

Background

Biological cells have specific dielectric properties that can be measured to provide information about their physiological state. For example, the dielectric properties of a cell can be used to determine if it is healthy or diseased, in a particular growth phase, or to determine its response when placed under stress conditions. Through utilization of microfluidic techniques and particle tracking processes, such dielectric properties may be measured and utilized in various applications.

Theory

Microfluidic Channel

- Cells will flow through the microfabricated fluid channel one-by-one and pass over an electrode array which is located at the bottom of the channel.
- The electrodes will produce a non-uniform electric field which will affect the cell's positional height and velocity in the microfluidic channel.
- By measuring the change in height and velocity, we can map that cell's dielectrophoresis response and get its dielectric properties.

High Speed Imaging

- A high speed imaging setup can be used to visually determine the flowing cell's change in velocity.
- Utilization of particle tracking processes enables us to find and track the cells as they pass through the field of view of the camera and over the electrodes.

Objective

A high-speed imaging system employing particle tracking processing can be used to determine the change in velocity of individual cells as they pass over a non-uniform electric field in a microfluidic chamber. Such a particle tracking program would require appropriate image processing, feature detection, and trajectory linking.

Methods

Design

- Employ open-source image processing and particle tracking routines to construct a custom program for the microfluidic system.
- Output information of particle's velocity, size, and shape(eccentricity) into a useable manner for experimental purposes.

Implementation

- This program was implement in Python, as Python is a modern programming language that is highly portable.
- Image processing was accomplished through a mixture of background subtraction and morphological processing for noise reduction. These techniques used the open-source computer vision library, OpenCV-Python.
- Feature finding and trajectory linking was applied to the processed frames. These techniques used the open-source particle tracking library, Trackpy.

Methods

Background Subtraction

- Background subtraction is accomplished through the help of OpenCV.
- Each frame appends to the background model.
- The current frame is subtracted from this model to yield a foreground mask.
- This mask is then applied to the current frame, focusing on objects of interest.
- Morphological processing is then applied to reduce noise.

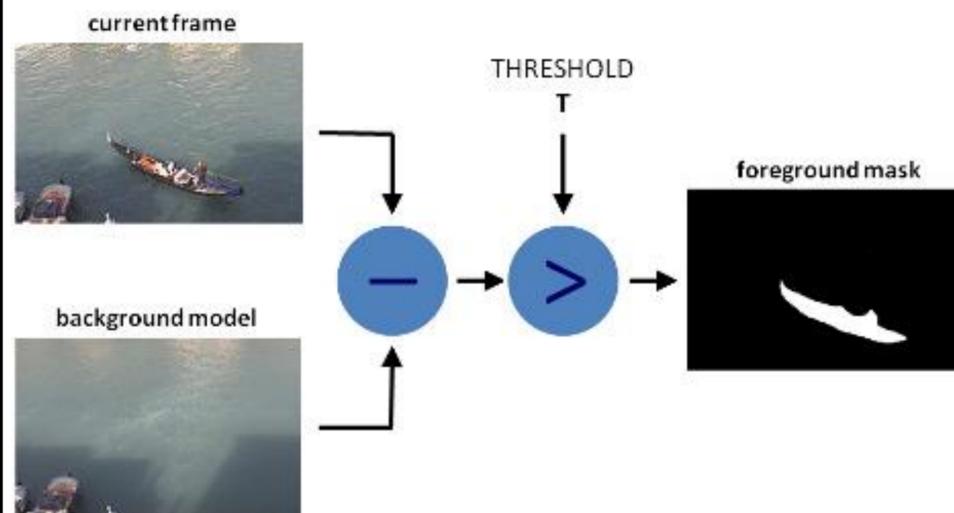


Fig. 1. Background subtraction visualization. Adapted from [1]

Feature Detection

- Feature detection is done with the help of Trackpy.
- The algorithm looks for features, given an estimated pixel size. It returns each feature's position, and other characteristics.
- This is done for each individual frame. This processes is sped up through parallelization.

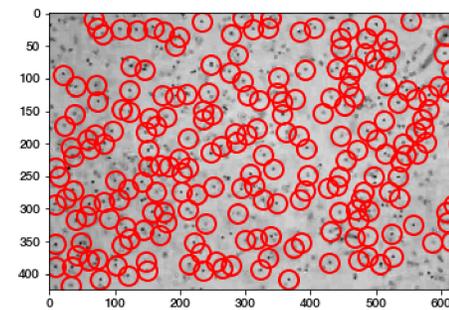


Fig. 2. Feature detection example photo. Adapted from [2]

Trajectory Construction

- Similarly, trajectory construction is done through Trackpy.
- Particles are linked from frame to frame.
- Each tracked particle is given a unique number.
- Trajectory data, along with other particle characteristics are output to a file.

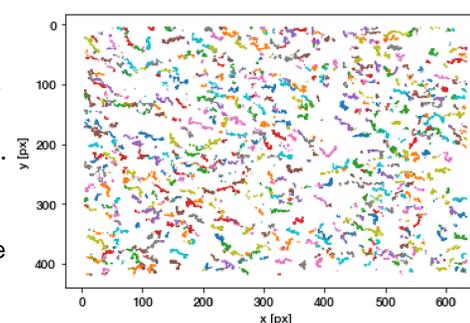


Fig. 3. Trajectory linking example plot. Adapted from [2]

Discussion

Testing

- Video used to conduct testing was of a simulated particle travelling across the field of view of a microfluidic chamber.
- Below is the first four data lines of the output file for the conducted test.

Output Data

y	x	Mass	Size	Ecc	Frame #	Particle #
453.1010	192.3604	150169.1	13.6326	0.01799	0	0
453.0654	203.1088	149992.3	13.6250	0.01985	1	0
453.0898	214.3713	150095.0	13.6316	0.01800	2	0
453.1253	226.1448	149677.4	13.6196	0.02077	3	0

- The first two columns give the centroid position of the particle, in relation to the dimensionality of the input video.
- The mass is the total integrated brightness of the found feature.
- The size is a representation of that particle's radius of gyration.
- The ecc is the eccentricity of the particle.

Applications of Data

- Velocity of each particle can be found through the use of the frame counter and positional data.
- Mapping the change in velocity to the change in height leads to the particle's dielectric response.

Future Work

- Improve performance, through parallelization and optimizations.
- Conduct velocity and other calculations in the program.
- Integrate with hardware setup to execute program on real data.
- Implement real-time and streaming features.

Acknowledgements

I would like to thanks Dr. Gregory Bridges, James Dietrich (Engineer in Residence), Behnam Arzhang (PhD Student), and Justyna Lee (Undergraduate Research Assistant) for their help on the project.

References

- [1] "How to use background subtraction methods," OpenCV. [Online]. Available: https://docs.opencv.org/4.5.3/d1/dc5/tutorial_background_subtraction.html. [Accessed: 13-Oct-2021].
- [2] "Trackpy: Fast, flexible particle-tracking toolkit," Trackpy: Fast, Flexible Particle-Tracking Toolkit - trackpy 0.5.0 documentation. [Online]. Available: <http://soft-matter.github.io/trackpy/v0.5.0/index.html>. [Accessed: 13-Oct-2021].