Can entropy-based image alignment metrics offer improved image aggregation of tissue density for mammographic risk assessment?

Final Report for CS39440 Major Project

Author: Laura Collins (lac32@aber.ac.uk)
Supervisor: Dr. Neil Mac Parthalin (ncm@aber.ac.uk)

4th March 2016
Version: 1.0 (Draft)

This report was submitted as partial fulfilment of a BSc degree in Artificial Intelligence and Robotics (inc Integrated Industrial and Professional Training) (GH7P)

Department of Computer Science
Aberystwyth University
Aberystwyth
Ceredigion
SY23 3DB
Wales, UK

Declaration of originality

In signing below, I confirm that:

- This submission is my own work, except where clearly indicated.
- I understand that there are severe penalties for Unacceptable Academic Practice, which can lead to loss of marks or even the withholding of a degree.
- I have read the regulations on Unacceptable Academic Practice from the University's Academic Quality and Records Office (AQRO) and the relevant sections of the current Student Handbook of the Department of Computer Science.
- In submitting this work I understand and agree to abide by the University's regulations governing these issues.

Name ....

Date ....

Consent to share this work

In signing below, I hereby agree to this dissertation being made available to other students and academic staff of the Aberystwyth Computer Science Department.

Name ....

Date ....

Acknowledgements

I would like to thank my Supervisor Neil for his constant help and guidance throughout this project.

Ryan for being my constant sound-board throughout the process, always happy to lend an ear when I needed to work through an issue, or bounce programming ideas off of someone.

Harry, Fangyi

Charlie

Abstract

Include an abstract for your project. This should be no more than 300 words.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Project Description

This project is concerned with the alignment of multiple mammographic images using an image-alignment technique called Congealing [1]. The aim will be to implement image-alignment software which allows the user to not only choose standard Shannon entropy to align the images as in [1], but also 2 different light-weight Fuzzy Entropy metrics for alignment - Non-Probabilistic and Hybrid entropy. The User will be able to generate 3 mean images of the input set, 1 for each metric. By utilising different alignment metrics on the same sets of input images the result should be a range of varying average output images, which further may be used to ascertain the most useful entropy algorithm for the alignment of mammographic images.

Each input set of images must belong to the same tissue density category (as covered in Sub-subsection 2.1.1.3), but from different women, to allow the resulting mean image to be an accurate depiction of the average breast structure within that category. Once a mean image is constructed of each category, this should aid radiographers in their qualitative categorisation of a new patient's scans.

Simple and accurate categorisation is important due to the increased risk factors associated with denser tissue breasts. Therefore if a radiographer can be confident in their categorisation of a patient's breast tissue, should the patient fall within the higher risk category they can receive more frequent, specialised scans to detect any abnormalities quicker should they arise.

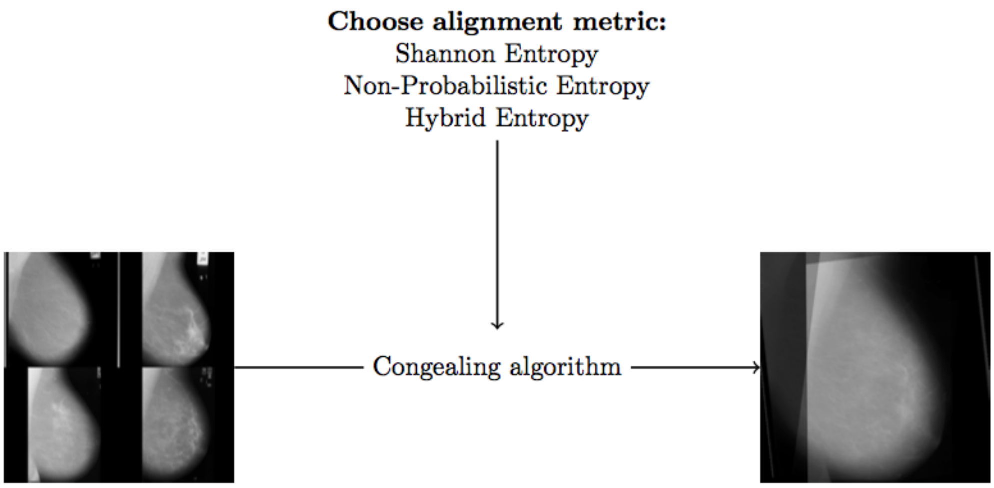


Figure 1.1: Graphical depiction of Project outline

Figure 1.1 outlines the major processes the User would take to align their input images. This includes:

- Selecting the appropriate input data of all the same BI-RADS classification
- Selecting which entropy alignment metric they would like to be run
- These are both fed into the Congealing algorithm
- An end average image is displayed

1.2 Project Structure

This section will give a brief overview of the structure of the project.

1.2.1 Research

The main piece of research to be undertaken in this project will be evaluating which Fuzzy Entropy algorithms will be light-weight and simple enough to be run quickly on a radiographer's own laptop. Typically, research implementations of Fuzzy Entropy algorithms tend to be complex, and therefore computationally expensive, something not ideal when a patient has a short time-slot with a radiographer.

1.2.2 Software Implementation

In order to assess the usefulness of basic fuzzy entropy algorithms in the alignment of mammographic scans, a tool must be built to handle the input images and all the output data. This tool will be created using MATLAB and it's Fuzzy Logic and Image Processing toolboxes.

The main functions of the tool will be:

- Allow the user to input a large image containing all the scans they wish to align
- Allow the user to remove any medical markers as they see fit
- Allow the user to choose their alignment metric and number of iterations to run on the input images
- Output the final mean image, the adjusted input images (how they look after aligning) and the entropy of the final image set

1.2.3 Testing

The testing to be undertaken during this project will include scientific and software.

1.2.3.1 Scientific testing

This will be testing the output after the Congealing process has been run using a fuzzy entropy alignment metric. One way to measure the result will be to evaluate the entropy value at the end of the alignment process - as the lower the entropy, the more aligned the images are. Another way in which to test the output of the experiments will be to visually inspect the final mean images produced to see how well aligned the input images are.

1.2.3.2 Software testing

Some software testing will be necessary to ensure the proper working of the tool developed for experimentation. Both Unit testing and acceptance testing off of the pre-defined user stories will be carried out.

1.3 Objectives

The Objectives for this project are are follows:

- **Align images using Fuzzy Entropy algorithms.** Images should be fed in, and it should be clear that an aligned version of the input images is calculated and output.
- **Answer any other relevant research questions associated with this project.** These are covered in Subsection 2.3.2
- **Create a tool to streamline inputting images and viewing the output.** As this project uses light-weight, simpler fuzzy entropy algorithms to hopefully speed up processing time *(See next objective)*, then the tool in which you run them should reflect this.
- **Create a quick tool which can be used on anyone's laptop or PC.** Not many people outside of the research community use tools such as MATLAB, so to be able to run a simple executable program is important.
- **Research and implement a solution to remove medical markers from mammogram scans.** As the Congealing algorithm looks to align the scans using grey-level pixel values, then the white medical markers in many mammograms create an issue as these will also try to align.

Chapter 2

Background

2.1 Background

In Europe, breast cancer is the leading cause of death through cancer for women, with 1 in 6 women dying from cancer having it in the glandular breast tissue [19]. The UK is contained within the higher mortality band which runs across the EU, sitting alongside countries such as the Netherlands, North-West France and Western Germany (see Figure 2.1). However the reason behind why these countries have a higher breast cancer mortality rate than their neighbours to the north and south is unknown.

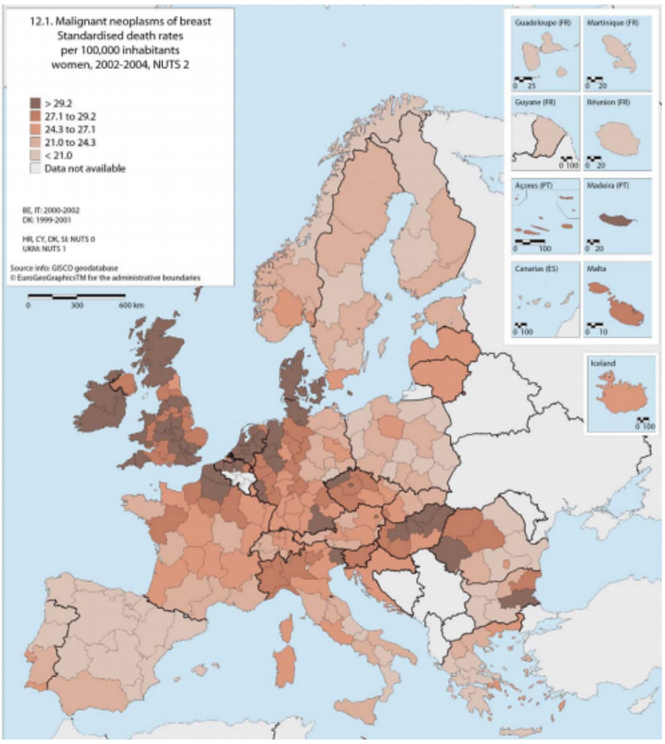


Figure 2.1: Breast tissue composition. *Image Source: EU Commission: Atlas on Mortality [19]*

2.1.1 Tissue density classification

The internal breast structure consists of different kinds of tissue and glands [3]:

- Fatty and connective tissue: protects the lobules and ducts, gives shape to the breasts
- Lobules - milk-production glands
- Ducts - carry milk from Lobules to Nipple

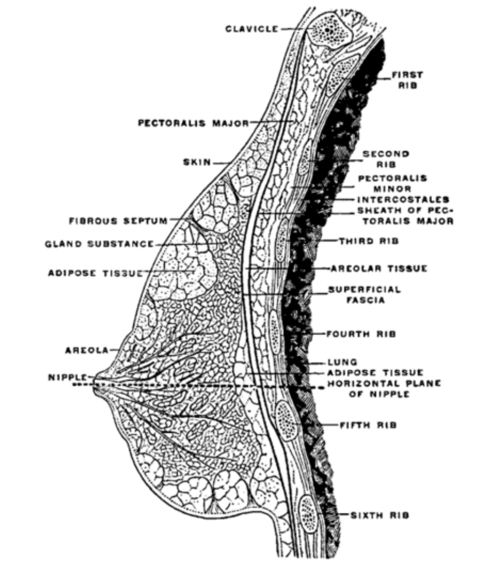


Fig. 1108.—Right breast in sagittal section, inner surface of outer segment. (Testut.)

Figure 2.2: Make up of breast structure. *Image Source: Gray's Anatomy [25]*

Fatty and connective tissue density can vary widely between women. After extensive research into the links between a higher proportion of fibrous/glandular tissue versus fatty tissue and a higher risk of breast cancer, it is widely accepted there is a strong link between dense tissue and breast cancer [15]. Therefore, simple classification of denser tissue is vital for both radiographers and patients alike.

There exists several methods for classifying the density of breast tissue, as outlined in the following subsections.

2.1.1.1 Wolfe classification

Wolfe described the first qualitative means in which to classify breast tissue density in 1976 [58].

- **N1:** consisting mainly of fat (lowest risk)
- **P1:** fat plus linear densities occupying no more than 25% of the breast (low risk)
- **P2:** linear densities occupying >25% of breast (high risk)
- **DY:** dense (highest risk)

2.1.1.2 Boyd classification

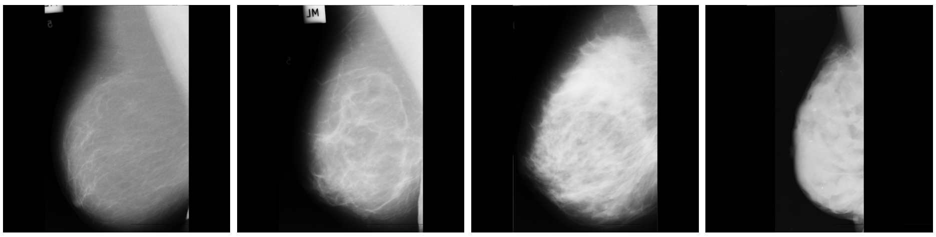
Boyd and colleagues proposed a quantitative means to categorising breast tissue density, based on a percentage of 'dense' tissue assigned by a radiographer [15].

- **A:** 0%
- **B:** >0% - 10%
- **C:** >10% - 25%
- **D:** >25% - 50%
- **E:** >50% - 75%
- **F:** >75%

2.1.1.3 BI-RADS classification

A widely accepted quantitative tool for the classification and risk analysis of mammography and ultrasounds is BI-RADS (Breast Imaging-Reporting and Data System) system, defined by the American College of Radiology [52].

- **a:** almost entirely fatty (Figure 2.3a)
- **b:** scattered areas of fibroglandular density (Figure 2.3b)
- **c:** heterogeneously dense, which may obscure small masses (Figure 2.3c)
- **d:** extremely dense, which lowers the sensitivity of mammography (Figure 2.3d)



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

Figure 2.3: Comparison of the 4 BI-RADS classification

This is the classification of choice for this project due to it's wide-spread acceptance and usage in the industry.

2.1.1.4 Tabár classification

This technique is somewhat different from the previous 3 by utilising anatomic-mammographic correlations, as developed by Tabár [24].

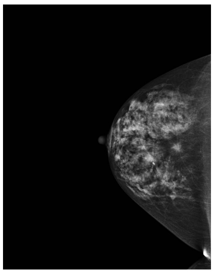
- **I:** balanced proportion of all components of breast tissue with a slight predominance of fibrous tissue
- **II:** predominance of fat tissue (fat breast)
- **III:** predominance of fat tissue with retroareolar residual fibrous tissue
- **IV:** predominantly nodular densities
- **V:** predominantly fibrous tissue (dense breast)

2.1.2 Mammograms

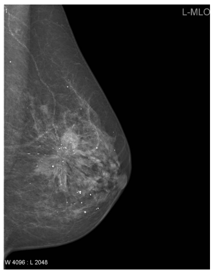
Quite simply, a Mammogram is an X-Ray of the breast tissue pressed between 2 plates from a number of different angles. Below are a selection of the most common [46] [12]:

- Cranial-Caudal (CC) - taken from above (Figure 2.4a)
- Medio-Lateral Oblique (MLO) - from the side, at an angle (usually 45deg) (Figure 2.4b)
- Medio-Lateral (ML) - from the centre outwards (Figure 2.4d)
- Latero-Medial (LM) - from the side, into the centre (Figure 2.4c)

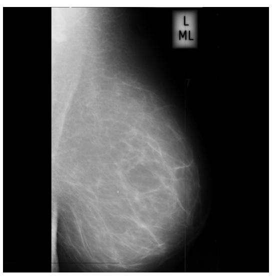
CC and MLO are generally standard practice angles, with ML and LM adding more information for the radiographer to assess.



(a) Cranial-Caudal: Case courtesy of Dr Garth Kruger, Radiopaedia.org, rID: 18580



(b) Medio-Lateral Oblique: Case courtesy of A.Prof Frank Gaillard, Radiopaedia.org, rID: 12608



(c) Medio-Lateral: Case courtesy of Mini-MIAS dataset [54]

(d) Latero-Medial: Case courtesy of Dr Paresh K Desai , Radiopaedia.org, rID: 5873

Figure 2.4: Comparison of the 4 mammogram angles typically used

Organisations such as Breast Test Wales invite women between the ages of 50 and 70 to attend a scan every 3 years [13]. However women with higher-density breasts, which is ascertained during a mammogram, could be called back for more regular screening, to ensure to catch any abnormalities sooner.

2.1.2.1 Alternatives to Mammograms

Although the input data of choice for this project will be Mammographic scans, it is important to remember that for some women, and under some circumstances, it may be more appropriate to use a different method of diagnosis.

Ultrasound

Women under 35 are often offered an ultrasound scan over a mammogram, due to their breasts being of a higher density naturally which makes obtaining a clear mammogram more difficult. Ultrasounds can also show the if the breast lump is a cyst, or if it is solid internally [55].

Biopsy

A Biopsy is usually a secondary step after diagnosis of a breast lump via mammogram or ultrasound. It can take a number of forms including:

- Needle biopsy
- Vacuum biopsy
- Needle aspiration
- Punch biopsy
- Wire guided biopsy

2.1.3 Existing Computer Systems

A lot of the current computer systems focus on mammography computer aided diagnosis due to the ease in which a Radiographer can misdiagnose cancer from a mammogram. Mammograms are difficult to read, or technological issues may occur and as a result, Radiographers can fail to detect between 10-15% of breast cancer cases [16].

However this is not the focus of this Project. This Project aims to focus upon healthy tissue, and the first steps towards computers classifying breast tissue into the correct density category. So what steps have been taken currently, in Industry and in research, towards this end goal?

Paper by Mohamed Abu ElSoud; Ahmed M. Anter - Automatic mammogram segmentation and computer aided diagnoses for breast tissue density according to BIRADS dictionary

Not sure if this section is needed? Cannot yet get ahold of above paper.

2.2 Research Method

For this project, a literature review was undertaken to assess the work completed by researchers in the fields of Entropy, Fuzzy Entropy and image alignment methods to help better understand what has been investigated, and to gain a personal background understanding.

2.2.1 Entropy

In terms of Information Theory, the Merriam-Webster Dictionary defines Entropy to be [1]:

Entropy (noun): the degree of disorder or uncertainty in a system

Shannon entropy, derived by Claude Shannon [50] can be mathematically defined as :

$$H(X) = -\sum_{i=0}^{N} p_i \log_2 p_i \tag{1}$$

Where p_i is the set of probabilities for all the variables in X.

Let us consider a fair coin toss. The probability of heads is exactly $\frac{1}{2}$, therefore, the entropy of landing on heads is:

$$\begin{aligned} H(heads) &= -\frac{1}{2}\log_2(\frac{1}{2}) - \frac{1}{2}\log_2(\frac{1}{2}) \\ &= 1.0 \end{aligned} \tag{2}$$

On the other side, if a system outputs solely the letter "M", then the entropy of receiving the letter "M" is exactly 0. This is because when either the positive or the negative outcome is 100%, then both sides equal "0" when fed into the entropy equation.

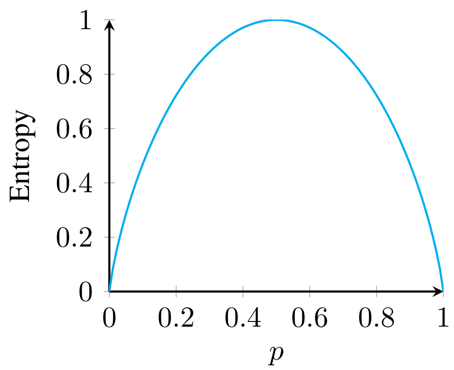


Figure 2.5: Entropy mapped against probability (p) of occurrence.

It follows that entropy can only ever be between 0 and 1, with an entropy of 0 have a 50% probability, and an entropy of 1 being 100% certain.

2.2.2 Uncertainty

However real life is not 100% certain - a certain amount of uncertainty in life is to be expected and sometimes desired. A surprise party for many is the nice kind of uncertainty, however uncertainty associated with risk - i.e. “Will I lose my job in the recession?” - is uncertainty with a negative impact. Modeling uncertainty is especially important so as researchers we can understand it, and use it to our advantage in techniques such as Fuzzy Entropy.

2.2.2.1 Probabilistic Uncertainty

By definition:

> *Probability: the chance that something will happen [7]*

Probabilistic distribution is a widely accepted and used technique for representing expert judgements of uncertainty [39]. Early work carried out by DeGroot (1970) [22], built upon that of Savage (1954) [49], gave a simple layman’s explanation:

> *For instance, if the person prefers decision A to B and B to C then they must also prefer A to C.*

2.2.2.2 Possibilistic Uncertainty

By definition:

> *Possibility: a chance that something might exist, happen, or be true : the state or fact of being possible [6]*

Possibilistic uncertainty (closely related to “fuzziness”) indicates the lack of information we hold about the possible outcome values from a system - a sort of ambiguity. Possibilistic uncertainty models the possible outcomes from a system, as estimated by a decision maker because it is possibly impossible to determine beforehand [56].

2.2.2.3 Indiscernibility Uncertainty

By definition:

> *Indiscernibility: the quality or state of being indiscernible [4]*
> *Indiscernible: impossible to see, hear, or know clearly [5]*

Find an explanation of Indiscernibility

2.2.3 Fuzzy Entropy

Fuzzy entropy stems from combining standard Shannon entropy with the practices of Fuzzy Set Theory, discovered by Zadeh in 1965 [59]. This introduces the idea of "Membership" to a category, where an object can belong to more than one category to a certain degree.

One common example of this is listing someone as 'Short', 'Average' or 'Tall' in height. If a tall person is someone over 6 feet in height, would a person who measured 5foot 11inches not be classified as tall? Given crisp sets, then they would be classified as 'Average'. In fuzzy set theory, they would be be a certain degree of tall, and a certain degree of average, with the highest membership likely to win out when categorising their height. Another example of this can be seen in Figure 2.6

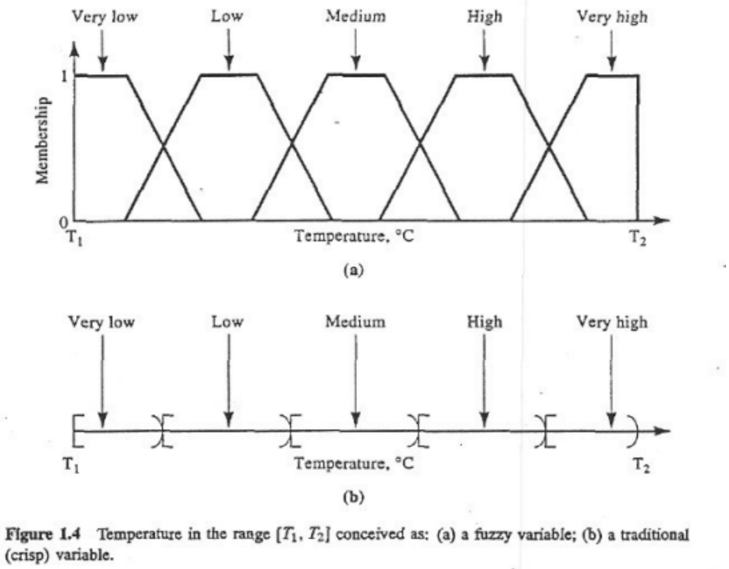


Figure 2.6: A comparison between Fuzzy Sets and Crisp sets. *Image Source: Fuzzy Sets and Fuzzy Logic: Theory and Applications [23]*

After combining Fuzzy Set Theory with Entropy, then the amount of fuzzy information gained from the fuzzy set(s) is known as Fuzzy Entropy.

2.2.3.1 Non-Probabilistic Entropy - 1972

De Luca and Termini are considered to be the first to have taken Shannon Entropy and extended it to include fuzziness [21]. They also defined properties which a fuzzy entropy must follow, in order to be classed as true.

Their non-probabilistic fuzzy entropy equation is as given:

$$H_A = -K \sum_{i=1}^{n} \{\mu_i \log(\mu_i) + (1 - \mu_i) \log(1 - \mu_i)\} \tag{3}$$

Where μ is the maximum membership across all the fuzzy sets.

The entropy given by equation (3) satisfies all 4 of De Luca and Termini's defined properties:

P-1 $H_A = 0$ iff A is a crisp set ($\mu_i = 0$ o r1$\forall x_i \in A$) (4a)

P-2 H_A is maximum iff $\mu_i = 0.5 \forall x_i \in A$ (4b)

P-3 $H \geq H^*$ where H^* is the entropy of A, a sharpened version of A (4c)

P-4 $H = \overline{H}$ where $\overline{H}$ is the entropy of the complement set $\overline{A}$ (4d)

2.2.3.2 Fuzzy Shannon Entropy - 1989

Sander [48] presented a characterisation of a Fuzzy Entropy some time after De Luca and Termini's work was published. His implementation of Shannon Fuzzy Entropy is laid out in equation (5) below:

$$H(f) = -c \sum_{i=1}^{n} f(x_i) ln f(x_i), c > 0 \tag{5}$$

Where the power of a fuzzy set is defined as:

$$P(f) = \sum_{i=1}^{n} f(x_i) \tag{6}$$

Sander further went on to propose some properties, which must be imposed on a Fuzzy Entropy d to ensure that $d(f) = H(f)$:

1. Sharpness: $d(f) = 0 \Leftrightarrow f(X) \subset 0, 1, f \in [0,1]^X$ (7a)

2. Valuation: $d(f \wedge g) + d(f \vee g) = d(f) + d(g), f, g \in [0,1]^X$ (7b)

3. Generalised additivity: There exists two mappings s,t: $[0, \infty) \rightarrow [0, \infty)$ such that $d(fxg) = d(f)t(P(g)) + s(P(f))d(g)$ for all $f \in [0,1]^X, g \in [0,1]^Y$, where X and Y are finite sets. (7c)

2.2.3.3 Object-background segmentation using new definitions of entropy - 1989

Pal & Pal outlined their first Fuzzy Entropy algorithm in 1989 [40], which satisfies all 4 of De Luca and Termini's 4 conditions (outlined in Equations(4)). It as as follows:

$$H = -k \sum_{i=1}^{n} \{\mu_i exp(1 - \mu_i) + (1 - \mu_i) exp(\mu_i)\} \tag{8}$$

2.2.3.4 Higher Order Fuzzy Entropy & Hybrid Entropy - 1992

In Pal & Pal's paper "Higher order fuzzy entropy and hybrid entropy of a set" [41], they not only prove some of De Luca & Termini's work to be flawed, but also defined two new Fuzzy Entropy algorithms, and a new set of definitions.

Higher Order Fuzzy Entropy

As defined by Pal & Pal:

- $P =$ Fuzzy property set
- $\mu =$ the degree to which x_i possesses the property P
- $n =$ number of elements, with $r =$ a combination of elements from group n
- $S_i^r =$ denotes the ith element of such a combination
- $\mu(S_i^r) =$ the degree to which the combination S' as a whole possesses P
- There are $\left[\binom{n}{r}\right]$ such combinations

The entropy of order r of the fuzzy set A is defined as:

$$H' = \left(\frac{I}{\binom{n}{r}}\right) \sum_{i=1}^{\binom{n}{r}} \{\mu(S_i^r) exp(1 - \mu(S_i^r))\} + \{1 - \mu(S_i^r)\} log\{\mu(S_i^r)\} \tag{9}$$

If $r = 1$, then (9) reduces to Equations (8) and (3)

Hybrid Entropy

Another Fuzzy Entropy implementation outlined in Pal & Pal's paper was Hybrid Entropy. This algorithm is particularly useful as it combines Probabilistic and Possibilistic (fuzziness) uncertainty and if fuzziness is removed or not present, it returns to that of a classical set.

Let us define Hybrid Entropy.

- Let p_0 and p_1 be the probabilities of receiving 0 and 1 symbols over a noisy digital communication line respectively.
- Let μ denote the membership functions of the fuzzy set "Symbol close to 1"
- Both E_1 is a monotonically increasing function of μ - E_0 can be perceived as the likelihood (possibility) of receiving a "1" symbol
  - as μ increases from 0 to 1, then E_1 also increases
  - e.g. with an incoming "0" symbol, if μ increases, than the difficulty of correct interpretation also *increases* - a wrong interpretation of a "0" becomes likely
  - e.g. for an incoming "1" symbol, if μ increases, then the difficulty of correct interpretation *decreases* - improving likelihood of correct classification

- At the same time, E_0 can be perceived as the likelihood (possibility) of receiving the "0" symbol for the same reasoning

E_0 and E_1 can be defined as:

$$E_0 = \frac{1}{n} \sum_{i=1}^{n} (1 - \mu_i) exp(\mu_i) \tag{10a}$$

$$E_1 = \frac{1}{n} \sum_{i=1}^{n} \mu_i exp(1 - \mu_i) \tag{10b}$$

Therefore, the hybrid entropy of fuzzy set A can be defined as:

$$H_{hy} = -p_0 \log(1 - E_0) - p_1 \log(E_1) \tag{11}$$

2.2.3.5 Fuzzy Entropy: a Brief Survey - 2001

Due to the older nature of some of the papers listed above, some were difficult to locate online. So when implementing the chosen algorithms (Non-Probalistic Entropy and Hybrid Entropy), Alsharhan et al's paper "Fuzzy Entropy: a Brief Survey" [14] was a useful tool.

It's concise nature, and chronological listing ensured a strong understanding of the basic principles, before introducing the more complex algorithms (such as Higher Order Fuzzy Entropy). The paper also highlights advantages and flaws to each solution.

2.2.4 Joint Image Alignment

Joint image alignment focuses on the alignment of several images, into one average image. This Subsection will look into a couple of the techniques available for this project.

2.2.4.1 Learned-Miller‘s Congealing

Learned-Miller's Congealing [29] is often cited as being one of the first to truly align simple sets of data (which must have minimal noise, no occlusions and illumination variation) [60] [43] [42]. Many more robust image alignment techniques have been developed off of the basis of this work, however with more computational-expense.

This algorithm works by iteratively reducing the pixel-wise entropy over the input images, using a set of standard image transformations such as:

- x & y translations
- rotation
- x & y sheer

- x & y scale

The entropy is calculated by assessing each individual set of pixel-locations in the 'Pixel Stack' (see Figure 2.7), and by calculating the entropy of the empirical distribution of values in the Pixel Stack.

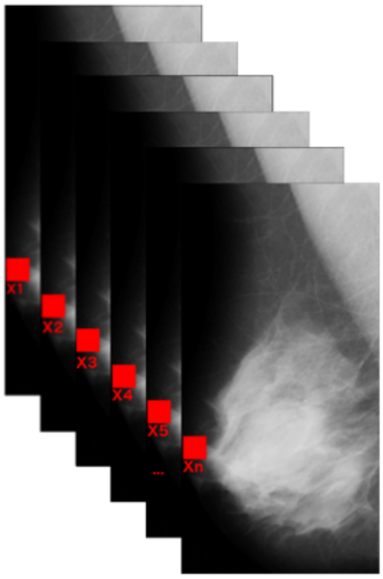


Figure 2.7: Each pixel from the same location throughout the set creates a 'Pixel Stack'

2.2.4.2 Least squares Congealing for unsupervised alignment of images

Further work was done upon the Congealing algorithm proposed by Learned-Miller by Cox et al. in 2008 [20]. They set out to address any performance issues and to remove the need for a pre-defined step size. It proposes to mitigate these issues by implementing an alternative method for aligning the images - utilising the Lucas & Kanade algorithm for aligning a single image to another using a gradient descent approach [30].

2.2.4.3 Unsupervised Joint Alignment of Complex Images

Huang and a team (notably including Learned-Miller) further extended the Congealing algorithm to be usable upon complex images - such as faces and cars at different orientations [26].

This method removes the need to hand-label the input data and improves the performance of face recognition systems, by ensuring the objects are properly oriented prior to recognition.

2.2.5 Image Alignment using Fuzzy Entropy

Research has been undertaken in the past to investigate image alignment using Fuzzy Entropy metrics, however typically they were found to be computationally costly, and therefore slow to run on a conventional PC or laptop. This project will be investigating whether there are simpler, more light-weight fuzzy entropy metrics which could be implemented, for more everyday use in image alignment. It will also be investigated if, and further how, the outputs of these alignments differ per each fuzzy entropy metric.

Some of this work which has implemented a more computationally-costly Congealing algorithm is that presented by Mac Parthaláin and Strange in their 2013 paper "Fuzzy-entropy based image congealing" [31]. Their implementation included dynamically-calculated fuzzy sets and a fuzzy similiarity relation matrix - allowing a comparison of all the objects to each other.

2.3 Analysis

2.3.1 Task composition

After both the background research and literature review were completed, a list of main "Tasks" to be undertaken in order to complete this Project was easily composed. These are outlined in the following subsections.

1: Decide how best to implement Membership functions

Fuzzy Entropy requires Membership functions, in which the data can be classified. This task would be to decide whether to dynamically calculate the Membership functions, or whether to statically define them within the back-end of the system.

2: Research and choose which Fuzzy Entropy algorithms would be best suited to this project

This project stresses the importance of running image alignment techniques on a standard laptop or PC. With this in mind, each Fuzzy Entropy algorithm would need to be analysed based on their simplicity of calculations, in order to run quickly.

3: Research and decide upon which image alignment technique would be best

Given this project is about aligning images using entropy techniques, this may indeed be an easy task to complete. However, other options should indeed be considered, especially if they are accurate and have low computing cost.

5: Decide which programming language to use

A lot of image processing can be run using programs such as Python, however the demo code given by Learned-Miller [29] has been compiled in MATLAB. Research would have to be undertaken to see if Learned-Miller‘s approach has been implemented in any other languages if this is the image alignment approach decided upon.

6: Decide how best to represent the input and output data

This most likely would be in the form of a GUI as the target User - Doctors - wouldn‘t have access to tools like MATLAB, nor be well-versed in using the Command Line or Terminal.

7: Determine how well to best assess the output

This could possibly involve a professional medical opinion? Or it could be based of the amount the entropy declined over time? Or it could simply be a visual inspection of how well the scans aligned.

8: Research different ways in which to build the GUI

Which programming language would this be done in? If MATLAB were to be chosen, it is capable of linking in which many other languages, such as Java, C and C++. However if the image alignment was done in another programming language, it may be possible to stick to just one language.

9: Determine which dataset to use

Given the data of choice - Mammograms - there exists an ethical issue of utilising people's medical scans. Research would have to be undertaken to determine whether there are freely-available, license-free Mammographic scans available for use in this project.

2.3.2 Research questions

The research questions are tightly interwoven with the Objectives of this project, outlined in Section 1.3

2.3.2.1 Does the use of Fuzzy Entropy alignment metrics improve the alignment of mammograms?

Through background research it would follow that there would be no issue in aligning images using fuzzy entropy techniques. However the implementation might be somewhat difficult or computationally-costly, which would undermine the objective of a quick tool.

And if so, could one be more useful than another? As the uncertainty in fuzzy entropy will help model different types of tissue, the way in which they assess uncertainty will affect the output image.

2.3.2.2 Do clinicians / radiographers / mammographers find the output at all useful?

Does the output show a Doctor something useful? Does one of the Fuzzy Entropy alignment metrics show something different to one of the others? These are questions which would help to validate the success of this research question.

This would be a step towards making a tool that one day could be utilised by Doctors worldwide to aid in their classification of a patient‘s breast tissue, by giving a second opinion based on image analysis.

2.3.2.3 What advantages / disadvantages does each fuzzy entropy alignment metric entail?

One algorithm may be slower, but produce better results, so it is important to weigh up the speed versus the quality of the output.

Chapter 3

Experiment Methods

3.1 Overview

In order to test the main hypothesis of “Does the use of Fuzzy Entropy alignment metrics improve the alignment of mammograms?” some kind of tool was needed to portray the visual output. It would be built to take input images, select alignment metric, select amount of iterations desired and output the final congealed image. Details of the decreasing entropy would be a key output, along with the average image after each iteration completed, for a full picture of improvements.

However, the decisions about how to implement membership, which fuzzy entropy algorithms to use and which image alignment techniques remain.

3.1.1 Pixel Membership

From the analysis of the planned Fuzzy Entropy algorithms, one major task to be undertaken would be to calculate the membership of each pixel. Membership stems from Fuzzy set theory, as outlined in Subsection 2.2.3.

There are two common methods to modeling degrees of membership. The first is to manually define the categorty boundaries, so in the case of trapezium functions, the two bases and the two shoulders. The other solution would be to iterate over the values you have and to computationally build the an even distribution throughout your membership functions, as in [31]. Whilst this is the preferred method for being dynamic in it’s calculations, it is also more computationally expensive as pre-processing of the image would have to be completed before the Congealing algorithm could be run.

Taking the computational-expense into account, for grey-level pixel values, ranging from 0 (black) to 255 (white), three trapezium functions would be sufficient, therefore modeling ‘Low’, ‘Medium’ and ‘High’ grey-level values. The bases and shoulders would be statically defined, as in Figure 3.7. For Non-Probabilistic entropy the highest membership for each pixel from each of the three trapeziums would be taken as the membership degree. Hybrid entropy would take a slightly different approach, which will be covered later.

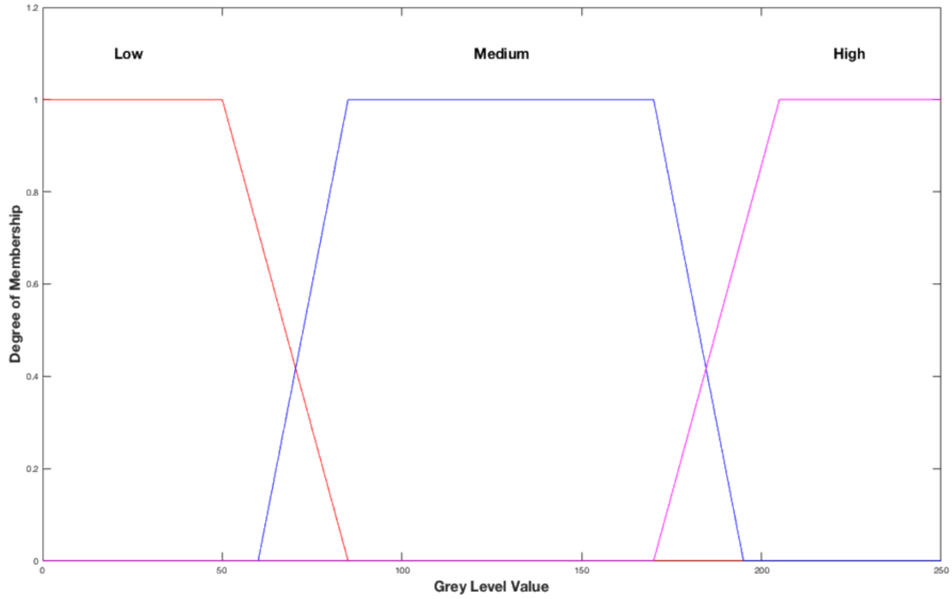


Figure 3.1: 3 trapezium-shaped membership sets

3.1.2 Fuzzy Entropy choices

Chosen algorithms:

- Non-Probability Entropy
- Hybrid Entropy

Given the simplistic nature of Non-Probabilistic entropy, this was one of the chosen Fuzzy Entropy algorithms to be implemented in the project.

Hybrid entropy was chosen for implementation in this project due to it's hybrid nature (implementing both Probabilistic and Possibilistic uncertainty) and for it's simplification nature - in the absence of fuzziness, then E_0 and E_1 reduce to p_0 and p_1 respectively, therefore classical Shannon entropy. This is especially useful in image processing, and other such areas which deal with a lot of noise.

Additionally, Shannon Entropy has already been implemented by Learned-Miller's Congealing algorithm, so this offers a non-fuzzy alternative to image alignment.

Discarded algorithms:

- Fuzzy Shannon Entropy
- Higher Order Entropy

The initial plan was to implement this algorithm in the project - however after further investigation which revealed that Fuzzy Shannon Entropy does not model Probabilistic uncertainty - it was decided that this algorithm was to be excluded.

This project does not implement Higher Order Fuzzy Entropy due to the computational-overhead needed to run - especially on images with as much detail as a mammogram.

3.1.3 Image Alignment choice

Image Alignment choice:

- Congealing

As this project will be working with mammograms, something with little variation nor inconsistency, Congealing is the perfect, light-weight image alignment algorithm to which to build upon, especially as the demonstration code available for research has an entropy implementation already developed.

Discarded Image Alignment choice:

- Least squares Congealing
- Joint Alignment of Complex Images

Least squares Congealing algorithm was disregarded for this project due to the preference to focus upon entropy-based alignment algorithms and the computational costs that the authors themselves regard to be a drawback of their algorithm.

The Complex implementation of Congealing was quickly identified as overly complex for this project. The original Congealing algorithm was more appropriate for grey-scale mammograms, with a consistent canonical pose.

3.2 Implementation tools

3.2.1 MATLAB

3.2.2 Version Control

3.3 Algorithms

3.3.1 Shannon Entropy

3.3.2 Non-Probabilistic Entropy

3.3.2.1 Fuzzy entropy description

De-Luca & Termini fuzzy entropy algorithm [21] is considered to be the first to build upon Shannon entropy. Their implementation takes into account a set of data, along with their various membership degrees.

$$H_A = -K \sum_{i=1}^{n} \{\mu_i \log(\mu_i) + (1 - \mu_i) \log(1 - \mu_i)\} \tag{1}$$

Al-sharhan et al's paper compiling several Fuzzy Entropy algorithms [14] contains a methodical,in-depth derivation of their algorithm, and has been instrumental in building my knowledge on the algorithm in question.

We will assume $-K$, the positive constant, is defined as $\frac{1}{n}$ as outlined in [21].

3.3.2.2 MATLAB implementation

After some research into current implementations of Fuzzy Entropy algorithms in MATLAB, it was concluded the best approach would be to implement De-Luca & Termini's algorithm from scratch. This entailed creating a membership class, which computes the grey-level membership of each pixel in the mean image (calculated from a set of input images).

This array of pixel memberships is fed into a 'De Luca' function where it is iteratively passed into latter part of equation 1 (after $\sum$). The output array is then summed and multiplied by $\frac{1}{n}$ as defined in Section 3.3.2.1. The final mean pixel entropy is calculated by taking the image entropy and dividing by the number of pixels in the image.

This is all relatively straight forward to implement in MATLAB, as it is designed to run mathematical equations.

3.3.2.3 Technical challenges

The main technical challenge for this implementation is ensuring maximum optimisation to keep running times to a minimum. Leveraging MATLAB's own functions for the membership saves a lot of time and lines of code, however it's been important to check what they call from within. One membership function was redrawing the trapeziums every time it was called, significantly slowing down the process - reducing the amount of times the initial function was called helped reduced the run-time by over 60seconds.

Another technical challenge faced whilst implementing the De Luca & Termini algorithm, isn't directly tied to the implementation of their specific equation, but more of my lack of experience in MATLAB, slowing down the programming rate. It has indeed been a steep learning curve, getting to grips with standard error messages, the debugger tool and knowing which 'Toolboxes' are needed to run specific MATLAB functions.

Finally, as can be ascertained from Figure 3.3, when writing the 4 seperate scans into 1 larger file, somewhere the images get rotated. This will be a reoccurring issue through the 3 Fuzzy entropy implementations, however as this is the first I will note it here. I think this is caused thanks to the swapping of the height and width values, however upon initial inspection of the file writing function, it is not clear as to which line is causing this issue. This issue has been marked low priority in the short term, due to all the scans being rotated in this fashion, and as such all have the same orientation. This means the Congealing algorithm can work with no issues upon these images, the rotation is more merely an aesthetic issue.

3.3.2.4 Results

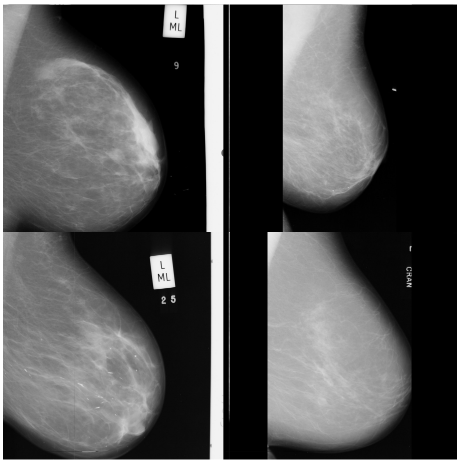
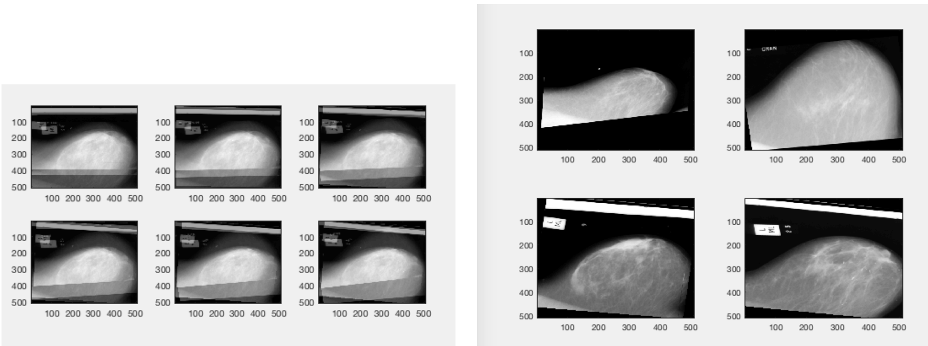


Figure 3.2: 4 input images of BI-RADS I classification



(a) Mean image over each iteration (b) Adjusted original input images

Figure 3.3: Output of 5 Congealing iterations

3.3.2.5 Entropy results

Iteration	Entropy
1	0.050519
2	0.043925
3	0.035679
4	0.029035
5	0.026194

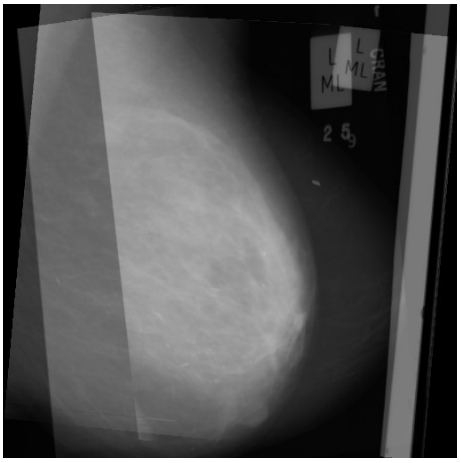
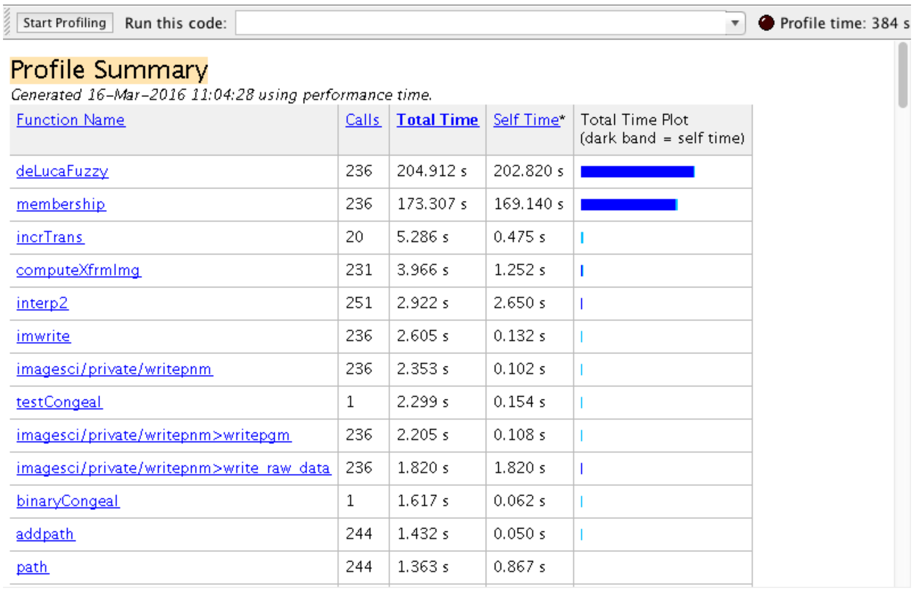


Figure 3.4: Final mean image after 5 iterations (bottom-right most in Figure 3.3)

3.3.2.6 Time to Run



Start Profiling Run this code: Profile time: 384 s

Profile Summary

Generated 16-Mar-2016 11:04:28 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
deLucaFuzzy	236	204.912 s	202.820 s	
membership	236	173.307 s	169.140 s	
incrTrans	20	5.286 s	0.475 s	
computeXfrmImg	231	3.966 s	1.252 s	
interp2	251	2.922 s	2.650 s	
imwrite	236	2.605 s	0.132 s	
imagesci/private/writepnm	236	2.353 s	0.102 s	
testCongeal	1	2.299 s	0.154 s	
imagesci/private/writepnm>writepgm	236	2.205 s	0.108 s	
imagesci/private/writepnm>write_raw_data	236	1.820 s	1.820 s	
binaryCongeal	1	1.617 s	0.062 s	
addpath	244	1.432 s	0.050 s	
path	244	1.363 s	0.867 s	

Figure 3.5: Snapshot of run-time statistics

3.3.3 Hybrid Entropy

As mentioned in Section 2.2.3.4, the Hybrid Entropy equation is as follows:

$$H_{hy} = -p_0 \log(1 - E_0) - p_1 \log(E_1) \tag{2}$$

Where E_0 and E_1 can be defined as:

$$E_0 = \frac{1}{n} \sum_{i=1}^{n} (1 - \mu_i) exp(\mu_i) \tag{3a}$$

$$E_1 = \frac{1}{n} \sum_{i=1}^{n} \mu_i exp(1 - \mu_i) \tag{3b}$$

And p_0 and p_1 are the probabilities of receiving 0 and 1 symbols respectively.

3.3.3.1 MATLAB implementation

Due to reasons covered in the Subsubsection 3.3.3.2, Hybrid Entropy membership was implemented using 2 trapeziums covering 2 fuzzy sets, as seen in Figure

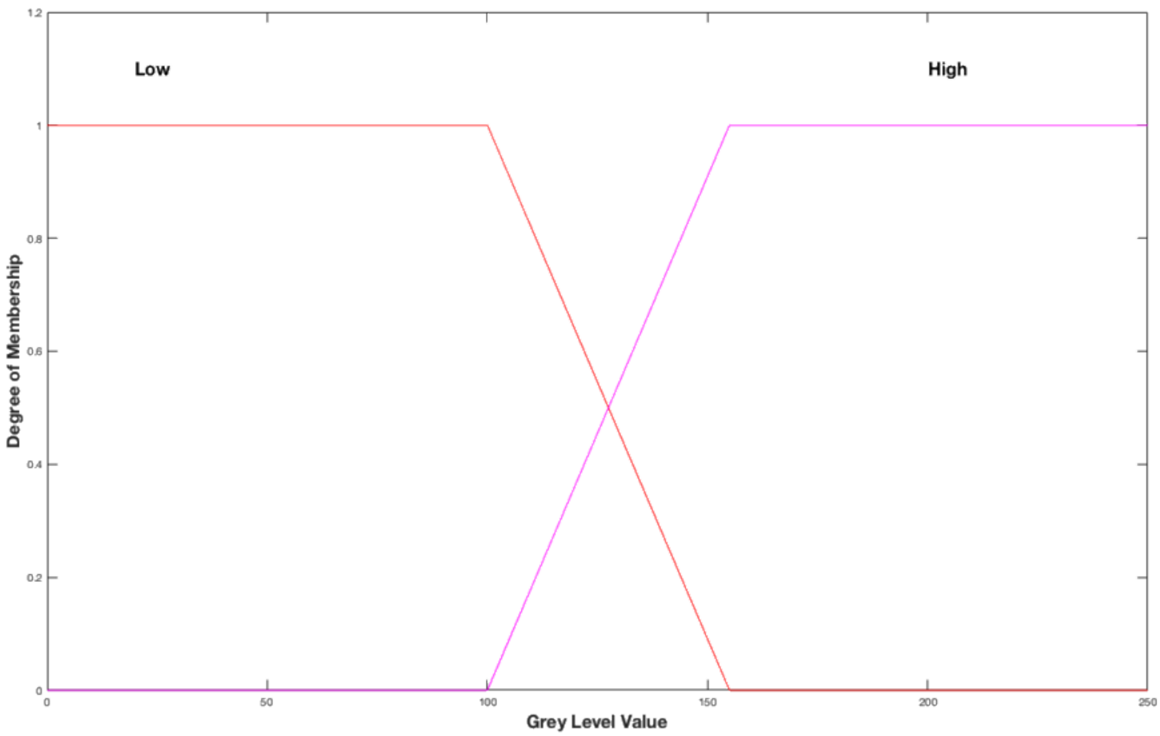


Figure 3.6: Two membership trapeziums for Hybrid Entropy - Low and High grey-level values.

Two arrays are then fed into the Hybrid Entropy function - one listing all the pixel membership values from the low trapezium, and the other from the high trapezium. The final entropy is taken as a comparison between the low and high fuzzy sets.

3.3.3.2 Technical challenges

Whilst Hybrid Entropy utilises a membership function, much like Non-Probabilistic entropy, it was derived to work with binary entropy, not the ternary membership modeled for Non-Probabilistic. Because of the binary nature, the equation uses 'inversion' to depict if not this fuzzy set, then must belong to the other.

Experimentation was done as to whether the equation could be adapted in such a way to continue using three separate membership trapeziums - low, medium and high grey-level values.

Initial ideas - check email between me and neil

Logic would dictate that if the comparison of two fuzzy sets works, then to compare the low fuzzy set to the medium, the medium to the high and the high to the medium should work.

For example:

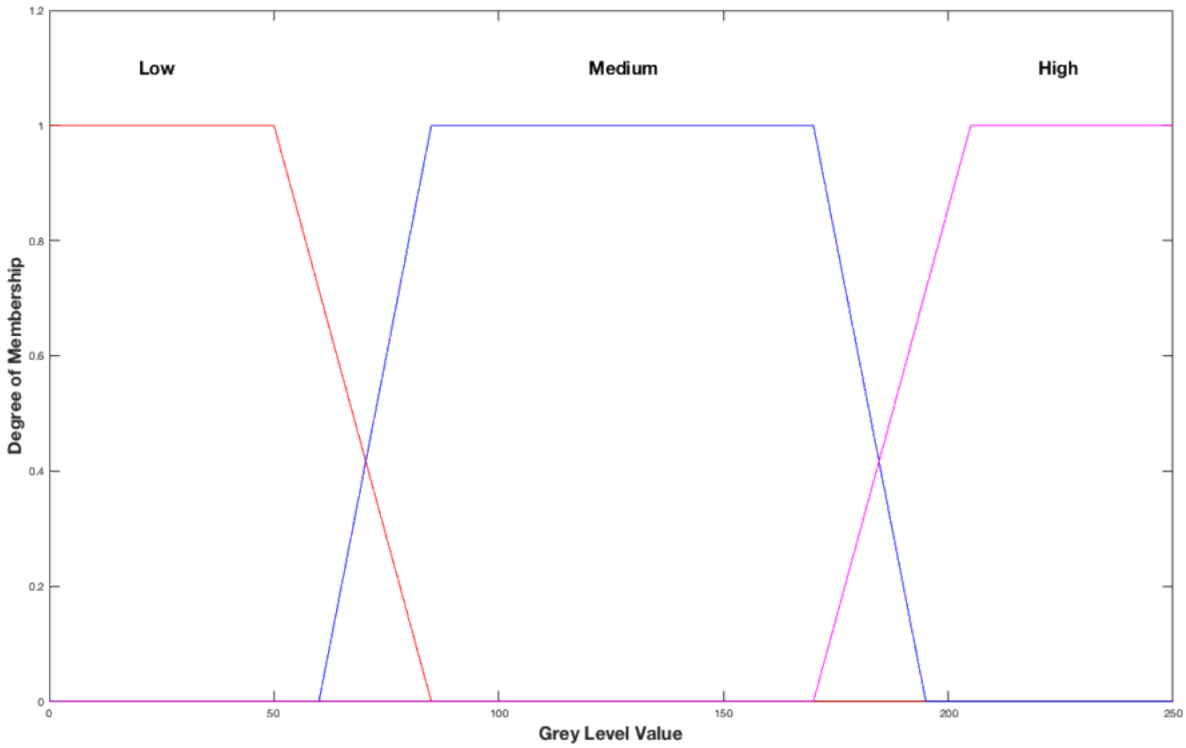


Figure 3.7: 3 fuzzy set trapeziums

In theory, calculating E_0 and E_1 for each trapezium, calculating the hybrid entropy for each, and then combining them, should work:

$$E_0 = \frac{1}{No.\_of\_pixels\_in\_low\_trapezium} \sum_{i=1}^{n} (1 - Low\mu_i) exp(Low\mu_i) \tag{4}$$

$$E_1 = \frac{1}{No.\_of\_pixels\_in\_low\_trapezium} \sum_{i=1}^{n} Low\mu_i exp(1 - Low\mu_i) \tag{5}$$

Where $Low\mu$ is the membership of the pixels in the low fuzzy set.

$$H_{hy} = -p_0 \log_{10}(1 - E_0) - p_1 \log_{10}(E_1) \tag{6}$$

Where

$p_0 = \frac{No.\_of\_pixels\_in\_low\_trapezium}{No.\_of\_pixels\_in\_low\_trapezium + med\_trapezium}$

and

$p_1 = \frac{No.\_of\_pixels\_in\_med\_trapezium}{No.\_of\_pixels\_in\_low\_trapezium + med\_trapezium}$

This was done for all 3 trapeziums, then combined and divided by 3 (for the mean entropy). As the result for each trapezium should be between 0 and 1 (as each is an entropy value), then combining them should be no issue. However this was not the case.

First of all, the hybrid equation output was deemed to be 'NaN' - something which generally occurs when attempting to divide by 0. Anomalous outputs from the high trapezium was to be expected, as there are very few pixels which fall within the range nearer the white end of the grey-level scale. This was mitigated by setting the output equal to 0, in effect ignoring any output from the highest fuzzy set.

After this mitigation, the third and fourth iteration had suitable entropy values, however the fifth entropy value was a negative, something which is not possible in terms of entropy, as it must be between 0 and 1 - see Figure 3.8.

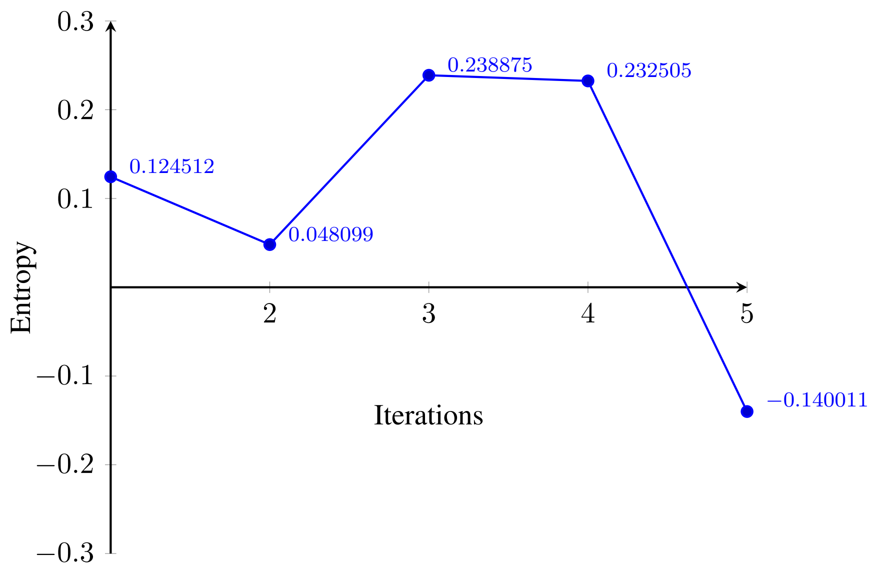


Figure 3.8: Graph showing the entropy output after 5 iterations

It was concluded that the implementation of three fuzzy sets within Hybrid Entropy would not be realistic within the remaining timeframe of the project, and the membership for Hybrid Entropy was redefined to the concept of 2 fuzzy sets, as set out by Pal and Pal. This would mean, one trapezium for pixel grey-level values with low values, overlapping with a high grey-level value trapezium at approximately 128, as seen in Figure 3.6.

3.3.3.3 Run-time

3.4 Software

3.4.1 Methodology

In the past, software projects followed a strict-plan driven approach, such as the Waterfall method, however more recently, Agile practices have become widely accepted, allowing the developer more freedom. This features an iterative development approach, with short "iterations" or "sprints" defined in which the developer should complete a block of work, typically a "story" or "feature" given the Agile methodology chosen.

The Agile Methodology has a manifesto [8], which perfectly encompasses all the values it strives to achieve:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

> *That is, while there is value in the items on the right, we value the items on the left more.*

Scrum is one of the most popular interpretations of an Agile Methodology, due to it's simplicity [2]. Scrum is *not* an agile methodology, however is a framework, to which agile practices such as Pair Programming and Test Driven Development (TDD) can be aligned.

Given it's flexible and light-weight nature, an adapted Scrum methodology has been undertaken for this project. The flexible nature is particularly useful given the research nature of this project, as the requirements were not fully defined at the start of the process, and changed as time went on given the outcome of experimentation with mathematical concepts for image alignment.

Additionally, eXtreme Programming (XP) [57] dates back to 1996, and is one of the most recognisable Agile Methodologies used in the software industry currently. XP claims to create successful software projects by following 5 key principles:

- **Communication:** constantly communicate with their customers and fellow programmers
- **Simplicity:** keep the design simple and clean
- **Feedback:** testing the software starting on day one
- **Respect:** every small success deepens their respect for the unique contributions of each and every team member
- **Courage:** deliver the system to the customers as early as possible and implement changes as suggested

Given that this project is a single-person project, neither framework/methodology would work well on it's own, so for this project, it was decided that Scrum would be the main framework, with elements of XP to help strengthen areas such as design and testing.

3.4.1.1 Tools to manage methodology

This project has been chiefly supported by the tool `taiga.io` - a beta web app [10], which aims to promote the use of Scrum and Kanban [44].

Having an online app to organise User Stories, Tasks, Issues and to track progress using a Burndown chart was extremely important in this single-person project, where work was carried out across several different devices and platforms. It also ensures a historical record of what was completed, and when, as is evident from Subsubsection 3.4.1.2.

3.4.1.2 User stories

User Stories are a bid to shift away from talking in technical-jargon, and to shift towards talking in plain english about project requirements. When working with a customer, this is obviously useful, as occasionally they can be non-technical, so this helps promote an open-dialogue between customer and developer, and a clear understanding of the customer's needs.

User Stories typically follow a template for consistency, usually something similar to:

As a <type of user>, I want <some goal> so that <some reason>. [17]

User Stories also have associated Story Points, which is a typically a numbering system leveraged to indicate the effort needed to implement the Story. Due to the uncertain nature of programming, it is not always an accurate reflection of effort, however through the Agile community it is generally accepted that to be consistent in your assignment of points is more useful than being accurate [51]. During the early stages of the project, it is often the case in which estimation is a little off what it should be, however as the project progresses, and the developer gains a better understanding of the tasks, and how to implement them, then estimation tends to become more accurate.

Table 3.1 outlines the User Stories used during this project, along with when they were working upon (during which Sprint) and how many Story Points are associated with it.

Reference	**User Story**	**Milestone**	**Story Points**	**Additional Comments**
1	Clinicians can upload a set of images (MATLAB Command Window) so they can control what images are input into the Congealing Algorithm	Sprint 0	5	
2	Developer will implement membership of a pixel so that Fuzzy Entropy can be calculated	Sprint 1	10	
3	Clinicians can align scans using Non-Probabilistic Entropy so it can be used in the Congealing Algorithm	Sprint 2 & 3	20	Due to complexity of the implementation, this was spread over 2 sprints

Reference	**User Story**	**Milestone**	**Story Points**	**Additional Comments**
4	Clinicians can select an alignment metric (MATLAB Command Window) so they can select which to align the images using	Sprint 4	5	
5	Developer will make standard GUI with no functionality so that this can be demoed as a proof of concept	Sprint 4	5	
6	Clinician can choose number of iterations (MATLAB Command Window) so they can run as many as they want to	Sprint 4	3	
7	Developer will implement Basic mammogram upload so that they can be aligned	Sprint 4	8	
8	Clinicians can align scans using standard Entropy so it can be used in the Congealing Algorithm	Sprint 5	8	
9	Clinicians can upload a set of images - GUI	Sprint 5	10	
10	Developer will optimise membership function so as to improve performance	Sprint 5	2	Promoted from an Issue
11	Developer will optimise Non-Probabilistic Function so as to improve performance	Sprint 5	2	Promoted from an Issue
12	Clinicians can clear an input image so that they can reselect an input image	Sprint 5	3	
13	Clinicians can align scans using Hybrid Entropy so it can be used in the Congealing Algorithm	Sprint 6	20	
14	Clinicians can select an alignment metric from a drop-down menu so it is easy to choose which alignment metric to use	Sprint 6	5	
15	Clinicians can select the number of iterations to be run using an alignment metric (GUI) so it is easy to select how many iterations to run	Sprint 6	5	

Reference	**User Story**	**Milestone**	**Story Points**	**Additional Comments**
16	Clinicians can see meta data about the input image so they can see if the uploaded image is the correct one	Sprint 6	2	
17	Clinicians can see each iteration mean image so they can compare the improvement over each iteration	Sprint 6	3	
18	Clinicians can see adjusted input images on final iteration so they can see how the input images have changed by the final iteration	Sprint 6	3	
19	Developer wants to know why Scans are rotated 90 to left as this is aesthetically displeasing	Sprint 7	8	Promoted from an Issue
20	Developer will research and implement removal of Medical Markers as this causes alignment issues	Sprint 7	5	
21	Clinicians can discard (clear) an alignment so they can start a new alignment	Sprint 8	5	
22	Clinicians can click on average image to view it bigger so they can see the detail easier	Sprint 8	2	
23	Clinicians can save the final mean image with a sensible name so they can easily find it again	Sprint 8	3	
24	Clinicians can see the iteration details so they can understand more about the improvement	Sprint 8	8	
25	Clinicians can see Congealing is running so they know it's in progress	Sprint 8	3	

Table 3.1: User stories defined during the project

3.4.1.3 Burndown chart

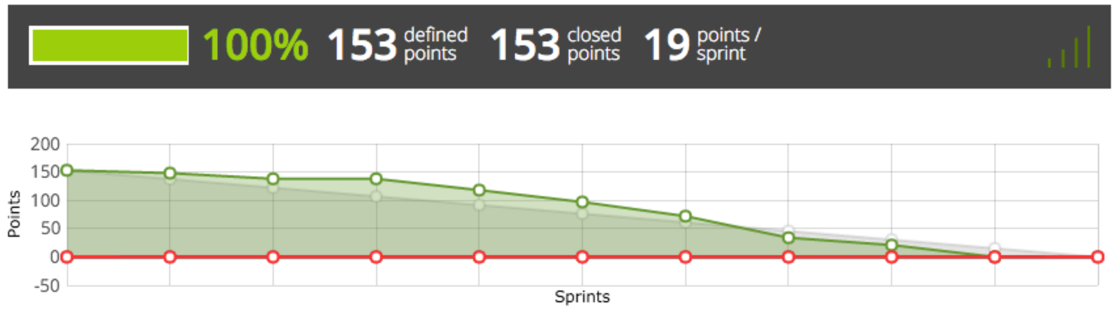


Figure 3.9: Project Burndown chart.

Figure 3.9 is a graphical representation of progress per week, as utilised in Scrum, called a Burndown chart. It allows the developer(s) a quick reference as to the progress of the project, and works by subtracting completed Story points as they're completed. Taiga includes the trend line which sets a target for completion per Sprint.

In Taiga, it was also possible to have a weekly burndown chart, so as to track progress throughout the week, rather than just the entire project.

3.4.1.4 Sprint Review & Retrospective

Sprint Reviews are held at the end of each Sprint, to assess what work was done during the week, and does the end product match the Sprint Goal set out at the start of the week. In this project, Sprint Goals and Sprint Reviews took shape in the form of an informal online blog. Sprints were defined as a week long in this project, running between supervisor meetings (Monday - Sunday). Weekly posts would outline what had been completed that week, how things went (good and bad) and what was to be completed during the following week. Whilst less structured than the conventional approach to Reviews, it works well within a single-person project, and was a good reflection of what had been accomplished.

In Agile Methodologies, Retrospectives are typically at the end of each Sprint, so the team can assess:

- What works well
- What doesn't work well
- What should they start doing

3.4.1.5 Daily Standup

Daily standups are a vital part of Scrum's teamwork ethos. Each morning (or during a set allotted time), the time would meet to discuss what was accomplished the day before, what are the plans for the day ahead, and what road-blocks are in their way. This provides the developer (and further the team) a clear picture of what has yet to be done, and allows fellow team-mates to offer expertise to help overcome obstacles. Whilst this project is not being developed by a team, the benefit of daily standups to productivity, organisation and planning still stands, along with the crowd-sourcing element of expertise.

Throughout the project, stand ups have been held with peers, who're also working upon their Major Projects. Whilst not daily, they tended to fall bi-daily, and it gave the developer a chance to hone skills in explaining the project to people not well-versed in the subject. It was also a good breeding ground for new ideas, and an open forum for discussion into the pros and cons of certain approaches.

3.4.2 Design

In traditional plan-driven methodologies, such as the Waterfall method, Design would take shape in the form of a Design document where all the requirements would be outlined and written up in detail. As mentioned previously, this would be impractical for such a fluid, experimental project, so practices were leveraged from eXtreme Programming (XP) to ensure that the system design was not compromised by the lack of early, solid requirements.

3.4.2.1 CRC Cards

In XP, Class, Responsibilities, and Collaboration (CRC) Cards are an important task in which the entire team can collaborate in the system design. Whilst there is no team in this project, they still play a vital role in structuring the system, and are easily discardable should plans change.

Typically CRC cards would represent Objects, with the class of the written at the top, the responsibilities down the left and the collaborating classes down the right-hand side. However as mentioned in Section 3.2.1, MATLAB is built around a scripting language, and all the "Classes" in this project are replaced by Functions and Scripts. Therefore, each CRC card represents a function or a script, and it's corresponding responsibilities and collaborations as normal - see Appendix **NEED TO REF HERE** for more detail.

Reference appendix

3.4.3 Membership Implementation

As covered in Section 3.1.1, the Membership trapeziums would be defined within the system, rather than dynamically being calculated. This made use of the Fuzzy Logic Toolbox by leveraging the following functions:

- **trapmf:** trapezium-shaped membership function
- **evalmf:** a generic membership evaluation function

Having pre-defined functions for creating and evaluating the membership trapeziums ensured that the function was a quick implementation. However it did reduce the number of Unit Tests I could have written to test the creation of such elements.

In this function, an image is read in, the membership of each pixel is evaluated against each of the two or three membership trapeziums, adding the outcome to a corresponding array (e.g. one array for the membership of all the pixels against the low-trapezium etc). The two/three arrays (depending on number of trapeziums) are then compared and the array which has the greater membership for each pixel, that membership value is added to another array which compiles all the highest values.

For example:

```
pixel[1,1]`s membership in trapezium low = 0.8
pixel[1,1]`s membership in trapezium medium = 0.4
pixel[1,1]`s membership in trapezium high = 0

Therefore as 0.8 is the highest membership value, this is
   added to the image membership array
```

The three/four arrays (one for each trapezium and one for the highest values) are then passed out of the function, and utilised in each of the algorithm functions.

3.4.4 GUI Implementation

Initially the Graphical User Interface (GUI) would be implemented using JavaFX, and any MATLAB additions would be linked in, however it‘s possible to create GUIs easily within MATLAB itself. GUIDE [34] is MATLAB‘s Application development environment where you can either build using purely drag-and-drop techniques, program the application as standard in the editor, or both.

The combination of both the drag-and-drop environment, and manually programming via the editor was undertaken during this project, to allow a greater amount of freedom and flexibility. Drag-and-drop was leveraged to style the GUI and the editor was used to program the functionality in the back-end and link in the Congealing and Fuzzy Entropy algorithms.

3.4.5 Algorithm Implementation

Details of the implementation of the Fuzzy Entropy Algorithms can be found:

Shannon Entropy:

Non-Probabilistic Entropy: Subsection 3.3.2

Hybrid Entropy: Subsection 3.3.3

3.4.6 Testing

Prior to the project beginning, it was outlined that this project should follow a TDD practice. In TDD, the Developer first writes a test, which will fail due to the lack of corresponding functionality. They would then go on to implement the functionality desired by the test. Finally, any refactoring of the initial test and/or code would take place.

However due to the nature of the research, it became increasingly more difficult to follow given the research which had to be undertaken alongside development. This led to a change from TDD to Retrospective Testing, in which I would write all tests, post functional implementation. This is a more traditional approach to testing, and still catches the same errors which might occur during TDD.

3.4.6.1 Unit Tests

Unit Tests were completed using MATLAB‘s Unit Testing Framework [53], which covers all the ways in which you can program in MATLAB:

- Script-Based Unit Tests
- Function-Based Unit Tests
- Class-Based Unit Tests

The majority of my work in MATLAB was Function-based, so this was the style followed for unit tests.

1 Name	2 Passed	3 Failed	4 Incomplete	5 Duration
'TESTnonProb/testNonProbEntropyLessThanOne'	1	0	0	1.8517
'TESTnonProb/testNonProbEntropyLargerThanZero'	1	0	0	1.8428
'TESThybrid/testHybridEntropyLessThanOne'	1	0	0	0.1140
'TESThybrid/testHybridEntropyLargerThanZero'	1	0	0	0.1190
'TESTpgmCreation/testImageType'	1	0	0	0.6913
'TESTpgmCreation/testCommentInsertion'	1	0	0	0.5880
'TESTpgmCreation/testImageSize'	1	0	0	0.6043
'TESTpgmCreation/testGreyScale'	1	0	0	0.5877
'TESTmembership/test3TrapeziumMembershipDegreeNotEmpty'	1	0	0	0.0828
'TESTmembership/testCalulatedMembershipDegreeForAllPixels3Trapeziums'	1	0	0	0.0841
'TESTmembership/test2TrapeziumMembershipDegreeNotEmpty'	1	0	0	0.0731
'TESTmembership/testCalulatedMembershipDegreeForAllPixels2Trapeziums'	1	0	0	0.0748
'TESTmembership/test3TrapeziumMembershipDegreeLessThanOne'	1	0	0	0.0810
'TESTmembership/test2TrapeziumMembershipDegreeLessThanOne'	1	0	0	0.0799
'TESTmembership/test3TrapeziumMembershipDegreeGreaterThanZero'	1	0	0	0.0798
'TESTmembership/test2TrapeziumMembershipDegreeGreaterThanZero'	1	0	0	0.0734

Figure 3.10: Results from MATLAB Unit Tests.

As Figure 3.10 demonstrates, all Unit tests passed.

3.4.6.2 Acceptance Tests

Table 3.2 outlines the results from the Acceptance Tests run. The left-most column "User Story Reference" aligns with Table 3.1, so more detail can be found.

User Story Reference	Expected Outcome	Actual Outcome	Pass/Fail
1	Image is loaded into the system	As expected	Pass
2	Membership array is passed out of the membership function and is usable in other functions	As expected	Pass
3	Images are aligned using Non-Probabilistic entropy and the output & entropy outputs are realistic	As expected	Pass
4	Images are aligned using the metric the user has selected when running the function	As expected	Pass
6	The number of iterations is run as specified by the User, then function stops	As expected	Pass
8	Images are aligned using Shannon entropy and the output & entropy outputs are realistic	As expected	Pass
9	Image(s) selected to be loaded into the GUI is displayed	As expected	Pass
12	Image box where input image appears goes blank after Clear button is selected	As expected	Pass
13	Images are aligned using Hybrid entropy and the output & entropy outputs are realistic	As expected	Pass
14	Images are aligned using the chosen alignment metric	As expected	Pass
15	The number of iterations is run as specified by the User, then function stops	As expected	Pass
16	When an input image is loaded in, Metadata is displayed in the GUI about the image	As expected	Pass

17	After Congealing, the user can press the “See all Mean images” button and a new Figure displays the mean image after each iteration	As expected	Pass
18	After Congealing, the user can press the “See Adjusted Inputs” button and a new Figure displays the adjusted input images after the final iteration	As expected	Pass
21	Image box where output image appears goes blank after Clear button is selected along with all other fields	As expected	Pass
22	Image is displayed larger in a new Figure	As expected	Pass
23	Save file dialog appears with a sensible name suggested (i.e. final image - alignment-chosen - number of iterations)	As expected	Pass
24	When “Entropy details” button is selected, a new Figure appears with a graph showing entropy decrease. Final Entropy & time taken also displays in the main GUI	As expected	Pass
25	Egg-timer appears when Congealing Algorithm is running	As expected	Pass

Table 3.2: Acceptance Test results

3.5 Technical Difficulties

3.5.1 Image Rotation

One issue which was faced when creating the large .pgm file containing all the input images, was that they were rotated 90°to the right, as demonstrated in Figure 3.11.

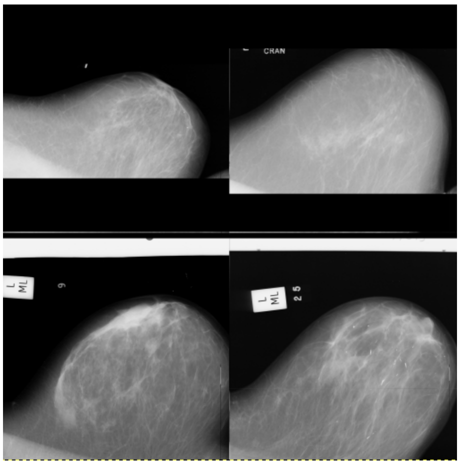


Figure 3.11: 4 rotated input images concatenated into one larger image.

It quickly became apparent that the order in which the image array was being written to file to create this larger pgm file was incorrect, however due to MATLAB's clever use of vectorisation, it was difficult to diagnose where the issue lay. After some investigation, it was revealed that the function `fwrite` by MATLAB [32], used for writing binary data to file, wrote each line out column-by-column, rather than the customary row-by-row approach.

To mitigate this issue, the array passed into fwrite would have to be transposed prior or during being passed into the function. There are two ways in which MATLAB permits the transposition of arrays:

3.5.1.1 Simple 2D array transposition

MATLAB has a "Transpose" function which simply flips two elements in a 2D array as utilised in:

```
fwrite(output,handles.finalImg.','uchar');
```

Where handles.finalImg is a GUI holder for a 2D array of pixel values. This example was taken from the `removeMarker.m` function - where the user can remove Medical Markers and save the output back to the original file.

3.5.1.2 3D+ array transposition

For arrays with more than 2 dimensions, simply swapping the values around will not work, so the MATLAB function `permute` [36] must be used.

```
  sers=zeros(squareImageSize(1),squareImageSize(2),noOfScans); %
     set size of array

  for i = 1:noOfScans

    scan = fopen(strcat(pathname,'/',scanDirectory(i).name)); %
       open each input image individually
    im=(fread(scan,[squareImageSize(1),squareImageSize(2)],'
       uchar'));
    sers(:,:,i) = im; %add each input image to a 3D array which
       compiles all the input images into one

  end

outfname=sprintf('%s/big_scan.pgm', pathname);
s=sers(:,:,:);
saveSeries(outfname,permute(s,[2,1,3])); %use the saveSeries
   demo function to write the final image arrays out to a file
```

This example was taken from the `pgm2bigPgm.m` function - where a set of input images are passed in, and a large pgm image containing all the input images is outputted (as in Figure 3.12). This image is then passed into the Congealing algorithm for alignment.

3.5.1.3 Final Outcome

After transposing all arrays which are to be saved out to file, whether directly through `fwrite` or `saveSeries`, all images are saved in the correct orientation.

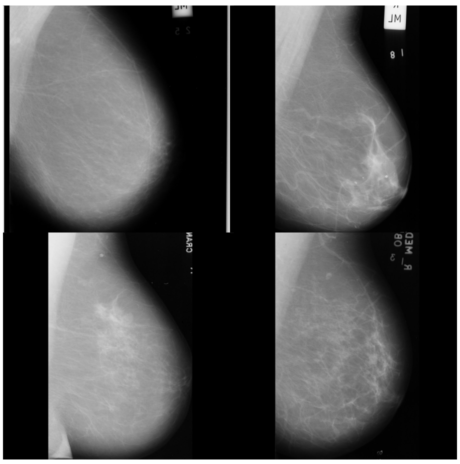


Figure 3.12: Final output image.

3.5.2 Medical Marker Removal

This subsection has been formalised from a blog post written on 28th March 2016 [18].

As the images are aligned using a comparison of the pixel-value, the Medical Markers included on mammograms cause an issue. This is because if more than one scan contains these white patches (left by the metal clip during scanning), then they will try and align with each other during the Congealing process.

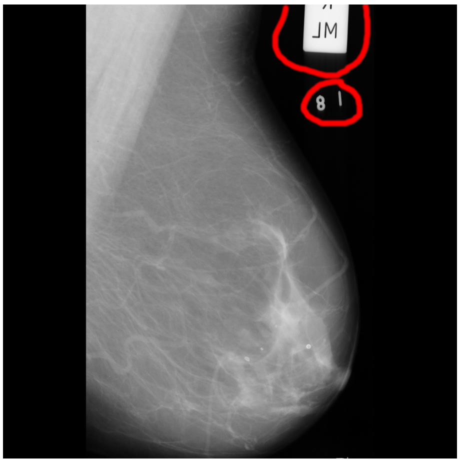
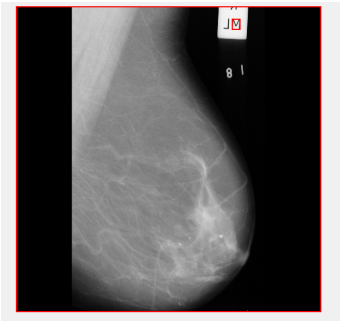


Figure 3.13: Image containing Medical Markers

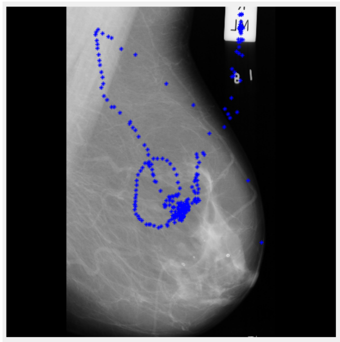
Two options were available for the avoidance of Medical Markers:

- Ask the User not to use scans containing Medical Markers
  - This is extremely restrictive
  - This could massively reduce their number of usable scans
- Find a computer vision and/or image processing technique to remove these clips
  - Preferably automatically
  - Manually removing would work for small input data sets

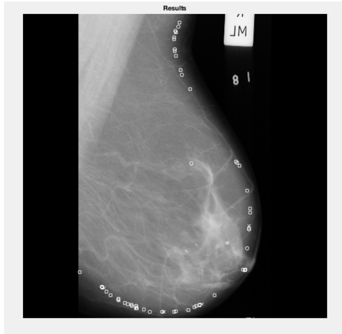
3.5.2.1 Discarded ideas



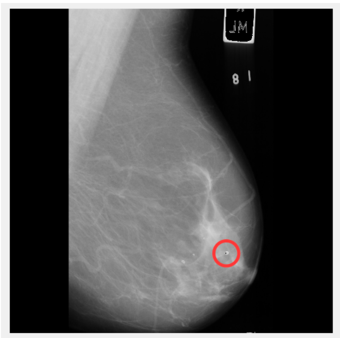
(a) Squares detected in a mammogram using morphological operations.



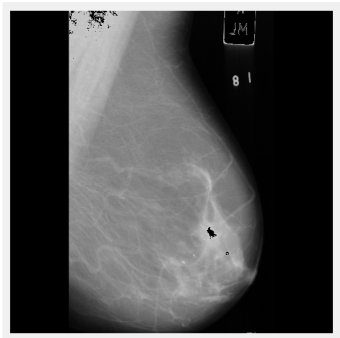
(b) Regionprops detecting areas and superimposing the area information on a mammogram.



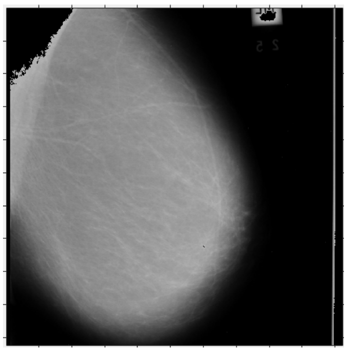
(c) Shape recognition picking out the rough boundary of breast tissue.



(d) Removing areas of grey-level value above 220.



(e) Lower grey-level removal threshold to 200.



(f) Applying removal over 220-value to another scan.

Figure 3.14: Output of discarded methods of marker removal

Morphological operation - remove squares from image

Utilising a morphological operation, such as the one demonstrated by Chandra Kurniawan in the thread [27]. However, as can be seen in Figure 3.14a, not only is the Marker itself not perfectly square, but because the image is square, it detects that instead. So removal would be made more difficult by the fact you would have to specify a maximum size square to remove, that smaller than the image size.

This idea was discarded due to the marker unlikely to ever be perfectly square in the scan.

MATLAB function regionprops

Another candidate function for removing Medical Markers was the MATLAB function `regionprops` [37]. The idea behind using this function would be to measure the area of the squares in the image, so then they could be removed. However, the output, as seen in Figure 3.14b, was not something desired, and without spending an inordinate amount of time tweaking the function, it is not useful to the detection of the markers.

Shape recognition demo

On the Mathworks File Exchange site, a community run to help MATLAB users, there was a demo created by Ahmed Samieh to aid in the recognition of certain shapes [47]. It classifies the shape by properties such as roundness, ratios of dimensions and centroids.

Modifying this demo slightly to make it compatible with the grey-scale mammograms, the output is somewhat promising, as seen in Figure 3.14c.

However, due to the slightly inaccurate identification of the tissue boundary, this is likely to remove data which is useful to the Congealing algorithm. Unless this can become a near perfect outline around the breast tissue, it is unlikely to be useful for selecting and focusing in the object of interest.

Removing white objects over a specified grey-level value

Back on the Mathworks forum, there is a thread about removing white glare from a jewellery photo [11]. This was adapted to detect the medical marker by specifying to find and remove patches over 220 grey-level value. As seen in Figure 3.14d, most of the marker has been removed, however it also removes a small bit of breast tissue.

To see if the entire marker could be removed, if you lower the grey-level threshold for removal to anything over 200 value, then the output is as in Figure 3.14e. Unfortunately it does not remove the entire marker, and some of the vital breast tissue is lost.

Further to that, by running the white removal at grey-level value at 220 (the suitable choice for my first test scan) on another test scan and absolutely nothing is removed. Lower the threshold to begin removing white areas (down to grey-level value of 180) the results are less desirable, as demonstrated in Figure 3.14f.

3.5.2.2 Chosen method

Another demo on the MATLAB forum outlined a way in which a user can draw an area to remove, then a mask can be applied over the top to hide any problem areas (such as the Medical Markers) [9].

After reading through the demo given as an answer by "Image Analyst" on the forum, I rewrote the function in order to fit the removal criteria. The User can utilise the MATLAB function `imfreehand` [35] to draw over the input image in order to indicate the area to be removed. This area is then filled in with the darkest grey-level value found in the drawn area (typically 0 for black, however may differ between scans).

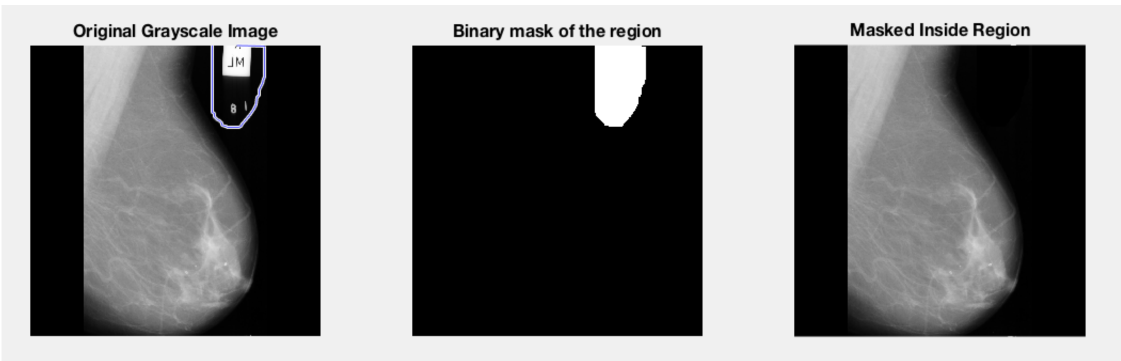


Figure 3.15: Image depicting the steps taken to remove Medical Markers from a scan.

As shown in Figure 3.15 this has been shown to be extremely successful and therefore was utilised in the project.

3.5.3 .pgm file header

As this project was building upon the work done by Learned-Miller [29], it was useful to utilise the load function that was already in place. However, the nature in which the image files were loaded into the system caused some unexpected hurdles.

In order to understand how the demo data was uploaded, and therefore implement a function to compile a large set of mammogram scans in the correct format, research was carried out into the nature of PGM files, and the function which comments play in the headers.

3.5.3.1 PGM file format

Portable Gray Map (PGM) file format is part of a package called Netpbm, which contains 220 separate programs for dealing with files such as PGM, pbm and pnm. As the name suggests, it is a lightweight-greyscale image format, which is simple for use in programs, making it ideal for this project.

The structure of PGM files is very specific and is defined as [45]:

1. A "magic number" for identifying the file type. A pgm image's magic number is the two characters "P5".
2. Whitespace (in the format of tab, space etc)
3. A width, formatted as ASCII characters in decimal.
4. Whitespace (in the format of tab, space etc)
5. A height (in the same format as width)
6. Whitespace (in the format of tab, space etc)
7. Maximum Grey Value (Maxval) - usually 255
8. Single whitespace character (typically new line)
9. A raster of Height rows, in order from top to bottom. Each row consists of Width gray values, in order from left to right. Each gray value is a number from 0 through Maxval, with 0 being black and Maxval being white. Each gray value is represented in pure binary by either 1 or 2 bytes. If the Maxval is less than 256, it is 1 byte. Otherwise, it is 2 bytes. The most significant byte is first.

A comment in PGM is proceeded by the # symbol, and is not counted in the above formatting.

3.5.3.2 Specific file format for Congealing

When investigating the Congealing demo code, it became apparent that comments were utilised in the reading-in of image information.

Listing 3.1: Example MNIST PGM file header

```
P5
# 28 28 6742
2324 2324
255
```

Listing 3.1 above shows the first 5 lines of the PGM MNIST data which was included in the Congealing demo. The second line, proceeded by a # - therefore a comment, includes information on height and width of each individual MNIST number (28 and 28), and how many of these numbers are included in the large file (6742).

This information is then used to set the number of images per row and to set an array to the appropriate height, width and number of included images in the `loadSeries` function.

3.5.3.3 Creating an appropriate save function

The next step was to write a function which would appropriately concatenate the MINI-MIAS dataset [54] to create a large PGM input image for Congealing. This led to the function `pgm2bigPgm.m`, which is a refined version of the original `saveSeries.m` demo function, which will:

- read in the number of images in the chosen directory
- identifies the dimensions of each scan in the directory (with MINI-MIAS, they are all the same dimensions)
- creates a string containing all the suitable information needed for reading (as outlined in Subsubsection 3.5.3.2)
- creates a file called "big_scan.pgm" and saves all the images out to the one file (after transposition, as in Subsection 3.5.1)

3.5.4 Vectorisation

Vectorisation is the process to replace loop-based code with MATLAB matrix and vector operations. As stated in the MATLAB documentation [38], Vectorisation is important for several reasons:

1. Appearance - more concise, more like what is seen in textbooks
2. Less Error Prone - less for loops = less lines of code for errors to appear
3. Performance - vectorised code usually runs a lot faster

The initial implementations of both `membership.m` and `deLucaFuzzy.m` contained for loops, so experimentation was run before, during and after vectorisation to evaluate the supposed performance increase.

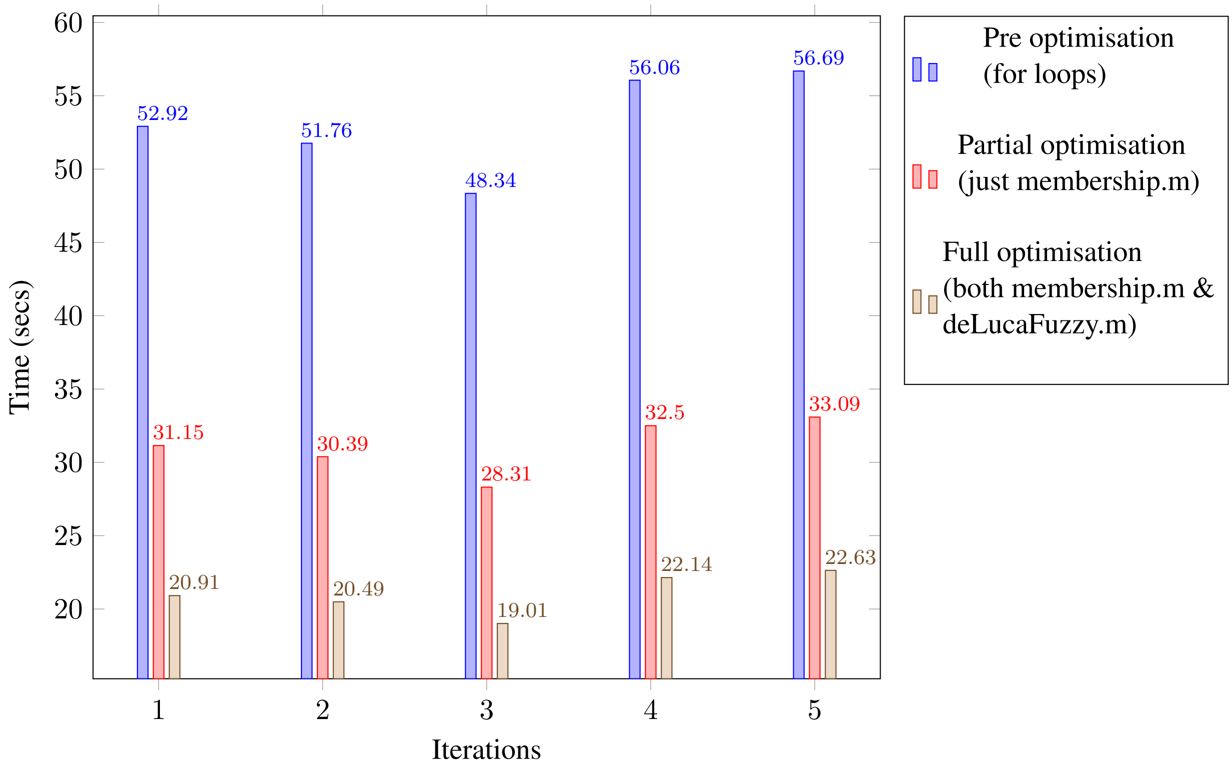


Figure 3.16: Time per iteration before, during and after vectorisation

Figure 3.16 demonstrates the time taken per iteration, in the same environment, to run the `binaryCongeal.m` [1] function on each iteration. A marked improvement can be seen just by vectorising the `membership.m` function, and further improvements once the `deLucaFuzzy.m` function for Non-Probabilistic entropy was vectorised.

[1]The function which calls the specified entropy algorithm.

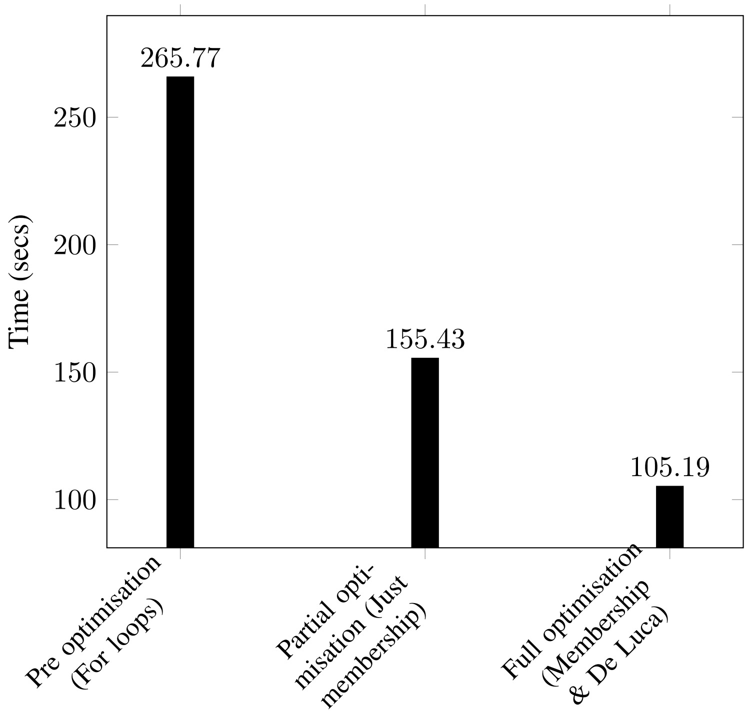


Figure 3.17: A comparison of the total time to run 5 iterations prior to vectorisation, during (part vectorisation) and post-vectorisation.

Figure 3.17 outlines the total time taken to run 5 iterations of Non-Probabilistic Entropy before vectorisation, once vectorisation was complete on the `membership.m` function, and finally after full vectorisation.

Chapter 4

Results and Conclusions

4.1 Results

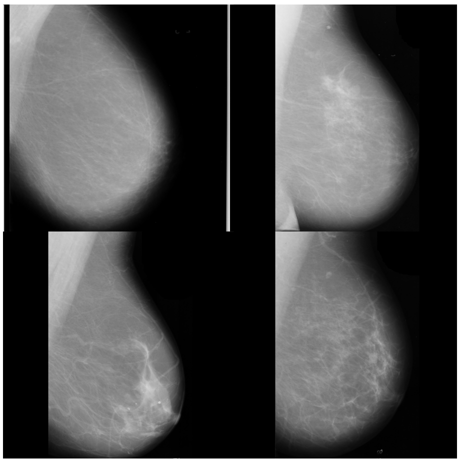


Figure 4.1: 4 input scans.

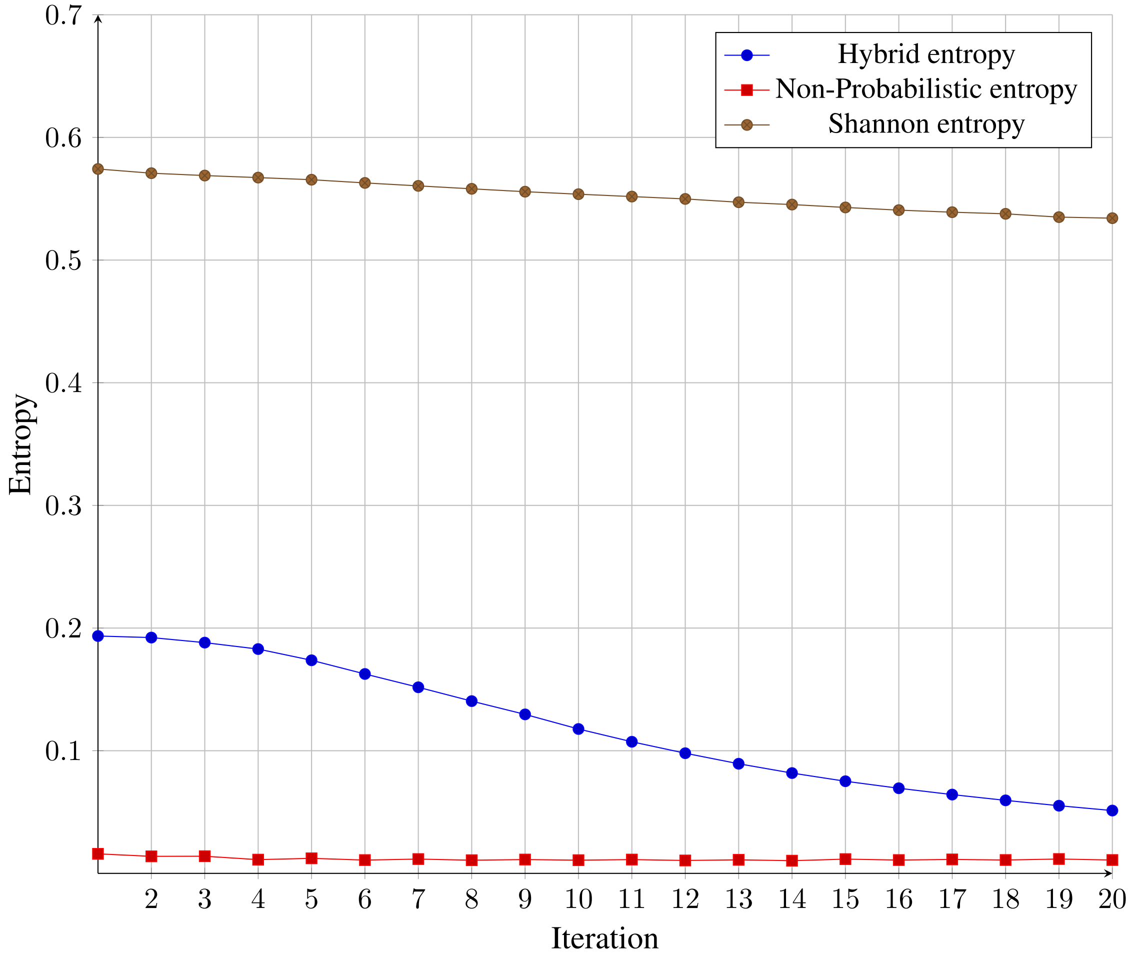
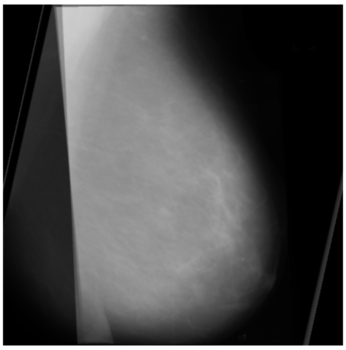
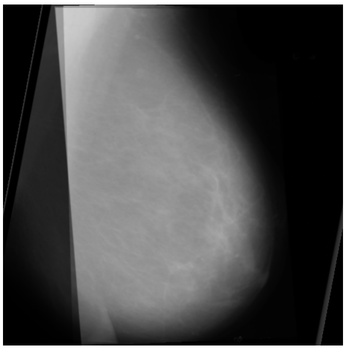


Figure 4.2: Comparison of the reduction in entropy over iterations.

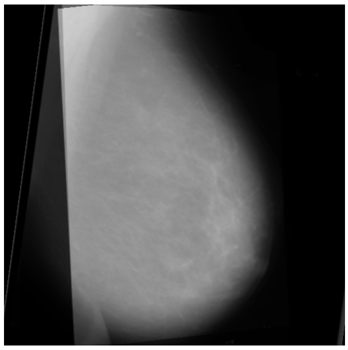
4.1.1 Shannon Entropy



(a) 5 Shannon Entropy iterations.



(b) 10 Shannon Entropy iterations.



(c) 20 Shannon Entropy iterations.

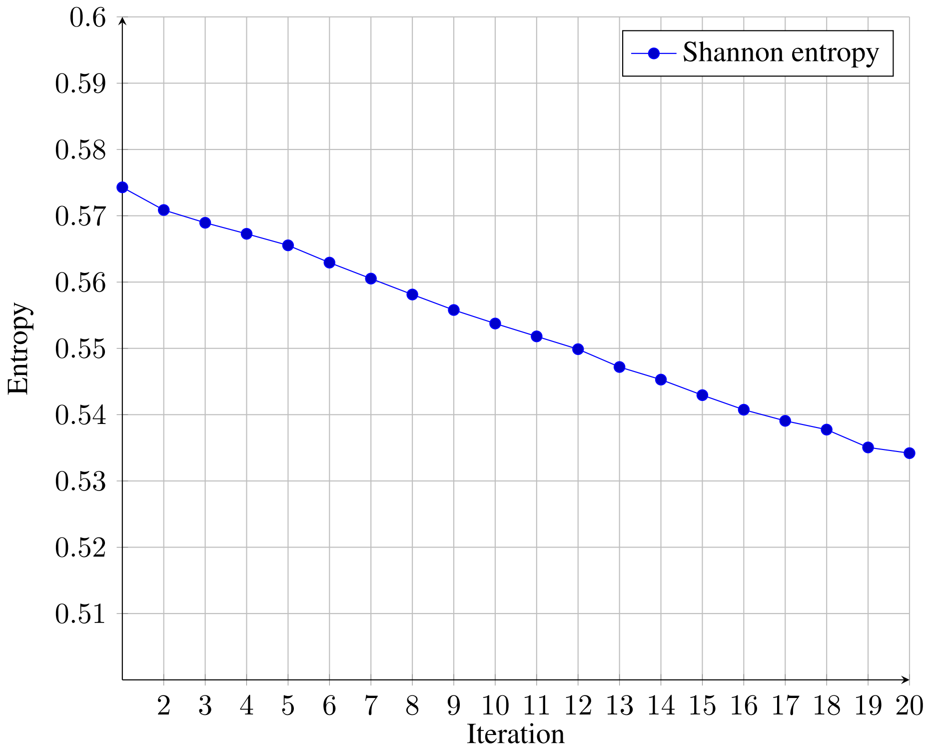
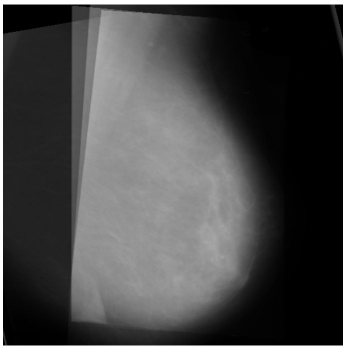


Figure 4.4: Shannon: Comparison of the reduction in entropy over iterations, as in Table 4.1.

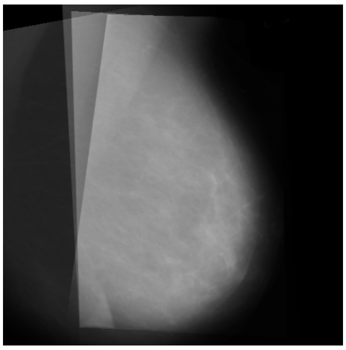
Iteration	Entropy
1	0.574292
2	0.570867
3	0.568946
4	0.567278
5	0.565542
6	0.562933
7	0.560522
8	0.558121
9	0.555785
10	0.553746
11	0.551807
12	0.549876
13	0.547188
14	0.545286
15	0.542942
16	0.540746
17	0.539067
18	0.537744
19	0.535056
20	0.534195

Table 4.1: Entropy table for Shannon

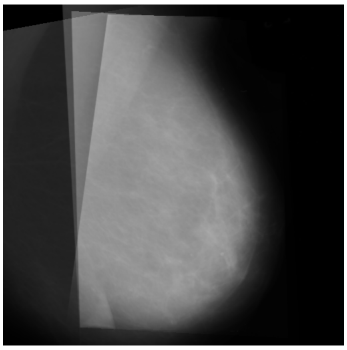
4.1.2 Non-Probabilistic Entropy



(a) 5 Non-Probabilistic entropy iterations.



(b) 10 Non-Probabilistic entropy iterations.



(c) 20 Non-Probabilistic entropy iterations

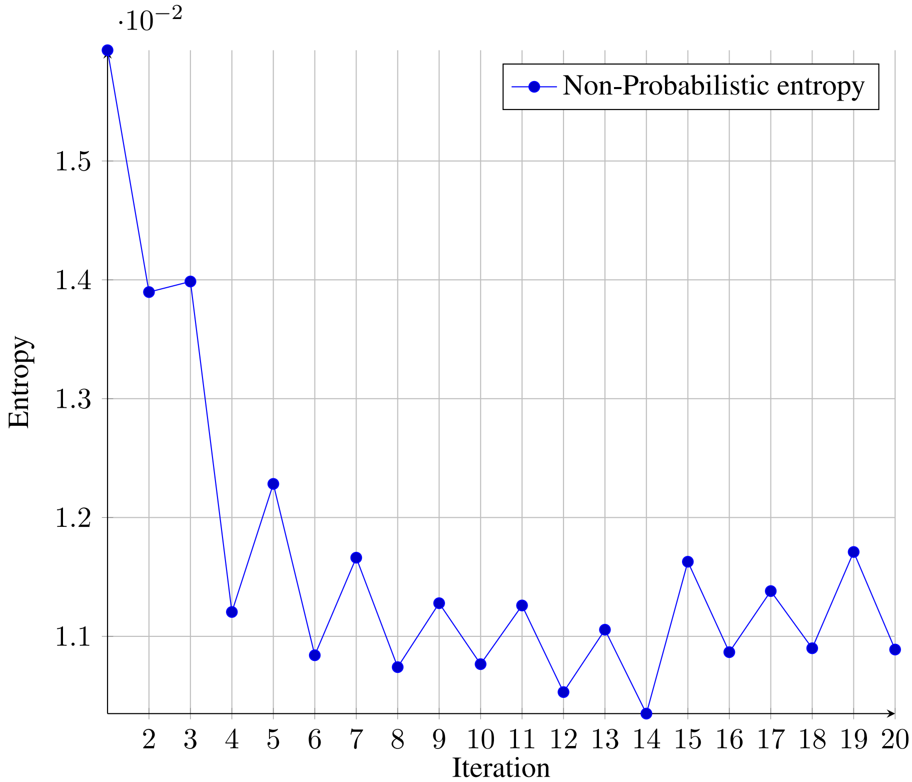
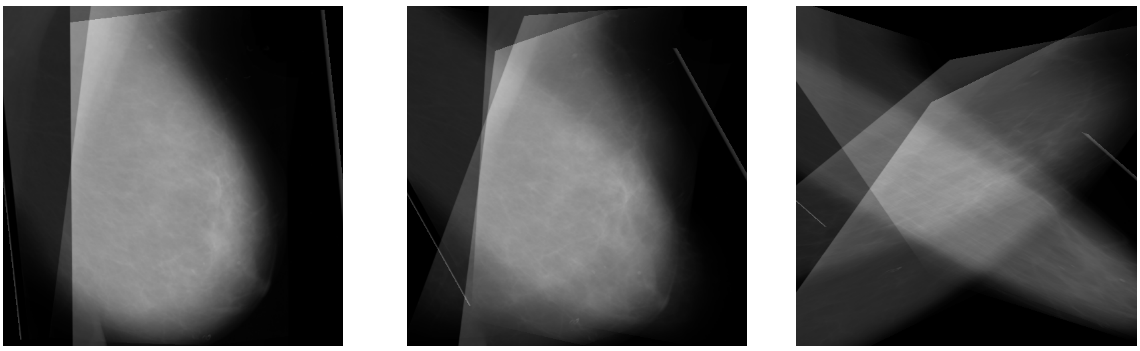


Figure 4.6: Non-Probabilistic: Comparison of the reduction in entropy over iterations, as in Table 4.2.

Iteration	Entropy
1	0.015932
2	0.013897
3	0.013986
4	0.011205
5	0.012283
6	0.010841
7	0.011662
8	0.010741
9	0.011279
10	0.010766
11	0.01126
12	0.010531
13	0.011056
14	0.01035
15	0.011628
16	0.010867
17	0.011381
18	0.0109
19	0.01171
20	0.010889

Table 4.2: Entropy table for Non-Probabilistic

4.1.3 Hybrid Entropy



(a) 5 Hybrid entropy iterations. (b) 10 Hybrid entropy iterations. (c) 20 Hybrid entropy iterations.

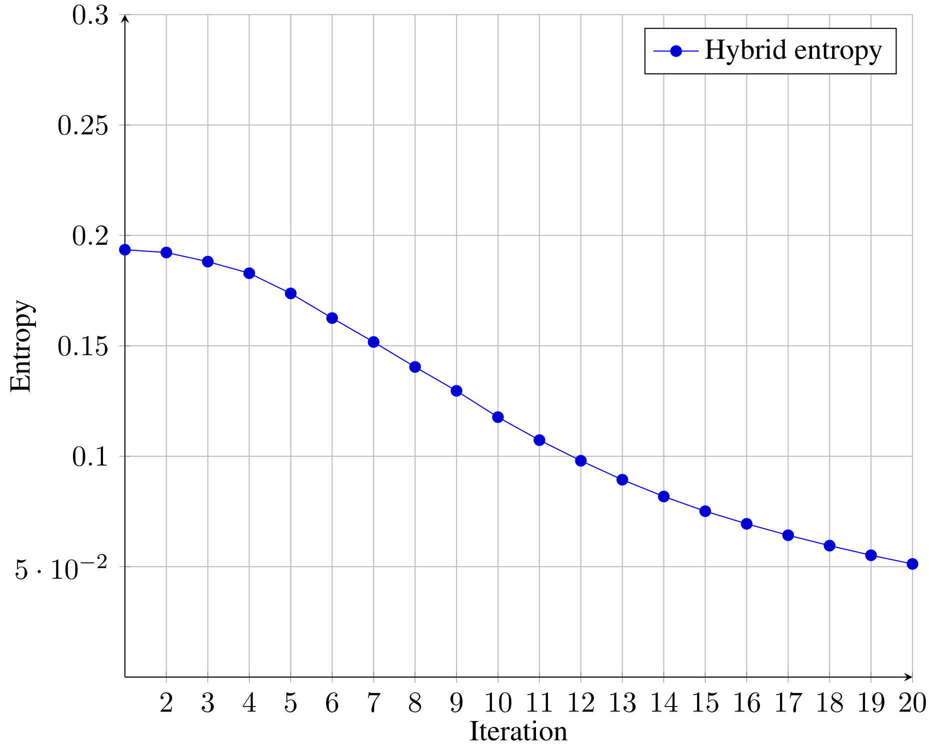


Figure 4.8: Hybrid: Comparison of the reduction in entropy over iterations, as in Table 4.2.

Iteration	Entropy
1	0.193522
2	0.192257
3	0.188139
4	0.182887
5	0.173737
6	0.162572
7	0.151736
8	0.140438
9	0.12962
10	0.117715
11	0.107298
12	0.097968
13	0.089378
14	0.08178
15	0.075122
16	0.069405
17	0.064241
18	0.059527
19	0.055169
20	0.051228

Table 4.3: Entropy table for Hybrid

Chapter 5

Critical Evaluation

Appendices

Appendix A

Third-Party Code and Libraries

1.1 Congealing Code

The project focused on extending the existing Congealing Code implemented by Learned Miller et al in 2005. A Congealing demo is available on the Congealing website [28] which is open for experimentation. The original demo code was modified and extended to be able to read in mammograms and to work with 2 Fuzzy Entropy algorithms.

Appendix B

Ethics Submission

2.1 Ethics Application Number: 3958

AU Status
Undergraduate or PG Taught

Your aber.ac.uk email address
lac32@aber.ac.uk

Full Name
Laura Collins

Please enter the name of the person responsible for reviewing your assessment.
Reyer Zwiggelaar

Please enter the aber.ac.uk email address of the person responsible for reviewing your application
rrz@aber.ac.uk

Supervisor or Institute Director of Research Department
cs

Module code (Only enter if you have been asked to do so)
CS39440

Proposed Study Title
Entropy based metrics for joint image alignment

Proposed Start Date
25th January 2016

Proposed Completion Date
4th May 2015

Are you conducting a quantitative or qualitative research project?
Mixed Methods

Does your research require external ethical approval under the Health Research Authority?
No

Does your research involve animals?
No

Does your research involve human participants?
Yes

Are you completing this form for your own research?
Yes

Does your research involve human participants?
Yes

Institute
IMPACS

Please provide a brief summary of your project (150 word max)
I will be investigating the use of Congealing multiple MIAS dataset mammograms using several fuzzy entropy alignment metrics. If time permits I plan on speaking to a specialist (radiologist) to determine whether the output mean images of the congealing process are of any significant use to the research into breast cancer detection.

I can confirm that the study does not involve vulnerable participants including participants under the age of 18, those with learning/communication or associated difficulties or those that are otherwise unable to provide informed consent?
Yes

I can confirm that the participants will not be asked to take part in the study without their consent or knowledge at the time and participants will be fully informed of the purpose of the research (including what data will be gathered and how it shall be used during and after the study). Participants will also be given time to consider whether they wish to take part in the study and be given the right to withdraw at any given time.
Yes

I can confirm that there is no risk that the nature of the research topic might lead to disclosures from the participant concerning their own involvement in illegal activities or other activities that represent a risk to themselves or others (e.g. sexual activity, drug use or professional misconduct). Should a disclosure be made, you should be aware of your responsibilities and boundaries as a researcher and be aware of whom to contact should the need arise (i.e. your supervisor).
Yes

I can confirm that the study will not induce stress, anxiety, lead to humiliation or cause harm or any other negative consequences beyond the risks encountered in the participant's day-to-day lives.
Yes

Please include any further relevant information for this section here:

Where appropriate, do you have consent for the publication, reproduction or use of any unpublished material?
Yes

Will appropriate measures be put in place for the secure and confidential storage of data?
Yes

Does the research pose more than minimal and predictable risk to the researcher?
No

Will you be travelling, as a foreign national, in to any areas that the UK Foreign and Commonwealth Office advise against travel to?
No

Please include any further relevant information for this section here:

If you are to be working alone with vulnerable people or children, you may need a DBS (CRB) check. Tick to confirm that you will ensure you comply with this requirement should you identify that you require one.
Yes

Declaration: Please tick to confirm that you have completed this form to the best of your knowledge and that you will inform your department should the proposal significantly change.
Yes

Please include any further relevant information for this section here:

Appendix C

Code Examples

Appendix D

CRC Cards

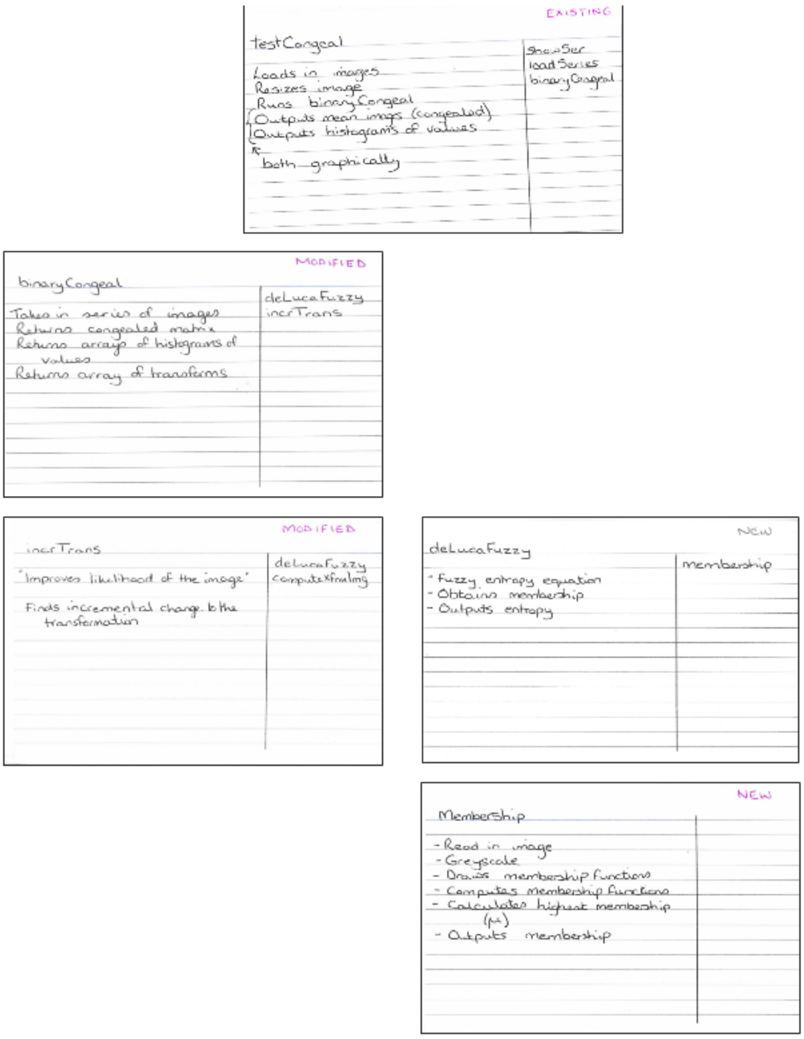


Figure D.1: Initial design

Figure D.1 details the initial planned design for the implementation of De Luca & Termini's Non-Probabilistic entropy.

Glossary

Congealing is an algorithm concerned with joint image alignment, developed by Learned-Miller [29].

CRC Class, Responsibilities, and Collaboration.

GUI Graphical User Interface.

PGM Portable Gray Map.

TDD Test Driven Development.

transposition to change the order of two or more objects.

XP eXtreme Programming.

Annotated Bibliography

[1] [Online]. Available: http://www.merriam-webster.com/dictionary/entropy

Definition of Entropy with regards Information Theory (and thermodynamics)

[2] "The agile movement." [Online]. Available: http://agilemethodology.org/

Site dedicated to the Scrum.

[3] "The anatomy and physiology of the breast." [Online]. Available: http://www.cancer.ca/en/cancer-information/cancer-type/breast/anatomy-and-physiology/?region=on

Website outlining the make up of breast structure.

[4] "Definition of indiscernibility." [Online]. Available: http://www.merriam-webster.com/dictionary/indiscernibility

Helps define Indiscernibility uncertainty.

[5] "Definition of indiscernible." [Online]. Available: http://www.merriam-webster.com/dictionary/indiscernible

Helps define Indiscernibility uncertainty.

[6] "Definition of possibility." [Online]. Available: http://www.merriam-webster.com/dictionary/possibility

Helps define Possibilistic uncertainty

[7] "Definition of probability." [Online]. Available: http://www.merriam-webster.com/dictionary/probability

Helps define Probabilistic Entropy

[8] "Manifesto for agile software development." [Online]. Available: http://agilemanifesto.org/

Agile manifesto.

[9] "Masking out image area using binary mask - matlab answers - matlab central." [Online]. Available: http://uk.mathworks.com/matlabcentral/answers/38547-masking-out-image-area-using-binary-mask

Chosen method for removing Medical Markers - draw an outline, create a mask and overlay the mask over the drawn area.

[10] "Taiga - love your projects." [Online]. Available: https://taiga.io/

Tool used to promote a scrum methodology

[11] "Want to remove white patches of a image - matlab answers - matlab central." [Online]. Available: http://uk.mathworks.com/matlabcentral/answers/141008-want-to-remove-white-patches-of-a-image

Thread on removing white objects/glare from an image - useful for removing medical markers from mammograms.

[12] "Basic information about the typical mlo and cc views of mammography," Apr 2016. [Online]. Available: http://breast-cancer.ca/mammopics/

An informal website outlining some information about MLO and CC breast scans.

[13] I. C. about Cancer Screening, "Nhs breast screening: Helping you decide," Jun 2013. [Online]. Available: http://www.breasttestwales.wales.nhs.uk/sitesplus/documents/1025/1554%20Yogi%20-%20Breast%20Screening%20English%20v4.pdf

Leaflet about Breast Screening created by Informed Choice about Cancer Screening with Cancer Research UK

[14] S. Al-Sharhan, F. Karray, W. Gueaieb, and O. Basir, "Fuzzy entropy: a brief survey," in *Fuzzy Systems, 2001. The 10th IEEE International Conference on*, vol. 3. IEEE, 2001, pp. 1135–1139. [Online]. Available: http://dx.doi.org/10.1109/fuzz.2001.1008855

Paper outlining the different implementations of Fuzzy Entropy, of which 3 will be selected and focused on during this Project.

[15] N. F. Boyd, J. W. Byng, R. A. Jong, E. K. Fishell, L. E. Little, A. B. Miller, G. A. Lockwood, D. L. Tritchler, and M. J. Yaffe, "Quantitative classification of mammographic densities and breast cancer risk: Results from the canadian national breast screening study," *Journal of the National Cancer Institute*, vol. 87, no. 9, p. 670675, May 1995, pMID: 7752271.

This paper clearly outlines significant links between breast density and breast cancer risk.

[16] J. L. Champaign and G. J. Cederbom, "Advances in breast cancer detection with screening mammography," *The Ochsner Journal*, vol. 2, no. 1, p. 33, Jan 2000, pMID: 21765659.

Paper outlining the advancements in screening mammography in the years up to 2000.

[17] M. Cohn, "User stories and user story examples by mike cohn." [Online]. Available: https://www.mountaingoatsoftware.com/agile/user-stories

User Story example given in the Software section

[18] L. Collins, "Hurdle #4 - removing labels from mammograms," Mar 2016. [Online]. Available: http://lauramcollins.co.uk/blog/removing-labels-from-mammograms/

My personal blog post containing information about the methods in which I tried to remove Medical Markers from the data set of mammograms.

[19] E. Commission, "Health statistics - atlas on mortality in the european union - product - eurostat," 2009. [Online]. Available: http://ec.europa.eu/eurostat/en/web/products-statistical-books/-/KS-AC-04-000

2009 paper published by the European Commission on statistics into mortality rates and causes in the EU.

[20] M. Cox, S. Sridharan, S. Lucey, and J. Cohn, *Least squares congealing for unsupervised alignment of images*, Jun 2008, p. 18.

A disregarded adaption of the Congealing algorithm - however was useful in highlighting performance issues in the original algorithm. Something which was near continuously faced when implementing heavier fuzzy entropy alignment metrics.

[21] A. De Luca and S. Termini, "A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory," *Information and Control*, vol. 20, no. 4, p. 301312, May 1972.

De Luca & Termini's 1972 paper focuses on their definition of Non-probabilistic entropy, along with several example derivations.

[22] M. H. DeGroot, *Optimal Statistical Decisions*. Reprinted: McGraw-Hill, 2004.

DeGroot gave a simple explanation about uncertainty in terms of probability.

[23] K. GEORGE J. and Y. BO, "Fuzzy sets and fuzzy logic, theory and applications," -, Mar 2008. [Online]. Available: http://digilib.uin-suka.ac.id/7049/

[24] I. T. Gram, E. Funkhouser, and L. Tabár, "The Tabár classification of mammographic parenchymal patterns. - pubmed - ncbi." [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9097055

Tabár method for categorising scans using an anatomic-mammographic comparison technique.

[25] H. Gray, *Anatomy, Descriptive and Surgical*. Philadelphia: Lea & Febiger, 1907, 1-58734-102-6.

Extremely influential book first written in 1858 by Henry Gray. It has been revised and republished, up until September 2015. Whilst this cited version is not 100% medically accurate by today's standards, the diagrams still hold true.

[26] G. B. Huang, V. Jain, and E. Learned-Miller, *Unsupervised Joint Alignment of Complex Images*. IEEE, Oct 2007, p. 18. [Online]. Available: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=4408858

Further work done upon Congealing by a team including Learned-Miller. This was disregarded for the project due to the complexity added for inputting complex images - something mammograms are not.

[27] C. Kurniawan, "Detect square in image." [Online]. Available: http://uk.mathworks.com/matlabcentral/answers/24943-detect-square-in-image

Mathworks forum thread on how to detect squares in an image using a morphological operation. This was investigated for possible use in medical marker removal.

[28] E. Learned-Miller, “The congealing page.” [Online]. Available: https://people.cs.umass.edu/~elm/congealing/

This webpage was the source of the demo Congealing code modified and built upon in this project. It also contains several useful papers where Congealing has been used in research - such as Learned-Miller’s own paper on Congealing MNIST handwriting data and MRI scans.

[29] E. G. Learned-Miller, “Data driven image models through continuous joint alignment,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 2, pp. 236–250, Feb. 2006. [Online]. Available: http://dx.doi.org/10.1109/tpami.2006.34

Learned-Miller’s original Congealing method is the basis for this Project - however I am looking to further extend the alignment capabilities using fuzzy entropy metrics, rather than standard Shannon entropy as currently implemented. This paper was extremely useful for understanding of the basic concepts behind it, and will be a good reference guide throughout the project.

[30] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*. Morgan Kaufmann Publishers Inc., Aug 1981, p. 674679. [Online]. Available: http://dl.acm.org/citation.cfm?id=1623264.1623280

Lucas and Kanade approach to aligning images using gradient-descent - this is what Cox et al based their Least Squares Congealing algorithm off of.

[31] N. Mac Parthalain and H. Strange, *Fuzzy-entropy based image congealing*, Jul 2013, p. 18.

Paper outlining a Fuzzy Entropy congealing implementation - something more computationally costly than this project desires.

[32] MathWorks, “fwite - write data to binary file.” [Online]. Available: http://uk.mathworks.com/help/matlab/ref/fwrite.html

fwrite function which write binary data to file. As it writes the data column-wise, this was causing a transposition issue initially.

[33] ——, “Using matlab with other programming languages.”

[34] MATLAB, “Guide or programmatic workflow.” [Online]. Available: http://uk.mathworks.com/help/matlab/guide-or-matlab-functions.html

GUIDE was used to create the GUIs in the project

[35] ——, “imfreehand - create draggable freehand region.” [Online]. Available: http://uk.mathworks.com/help/images/ref/imfreehand.html

Tool the user utilises to draw freehand the area they wish to remove from the scan.

[36] ——, “permute - rearrange dimensions of n-d array.” [Online]. Available: http://uk.mathworks.com/help/matlab/ref/permute.html

For the transposition of ND arrays (arrays greater than 2-dimensions.)

[37] ——, “regionprops - measure properties of image regions.” [Online]. Available: http://uk.mathworks.com/help/images/ref/regionprops.html

A possible solution for removing medical markers from mammograms.

[38] ——, “Vectorization.” [Online]. Available: http://uk.mathworks.com/help/matlab/matlab_prog/vectorization.html?refresh=true

Vectorisation was extremely useful in this project for reducing the run time of the Non-probabilistic entropy and membership functions. This was a new concept to myself.

[39] A. OHagan, “Probabilistic uncertainty specification: Overview, elaboration techniques and their application to a mechanistic model of carbon flux,” 2011. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.319.9570

A paper outlining a way in which to apply probabilistic uncertainty to an engineering problem. The opening sections of the paper give a detailed explanation of probabilistic uncertainty, enough to glean enough understanding to write a detailed analysis and apply it to my own case.

[40] N. R. Pal and S. K. Pal, “Object-background segmentation using new definitions of entropy,” *IEE Proceedings E - Computers and Digital Techniques*, vol. 136, no. 4, p. 284295, Jul 1989.

Early fuzzy entropy working - similar to that of De Luca & Termini.

[41] ——, “Higher order fuzzy entropy and hybrid entropy of a set,” *Information Sciences*, vol. 61, no. 3, p. 211231, Jun 1992.

Pal and Pal’s paper outlining both Hybrid Entropy and Higher-order entropy. Hybrid entropy was implemented following their 2 fuzzy-set approach, to ensure that inversion is preserved (sets A and B: A = x and B = 1 - A).

[42] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, “Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, p. 22332246, Nov 2012.

Alternative Joint Image alignment technique.

[43] ——, “Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 11, pp. 2233–2246, 2012.

Alternative Joint Image alignment technique.

[44] D. Peterson, “What is kanban?” [Online]. Available: http://kanbanblog.com/explained/

Blog outlining Kanban - something not utilised specifically in this project.

[45] J. Poskanzer, “Pgm format specification.” [Online]. Available: http://netpbm.sourceforge.net/doc/pgm.html

Webpage outlining the header file of pgm images.

[46] Radswiki, "Mammography views — radiology reference article — radiopaedia.org." [Online]. Available: http://radiopaedia.org/articles/mammography-views

Detailed online encyclopedia outlining the different types of mammogram scan which can be performed and their benefits/weaknesses.

[47] A. Samieh, "Shape recognition - file exchange - matlab central," 2016. [Online]. Available: http://uk.mathworks.com/matlabcentral/fileexchange/15491-shape-recognition

Demonstration on shape recognition which was investigated for the removal of medical markers from the mammograms.

[48] W. Sander, "On measures of fuzziness," *Fuzzy Sets and Systems - FSS*, vol. 29, no. 1, p. 4955, 1989.

Presents a characterisation of a fuzzy entropy - decided not to be included in the project due to the lack of probabilistic modeling.

[49] L. J. Savage, *The Foundations of Statistics*, 1954. [Online]. Available: https://books.google.co.uk/books/about/The_Foundations_of_Statistics.html?id=zSv6dBWneMEC

[50] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal 27*, 1948.

Paper outlining Shannon‘s work which resulted in Shannon Entropy.

[51] A. SHIRALIGE, "Agile estimation:8 steps to successful story point estimation." [Online]. Available: http://www.agilebuddha.com/agile/agile-estimation-8-steps-to-successful-story-point-estimation/

Blog discussing the benefits of consistent estimation and how being accurate in estimation is not necessarily as important.

[52] E. Sickles, D. CJ, B. LW, *et al.*, *ACR BI-RADS® Mammography*. In: ACR BI-RADS® Atlas, Breast Imaging Reporting and Data System, 2013.

BI-RADS breast density classification commonly used today.

[53] M. . Simulink, "Testing frameworks." [Online]. Available: http://uk.mathworks.com/help/matlab/matlab-unit-test-framework.html

MATLAB's unit testing documentation.

[54] J. Suckling, "The mammographic image analysis society digital mammogram database," *Exerpta Medica International Congress Series*, vol. 1069, 1994.

The Mini-MIAS dataset which was used during this project. These images have been reduced in size to allow a more sensible computation time when compared to the orignal MIAS scans.

[55] C. R. UK, "Breast cancer tests," Apr 2015. [Online]. Available: http://www.cancerresearchuk.org/about-cancer/type/breast-cancer/diagnosis/breast-cancer-tests

Webpage outlining the different types of breast scan which are available, including Mammogram via MRI, Ultrasound and needle biopsy.

[56] E. Untiedt, *A Parametrized Model for Optimization with Mixed Fuzzy and Possibilistic Uncertainty*. Springer Berlin Heidelberg, 2010. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-13935-2_9 978-3-642-13934-5.

Good description of possibilistic uncertainty versus fuzzy uncertainty.

[57] D. Wells, “Extreme programming: A gentle introduction.” [Online]. Available: http://www.extremeprogramming.org/

Practices such as CRC cards were leveraged from XP during this projects

[58] J. N. Wolfe, “Risk for breast cancer development determined by mammographic parenchymal pattern,” *Cancer*, vol. 37, no. 5, p. 24862492, May 1976.

Paper outlining Wolfe’s classification of breast tissue density.

[59] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, p. 338353, Jun 1965.

Zadeh’s work into Fuzzy Set Theory paved the way for Fuzzy Entropy as implemented in this project.

[60] T. Zhou, Y. J. Lee, S. X. Yu, and A. A. Efros, *FlowWeb: Joint image set alignment by weaving consistent, pixel-wise correspondences*. IEEE, Jun 2015, p. 11911200. [Online]. Available: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=7298723

Alternative Joint Image alignment technique.