İzmir Democracy University
Faculty of Engineering
Electrical and Electronics Engineering
Introduction to Computer Vision
2024-2025 Fall Filter Project

Prepared By:	Suat Deniz	2006102002
	Abdulkadir Dağlar	2006102033
	Berkay Caplık	2206102900

1. Introduction

In this project, we created a filter that makes an image look unique by using two different effects. The left side of the image shows edges to highlight important shapes, while the right side uses K-Means clustering to create a colorful heatmap. This mix of effects makes the image both interesting and useful.

Our main goal was to design the filter without using advanced libraries. We wrote the code with using basic math and matrix operations. This helped us learn more about how image processing works.

In this report, we will explain how we made the filter, show the results, and talk about what we learned during the project.

2. Methods and Tools

In this project, we used MATLAB to create a filter that makes the image look interesting by dividing it into two parts. Each part has a different effect, and we didn't use any advanced libraries to keep it original.

2.1 Edge Detection (Left Half)

On the left side of the image, we used a Sobel filter to find edges. This filter helps us see the lines and shapes in the picture. Here's how we did it:

1. We used two small grids (called kernels) to find horizontal and vertical edges.
2. Then, we combined these results to calculate the edge strength for each pixel.
3. Finally, we adjusted the values to make the edges clear and easy to see.

This method shows the important details and lines in the left half of the image.

2.2 K-Means Clustering (Right Half)

For the right side, we used K-Means clustering to make a heatmap effect. Here's what we did:

1. We took the pixel colors from the right side of the image and grouped them into 8 clusters.
2. Each cluster got a new color, which is the average of all the colors in that group.
3. After this, the right side looked like a colorful, simplified version of the image.

This effect makes the colors stand out and gives the picture a unique look.

2.3 Combind It All Together

After processing both sides, we combined them into one image. The left side shows edges, and the right side has colorful clusters. This makes the image both functional and fun to look at.

Tools We Used

- **Software:** We used MATLAB because it's great for working with numbers and images.
- **Rules:** We didn't use ready-made image functions. Instead, we wrote everything ourselves to make sure it's original.

3. Steps of Implementation

3.1 Project Planning

- **Objective Definition:** Our primary objective was to create a filter that applies edge detection to the left side of an image and color clustering to the right side.
- Tool and Method Selectio**n:** We selected MATLAB for its strong numerical computing capabilities. We chose not to use any advanced libraries and instead

focused on implementing everything using basic operations.

3.2 Input Preparation

- **Image Selection:** The input images were provided by our instructor. These images included diverse colors and details, which allowed us to evaluate the performance of both filters effectively.
- **Preprocessing:** We did not apply any additional preprocessing to the images. Instead, we used them directly as inputs for the edge detection and clustering algorithms.

3.3 Edge Detection

- **Sobel Filter Design:** We implemented two kernels, one for detecting horizontal edges and another for vertical edges.
- **Edge Strength Calculation:** We computed the edge strength for each pixel by combining horizontal and vertical gradients.
- **Grayscale Visualization:** We visualized the detected edges in grayscale on the left side of the image to highlight shapes and structures effectively.

3.4 Color Clustering (K-Means)

- **Data Preparation:** We extracted the color values of pixels from the right side of the image and prepared them as a three-dimensional matrix.
- **K-Means Implementation:** We grouped the pixels into 8 clusters using the K-Means algorithm and assigned the average color of each cluster back to the corresponding pixels.
- **Heatmap Creation:** The clustered colors created a simplified and visually appealing heatmap on the right side of the image.

3.5 Combining the Results

- **Merging the Outputs:** We combined the processed left and right halves into a single image. This merging process ensured that the final image clearly displayed both the structural details and the color grouping effects.

3.6 Coding and Debugging

- **Step-by-Step Development:** We wrote each part of the algorithm in MATLAB, focusing on modularity and clarity.
- **Error Detection and Fixing:** We analyzed intermediate results to identify and correct any errors in the code.
- **Parameter Optimization:** We fine-tuned the parameters to achieve better visual clarity and accuracy.

3.7 Reporting and Evaluation

- **Documentation:** We documented the entire project workflow for future reference and learning.
- **Analysis:** We evaluated the output against our objectives to verify the effectiveness of the implemented algorithms.

3.8 Team Collaboration

- Due to our small team size, we decided to work on all parts of the project together. Every step, from coding to writing the report, was completed collaboratively. This approach not only ensured consistency across the project but also allowed us to learn from one another and share responsibilities effectively. By working closely as a team, we were able to address challenges more efficiently and produce a cohesive final result.

4. Results

In this section, we show the results of our filter and explain how it works on an example image. The output combines the left side's edge detection and the right side's K-Means clustering.

4.1 Original Image

Input Images

We started with an original image that has a mix of colors and details. This image was divided into two parts: the left side for edge detection and the right side for clustering.

4.2 Final Image

Expected Outcome:

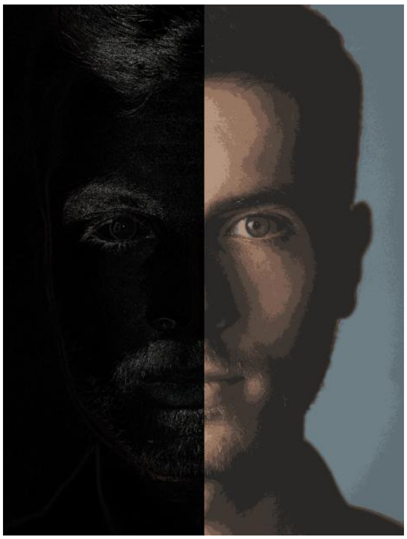
- **Left Side (Edge Detection):** We expected the left side of the image to clearly highlight the edges and outlines of objects. The edges were anticipated to be sharp and well-defined, with minimal noise or unnecessary details.
- **Right Side (K-Means Clustering):** On the right side, we aimed for a simplified yet vibrant heatmap effect. Each cluster of pixels was expected to represent a distinct group of similar colors, resulting in a visually appealing color segmentation.

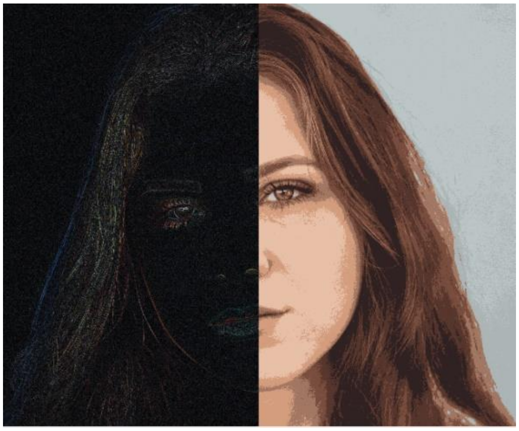
Achieved Outcome:

- **Left Side (Edge Detection):** The edge detection process successfully identified most of the prominent edges in the image. The Sobel filter produced clear outlines, especially for high-contrast areas. However, some minor noise was observed in areas with less contrast, which could be addressed by fine-tuning the threshold parameters.
- **Right Side (K-Means Clustering):** The K-Means clustering algorithm effectively grouped similar colors into clusters, producing a heatmap effect as expected. The transitions between clusters were smooth, and the visual output was both distinct and colorful. In some regions, the clustering results showed minor overlaps in color, which slightly reduced the segmentation clarity.

Comparison (Expected vs. Achieved):

- The overall output closely matched our expectations. The edge detection on the left side met our goal of emphasizing structural details, while the right side achieved the desired heatmap effect. Despite small areas of improvement, such as reducing noise in edge detection and enhancing segmentation in clustering, the final result demonstrates the effectiveness of our approach.





5. Conclusion

This project showed how we can create unique effects on images by using simple tools and algorithms. The combination of edge detection and clustering gave our images a special look while also teaching us a lot about image processing. Writing the code ourselves helped us understand the process better and improved our problem-solving skills.