Bacterial image segmentation using unsupervised learning

Ofir Herrera-318971694

April 2025

git clone link : https://github.com/Ofir-Herrera/UnsupervisedLearning.git

Abstract

This project addresses the challenge of segmenting E. coli bacterial images in the absence of labeled data, a critical step toward enabling quantitative analysis of bacterial localization. Leveraging handcrafted feature extraction and unsupervised KMeans clustering, we successfully distinguished dense bacterial colonies from complex backgrounds in high-resolution microscopy images. The resulting segmentation provide a foundation for future statistical studies of bacterial swarm dynamics.

1 Introduction

Understanding the spatial distribution of bacteria is a critical step in studying their behavior both as individual organisms and as collective swarms. Microscopy images of bacterial colonies often capture high-density regions where bacteria appear tightly packed, overlapping, and morphologically similar. This presents a significant challenge for automatic segmentation methods.

Traditional image processing techniques, such as thresholding, edge detection, or morphological filtering, often fail to accurately distinguish bacteria from the background in such high-density images. These methods struggle particularly in areas where bacteria blend into each other or into uneven backgrounds, resulting in either fragmented segmentation or merged clusters.

Furthermore, due to the lack of annotated ground truth data, supervised learning approaches cannot be applied directly. Manually labeling bacterial boundaries at the pixel level is time-consuming, error-prone, and requires expert biological knowledge.

Given these challenges, we explore an unsupervised learning approach to address the segmentation task. Specifically, we apply the KMeans clustering algorithm to handcrafted pixel-level features extracted from grayscale microscopy images. The goal is to determine whether this method can effectively separate bacterial regions from the background, without relying on labeled data.

The output of this segmentation process — binary masks highlighting the bacterial regions — will later serve as a foundation for building a supervised learning pipeline. By using the unsupervised masks as pseudo-labels, we aim to generate a training dataset that can be used to train deep models or classical supervised methods, thereby enhancing scalability and automation in the analysis of bacterial swarming behavior.

2 Methodology

2.1 Dataset

The dataset consists of grayscale microscopy images of gut bacteria, captured in high resolution. Each image is of size 1024×1024 pixels and contains bacterial colonies with varying densities — some images show highly crowded regions with significant overlap between bacteria, while others exhibit sparser distributions. This variability in bacterial density adds complexity to the segmentation task. The figures are shown in Figure 1a and Figure 5a.

2.2 Preprocessing

The bacterial microscopy images used in this study present several challenges: high density of colonies, overlapping structures, and low contrast in some regions. Therefore, careful feature engineering was applied to enhance the structural information in each image while suppressing noise.

Noise Assessment via Histogram Analysis. We analyzed the histogram of each image to evaluate whether there is a clear separation between pixel intensities corresponding to bacteria and background. The histogram did not show such separation, and irregular jumps in bar heights were observed, indicating a degree of noise in the images.

Gaussian Blur Difference. We applied two Gaussian blurs with different kernel sizes and computed their difference. This technique emphasizes mid-frequency variations by subtracting low-frequency components. In our images, this helped to reduce noise and distinguish textured bacterial regions from flat background areas, even when boundaries between colonies were soft.

Laplacian Filter. The Laplacian operator calculates the second-order derivative, accentuating sharp intensity transitions. In the context of our images, this highlighted internal edges between tightly packed bacteria, which were often missed by simpler gradient-based filters.

Canny Edge Detection. Canny was used to extract strong local edges. While it works best on high-contrast boundaries, it also helped highlight the outer contours of dense colonies. However, in regions with low signal-to-noise ratio, Canny alone was not sufficient — which justified the need for the additional filters above.

Median Filtering. Before extracting features, we applied a median filter to reduce salt-and-pepper noise while preserving edges. This was especially important in bacterial images where random bright or dark spots could be mistaken for structural features. The median filter preserved the clarity of cell boundaries while cleaning up isolated noise pixels.

Normalization. Each resulting feature image was normalized to the range [0, 1] to ensure that no feature would dominate the clustering due to scale. This also ensured that KMeans clustering treated all features with equal importance.

Overall, the combination of these features allowed us to capture both global and local characteristics of the bacterial patterns. This was clearly reflected in the visual results: bacterial regions were separated from background, and in many cases, sub-regions within colonies formed distinct clusters due to edge and texture variations.

2.3 Clustering with KMeans

Each pixel in the image was represented as a vector in a three-dimensional feature space, composed of values from the blur difference, Laplacian, and edge-detection filters. KMeans clustering was applied to group these pixel vectors into K distinct clusters.

KMeans was chosen because it is effective when the data in each cluster follows approximately Gaussian or convex distributions. In class, we learned that KMeans assumes that clusters are spherical and separated in Euclidean space — which aligns well with the nature of the extracted features. Each feature image was normalized, and the combined feature vectors tend to form blob-like groups in the feature space, especially when texture and intensity differences reflect underlying biological structures. Although the overall image is not globally Gaussian-distributed, the pixel features form localized structures that approximate Gaussian-like clusters, justifying the use of KMeans in this context.

2.4 Statistical Validation of K Selection

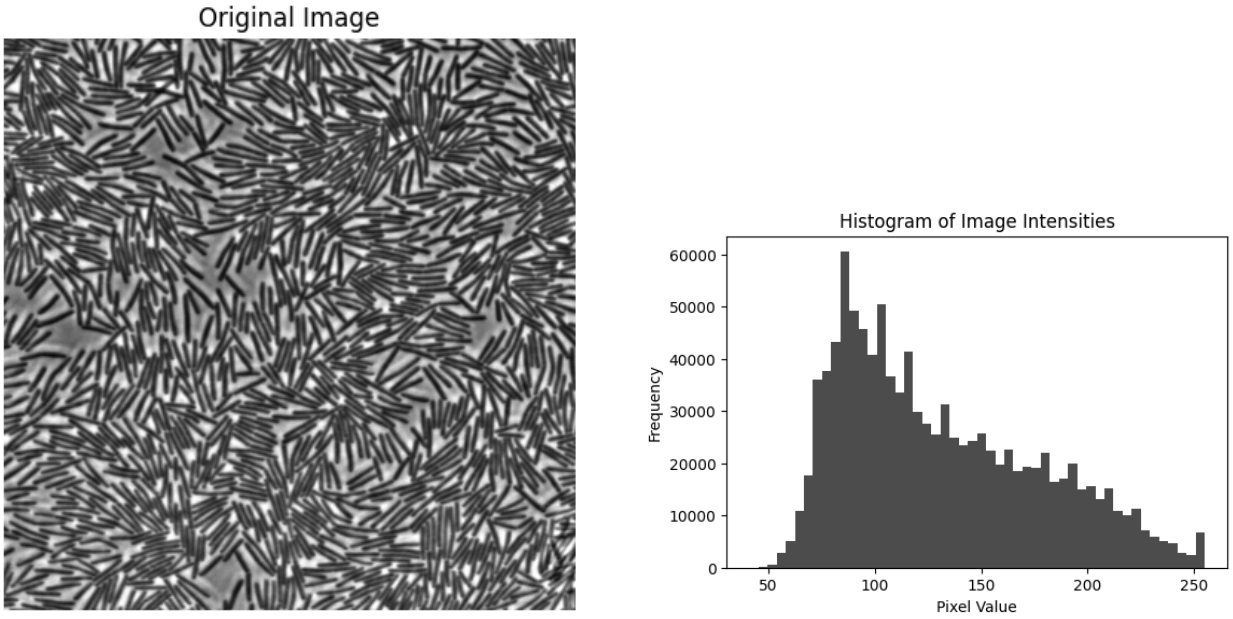
To determine the most appropriate number of clusters K, we applied KMeans clustering for a range of values from 2 to 10 and calculated the silhouette score for each result as shown in Figure **??** and 7b. The silhouette score is a metric that quantifies how well-separated and compact the clusters are, with values closer to 1 indicating better clustering quality.

Since the original images are very large (1024×1024), computing the silhouette score directly on the full-resolution feature matrix would be computationally expensive. Therefore, each feature image was resized to 256×256 before constructing the reduced feature matrix used for silhouette evaluation. This resizing allowed us to efficiently estimate the optimal number of clusters while preserving the overall structure of the data.

This approach allowed us to compare clustering performance objectively without requiring ground-truth labels. Visual inspection of the clustered images also supported this finding, with the selected K value producing segmentation that aligned well with biologically interpretable regions in the image.

3 Results

3.1 Original Image and Intensity Distribution

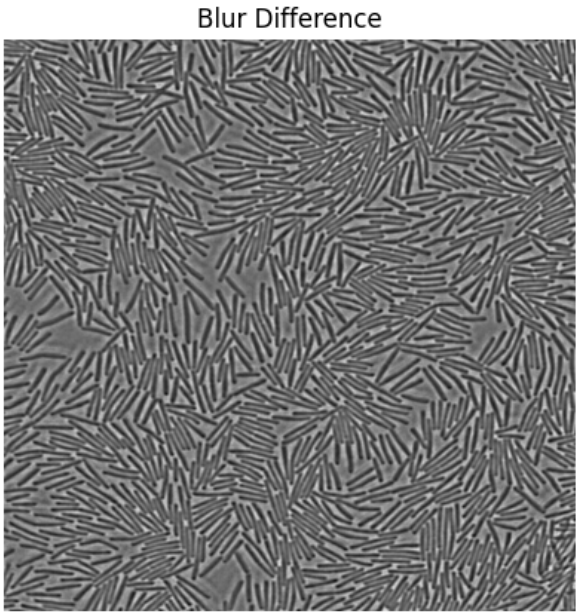


(a) Original Image 1.

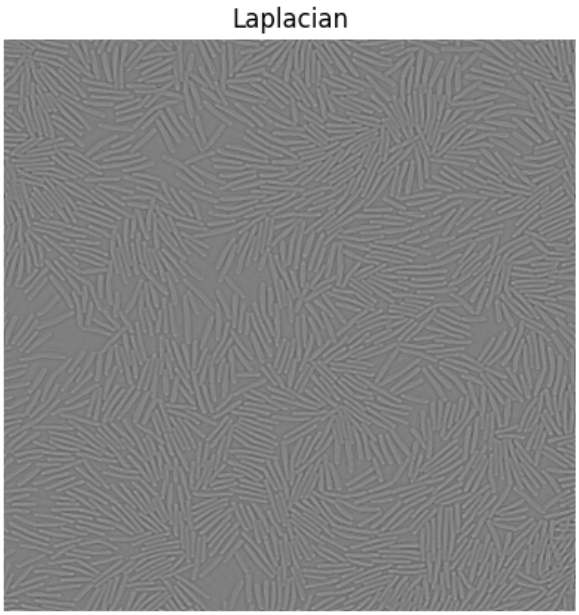
(b) Histogram of Image 1.

Figure 1: Original microscopy image 1 and its pixel intensity distribution: **(a)** Example from the dataset showing a high-density *E. coli* bacterial colony. The close proximity and overlapping of bacterial structures present challenges for segmentation. **(b)** The histogram of Image 1 shows a wide and uneven distribution of pixel intensities, with multiple local peaks. This pattern reflects the high density of bacterial colonies, where close proximity and overlapping between bacteria lead to complex intensity variations. The spread across a broad intensity range indicates that differentiating bacterial structures from the background is challenging.

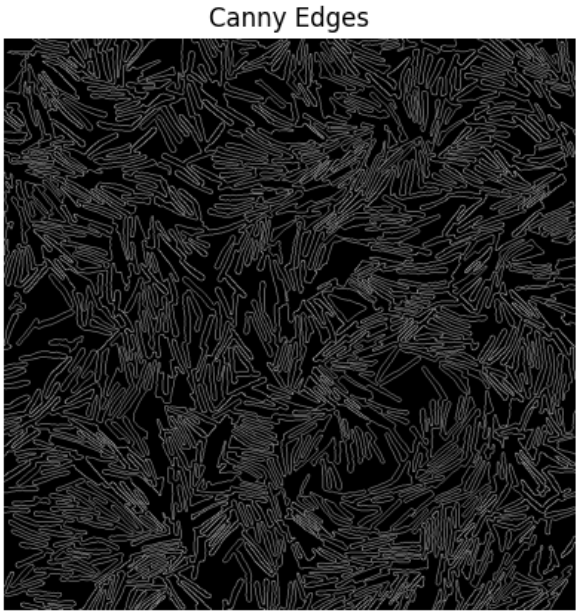
3.2 Feature Extraction



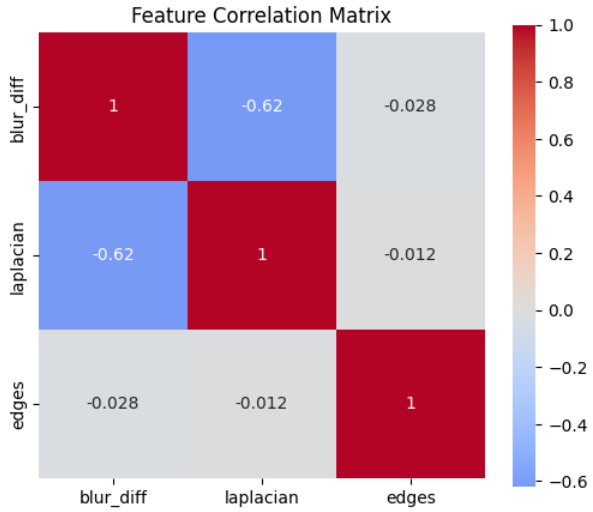
(a) Blur Difference (Image 1).



(b) Laplacian (Image 1).



(c) Canny Edges (Image 1).



(d) Feature Heatmap Correlation (Image 1).

Figure 2: Extracted features and feature correlation matrix for Image 1: **(a)** Blur Difference highlighting regions with intensity changes between blurred and original images. **(b)** Laplacian emphasizing second-order intensity changes corresponding to edges and fine details. **(c)** Canny Edges extracting strong edges within the bacterial structures. **(d)** Feature correlation heatmap showing the pairwise linear correlation between extracted features. The heatmap indicates low correlation between the features, suggesting that they provide complementary information for clustering.

3.3 Feature Space Visualization and Optimal K Selection

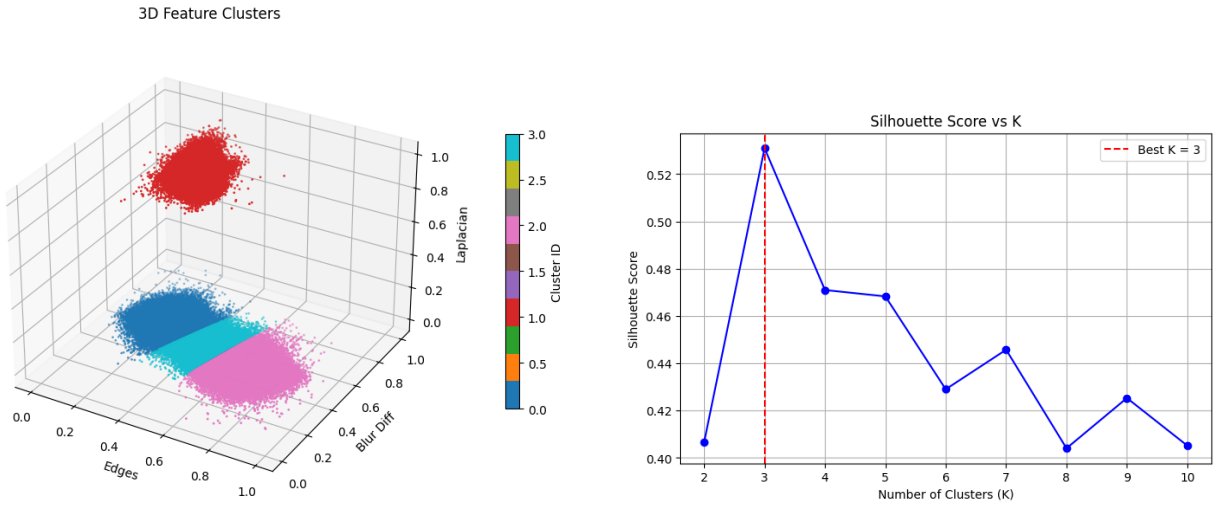
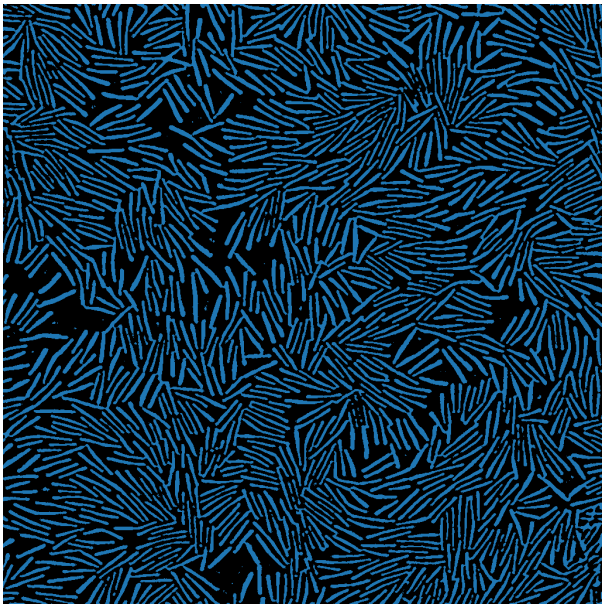


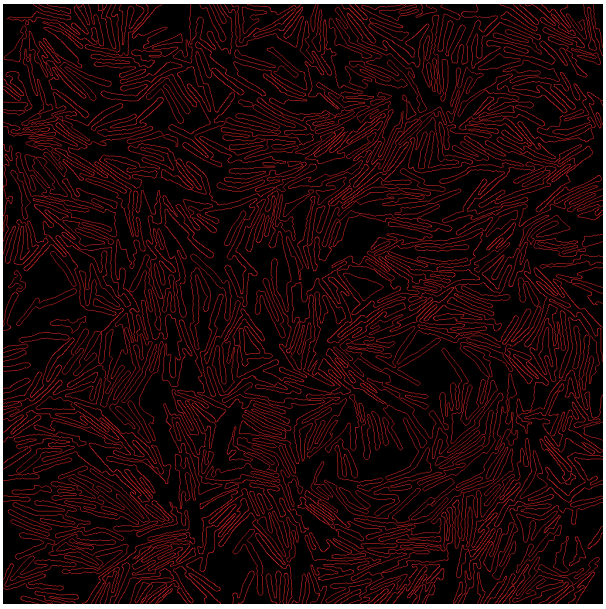
Figure 3: (a) 3D scatter plot of the extracted feature vectors for Image 1. Each point represents a pixel, and colors correspond to cluster assignments determined by KMeans. The axes represent the three extracted features: Edges, Blur Difference, and Laplacian.

(b) Silhouette scores as a function of K for Image 1. Although the maximum silhouette score was achieved at $K = 3$, visual inspection of the clustering results indicated that $K = 4$ provided a more meaningful segmentation of the bacterial structures. Therefore, $K = 4$ was selected for the final clustering.

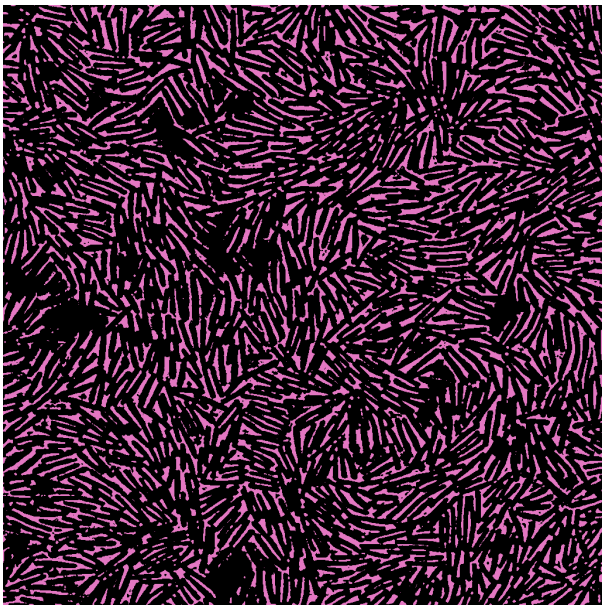
3.4 Segmentation Results



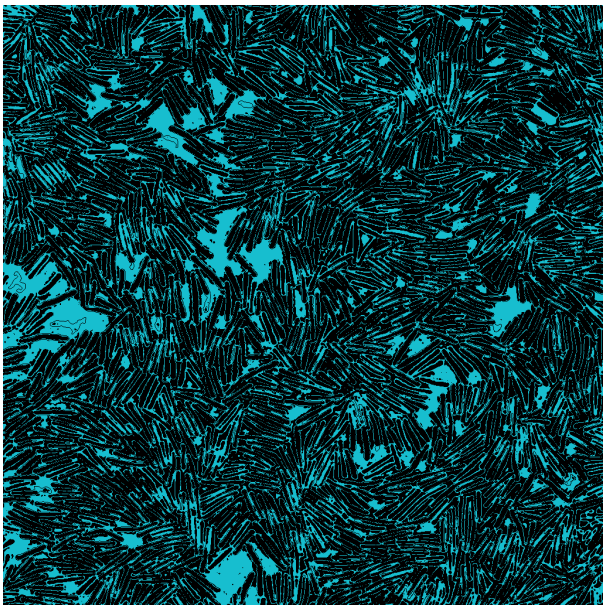
(a) Cluster 0 (Image 1)



(b) Cluster 1 (Image 1)



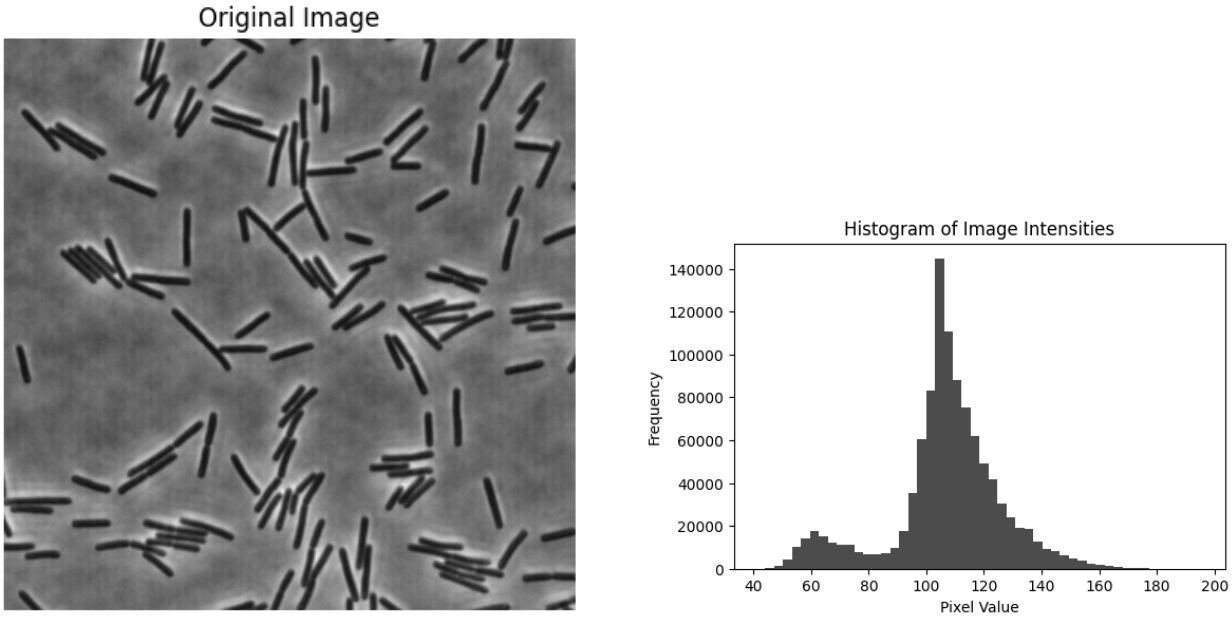
(c) Cluster 2 (Image 1)



(d) Cluster 3 (Image 1)

Figure 4: Cluster segmentation results for Image 1: **(a)** Captures primarily the pixels corresponding to the bacterial structures; this cluster will be used to generate the segmentation mask. **(b)** Pixels in this cluster are mainly the borders of the bacteria. **(c)** Cluster highlighting the edges and extremities of individual *E. coli* bacteria, as well as dense regions where bacterial cells overlap. **(d)** Pixels in this cluster are mainly the background.

3.5 Original Image 2 and Intensity Distribution

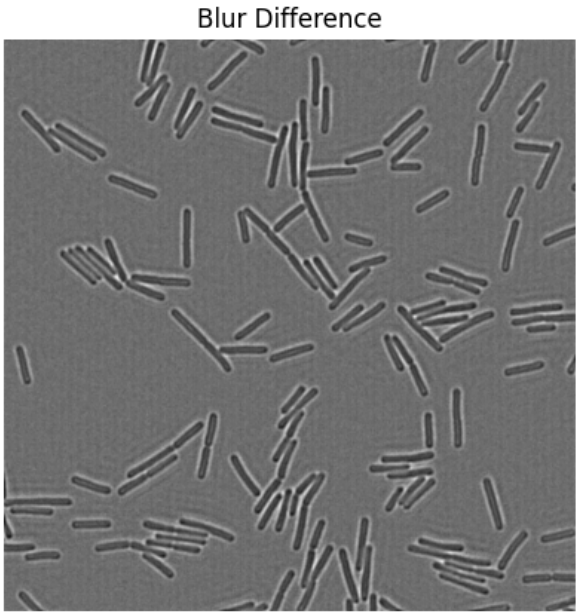


(a) Original Image 2.

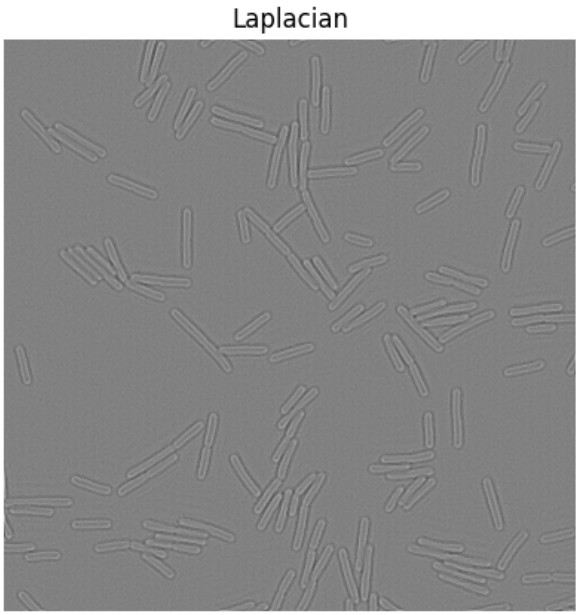
(b) Histogram of Image 2.

Figure 5: Original microscopy image 2 and its pixel intensity distribution: **(a)** Example from the dataset showing a low-density *E. coli* bacterial colony. **(b)** The histogram of Image 2 exhibits a narrower, unimodal distribution of pixel intensities. This reflects the sparse arrangement of bacterial colonies, where clearer separation between bacteria and background simplifies the segmentation task. The concentrated intensity range suggests lower image complexity compared to the dense sample.

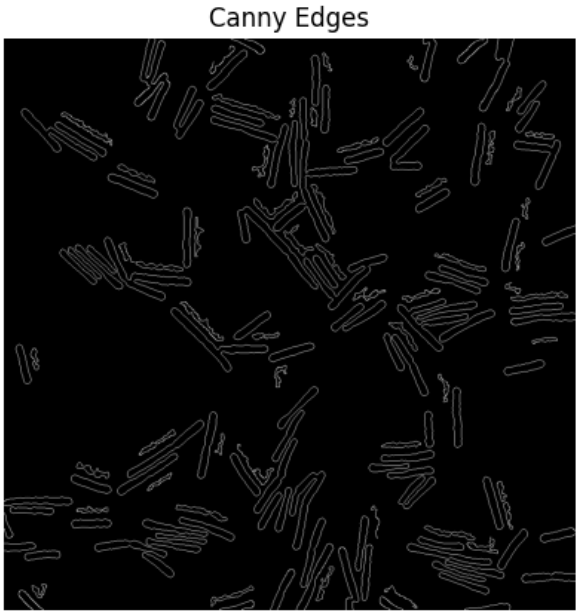
3.6 Feature Extraction



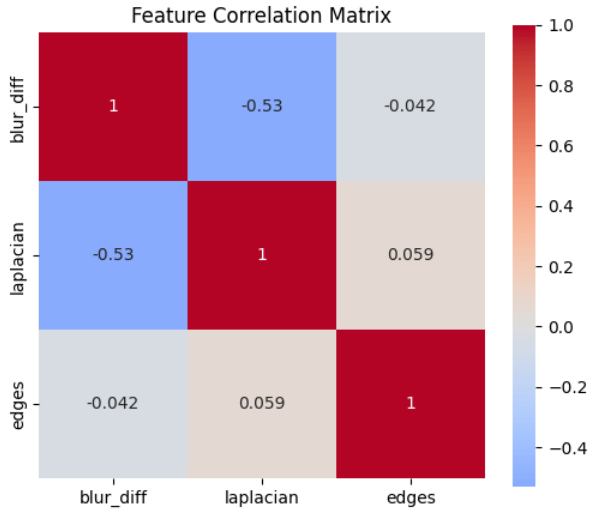
(a) Blur Difference (Image 2).



(b) Laplacian (Image 2).



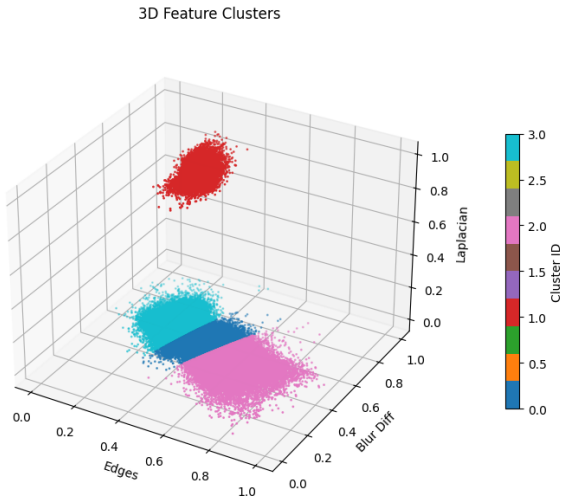
(c) Canny Edges (Image 2).



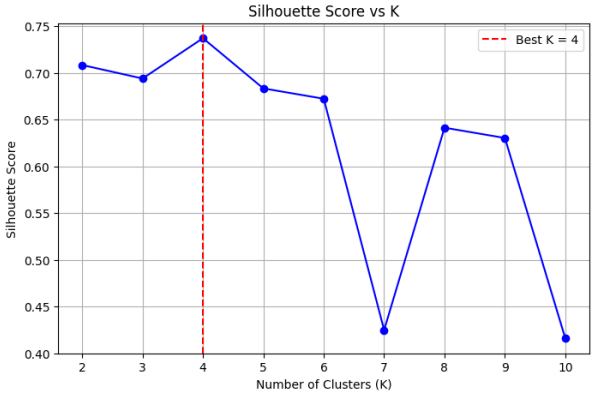
(d) Feature Heatmap Correlation (Image 2).

Figure 6: Extracted features and feature correlation matrix for Image 2: **(a)** Blur Difference highlighting regions with intensity changes between blurred and original images. **(b)** Laplacian emphasizing second-order intensity changes corresponding to edges and fine details. **(c)** Canny Edges extracting strong edges within the bacterial structures. **(d)** Feature correlation heatmap showing the pairwise linear correlation between extracted features. The heatmap indicates low correlation between the features, suggesting that they provide complementary information for clustering.

3.7 Feature Space Visualization and Optimal K Selection



(a) 3D scatter plot of feature vectors.

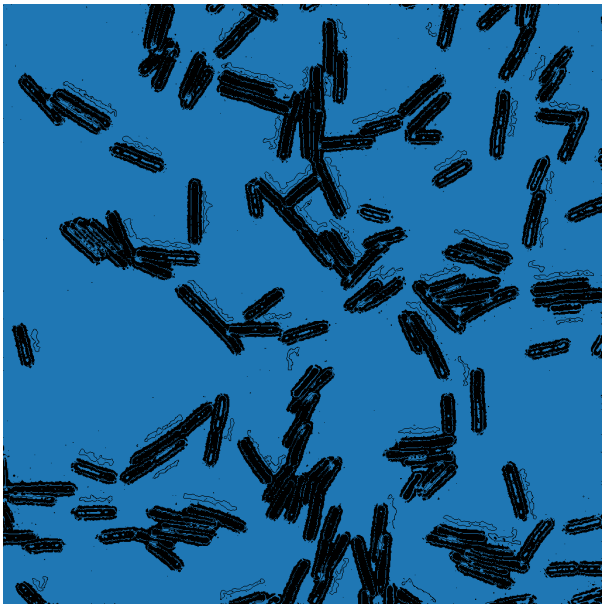


(b) Silhouette scores vs. K.

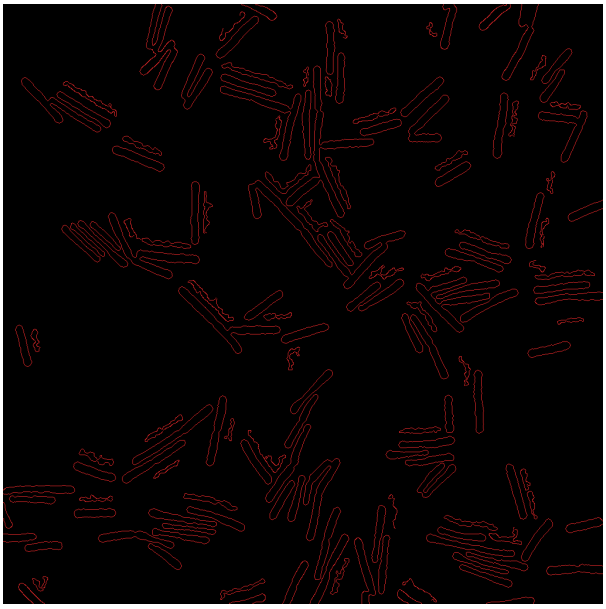
Figure 7: (a) 3D scatter plot of the extracted feature vectors for Image 2. Each point represents a pixel, and colors correspond to cluster assignments determined by KMeans. The axes represent the three extracted features: Edges, Blur Difference, and Laplacian.

(b) Silhouette scores as a function of K for Image 2. The highest silhouette score was achieved at $K = 4$, and this value was selected as the optimal number of clusters for segmenting the image.

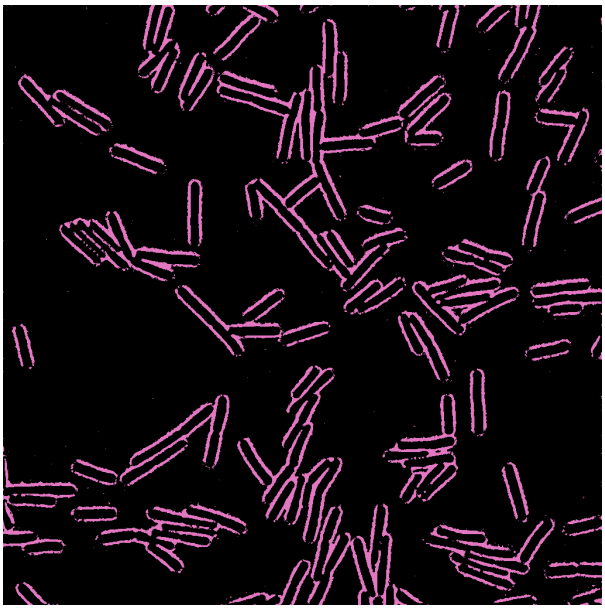
3.8 Segmentation Results



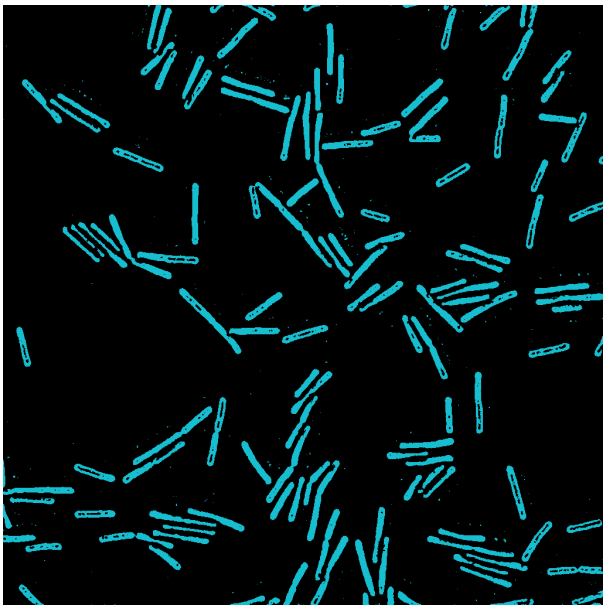
(a) Cluster 0 (Image 2)



(b) Cluster 1 (Image 2).



(c) Cluster 2 (Image 2)



(d) Cluster 3 (Image 2)

Figure 8: Cluster segmentation results for Image 2: **(a)** Pixels in this cluster are mainly the background. **(b)** Pixels in this cluster are mainly the borders of the bacteria. **(c)** Cluster highlighting the edges and extremities of individual *E. coli* bacteria, as well as dense regions where bacterial cells overlap. **(d)** Captures primarily the pixels corresponding to the bacterial structures; this cluster will be used to generate the segmentation mask.

4 Discussion

4.1 Interpretation of Segmentation Results

Clustering-based segmentation successfully distinguished bacterial regions from background in dense and sparse images. The use of hand-made features, blur difference, Laplacian, and Canny edges, provided sufficient information for Kmeans to form coherent clusters corresponding to biologically meaningful structures.

In dense images, segmentation was more challenging due to overlapping bacterial colonies and blurred boundaries, but the results remained visually consistent with expected bacterial distributions. Sparse images exhibited clearer separations between clusters, likely due to reduced interference between bacterial structures.

4.2 Challenges in Feature Extraction and Interpretation

This project marked my first experience performing feature engineering on image data. Identifying suitable features that could separate bacterial regions from the background proved to be challenging. Although classic image filters such as blur difference, Laplacian, and Canny edges provided a reasonable starting point, selecting and combining these features required intuition, experimentation, and careful observation.

Despite relying on statistical methods, such as silhouette scores, to validate clustering quality, visual inspection of segmentation results played an equally important role. In some cases, even if the statistical analysis suggested one value of K as optimal, visual examination of the resulting segmented images highlighted that slightly different values produced more biologically meaningful separations. Thus, both quantitative metrics and qualitative visual evaluation were considered in determining the best clustering configuration.

4.3 Alternative Statistical Validation via Synthetic Bacterial Images and Monte Carlo Approach

Another possible strategy to strengthen the validation of the chosen K involves a Monte Carlo simulation based on synthetic image generation. Specifically, a generative adversarial network (GAN) could be trained to generate realistic artificial images of bacterial colonies, closely mimicking the structural and visual properties of the real microscopy images.

The proposed validation process would involve the following steps:

- Training a GAN model on the real bacterial images to generate new, synthetic bacterial colony images.
- Extracting the same handcrafted features (blur difference, Laplacian, Canny edges) from the synthetic images.
- Applying KMeans clustering to the extracted features from multiple independently generated synthetic images.
- Calculating silhouette scores for each synthetic dataset over a range of K values.
- Comparing the distribution of silhouette scores obtained from the synthetic datasets with those from the real bacterial images.

If the real bacterial data consistently produce significantly higher silhouette scores compared to the synthetic images, it would provide stronger statistical evidence that the chosen clustering captures meaningful biological structure rather than random or visually plausible patterns.

However, due to hardware limitations, it was not feasible to train a GAN and conduct extensive Monte Carlo simulations within the scope of this project.