COLLOIDAL PARTICLE SIZE ANALYZER

Submitted By

- Soumyajyoti Dey (20045162)
- Vaibhav (20045158)
- Ankush Jaiswal (20045019)
- Amar Singh (20045155)

INTRODUCTION

Knowing the size distribution of colloidal particles in an emulsion is important because it can greatly affect the physical and chemical properties of the solution. The size of colloidal particles can influence their ability to adsorb onto surfaces, which is important for applications such as catalysis.

For example, TiO_2 is a widely used catalyst in various applications due to its high surface area and high stability. The size distribution of colloidal particles of TiO_2 can affect these properties in several ways:

- **Surface area** - Smaller size of particles indicate a higher surface area to volume ratio which in turn increases the number of active sites available for the reaction to take place.
- **Stability** - The size distribution of TiO_2 particles can also affect the stability of the catalyst. Smaller particles tend to be more unstable and prone to aggregation, while larger particles tend to have a longer lifespan.

OBJECTIVE

Currently, researchers commonly use a software called ImageJ for analyzing the size distribution of colloidal particles.

Limitations of ImageJ

- **Inability to analyze multiple samples simultaneously**: Cannot be used to analyze multiple samples simultaneously, which can be time-consuming for large datasets.
- **Limited accuracy:** ImageJ relies on manually drawing regions of interest around each particle in an image, which can introduce human errors.

Through this project, we plan to automate this process in order to reduce manual effort and chances of human error by creating a web application where the user will be able to upload images like this and get the size distribution of colloidal particles in that image automatically.

INITIAL APPROACH

It was decided to use the OpenCV library for detection of circles in images. It contains a function called HoughCircles transform which is widely used for this purpose.

Important Parameters of HoughCircles function

- **minDist -** Minimum distance between the center (x, y) coordinates of detected circles.
- **minRadius and maxRadius -** The radius range of detected circles (in pixels).
- **param1 -** This parameter is the gradient value used to handle edge detection.
- **param2 -** This parameter is the threshold value for detection of centers of circles in the images.

LIMITATIONS OF INITIAL APPROACH

- Manual customization of parameters (param1 and param2) are required for every image to get accurate results.
- Only circles having a particular range of radii (minRadius, maxRadius) can be detected.

DEEP LEARNING APPROACH

In this approach, it was decided to train a deep learning model using the YOLO (You Only Look Once) Algorithm.

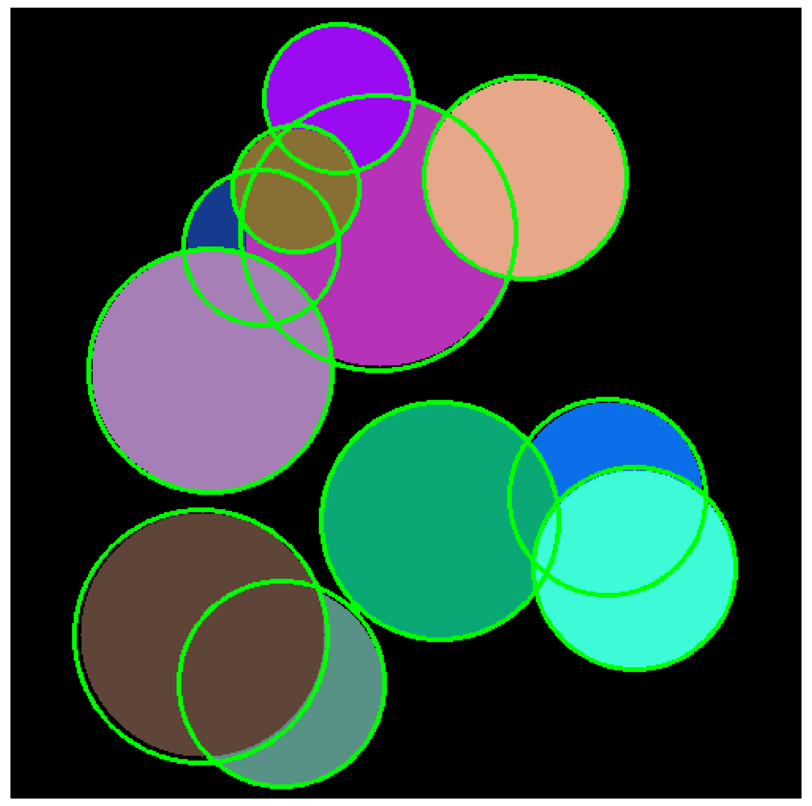
YOLO is currently the most widely used algorithm for object detection because it is lightweight and faster than the other algorithms. It requries a labelled dataset of images containing objects we want to detect. YOLO uses these labels as "ground truth" for training. A typical dataset should contain atleast 1000 images. In this case, the objects are circles.

ACKNOWLEDGEMENT

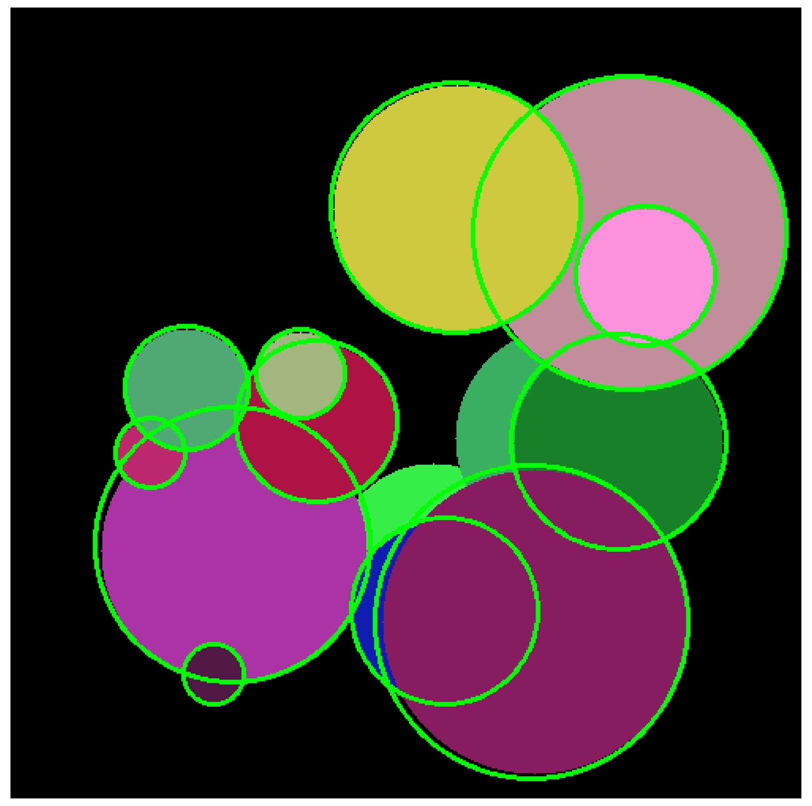
We want to express our deep and sincere gratitude to our supervisor Dr. Nitai Chandra Maji, Professor Department of Chemical Engineering and Technology who gave us the golden opportunity to work on this wonderful project and who took a keen interest in our project work and guided us along till the completion of our project work. We came to know about so many new things and we are very thankful to them. This project would not be effective without the efforts and co-operation from the team members.

RESULTS AND DISCUSSION

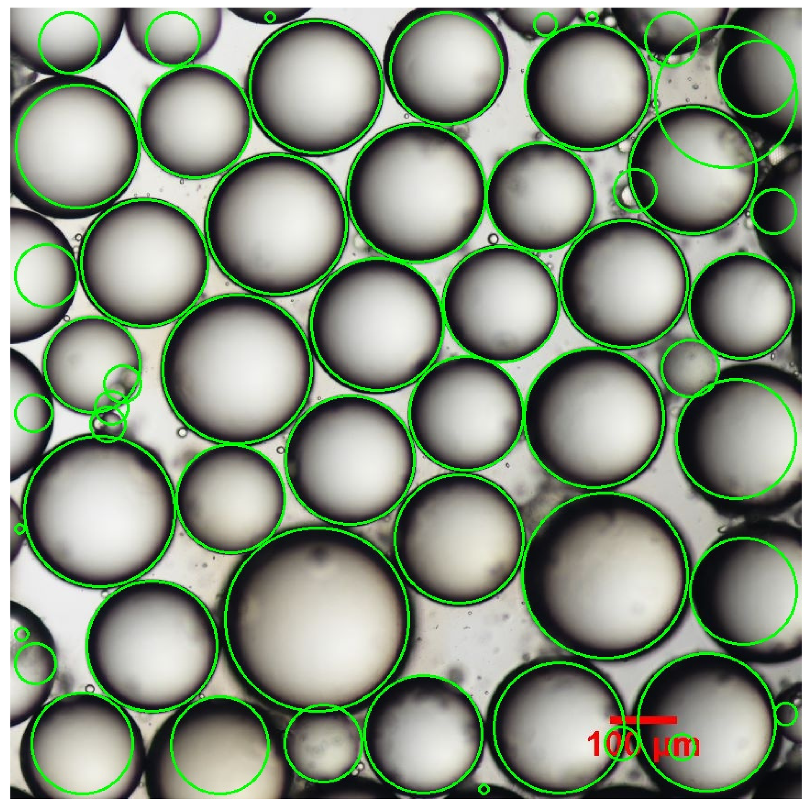
The model was trained using 1000+ images containing both transparent and solid circles. Following are some of the predictions made by the model after training (green borders around the circles signify that the circle has been detected):



In this case, all the circles got detected by the model.



In this case, most of the circles got detected. However, two circles (green and dark green) which are barely visible got missed out.



This is a real microscopic image of an emulsion on which we were supposed to detect circles. Here, the model was able to detect most of the bigger sized circles. However, there are some false detections too. Also, the circles which are not fully visible (in the corners) got missed out.

CONCLUSION

The accuracy of the model came out to be about 94% for simple images as shown in the above section. The dataset did not include real microscopic images for training. Hence, the next step would be to collect more of these images and label them in order to retrain the model.

Name of the Supervisor –
Dr. Nitai Chandra Maji

Grade –
A / B / C

Signature of Supervisor –