Computer Vision Algorithms for Analyzing Drying Colloidal Suspensions

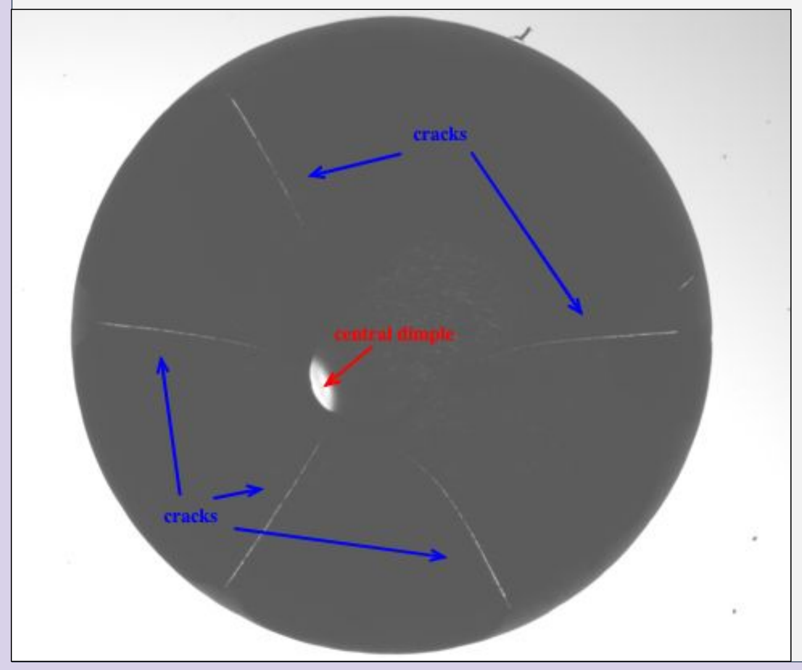
Max Paik | Driscoll Lab

Northwestern University

Northwestern Office of Undergraduate Research

Background

- Colloidal droplets, which are composed of small (nanometer to micron scale) particles suspended in a solution, are found everywhere from paints to commercial printing products.
- Colloidal suspensions interesting interesting, dynamic behavior as they dry, including the formation of cracks and dimples. This behavior can be investigated by capturing images under a microscope.
- The number, size, relative angles, and shapes of these patterns can vary greatly across different volume fractions, particle sizes, and drying conditions.



Left: Example image of a dried colloidal droplet.

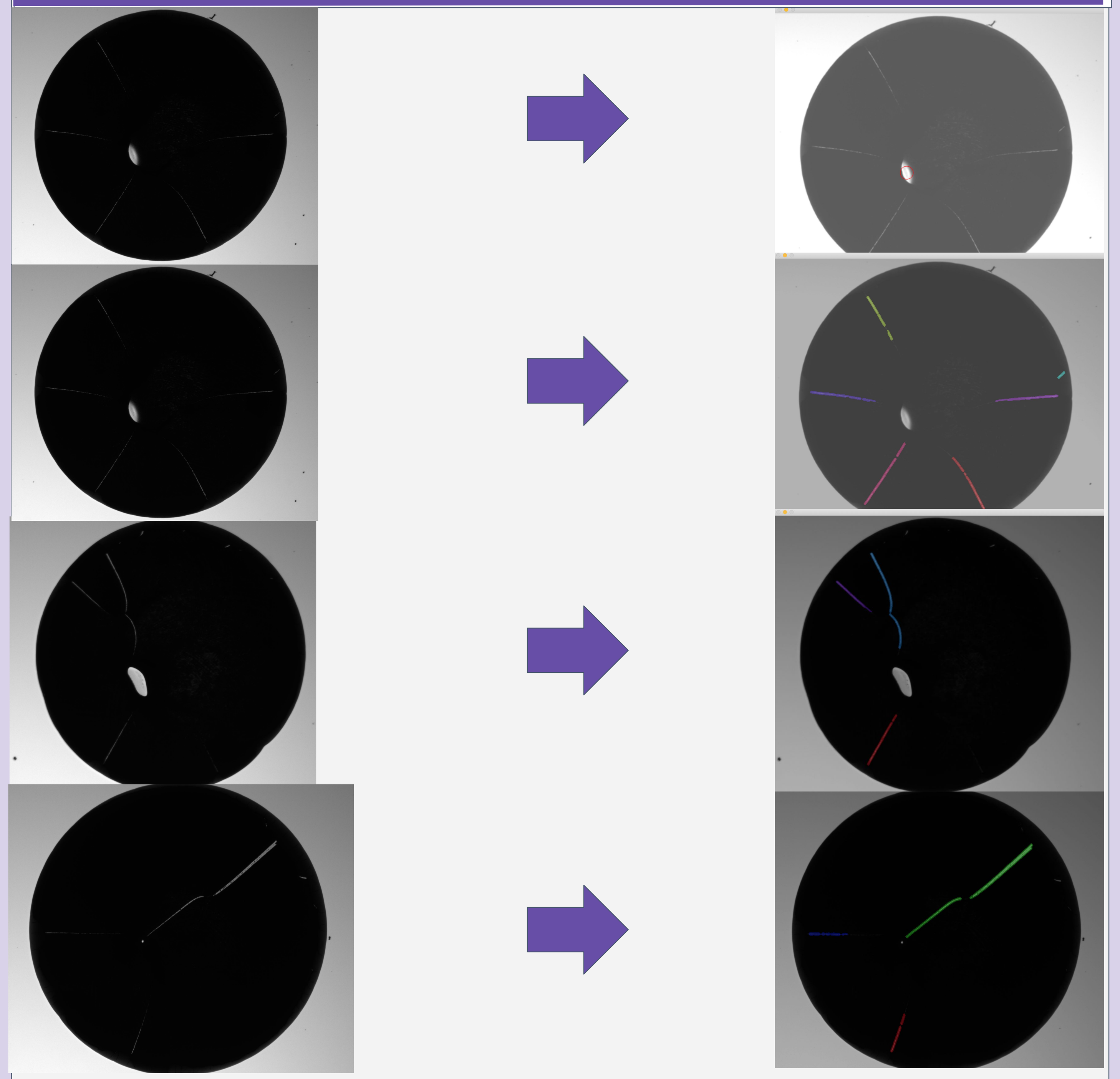
Objectives

- Originally, the goal of this project was to gain an understanding of the fundamental phenomena underlying drop drying by taking images of drying droplets at different volume fractions.
- Due to COVID, the project's goal shifted towards writing code for an image processing tool capable of describing and analyzing the patterns that form in these suspensions. More specifically, the goal of this project was to develop a codebase capable of accurately recognizing and describing the dimples and cracks present in an image of a drying colloidal suspension.
- Since eventually Driscoll Lab intends to capture hundreds of movies of these trials (each of which may have thousands of individual frames), automating analysis will be a great help in the future of this research.

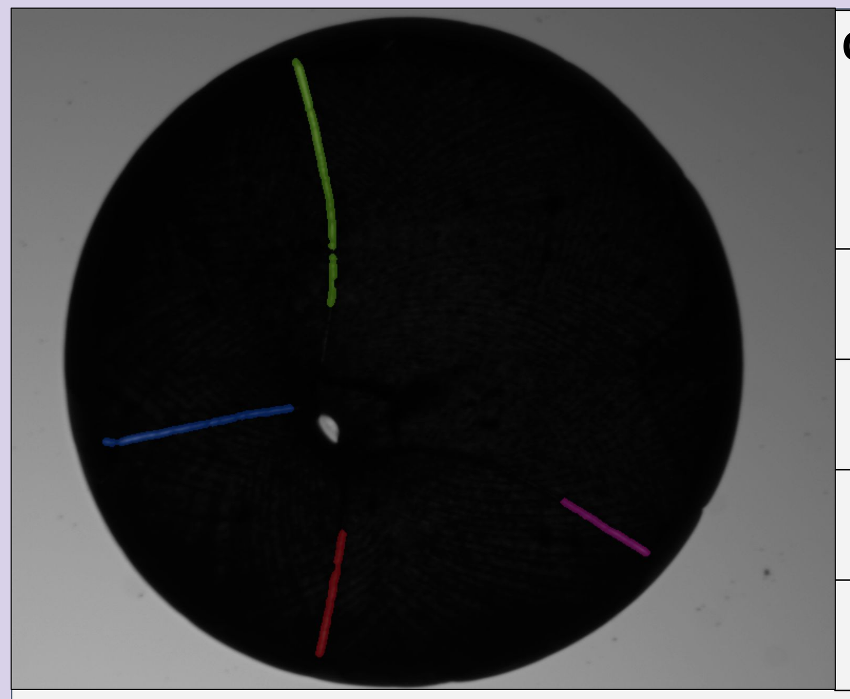
Approach and Algorithms

- The code includes several preprocessing steps, including blurring, edge detection, dilation, and detecting the entire drop.
- The dimple finding algorithm searches for blobs of bright pixels.
- On the other hand, the crack finding algorithm includes many steps. First, the code finds every connected component of pixels and loads them into a graph. The core of the algorithm is a filtering function that runs a breadth first search on each component, keeping track of the direction and length of a crack as it goes. This step is where cracks are separated from other image artifacts.

Example Images

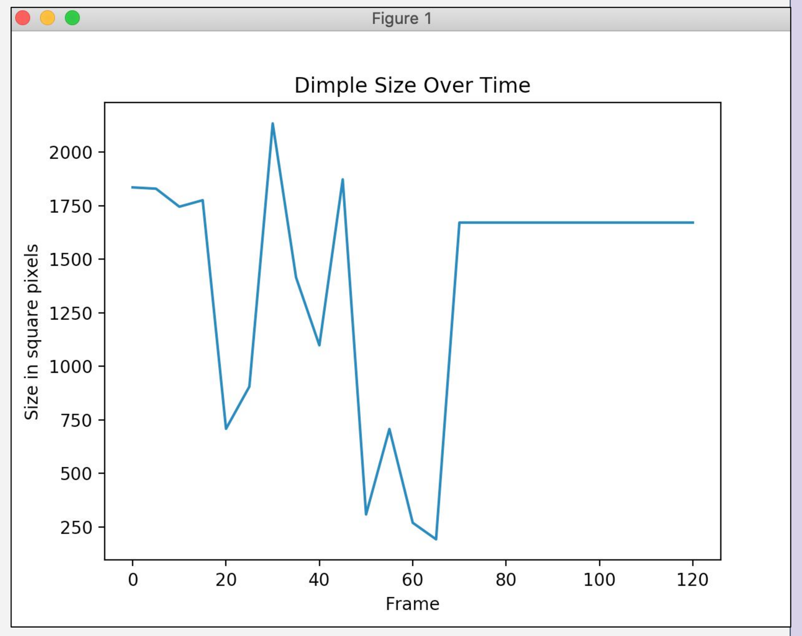


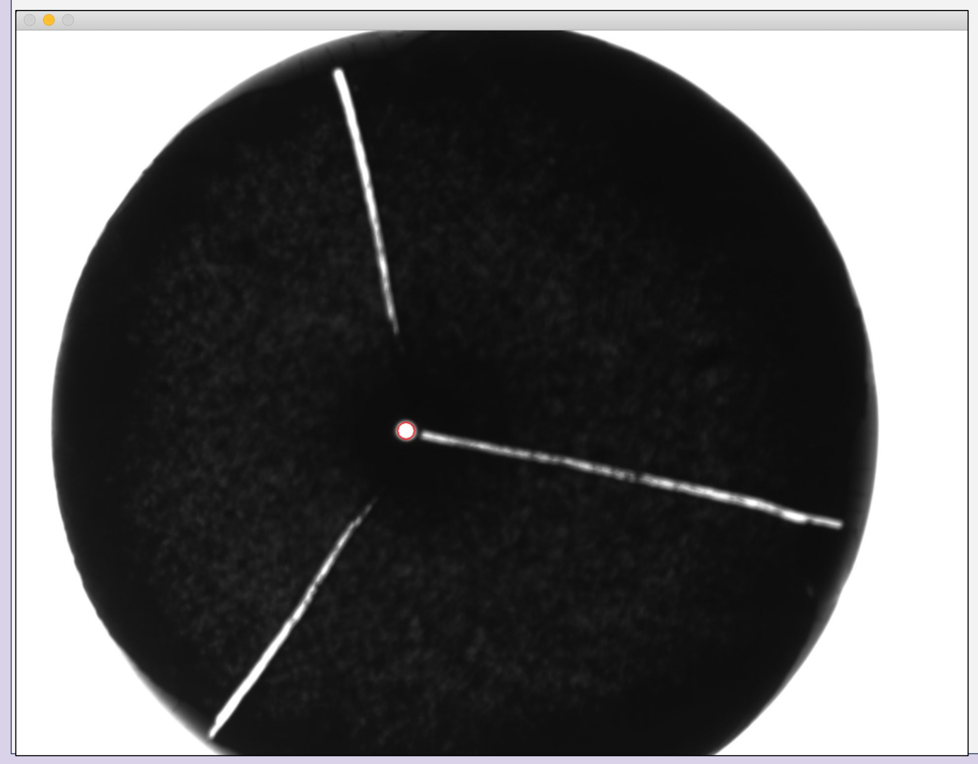
Images of the cracks and dimples found by the code. The red circle in the first image represents an equivalent circle to the dimple. Colored lines in other images represent cracks as detected by the code.

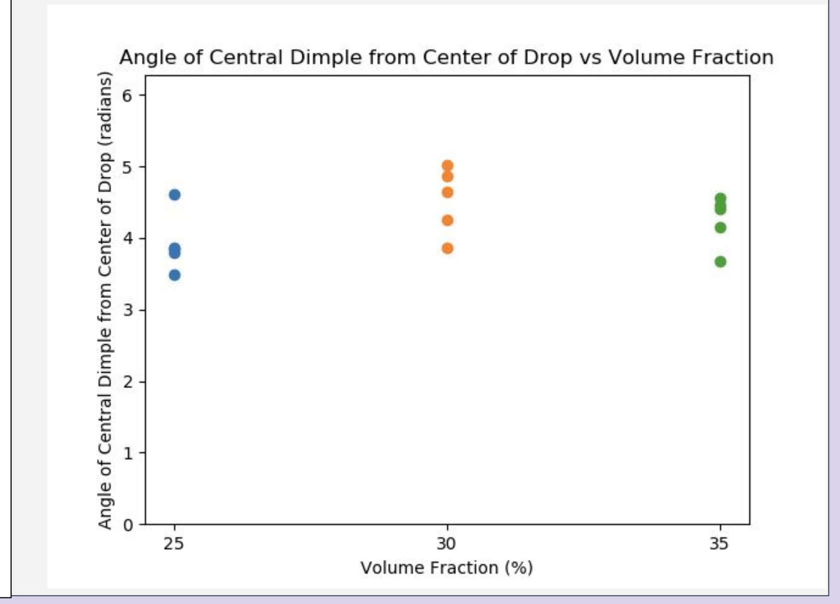


Crack	Code Angle (degrees)	ImageJ Angle (degrees)	Code Length (px)	ImageJ Length (px)
green	101.42	97.54	755.61	833.62
blue	190.22	193.59	528.14	553.64
red	258.98	261.41	386.21	362.52
pink	330.02	329.8	294.18	431.68

Above: Annotated results of crack finding algorithm. The code is very accurate for all cracks except the pink one, which becomes very thin.
Below: Results of dimple finding algorithm for each frame of a single movie (upper plot) and over all example trials (lower plot).







Results, Conclusions, and Future Work

- The image processing tool is capable of accurately finding and describing cracks of various geometries, though there are certainly edge cases for which the code struggles.
- The most exciting next steps in this project are to actually take images of colloidal suspensions across different volume fractions and analyze their drying behavior. This will begin with fixing the tilt in the microscope apparatus.

Acknowledgments

The study resulting in this presentation was assisted by a grant administered by Northwestern University's Office of Undergraduate Research. However, the conclusions, opinions, and other statements in this presentation are the author's and not necessarily those of the sponsoring institution. The author would also like to thank the Driscoll Lab, especially Professor Driscoll and Srishti Arora, for their support through all stages of this project.