Image Processing in the National Plant Phenomics Centre

Report Name	Progress Report
Author (User Id)	Andrew Tindall (ajt7)
Supervisor (User Id)	Hannah Dee (hmd1)
Course	GG4R Computer Science & Artificial Intelligence
Module	CS39440
Date	November 19, 2012
Revision	1.0
Status	Release Candidate 2
Word Count	3123

Contents

1 Project Summary

1.1 Phenomics

Before explaining the project, it is first important to understand the field in which it operates - namely that of plant phenomics.

Phenomics is a field of research in biology related to the systematic observation and analysis of "phenomes", or biochemical and physical trait expression; and how their expression changes in result to environmental and genetic factors.

Phenomics is considered a rapidly emerging transdiscipline, requiring expertise in fields including "genetics, molecular biology, cell biology, systems biology, and higher levels of phenotypic expression" alongside wider understanding of mathematical modelling and information sciences. [1]

The discipline has applications across many fields including public health [1] ; human genetics; biofuels; global food security [4] ; and others due to its ability to allow understanding of how factors can affect traits, allowing for more development of more resilient crops, and predispositions to disease.

Plant Phenomics is a specific subset of phenomics, and concerns itself with "the study of plant growth, performance and composition" [4], and is the subset observed in this project.

1.2 Plant Phenomics and Food Security

Food security is defined by the World Food Summit as a circumstance by which "all people, at all times, have physical and economic access to sufficient, safe and nutritious food to meet their dietary needs and food preferences for an active and healthy life" [3]. This is increasing taken within the context of climate change, globalisation and corporatism, international relations, and the global economy, all of which impact on the ability to provide food through links at each stage of the supply change. With environments changing and often becoming less hospitable to life, there is a turn towards breeding high-yield crops which are resilient and adapted to future climates [4].

With traditional breeding no longer resulting in yield increases that can meet projected demand of staple crops [4], there is a need to turn to phenomics, which offers the ability to rapidly analyse large plant populations and thereby develop germplasm stocks that can meet current and future needs.

1.3 Use of Image Processing in Phenomics

Traditionally, methods for phenome observation would involve destructive techniques that remove entire plants or parts, and this results in the need for larger physical space and longer time periods for research. [2]

In recent years there has been a large focus on the development of high-throughput and non-destructive techniques, particularly in the analysis of Arabidopsis - a small flowering plant including Thale Cress which is used as a model organism due to being the first plant to have its genome sequenced in its entirety [7]. Principle among this area is the use of image processing and analysis, which allows for remote screening of multiple traits with minimum disruption to specimens. [2]

LemnaTec are one of the eminent organisations involved in the field of plant phenomics, supporting research institutes around the world through provision of hardware and algorithms. This includes infrastructure at the recently developed National Plant Phenomics Centre (NPPC) at Aberystwyth University which utilises robotic plant handling and automated image capture and analysis to conduct phenomics on entire plant populations in the hopes of identifying plants with increased tolerance to adverse conditions [5]; the work of which provides the

basis for this project.

1.4 Project Details

This project seeks to develop a system capable of using multiple algorithms for the processing and analysis of automated image data sets of arabidopsis populations grown in a controlled environment in order to provide phenotype information and analysis about the plants. In doing so, it shall be comparable to, and build upon the the work of the NPPC, and Rosette Tracker [2].

The work has several limitations arising from multiple factors. Foremost is that of data collection. That the project requires the growth of populations of plants introduces inherent time factors into the project, which means datasets are time-limited, and this can hinder early analysis and testing of software. Additionally, variations in the environment for growth, and in terms of imaging, can present discrepancies in the data set, or even render parts of the set unviable for inclusion, such as due to an image being overexposed, or a camera being out of position for several frames of the set. Limitations also exist in terms of what can be achieved through visible-spectrum imaging, which is the broad focus of the project, as not everything can be observed at the scales and spectrum being used; however it may be possible for the project to utilise further imaging techniques such as IR imaging later into the project subject to the provision of data from the NPPC.

Ultimately, the finished work should output data that provides answers to important questions such as "to what extent does the ede1 mutation affect growth rates and patterns?" and flowering times between different populations. The project may be judged a success should be it capable of providing the prerequisite steps and analysis to reach this stage, as well as data that allows conclusions about phenotypes to be made on these questions and others.

2 Current Progress

2.1 Technologies and libraries

There exists numerous libraries for image processing and analysis, including ImageJ and OpenCV. Each of these tends to implement common processing techniques such as Canny and Sobel Edge Detection, image segmentation, etc.

ImageJ is a public domain image processing and analysis tool written for the java programming language. OpenCV is a similar, open source implementation for C++, C and Python and has over 2500 optimised algorithms [6]. Wrappers exists for OpenCV, including JavaCV, which allows for its use in java.

Both are common in academic environments, although OpenCV appears to enjoy wider uptake across sectors, perhaps due to being available across multiple programming languages, and comprehensive documentation for the C++ implementation.

After preliminary reading and research into the kind of techniques required to undertake this project, and after consideration of familiarity with each library, and the required programming languages for such; it was decided that initial prototyping would make use of JavaCV to provide the underlying functionality of the project. Whilst additional methods and algorithms will likely be required to written regardless of library, it is possible that should issues with JavaCV arise in any systematic or seriously hindering way, use of switching to other libraries shall be considered prior to formalisation of a stable code branch.

2.2 Language selection

Java was selected as the programming language for the implementation of this project due to two primary reasons. Firstly, it is the language in which the most experience is currently

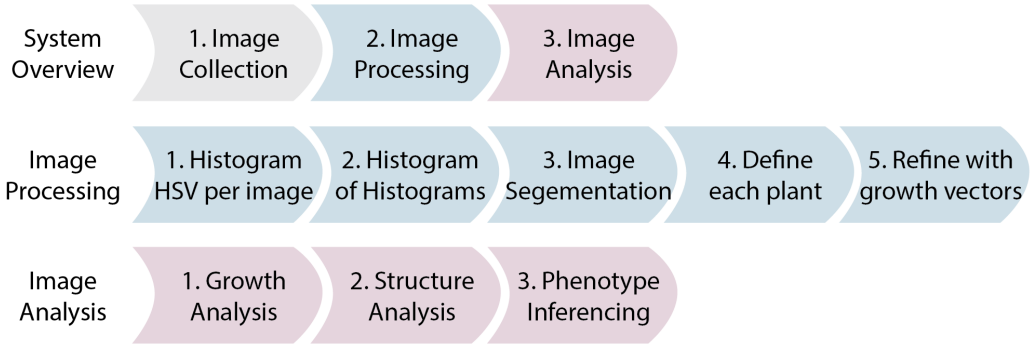

Figure 1: Breakdown of system processes including image processing and analysis.

held, and so work can focus on actual implementation rather than learning the language itself. Secondly, Java is capable of running on the majority of platforms and hardware configurations without any notable or significant differences in each instance.

Subjection to the completion of all planned features, and excluding any earlier switches due to any arising issues that render the current language and libraries unusable for the project, it shall be explored the possibility of porting the project to other languages as part of the refactoring process - for example Ruby, with which some familiarity is held, or Python or C++ which are commonly used for the development of image processing and analysis systems.

In doing this, it should be achievable to decrease computation times that result from the java runtime environments resource consumption, and allows for greater understanding of core concepts and furtherment of knowledge of programming languages with which current knowledge is lesser compared to java.

Any such decisions on language or library porting shall be done on the basis of maintaining a stable branch in its then-current state so that a functioning system may be delivered regardless of any issues occurring in these forks.

Should any forks be completed, they shall be evaluated and possibly replace the stable branch for submission, or if entirely equivalent, both be submitted as viable code bases.

2.3 System Overview

Before any prototyping was begun, it was reviewed at a top-down level what features the project would require. Broadly, the system can be broken down into three categories: Image Collection, Image Processing, and Image Analysis.

Each of these stages were further broken down until individual processes were defined, and this was used as the basis for prototyping and development, by working chronologically through the processes, amending and introducing new processes where needed or beneficial.

As a result of this methodology, the current system overview is as displayed in figure 1, and although likely to remain broadly the same, sections are likely to be added, removed, or otherwise adjusted as the project progresses.

2.4 Prototyping and Experimentation

The project is being developed incrementally, with prototypes of each feature being developed and refined before moving onto the next feature. Eventually these refined prototypes will be revised and merged into a stable code branch, with additional prototyping forks being merged in at later dates.

The first code developed was for handling image input and output, and under OpenCV/JavaCV, is mere lines long. This code forms the basis of all following features, as the vast majority of work requires access to the raw data set or processed images.

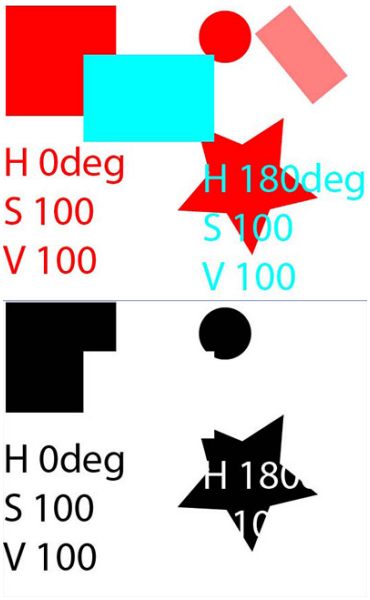

Figure 2: Comparison of test image input with results of preliminary image segmentation

The next feature to be developed was a rudimentary image segmentation method, which initially took a pre-defined hue, and matched this against the HSV colour space for a given image, returning a segmented image showing just pixels of that hue. Figure 2 shows the result of this code run against a test image.

Following the successful test, the segmentation method was adapted to make use of histogram information, as real life objects are not just a single hue, and so to adequately detect them, we must look at multiple values across specific ranges. Segmentation at this stage would only highlight the most common hue value bucket, which in images from earlier in the dataset, would not correspond to the plants; and depending on the fuzziness of each bucket, would effectively just create a black and white version of the original image.

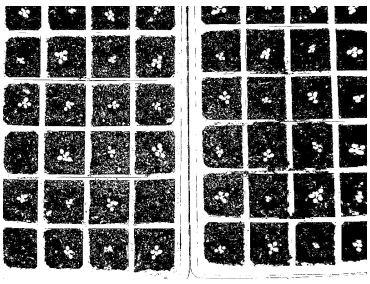

Figure 3: Naive histogram-derived image segmentation

It was decided that a solution to the issues that presented when using histograms to define segmentation, was to use histograms across the entire data set as a form of voting for the hue ranges to segmentate on the current image in the set. This would thereby allow images later in the set, which would in most cases contain an increasing amount of plant-specific colours inherent due to the growth of plants meaning they cover more of an image, have an influence over the dominant colours that define a plant. This solution is not yet fully implemented however is projected for completion by the end of week 47, 2012.

As per the system overview, the next feature to be prototyped shall be that of plant detection, including defining boundaries for each plant, potentially through use of environmental features, as well as plant diameter and compactness as used in Rosette Tracker [2].

2.5 Technical Issues and Risks

In developing this system, there are several potential issues and risks to overcome, all of which may have a differing level of impact upon the overall work of the project.

A consideration for any project is the risk of data loss, be it through hardware or component failure, file and data corruption, or saving over functional code with non-functional work and not being able to restore to a previous state. To resolve this, the project is sorted into a clear directory tree, with stable and forks being kept separate. On regular intervals this work is backed up to multiple local storage devices, both internal to the originating workstation, as well as portable media. At the same time, changes are committed to a git repository, providing a remote backup and change history of all versions in a remote setting, adding an additional level of redundancy.

Some aspects of data loss prevention are beyond the scope of control, for example, dataset collection is not handled personally, and external factors may influence the ability to collect and store this information; with one dataset being lost due to power failure since the start of the project, albeit with no real-term impact as previous data sets were available.

Another issue to consider is that of computation time when running the software. Although image processing libraries promise computation rates allowing for high-throughput, this will be affected by numerous factors such as number of images in a dataset, resolution of images, image quality if dataset quality assurance is automated, hardware specifications, complexity of analysis, and whether any other intensive software is running at the same time. To ensure adequate timeframes for computation, these factors should be as controllable as possible, such as through ensuring data sets only contain relevant information before processing - discarding images where colour balance is off due to environmental factors such as lighting discrepancies, or where positioning of plants is significantly divergent from other images in the set which would present issues in identification and tracking of individual plants.

3 Planning

3.1 Methodology

In undertaking this project, it was explored what development methodologies could be applied, such as the waterfall model, rapid application development, or iterative and incremental design.

Under the waterfall model, the project is completed in seven phases which are followed in order: requirements specification; design; implementation; integration; testing; installation; and maintenance. In this way, all requirements and concepts are laid out in detail before any implementation begins. This can create issues where it is not clear the exact route to be taken to complete the tasks at hand, or serious difficulties arise preventing implementation of a section; and if requirements change, potentially large sums of work can be invalidated.

With rapid application development (rad), a project consists of just four phases: Requirements planning, where the requirements, scope, and constraints are defined; user design, where prototyping is used to understand and modify the specification to fit the needs of the task; construction, where the application is developed; and cutover, where testing and delivery take place. This allows for a far more flexible approach than under waterfall, with input and changes welcomed at each stage, whilst still maintaining a design-first approach.

In Iterative and Incremental Design, there are four phases, which may iterate numerous times. Inception is used to outline requirements and scope at a high level; elaboration then allows for the drawing up of a more defined working structure; this is then incrementally produced in the construction phase, after which the system is transitioned into the operating environment. Through the use of iteration, the project can evolve over time beginning with

simple implementation of parts of the structure, making use of previous iterations to expand and improve existing and future features.

It was decided that as the project was wide enough to provide uncertainty in low-level requirements, an iterative and incremental model would be most suitable, so as to provide for development of features in a modular form that could expand in scope as and when needed.

In doing this, a working product can be derived from a broad system overview, or system control list, and each feature can be implemented easily and in a manageable and scheduled manner. This means a clear road-plan exists whilst also providing fluidity and adaptability in the project.

3.2 Project Schedule

Figure 4 depicts the current project road-plan through to the final deliverable of the final presentation, taking into account likely risks or delays caused by external factors such as exams starting in week 3, 2013; conference attendances throughout the same timeframe as the project, and temporary incapacity due to illness.

Each task prior to Week 1, 2013, corresponds to development of key features through an iterative process, and this deadline is set to ensure a core set of features are delivered. Following weeks account for the formalisation and refactoring of this code base, whilst exploring other opportunities to build upon this core. The plan provides a full six weeks where the code should be locked and work is focused entirely on delivery of the final report. This is reflective of the weighting of the report, as well as the need to ensure functional, documented code prior to the final delivery of the project.

Given the nature of iterative and incremental development, several weeks have been set aside for the exploration of techniques not yet specified in the control list / system overview.

By week 50, the code base should contain functionality for the core features required for basic analysis, namely colour frequency analysis, image segmentation, and plant classification. In the following weeks, additional algorithms are to be added to further refine these features, as well as to provide analysis in other areas, for example, leaf density.

In addition to the tasks listed, it is hoped that discussions shall be held with staff and researchers involved in the National Plant Phenomics Centre at IBERS; and it is likely that such talks will help shape the direction of the project, such as by providing insight into potential problems for the system to attempt to solve, or access to new datasets potentially using different imaging techniques. Changes due to this are already partially factored into the assigned time frames for the completion of each task.

3.3 Demonstrations

Completion of the project includes two demonstrations of the system. Once during week 5, 2013, and again during week 18, 2013. In both instances, it is hoped that results of the analysis from implemented features can be presented, where possible demonstrating the processing and analysis live.

In the mid-project demonstration, it is to be expected that there would be less data to show due to not all analysis features being implemented at that stage, as well as the possibility of additional data sets being made available throughout the duration of the project, including after this demonstration. However, it should be considered sufficient to demonstrate, more than merely proof of concept, that the system is functional and capable of producing real phenomic information based on the provided input data, even if the output data is limited in its scope at this stage.

For the final demonstration, it is expected that more findings of project can be reported, namely at least some of the questions laid out in prior sections of this document. In addition,

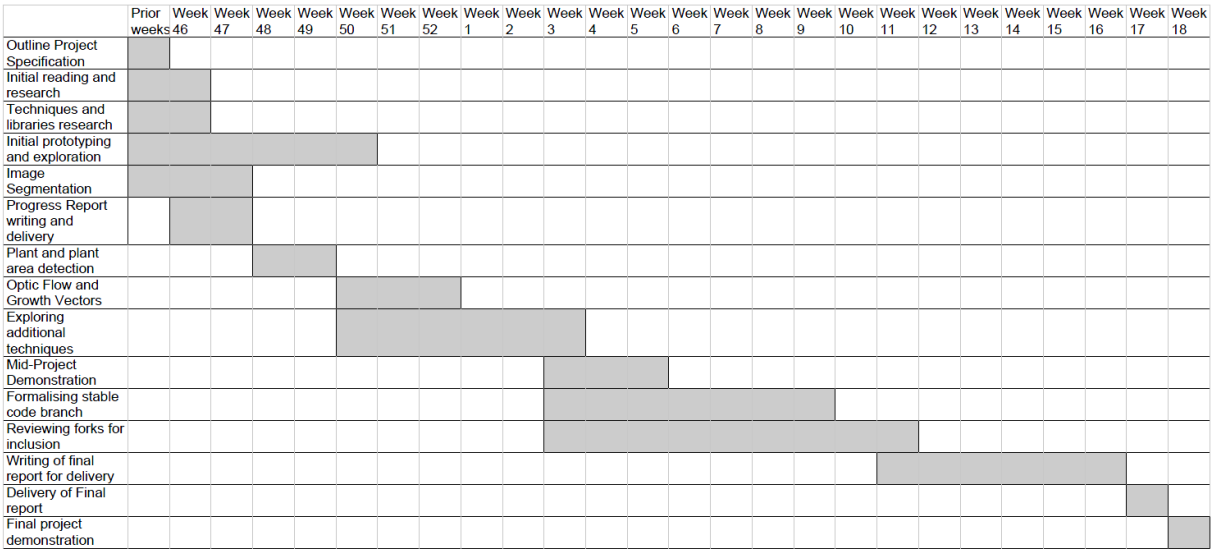

Figure 4: Gantt diagram showing project road-plan through to week 18, 2013

a live demonstration of the software, should also be conducted, as well as potential discussion of future scope for research and development from this project post-delivery.

With both demonstrations, access to a projector, and a device capable of running presentation software is a clear requirement. Ensuring the device is also capable of processing data in real-time and within the timeframe of the demonstration is also a major consideration, which may involve the selection of a subset of a data set or other such preparation to ensure execution times are permissible.

Annotated Bibliography

[1] R. Bilder, F. Sabb, T. Cannon, E. London, J. Jentsch, D. S. Parker, R. Poldrack, C. Evans, and N. Freimer, "Phenomics: the systematic study of phenotypes on a genome-wide scale," *Neuroscience*, vol. 164, no. 1, pp. 30 – 42, 2009, linking Genes to Brain Function in Health and Disease. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306452209000487

This paper explores and defines the discpline of phenomics from the perspective of neuroscience, medicine, and in relation to the human genonome. It provides a useful overview of the subject, whilst going into detail of its applications within areas not touched upon in this project.

[2] J. De Vylder, F. Vandenbussche, Y. Hu, W. Philips, and D. Van Der Straeten, "Rosette tracker: An open source image analysis tool for automatic quantification of genotype effects," *Plant Physiology*, vol. 160, no. 3, pp. 1149–1159, 2012. [Online]. Available: http://www.plantphysiol.org/content/160/3/1149.abstract

This paper explores image analysis as a nondestructive method for studying plant growth in Arabidopsis, and presents "Rosette Tracker", which is an open source tool designed to work on both high-throughout and small-scale and low-tech phenomic projects. It looks at reasons for such methods, and outlines clear procedures of a method for plant detection through the use of hue modelling, segmentation, and connected component detection.

[3] *Rome Declaration on World Food Security*. FAO, 1996, world Food Summit 13-17 November 1996, Rome. [Online]. Available: http://www.fao.org/docrep/003/w3613e/w3613e00.htm

This Declaration outlines one of the principle definitions of Food Security, primarily in the context of availability to individuals.

[4] R. T. Furbank and M. Tester, "Phenomics - technologies to relieve the phenotyping bottleneck," *Trends in Plant Science*, vol. 16, no. 12, pp. 635 – 644, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1360138511002093

This paper explores phenomics as applied to plant phenomics, and its role in tackling food security concerns through including crop yield.

[5] Institute of Biological, Environmental, and Rural Sciences; Aberystwyth University, "National plant phenomics centre - accelerating plant improvement." [Online]. Available: http://www.aber.ac.uk/en/media/Example-of-Research---National-Plant-Phenomics-Centre.pdf

An article published by IBERS which outlines the NPPC project, its uses, and the wider context surrounding its work.

[6] OpenCV. Opencv wiki. [Online]. Available: http://opencv.willowgarage.com/wiki/

OpenCV homepage, which outlines details of what the library does, and statistics on usage and contents.

[7] The Arabidopsis Genome Iniative, "Analysis of the genome sequence of the flowering plant arabidopsis thaliana," *Nature*, vol. 408, pp. 796–815, 2000. [Online]. Available: http://www.nature.com/nature/journal/v408/n6814/full/408796a0.html

This paper reports the work of the Arabidopsis Genome Initiative in analysis the genome sequence of Arabidopsis, indicating it to be the first time a complete genome sequence of a plant has been presented, and its applications in crop improvement.