Report on Image Alteration using K-Means Clustering

This report explains my Python program that alters an image based on a provided seed image. The alterations are done using a technique called K-Means Clustering, which groups similar pixels together.

Purpose:

The purpose of this program is to modify an image based on a user-defined seed image. The seed image specifies regions that need to be altered in the original image.

How It Works:

Input Images:
The program takes two images as input:
Original Image:
The image to be altered.
Seed Image:
An image indicating which parts of the original image should be modified.

Conversion to Binary Mask:

The seed image is converted into a binary mask where pixels with non-zero values represent the region to be modified, and pixels with zero values represent the background.

Clustering Seed Pixels:

K-Means clustering is applied separately to the foreground (region to be modified) and background pixels in the original image.

Likelihood Calculation:

For each pixel in the original image, the program calculates the likelihood of it belonging to the foreground or background cluster using Euclidean distances and exponential functions.

Altering Pixels:

Based on the calculated likelihoods, the program determines whether each pixel should belong to the foreground or background. If the likelihood of belonging to the foreground is higher, the pixel remains unchanged from the original image; otherwise, it is set to black.

Displaying Results:

The original image, seed image, and altered image are displayed side by side for comparison.

Key Parameters:

Number of Clusters:
Determines the number of clusters used in K-Means clustering. More clusters can capture finer details but may increase processing time.
Beta Values:
Control the smoothness of the likelihood function. Higher values make the decision boundary sharper.

Result:

The program produces an altered image where the specified regions in the seed image are modified while the rest remains unchanged.

Limitations:

- The effectiveness of the alterations heavily depends on the accuracy of the seed image.
- Processing time can be significant for large images or a high number of clusters.

Conclusion:

This simple program demonstrates how K-Means clustering can be utilized to alter images based on a seed image, allowing for targeted modifications while preserving the original content.