Grain Size Analysis Using Machine Learning

Under the guidance of

Dr. Gangi Reddy Salla

Contribution

- Bogineni Krishna Satya Sai Ram — Visualization and pixel size calculation
- Chidipothu Rushendra Sai — Image.j execution and report making
- Nayudu Ravindranath Chowdary — Multi-stage processing and outline filtering
- Poola Harish. — Utilizing Gray Level Co-occurrence Matrix
- Kakarla Sashank Chowdary — Integration of image preprocessing

AGENDA

01 — Aim

02 — Abstract

03 — Introduction

04 — System Requrements

05 — Image Processing

06 — Boundaries Detection

07 — Mean Grain Size

08 — Applications

09 — Results & Discussion

AIM

d

This project aims to develop an advanced image processing system for precise microscopic particle characterization in grain images, utilizing innovative techniques such as precision cropping, sharpening kernels, 3D scatter plots, and GLCM-based texture analysis. The goal is to offer a robust methodology for detailed particle analysis and contribute valuable insights to microscopic image processing.

Abstract

This work introduces a sophisticated method for categorizing particles in microscope images, employing editing, multi-stage processing, and particle segmentation. It conducts detailed analysis of particle sizes, locations, and aspect ratios, utilizing 3D scatter maps and GLCM texture analysis. The technique offers a valuable framework for particle identification in various scientific fields.

Rushendra-AP21110010612

Introduction

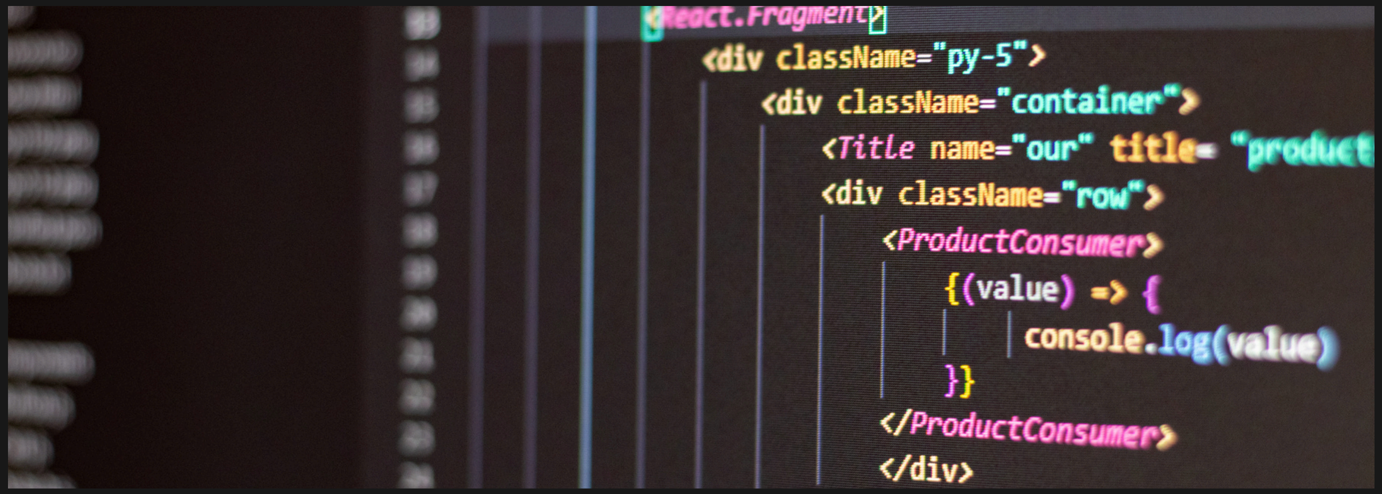
This project explores microscopic image analysis, employing innovative techniques like format conversion, precision cropping, and multi-stage processing to enhance particle visibility and segmentation. It utilizes multidisciplinary approaches, 3D scatter plots, and GLCM texture analysis to gain insights into particle composition, structure, and distribution, culminating in comprehensive findings and future research recommendations.

System Requirements

Hardware requirements:

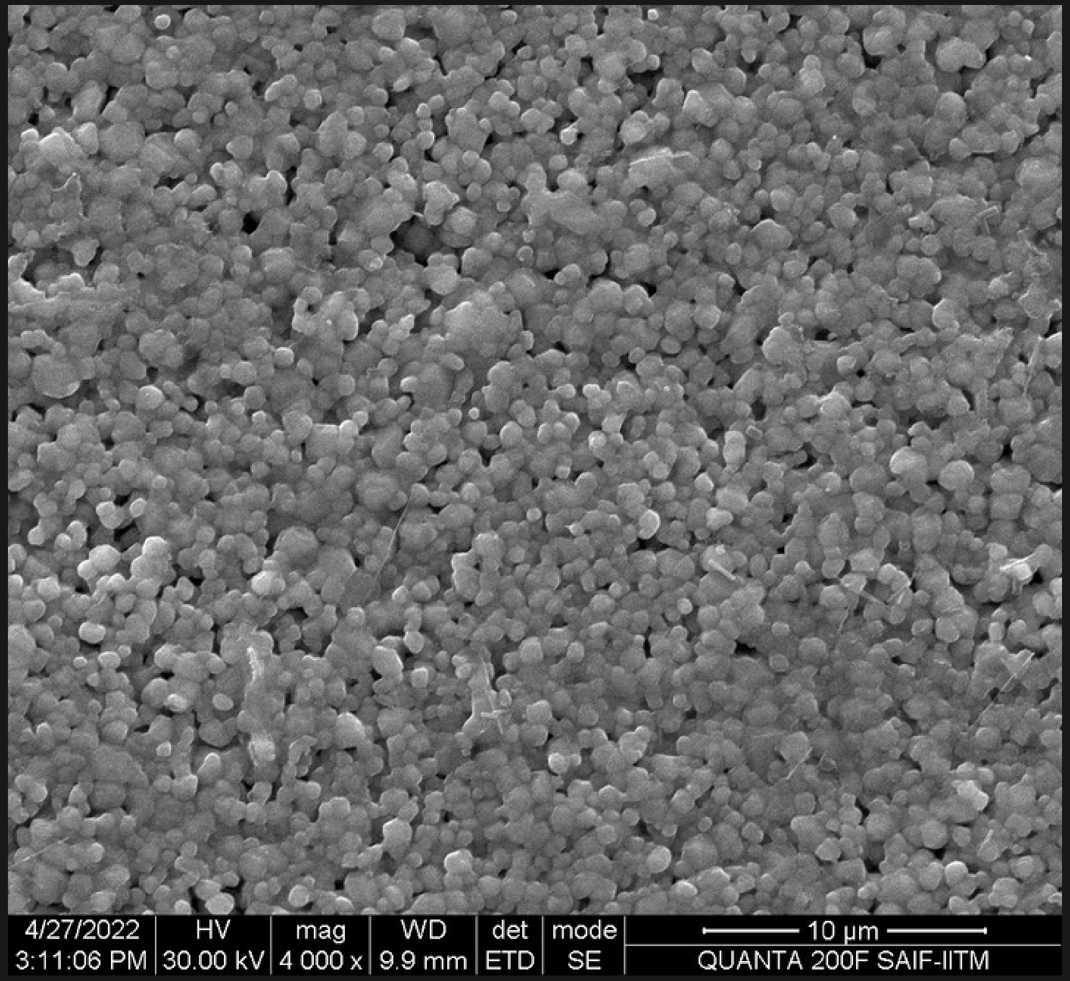
8GB RAM
256 or above ssd
Multicore Processor
Preinstalled OS(Windows, macOS, or Linux).

Software requirements:



An IDE such as Jupyter Notebook or Visual Studio Code is recommended
Python 3.x, along with data analysis libraries like Pandas, NumPy, Matplotlib, and Seaborn are necessary
windows XL or any other prefered apps for data sets
ImageJ.JS Tool for size detection anaysis

Sashank-AP21110010595



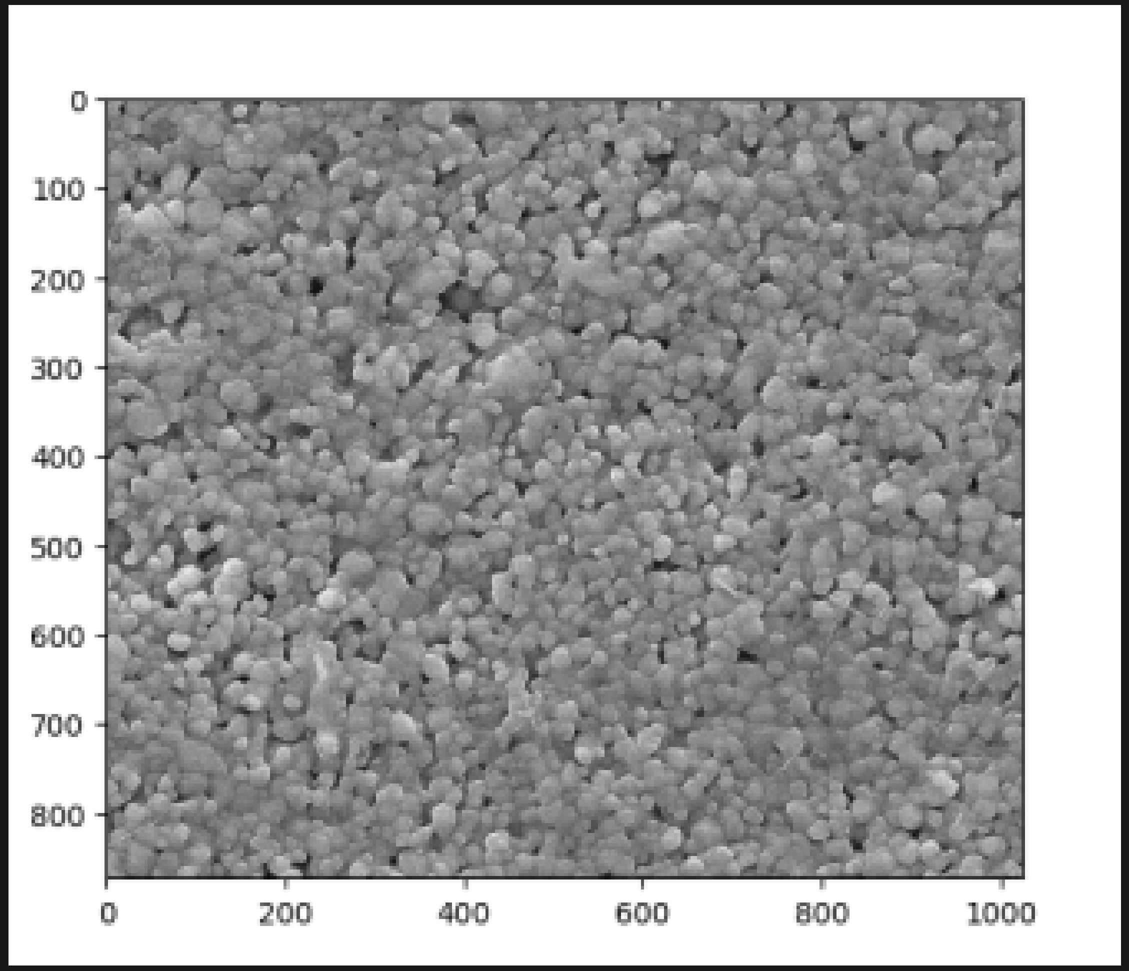


Image Processing

1. Load TIFF and Convert to JPEG:
   - Open a TIFF image, convert it to JPEG, and save the result.

2. Open, Crop, and Save JPEG Image:
   - Open a JPEG image, crop it to specified dimensions, and overwrite the original file.

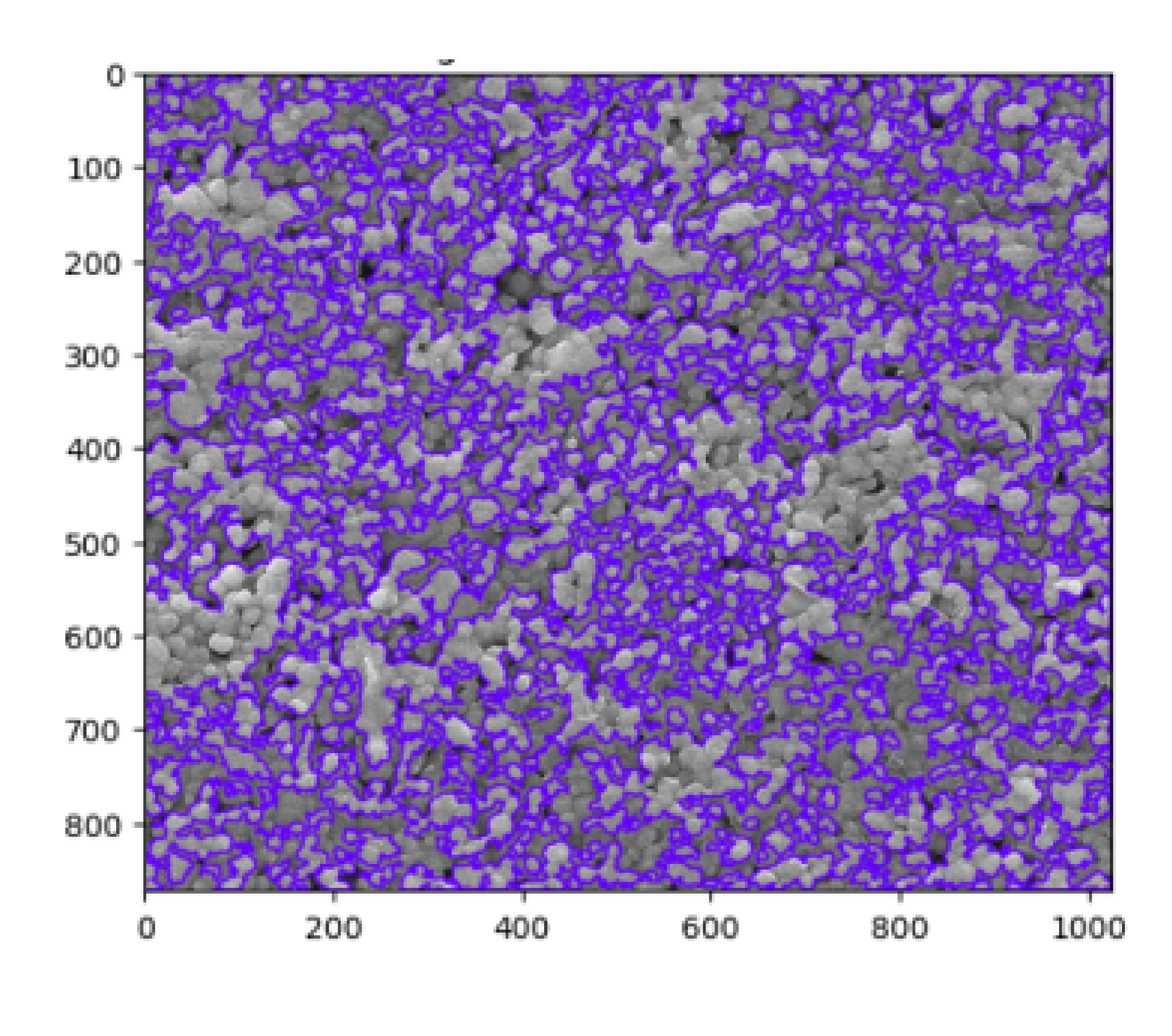
3. Adjust Brightness and Contrast:
   - Apply brightness and contrast adjustments to an image using OpenCV, and visualize the result.

4. Create and Apply Sharpening Kernel:
   - Define a sharpening kernel, apply it to an image using OpenCV, and display the sharpened result.

5. Show Images and Visualization:
   - Use Matplotlib to display the original adjusted image and the final sharpened image. Note the potential overwrite issue in saving the cropped image.

Sashank-AP21110010595

Boundaries Detection



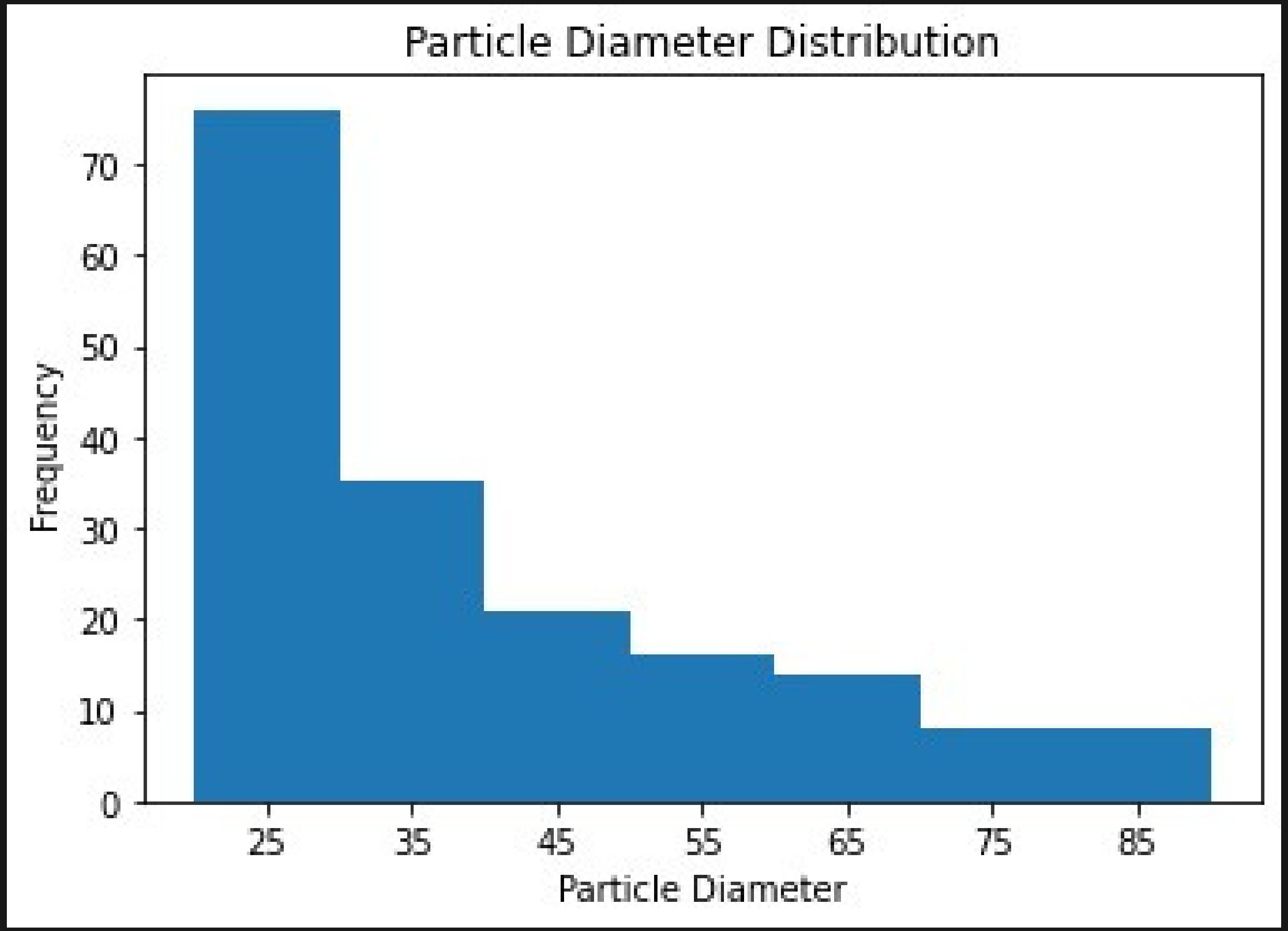
1. Grayscale Conversion and Gaussian Blur:
   - Convert the cropped image to grayscale and apply Gaussian blur for noise reduction.

2. Particle Segmentation with Thresholding:
   - Use thresholding to segment particles in the blurred grayscale image.

3. Contour Detection and Particle Diameter Calculation:
   - Find contours in the thresholded image, calculate particle diameters, and store them.

4. Draw Particle Boundaries and Display Result:
   - Draw particle boundaries on the original cropped image, calculate and print the average diameter, and display the modified image with Matplotlib.

Mean Grain Size



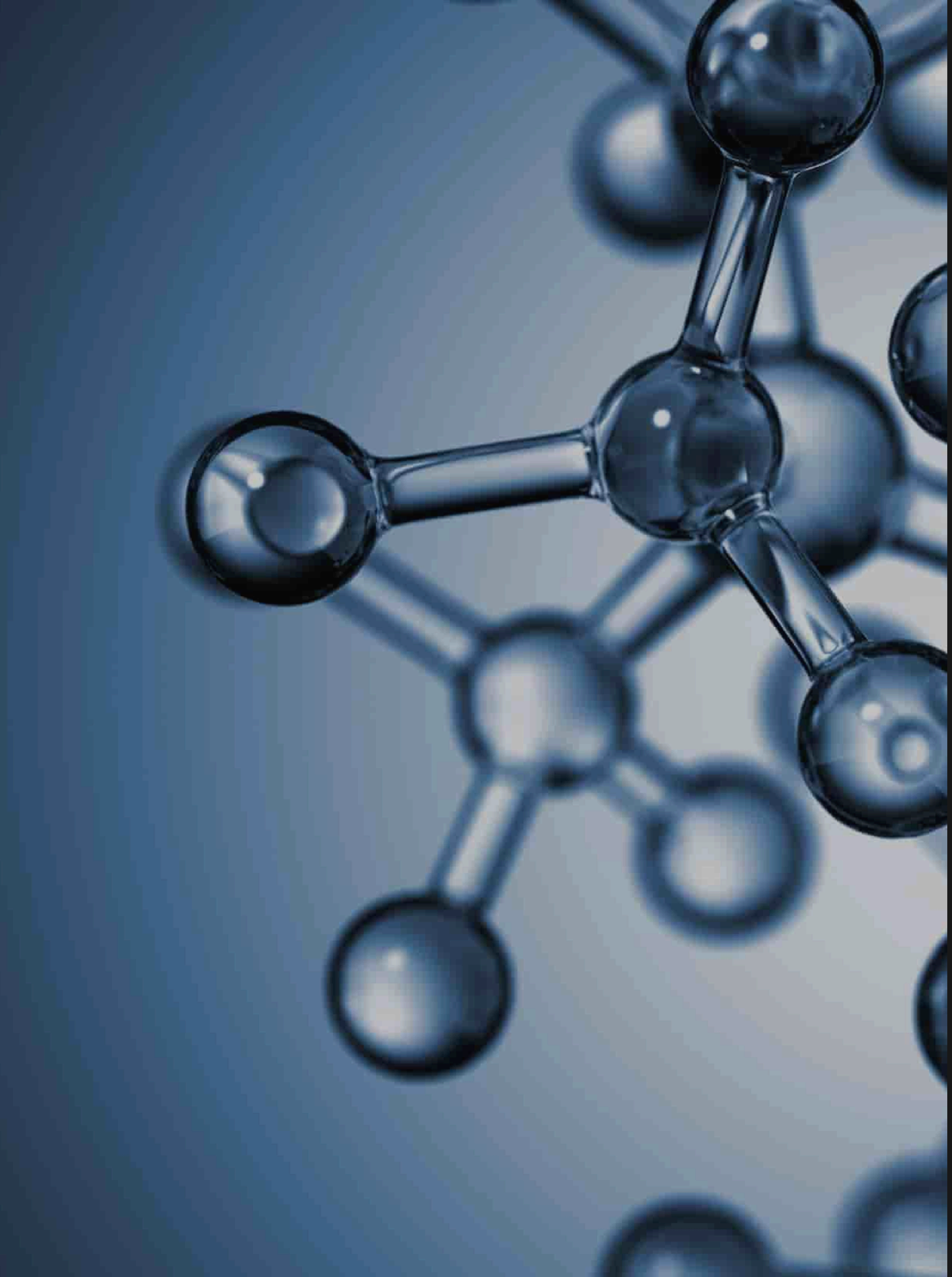
1. Round and Sort Particle Diameters:
   - Round each particle diameter to two decimal places and sort the list.

2. Remove Outliers and Inappropriate Diameters:
   - Remove diameters equal to 0.0 or outside the range [20, 90) from the sorted list.

3. Calculate Mean and Display Filtered Data:
   - Calculate the mean of the filtered particle diameters and print the result.

4. Visualize Particle Diameter Distribution:
   - Create a histogram using Matplotlib with defined bins andplot the frequency distribution of particle diameters.

Krishna-AP21110010619

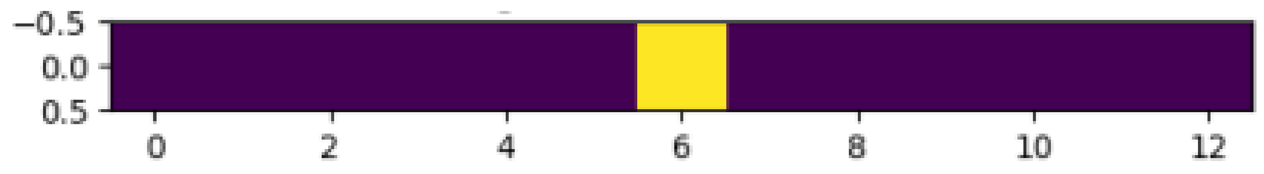


Applications

1. Materials science and engineering
2. Nanotechnology and Nano materials
3. Energy Storage Devices
4. Environmental Monitoring
5. Biomedical Research
6. Advanced Manufacturing
7. Geological and Earth Sciences
8. Pharmaceutical and Chemical Industries
9. Educational and Research Institutions

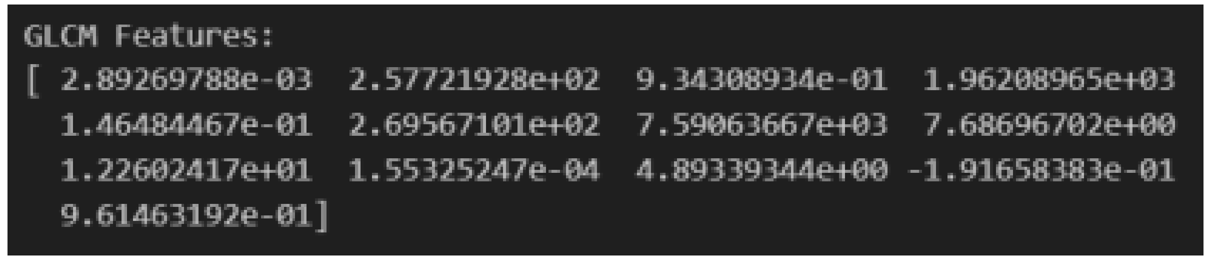
Krishna-AP21110010619

Results and Discussion



Clustered Regions Based On Texture Features

The recommended particle characterization approach improves visibility and segmentation, and enhances the visualization of size distribution patterns in a three-dimensional scatter plot.


```
GLCM Features:
[ 2.89269788e-03  2.57721928e+02  9.34308934e-01  1.96208965e+03
  1.46484467e-01  2.69567101e+02  7.59063667e+03  7.68696702e+00
  1.22602417e+01  1.55325247e-04  4.89339344e+00 -1.91658383e-01
  9.61463192e-01]
```

Output displaying GLCM Features

Texture analysis using GLCM features enhances the ability to differentiate between particles based on their textural properties and employs GLCM-based KMeans clustering to detect plausible heterogeneities in the image.

Results and Discussion

```
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 2), Non-Zero Pixels: 510
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 1), Non-Zero Pixels: 255
Error computing haralick features: mahotas.haralick_features: the input is empty. Cannot compute features!
This can happen if you are using `ignore_zeros`
Region Shape: (1, 2), Non-Zero Pixels: 510
...
Region Shape: (27, 68), Non-Zero Pixels: 121380
Region Shape: (25, 22), Non-Zero Pixels: 36210
Region Shape: (38, 22), Non-Zero Pixels: 45135
Region Shape: (7, 6), Non-Zero Pixels: 4335
```

Output displaying texture features for each contour region

Morphological image processing theories by Sciaini and Fritsch are utilized to identify contours on edge-detected images, with subsequent texture analysis enabling a deeper understanding of the spatial structure of the data within each contour area.

```
Mean Diameter of Particles: 59.917178957294134 (pixels)
Sorted Particle Diameters: [ 13.83990574  14.45306015  14.77899456  15.22924042  15.23174667
  15.44304466  15.52437496  15.81158829  15.85589886  15.87128448
  16.1590023   16.17395782  16.41689873  16.6496048   17.20485115
  17.26287651  17.26287651  17.27584839  17.49305534  17.49305534
  17.51092148  17.69200706  17.82909966  17.90940475  18.05908585
  18.50143242  18.78849411  19.03969193  19.10085678  19.10517311
  19.10517311  19.24655151  19.31678391  19.31863594  19.41668892
  19.41668892  19.52396011  19.64708328  19.65820694  19.83530045
  20.09995079  20.09995079  20.12481117  20.22394943  20.46699715
  20.59146118  20.59353828  20.81035423  21.0438633   21.09522247
  21.10778809  21.26049232  21.40113449  21.75422096  22.02291489
  22.04895782  22.10722351  22.12602043  22.13049889  22.20380402
  22.31461716  22.3608799   22.3608799   22.47240448  22.71713638
  22.80370903  23.08699226  23.4095993   23.47221184  23.60104752
  23.60104752  23.85347557  23.94628716  24.04183006  24.04183006
  24.20763779  24.35179138  24.51550102  24.69837761  24.9458065
  25.00020027  25.06576157  25.0800724   25.0800724   25.11984634
  26.00020027  26.03109932  26.0770092   26.20997047  26.40095711
  26.41988945  26.6835289   26.90744781  27.018713    27.20314217
  27.48758125  27.65883446  27.72887611  28.08558846  28.67417145
  28.84461021  28.84461021  28.85869217  29.06908417  29.1648922
  29.41108322  29.49799919  29.54677391  29.64186287  30.1721611
  30.46653748  30.86502075  31.14502335  31.24119949  31.30515289
  31.93328857  32.08691788  32.25271606  32.55783844  33.24173737
...
 168.52896118 186.74671936 191.9090271  196.65725708 201.10215759
 203.00323486 262.12438965 269.02993774 269.09628296 285.61709595
 326.55877686 334.13259888 335.94448853 343.65701294 376.38796997
 621.51477051]
```

Output displaying Mean Diameter Of Particles

Quantitative analysis utilizes a violin plot to visually represent particle diameters and mean diameter, complemented by modern technologies like edge detection and GLCM-based clustering to further enhance the inquiry.

Conclusion & Future work

1. Explore machine learning algorithms for automated particle categorization to enhance study efficiency.

2. Extend the system to handle three-dimensional data, enabling a more comprehensive exploration of particle interactions and structures.

3. Investigate state-of-the-art classification approaches for improved efficacy and scalability in particle identification and categorization.

Our integrated image processing and analysis approach successfully provides precise characterizations of microscopic particles, showcasing efficacy in format conversion, cropping, and pipeline processing. The results, including 3D scatter plots and texture analysis, affirm the precision and clarity achieved in understanding particle characteristics.

Reference :

Rosebrock

A. Mishra and T. Pathak, “Estimation of grain size distribution of frictionstir welded
joint by using machine learning approach,” ADCAIJ: Advancesin Distributed
Computing and Artificial Intelligence Journal 10, 99–110(2021)

Ravindranath-AP21110010621


THANK
YOU