RULER

An Automated method of Sperm Measurement

Application, Analysis, and Assessment By Nevin Ndonwi

CSE 5504 Geometric Computing for Biomedicine

Introduction:

RULER is an application that supports an automatic algorithmic measuring approach to the problem of measuring the length of sperm cells of fruit flies (it is also extendible to other animal cell types). In addition, it also allows for hand tracing and a mix of other features to maximize the user experience, assist researchers, and be extensible to any other research imaging problem.

General Resources:

- *Github repository containing the code, executable file access, file testing, documentation, and other important submission components:* https://github.com/nevinndonwi/RULER-application-
- *Program Executable files (either use the linux executable with windows ubuntu/WSL or another method or clone the repository and build dependencies locally for maximum utility, the windows executable is present if all else fails):* https://drive.google.com/drive/folders/16GqBbQO1myYrcalqOrmq4C-Ca_wnQcrR?usp=sharing

Application Features Completed:

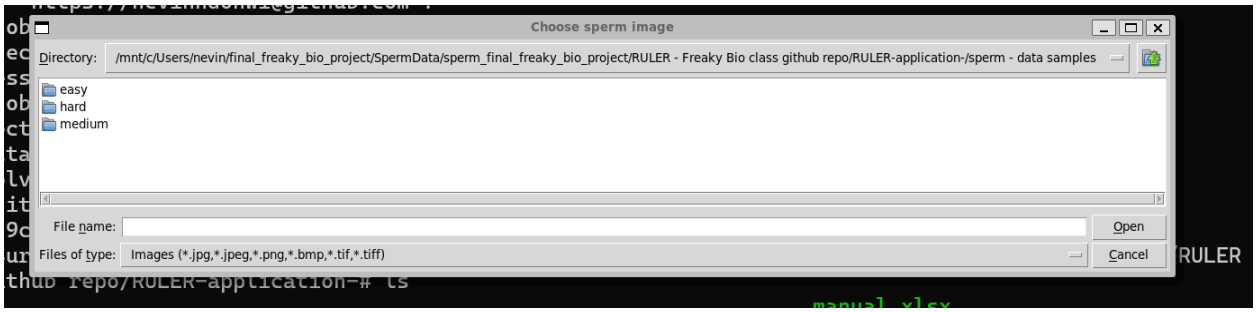
- A UI is constructed so that the user can specify the image input and see the output. (Required)
- The user can gain measurement insights on different sperm cells within the application. (Required)
- The user can interact with the image by selecting a region and highlighting all of the sperm cells in that region in threshold measuring mode. (Required)
- The user can click on points in the cell to highlight the cell in threshold mode. (Required)
- The trace of threshold mode component selection or automeasurement component results can be put over the image in order to provide a rough sketch over the cell for user measurement verification. (Required)
- The GUI can also show the resulting segmented sperm cell (e.g., as a skeleton or a binary mask) overlayed on top of the image, so that the user can check its correctness. (Required)
- Customize the color highlighting for different sperm cells to better label it for researchers (present in threshold mode). (Wishlist)
- Create a threshold toggle so that the user can see the selections under different threshold values to gain insights (threshold mode). (Wishlist)
- Provided an image inverter button to see if different thresholds or if inverting the image would work better on different images depending on the inversion in order to make this project extendible to other computing for biomedicine projects and not just this one depending on the data and the cells that were used. (Wishlist).

Application breakdown:

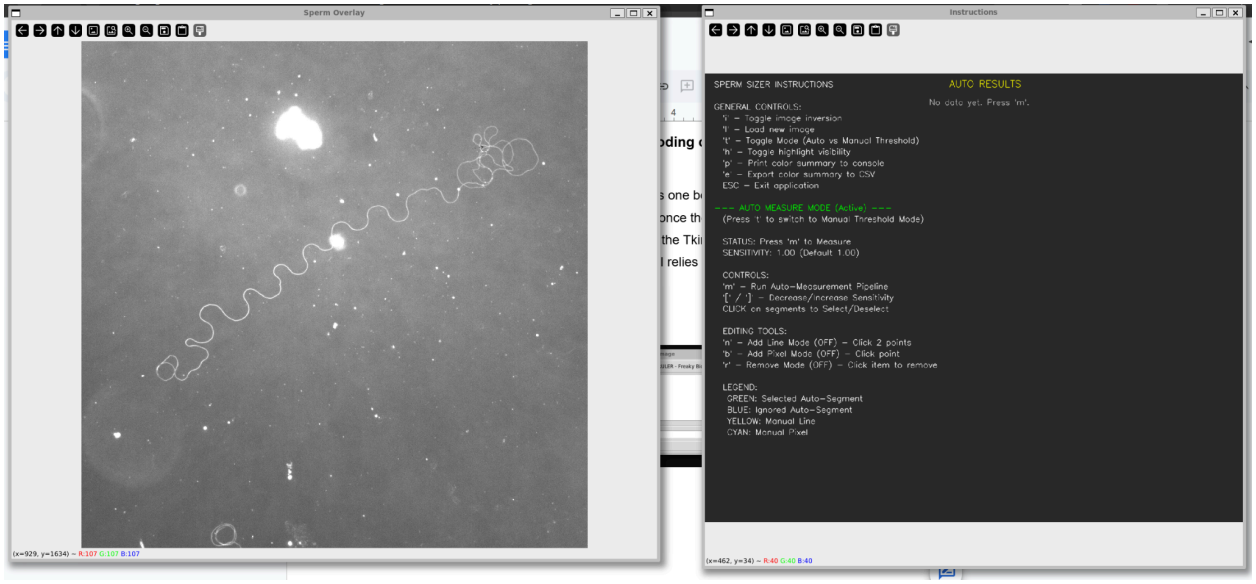
Readme on how to setup and use the application linked here:
https://github.com/nevinndonwi/RULER-application-/blob/main/README.md

Description of core algorithms, significant coding components, GUI development:

I created a GUI that is made up of 2 windows one being an instructions window and one being the image window which will display once the user goes through the first GUI to pick their image file. It uses the openCV and the Tkinter library which supports zoom in and zoom out and saving the image. The UI relies on keyboard controls and clicks to be able to function and best help the user.



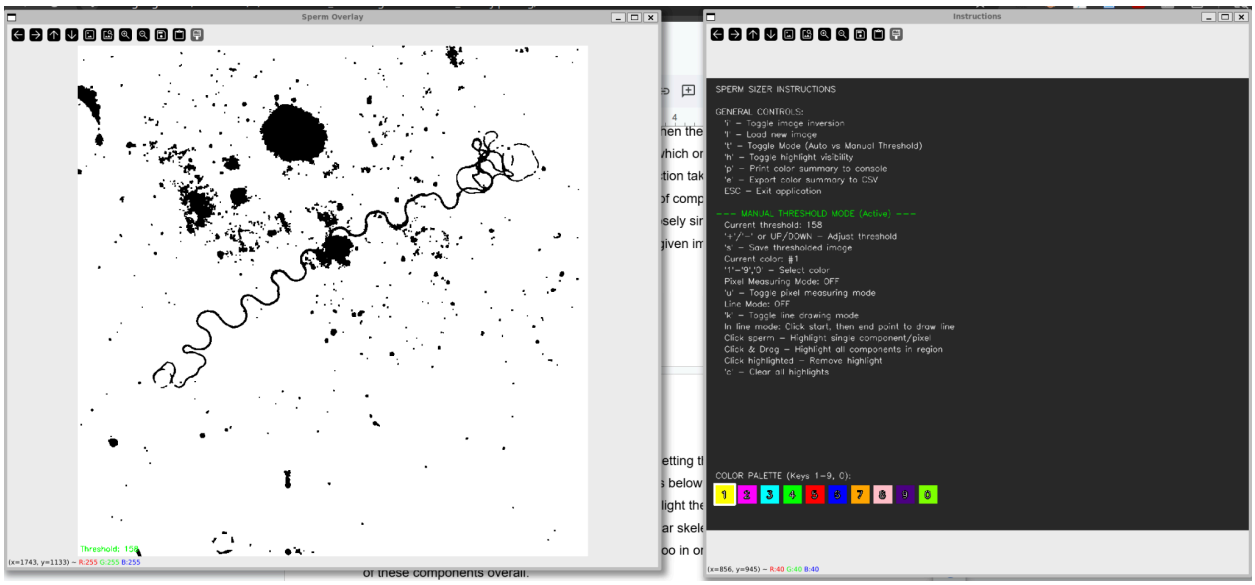
- Initial “pick a file” GUI



- Windows loaded with info (image and instructions) [in Automeasure mode]

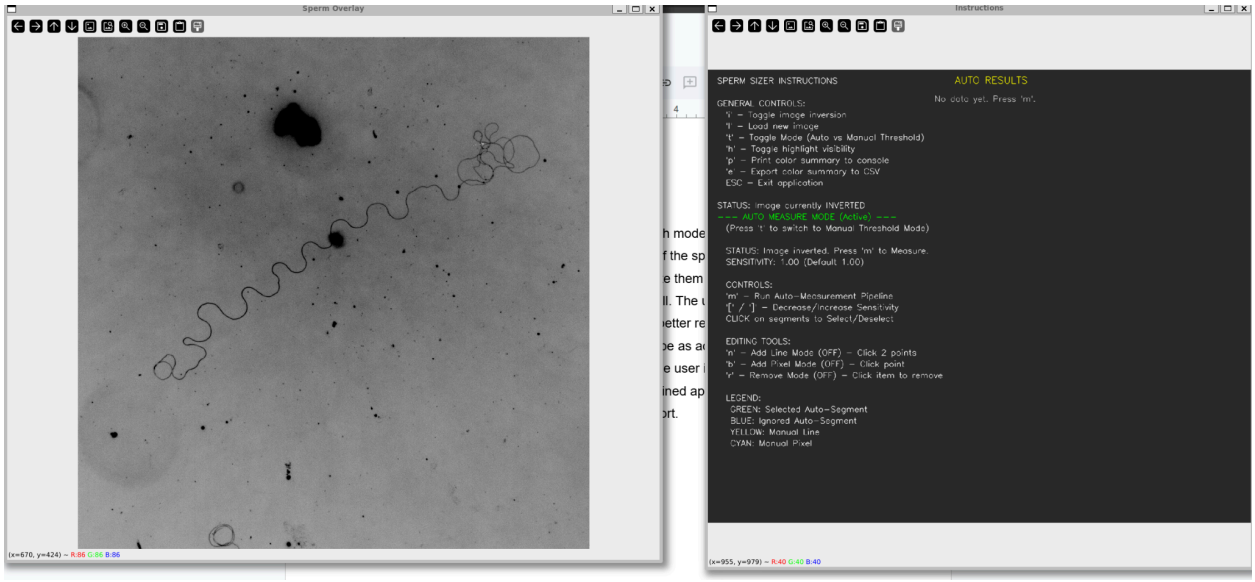
RULER lets users choose between automeasuring mode which then will measure components of sperm cells in a given sample image and then let the user choose which components to consider in measurement. The approach for this is to use Gaussian blurring, a Sato filter, and ridge detection in order to use eigenvalues to get the sperm structure. Then Otsu's method is applied (which calculates an adaptive threshold in order to get the most out of an image) in order to get the image binary and detect components within the given image sample. The components are then skeletonized and then cleaned with opening and closing operations to get rid of small objects and small holes before being measured in order to mitigate noise and other irrelevant data. Once this is complete, all the total components and their lengths are calculated so that these have been found, so that when the user selects a component, it will get highlighted for them and let the user select which ones they want included in the measurement so the actual measurement and selection takes place at a faster rate after the initial loading of the component and calculation of component length data after conducting the steps above for the entire image (loosely similar to how in our Mathematica assignments we made an octree of a given image to make subsequent operations faster).

In addition, I also created a threshold mode setting that increases and decreases the threshold of an image in order to eliminate pixels below a certain factor in order to help the user to manually find components and highlight them for measurement by interacting with the GUI with a gaussian blur. A similar skeletonization and cleaning approach that the automeasurer uses is used here too in order to get the measurement of these components overall.

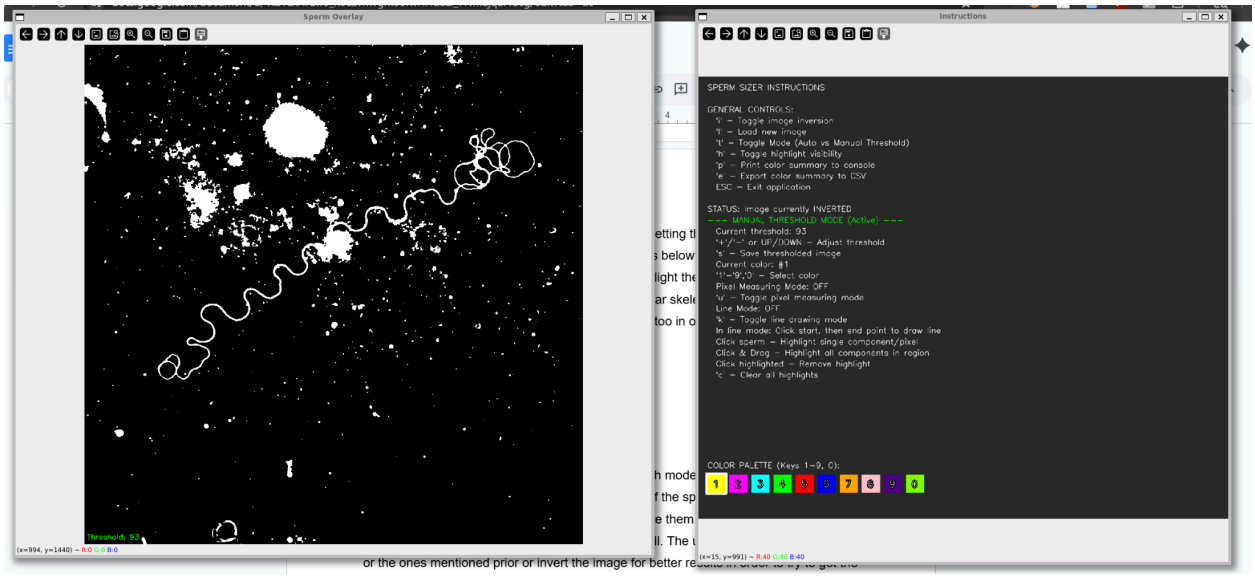


- Seeing the image in threshold mode

Users are able to draw lines and pixels in both modes to connect sperm cells together which are measured and treated like part of the sperm measurements. In threshold mode, users can select cells and designate them to be different colors to select multiple sperm cells for measurements as well. The user can use these features or the ones mentioned prior or invert the image for better results in order to try to get the measurement of a given sperm cell in the image to be as accurate as possible. The application overall, supports multiple modes to let the user identify, select, and measure the length of the sperm(s) by using previously explained approaches to create a streamlined experience and best provide user support.



- Inverting the image in automeasure mode



- Inverting the image in threshold mode

Quantitative Analysis

Resources

- *All Data, figures, tests, and conclusions are linked here for reference: https://github.com/nevinndonwi/RULER-application-/blob/main/biocomputing%20document%20feature%20list%20archive%20plan%20and%20data%20collection.pdf*
- *All Error Measurements, conclusions, and additional notes regarding measuring can be found here as well:* https://github.com/nevinndonwi/RULER-application-/blob/main/relative_error.csv

Overall Findings

In order to calculate the error of my sperm measurements, I used the relative error formula consisting of (expected_value - achieved_value)/expected_value and converted that into a percentage. I did that for every single image that I measured and found that the easy photos had a mean relative error of 2.89%, medium photos had a mean relative error of 18.34%, and hard photos had a mean relative error of 27.54%. These results were mostly dependent on the contrast between the sperm pixels and the noise present in the image and based on the tests recorded, the application works and is viable, more detail on each test can be found in the github repository regarding image time, what the images looked like, and other interesting behaviors.

In terms of wall clock time to measure, easy photos took on average 1 minute , medium photos took on average 2.2 minutes, and hard photos took on average 5.25 minutes. These measurements show that the application works and that significant time is reduced in measuring compared to the hand tracing approach since it was not as resorted to since for easy, medium, and some hard images depending on the contrast, most times, more components were able to be selected when measuring which reduced the need for hand tracing.

Overall, more testing and finetuning with threshold mode would be ideal but it is usable in its current form for basic and semicomplex measuring tasks and takes significantly less time than hand tracing overall. Different photos needed different tools to get the best measurement so the wall clock time value considers whether or not the algorithm was run or if one had to try thresholding or tracing or something else. More information can be found in the relative_error.csv file in terms of what the tester did to measure the sperm at that point but in conclusion RULER works well overall but higher image complexity and noise can cause more problems as they appear.

Tips on Application Usage

- This is only present in threshold mode but the line addition function is very sensitive, if you click anywhere near a line then it will be deleted. This can make it hard to connect 2 lines together to make traces or to connect pixels to lines to fill in space, generally you won't run into this too much when using the application since in threshold mode there are usually pixels and components small enough to buffer the effect but it is still something to note.
- Threshold mode's sperm measurement accuracy is very sensitive to noise and pixels, it is coded to have a different amount of pixels per sperm than the auto measure implementation to account for this but the results may be more off depending on the noise content in the image.
- None of the UI enhancement functions (zoom in, zoom out, save image) port well to windows exe's due to the cv2 library dependency being finicky with them. So for full UI functionality use the linux exe and/or compile and run dependencies locally. The keyboard functions all work but the UI is not as good on the windows exe due to windows exe limitations. (This is not really an error as much as it is a warning on the window exe's limitations).

Future Development Considerations (Future Work)

- Improve the front end and maybe port it onto react for better ease of use. Maybe use a different UI library instead of cv2 so that the windows exe files can have the same augmented features as the linux executable file.
- Highlight multiple components at the same time with a drag and drop box for all components obtained in automeasure mode (This feature already exists in Threshold mode)
- Make it so you can measure multiple different sperm cells in automeasurement mode at the same time like how it is already coded in threshold mode by letting the user highlight different components in different colors and make sums for those colors like in threshold mode. Right now it sums all the selected components but there is no way to make different sums of different sperm components count as multiple entries at the same time without measuring them one by one in automeasure mode. (This is already a feature in threshold mode but it is not in automeasurement mode.)
- Finetune manual measuring threshold mode and figure out how to reduce the noise impact and improve accuracy in the medium and harder cases. (Depending on the sample and the noise, the values may be off to a more significant degree)