Empty Space Detection using Machine Learning

Akhilan Ashokan
Pradyumna Heddur Nagendra

Supervised by - Dr. Ing. Prof. Sebastian Houben

Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Introduction

Problem Statement:

- Detect and segment empty spaces in an image to identify potential areas for object placement.
- Achieve this using a clustering algorithm (K-Means) for pixel segmentation.

Why this is Important:

- Relevant for robotics and automation tasks like pick-and-place operations.
- Supports efficient space utilization and task planning.

Figure 1: Workstation* Image

* Workstation image is obtained from the dataset of B-it Bots @ work. There are images with varying background and objects.

Methodology

Step 1: Image Preprocessing

- Load the image.
- Crop to region of interest (ROI).
- Convert to RGB format.
- Reshape image to a 2D array of pixel data.

Step 2: K-Means Clustering

- Initialize centroids randomly.
- Assign pixels to the nearest centroid.
- Recalculate centroids iteratively until convergence.

Step 3: Visualization

- Compare original and segmented images.
- Visualize centroids in RGB space.
- Highlight individual clusters.
- Display results for varying “k” values
- Colormap of Distance Transform
- Post Processing Images to detect empty space

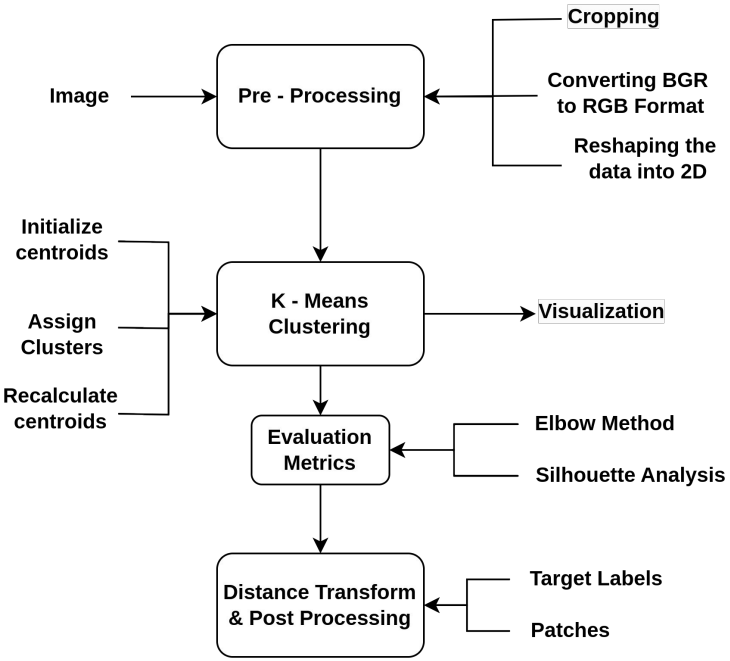


Figure 2: Outline of the methodology

Original vs Cropped Image

Figure 3: The Original Image

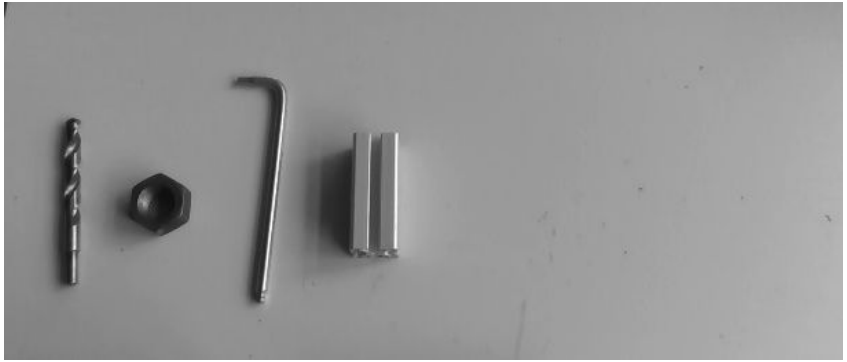
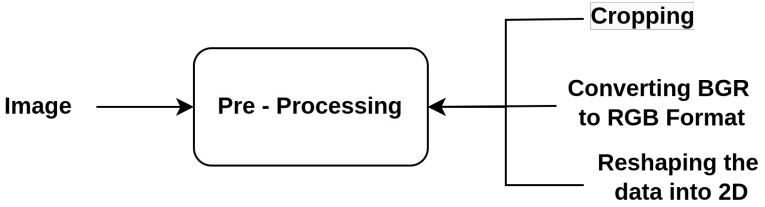


Figure 4: The Pre-Processed Image



2D and 3D Visualization of Cluster Centroids

Description:

- 3D scatter plot showing distribution of pixel colors (R, G, B channels).
- Cluster centroids highlighted to indicate grouped regions.
- The centroids are hid behind these clusters as it is a 3-D graph.

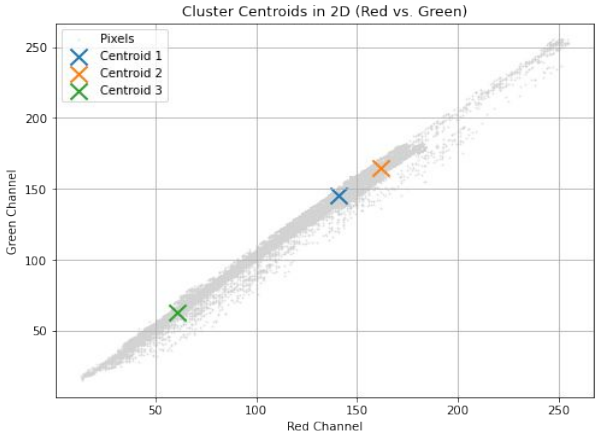


Figure 5: 2D Scatter Plot to Visualize the Clusters

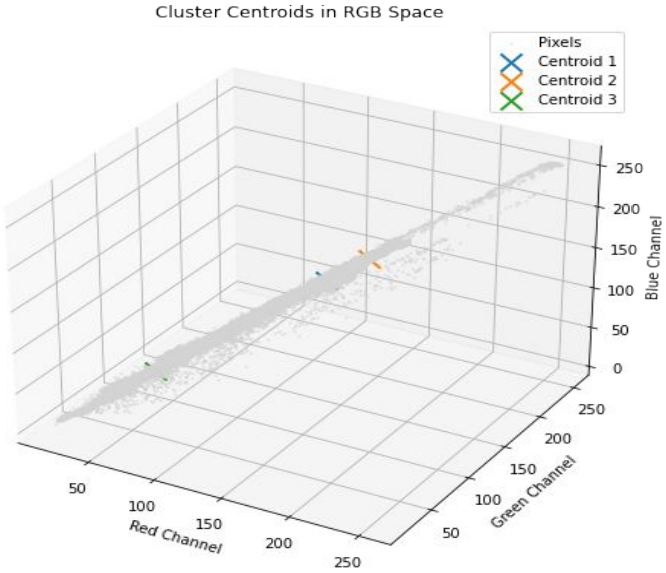


Figure 6: 3D Scatter Plot to Visualize the Clusters

Centroid Convergence

Description:

- Plot showing the movement of centroids over iterations only for Red Channel.
- Illustrates the optimization process of the K-Means algorithm.

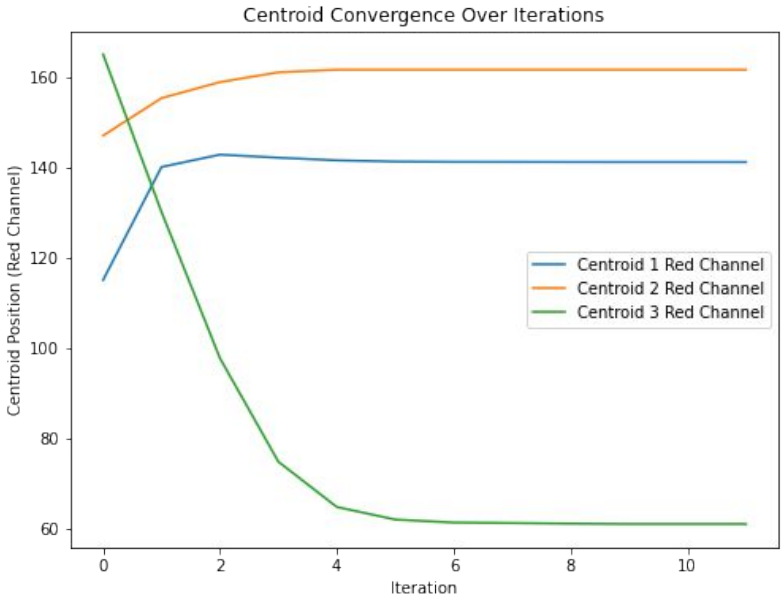


Figure 7: Iterative Centroid Convergence

Highlighted Clusters

Description:

- Each cluster is displayed individually to show the specific regions segmented.
- Helps in understanding the spatial separation of clusters.

Figure 8: Cluster Specific Images

Comparison for Different “K” Values

Description:

- Segmentation results for different cluster counts (K).
- Shows the effect of influence of number of clusters influencing the segmentation of the image.

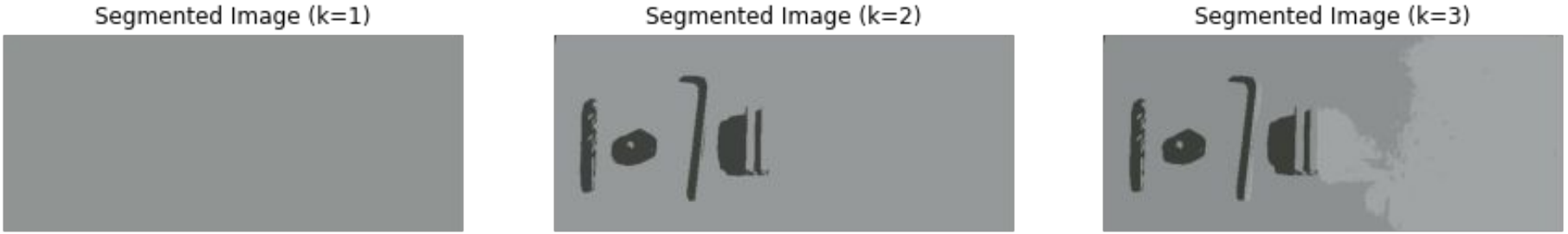


Figure 9: Effect of Image Segmentation based on Different ‘K’ Values

Evaluation Metrics - I

Description:

- Elbow Method
  a. By calculating the WCSS (Within cluster Sum of Squares).

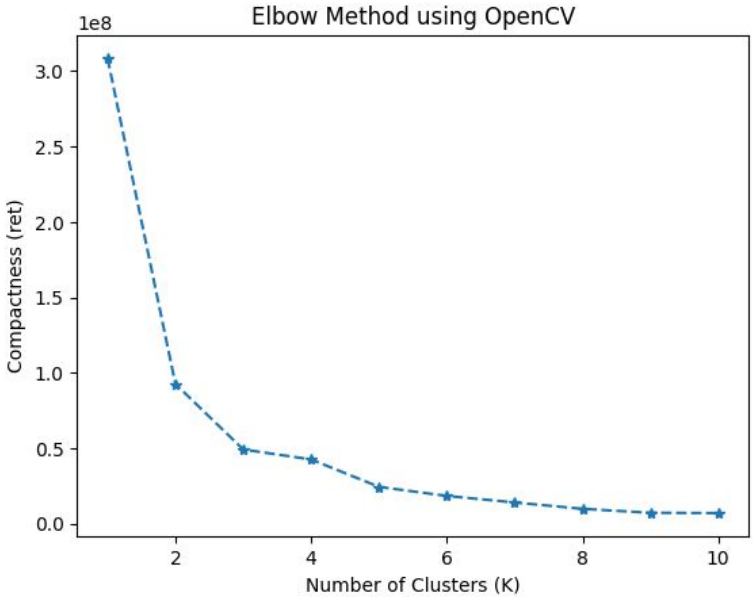


Figure 10: Elbow Analysis to find the Optimal 'K' Value.

Evaluation Metrics - II

Description:

- Silhouette Analysis
  a. The thickness of each clusters directly corresponds to size of the cluster
  b. Width of the cluster represents the sorted silhouette coefficients.

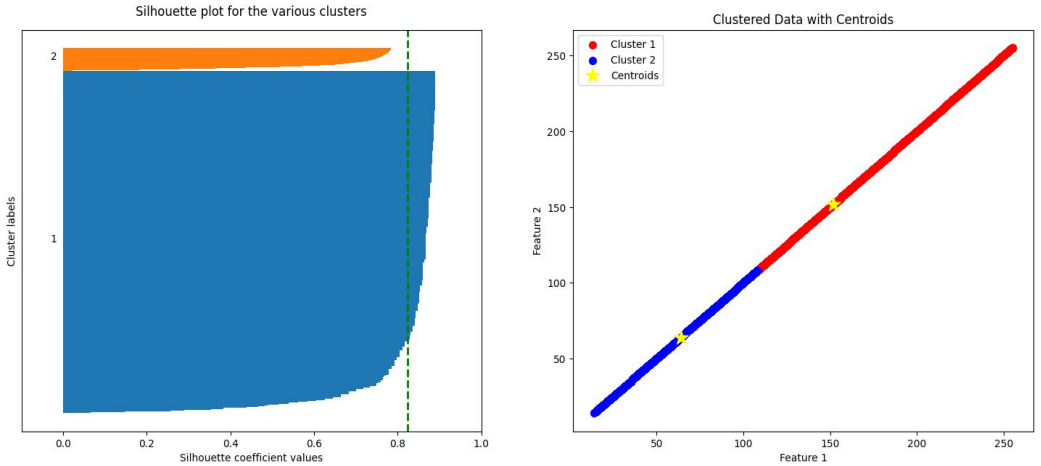


Figure 11: (a) Silhouette Analysis Resulting a Coefficient of 0.82
(b) 2D Visualization of Clusters and its Centroids

Distance Transform - I

Description:

- From the obtained labels from the K-Means.
- We compute euclidean distance between the target label and other labels.
- Results in the colormap as shown in the image.

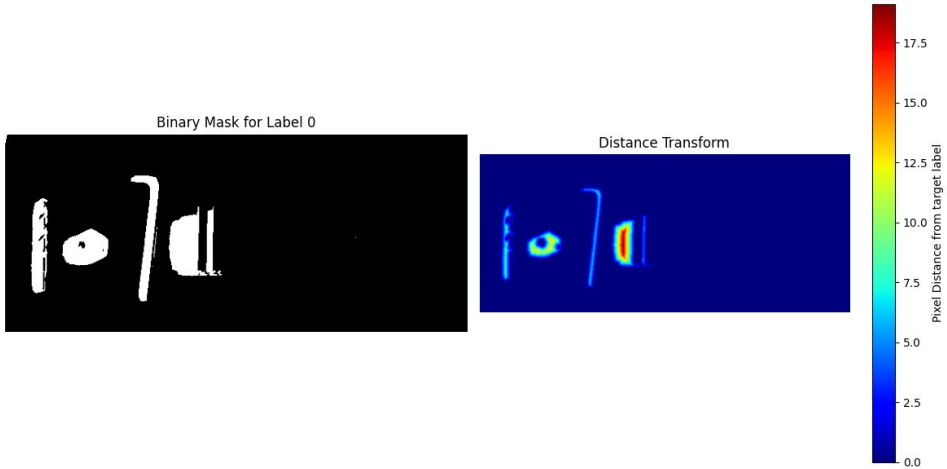


Figure 12: (a) Binary Mask with Respect to Target Label '0'. (b) Colormap of Distance Transform that Differentiates the Empty Space from others.

Distance Transform - II

Description:

- Patches are created from the above distance transform images.
- From these patches, we check for the column value that is absolutely the same.
- Then the value of the calculated distance transform is manipulated to differentiate the empty space.

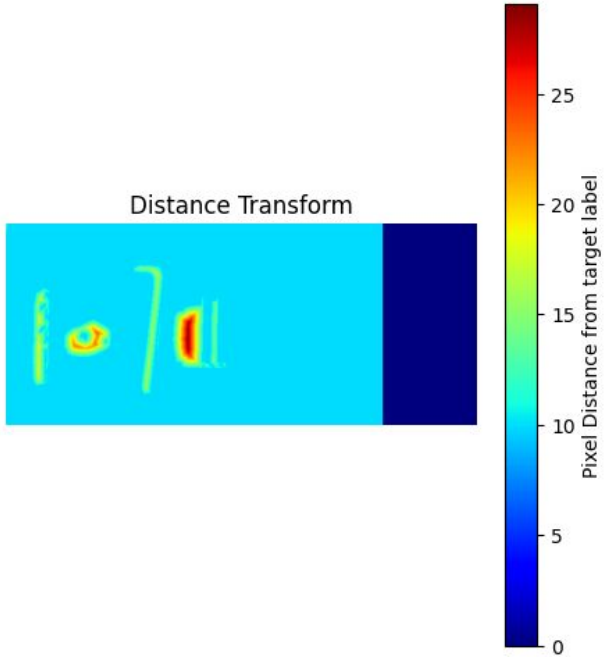


Figure 13: Distance Transform Images with Respect to Patches

Github link :
https://github.com/Akhilan-xd/empty_-_detection_ML

Thank You