**Michael Painter**

# Spectral Image Analysis for Medical Imaging

Part II Computer Science Tripos

Churchill College, 2016

March 10, 2016

# Proforma

| | |
|---|---|
| Name: | **Michael Painter** |
| College: | **Churchill College** |
| Project Title: | **Spectral Image Analysis for Medical Imaging** |
| Examination: | **Computer Science Part II Project Dissertation, May 2016** |
| Word Count: | **579**[1] |
| Project Originator: | Dr Pietro Lio' |
| Supervisor: | Dr Pietro Lio' & Dr Gianluca Ascolani |

## Original aims of the project

*TODO*

## Work completed

*TODO*

## Special difficulties

None.

[1]This word count was computed by `texcount -total diss.tex`

# Declaration

I, Michael Painter of Churchill College, being a candidate for Part II of the Computer Science Tripos, hereby declare that this dissertation and the work described in it are my own work, unaided except as may be specified below, and that the dissertation does not contain material that has already been used to any substantial extent for a comparable purpose.

Signed

Date

# Contents

# List of Figures

# List of Tables

# Listings

# Acknowledgements

I thank my supervisors, director of studies and tutor for their extensive support, especially for putting up with me throughout the year. [TODO - anymore thanks]

# Chapter 1

# Introduction

In this chapter we introduce the problem: what data we are given and what would we like to do with it? We will suggest some possible methods that could be used to implement a solution. We describe how we might want to evaluate the performance of the system. Finally we will outline the overall pipeline of the system.

## 1.1 What are we trying to do?

*TODO*

We can briefly describe the system as taking an image with a spectrum of intensity values at each pixel and we wish to produce a pixel labelling. A pixel labelling is... and is often referred to as an image segmentation.

## 1.2 Possible solutions

*TODO*

## 1.3 My approach

*TODO*

## 1.4 Evaluation of system performance

*TODO*

# Chapter 2

# Preparation

In this section we investigate multiple machine learning algorithms, and decide on which may be reasonable to use in this project. We then investigate noise models for images, how they can be applied to our 'hyperspectral' case and what can be implemented to reduce or eliminate the noise.

## 2.1 A summary of machine learning algorithms

*TODO*

### 2.1.1 Method 1

*TODO*

### 2.1.2 Method 2

*TODO*

### 2.1.3 Method 3

*TODO*

## 2.2 Image noise and noise simulation

*TODO*

## 2.3 De-noising

*TODO*

# Chapter 3

# Implementation

In this chapter we elaborate on the implementations of the randomised forests and neural network solutions. Similarly we discuss the de-noising aspect of the system. We then look at how each of the random forests and neural networks were trained for the example data sets provided. We finally return to look at the overview of the system and how all of the components fit together into a pipeline.

## 3.1 Randomised Forests

*TODO*

### 3.1.1 Training

*TODO*

### 3.1.2 Classification

*TODO*

## 3.2 Neural Networks

*TODO*

### 3.2.1 Training

*TODO*

### 3.2.2 Classification

*TODO*

## 3.3 De-noising

*TODO*

# Chapter 4

# Evaluation

In this section we will evaluate the performance of the implemented machine learning solutions, comparing them based on the following metrics: accuracy of pixel labellings from a test set with known classifications prior, training time (as a function of the size of the training sequence) and classification time (as a function of the size of the image). I will also look at how each of the components of the system effects each of the metrics mentioned in isolation.

## 4.1 Accuracy

*TODO*

## 4.2 Training time

*TODO*

## 4.3 Classification time

*TODO*

## 4.4 The effect of the de-noising component

*TODO*

## 4.5 The effect of the number of trees

*TODO*

## 4.6 The effect of the depth of trees

*TODO*

## 4.7 The effect of the randomness of trees

*TODO*

# Chapter 5

# Conclusion

*TODO* - Summarise how the decision forests and neural networks compared (accuracy on the test data etc), basically discuss the results from the evaluation.

## 5.1 Section 1

*TODO*

## 5.2 Section 2

*TODO*

## 5.3 Section 3

*TODO* [1]

# Bibliography

[1] A. Siri Luthman and Sarah E. Bohndiek. Experimental evaluation of a hyperspectral imager for near-infrared fluorescent contrast agent studies. Technical report, University of Cambridge, 2015.

# Appendix A

# Benchmark descriptions

*TODO*

## A.1 Section 1

*TODO*

## A.2 Section 2

*TODO*

## A.3 Section 3

*TODO*

# Appendix B

# Evaluation results

*TODO*

## B.1 Section 1

*TODO*

## B.2 Section 2

*TODO*

## B.3 Section 3

*TODO*

# Appendix C

# Project Proposal

Computer Science Project Proposal

Spectral Image Analysis for Medical Imaging

M. Painter, Churchill College

Originator: Dr. Pietro Lio'

**Project Supervisor:** Dr Pietro Lio', Dr Gianluca Ascolani

**Director of Studies:** Dr John Fawcett

**Project Overseers:** Prof John Daugman & Dr David Greaves

## Introduction and Description of the Work

The core idea of the project is to use spectral algorithms and machine learning to analyse biomedical images. I will explore the use of multiple learning techniques (namely Random Decision Forests and Neural Networks) and compare them via a number of metrics, described later.

The aim of the project will be to build a classifier that is capable of handling a wide variety of noisy medical images. Different types of medical images will present different challenges such as contrast between tissues and amount and type of noise. It will classify images per pixel into classes (dependent on the image), using the spectral analysis.

During the implementation I will start by implementing something for a toy problem, by which I mean an artificially produced, noiseless image. From this solution I will move onto solving the same problem with the introduction of artificial noise, at which point I will explore the use of de-noising techniques to improve the accuracy of the classifier. Finally after this I will move onto an implementation for real images.

For the real images I will use MRI images from Zhongzhao Teng from the Department of Radiology, Engineering, which include fatty tissues from patients with atherosclerosis (the build up of fatty tissues in arteries). In this case I can train the classifier to recognise regions of calcium, lipids, haemorragic tissue, and mixtures of these.

To demonstrate the versatility of the tool I will also use another set of images obtained in a completely different way, so will present a different set of challenges including the nature of noise in the image. The images are from Siri Luthman of the BSS group in the Department of Physics, and are obtained using a hyper-spectral camera with 72 spectral bins. The images use contrast agents with various spectral profiles, each of which has a negative binding response or neutral binding response for cancerous cells. By 'negative binding response' I mean that the contrast agent binds to only non-cancerous cells, and 'neutral' means that it binds to both cancerous and non-cancerous cells. Here the classes will be cancer cells, non-cancerous cell, and a mixture of both.

## Starting Point

I will use Java for the implementation of my algorithms and use libraries provided for Java. I will also use OpenCV's machine learning library as a benchmark to test my implementations against.

## Resources Required

I will use my own computer to code on as I am more comfortable with it than the MCS machines.

I will also need example (spectral) image sets to be used in training and testing. I have kindly been provided with some MR images by Zhongzhao Teng of the Department of Ragiology and some hyperspectral images from Siri Luthman of the BSS group in the Department of Physics, as mentioned in the introduction.

## Backup Plan

I will be using a git repository for my project, which will be a private repository on GitHub. I will have two branches (at least, more if necessary) one 'master' for completed features and one 'in progress'. The in progress branch will be to push regularly any unfinished/in progress work onto so that my work is always backed up, and I don't loose any work in progress.

The local folder will also be in my Google Drive folder and so will be automatically synced to the Google servers, and I also have a time machine set up, which will take backups of the whole file system every hour.

## Work to be done

I can break down the project into the following stages:

1. Implement the infrastructure (data structures etc) and reading in of the raw data.
2. Create a tool used to indicate areas of interest on the teaching data to be used with the training data.

3. Implement the main machine learning algorithm(s), and implement an out the box solution, initially to solve the 'toy problem'.

4. Extend the implementation to work for noisy images and then real images.

## Success Criterion for the Main Result

I will use an 'out the box' solution (such as OpenCV's machine learning libraries) as a benchmark to compare my implementation(s) against. I will provide each machine learning algorithm with the same inputs for training and the same inputs for testing and will use the following metrics to compare the performance of the systems and which has 'learnt' better:

- Overall run time (of classifying a single image);

- Accuracy of classifier (percentage images correctly identified).

## Possible Extensions

- *Image Segmentation.*

  It would be useful to be able to segment the images given into separate regions of interest. (For example if we want to classify an image with cancer cells present, we indicate the regions containing cancerous cells). This is opposed to just returning the output per pixel, and would provide a more useful output for users.

- *Dimensionality Reduction.*

  If I finish the core of my project then one of the additional tasks that I could look at would be to implement a data reduction (learning) algorithm. In a spectral image with a high number of spectral bands it would be useful to identify which spectral bands are important to the classification and which are not.

  This problem more generally is called "dimensionality reduction" as we are looking to reduce the dimension of data input to the algorithm.

# Timetable: Workplan and Milestones to be achieved.

**Michaelmas weeks 3–4 (26th Oct to 4th Nov)**

In preparation for the main implementation I will read about many machine learning algorithms. For example I will use the book "Decision Forests for Computer Vision and Medical Image Analysis" to learn about Random Forests. I will similarly research about Neural networks, and I will also research some other forms of learning algorithm.

*Deliverables:*

- A small description/overview of random forests and how they work, and a similar description for any other learning algorithms researched. This should be written up in LaTeX for easy embedding into the preparation chapter of the dissertation.

*Milestones:*

- Preparation reading completed, so that I am sufficiently able to implement the learning algorithm(s) and handed in description of reading to project supervisor for review - 4th Nov.

**Michaelmas weeks 5–6 (5th Nov to 18th Nov)**

I will spend this block familiarising myself with any technologies that I will possibly use. This will include the OpenCV library. I will also design a simple UI that will be used for the supervised learning. I will also familiarise myself with OpenCL and GPU programming.

*Deliverables:*

- A sketch of the UI for the tool which will be used for the supervised learning.
- A small overview of what OpenCV, GPU programming (or other technology(s) I have looked at), and a description of why/where they will be useful. This should again be written up in LaTeX for easy embedding into the preparation chapter of the dissertation.

*Milestones:*

- Handed in the UI design to supervisor for review - 9th Nov

- Handed in the description of familiarisation of technologies to supervisor for review - 18th Nov.
- Demonstrated any small programs written for familiarisation to supervisor - 18th Nov.

**Michaelmas weeks 7–8 (19th Nov to 2nd Dec)**

*Implementation block 1.* Implement the infrastructure that will be used for the project. This will include loading the raw data into the appropriate data structures and do so efficiently as possible.

*Deliverables:*

- A bullet point list indicating what has been implemented during this block. Written up in LaTeX and to be used as a basis for the implementation portion of the dissertation. To be handed in for review.

*Milestones:*

- Have the framework that I will use completed, including unit tests, and demonstrate the tests to supervisor to check that this has been done - 2nd Dec.

**Christmas vacation weeks 1–2 (3rd Dec - 16th Dec)**

I will be on holiday during this period and so I will work on little bits when I can, this may be catching up on any work that I got behind on in Michaelmas term (effectively making this a 'slack' block), and I will begin some work on the next block if up to date.

**Christmas vacation weeks 3–4 (17th Dec - 30th Dec)**

*Implementation block 2.* Write the tool that will be used to indicate which images (or image regions) belong to a given class to be used for the supervised learning.

*Deliverables:*

- A bullet point list indicating what has been implemented during this block. Written up in LaTeX and to be used as a basis for the implementation portion of the dissertation.

*Milestones:*

- Sent the completed tool to my supervisor as proof of completion. (As it is the vacation I will not be in Cambridge and so will not be able to demonstrate in person). - 30th Dec

**Christmas vacation weeks 5–6 (31st Dec - 13th Jan)**

*Implementation block 3.* Implement the machine learning algorithm(s) chosen, and also write the 'out of the box' solution using the OpenCV library. At this stage I will only aim to have

If this block is delivered on time then I am close to having a completed project. I have purposely stacked more of the work in the holiday compared to term time, as I have other commitments in term than out of term, such as lectures and supervisions.

*Deliverables:*

- A bullet point list indicating what has been implemented during this block. Written up in LaTeX and to be used as a basis for the implementation portion of the dissertation.

*Milestones:*

- Finished writing the machine learning algorithm, and demonstrated it classifying some 'toy images' to supervisor - 13th Jan. (I will be back in Cambridge by this time).

**Lent weeks 1–2 (14th Jan - 27th Jan)**

The progress report needs to be given up by noon on the 29th Jan, and so should be written in this block.

This rest of this block will be kept free for 'slack'. This slack should include incorporating any feedback from my supervisor.

If I am up to date and there is no additional feedback that needs to be resolved, then I will proceed to work a block ahead. This will give me an additional block at the end of lent to implement any extensions.

*Deliverables:*

- Progress report.

- Item written in LaTeX which begins with the supervisors comments/feedback and ends with how that was incorporated into the system.

*Milestones:*

- Demonstration of any features added made to supervisor - 27th Jan

**Lent weeks 3–4 (28th Jan - 10th Feb)**

*Implementation block 4.* In this block I will finish implementing my solution, and extend my simple solution to a more complex one capable of handling 'toy images' with artificial noise, and then real images. This will require implementing de-noising of the images.

Also after the progress report a small presentation needs to be prepared. I will use the beginning of this time block to do this.

*Deliverables:*

- A bullet point list indicating what has been implemented during this block. Written up in LaTeX and to be used as a basis for the implementation portion of the dissertation.

*Milestones:*

- Hand in progress report (completed in the previous block) - 29th Jan.
- Perform a small mock presentation/interview with supervisor - 3rd Feb.
- Made progress presentation to overseer group - 5th Feb.
- Finished implementing any de-noising algorithms, and have a working solution for real images, including a demonstration to supervisor - 10th Feb.

**Lent weeks 5–6 (11th Feb - 24th Feb)**

Evaluation of the system. This will require writing tests that will gather the quantitative data (run time and accuracy of classification) to be used in the comparison required to satisfy the success criteria.

I will generate graphs from data in this phase using MATLAB, so that I can decide if the data is useful or not. This will prevent me from realising that the data isn't good at too late a stage in the project and when there is not enough time to re-evaluate the system. This has the advantage that the graphs will be ready to be exported directly into the dissertation.

*Deliverables:*

- Table for any qualitative data evaluated and spreadsheet of quantitative data, which must include values for accuracy of the system (in a specific test case) and timings for how long classification takes (on my home machine).
- Generated meaningful graphs using MATLAB with data gathered from evaluation.

*Milestones:*

- Table/spreadsheet completed and filled out with all data required for a good write up. Spreadsheet and graphs sent to supervisor for reviewing - 24th Feb.

**Lent weeks 7–8 (25th Feb - 9th Mar)**

I will use my last block of work in lent as another 'slack' block for if I get behind on any work for any reason such as a high load of supervision work through the term. This should also be used to incorporate any final feedback from my supervisor.

If am up to date at this point I will work on implementing some of the extension tasks outlined in this document.

*Milestones:*

- System complete, success criterion met and demonstrated these to supervisor - 9th Mar.

**Easter vacation weeks 1–2 (10th Mar - 23rd Mar)**

Set up the dissertation document and write introduction and preparation chapters of the dissertation.

*Deliverables:*

- Introduction and preparation chapters of dissertation.

*Milestones*:

- Completed first draft of introduction chapter and given to supervisor - 17th Mar.
- Completed first draft of preparation chapter and given to supervisor - 23rd Mar.

**Easter vacation weeks 3–4 (24th Mar - 6th Apr)**
Write up the implementation and evaluation chapters of the dissertation.

*Deliverables:*

- Implementation and evaluation chapters of dissertation.

*Milestones:*

- Completed first draft of implementation chapter and given to supervisor - 31st Mar.
- Completed first draft of the evaluation chapter and given to supervisor - 6th Apr.

**Easter vacation weeks 5–6 (7th Apr - 20th Apr)**
Complete dissertation by completing the conclusion section and any appendices. I will also re-iterate through dissertation see if anything can be improved.

*Deliverables:*

- Draft of complete dissertation.
- 2nd draft of complete dissertation.

*Milestones:*

- Full first draft of dissertation handed into supervisor for review - 11th Apr.
- Hand in second draft of completed dissertation to supervisor - 18th Apr.

**Easter term weeks 1–2 (21st Apr - 5th May)**
Final 'slack' block for the dissertation. If the dissertation is completed then I will proof read multiple times. I would like to have the dissertation 'finished' by now to focus on revision for the exam and on learning any of the Easter term courses.

*Milestones:*

- System complete and success criterion met - 5th May.

**Easter term week 3 (6th May - 13th May)**
Final proof read and then an early submission so that I can focus on exam revision for the remainder of the term.

*Deliverables:*

- Dissertation.

*Milestones:*

- Hand in dissertation on time. (Project finished). - 13th May.

# Appendix D

# Glossary