Image Segmentation

L M Y C Lansakara

KIC-HNDCSAI-251F-005

1.INTRODUCTION

Image segmentation is a fundamental process in computer vision used to simplify and analyze images by partitioning them into meaningful segments. This report details the application of the non-supervised machine learning algorithm, K-Means Clustering, to perform color-based image segmentation.

The objective is **Fruit Segmentation (Strawberry)**, specifically to clearly delineate the red fruit body, the green calyx (leaf) structure, and the background, which is a common requirement in automated quality inspection.

2.METHODOLOGY

The process of segmenting the image data using K-Means clustering involves several sequential steps, implemented using Python libraries such as OpenCV and NumPy.

2.1. IMAGE LOADING AND PRE-PROCESSING

1. **Image Loading:** The input strawberry image was loaded using OpenCV's cv2.imread. OpenCV loads images in the default BGR (Blue, Green, Red) format.
2. **Color Space Conversion:** To ensure that the clustering process relies on perceptually uniform color distances, the loaded image was converted from BGR to Lab color space (Luminosity, A-axis, B-axis) using cv2.cvtColor. The Lab space's separation of luminance (L) from chroma (a and b) makes it superior to RGB for color-based clustering.
3. **Data Reshaping:** The 3D image array (Height * Width * Channels) was flattened into a 2D matrix of pixel vectors (Total Pixels * Channels). This preparation step, often using numpy.reshape, is necessary to feed the pixel color values as data points into the K-Means algorithm.

2.2. K-MEANS CLUSTERING

1. **Algorithm Selection:** K-Means clustering was executed via OpenCV's highly optimized cv2.kmeans function. This function takes the pixel data, the number of clusters (K), criteria for stopping, attempts, and initial centers as inputs.
2. **Number of Clusters (K):** The number of clusters, K, was chosen to be 3.
   - Justification for K: The choice of K=3 was explicitly based on the expected number of visually distinct, high-level components in the image: 1) The red fruit body, 2) The green calyx (leaf) structure, and 3) The background/shadow region. This value directly supports the goal of separating these three meaningful regions.

3. **Clustering Execution:** The cv2.kmeans function iteratively calculates the cluster centroids (color centers) and assigns a label to every pixel, grouping them based on minimal distance in the Lab color space.

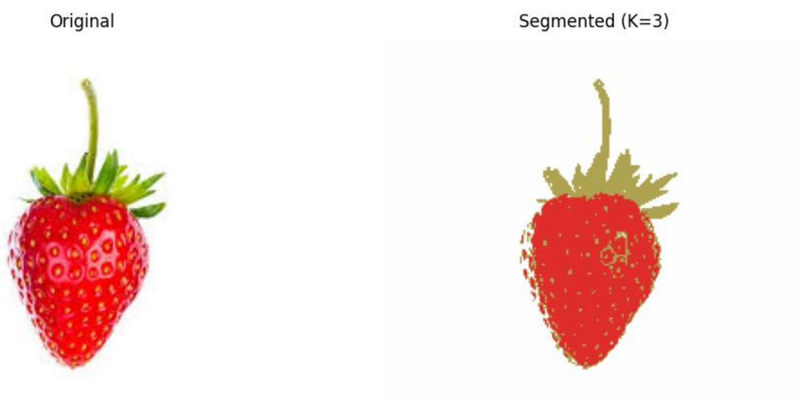
2.3. IMAGE RECONSTRUCTION

After clustering, the segmented image was reconstructed:

1. Each pixel's assigned cluster label was used to replace its original color with the color of its cluster centroid. This process effectively quantizes the image colors to only K distinct values.
2. The resulting segmented 2D array was then reshaped back into the original image dimensions (Height * Width * Channels) and converted from the Lab color space back to **BGR** (and then potentially to RGB for display) for the final output.

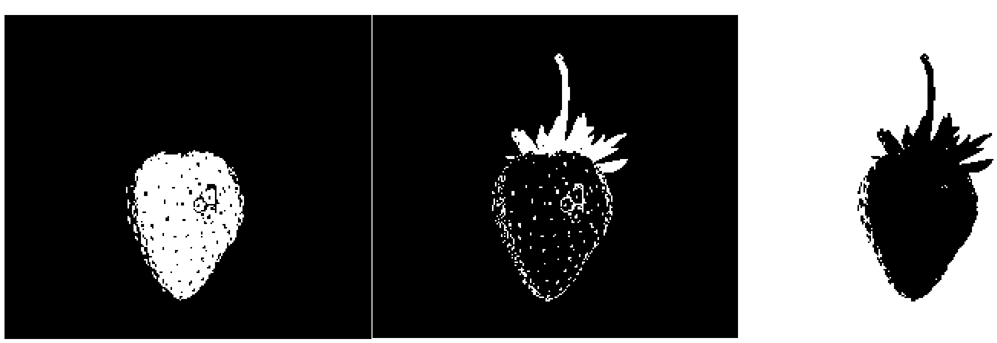
3. RESULTS

3.1. VISUAL COMPARISON



3.2. INDIVIDUAL CLUSTER MASKS

The three distinct clusters generated by the K-Means algorithm (with K=3) are visualized below as binary masks, clearly showing the isolated regions:



cluster 2: fruit body | cluster 1: leaf structure | cluster 0: background

The segmented output clearly demonstrates the effectiveness of K-Means clustering in grouping pixels with similar color properties, resulting in distinct boundaries for the target regions.

a. **Cluster 2 (Fruit Body):** This cluster successfully isolated the main red/pink fruit surface and any red-toned shadows on the fruit.

b. **Cluster 1 (Leaf Structure):** This cluster corresponds to the green calyx and stem components, demonstrating precise color separation of the non-fruit parts attached to the object.

c. **Cluster 0 (Background/Shadow):** This cluster successfully isolated the background and the shadow cast by the fruit, achieving the primary goal of separating the foreground object from its environment.

4. DISCUSSION AND ANALYSIS

4.1. JUSTIFICATION OF K=3 AND OBSERVED RESULTS

The choice of K=3 was optimal as it directly maps to the three required meaningful image components (fruit, leaf, and background). Using K=2 would have forced the merger of the fruit and leaf into a single foreground cluster, losing critical feature detail. Conversely, a high K (e.g., K=5) would have over-fragmented the homogeneous regions, such as splitting the fruit body into minor 'light red' and 'dark red' sub-clusters, which is counterproductive to the object separation goal.

The use of the Lab color space was critical. By separating luminosity (L) from chroma (A, B), the clustering was based on true color similarity, not brightness variation. This ensured that shaded red areas on the strawberry were correctly grouped with brightly lit red areas, resulting in a robust, uniform cluster for the fruit body, as demonstrated by the clarity of the Cluster 2 mask.

4.2. LIMITATION AND FUTURE WORK

The primary limitation is that K-Means is purely color-based and ignores spatial proximity. If a bright red object existed in the background, it would be incorrectly clustered with the strawberry body. Future work should augment the feature vector to include **spatial coordinates** (x, y) along with the Lab values (L, a, b) to create super-pixels, which would enforce spatial continuity and improve boundary smoothness.

5. CONCLUSION

This experiment successfully applied K-Means clustering for strawberry image segmentation. The strategic choice of the Lab color space and the optimal cluster count of K=3 allowed for the effective isolation of the fruit, leaf, and background. This simple, efficient method provides a strong foundation for real-world computer vision tasks like automated quality control.