averaging, connecting nearby segments.

Figure 3.20: An illustration of the full process of segmenting a scan into foetal movements.
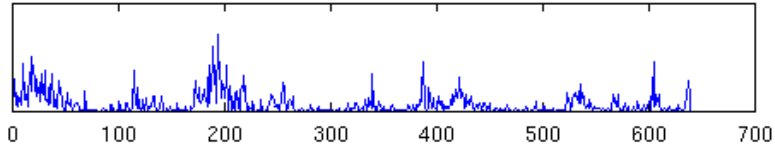
52

(a) Time series showing the sum of the magnitudes of all the movement located in each frame of a scan.
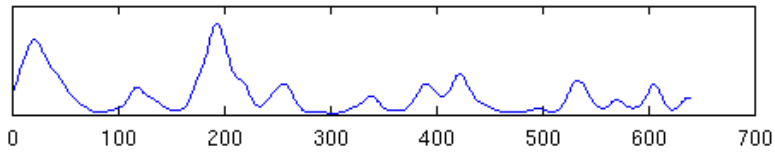
(b) The result of locally averaging the time series to reduce noise.
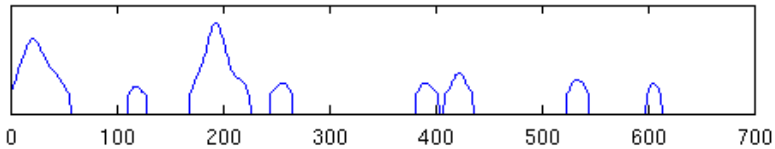
(c) The result of zeroing every point less than the mean in order to find periods of unusually high movement.
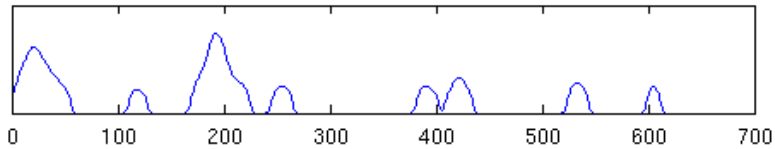
(d) The result of applying another local averaging, connecting nearby segments.

Figure 3.21: An illustration of the full process of segmenting a scan into foetal movements.

3.8   Identification of Body Parts

This stage involves spatially segmenting the previously extracted movement signatures into the individual parts of the foetus. The ideal outcome from this stage would be identify the locations of the head, abdomen, legs and arms of the foetus. This is, however, an extremely challenging proposition. We therefore just look for parts of the scan which could be part of the foetus. These sections, which we will call features, will later be classified in order to decide exactly what part of the foetus they belong to. For now, therefore, it is sufficient to identify areas which are likely to be part of the same part of the foetus, and we shall leave the job of deciding what part of the foetus this is for a later stage.

A clustering approach was due to the observation that areas which con-

sistently move at the same time, in the same direction and are close to each other are likely to form part of the same foetal body part. K-means (similar to K-Nearest-Neighbour, see Section 2.4.1) was chosen as the clustering algorithm, as the units along most dimensions are the same. The x and y position of each location is also included, which is multiplied by a factor which can be tuned in order to alter the effect of locality on the clustering.

A value of 4 was chosen for K to reflect the fact that, ideally, we would like the image to be clustered into head, body, arm and leg. It is rare to be able to see both arms or both legs at the same time in a scan, so we don't want separate clusters for each arm and leg. The algorithm is run on each movement signature, with each pixel as a data point, and the direction of the optical flow field at that point in each frame as the attributes, along with their x and y location. Since each period of movement tended to be between 10 and 100 frames long, this meant the problem is in 12-102 dimensions, and proves perfectly manageable.

Figure 3.22 shows the areas exacted using this technique. Once the general location of a feature is found through clustering, that area undergoes the second part of the cropping stage described in section 3.5. This will remove other, non moving, parts of the foetus which may be in the vicinity of the extracted area leaving just the feature. The effect of this can be see in Figure 3.23. Keep in mind that these are only the first frames of a period of movement, therefore the cropping must cover the whole range of positions the foetus takes in that period. Common features which are successfully identified using this method are the head, the upper back/neck, the lower back, the leg and the arm.
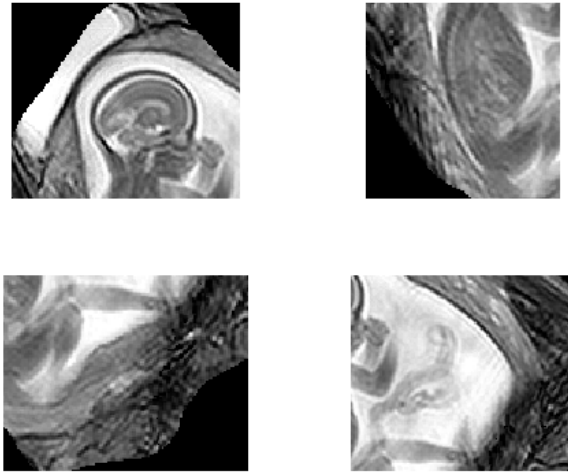
Figure 3.22: The results of clustering, identifying the rough locations of the head, body, legs and arms.
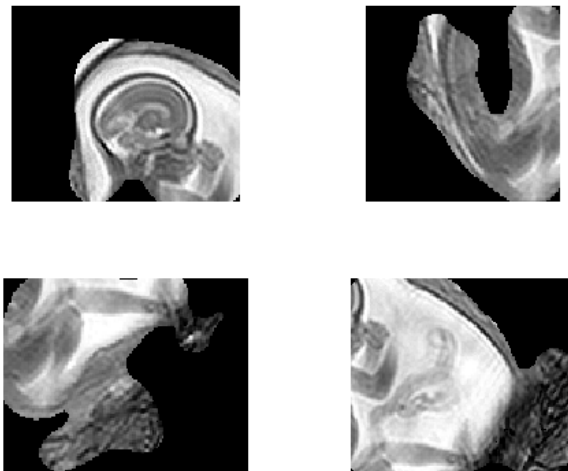
Figure 3.23: The results of cropping the locations in Figure 3.22 to cover the areas which contain movement. Note how maternal tissue is still present in the leg and arm images, and how the centre of the body is not included. These are both side effects of cropping to areas of movement. The intensity of the centre of the body is too uniform for movement to be detected, whereas the intensity of some the maternal tissue is very inconsistent, so movement is identified much more easily.

3.9 Initial Classifications

As alluded to in the previous section, the remaining stages must classify each of the extracted features according to the foetal body part they cover. A machine learning (see section 2.4) approach was chosen to solve this problem. The idea being that with enough examples of features taken from a large number of different foetuses, it will be possible to train a classifier to distinguish between them. However, in order for any machine learning system to learn to distinguish between different examples, it must be first trained on a number of labelled examples. This training data must come from manually labelling a large number of extracted features as being an example of either a head, leg, arm or body. In addition to these 4 categories, it was decided to also use label for maternal tissue. The reason for this is that some scans can still contain a large amount of maternal tissue which can, on occasion, be extracted as a feature. Since this will also be the case with future scans we wish to segment without manual labelling, it is important to accommodate these features, despite them being of little use to identify.

As one final piece of preprocessing, we segment the individual features into shorter movements using the same algorithm as used previously in Section 3.7. The reason for this was to try and get all the features to have a similar length. The movement profile of a 100 frame long series of kicks will look quite different to a shorter movement covering only a few frames. This gives us a much better chance of being able to classify features correctly.

Manually labelling features is an extremely labour intensive task, with nearly 3000 features extracted. The training data was chosen to include all the available scans for which movement was successfully detected to give the classifier as many examples as possible in order to help it discriminate between future examples.

It is now that we must consider what measurements we are going to use in order to give us the best chance of discriminating between different body parts. This is not a simple problem, extracting the most descriptive information, in a format which can easily be compared across different features. After experimenting with a number of different signatures and checking their ability to classify correctly, 4 different signatures were chosen, with the aim of combining the results from using each to classify a feature. Using multiple classifiers and aggregating the results is a common practice and proved very effective in this case. In particular, it was noticeable that certain signatures were better at discriminating between different body parts than others. The signatures used are summarised below. Examples of features, any intermediary steps, and signatures generated can be seen in Appendix 7.1.

- **Movement Weighted Histogram of Oriented Gradients**
  Histogram of Oriented Gradients (HoG) is a common technique used to generate a signature based on the edges of an image. Edge direc-

tions are calculated by applying a filter to the image. A histogram is then produced with a certain number of bins each corresponding to a range of angles. Each edge then votes for a bin in the histogram corresponding to its direction. This histogram can then be used as a signature to discriminate between different classes of image.

A modified version of this technique was used by weighting the contribution of each edge by the amount of movement observed at that edge. This has the effect of making edges corresponding to objects which move more have a greater impact on the histogram. A normalised histogram with 16 bins was chosen, and a certain degree of rotation invariance was added by aligning the direction corresponding to an angle of 0 degrees to the most highly represented edge direction.

- **Texture Information**
  Another signature found to be effective was a combination of measures relating to the texture of the feature. Each of:

  - Entropy
  - Contrast
  - Correlation
  - Energy
  - Homogeneity

  proved to be a useful measure to discriminate between classes. By normalising the ranges and combining the 5 measures into a single vector, signature could be created which showed considerable discriminatory power.

  Motion Histogram
  The third signature chosen was an overlapping motion histogram, as described towards the end of section 2.3.2. The only alteration made was to make the signature rotation invariant by aligning the 0 degree direction to the most common direction present in the histogram.

- **Fourier Descriptor of Movement Information**
  The final signature used attempts to discriminate using the pattern of movement in the feature by taking the magnitudes of the Fourier transform of the sum of the optical flow vector field for each frame. The values in radial sections of half of the image produced, each corresponding to 45 degrees, are then summed in order to form 8 parts of a signature. Only half the image needs to be used since the 2d Fourier Transform has two orders of rotational symmetry.

A K-Nearest-Neighbour Classifier (see 2.4.1) was ultimately chosen as the classifier used. Although other classifiers were tested, none had an accuracy significantly higher than KNN, and, very importantly, KNN allows us

to calculate not only the most likely class a feature may belong to, but also calculate a probability that a feature is of a particular class. This is done by looking at what proportion of the K closest matches belong to each class, and allocating a probability accordingly. This gives us much more flexibility, and the ability to refine and improve upon the initial classifications. A value of 10 was chosen for K by performing cross validation on the entire data set for values of K between 1 and 20 and choosing the value which gave the greatest accuracy.

The final probability distribution for each feature at the end of this stage is therefore a normalised sum of the probabilities calculated by the 4 classifiers.

3.10   Refine Classifications

So far, we have only used the signatures derived from the features as the means for classification. However we also have the additional information of the spacial location the feature was extracted from. Provided the foetus does not significantly change location over the course of a scan, which was never observed in any of the scans in the entire data set, we are able to use this information to improve upon our initial classifications. Since we are using a K-nearest-neighbour classifier, we can use the distribution of the labels of the closest examples as an approximation to the probability that that feature is a certain body part, see Figure(3.24).
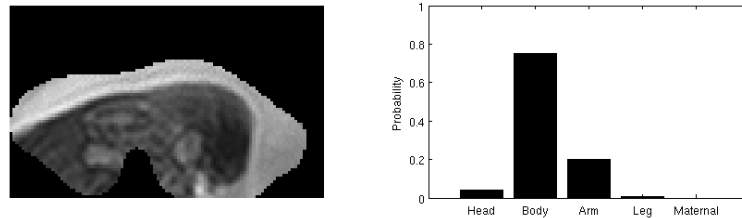
Figure 3.24: Left hand image shows a feature extracted from the scan shown in Figure 3.25. The chart on the right hand side shows the belief we have about that feature corresponding to a head, body, arm, leg or maternal tissue. As can be seen, the feature has been correctly identified as being most likely to correspond to a body.

This means we can create a number of heatmaps showing our belief that a particular location contains a particular body part, as seen in Figure 3.26, with the first frame of the scan shown for reference in Figure 3.25.
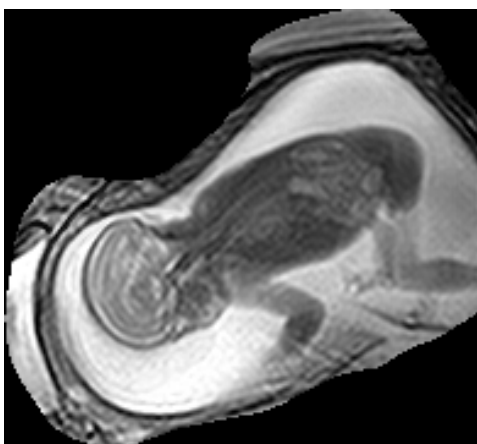
Figure 3.25: The first frame of the scan we are attempting to segment and classify. Note the location of the head, body, arm, leg and maternal tissue
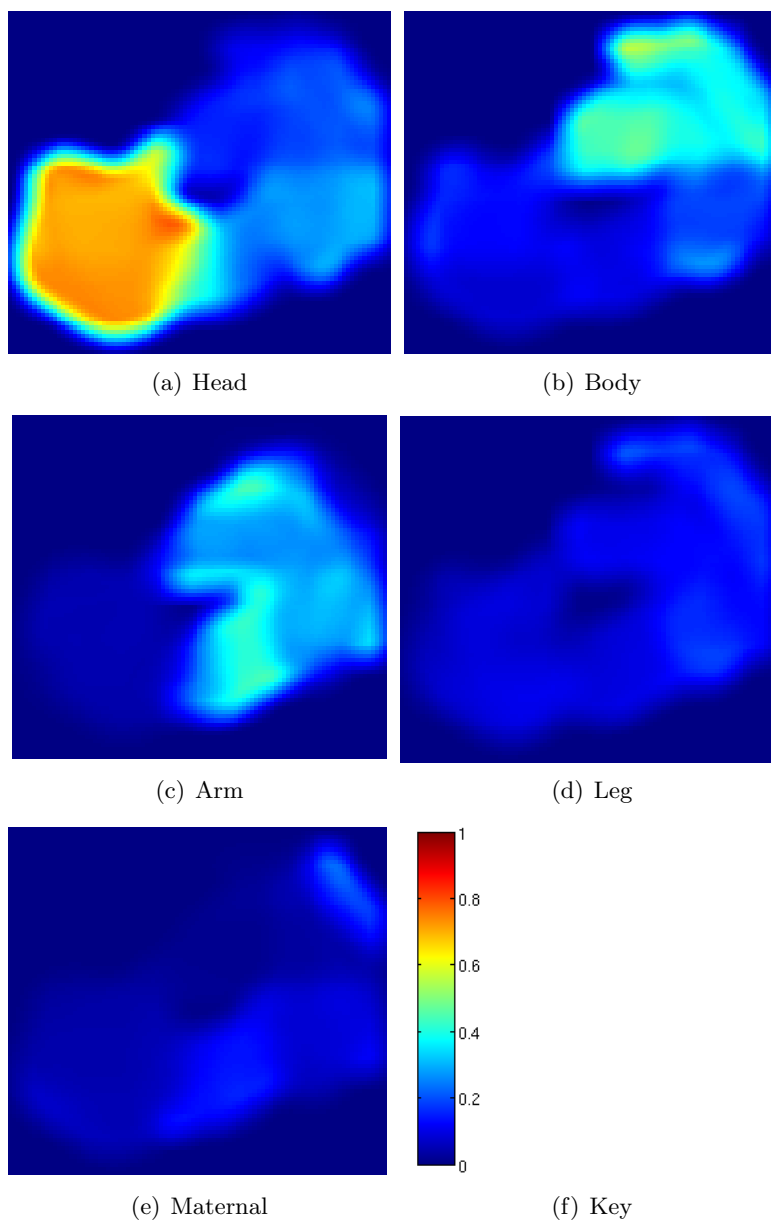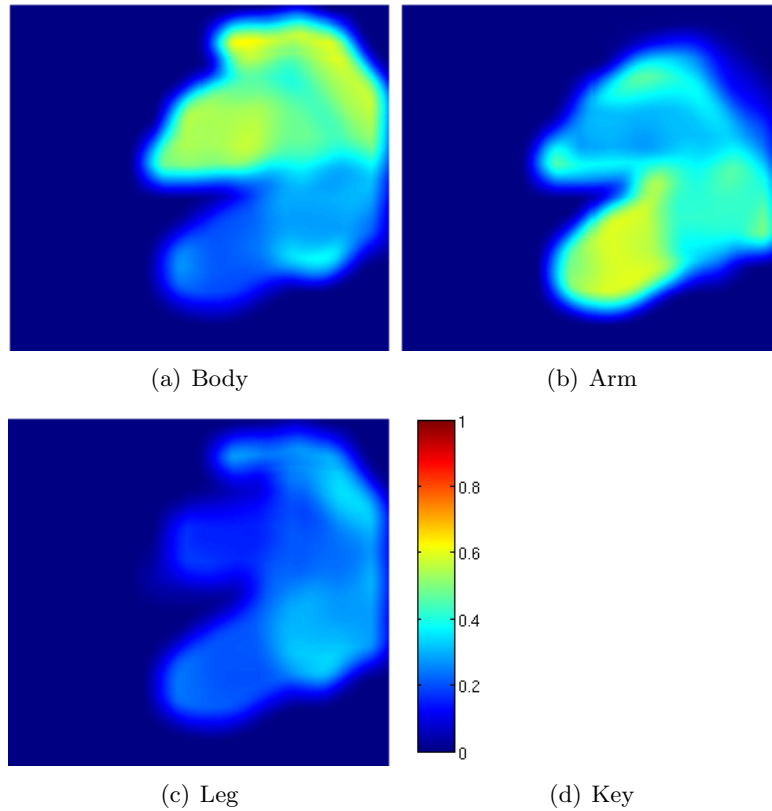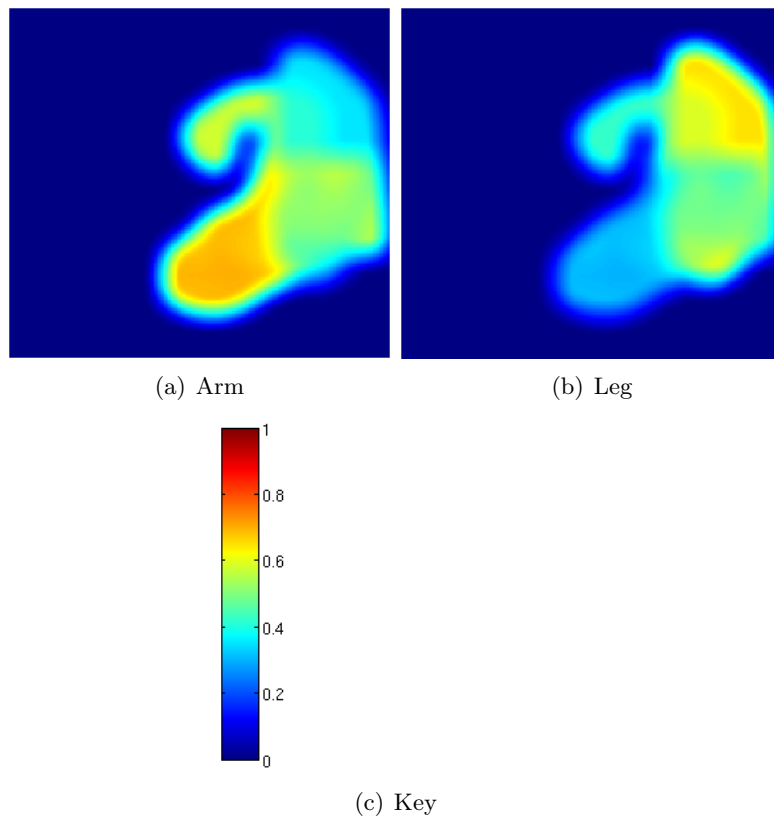
Figure 3.26: A series of heat maps indicating the probability that a certain location in the scan corresponds to a particular body part once maternal tissue has been removed.

Since there can be multiple locations with maternal movement, we first try and remove these areas. The first step is to find the locations which are more likely to be areas of maternal movement than any body part. We then look for features which occupy these areas and reclassify them as maternal tissue. This is done by looking at the area covered by each feature

in turn, and seeing if there is significant intersection with the locations we now believe to correspond to maternal tissue. If so, we reclassify the feature as maternal tissue, disregarding how it might have been initially classified. This process can be improved further by altering the thresholds used to indicate significant intersection to reflect our prior belief about the feature. For example, if a feature had an initial probability of being maternal tissue of 0.1, then it requires a greater degree of intersection with the area of suspected maternal tissue than a feature with an initial probability of 0.8.

We now assume we have correctly classified all the features corresponding to maternal tissue. This lets us remove all examples of maternal tissue from the training data, reclassify the remaining features, and generate new heat maps, seen in Figure 3.27.

(a) Head   (b) Body

(c) Arm   (d) Leg

(e) Key
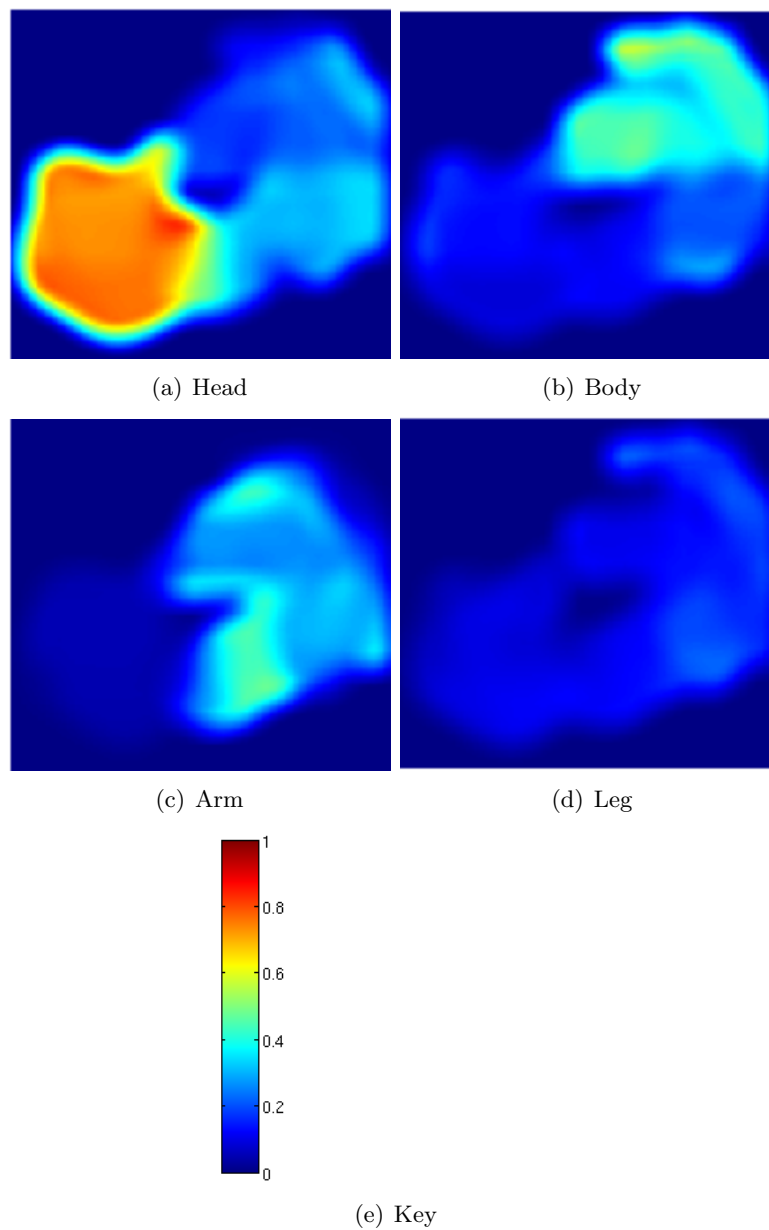
Figure 3.27: A series of heat maps indicating the probability that a certain location in the scan corresponds to a particular body part once maternal tissue has been removed.

As can be seen, the location of the head can be quite accurately found. We can therefore use the information that the head probably occupies this location throughout most of the scan and to reclassify the features which also occupy that area as heads.

We can continue this process, removing all examples of heads from the

training data and reclassifying the remaining features, giving us a new set of heatmaps, seen in Figure 3.28.

(a) Body

(b) Arm

(c) Leg

(d) Key

Figure 3.28: A series of heat maps indicating the probability that a certain location in the scan corresponds to a particular body part once maternal tissue has been removed.

Repeating this process for bodies we get Figure 3.29 showing our beliefs about the location of legs and arms. Using this we can reclassify the remaining scans.

(a) Arm

(b) Leg

(c) Key

Figure 3.29: A series of heat maps indicating the probability that a certain location in the scan corresponds to a particular body part once maternal tissue has been removed.

3.11   Visualise Classifications

Having achieved the main objective of the project, we looked ahead to the uses of this tool. One potential use would be to fully segment a scan so as to crop to each individual body part in every frame. Using the classifications, it is possible to locate and extract the areas of each body part. A prototype system was designed and produced providing a proof of concept. A single frame from a scan having undergone this process can be seen in Figure 3.30. Figure 3.31 shows a frame without colour coded classifications. The full videos can be viewed at `youtu.be/uBh4ngNJSvc` and `youtu.be/AiYaRIfWNNs` respectively.
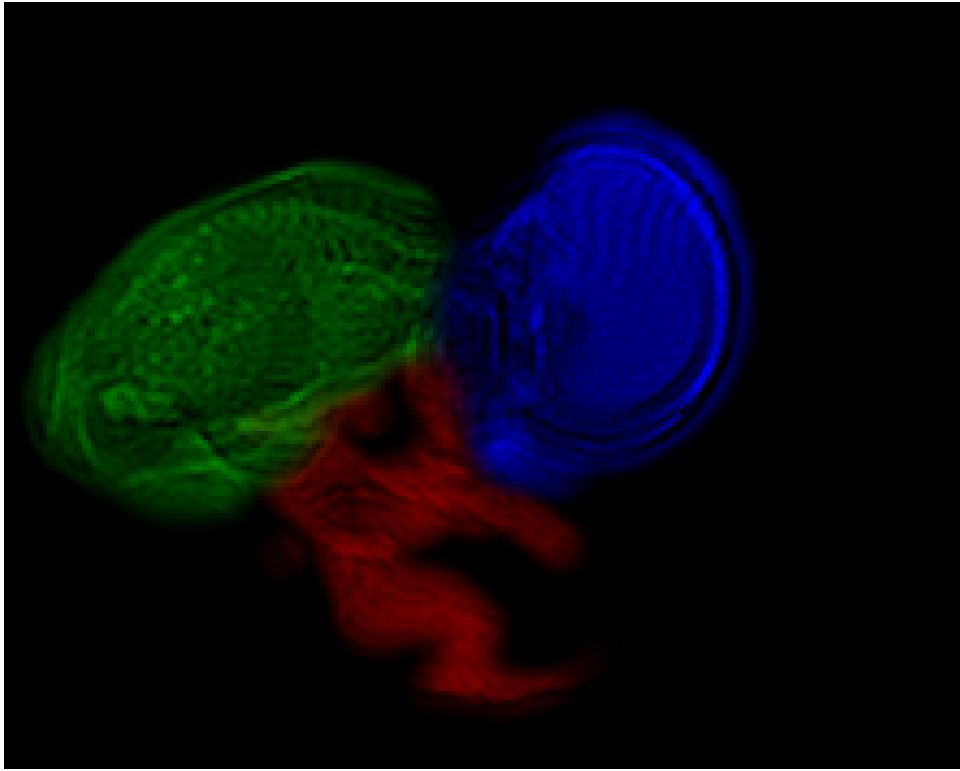
Figure 3.30: Single frame form a scan having had the location of the head (blue), body (green) and limbs(red) extracted.

Figure 3.31: Single frame form a scan having undergone segmentation.

A fully functioning system which can accurately perform this process on any scan would provide huge benefits to a number of interesting research topics. For example, a system which can consistently and accurately extract the limbs from a scan would allow significantly easier localisation and tracking of key points, such as joints. Another exciting potential application would be the ability to extract 3d models of the foetus. The scans used in this project are just one slice of a 4 dimensional volumetric scan. If this process was applied to each layer of a 4d scan, a model of the foetus could be produced. This would open up research into feotal movement in 3 dimensions.

The steps to locate the head and body are very similar, revolving around a series of morphological operations and edge detection. The general structure is described in Figure 3.32.

1: segment = find_initial_estimate(classifications)
2: mask(1) = segment
3: **for all** $i = 2$ to length($scan$) **do**
4:     frame = scan(i)
5:     segment = mask($i - 1$)
6:     local_mask = grow(segment)
7:     segment = shrink(segment)
8:     segment = segment $\parallel$ edge(f)
9:     segment = segment & local_mask
10:     segment = bridge(segment)
11:     segment = fill(segment)
12:     segment = remove_spur(segment)
13:     segment = morphological_opening(segment)
14:     mask(i) = segment
15: **end for**

Figure 3.32: The main stages of the algorithm which attempts to find and track the head and body.

The idea behind the algorithm is to exploit the generally clear edges around the foetus. The initial estimate for the location of the body part is found from the heat maps used to improve classification in Section 3.10. This provides the estimate for the position in the first frame. The estimate for each subsequent frame is found by first shrinking the estimate from the previous frame, and then growing it again to line up with nearby edges. Figure 3.33 shows this process visually.

(a) Shrunk location from previous frame.

(b) Nearby edges added.

(c) Close edges connected.

(d) Holes filled.

(e) Remaining edges removed.

(f) Mophological opening smooths edges.

Figure 3.33: An illustration of the full process of segmenting a scan into foetal movements.

- The first step is to take the segmentation from the previous frame and shrink it. This will move it away from the edges, and ensures every part is within the edges of foetus even in the presence of movement.

- The next step is to locate the edges present in the frame around the area of the current segmentation estimate using a Canny edge detection algorithm and to add these to the estimate.

- A bridging algorithm is then used to connect nearby unconnected edges.

- Any holes now present in the segmentation estimate are then filled.

- Any remaining edges are then removed using a spur removing algorithm.

- Finally, morphological opening is performed which removes small objects and smooths edges.

This process tends to find and match to the edges of the body part within 10 frames, and remain accurate throughout the remainder of the scan. As a result, the segmentation is poor for the first few frame. This can be solved by reversing the scan and performing the process again in reverse. By merging the two segmentations together we can improve on the segmentation in these early frames.

The algorithm for extracting the limbs is different. Limbs tend to move in and out of the plane of the scan, and therefore come and go during the

scan. They also tend to have less clearly defined edges. The approach described here makes no attempt to differentiate between arms and legs. A more sophisticated approach could attempt to do this.

The process revolves around segmentation using Otsu's method [35]. This attempts to segment an image by defining a threshold such so as to minimise the variance of the intensity values of the pixels greater than this threshold, and of those less than this threshold and is implemented in the Matlab function $graythresh(I)$. The main steps of the algorithm are described in Figure 3.34.

```
 1: segment = find_initial_estimate(classifications)
 2: mask(1) = segment
 3: for all i = 2 to length(scan) do
 4:    frame = scan(i)
 5:    segment = mask(i − 1)
 6:    local_mask = grow(segment)
 7:    threshold = graythresh(frame * local_mask)
 8:    segment = (frame * local_mask) < threshold
 9:    mask(i) = segment
10: end for
```

Figure 3.34: The main stages of the algorithm which attempts to find and track the limbs.

The idea here is similar to that of the algorithm for extracting the head and body.

- The first step is to take the segmentation from the previous frame and grow it. This ensures that the area still contains the limb if it has moved. This also has the effect of including areas of background.

- A threshold is found using Otsu's method looking at only the pixels covered by the grown estimate. Pixels greater than this threshold should correspond to background, while those less should correspond to the limb.

- The new segmentation is therefore found by selecting the pixels less than this threshold.

3.12 Feature Tracker

In addition to the main objective the project, some work was done on a feature tracker. The aim was to allow the user to select any point on the

scan, and for that part of the foetus to be tracked throughout the scan. The first attempt involved naively integrating the optical flow information, from one point, taken directly from Matlab's optical flow class. The result of this was pretty poor, with quite erratic behavior, possibly due to noise. The next step, was to take information from a larger window around the point, and average the resulting vectors. This produced slightly better results, but still not good enough for anything but the most simple cases. The tracker tended to lag behind and never truly stayed with the feature during large movements. Without any form of re-registration, any error in the tracking will never be recovered from, and will be compounded by future inaccuracies, leading to poor performance over a long time. A possible solution to this problem would be to use the optical flow estimations to guide a feature matching system to try and match sets of features in consecutive frames. This was only attempted by trying to directly line up edge information calculated using an edge detection algorithm. This simple approach actually reduced the accuracy of the feature tracking. If there was more time to work on this, the next step would be to try the same approach using a more sophisticated method of feature extraction and matching.

However, it was possible to improve further on the pure optical flow method. The problem with averaging the vectors from a large window is that if there are vectors pointing in opposite directions, they will cancel each other. This means the estimated movement vector will never be correct if it is being influenced by vectors which are pointing in different directions. The solution to this problem is to first decide what direction the feature has moved in, and then average the magnitudes of the vectors pointing in roughly that direction.

To implement this we must first assign each vector a number of "votes". Vectors towards the center of the window get more votes. This is because of the fact that the vectors from the pixels near the point we wish to track are likely to be more important than those near the edges. We then create two histograms, each with 8 bins. The bins correspond to different directions. The first histogram has bins corresponding to N, NE, E, SE, S, SW, W, NW directions. The second has bins corresponding to NNE, NEE, SEE, SSE, SSW, SWW, NWW, NNW. Each vector is then considered in turn and votes for one bin in each histogram, according to its direction. The bin is incremented by the number of votes allocated to that vector. The bin which has the most votes across the two histograms at the end of this process will correspond to the rough direction we expect the feature we wish to track to be moving in. A weighted average (according to the number of votes each vector was given) of the vectors which point in this direction is calculated, and forms the estimate we use for the movement of the feature.

The reason two histograms are used is because if the true direction of motion is roughly between two bins, the votes will be split between the two. In the best case, this makes our direction choice slightly out, in the worst

case, this splitting of the votes between two bins causes a third bin to be the biggest, and therefore chosen. Having two overlapping sets of bins avoids both of these problems.

Using this new technique, the feature tracker sticks with the feature for much longer and appears more robust.

As a side note, if we attempt to track a point in a scan which hasn't had the maternal movement removed, the feature tracker gets lost almost instantly. This is down to the large amount of maternal movement overriding everything else, so the feature tracker only responds to this.

3.13   Implementation Details

All of the work contained in this project has be created using Matlab. A UI, seen in Figure 3.35, was created to aid development, however the final process of classifying a scan is achieved by running a number of scripts. In order to prevent the need for a scan to be processed from the raw data every time it is used, various information about it is saved as it progresses through the pipeline. A version of the scan after maternal movement suppression is saved, along with the movement signatures and features extracted from these signatures together with their temporal and spatial locations. The collection of classification signatures for each scan are also saved on creation, along with a mask showing the locations of each feature. Finally, a collection of all the scans are saved together for use as training data. This final data structure also contains the number of features in each scan, to aid cross validation, as well as the masks and spatio-temporal locations of each feature, to allow us to generate the heat maps.
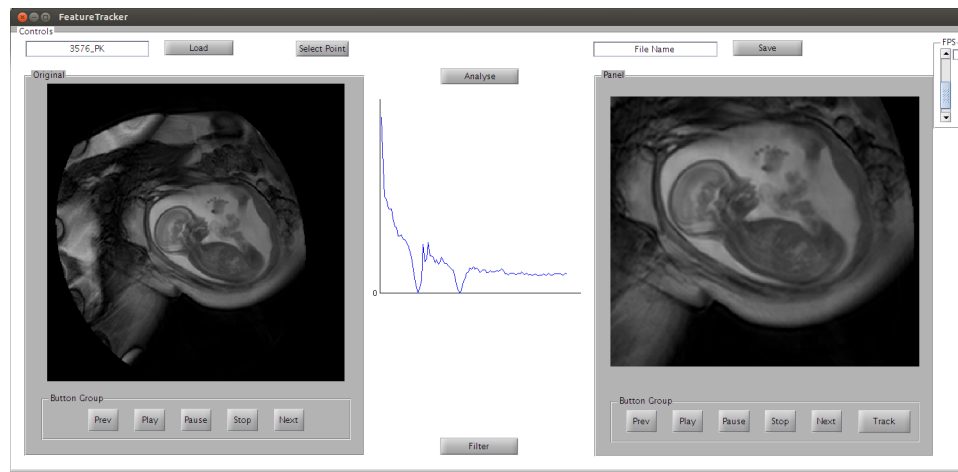
Figure 3.35: A screen-shot of the UI created to aid development

A large number of inbuilt Matlab functions were used throughout this

project, some of the most key functions are listed below.

- **imsharpen** from the Image Processing Toolbox, is used to implement the image sharpening method described in Equation 3.1.

- **imadjust** again from the Image Processing Toolbox, is used to enhance the contrast as required in Section 3.3.

- **fft** is used to calculate the Fourier transform as described in Equation 3.3

- **findpeaks** is used to locate peaks in the averaged Fourier transforms for Equation 3.5.

- **ForegroundDetector System object**, found in the Computer Vision System Toolbox is used for foreground detection in Section 3.5.

Chapter 4

Evaluation and Results

4.1 Raw Data Processing

The aim of these processes is to adjust the scans so as to improve the performance of the later stages in the pipeline, in particular those which use optical flow information, and edge detection. Therefore, a good measure of the effectiveness of this stage would be to examine the effect it has on the results of these operations.

The effect on the optical flow calculation can be seen in Figure 4.1. The intensity of the image reflects the amount of movement which has been detected in that area. As can be seen, the image corresponding to the processed scan has significantly brighter areas of movement. A comparison of the total amount of movement detected in each scan suggests that the preprocessing contributed to an increase in detected movement of 69%.
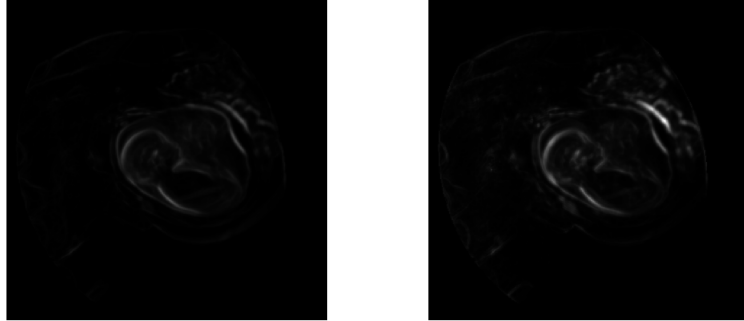
Figure 4.1: Total amount of movement detected. The scan on the left has not been preprocessed, wheras the scan on the right has been preprocessed to improve clarity.

The effect on the edge detection is harder to quantify in an objective way, however by examining images of the edges detected with and without the preprocessing steps suggests that more detail is identified in the preprocessed image. An example can be seen in Figure 4.2. The difference can be seen most strongly in the hand, where individual fingers are extracted in the preprocessed image, which are not in the original.
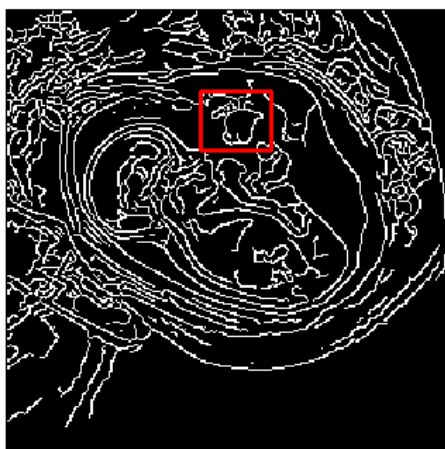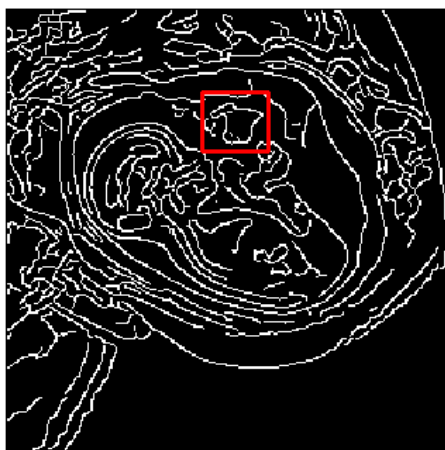
Figure 4.2: Edges found using a Canny edge detection algorithm. Cropped around the foetus for clarity. Top image is not preprecessed, bottom image has undergone preprocessing to improve clarity. Note the differences between the hands (indicated).

Whilst it is not obvious that these effects will have a direct impact on

the future processes, it is hard to argue that they are going to have a negative impact. The effect on both the movement and the edges detected suggest more information is being extracted from a preprocessed scan than an unprocessed one, and the ability to extract more information and detail is likely to help when it comes to discriminating between features at a later stage.

4.2 Maternal Movement Suppression

A visual analysis of the videos side by side show a significant reduction in maternal breathing, at some points, completely removing it. However, trying to apply the evaluation techniques mentioned in section 2.6 suggests there has been no reduction. Despite the clear reduction in movement visible in the filtered video, to the eye the average frames are identical. A comparison of the pixel intensities in the two images shows only the occasional and negligible difference which can be attributed to arithmetic error in the Fourier transforms. The reason for this comes from the fact that the average value of a single sin wave is 0, whatever the frequency. Therefore removing particular frequencies will not alter the average of the time series. The information in the frequency domain which corresponds to the average is held in the 0 frequency. This means that the average frame evaluation technique will not work when suppressing movement using Fourier transforms in this way.

As an alternative method, an optical flow field was computed and the average magnitude of the resulting vectors taken. The less movement there is, the lower this average should be.

The average movement vector magnitude in an unfiltered scan is 0.0079, whereas for a filtered scan, this number is 0.0056.

This is not enough to prove that the effects of maternal movement have been reduced, the loss in movement could have come from other places. However, combined with the observation that breathing does appear to have been reduced, and in some cases eliminated in filtered scans, we can safely say that this process does work, and is therefore a useful stage in the pipeline.

4.3 Crop

The most thorough objective method for evaluating the effectiveness of the cropping stage would be to manually define a perfect cropping for a number of scans, and calculate the differences between them and the automatic cropping reached using the method described. Unfortunately, manually defining a perfect cropping is quite labour intensive. Instead, we can subjectively approximate this method by examining the automatic cropping of each scan

and classify them to reflect the quality of the cropping around the foetus. The results of this analysis can be seen in table 4.1.

The ideal cropping would include the whole foetus and no maternal tissue. As it is more important to avoid losing foetal tissue than to eliminate all the maternal tissue, morphological operations were chosen which were more likely to lead to an over estimate of the foetal area than an underestimate. With this in mind, the cropping can be considered to have failed in just 19% of cases - Those with any loss of foetal tissue or when it failed to crop at all. Of the remaining 81%, the majority can be considered good, whilst the rest are acceptable.

No crop	2
Significant maternal tissue remaining - Figure 4.3	15
A small amount of maternal tissue remaining - Figure 4.4	28
Loss of a small amount of foetal tissue - Figure 4.5	3
Loss of a significant amount of foetal tissue - Figure 4.6	5

Table 4.1: Table describing the quality of cropped scans produced by the cropping system.
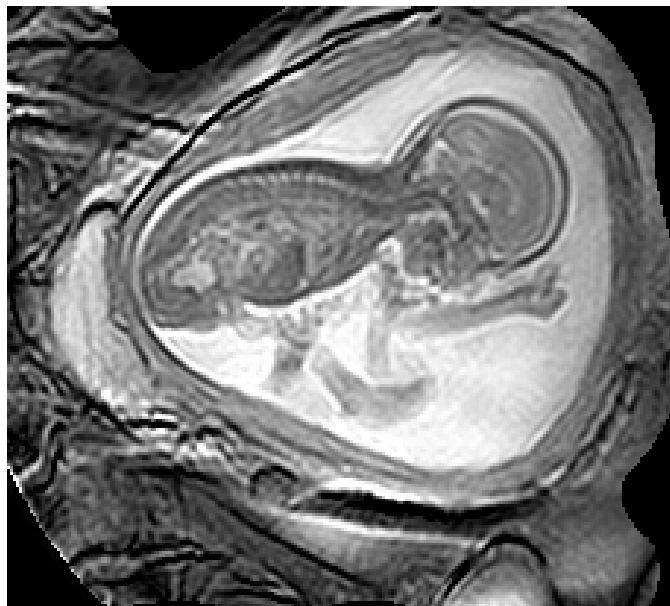
Figure 4.3: An example cropping with significant maternal tissue remaining.
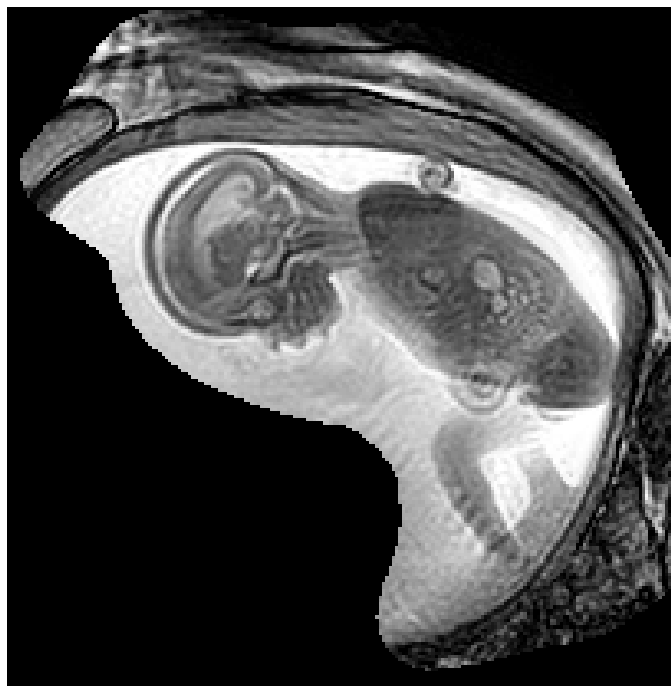
Figure 4.4: An example cropping with little maternal tissue remaining.

Figure 4.5: An example cropping with little loss of foetal tissue.
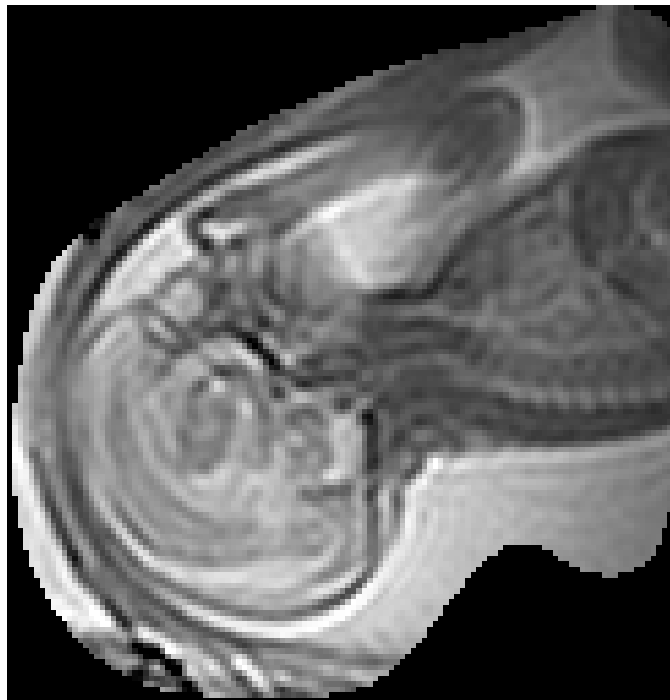
Figure 4.6: An example cropping with significant loss of foetal tissue.

4.4 Motion Detection

Since all the scans in the data set have been previously classified, evaluating the performance of the motion detection method is simply a matter of comparing the results to my manual classifications. A summary of this comparison can be seen in Table 4.2.

	Movement	No movement
Movement	53	3
No movement	14	19

Table 4.2: A confusion matrix showing the results of classifying scans according to whether there is any significant foetal movement. Rows correspond to the true classification while columns correspond to how they are classified using the technique described.

This gives us an accuracy of 81%. As can be seen from the Table 4.2, the majority of the inaccuracies come from incorrectly classifying scans containing no movement as having movement. We could adjust the threshold to try and improve on this, however that would likely result in a larger number of scans containing movement being classified as not containing movement.

The performance in this regard is very good, with a sensitivity to movement of 95%. Considering the system as a whole, we would rather a scan containing no movement to be classified as having movement than vice versa. We don't want to be filtering out perfectly valid scans, whereas allowing scans with no movement to progress to the later stages will not cause problems.

4.5 Movement Signatures

To evaluate the performance of this stage, a number of scans were selected at random and the results examined. We can treat the process as dividing the scan into examples of movement and no movement, which means we treat it as a classification problem. For each scan selected, every period it was divided into was examined and labelled as containing foetal movement or not. This labelling was then compared with the classification given and the confusion matrix shown in Table 4.3 could be created.

	Movement	No movement
Movement	56	5
No movement	16	47

Table 4.3: A confusion matrix showing the results of extracting movement signatures. Rows correspond to the desired classification while columns correspond to how they were classified.

These results give us an accuracy of 83%. Periods which contained foetal movement, but were mistakenly classified as not containing foetal movement tended to contain very short movements of a single body part, lasting two frames or less. Periods which did not contain foetal movement which were classified as containing foetal movement usually contained a global shift in the mother's position. These shifts, though short, cause a lot of movement to be detected, which is mistaken for a longer period of foetal movement.

4.6 Identification of Body Parts

The one major drawback to this approach is that parts of the foetus which don't move will not form part of a cluster. However this does not prove to be a significant omission, since there is usually enough movement to extract enough features to be able to locate the foetal body parts.

In order to evaluate the performance of this stage, a random subset of processed periods of movement have been examined along with the features extracted in order to ascertain what proportion of the foetus has been identified. It is easy to see what part of the foetus has been identified in each extracted feature. Since we are aiming to locate the head, body, legs and

arms, it is a simple task of seeing which of these have been identified in each period of movement. Table 4.4 shows the results of this analysis.

The results show that approximately 19% of body parts are not identified through clustering. The majority of these are arms at 32%. The most likely reason for a body part not to be identified is if there is insufficient movement to make it stand out from the background. The fact that arms are the most missed body part is not surprising, as it tends to be arms which exhibit the least movement.

		Present				Identified			
Scan	Segment	Head	Leg	Arm	Body	Head	Leg	Arm	Body
1	1	×	×	×	×	×	×	×	×
1	2	×	×	×	×	×	×	×	×
2	1	×	×	×	×	×	×	×	×
2	2	×	×	×	×	×	×	×	
2	3	×	×	×	×	×	×	×	
2	4	×	×	×	×	×	×	×	×
2	5	×	×	×	×	×			
2	6	×	×	×	×	×	×		×
3	1	×	×	×	×	×		×	×
4	1	×	×		×	×	×		×
4	2	×	×		×	×			
5	1	×	×	×	×	×	×	×	×
6	1	×	×	×	×	×		×	×
6	2	×	×	×	×	×		×	×
7	1	×	×	×	×	×			×
7	2	×	×	×	×	×	×		×
8	1	×	×		×	×	×		
9	1	×	×	×	×	×	×	×	×
9	2	×	×	×	×	×	×	×	×
9	3	×	×	×	×	×	×		×
9	4	×	×	×	×	×	×		×
9	5	×	×	×	×	×	×		×
9	6	×	×	×	×	×	×	×	×
9	7	×	×	×	×	×	×	×	×
9	8	×	×	×	×	×	×	×	×
9	9	×	×	×	×	×	×	×	×
9	10	×	×	×	×	×	×		×
10	1	×		×		×		×	
10	2	×		×		×		×	
10	3	×		×		×		×	
10	4	×		×		×		×	
10	5	×		×		×			
10	6	×		×		×		×	
	Total	34	28	31	28	34	21	21	22
	Percentage					100	75	68	79

Table 4.4: A table showing results of analysis of 10 randomly selected scans. Columns 3 to 6 indicate what body parts are visible in the scan. Columns 7 to 10 indicate what body parts were identified through clustering.

4.7 Initial Classifications

Combined results

There are many well established ways to measure the performance of a classifier, see section 2.6. It is important to remember that, as we are training on all of the available data, we must also ensure that when testing the performance of the classifier on one of these scans, the features from that scan are not included in the training data. If not, there will be examples of the feature we are looking to classify already labeled in the training data, thereby biasing the results considerably. By using leave-one-out cross validation, which involves testing performance on each scan in turn, using the remaining scans as training data, the following results were achieved. Table 4.5 shows the aggregated confusion matrix over the whole data set while Table 4.6 summaries a number of measurement statistics.

The high sensitivity and low specificity for locating heads is as a result of there being significantly more examples of heads in the data set than other features. This leads to head being treated as a default classification. However, these measurements are assuming each feature is assigned its most likely class as a result of classification. This will not be the case in the later stages of the pipeline, as each feature retains a certain probability to be of a particular class. The extremely low sensitivity for the body, arm, leg and maternal is also a result of this.

Another interesting observation is how there appears to be difficulty distinguishing between a body and a leg, reflected in both the confusion matrix and the precision metric. This as a result of the training labellings given to features, more than a reflection on the quality of the classifier. Due to the way clustering is performed - by looking for areas with common movement patterns - it is quite often the case that both the leg and lower back move at the same time, which causes the feature to cover both the leg and some of the body. Labeling such features is a challenge, and is taken on a case by case basis. The decision would be made on what label to give such features based upon what we would have ultimately liked the feature to be classified as, which tended to be whichever body part took up most of the feature. As a result, a number of features which contained both body and leg were classified as leg, and some were classified as body. This means that differentiating between features containing legs and bodies becomes more difficult. Fortunately, this will also not pose a great problem, since it is likely that the probabilities connected to such scans are high for both leg and body which, ultimately, is exactly how we would like a feature containing both head and body to be considered.

Finally, it is worth noting how the precision statistic corresponding to the classification of arms is significantly higher than in all of the four individual classifiers. This shows that by calculating a probability using each classifier

and then combining them, we are able to correctly classify features which we would otherwise have misclassified.

	Head	Body	Arm	Leg	Maternal
Head	908	56	13	32	40
Body	269	158	23	55	73
Arm	144	37	78	38	26
Leg	242	74	40	68	57
Maternal	83	69	4	25	163

Table 4.5: A confusion matrix showing the results of classification using all 4 signatures. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.87	0.57	0.55
Body	0.27	0.89	0.4
Arm	0.24	0.97	0.49
Leg	0.14	0.93	0.31
Maternal	0.47	0.92	0.45

Table 4.6: A summary of the statistics corresponding to classification using all 4 signatures.

Whilst an accuracy of just 47.2% is quite low, it is important to remember that these values have been calculated assuming each feature is classified as what is seen to be most likely. This figure will be improved upon in the refinement stage, when we use the probabilistic information to improve the classifications.

Another method of measuring the performance of a classifier is through a ROC curve. We consider each classification as a binary classification as either of a particular class, or not of that class and plot the true positive rate against the false positive rate as the probability required to be classified as that class ranges form 0 to 1. The area under the ROC curve (AUC) can be used as a measure of performance.

ROC curves for the five different classes can be seen in Figure 4.7.
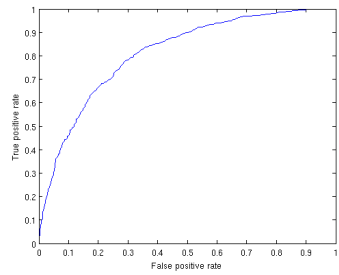
(a) ROC curve for head classification
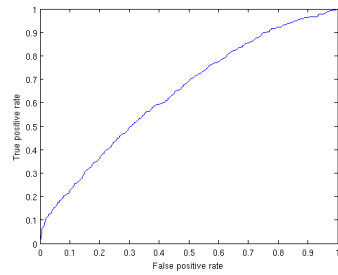
(b) ROC curve for body classification
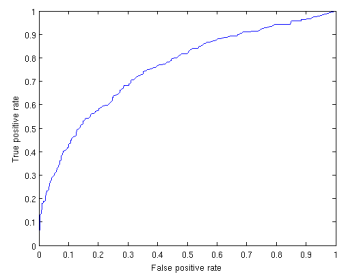
(c) ROC curve for arm classification
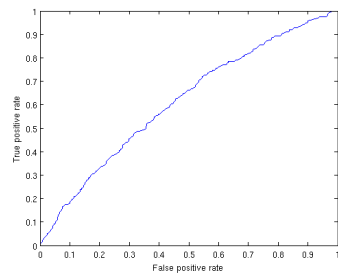
(d) ROC curve for leg classification
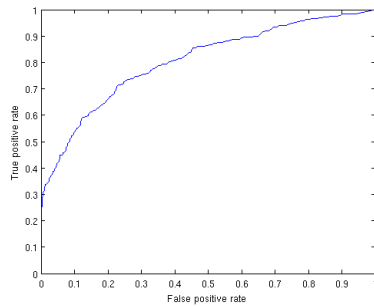
(e) ROC curve for maternal tissue classification

Figure 4.7: ROC curves for each class.

The AUC statistics for these curves are:

- **Head** - 0.81

- **Body** - 0.64

- **Arm** - 0.76

- **Leg** - 0.61

- **Maternal tissue** - 0.81

- **Average** - 0.73

These values reinforce our previous results, showing the classifier performs best when classifying heads and maternal tissue, and worst when classifying bodies and legs.

An interesting observation from the development of this classifier is how the classification rate increased as more data was added to the training set. By training on just a certain percentage of the training data, we are able to plot a graph of the relationship between the number of training examples and classification rate, seen in Figure 4.8.
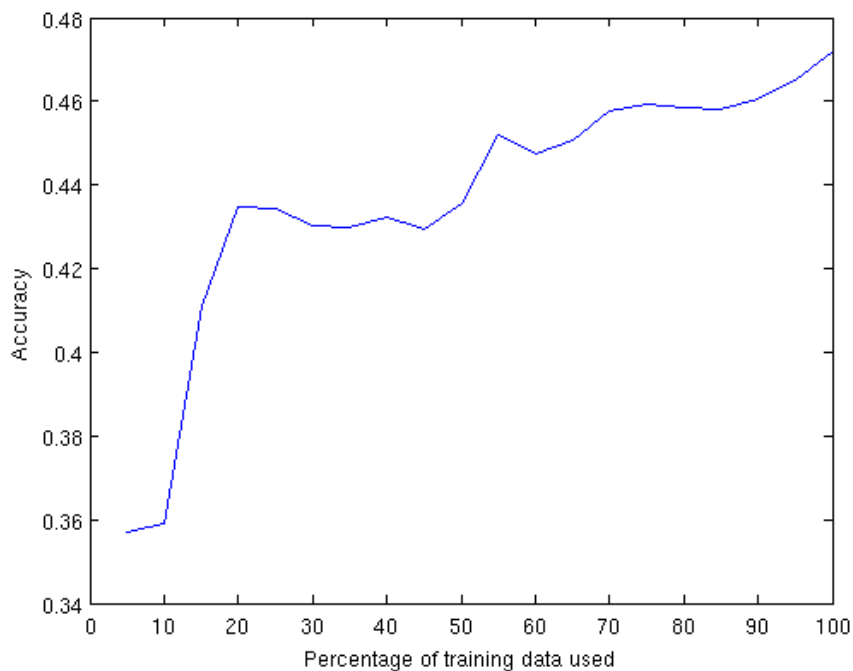
Figure 4.8: Learning curve showing the relationship between the percentage of training data used to train the classifiers and the accuracy of the resulting classifications averaged over each scan in the data set.

As expected, the more data in the training set, the better the accuracy of the resulting classifiers. However it is interesting to note that the performance is still improving up to the point at which all the data is used. This suggests that if we had more data to train on the accuracy could still increase.

Individual Signature Results

Evaluating the performance of the individual classifiers allows us to ensure that each signature provides good discriminatory power to aid classification.

- **Movement Weighted Histogram of Oriented Gradients**
  Performing initial classification using only this signature gives the results shown in Table 4.7 and Table 4.8. This corresponds to an accuracy of 35.0%. Performance appears to be best when it comes to identifying heads. This is perhaps not unexpected, as there are a larger number of example of heads in the training data than of other body parts. This also causes head to act as a default class should a feature be difficult to classify, and is the reason for the low specificity. There also appears to be difficulty distinguishing between legs and arms. Given this signature looks at the shape of the feature, it is again unsurprising that it struggles to discriminate between arms and legs, which often have a similar shape.

	Head	Body	Arm	Leg	Maternal
Head	633	145	42	70	159
Body	254	135	29	64	96
Arm	152	48	26	54	43
Leg	228	71	49	74	59
Maternal	143	66	11	15	109

Table 4.7: A confusion matrix showing the results of classification using the Movement Weighted Histogram of Oriented Gradients signature. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.6	0.55	0.45
Body	0.23	0.85	0.29
Arm	0.08	0.95	0.17
Leg	0.15	0.91	0.27
Maternal	0.32	0.85	0.23

Table 4.8: A summary of the statistics corresponding to classification using the Movement Weighted Histogram of Oriented Gradients signature.

- **Texture Information**
  Performing initial classification using only this signature gives the results shown in Table 4.9 and Table 4.10, corresponding to an accuracy of 44.5%. As can be seen, this signature looks to be very effective at identifying maternal tissue, and reasonably effective at identifying heads and arms. Unfortunately, it has poor performances when it comes to bodies and legs. If fact, there is little evidence to suggest that it can discriminate between them at all.

	Head	Body	Arm	Leg	Maternal
Head	810	126	35	56	22
Body	264	137	51	63	63
Arm	76	52	126	67	2
Leg	195	100	88	75	23
Maternal	65	95	14	28	142

Table 4.9: A confusion matrix showing the results of classification using the Texture Information signature. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.77	0.65	0.57
Body	0.24	0.83	0.27
Arm	0.39	0.92	0.4
Leg	0.16	0.91	0.26
Maternal	0.41	0.95	0.56

Table 4.10: A summary of the statistics corresponding to classification using the Texture Information signature.

- **Motion Histogram**
  Performing classification using only this signature gives the results shown in Tables 4.11 and 4.12, giving a 35.0% accuracy. This signature also suffers when it comes to discriminating between legs and bodies, as well as with maternal tissue.

	Head	Body	Arm	Leg	Maternal
Head	620	144	38	108	139
Body	247	106	26	86	113
Arm	122	54	34	42	71
Leg	192	96	19	72	102
Maternal	92	73	9	41	129

Table 4.11: A confusion matrix showing the results of classification using the Motion Histogram signature. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.59	0.62	0.49
Body	0.18	0.83	0.22
Arm	0.11	0.96	0.27
Leg	0.15	0.88	0.21
Maternal	0.38	0.83	0.23

Table 4.12: A summary of the statistics corresponding to classification using the Motion Histogram signature.

- **Fourier Descriptor of Movement Information**

  Performing classification using only this signature gives the results shown in Table 4.13 and Table 4.14. This gives us an accuracy of 39.8%. As can be seen from the precision statistics, the classifier also appears to work well when it comes to identifying heads and maternal tissue, but less well when discriminating between the other classes.

	Head	Body	Arm	Leg	Maternal
Head	757	111	69	79	33
Body	235	181	40	64	58
Arm	206	53	34	22	8
Leg	256	100	19	67	39
Maternal	90	76	9	52	117

Table 4.13: A confusion matrix showing the results of classification using the Fourier Descriptor of Movement Information signature. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.72	0.54	0.49
Body	0.31	0.85	0.35
Arm	0.11	0.94	0.2
Leg	0.14	0.91	0.24
Maternal	0.34	0.94	0.46

Table 4.14: A summary of the statistics corresponding to classification using the Fourier Descriptor of Movement Information signature.

4.8   Refine Classifications

Performance of this refinement stage can be measured in the same way as for the initial classification. Table 4.15 shows the aggregated confusion matrix over the whole data set, whilst Table 4.16 summaries a number of measurement statistics.

	Head	Body	Arm	Leg	Maternal
Head	894	61	21	59	24
Body	224	180	29	93	54
Arm	90	85	90	55	3
Leg	139	176	21	108	42
Maternal	42	109	8	29	156

Table 4.15: A confusion matrix showing the results of classification after refinement. Rows correspond to known values while columns correspond to how they are classified.

	Sensitivity	Specificity	Precision
Head	0.84	0.71	0.64
Body	0.31	0.81	0.29
Arm	0.28	0.97	0.53
Leg	0.22	0.9	0.31
Maternal	0.45	0.95	0.56

Table 4.16: A summary of the statistics corresponding to classification after refinement.

An accuracy of 48.5% suggests there has been very little improvement on the initial classification. Comparisons of results indicate that there may be small improvements in the classification of particular features, most notably legs, however the overall accuracy has hardly increased. As this is quite a small increase, despite a seemingly sound process, it is important to look beyond these statistics. In order to find out why the improvement is so small, a more in depth analysis of the results has been performed, looking at performances on individual scans. In particular, comparing the results of initial classification of the scan to the performance of the refinement process. Table 4.17 displays the performance for each of the scans in the data set, recording the accuracy of the initial classification, which stages of the refinement process were successful, and the accuracy after refinement.

		Located				
Scan	Initial Accuracy	Head	Body	Arm	Leg	Final Accuracy
1	0.94	×	×	×	×	1
2	0.8	×	×	×	×	0.88
3	0.36					0
4	0.31					0.19
5	0.22	×	×	×		0.59
6	0.52	×				0.57
7	0.52			×		0.24
8	0.74					0.74
9	0.31	×				0.27
10	0.68	×	×	×	×	0.82
11	0.66	×	×	×		0.81
12	0.34	×				0.34
13	0.25					0.16
14	0.77	×		×		0.79
15	0.82	×				0.74
16	0.79	×				0.70
17	0.52	×	×		×	0.71
18	0.3	×	×			0.44
19	0.2			×		0.29
20	0.37	×				0.34
21	0.18					0.25
22	0.42	×	×			0.51
23	0.83					0.75
24	0.31					0.31
25	0.3					0.14
26	0.64	×				0.64
27	0.25	×				0.28
28	0.16		×			0.25
29	0.52	×	×	×	×	0.63
30	0.68	×	×			0.84
31	0.3					0.23
32	0.4					0.3
33	0.31	×				0.35
34	0.33	×				0.33
35	0.37	×			×	0.63
36	0.6	×				0.66
37	0.51	×				0.69
38	0.59	×	×			0.5
39	0.63	×				0.46
Continued on next page						

Table 4.17 – continued from previous page

Scan	Initial Accuracy	Located				Final Accuracy
		Head	Body	Arm	Leg	
40	0.5	×	×	×		0.68
41	0.2					0.08
42	0.53	×				0.49
43	0.73	×	×	×		0.90
44	0.76	×	×	×		0.76
45	0.52			×		0.36
46	0.57					0.52
47	0.38	×				0.38
48	0.53	×				0.53
49	0.29			×		0.43
50	0.44	×				0.48
51	0.14					0.16
52	0.48	×				0.44
53	0.47					0.11

Table 4.17: A table showing results of more in depth analysis of the refinement process, recording the initial accuracy, what body parts were correctly located during the refinement process, and the post refinement accuracy. Note that scans 8, 23 and 46 are scans where cropping failed, and so contain mostly maternal tissue.

There are a number of observations we can make and conclusions we can draw from this information.

Firstly, as would be expected, the final accuracy depends a lot on both the initial accuracy, and the performance of the refinement process. The accuracy of scans in which three or four body parts were correctly located invariably improves through refinement, whilst the accuracy of scans in which no body parts were located nearly always reduced through refinement.

In fact, as long as the head is correctly located, the accuracy rarely gets worse, and only ever by a small amount. A closer examination of the scans in which the head is not correctly located will therefore provide a better understanding as to under what conditions the refinement fails. Scans 8, 23 and 46 retain a reasonably good accuracy due to the fact that they are mostly made up of maternal tissue. The average initial accuracy of the the remaining 15 scans is just 26%. It is a lot to ask for the refinement process to be successful when the initial classifications are so inaccurate. The fault therefore lies in the initial classification stage. An improvement here, perhaps as a result of more training data, or improved signatures, could have a large impact on the final accuracies.

The only scans which have a high initial accuracy and a significantly

reduced final accuracy are 7 and 53. In the case of scan 7, the the lower body of the foetus appears to have some of the characteristics of a head, see Figure 4.9. This leads to the head being mis-located, and hence the rest of the features are also misclassified.
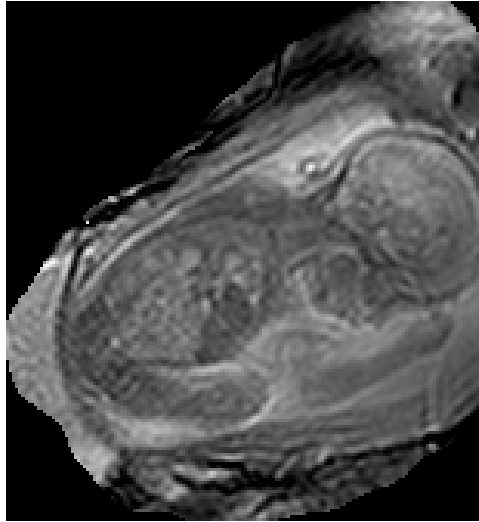
Figure 4.9: The first frame from scan 7. Note how the lower back / leg look similar to the head.

Scan 53 includes a lot of maternal tissue, but very little of it moves, so is not extracted as a feature. This means that the area which contains the foetus occupies a small part of the scan. One of the assumptions made when trying to locate the location of a body part from a heat map is that it is likely to be of a particular size relative to the scan. This assumption is correct in the vast majority of scans, and is necessary to produce an accurate estimate of the location, however in this case, it leads to the estimated location of the head to cover most of the foetus.

These cases go to show how difficult it is to design a system which will work well in all cases, and that often it is necessary to reduce the performance on a minority of cases, in order to increase it for the majority.

A potential source of inaccuracy remaining in the cases where refinement appears to be successful is from the labellings of the features. It is possible to argue for many features that they could be labelled differently, see Figure 4.10. In fact, it is likely that a fully successful refinement process, where each body part is correctly located, proves to be more accurate than the labels manually assigned to each feature.
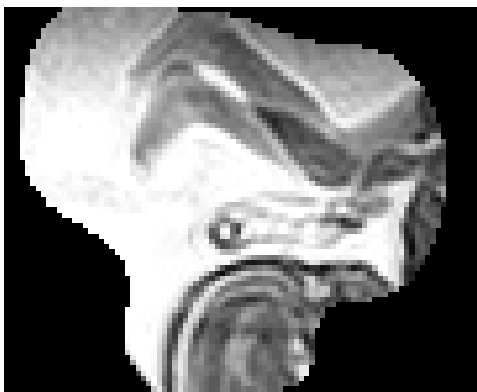
Figure 4.10: An extracted feature. Should this be labelled as an arm, leg, or head?

Chapter 5

Uses Cases and Future Work

Since cinematic MRI is such a new technology, there are a huge number of areas still to explore. The tool that has been created in this project can be the starting point for many different areas of research, a few of which are summarised below.

3D Segmentation

During this project, the scans used have been covering two dimensions + time. However, cinematic MRI scanners can record scans in three dimensions + time. Many of the techniques used can be expanded to cover three dimensions, potentially to the extent that an animated 3 dimensional model of the foetus can be automatically created. The potential uses of this for both diagnostic and research purposes would be huge.

Improved Feature Tracking

The feature tracker developed is very primitive, and has a lot of room for expansion and improvement. The utility of the tracker can be improved by tracking multiple points and automatically locating points to track. The actual tracking could be improved by using a feature extraction method to re-register the tracker and avoid drift over time.

Twins

In order to reach a reasonable classification rate, we had to make the assumption that there was only one foetus in the scan. If these technique were to be used on a scan containing twins the scan would have to be manually cropped to just one foetus at a time, else the results would be very poor. However, modifying the various algorithms with twins in mind would allow for a system with similar performance to be created.

Improved Classification Signatures

There is still room for the system developed to be improved. No stage of the pipeline yielded perfect results, and there is room for improvement in each. In particular, the signatures used to discriminate between different feature types could probably be improved. A higher initial classification rate could significantly improve the performance of the rest of the classification process. A significantly higher classification rate could reduce the need for further refinement entirely.

Other Areas of Medical Image Analysis

The techniques developed could, with sufficient alterations, be used in other areas of medical image analysis. For example, the technique used to reduce the effects of maternal breathing could well be used to reduce the effects of heartbeat in cardiac scans.

Automatic Recognition of Abnormal Foetal Movement

There is evidence to suggest that the type and extent of foetal movement can be used as an indicator of foetal health [4]. The system developed provides a perfect platform to investigate techniques to identify abnormal movement. One such method could be to create a timeline of movement throughout the scan and to use machine learning techniques to see if it is possible to discriminate between the timelines of scans containing normal and abnormal movements. This would allow us to see if there are certain combinations of movements which are indicative of abnormal movement. Another area to explore would be to look closer at the movement profiles of individual body parts and compare these between scans of normal and abnormal foetuses.

Still Foetuses

One of the major limitations of the system developed is that it requires a certain degree of foetal movement in order to segment and classify a scan. Similar techniques could be developed which attempt to use raw pixel intensity levels and pattens to differentiate between different body parts.

Chapter 6

Conclusion

In this project, we have shown that it is possible to correctly locate, extract and identify foetal body parts by using a mixture of traditional computer vision techniques and machine learning. We have produced a processing pipeline and, by analysing each stage individually, are able to say with confidence that it performs to a reasonable standard. As a result, we have opened up several areas of future research by automating procedures which would previously have been highly labour intensive. As well as this, we have shown that it is possible to extract a lot of information from foetal cinematic MRI scans. The work we have done could form the foundation of a broad investigation into many different uses of foetal cinematic MRI scans for both diagnostic purposes, and for better understanding of early human development.

We have also provided a number of techniques which could have additional applications across other areas of research. For example, the techniques used to suppress periodic movement and then identify periods of unusually activity, could perhaps have applications in identifying abnormal heart rhythms from a video. The use of HoG weighted by optical flow magnitudes to create a signature for a moving object could also have a wide range of applications in computer vision.

Finally, we have provided proofs of concept for a feature tracking tool and tool to either fully segment or extract individual body parts from a scan, which demonstrate some of the potential uses of our system. We were, however, unable to do any work regarding the bonus objective mentioned in Section 1.2.3.

6.1   Lessons Learnt

It has become clear to me over the course of this project that there is a lot of research potential in this area, and I feel I have barely scratched the surface of what could be a very important resource in the quest to better

understand human development. Hopefully what I have managed to achieve and the techniques I have used can aid future work in this area. I have demonstrated one of the potential uses of my work in the full segmentation of body parts, however there are many other areas which can be explored with the aid of this system, see Chapter 5 for some examples.

I would like to have been able to achieve more over the course of this project, there are so many interesting areas I would have liked to have looked at had I the time. However, initially I did not appreciate the challenges involved when working with these cinematic MRI scans and am ultimately pleased with what I achieved. The hardest challenge has been to deal with the huge variance between scans - See Figure 6.1. Each stage in the pipeline must attempt to handle all kinds of scan. Creating systems which generalise so well to different types of scan, yet are powerful enough to perform their desired function to the level required has proved to be very challenging and compromises have had to be made.
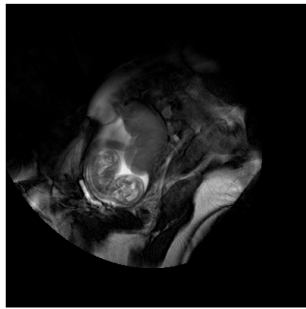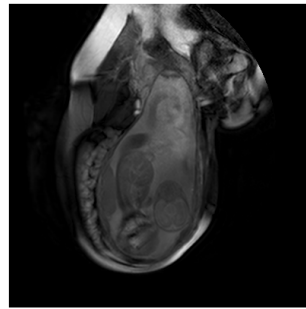
(a)   (b)
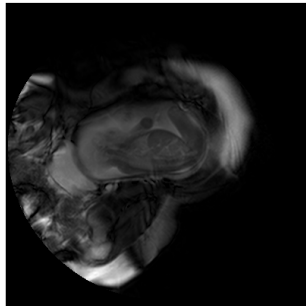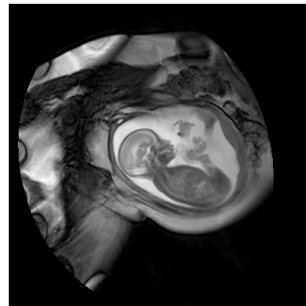
(c)   (d)
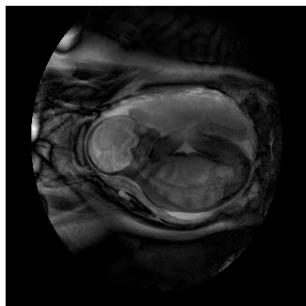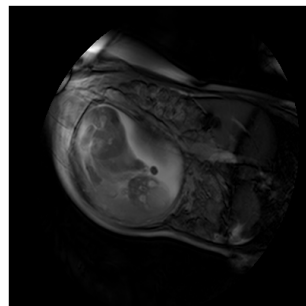
(e)   (f)

Figure 6.1: The first frames taken from a random sample of 6 scans. Note the variation in contrast, position, poise, size and the appearence and amount of maternal tissue.

Working with Matlab has definitely benefited me greatly over the course of the project. The large number of in built functions have allowed me to spend much more of my time investigating, developing and evaluating

different methods than I would have had if I had to implement all these functions myself. This means I have been able to progress much further and cover a large range of areas, despite the challenges involved.

Overall I feel the project has been a success. I have managed to achieve my initial objective, as well as a number of side objectives. I have produced a classification system which, whilst not perfect, performs significantly better than random, and demonstrated that it can be used effectively in practical situations. There are, however, a number of limitations which should be discussed.

Firstly, the number of scans from the data set which could be used in the system is quite low. This is mainly as a result of short scans being removed. This decision was taken because short scans tended not to exhibit enough movement to be successfully processed. An improved system may include special cases for shorter scans, and use different techniques to the movement based ones used in this system. Scans including twins were also removed from the data set as some techniques used assume, for example, that there will only be one head present in the scan. This could be overcome by manually partitioning the scan prior to processing, or though another stage of the pipeline which identifies the presence of twins and partitions accordingly.

The classification stage also had its limitations. There is a high variance in the performance of the initial classifier, which is made even larger through the refinement process. This means that while, on average, classification is relatively successful, in a significant number of cases it fails catastrophically. Before this tool can be used with confidence, these stages should be improved.

Bibliography

[1] Center for Disease Control. `http://www.cdc.gov/nchs/fastats/birth.htm`.

[2] L Rynn et al. Update on overall prevalence of major birth defects, morbidity and mortality weekly report (mmwr). Technical report, Center for Disease Control, 2008.

[3] Kitzman et al. Noninvasive whole-genome sequencing of a human fetus. *Science Translational Medicine*, 2012.

[4] Philip J. Dutton et al. Predictors of poor perinatal outcome following maternal perception of reduced fetal movements – a prospective cohort study. *PLOS One*, 2012.

[5] Robert G. Resta. The first prenatal diagnosis of a fetal abnormality. *Journal of Genetic Counseling*, Volume 6:81–84, 1997.

[6] National Health Service. `http://www.nhs.uk/conditions/amniocentesis/pages/introduction.aspx`.

[7] Medical Ultrasound Ltd. `http://www.medicalultrasound.co.uk/prices.html`.

[8] Private Healthcare UK. `http://www.privatehealth.co.uk/private-healthcare-services/diagnostic-imaging/mri-scans/mri-scan-prices/`.

[9] Verinata. `http://www.verinata.com/providers.html`.

[10] RJP Rijnders, RB Van Der Luijt, EDJ Peters, JK Goeree, CE Van Der Schoot, JK Ploos Van Amstel, and GCML Christiaens. Earliest gestational age for fetal sexing in cell-free maternal plasma. *Prenatal diagnosis*, 23(13):1042–1044, 2003.

[11] Indira Jaising. *Pre-conception and pre-natal diagnostic techniques act.* Universal Law Publishing, 2004.

[12] Anette Kersting et al. Psychological impact on women after second and third trimester termination of pregnancy due to fetal anomalies versus women after preterm birth—a 14-month follow up study. *Archives of Women's Mental Health*, 12:193–201, August 2009.

[13] Vatican Congregation for the Doctrine of the Faith's document Donum Vitae (1987).

[14] Ibrahim A. Alorainy et al. Attitude towards mri safety during pregnancy. Technical report, Ann Saudi Med, 2006.

[15] Martin Uecker, Shuo Zhang, Dirk Voit, Alexander Karaus, Klaus-Dietmar Merboldt, and Jens Frahm. Real-time mri at a resolution of 20 ms. *NMR in Biomedicine*, 23(8):986–994, 2010.

[16] Norbert J Pelc, Robert J Herfkens, Ann Shimakawa, and Dieter R Enzmann. Phase contrast cine magnetic resonance imaging. *Magnetic resonance quarterly*, 7(4):229–254, 1991.

[17] Christian Bamberg, Grit Rademacher, Felix Güttler, Ulf Teichgräber, Malte Cremer, Christoph Bührer, Claudia Spies, Larry Hinkson, Wolfgang Henrich, Karim D. Kalache, and Joachim W. Dudenhausen. Human birth observed in real-time open magnetic resonance imaging. *American Journal of Obstetrics and Gynecology*, 206(6):505.e1 – 505.e6, 2012.

[18] Niamh C Nowlan, Paula Murphy, and Patrick J Prendergast. Mechanobiology of embryonic limb development. *Annals of the New York Academy of Sciences*, 1101(1):389–411, 2007.

[19] A Hosseini and DA Hogg. The effects of paralysis on skeletal development in the chick embryo. i. general effects. *Journal of anatomy*, 177:159, 1991.

[20] Rami Ben-Ari and Nir Sochen. A general framework and new alignment criterion for dense optical flow. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 529–536. IEEE, 2006.

[21] Levi Valgaerts. Combining variational and feature-based methods for motion estimation. Master's thesis, Bavarian Graduate School of Computational Engineering, 2007.

[22] Matlab Comnputer Vision System Toolbox OpticalFlow. http://www.mathworks.co.uk/help/vision/ref/vision.opticalflowclass.html.

[23] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1):185–203, 1981.

[24] Jean-Yves Bouguet. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5, 2001.

[25] James Davis. Recognizing movement using motion histograms. *Technial Report 487, MIT Media Lab*, 1(487), 1999.

[26] James Davis. Recognizing movement using motion histograms. *Technial Report 487, MIT Media Lab*, 1(487), 1999.

[27] Tom M Mitchell. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, 45, 1997.

[28] Maja Pantic and Stephen Muggleton. Imperial College Department of Computing Machine Learning Course notes.

[29] Aldo Faisal. Imperial College Department of Computing Machine Learning and Neural Computation Course notes.

[30] Stefanos Zafeiriou. Imperial College Department of Computing Advanced Statistical Machine Learning and Pattern Recognition Course notes.

[31] Duncan Gilles. Imperial College Department of Computing Intelligent Data and Probablistic Inference Course notes.

[32] Aldo Faisal. Imperial College Department of Computing Machine Learning and Neural Computation Course notes. Lecture 5. Page 26.

[33] Prof Guang-Zhong Yang. Imperial College Department of Computing Computer Vision Course notes. Lecture 5.

[34] Oliver Wilkie. Motion stabilisation for video sequences. Master's thesis, Imperial College London, June 2013.

[35] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

Chapter 7

Appendix

7.1 Signature examples

7.1.1 Movement Weighted Histogram of Oriented Gradients

Format: Feature — Edges — Weights — Signature
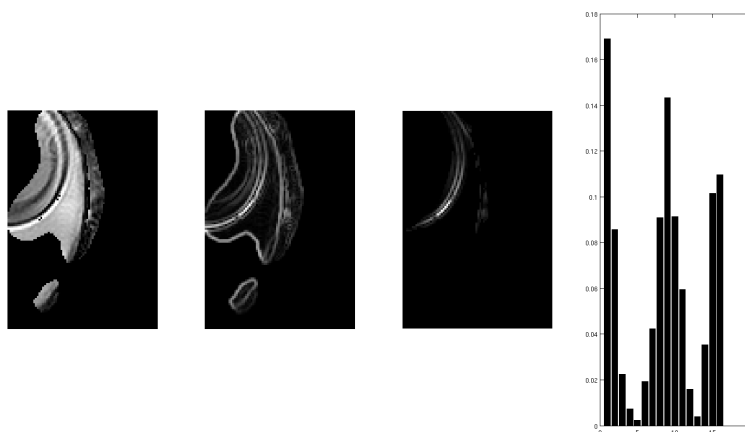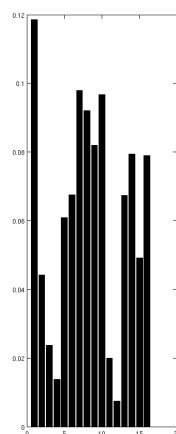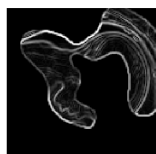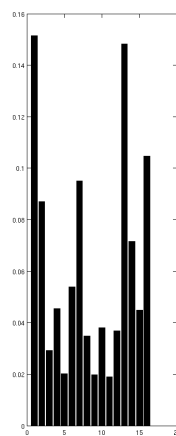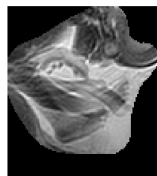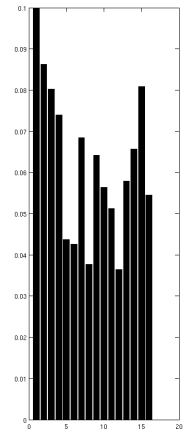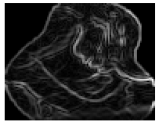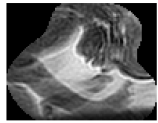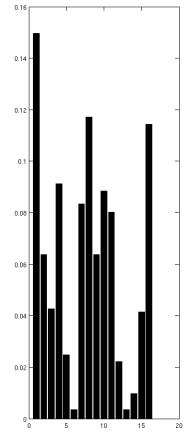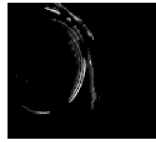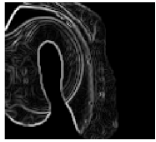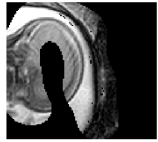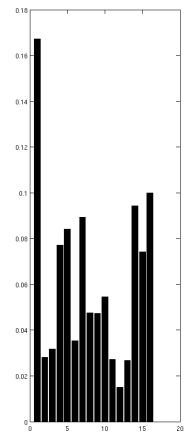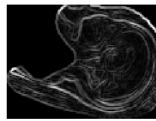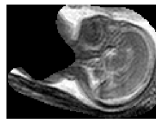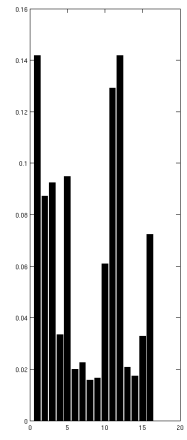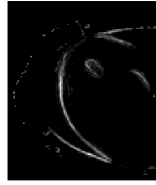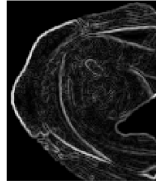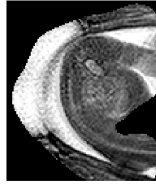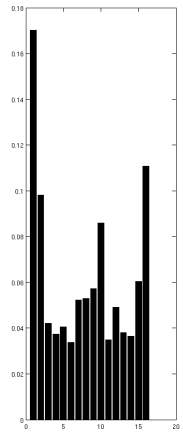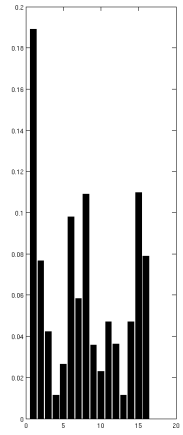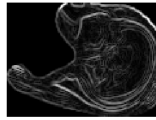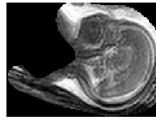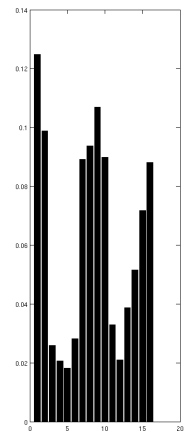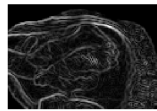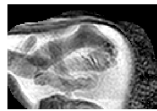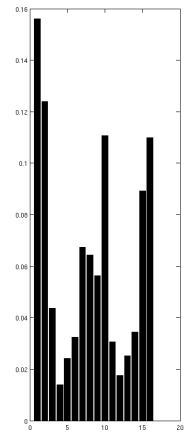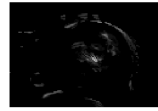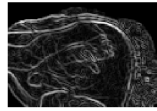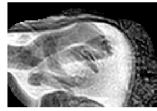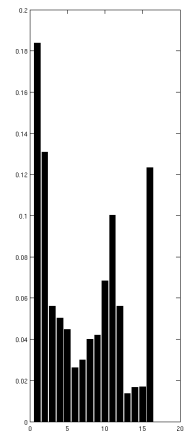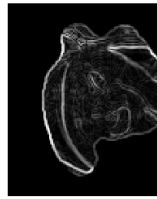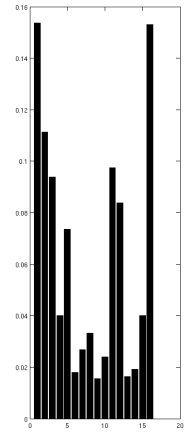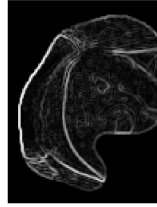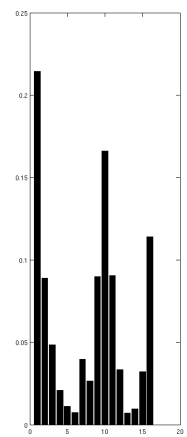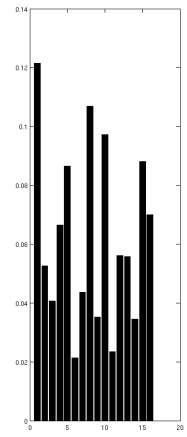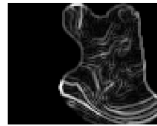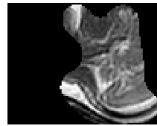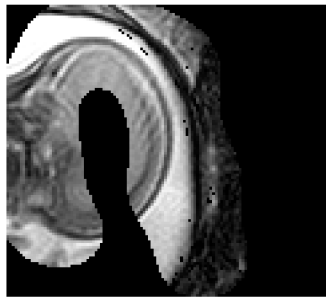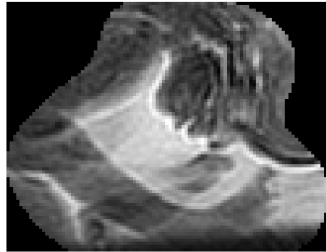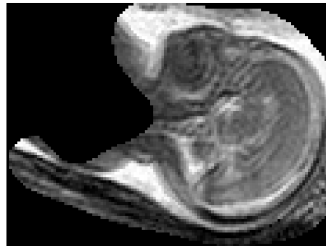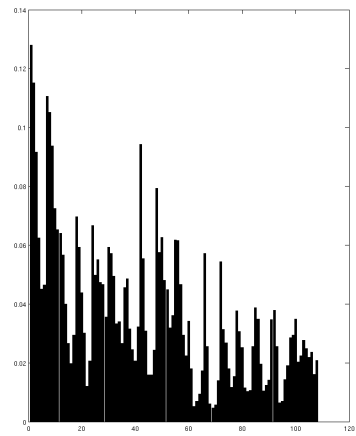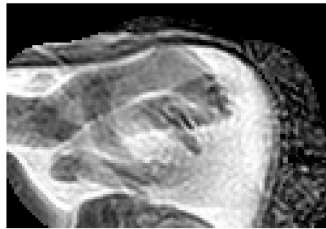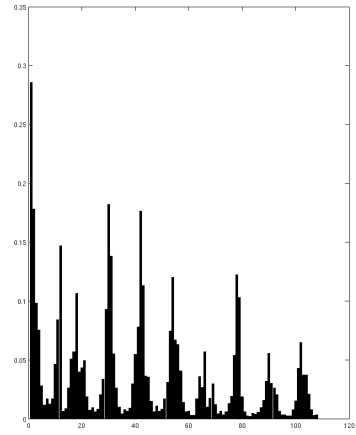
7.1.2 Texture Information

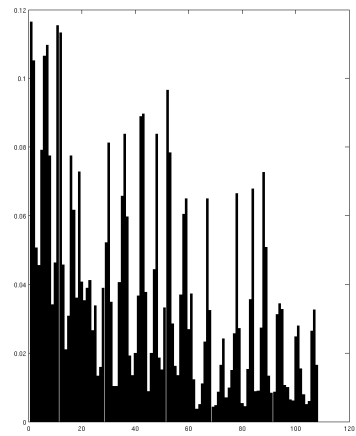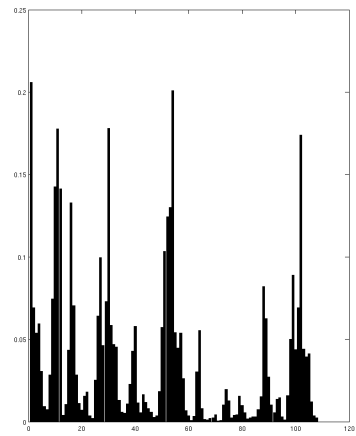Format: Feature — Signature
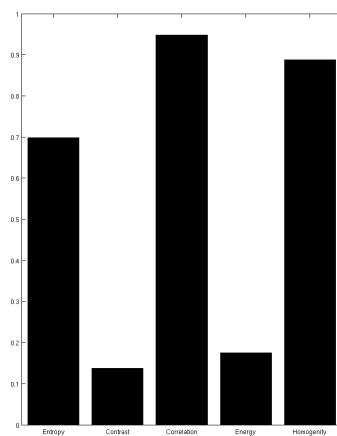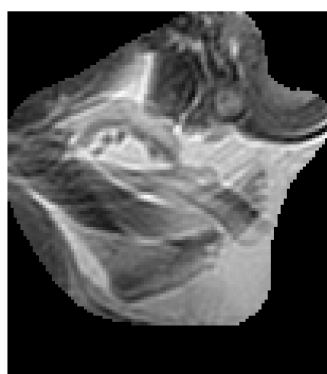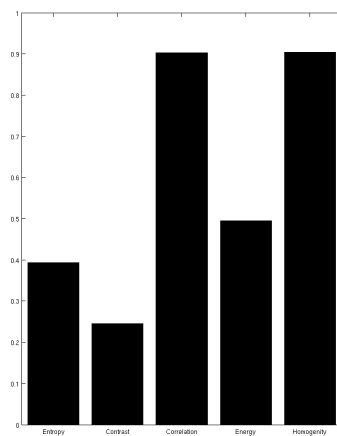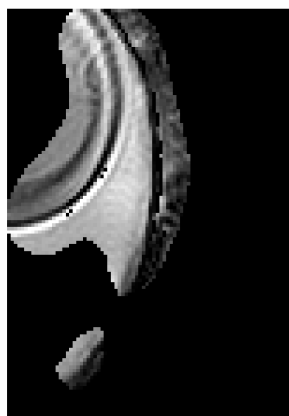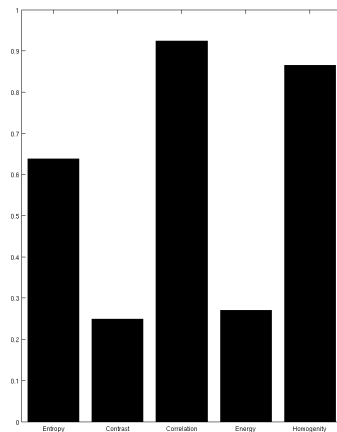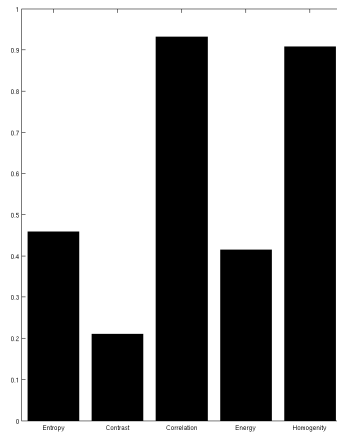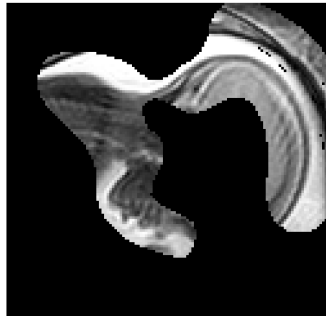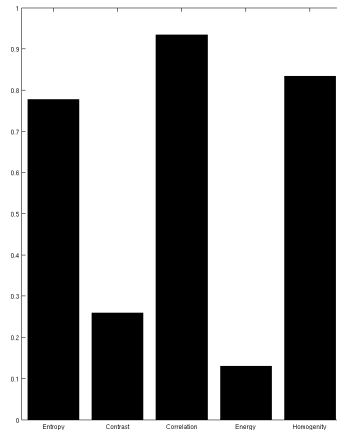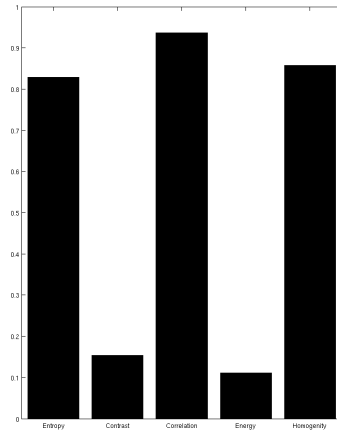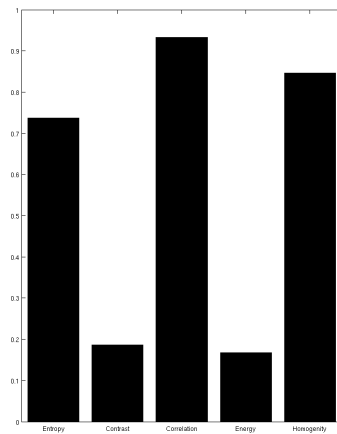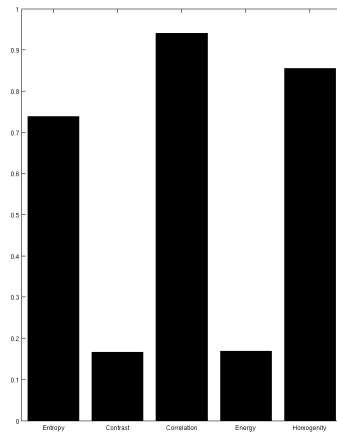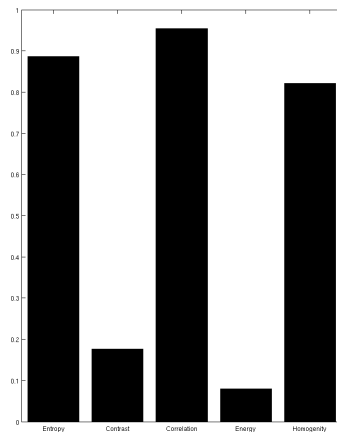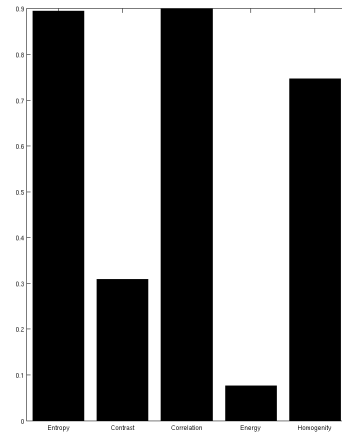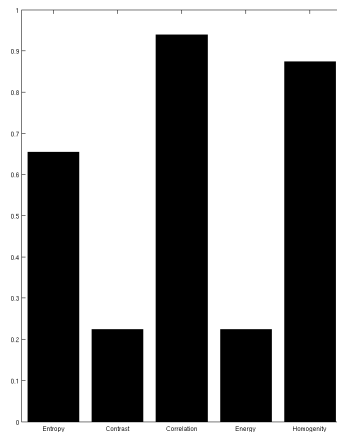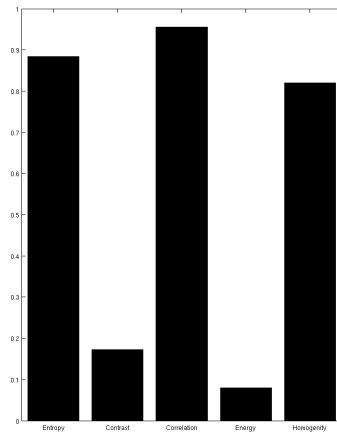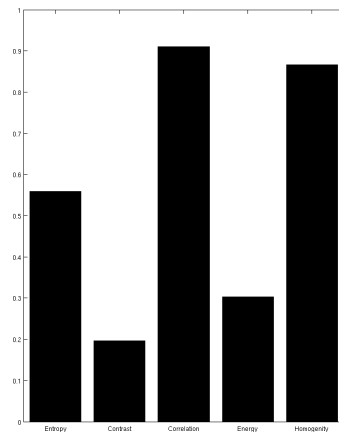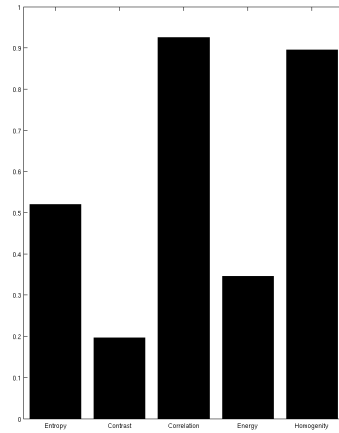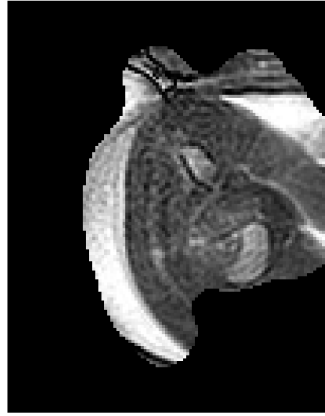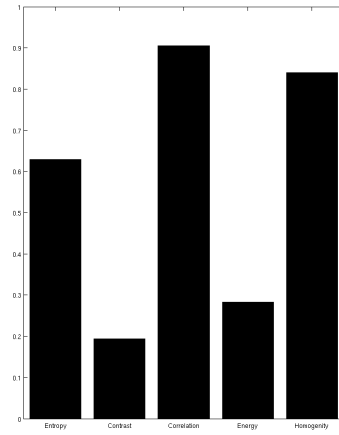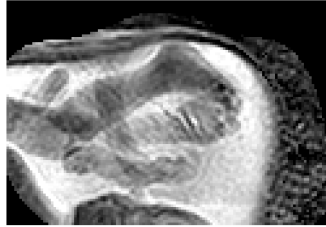
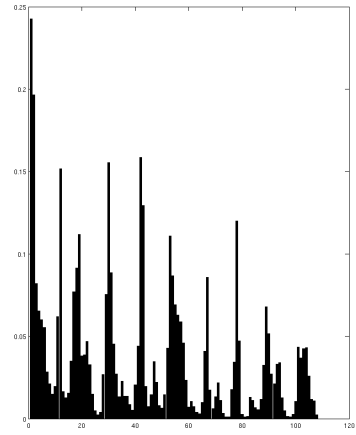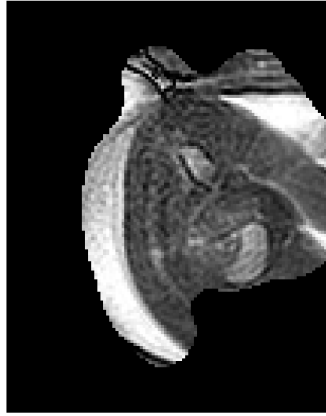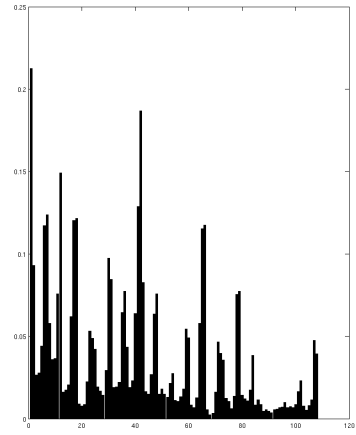### 7.1.3   Motion Histogram

Format: Feature — Signature
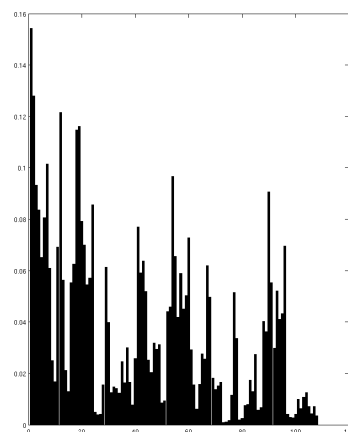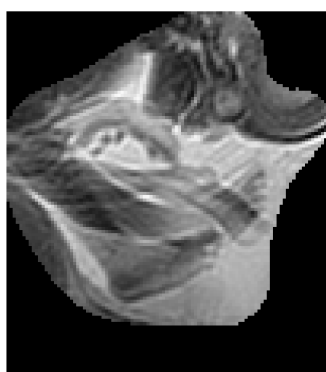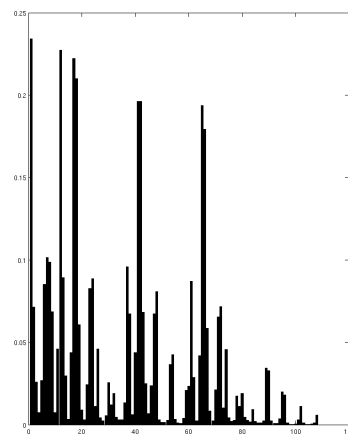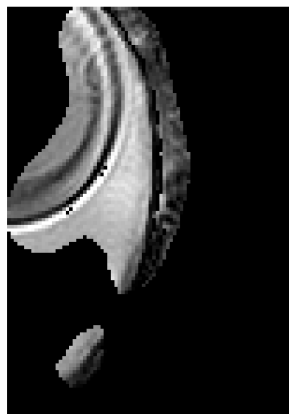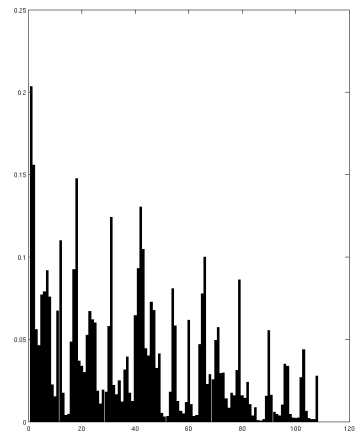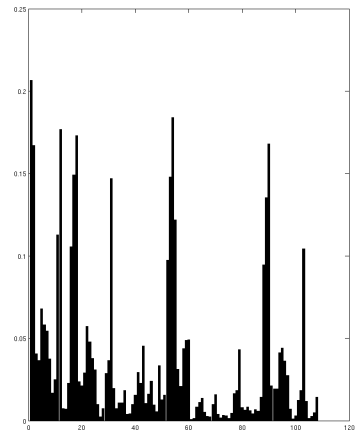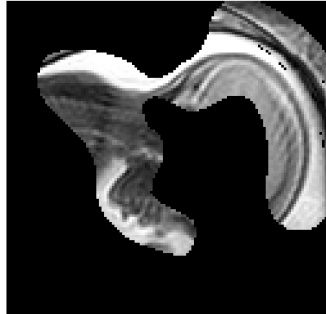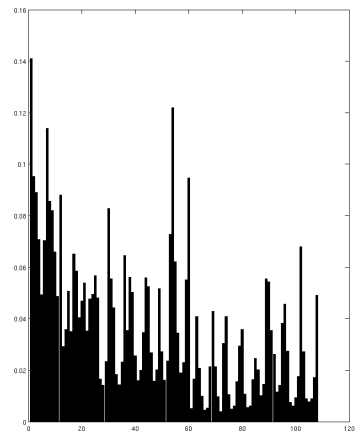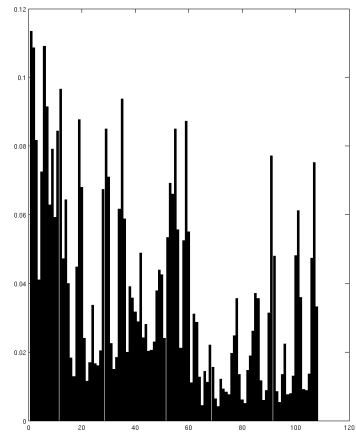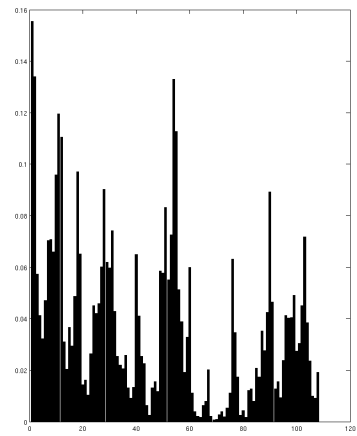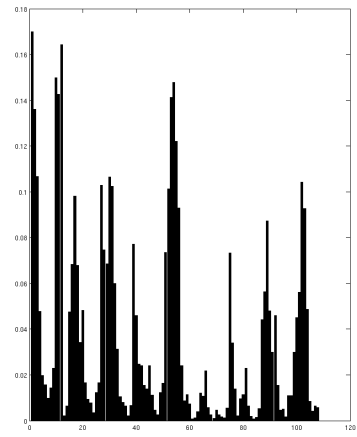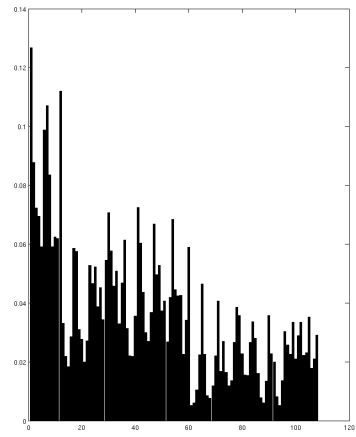
7.1.4 Fourier Descriptor of Movement Information

Format: Feature — Log of magnitude of Fourier transform — Signature