Analysis of segmentation and clustering of leukocyte images

1st Mateus S. Herbele
Department of Informatics
Federal University of Paraná
Curitiba, Brazil
msh22@inf.ufpr.br

2nd Ana Paula Sodré
Department of Informatics
Federal University of Paraná
Curitiba, Brazil
apas19@inf.ufpr.br

***Abstract*—This document presents an analysis of the segmentation and clustering of 260 leukocyte images using the k-means algorithm. The study evaluates the silhouette scores generated with clusters for two distinct K values in both RGB and HSV color spaces. The best configuration is further analyzed by applying blur with two different kernel size, and recalculating the silhouette scores. This comprehensive evaluation aims to identify the optimal preprocessing and clustering settings for improving automated leukocyte classification accuracy.**

***Index Terms*—cluster, segmentation, rgb, hsv, images, k-means**

I. Introduction

This document focuses on the application of the k-means algorithm to segment and cluster a dataset of 260 leukocyte images. The analysis is conducted in two color spaces: RGB (Red, Green, Blue) and HSV (Hue, Saturation, Value). The choice of color space can significantly impact the clustering results, as different color representations can highlight various aspects of the images. The primary objective of this study is to evaluate the silhouette scores—a metric that assesses the quality of clustering—across different values of K (the number of clusters) in both color spaces. Silhouette scores provide insight into how well-separated and cohesive the clusters are, which is crucial for determining the effectiveness of the clustering process. Furthermore, the study explores the impact of preprocessing techniques, specifically image blurring with different kernel sizes, on clustering performance. Blurring is applied to investigate its effect on the silhouette scores and to identify the optimal preprocessing strategy for enhancing classification accuracy. This comprehensive analysis aims to establish best practices for preprocessing and clustering settings, ultimately contributing to improved automated leukocyte classification systems.

II. K-means Clustering Algorithm

The k-means clustering algorithm is a widely used method for partitioning a dataset into K distinct clusters. The core idea is to group data points into clusters such that the within-cluster variance is minimized. The specific implementation of the k-means algorithm described here is designed to handle RGB and HSV color spaces for image segmentation. [1]

[1]The implementation was based on the method presented in [1].

A. Algorithm Overview

The k-means algorithm operates iteratively to refine the clusters. The steps of the algorithm are as follows:

1) **Initialization**:
   - The algorithm begins by randomly selecting K initial centroids from the dataset. These centroids represent the initial cluster centers.
2) **Distance Calculation**:
   - For each data point, the algorithm calculates the Euclidean distance to each of the K centroids. The distance function used depends on the color space:
     - **RGB Space**: The distance between two points is calculated based on their RGB values.
     - **HSV Space**: The distance is computed using HSV color values.
3) **Cluster Assignment**:
   - Each data point is assigned to the cluster with the nearest centroid based on the calculated distances. This assignment creates K clusters where each data point belongs to the cluster with the closest centroid.
4) **Centroid Update**:
   - After assigning all data points to clusters, the algorithm updates the centroids. The new centroid for each cluster is computed as the mean of all data points assigned to that cluster.
5) **Iteration**:
   - The algorithm repeats the distance calculation and centroid update steps until convergence. Convergence is achieved when the centroids no longer change significantly between iterations.
6) **Termination**:
   - The algorithm terminates when the centroids stabilize. At this point, the final clusters are produced.

B. Algorithm Implementation

The implementation of the k-means algorithm in this study follows these steps:

- **Initialization**: The centroids are initialized by randomly sampling K data points from the dataset.

- **Distance Calculation (RGB and HSV)**:
  - In the RGB space, the distance between two points (R_1, G_1, B_1) and (R_2, G_2, B_2) is calculated using the Euclidean distance formula:

$$d = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2}$$

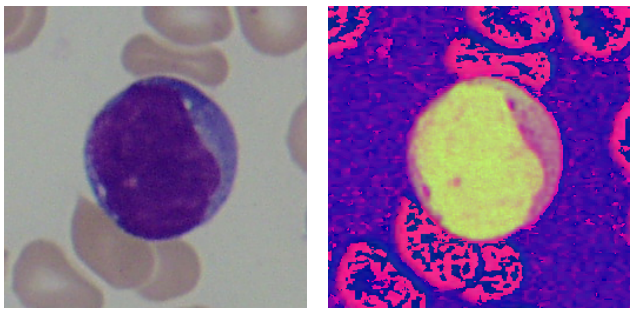
  - In the HSV space, the distance calculation follows a similar approach, but operates on hue, saturation, and value components.
- **Cluster Assignment and Centroid Update**:
  - Each point is assigned to the nearest centroid, and centroids are updated by calculating the mean of points in each cluster:

$$\text{New Centroid} = \frac{1}{|C_k|} \sum_{i \in C_k} x_i$$

    where C_k is the cluster k and x_i represents the data points in C_k.
- **Convergence Check**:
  - The algorithm checks for convergence by evaluating the change in the mean position of the centroids. If the change is below a specified threshold, the algorithm converges.

C. Images after clustering

After executing the algorithm, it was generated the images resulted from K-means clustering for each sample. As a example to show, it is possible to notice the differences between the images generated for Image 001[2] (in ALL IDB folder).

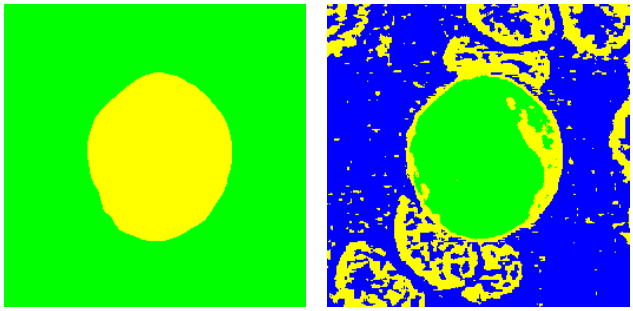


(a) RGB Original Image (Image 001) (b) HSV Converted Image

Fig. 1: Original Images

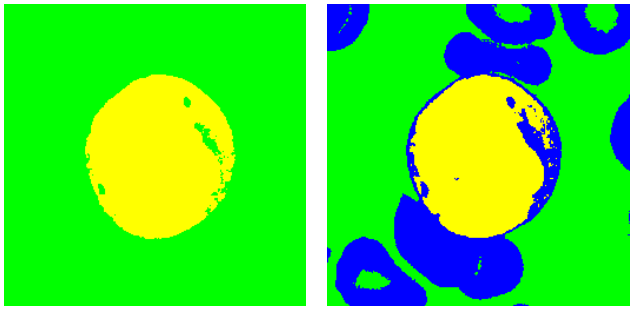
The first two images (Figure 3) shows the results of K-means applied to the Image 001, in the RGB color space, for k = 2 and k = 3. In the first image, the clusters are represented by two distinct colors (yellow and green). Here the boundaries between the clusters are clearer, with the cellular distinctly separated from the background. And in the second image, the clusters are represented by the colors yellow, green and blue.

[2]The images used in this study were obtained from Scotti's Lab [2]



(a) HSV Image generated after K-means with K = 2 (b) HSV Image generated after K-means with K = 3

Fig. 2: HSV Images result after clustering



(a) RGB Image generated after K-means with K = 2 (b) RGB Image generated after K-means with K = 3

Fig. 3: RGB Images result after clustering

In this case, the boundaries are a little less clear, with more detailed segmentation within the cellular and its surroundings.

In the HSV images (Figure 2), the first case is not very different from the RGB image, for k = 2. It has the same colors representing the two clusters. In the second case, despite having the same colors present in RGB, it is possible to notice that the background has more noise than in RGB image generated.

III. Silhouette Score

The Silhouette Score is a metric used to evaluate the quality of clustering in a dataset. It measures how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The Silhouette Score ranges from -1 to 1, where:

- 1 indicates that the data points are well-clustered, meaning that the clusters are dense and well-separated.
- 0 indicates that the data points are on or very close to the decision boundary between two neighboring clusters.
- Negative values indicate that the data points might have been assigned to the wrong cluster, as they are closer to a neighboring cluster than to the cluster they are assigned to.

For each sample, the metric is calculated by first determining the average distance from the sample to all other points within the same cluster, known as *a(i)* (the mean intra-cluster distance). Next, the average distance from the sample to all

points in the nearest cluster that the sample is not a part of is calculated, known as *b(i)* (the mean nearest-cluster distance). The Silhouette Score for the sample is given by the following *s(i)* formula.

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

A Silhouette Score close to +1 indicates that the sample is far away from the neighboring clusters, implying well-clustered data. A score around 0 suggests the sample is near the boundary between clusters, while a score less than 0 indicates the sample might be in the wrong cluster.

IV. Results

The analysis of clustering performance using the average silhouette score revealed that the HSV color model with k = 2 achieved the highest average silhouette score of 0.7981. This result indicates that, for this configuration, the HSV color space and two clusters provide the most distinct separation between different clusters, leading to well-separeted groups with minimal overlap. The Table I shows all the results generated in the study. It is possible to notice that, when the number of clusters increase to k = 3, the average silhouette score for both color models declined. This behavior can be explained by several factors. As the number of clusters increases, the clustering problem becomes more complex, which can lead to overfitting where the model capture noise or minor variations rather than underlying data structure. Furthermore, the increased number of clusters can make the model more sensitive to noise, which distorts cluster boundaries and lowers the overall effectiveness of clustering.

Color System	K	Average Silhouette Score
RGB	2	0.7526
RGB	3	0.7087
HSV	2	0.7981
HSV	3	0.6797

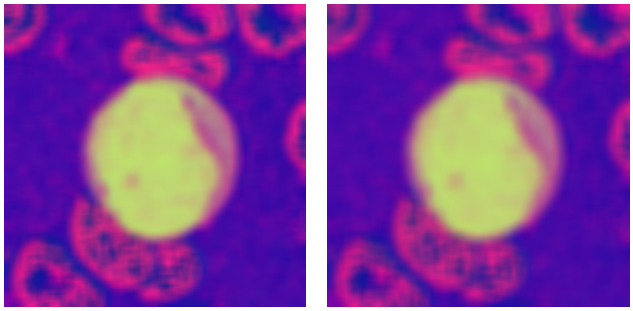
TABLE I: Silhouette Scores for Different Color Systems and Cluster Numbers

V. Blurring Images

After defining the best configuration of color model and value of k, the HSV converted images and k = 2 was used to the next step of this study. The images in the HSV color model were blurred by applying, using the cv2.blur [3], a kernel size of 11x11 and 13x13 to each image, which aimed to reduce noise and enhance the clustering process.

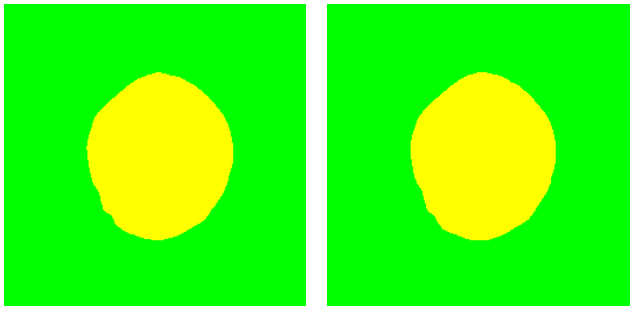
VI. Blurred Images Results

The results showed an average silhouette score of 0.7901 for the 11x11 blur and a slightly higher score of 0.7990 for the 13x13 blur. These results suggests that blurring the images improves clustering performance, with the 13x13 blur yielding the highest average silhouette score. This indicates that a larger blur size helps in achieving better defined clusters, possibly by effectively smoothing out noise while preserving essential cluster boundaries.



(a) HSV 11x11 Blurred Image (b) HSV 13x13 Blurred Image

Fig. 4: HSV Blurred images result before clustering



(a) HSV 11x11 Blurred Image generated after K-means with K = 2 (b) HSV 13x13 Blurred Image generated after K-means with K = 2

Fig. 5: HSV Blurred images result after clustering

Blur	Average Silhouette Score
11x11	0.7901
13x13	0.7990

TABLE II: Silhouette Scores for Different Blur Configurations

Conclusion

[3]This study evaluated the effectiveness of k-means clustering on leukocyte images using different color models (RGB and HSV) and cluster sizes (k = 2 and k = 3). The findings indicated that the HSV color model with k = 2 provided the highest average silhouette score of 0.7981, suggesting that this configuration achieved the most distinct separation between clusters. To further refine clustering accuracy, image blurring was introduced as a preprocessing step using kernel sizes of 11x11 and 13x13. The results demonstrated that the blurring processing improved clustering outcomes, with the 13x13 blur achieving the highest average silhouette score of 0.7990. This improvement suggests that appropriate blurring can effectively reduce noise and enhance cluster definition, thereby improving the reliability of the clustering process. Overall, the study highlights the importance of selecting suitable color models and preprocessing techniques to optimize clustering performance. The insights provided by this work inform future research and

[3]The image processing and clustering results were obtained using the code from the GitHub repository [4]. The repository contains all the scripts and configurations used to produce the results discussed in this paper.

practical applications in biomedical image analysis, emphasizing the need for tailored approaches to enhance the accuracy and effectiveness of clustering methods.

REFERENCES

[1] Towards Data Science, "Semantic Segmentation of Remote Sensing Imagery Using K-Means," 2024. [Online]. Available: https://towardsdatascience.com/semantic-segmentation-of-remote-sensing-imagery-using-k-means-e4c165d9218e. [Accessed: 05-Aug-2024].

[2] G. Scotti, "All images from Scotti's Lab," 2024. [Online]. Available: https://scotti.di.unimi.it/all/. [Accessed: 05-Aug-2024].

[3] OpenCV, "OpenCV: Filtering," 2024. [Online]. Available: https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html. [Accessed: 05-Aug-2024].

[4] Mateus Herbele, "clustering-leukocyte-images," GitHub repository, 2024. [Online]. Available: https://github.com/MateusHerbele/clustering-leukocyte-images. [Accessed: 05-Aug-2024].