# Image Compression by K-Means Clustering

BY; (WILLIAM) AUSTIN GAY

# Project Specifics

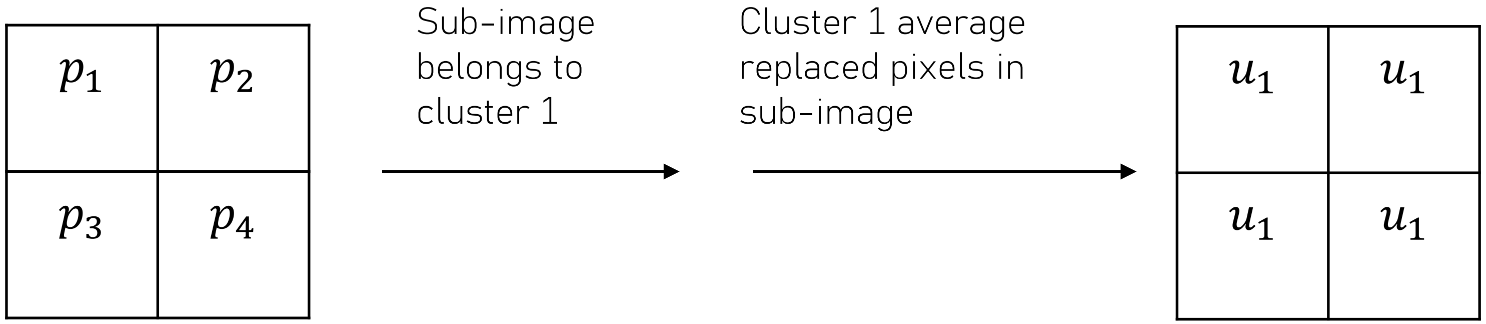
- Goal/Problem: Image Compression
- Method: K-Means Clustering
- Applied to sub-images of an image
- Grayscale Pictures
- Python

# K-Means Clustering Algorithm

- Looks for k clusters in a dataset (mean/centroid)
- Initially, chooses k centroids at random which serve as the center of the clusters
- Assigns each data point to its nearest cluster
- Calculates new centroid by averaging data assigned to each cluster
- Repeats calculations until the centroid doesn't move or the max iterations is reached
- To summarize: K-Means Clustering Algorithm clusters data by assigning it to the cluster with the nearest mean

# Problem: Image Compression

- Goal is to achieve image compression by decreasing the number of colors used in an image (note: the method used will not decrease the size of the image)
- Image is split into sub-images that are run through k-means clustering
- Sub-images will be filled with the same-colored pixels at the end of the algorithm



|  |  |
|---|---|
| $p_1$ | $p_2$ |
| $p_3$ | $p_4$ |

Sub-image belongs to cluster 1

Cluster 1 average replaced pixels in sub-image

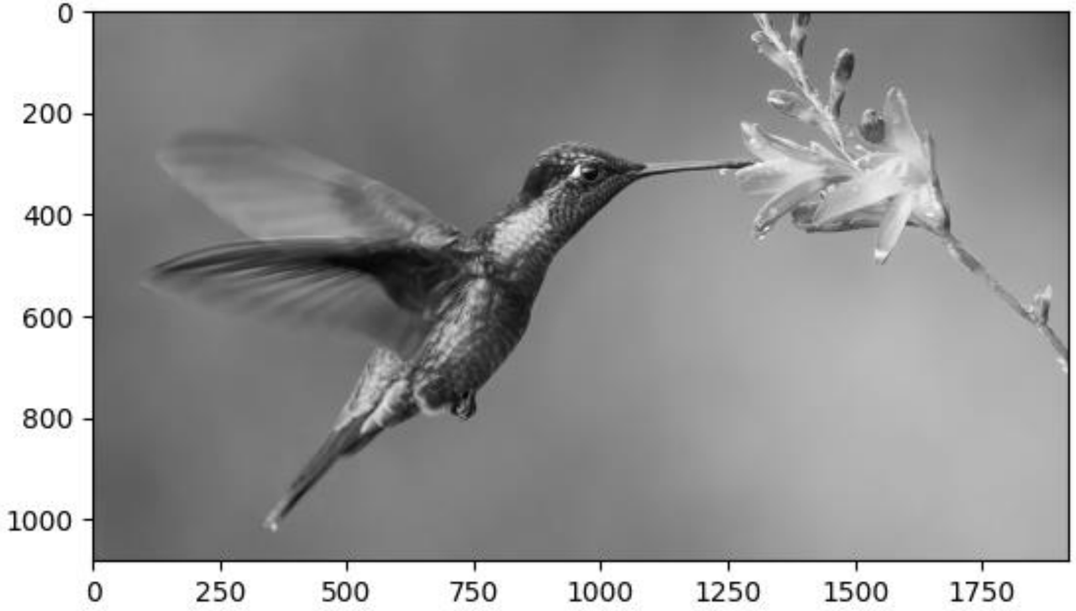
|  |  |
|---|---|
| $u_1$ | $u_1$ |
| $u_1$ | $u_1$ |

# Questions/Experimental Focus

- Number of Sub-Images
- Number of Clusters
- Clarity of image
- Still Recognizable
- Image Compression

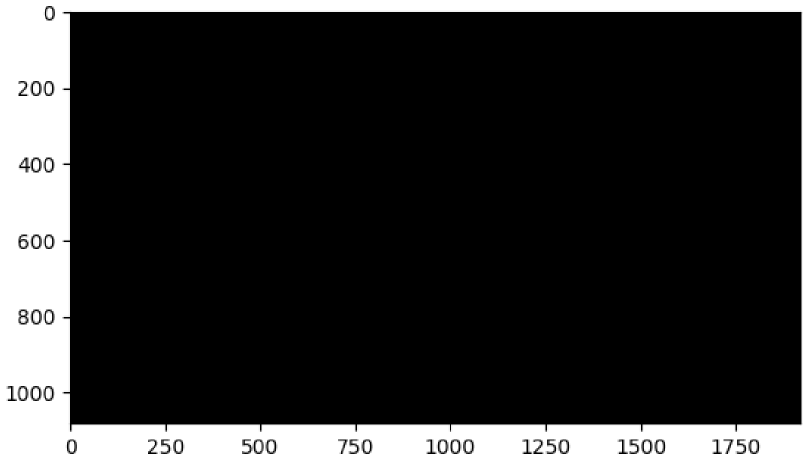
# Experimental Results

# Base Picture in Grayscale

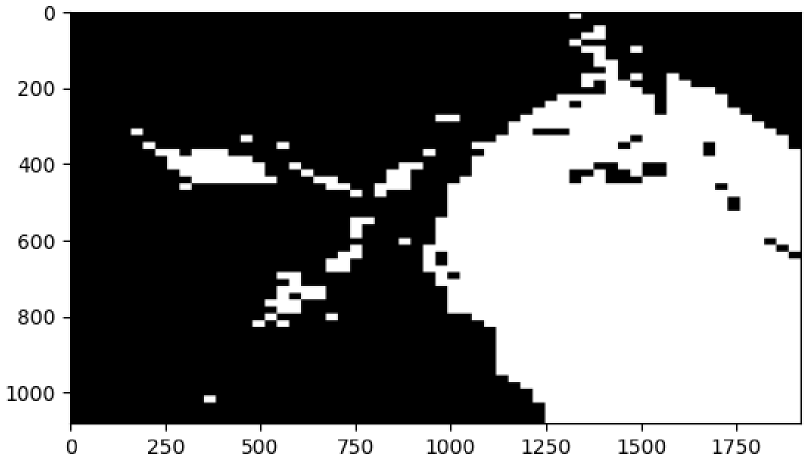


# Number of Clusters (60x60 sub-images)

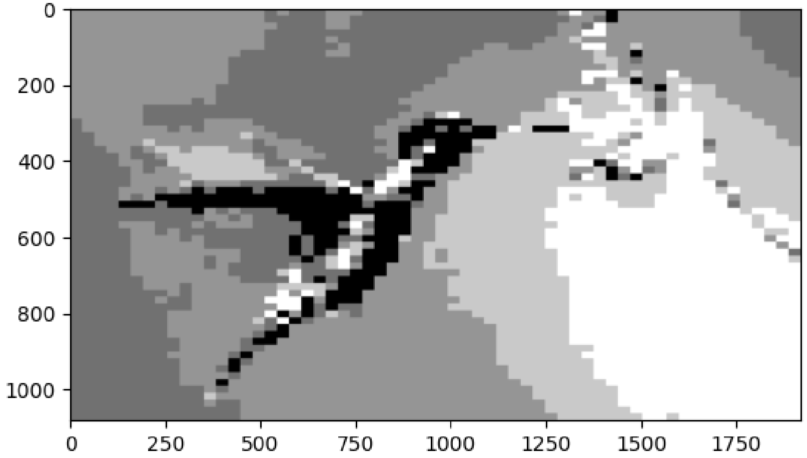
1 cluster



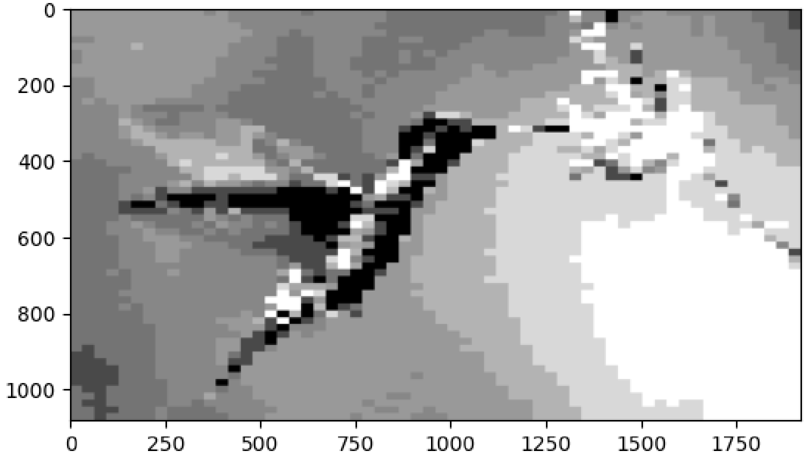
2 clusters



# Number of Clusters (60x60 sub-images)



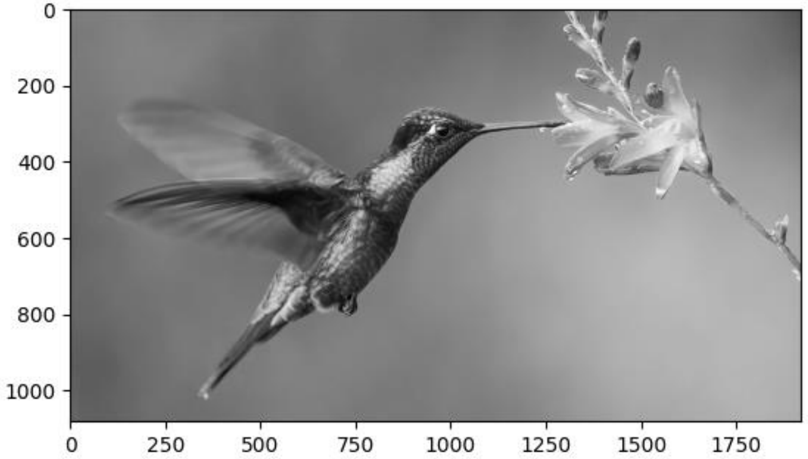
5 clusters



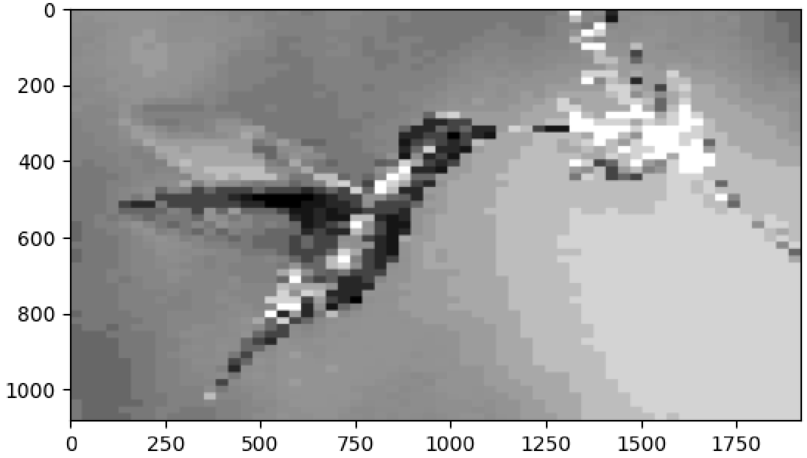
10 clusters

# Number of Clusters (60x60 sub-images)

Original

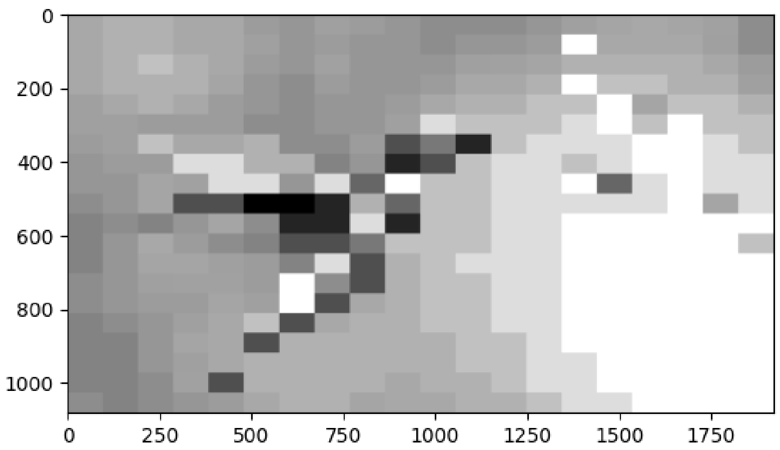


25 Clusters

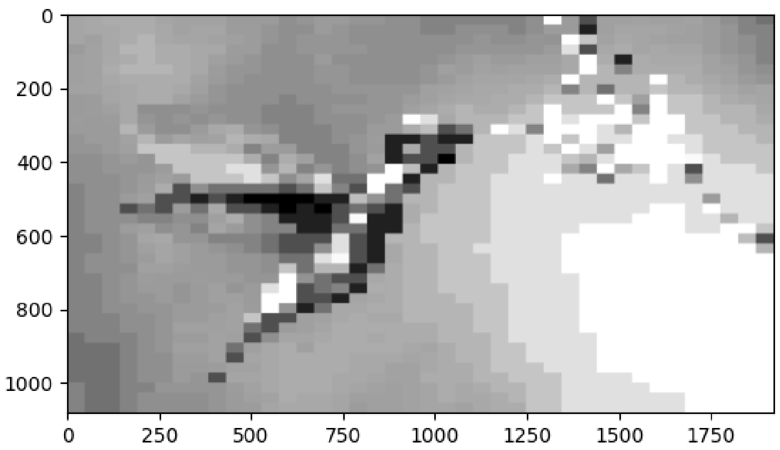


# Number of Sub-Images

20x20 sub-images

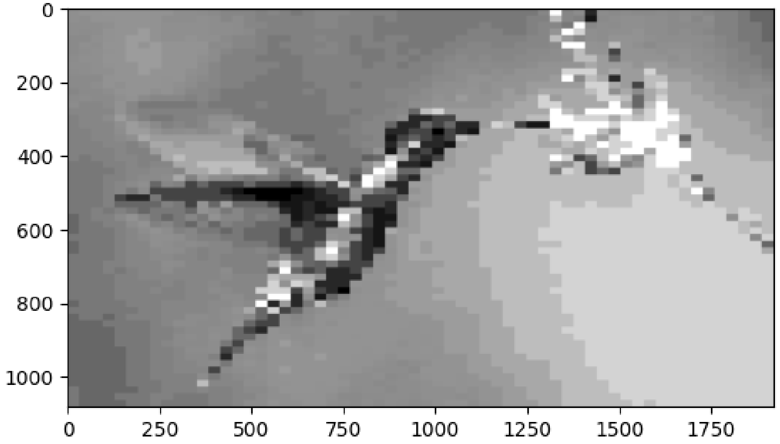


40x40 sub-images

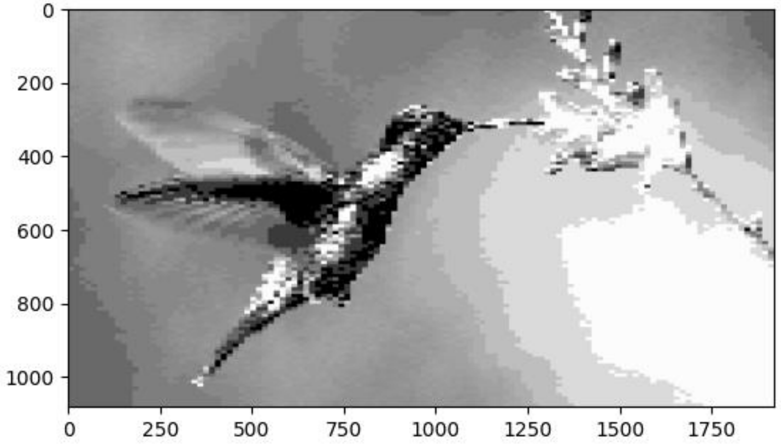


# Number of Sub-Images

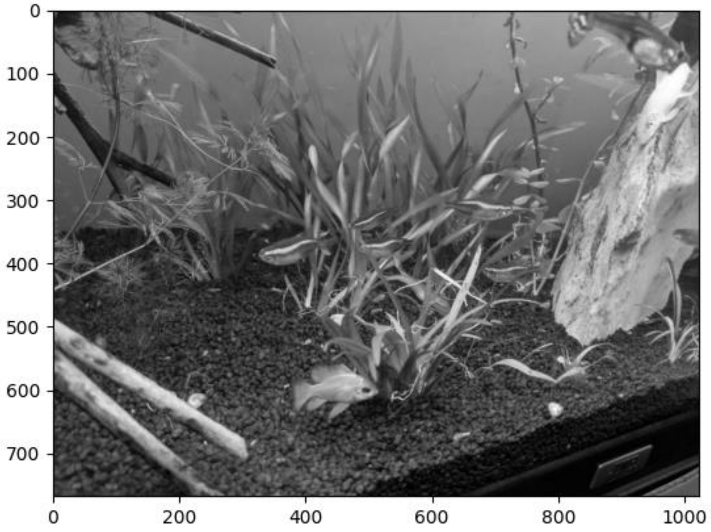
60x60 sub-images

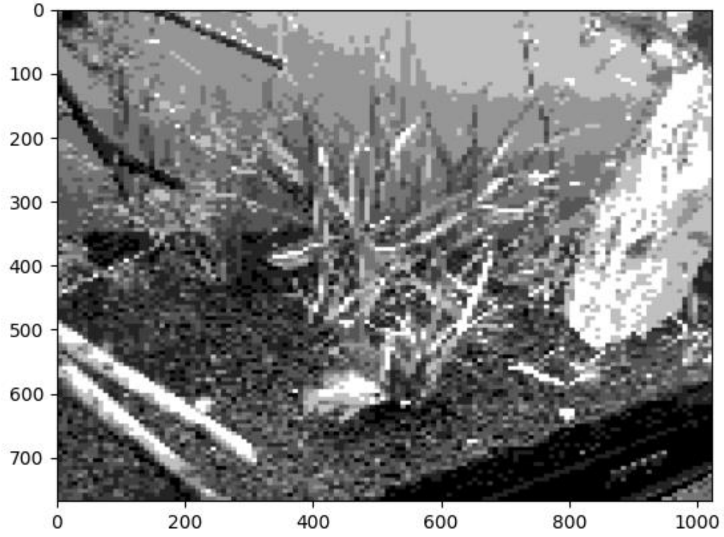


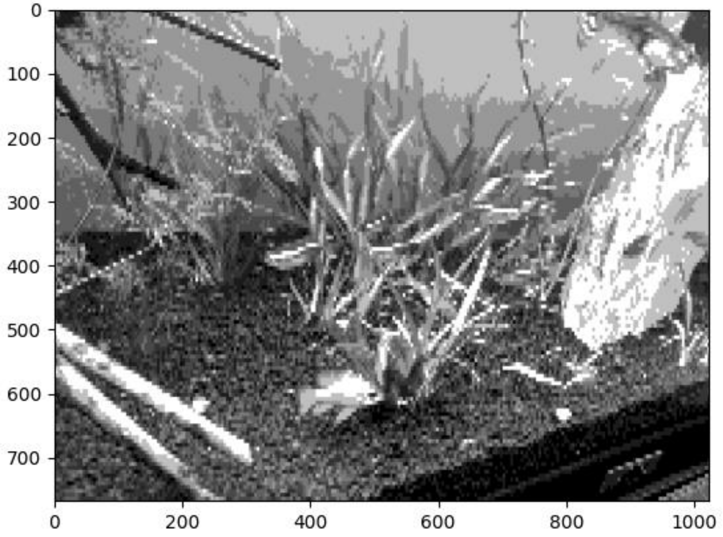
120x120 sub-images



# Number of Sub-Images




120x120 sub-images


128x256 sub-images

# Observations

- After 5-10 clusters you can easily make out most pictures.
- The amount of sub images impacts the clarity of the image the most
- More complicated images require more sub-images to maintain content
- Overall, a good all-around number that seems to be very efficient is 60x60 with 10+ cluster.

# Future Work

- RGB Pictures
- Compare Sub-Image Method to Individual Pixel Method
- Instead of eyesight, use classification to determine if a picture retained its contents after image compression