**Artificial Intelligence and Augmented Intelligence for Automated Investigations for Scientific Discovery**

Computer Vision in High Throughput Chemistry
Project Report
Project Dates: 21/06/2021 – 27/08/2021
University of Strathclyde

Project Student: Jamie Longino, University of Strathclyde
Supervised by: Dr Marc Reid, Reid Group, University of Strathclyde

Report Date: 28/09/2021

AI3SD-Intern-Series:Report-12_Longino

Computer Vision in High Throughput Chemistry
AI3SD- Intern-Series:Report-12_Longino
Report Date: 28/09/2021
DOI: 10.5258/SOTON/AI3SD0151
Published by University of Southampton

**Network: Artificial Intelligence and Augmented Intelligence for Automated Investigations for Scientific Discovery**
This Network+ is EPSRC Funded under Grant No: EP/S000356/1
Principal Investigator: *Professor Jeremy Frey*
Co-Investigator: *Professor Mahesan Niranjan*
Network+ Coordinator: *Dr Samantha Kanza*

## Table of Contents

# 1. Project Details

| | |
|---|---|
| Title | Computer Vision in High Throughput Chemistry |
| Project Reference | AI3SD-FundingCall3_014 |
| Supervisor Institution | University of Strathclyde |
| Project Dates | 21/06/2021 – 27/08/2021 |
| Website | https://www.dr-marc-reid.com/reid-group-research/overview |
| Keywords | Computer Vision, High Throughput Experimentation, HTE, Analytical Chemistry, Edge Detection |

# 2. Project Team

## 2.1 Project Student


| | |
|---|---|
| **Jamie Longino** | |
| **Department of Computer and Information Science, University of Strathclyde** | |
| **jamie.longino.2019@uni.strath.ac.uk** | **www.linkedin.com/in/jamie-longino** |


## 2.2 Project Supervisor


| | |
|---|---|
| **Dr Marc Reid** | |
| **Department of Pure & Applied Chemistry, University of Strathclyde** | |
| **marc.reid.100@strath.ac.uk** | **https://www.dr-marc-reid.com/** |


## 2.3 Researchers & Collaborators

Dr Marc Reid – UKRI Future Leaders Fellow and head of Reid Group Research.
Dr Jake McGuire – Postdoctoral Research Associate.
Chunhui Yan –PhD researcher the group.
Nathalie Bugeja – MSc Forensic Science student in the group.
Dr Martin Goodfellow – Teaching Fellow within the Department of Computer and Information Science.

# 3. Lay Summary

**Intro/ Lay Summary**

High-Throughput experimentation (HTE) is a field of growing importance in modern chemistry. (Luigi da Vià, 2020) It allows huge numbers of experiments to be ran simultaneously at a smaller scale and cost per experiment than traditional, more labour-intensive synthetic methods. From its origin in the 1950s, it has become common place in biology laboratories worldwide. However, with the comparatively more challenging range of reaction conditions in chemistry, HTE remains confined primarily, but not exclusively (Shevlin, 2017), to industrial laboratories. Continued developments in chemistry-focussed HTE has the potential to greatly multiply productivity, lowering costs, and total research-to-commercialisation times.

Among the various challenges affecting HTE chemistry is the HPLC analysis bottleneck, HPLC, or High-Performance Liquid Chromatography, is a technique used to identify components in a mixture. This is a sequential analytic technique where analysis can take from a few minutes to as much as 15 minutes per sample, dependant on the component you are testing for. In HTE chemistry, HPLC analysis is applied using offline samples from a well plate. The number of samples, one per well, can range from 96 to as high as 1536. For a 96 well plate, sequential analysis of 5 minutes per sample would take a full working day (8 hours) to analyse.

Finding methods to reduce the HPLC analysis bottleneck could result in greatly increased productivity, enabling chemists to make informed, data-driven research decisions in currently unprecedented timescales.

Our approach towards reducing the HTE analysis bottleneck is through implementing computer vision (CV), through which the potential analysis time savings are estimate (through conversation with our industrial partners) to be as great as a 90% reduction from traditional methods.

Computer vision falls within the broader fields of artificial intelligence (AI) and machine learning (ML). In short, it is the combination of hardware and software techniques, primarily used to derive meaningful analysis and information from data contained within digital images/ videos recorded on cameras and camcorders (Bernd Jähne, 2000). It's application in chemistry, though growing, (Luigi da Vià, 2020)is still in its fledgling state. Towards application in HTE, this internship has focused on strategic developments in the field of computer vision for analytical chemistry.

## 4. Aims and Objectives

My placement has been with Reid Group Research, a physical organic and analytical chemistry research group at the University of Strathclyde. For a number of years and, since this year, funded by a UKRI Future Leaders Fellowship, they have used CV tools to assist chemical research. Existing projects include the development of tools for kinetic analysis of single reactions, enabling ex situ real time determination of the success or efficiency of a reaction. Within the group there has been a push to develop these techniques further for use in HTE. My role was to take developmental video analysis software in the group and create a more advanced version of the software to enable the analysis.

The first aim of the internship was to review the specifics of the project and develop an understanding of the project aims and deliverables, catching me up to speed with how I'd fit into the group.

Next, I aimed to acquaint myself with the class and package structure of the existing software, with the goal of becoming familiar with it and creating a Package Relation Diagram (PRD). After this, the aim was to get all the experimental software already belonging to the group to work on my own machine, so that I could run the software and fully understand the inner workings of the program. The first step of this was to run the current version of a program called *Kineticolor* and successfully generate outputs using sample videos. To achieve this, I needed to collect sample videos of chemical reactions to work with.

After reviewing the software engineering project as it stood for the group, the next aim of my internship was to begin familiarising myself with CV concepts, and through this explore different image conversion methods of relevance to this HTE-focused project. From this, the objective was to develop code for manipulating the colour channels and colour spaces in use, to allow for full colour space analysis of the image to determine how to separate components from the background in future developments. To facilitate effective development and feedback, it was also necessary to consider how to gather videos of well-plate reactions for use in developing HTE edge detection methods.

Thereafter, the core aim of the project was to assess more sophisticated edge detection methods as candidates for use, and then to develop software as a minimum viable product (MVP) implementation of the HTE code. I aimed to study and write implementations of various techniques through the use of morphology algorithms to develop image masks, investigate edge detection algorithms, edge hierarchies, and find suitable candidates for combined use in the project.

The overall aim for my tenure as an intern was to take the code from where it was at the start of my time, which was a single instance kinematic analysis software, and propel development into achieving a HTE implementation. With the aim of facilitating further development into this implementation after my time was up. The solution had to be applicable within hardware requirements, meaning that the software should be able to run on a setup comprising of a camera and laptop, thus making it more widely accessible.

Other aims for the project included taking the opportunity to speak with chemists in industry to better understand their needs, in turn allowing for the software to be a better fit for end use. Additionally, I was made responsible for creating update presentations, and to regularly log my code developments for the purposes of back-up and version control.

## 5. Methodology

All code for the project was written in Python, using the Spyder development environment through Anaconda. All code was written in accordance with the PEP8 style guidelines. Most of the work done was through developing a single script, which sought to interface with the existing body of code developed in the group. Diagrams such as the PRD were created using DrawIO and were designed according to standard Project Relation formatting.

High quality videos and still images of a series of reactions taking place on a 3x4 (12) well plates were needed, and were to be provided by team members working in the

lab. The development of the software was iterative, using the collected test footage through the program and analyse the results. After analysing the results, I would use my findings to adjust variables within the program during the next development phase, with the aim of over time narrowing the margins of error and getting more precise results in the next iteration.
This follows what is known as the Agile development approach, also referred to as sprint developing. This is a commonly, and widely used development cycle worldwide. The basic form of it is a series of 'sprints' where, at the beginning of each cycle, you decide the requirements of the stage you're working on. This is followed by designing a solution to those requirements, developing said solution, after developing you test the solution before deploying it, and then finally reviewing the work before moving on to the next cycle.

An example of Agile in practice is planning the goal for the week with Marc, my project supervisor, and gathering the requirements from him. I would then go and work on designing how I thought I could meet those requirements, before working on developing them in Python. Using the test footage, I would run the new addition to the code. Once it was working as intended with the core of the existing software, I would deploy that as the current version of the software I was working on. Towards the end of the week or start of the next week before planning the next stage. I would meet with Marc to review the progress I had made, before planning the next stage of development.

## 6. Results

The culmination of the first objectives for the internship was creating a PRD (Fig. 1) to show understanding of the existing software and to provide reference when doing further work. The diagram – simplified for the purposes of this report to avoid compromising proposed sister projects with commercial applications – was made to show what third party modules were needed for each class and how the classes interact and inherit data from one another.

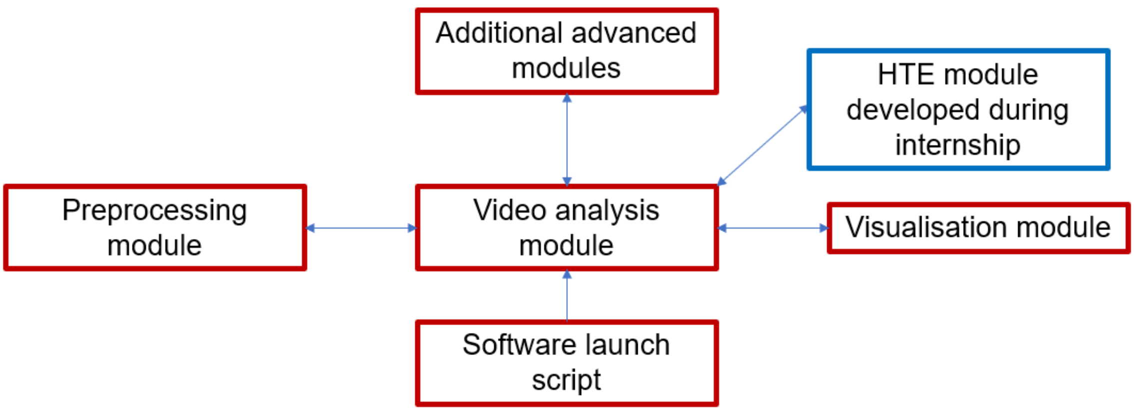


*Figure 1 – Simplified Package Relation Diagram, highlighting where internship developments connect to existing research.*

The next stage was to run the existing code, ensuring that understanding could be reduced to practice. In so doing, previously unidentified challenges with module dependencies and library versions were highlighted and patched. Thereafter, everything worked as intended and test footage output expected analysis (Fig. 2).

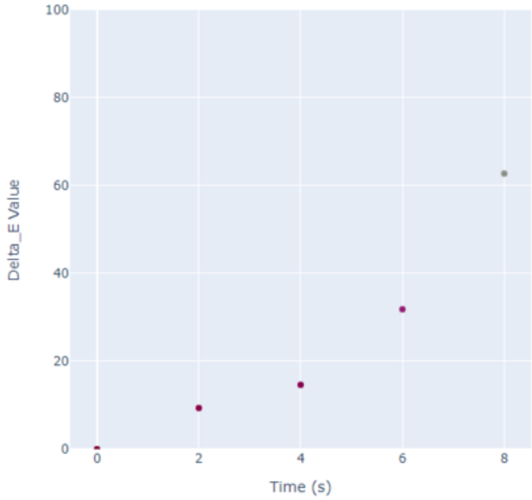


*Figure 2 – Sample output from existing computer vision software under development in Reid Group Research.*

Moving on to writing new HTE-focused code, initial efforts focused on exploring raw image manipulations in order to appreciate possible opportunities enabling detection of multiple reactions (Fig. 3).

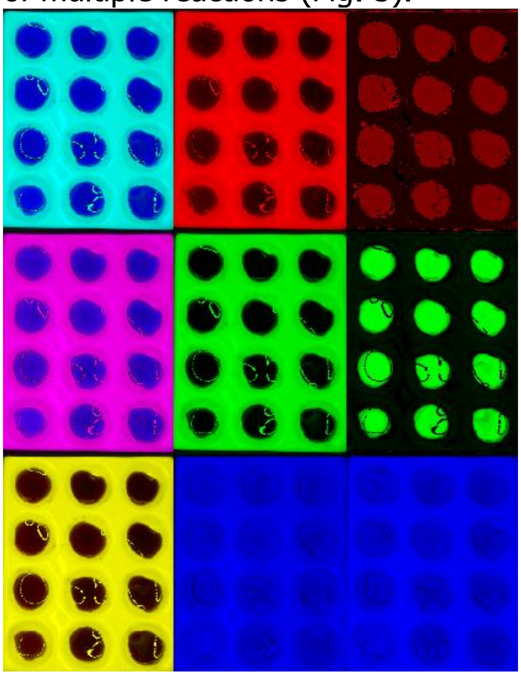

*Figure 3 - Image split into colour spaces and channels.*

A subsequent analysis step was to create histograms (Fig. 4) of the channel data to provide quantitative values to reduce pixel ranges for sample isolation. Below is the histogram exemplifying the saturation channel of the image above.

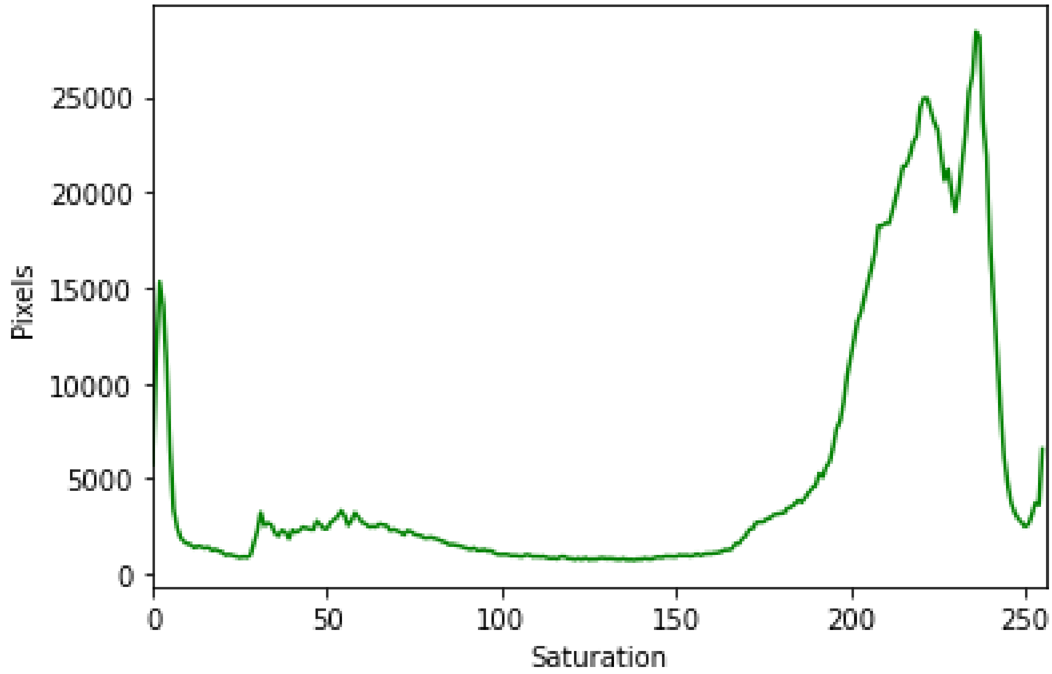


*Figure 4 – Histogram of colour space analysis.*

Here you can clearly see the spike of pixels with saturation between 0 and 20. This 0-20 range is where the majority of the background plate sits. Some small artifacts remain; however, these will be handled in a further stage.

The above images of the well plate were cropped by user input values, however, further along in the project edge detection algorithms were used to automatically crop the image to just the well plate. This is where the analysis done to find the reduced pixel ranges was implemented.

Having achieved exploratory colour space analysis, the project progressed on to implementing edge detection. To do this we take the image and following several steps of applying algorithms we produce a 'mask' (Fig. 5) which is then used to locate objects. The mask is a version of the image where pixels are either completely black, or completely white. Black pixels are a part of the background and white denotes the presence of the desired subject.

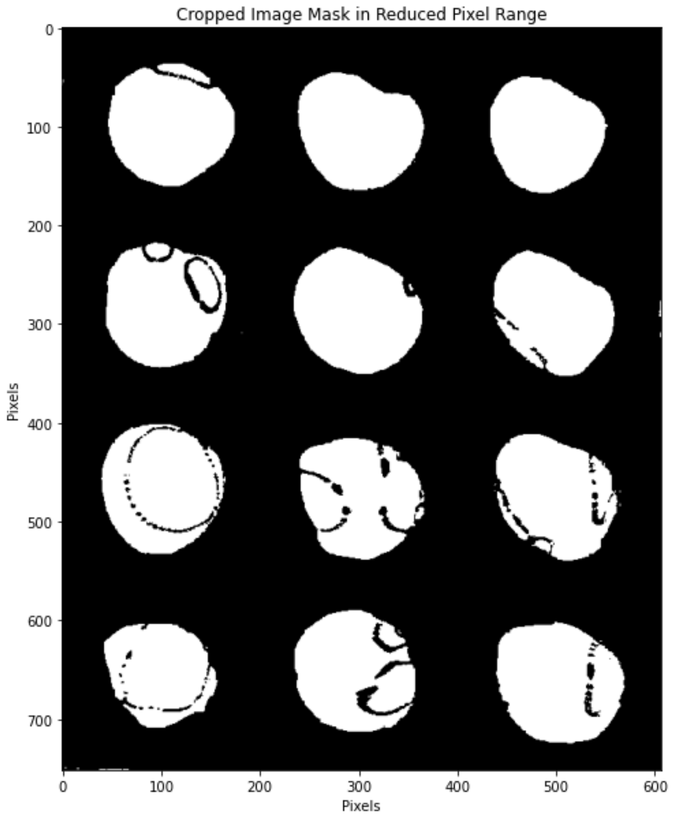


*Figure 5 - Image in reduced pixel range.*

To get rid of these artifacts we applied morphological algorithms, which enabled filling of all voids such as the ones produced by light glare.

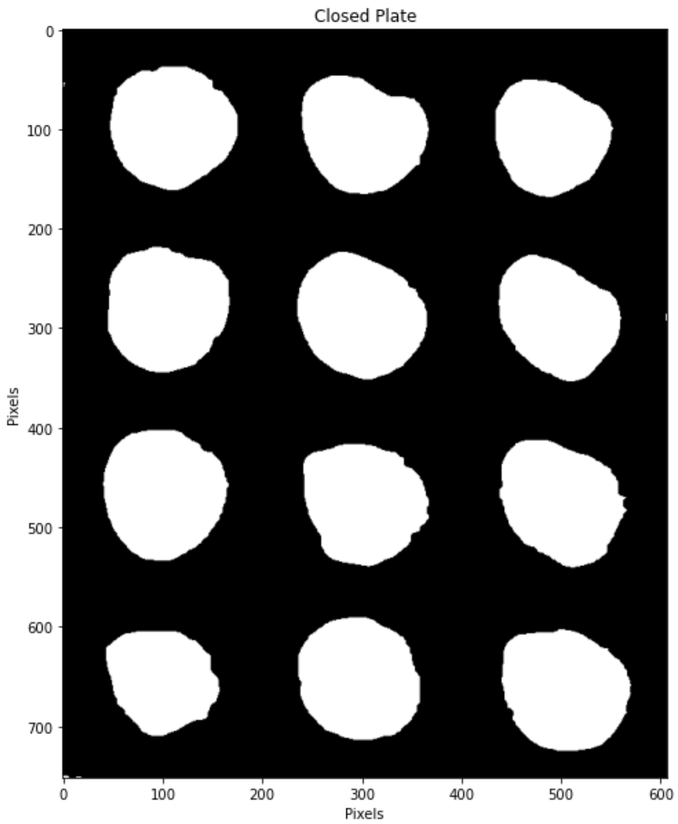


*Figure 6 - Mask after closing morphology.*

The closing stage removed almost all artifacts, within the limitations of the sample videos provided (Fig. 7).

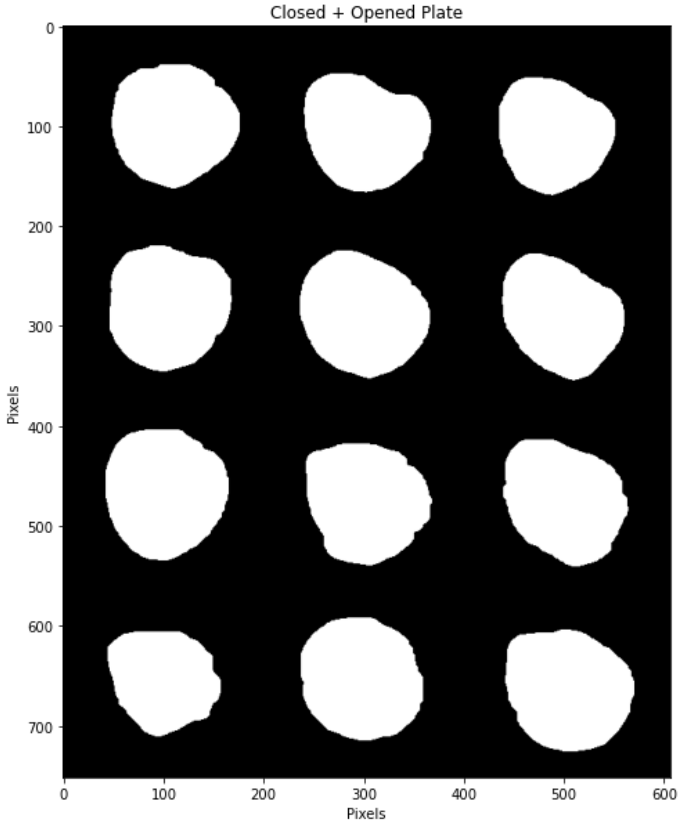


*Figure 7 - Mask after closing morphology.*

These final steps, though appearing to be a minor change; however, it is important as it helps ensure we are not missing any information and there are cases where larger areas need opened. Finally, we checked it against the original image by performing a simple overlaying function, this provides us with the following image.

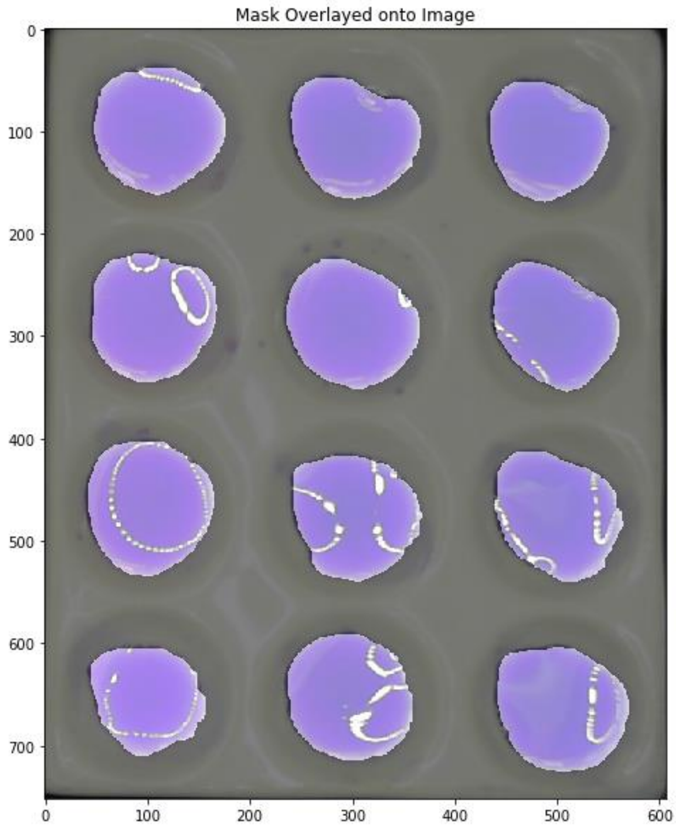


*Figure 8 - Overlay of mask onto original image.*

Subsequent workflow improvements involved automating image crops to the plate (Fig. 9) which, when confirmed, displays a bounding box of where it thinks the crop should occur (Fig. 10).

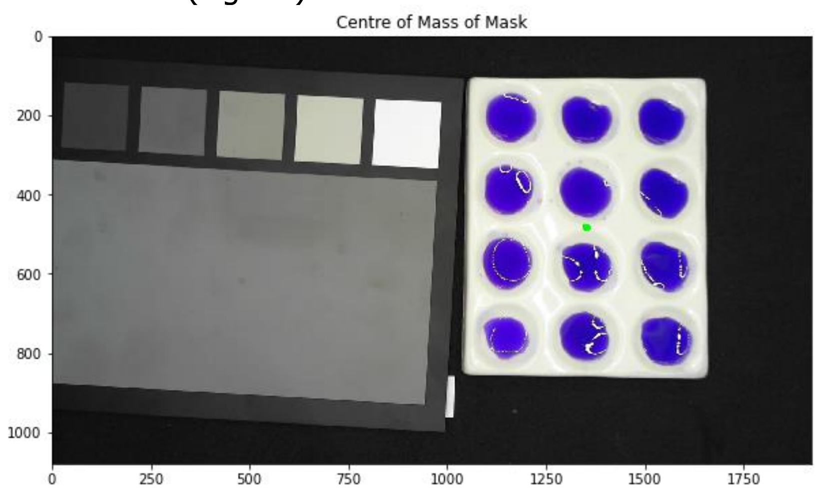


*Figure 9 – Centre of mass displayed on image.*

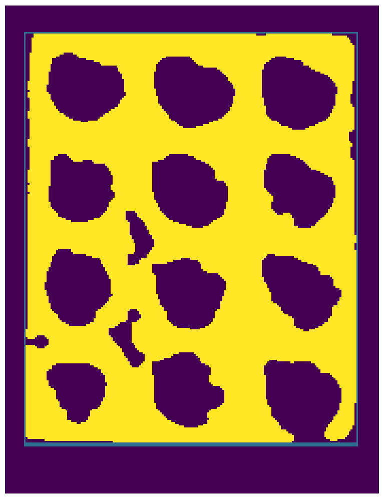

*Figure 10 - Bounding box on image.*

The next step was to find the location of the samples from the mask, and was achieved using edge detection methods to locate the contours in the image and increases detection robustness against poor image quality (Fig. 11).

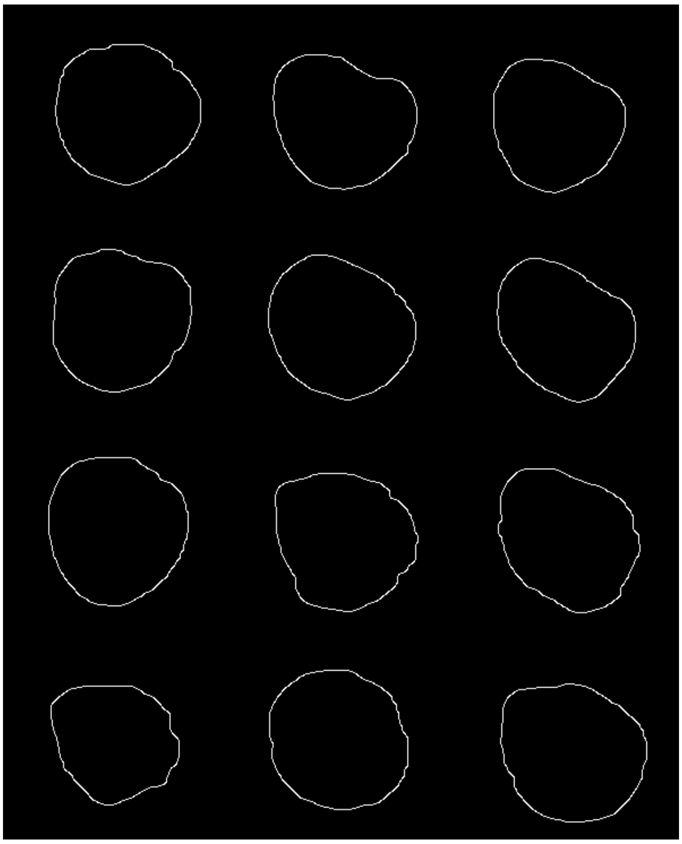

*Figure 11 - Canny edge detection applied.*

We next input the mask into a contour location function, outputting a hierarchical list of contours. The software then draws bounding boxes around each of the detected samples and presents the result to the user (Fig. 12).

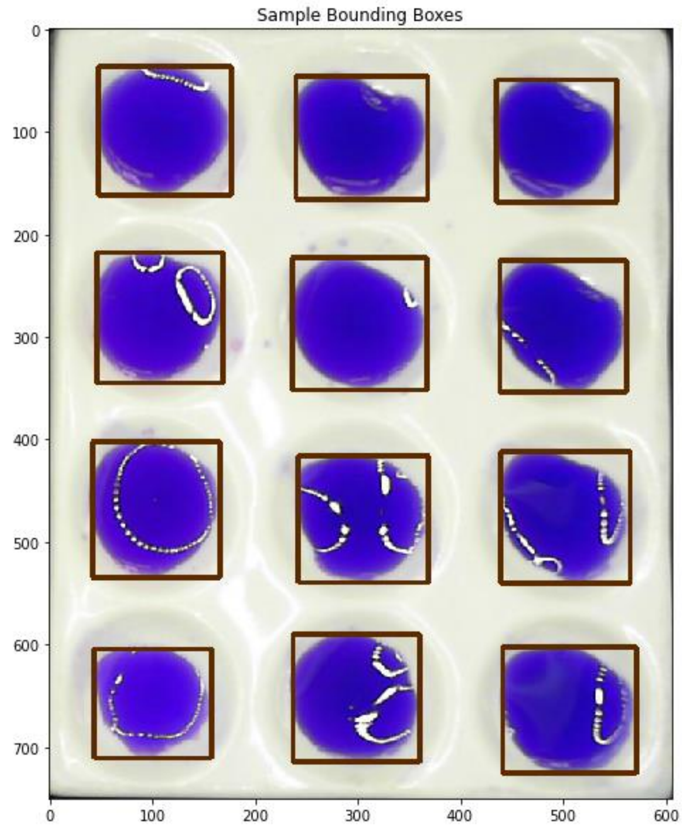

*Figure 12 - Bounding boxes drawn around samples.*

The location of the samples was then curated, creating the necessary bridging input into the existing code under development in the group.

## 7. Conclusions & Future Work

The main aim of the project was to upgrade existing vision-enabled kinetics software towards achieving a HTE implementation. Overall, a basis for future HTE has been achieved and integration of the new proof-of-concept module into the main software is now the subject of longer term investigation in Reid Group Research. The new software extension has been designed to be scalable and thus part of future work involves investigating the flexibility of the well plate environment being analysed.

Even compared to the existing software it presents the opportunity for a greatly reduced analysis time, rather than having to manually select each sample and then run the program multiple times. The user only has to input the video and the software implementation automatically will locate and select all samples on the plate.

Additional future work will include automation of certain colour range calculations and selections, as this remains the most involved (manual) step of the process for locating samples. Another advancement would be to better handle samples which are discoloured from the mean and to better achieve location of samples that provide poorer contrast with the background. To achieve this, more sophisticated detection methods may need to be implemented.

Finally, conversations with selected industry partners will continue in order to produce an implementation of full, tailored use across a range of industrial HTE applications.

## 8. Outputs, Data & Software Links

**Software development support links:**
Anaconda - https://www.anaconda.com/
Spyder - https://www.spyder-ide.org/
PEP8 - https://www.python.org/dev/peps/pep-0008/
OpenCV - https://opencv.org/
drawIO – draw.io

## 9. References

Bernd Jähne, H. H., 2000. *Computer Vision and Applications.* 1st ed. s.l.:Elsevier Inc..
Luigi da Vià, L. J. E., 2020. High-Throughput Photochemistry: The dawn of a new era with a bright future. *Chemistry Today,* 38(3), pp. 38-41.
Shevlin, M., 2017. Practical High-Throughput Experimentation for Chemists. *ACS Medicinal Chemistry Letters,* 8(6), pp. 601-607.