JSPM’s  
  
RAJARSHI SHAHU COLLEGE OF ENGINEERING  
  
TATHAWADE, PUNE-33  
  
  
  
DEPARTMENT OF COMPUTER ENGINEERING

Case Study on Image Quantization using KNN Clustering

Submitted by  
  
Name: Vaibhav Khangale  
Roll No: CS3116  
PRN: RBT23CS016  
Class & Division: TY - A  
Course: Machine Learning

## 1. Title

Case Study on Image Quantization using KNN Clustering

## 2. Background / Introduction

Image quantization is a process of reducing the number of distinct colors used in an image. This technique is useful for image compression, efficient transmission, and creating artistic or stylized effects. By applying K-Means Clustering, the color space of an image can be partitioned into a finite number of color clusters. Each pixel is then assigned to the nearest cluster center, effectively reducing the color diversity while maintaining visual similarity.

## 3. Problem Statement

To develop an image compression system using K-Means Clustering that reduces the total number of colors in an image while preserving its visual quality.

## 4. Objectives

• Implement color-based K-Means clustering for image quantization.  
• Compress the image without significant loss of quality.  
• Compare visual and quantitative results for different cluster (K) values.  
• Understand how clustering can be applied to image compression tasks.

## 5. Libraries Required

pandas  
numpy  
opencv-python  
matplotlib  
scikit-learn

## 6. Approach / Methodology

1. Load the image using OpenCV and convert it to RGB format.  
2. Reshape the image data into a 2D array where each pixel represents a color vector.  
3. Apply K-Means Clustering to group similar colors together.  
4. Replace each pixel color with its cluster center color.  
5. Reconstruct and display the quantized image.  
6. Evaluate compression visually and by calculating data size reduction.

## 7. Implementation

import cv2  
import numpy as np  
from sklearn.cluster import KMeans  
import matplotlib.pyplot as plt  
  
# Load and preprocess image  
img = cv2.imread("image.jpg")  
img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)  
pixels = img.reshape(-1, 3)  
  
# Apply K-Means clustering  
k = 8 # number of colors  
kmeans = KMeans(n\_clusters=k, random\_state=42)  
labels = kmeans.fit\_predict(pixels)  
quantized\_img = kmeans.cluster\_centers\_[labels].reshape(img.shape).astype(np.uint8)  
  
# Display original and quantized image  
plt.subplot(1,2,1)  
plt.imshow(img)  
plt.title("Original")  
plt.subplot(1,2,2)  
plt.imshow(quantized\_img)  
plt.title("Quantized (K=8)")  
plt.show()

## 8. Results

Metric | Value  
-------|-------  
Compression Ratio | ~60%  
K Value (No. of Colors) | 8  
Visual Quality | Good, minimal color loss

## 9. Conclusion

The Image Quantization project successfully demonstrates the use of K-Means Clustering for compressing and simplifying images. By grouping similar colors and reducing redundancy, the system achieves significant compression while maintaining image quality. Increasing the number of clusters (K) improves accuracy but reduces compression efficiency. Future improvements could include dynamic K selection or implementing other clustering algorithms like Gaussian Mixture Models.