

Extracting the Coordinates of the Center of a Drop from an Image

Kulwinder Kaur (2021PHS7190), Harikesh Kushwaha (2021PHS7181)

A Brief Overview

The goal of the article is to give a brief overview of the steps involved in extracting information about the center of a liquid drop from an image. The steps are as follows:

- 0. Reading the Image:** Reads the image using `matplotlib` to convert them into `numpy` arrays.
- 1. Cropping the Image:** Only a small portion of image is relevant to our analysis. To this end, the second step involves identifying the *region of interest* and crop the image to include just that.
- 2. Subtracting from a Reference Image:** The image contains a lot of *noise* which can result in our method to fail while extracting the center of the drop. To overcome this, the third step involves subtracting the image in consideration with a *reference image* giving a "noise-less" image.
- 3. Thresholding the Image:** This step converts the subtracted *grayscale* image into a *binary* one where all the pixels are either 0 or 1.
- 4. Finding All Points on the "Circumference":** The step involves looping through both *rows* and *columns* of the image and finding the coordinates of all the *pixels* with value 1 (corresponding to the white pixels). These points form the circumference of the drop.
- 5. Fitting an Ellipse to the Points:** In the final step, a general ellipse is *fitted* to the points found in step 4. The method uses the `EllipseModel` class from the `skimage` library. The class takes the X and Y coordinates of all the points and fits an ellipse using the *least square* method. After fitting, the model returns the *x* and *y* coordinates of the center of the ellipse, the *semi major* and *semi minor* axes and the angle θ between major axis and x axis.

Please see the images below for a visual representation of the steps.

These steps are then repeated for all the images.

Steps To Extract Coordinates Of The Center Of An Image

