Automatic Monolayer Detection with Fluorescence Microscopy

Lachlan Catto[1,2] and Glen Pearce[1,2]

[1]School of Mathematical and Physical Sciences, Macquarie University, Australia.
[2]Quantum Optoelectronics, Manufacturing, CSIRO, Lindfield, Australia

September 2024

Acknowledgements

We would like to extend our sincere thanks to both CSIRO and Macquarie University for making this project possible. In particular, we wish to acknowledge the guidance and mentorship of our supervisor, Dr. Matt Broome from CSIRO Quantum Optoelectronics, who conceptualised the project and aided us in developing our skills and knowledge throughout.
Additionally, we are grateful for the support of our academic mentor Dr. Daniel Terno from the School of Mathematical and Physical Sciences, Macquarie University, who provided us with valuable academic feedback and perspectives throughout the duration of the project.

Contents

1 Executive summary

Locating single layer (monolayer) samples of transition metal dichalcogenides (TMDs) produced by mechanical exfoliation is a critical step in 2D material research. This step is both time and labour-intensive, but is key to researching devices built from 2D materials. Monolayer TMDs are particularly interesting due to their unique optical and electrical properties with applications technologies including transistors, sensors, and other optical devices.

1.1 Aims and Objectives

The primary aim was to automate the identification of monolayer TMDs, through fluorescence microscopy.

Objectives included:

- Developing software to control a fluorescence microscope equipped with a camera and motorised stage.
- Automate the scanning, imaging and stitching of sample images.
- Implementing automatic identification and analysis of the monolayers.
- Package the software within an accessible interface.

1.2 Methods and Equipment

- **Hardware**: A Thorlabs Cerna based fluorescence microscope with CMOS camera and a two-axis translation stage
- **Software**: Custom Python code was developed to automate the monolayer identification process and to consolidate the scanning, detection and analysis logic into one accessible interface. Multiple threads were used to simultaneously run many different processes.
  The main processes were:
  - Sample scanning and image stitching, using Thorlabs software kits and PyTorch.
  - Image processing and monolayer identification, primarily with OpenCV.
  - Results display and monolayer analysis, using OpenCV, NumPy, SciPy, and Tkinter.
  - Graphical User Interface (GUI), developed with Tkinter.

1.3 Results

- A functional and accessible software interface was developed.
- Device throughput at this step was increased by approximately ten times. Removing one of the largest bottlenecks to device production with monolayer TMDs.
- The software was estimated to be able to detect over 95% of monolayers within a single scan with a variable false positive rate.

1.4 Conclusion

The identification of monolayer TMDs by fluorescence microscopy was successfully automated through the use of Thorlabs equipment and custom Python code. Achieving a significant increase in device throughput, helping to enable further advancements in 2D device research.

1.5 Recommendations

We recommend that CSIRO and other institutes which rely on fluorescence microscopy incorporate and modify this software to improve the efficiency of their imaging and detection methods.

2 Key terms

- **2D Material**: A crystal with negligible thickness, normally only one or a few atoms thick.
- **Monolayer**: The thinnest possible layer of a crystal.
- **Bilayer**: A crystal with a thickness equivalent to two monolayers.
- **Fluorescence Microscopy (FM)**: A method of microscopy in which light is directed onto the microscope stage, triggering fluorescence of the sample (or part there of). The incident light is filtered out, isolating the light from the fluorescence of the sample.
- **Transition metal dichalcogenides (TMDs)**: Semiconductors formed from a lattice of transition metal and chalcogen atoms.

3 Relevance & Introduction

As humans, we interact with materials every day, and understanding the properties of these materials has given rise to many technological advances throughout history. Within material science, semiconductors are a key area of study due to their multitude of interesting properties. Properties which have resulted in their crucial role in modern technologies including computers, solar panels and practically every electronic device. One particularly interesting area of study, is that of 2D materials. A 2D material is a crystal of negligible thickness, usually only a few atoms thick depending on the crystal structure [1]. These 2D materials can be formed from many different types of materials, including semiconductors, and they display properties which are often very different to the properties of the 3D bulk material [2]. This is largely due to their diverse electrical, physical, and chemical properties, including large surface-to-volume ratios, photoluminescence characteristics, biological compatibility, and tunable electric properties [3]. There are numerous applications for such properties including, sensing, transistors, photodetectors, ultra-fast lasers, new optoelectronic devices and wearable electronics [1][3].

One promising subcategory of 2D semiconductor materials is transition metal dichalcogenides (TMDs), which display many of these desirable optoelectronic properties. TMDs are a key area of research in the Commonwealth Scientific and Industrial Research Organisation (CSIRO). However, the process to obtain monolayer (single layer thick) samples of a crystal can be difficult and time-consuming. Production of 2D TMD samples has proven difficult to scale within industry. However, research often requires much smaller quantities of these materials, and one of the most common methods of small scale sample production is mechanical exfoliation (ME). This is commonly referred to as the scotch tape method and involves using tape to peel layers off the bulk material to reduce the thickness before attaching it to a substrate. The tape is then slowly peeled from the substrate to hopefully leave behind a 2D sheet of material [4]. This method often results in a wide variety of small crystal samples on the substrate with varying thicknesses.

To identify the monolayers on which research can then be conducted, the substrate is viewed under a fluorescence microscope, where light excites the smaller band gap in the monolayer crystals, causing them to fluoresce. This fluorescence can be observed and used as a reliable method of identifying which sections of the crystal samples are monolayer. However, the density of these samples on the substrate is often quite low, meaning a significant portion of time is spent searching for these monolayer samples under the microscope instead of actively progressing 2D material research. Currently, suitable samples must be found manually through tedious inspection of the entire substrate and noting down the approximate positions of desirable samples. Our goal is to simplify this process of locating monolayers post exfoliation by automating the sample location procedure through the use of a camera, motorised microscope stage and Python.

4 Aims and Objectives

This project aims to optimise the identification of monolayer samples by automating the fluorescence microscopy setup. Extended goals of this project include the automatic identification and basic analysis of these samples within an accessible software interface.

5 Methods

5.1 Equipment

- Fluorescence microscope setup using the Thorlabs Cerna platform
  - SOLIS-3C High-Power LED
  - CES2200 Epi-Illuminator Module
  - PLS Two-Axis Translation Stage with MCM301 Stepper Motor Controller
  - MCMK3 3-Knob Joystick
  - kiralux CMOS Sensor Camera
  - Nikon LU Plan EWLD 20x/0.4 and 5x/0.15 objective lenses
- Dell Opiplex Tower Plus 7020 running Windows 11 Pro v23H2 with Anaconda
- Conda v24.5.0 virtual environment running Python v3.11.7

5.2 Method

This project sort to develop a sequence where a user could input the location of the sample and software would automatically move the microscope, taking images as it goes, until the entire sample has been imaged. These images could then be stitched together to form an overview of the entire sample, from which monolayers can be identified and assessed.

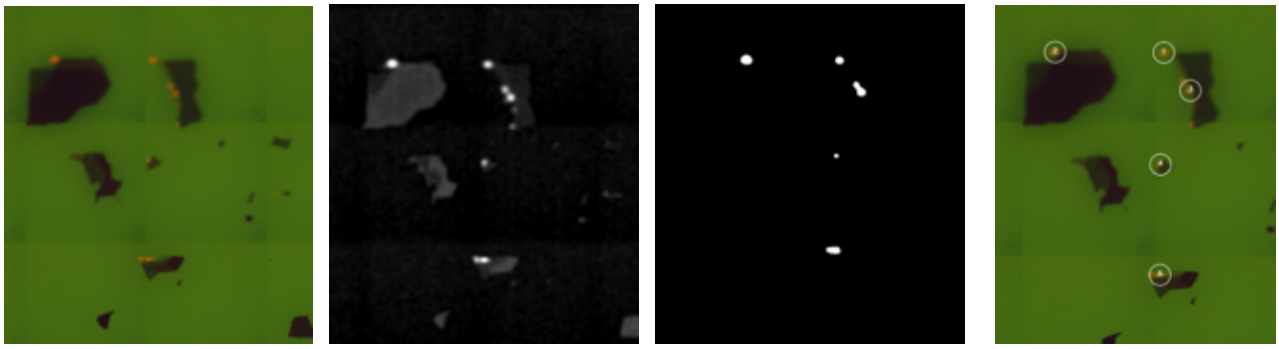
Python code was developed to interface with both the Thorlabs camera and motorised stage, utilising the associated Thorlabs software development kits. A coordinate system was set up based on the position of the stage given by the stage controller, and conversion functions were developed to convert the coordinates of the stage to pixel locations on the images. A basic algorithm was created to scan across an input region of the stage, defined by two opposite corners of the sample, and taking images throughout the scan. Based on the size of the image in the coordinates of the stage, the stage would move in increments slightly smaller than the image size before recording the image and its coordinates, ensuring the entire area was scanned. These images were then stitched together on a larger canvas based on their locations.

The image stitching presented a particular challenge due to the large amount of pixel data and thus processing time. To counteract this issue, the program was designed to use multi-threading, allowing the image processing code to run simultaneously with the algorithm code. Extending the multi-threading to the entire program also allowed image collection, the user interface and various other components to be processed in parallel, improving the overall user experience. In order to further speed up the image stitching, the images were scaled down where possible and the image blending code was modified to use PyTorch for the performance gains and GPU acceleration capability.

A GUI was developed using Tkinter to improve the usability of the software. This provided the user with a live view of the camera feed and image stitching process, as well as an intuitive interface with which to interact with the software. Multiple separate tabs were used in the GUI to manage the variety of features developed. These included a main tab for the overall flow of the program, a calibration tab for adjusting settings, and a results/analysis tab for viewing the final results from the program. The main tab included buttons to set the location of the area to be scanned, a live view of

both the camera and the stitched result, a progress bar and other critical elements such as movement buttons. The calibration tab contained a variety of settings including the camera exposure, gain and rotation, adjustments for different lenses and a list of the connected devices.

Once the scanning algorithm was working consistently, the monolayer identification process could then be developed and tested. Firstly, the final image of the sample area was passed to a post-processing function, where the image would be down-scaled to reduce noise and speed up the computation. Next, the image was slightly blurred to reduce the likelihood of accidental monolayer detection from pixel noise or sample contamination. The image was further processed by converting it to grey-scale with a custom bias towards colours matching the expected fluorescence and against all other colours. This resulted in an image where white patches represented potential fluorescence and darker patches to be ignored. By removing all pixels below a certain brightness threshold, the monolayers could be clearly identified, and the outlines found using OpenCV. The location, area and individual image of each monolayer were identified using the data given by the OpenCV outlines. Basic quality metrics, such as intensity, variance, were calculated from the monolayer images. Finally, this data was summarised and passed to the results tab of the GUI for inspection by the user.



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

Figure 1: Post-Processing, Monolayer Identification Process: (a) Original sample image produced by the scanning algorithm. (b) Image blurred for noise reduction and converted to grey-scale, with pixels near the fluorescence colour having higher intensity. (c) Threshold applied to the image to remove low intensity pixels, isolating areas of fluorescence. (d) Final image with detected monolayers marked.

To ensure constant scan quality across large samples, an autofocus routine was implemented. This allowed the software to adjust to focus changes caused by fluctuations in substrate height. A focus measure was needed which was robust against noise, fast and exposure independent. To achieve this, frames of the live video feed were routinely analysed by taking a 2D Fourier Transform. The result was radially averaged and converted into a Power Spectral Density (PSD). To simplify the power law relation, the PSD was plotted on a log-log scale and a linear fit determined using SciPy. The gradient of this fit was used as the focus measure. In-focus images displayed stronger high-frequency components due to sharper edges within the image, resulting in a shallower gradient.

To focus on the sample, an iterative search was used to quickly approach the optimal focus. This involved testing the focus at a wide range of points, and then searching a narrower range around the best focus point. This process was iterated until the focus was within a dynamically adjusted threshold determined from the lens magnification. This search method reduced issues with local minima and maxima caused by other focal planes.

6 Results & Analysis

The software was able to successfully identify and locate monolayer TMDs with an optically excitable band gap. Precise measurements of the monolayer identification rates were difficult to achieve. However, it is estimated that the final software solution identified more than 95% of monolayers on the sample, with any undetected monolayers being unsuitable for device fabrication due to their small size.

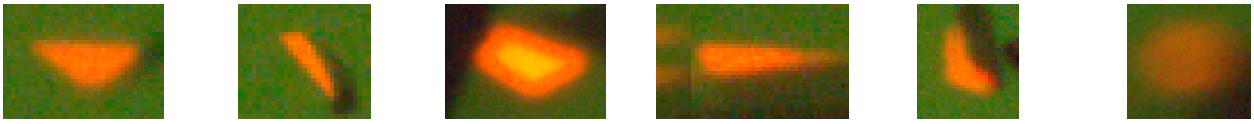
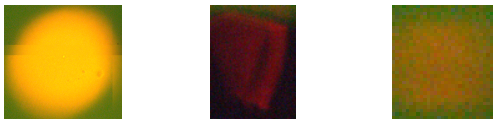


Figure 2: Successfully extracted images of monolayers within the sample, as would be displayed in a table within the results tab.

A false positive rate between 1% and 15% was obtained, caused most often by the detection of other fluorescent debris (fig. 3). However, this was highly sample dependent, and samples with excess contamination could have much higher rates. These false positives were easily filtered out manually when viewing the monolayer images within the results table and were not an issue overall. This could be mitigated in future with additional filtering with machine learning approaches, if required.

The software would also detect other monolayers which are not suitable for device fabrication, such as monolayers atop other bulk material or small, broken monolayers. Depending on the sample quality, this could be upward of 50% of the monolayers detected. Fortunately, these cases could also be easily filtered, either manually, or using the quality metrics.



(a) (b) (c)

Figure 3: Examples of undesirable detection cases: (a) False positives, (b) Monolayer directly on top of bulk material, (c) Monolayer out of focus, potentially detecting multiple layers as one.

The detected monolayers were successfully highlighted on the overall sample image.

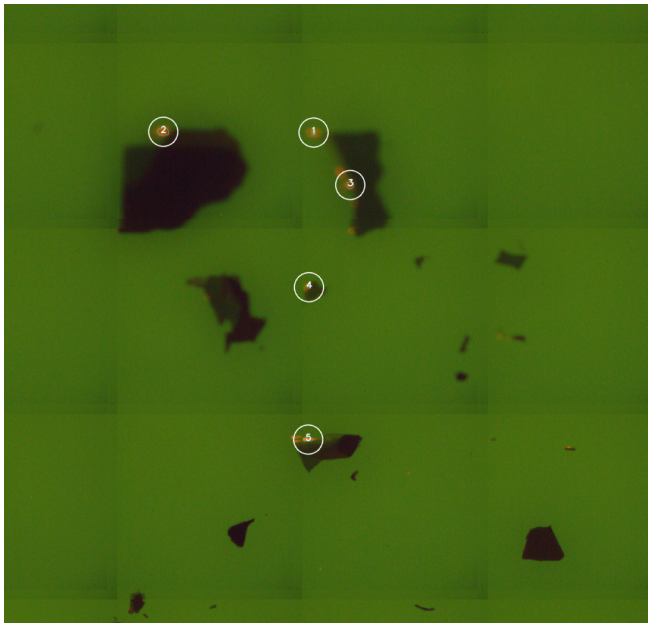


Figure 4: Subsection of the final output image with detected monolayers circled and labelled.

The scanning algorithm managed to scan large samples consistently while staying within a reasonable threshold of focus. A 3 cm by 3 cm sample took approximately 30 minutes to scan and process on 5x magnification, while a 0.85 cm by 2 cm sample took approximately 30 minutes on 20x magnification.

Additionally, the autofocus functionality successfully maintained focus over the majority of the sample, with only some areas out of focus, most evident when the scan encountered a prolonged absence of material to focus on.

The memory and CPU usage of the computer (see 5.1) was maintained well below 70% even for the larger scans, indicating that the software should be able to run on most modern PC hardware without issue.

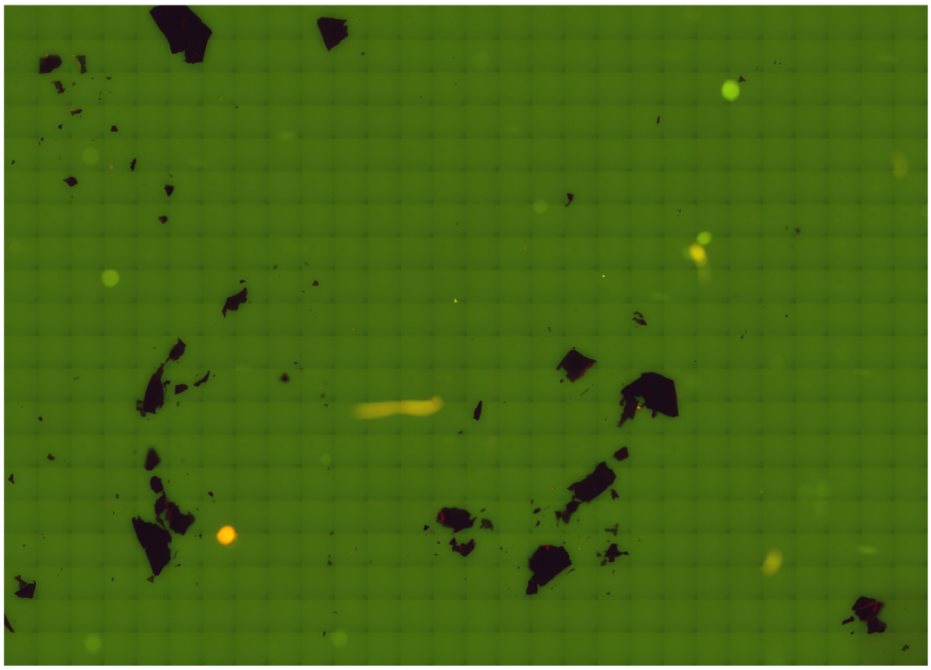


Figure 5: Example scan of a sample containing TMD flakes under 20x magnification. This sample was approximately 0.7 cm by 0.95 cm in size and scanned within 20 minutes.

Based on feedback from users, it is estimated that this runtime equates to a greater than ten-fold increase in device throughput due to the greater efficiency of monolayer identification and categorisation compared to manual methods. This effectively consolidated a day of continuous manual work into an equivalent 30–60 minutes of work, operating the software intermittently. Overall, this produced a marked increase in device throughput within this step of the fabrication process.

The final software interface was accessible and easy for users to interact with. Many features were incorporated to ensure the software was as accessible as possible. These included:

- Intuitive movement interface, via buttons, the live view window or by double-clicking on the results image.
- Monolayer summary table in the results tab. Sorted by largest to smallest, displays the statistics and allows quick flake finding by clicking on a monolayer. This automatically moves the stage such that the monolayer is within the microscope live view.
- Live view scale bar
- Simple calibration options including a fluorescence colour picker, rotation adjustment, save directory selector and camera settings.

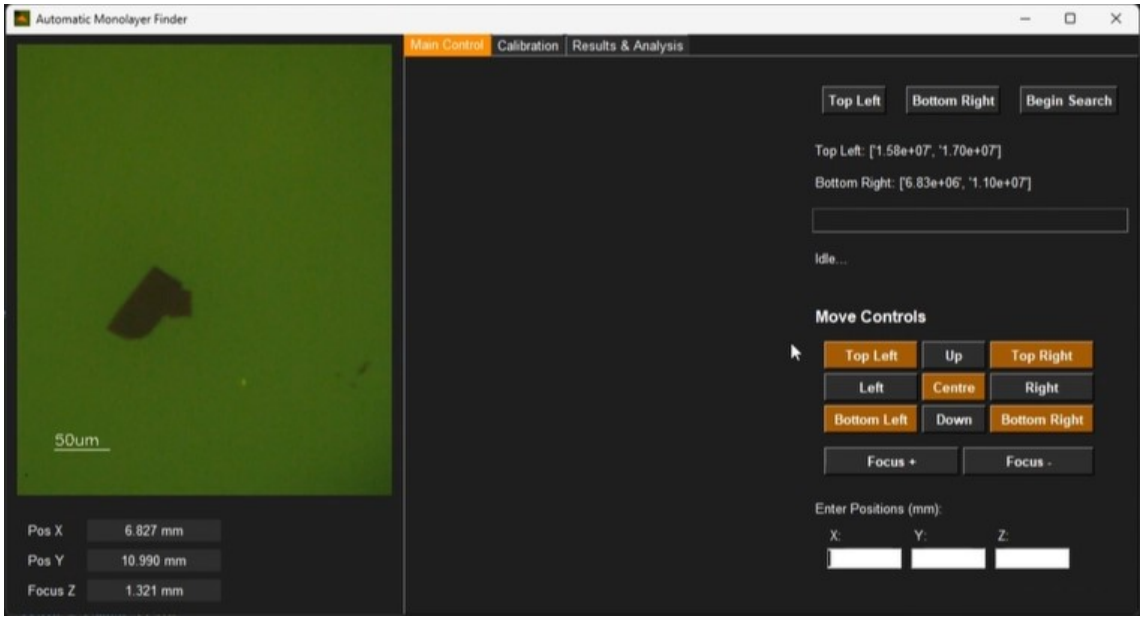


Figure 6: Final software interface: Main Tab

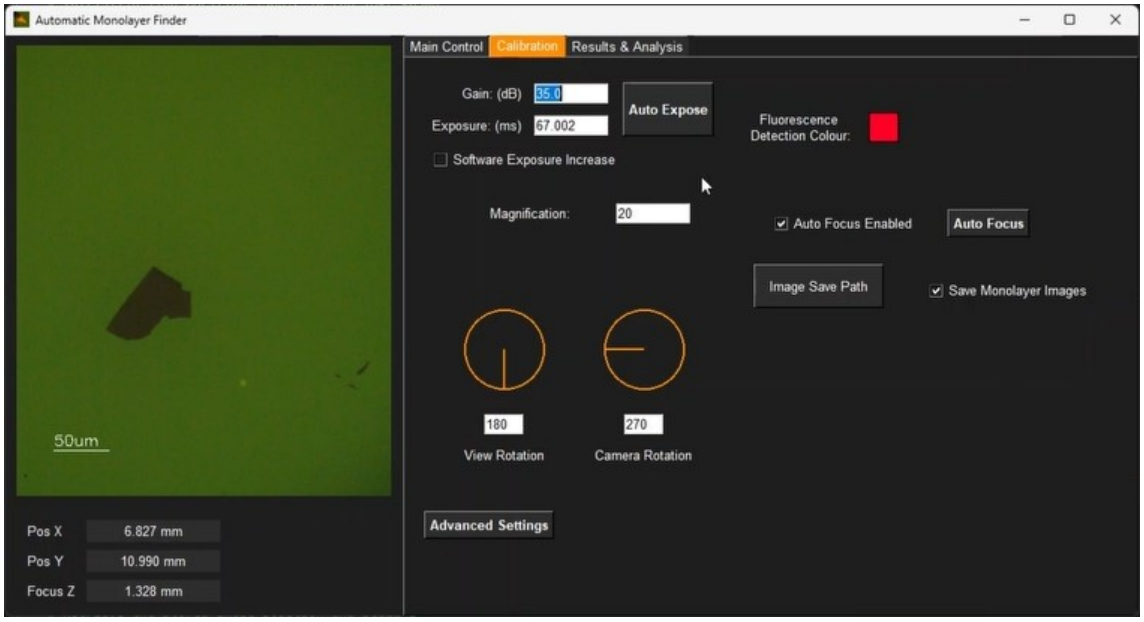


Figure 7: Final software interface: Calibration Tab

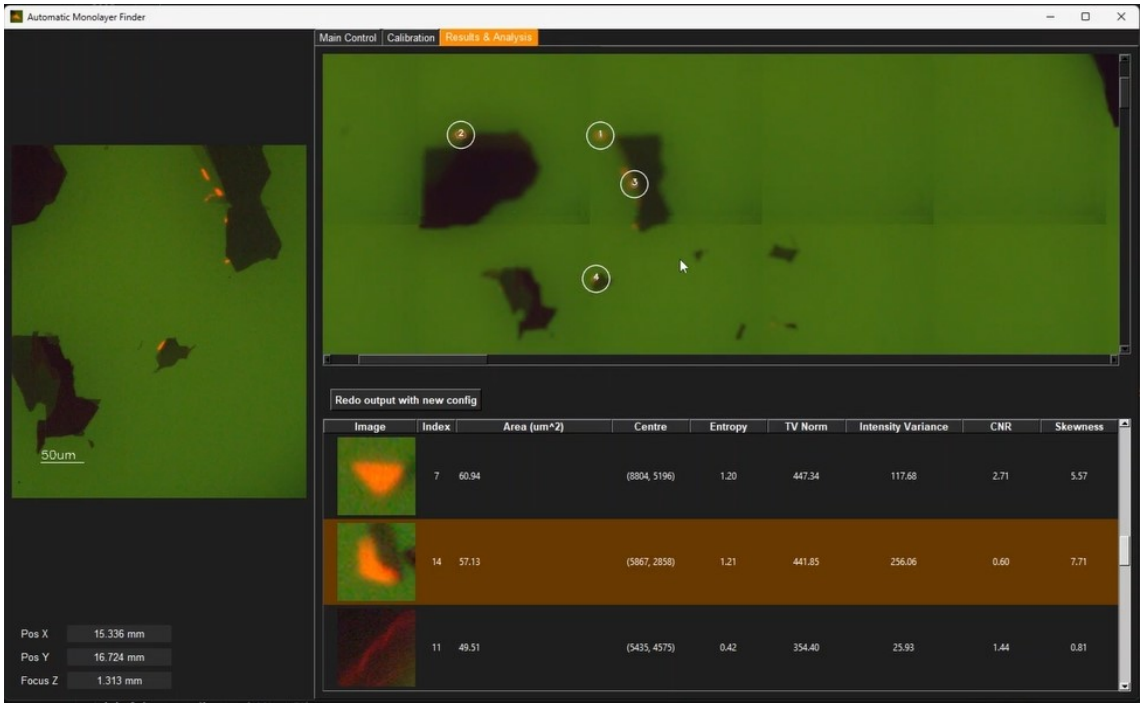


Figure 8: Final software interface: Results Tab

7 Conclusions

A software solution was successfully developed to optimise the identification process of monolayers through automation with fluorescence microscopy. The device throughput at this step of the fabrication process was increased more than ten-fold. Automatic identification and analysis of monolayer samples was achieved within the software, providing quantified quality metrics and easy access to ideal monolayer samples. The software was successfully packaged into an accessible, easy-to-use GUI, simplifying the functionality and user experience. Overall, monolayer detection was shown to be accurate and efficient, enabling further access to 2D material research.

8 Recommendations

The increase in device fabrication throughput obtained with the implementation of this software is promising. We recommended that CSIRO implement this into the 2D material research workflow. This may aid the accessibility of future research goals by increasing the quantity and quality of devices and data. These automated processes also have potential for use in other areas of science which rely on fluorescence microscopy, such as biology.

The software has substantial room for improvement and would be particularly essential if the use case is expanded.
Such improvements include:

- Adjusting the autofocus routine to prioritise objects that match the fluorescence colour. This would aid in ensuring that clear images of the monolayers are obtained for analysis.
- Refactoring to handle threads more properly and safely, particularly when interfacing with the hardware. Additionally, a stop button could also be added, allowing the search to be aborted and restarted partway through without a software restart.
- Machine learning could be used to further filter results, removing false positives or other undesirable samples. These methods could also be used to detect bilayers. [5]
- A rigorous method of sample positioning on the stage, in conjunction with a save and load feature, could be implemented to precisely view previously scanned samples again without the need for a rescan.

9 Software Availability

The software has been designed to be as open source as possible, working around the limitations of the proprietary licensing of the Thorlabs software development kits. As such, our source code is available on GitHub at `https://github.com/GlenAlan/CSIRO-Monolayer-Fluorescence`. Thorlabs software development kits for both the camera and stage controller may need to be added by the user.

10 Statement of roles and contribution

Lachlan was responsible for the initial GUI design in the R&D work, including some useful features such as button functionality, image zoom following the final scan image, and a scale on the live view image. For this report, Lachlan contributed to the relevance & introduction, aims, methods, results & analysis, and conclusion sections.

Glen was responsible for the development of the scanning algorithm, image stitching, monolayer detection logic, hardware interfacing, automatic focus/exposure, and other backend logic. For this report, Glen contributed to the executive summary, relevance & introduction, aims, methods, results & analysis and software availability.

References

[1] T. Tan, X. Jiang, C. Wang, B. Yao, and H. Zhang, Advanced Science **7**, 2000058 (2020).

[2] K. S. Novoselov, A. K. Geim, S. V. Morozov, D.-e. Jiang, Y. Zhang, S. V. Dubonos, I. V. Grigorieva, and A. A. Firsov, science **306**, 666 (2004).

[3] J. Z. Hassan, A. Raza, Z. U. D. Babar, U. Qumar, N. T. Kaner, and A. Cassinese, Journal of Materials Chemistry A **11**, 6016 (2023).

[4] E. Gao, S.-Z. Lin, Z. Qin, M. J. Buehler, X.-Q. Feng, and Z. Xu, Journal of the Mechanics and Physics of Solids **115**, 248 (2018).

[5] J.-L. Uslu, T. Ouaj, D. Tebbe, A. Nekrasov, J. H. Bertram, M. Schütte, K. Watanabe, T. Taniguchi, B. Beschoten, L. Waldecker, *et al.*, Machine Learning: Science and Technology **5**, 015027 (2024).

Appendix

Video Demonstration