# Kenneth Mead CSI-5130-13154 / CSI-4130-15496.202110

# Homework 2 K-Means Clustering

**Initial Info**

OS: Windows 10

Libraries used:

import time

import numpy as np

from scipy.spatial.distance import cdist

import matplotlib.pyplot as plt

import matplotlib.image as mpimg

Initial centroids are chosen randomly

  # Randomly choose points to be our centroids

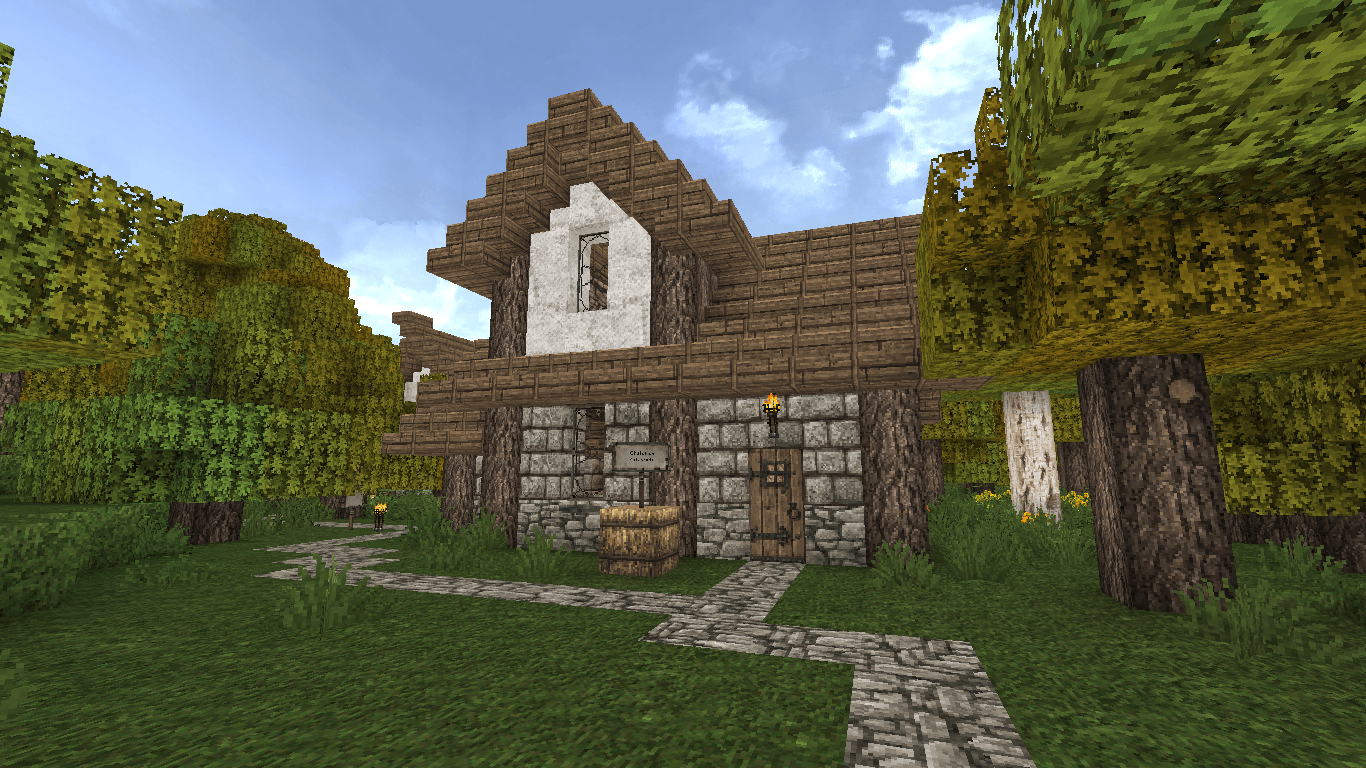
  idx = np.random.choice(len(x), k, replace=False)

For my 1D clustering, I ran my K-means with the 1st color channel of the image’s RGB, and then plotted the cluster labels onto the graph.

For my 3D clustering, I ran my K-means with the RGB values.

For my 5D clustering, the K-Means took in the RGB values, and each pixels XY coordinate.

Images used

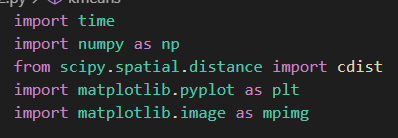




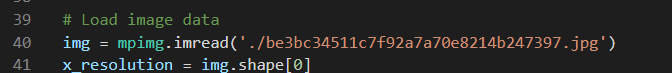


**How To Run**

Install these python packages

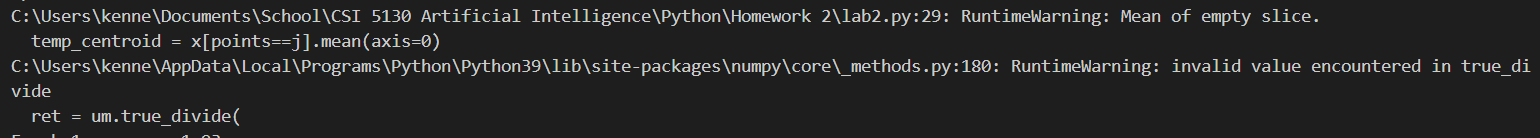


**Line 40:** Specify image to be loaded in

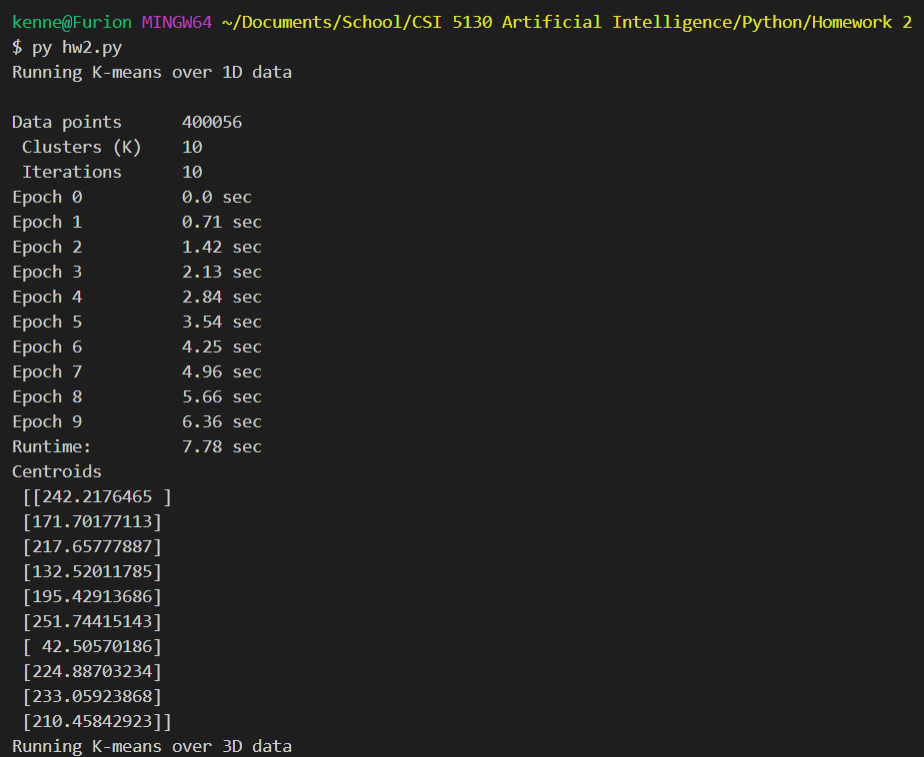


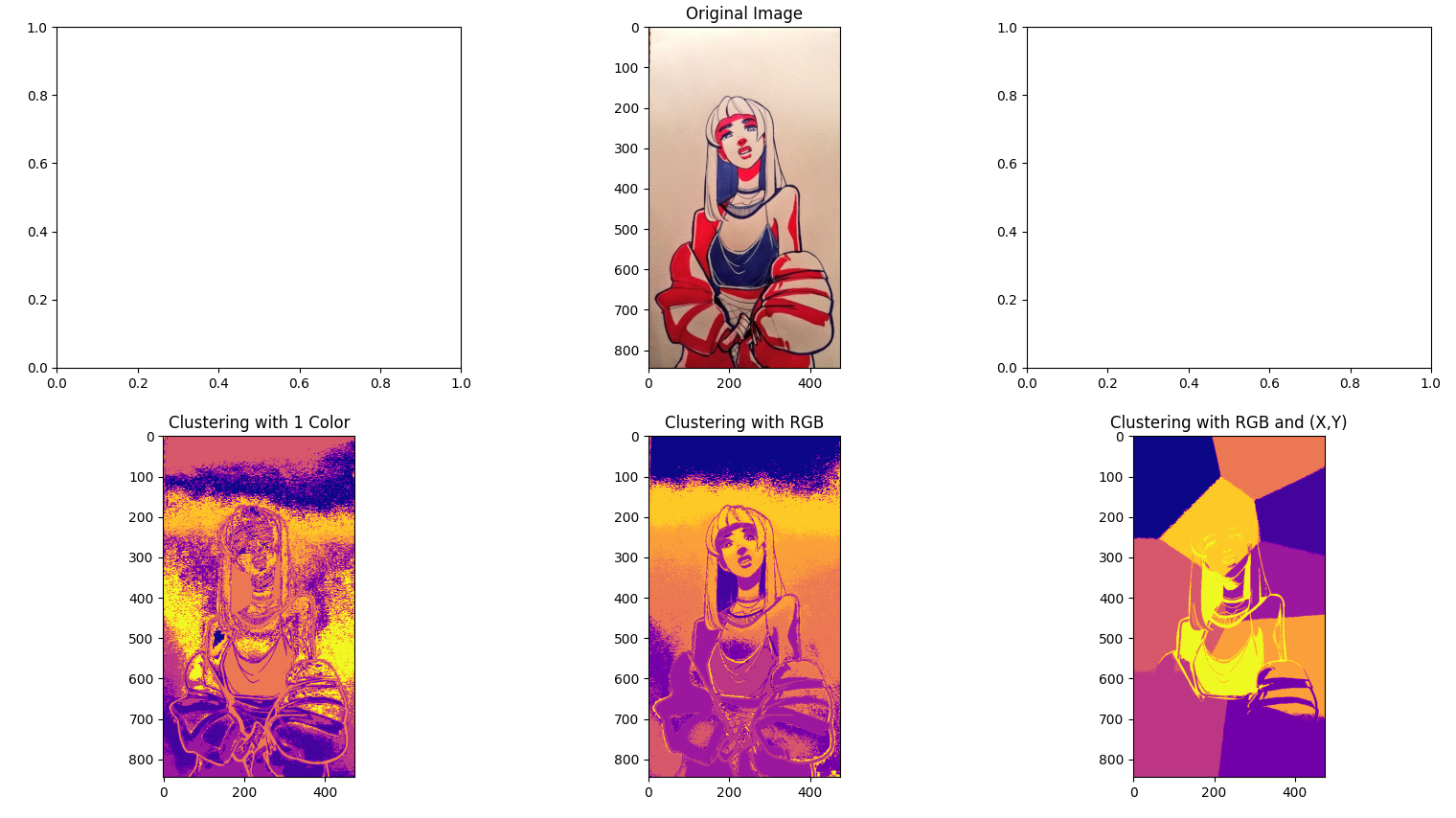
**Line 45-46**: Assign cluster count (k) and epochs/iterations.



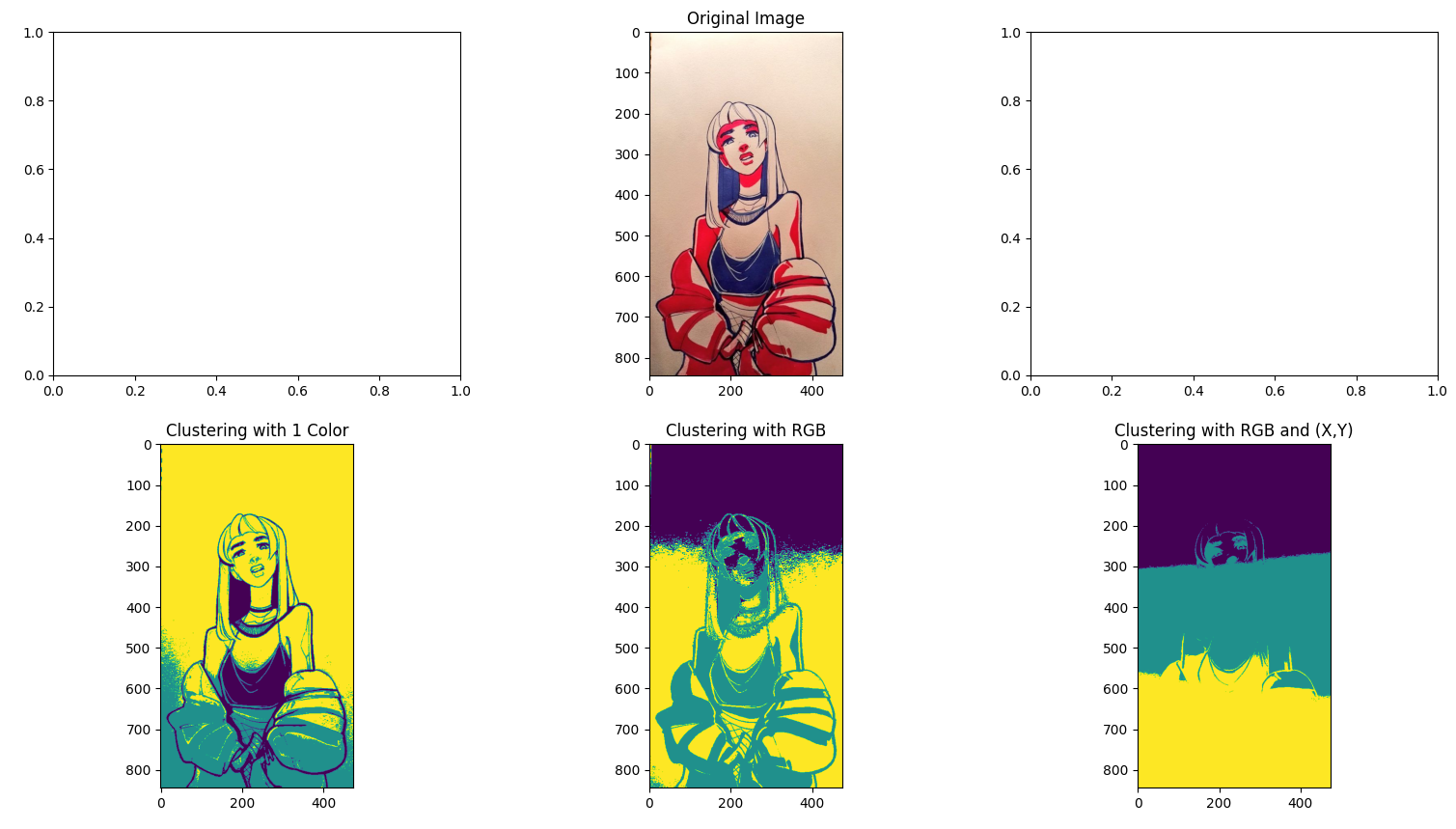
WarningLarge cluster counts can cause the algorithm to not work if the image does not have diverse enough data. **This is especially common when running the 1D data set.** It will still produce images; they will just be garbage data. Try rerunning the code or decreasing cluster count to fix this issue.

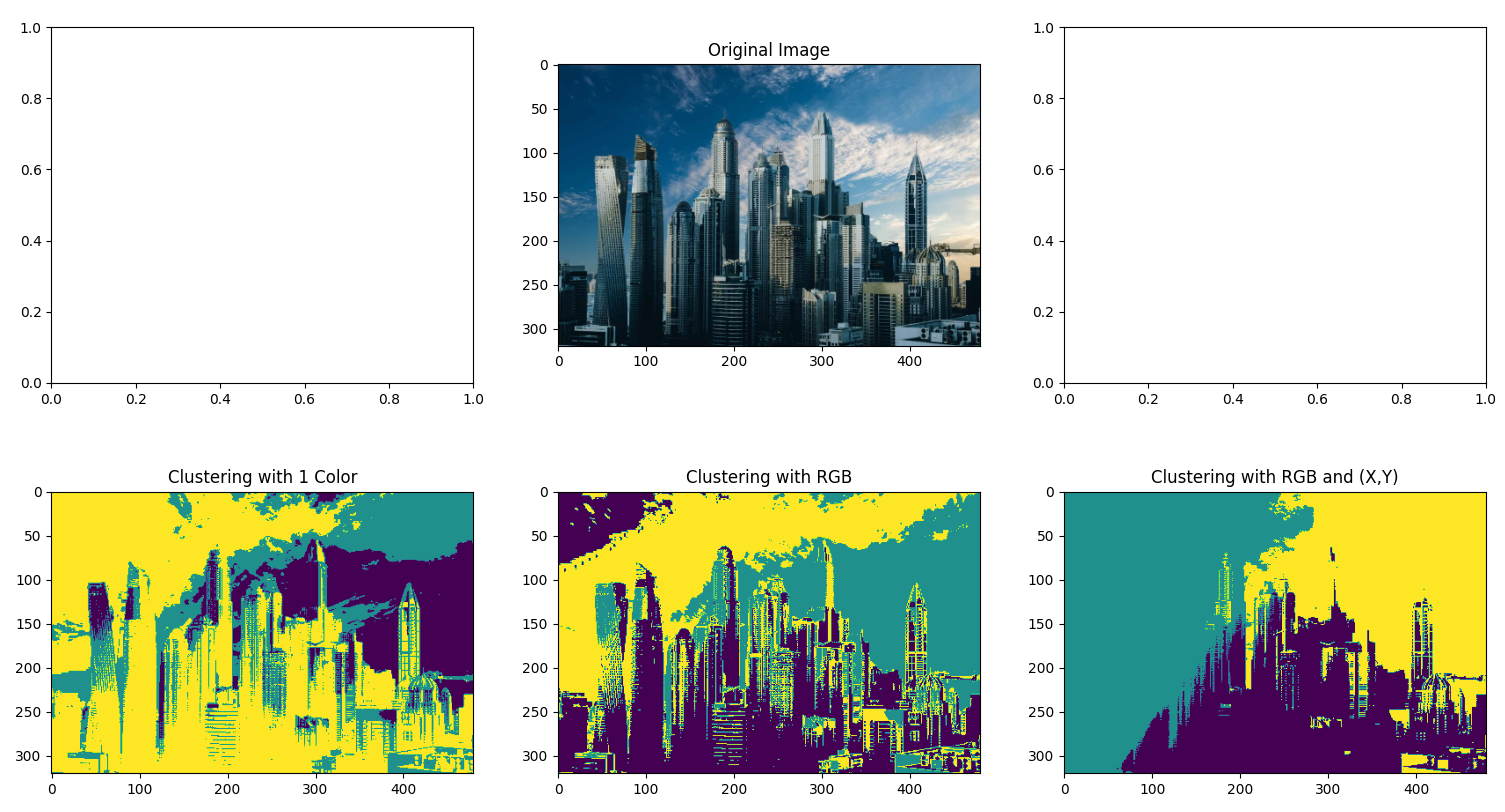
**Sample output**

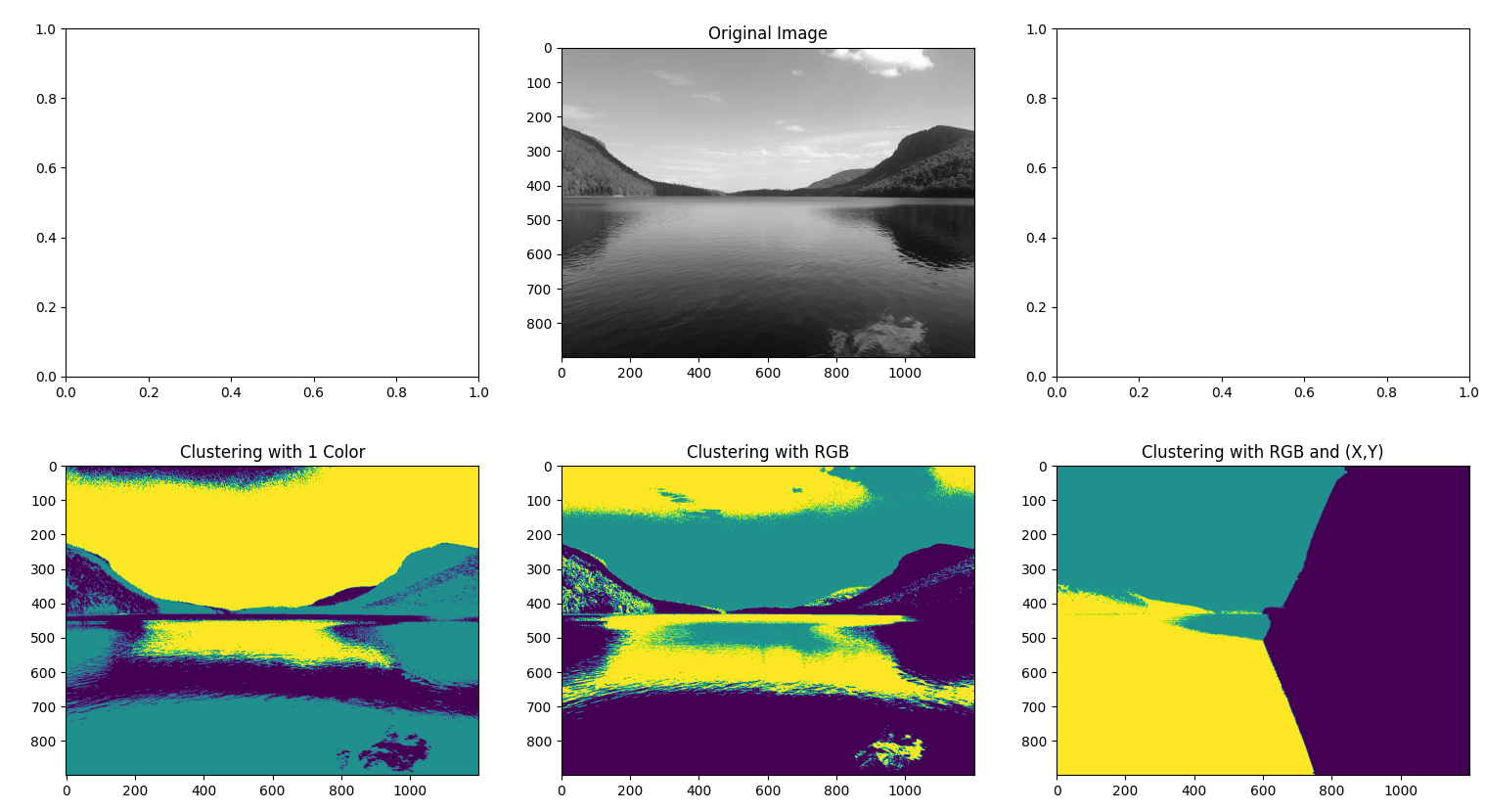




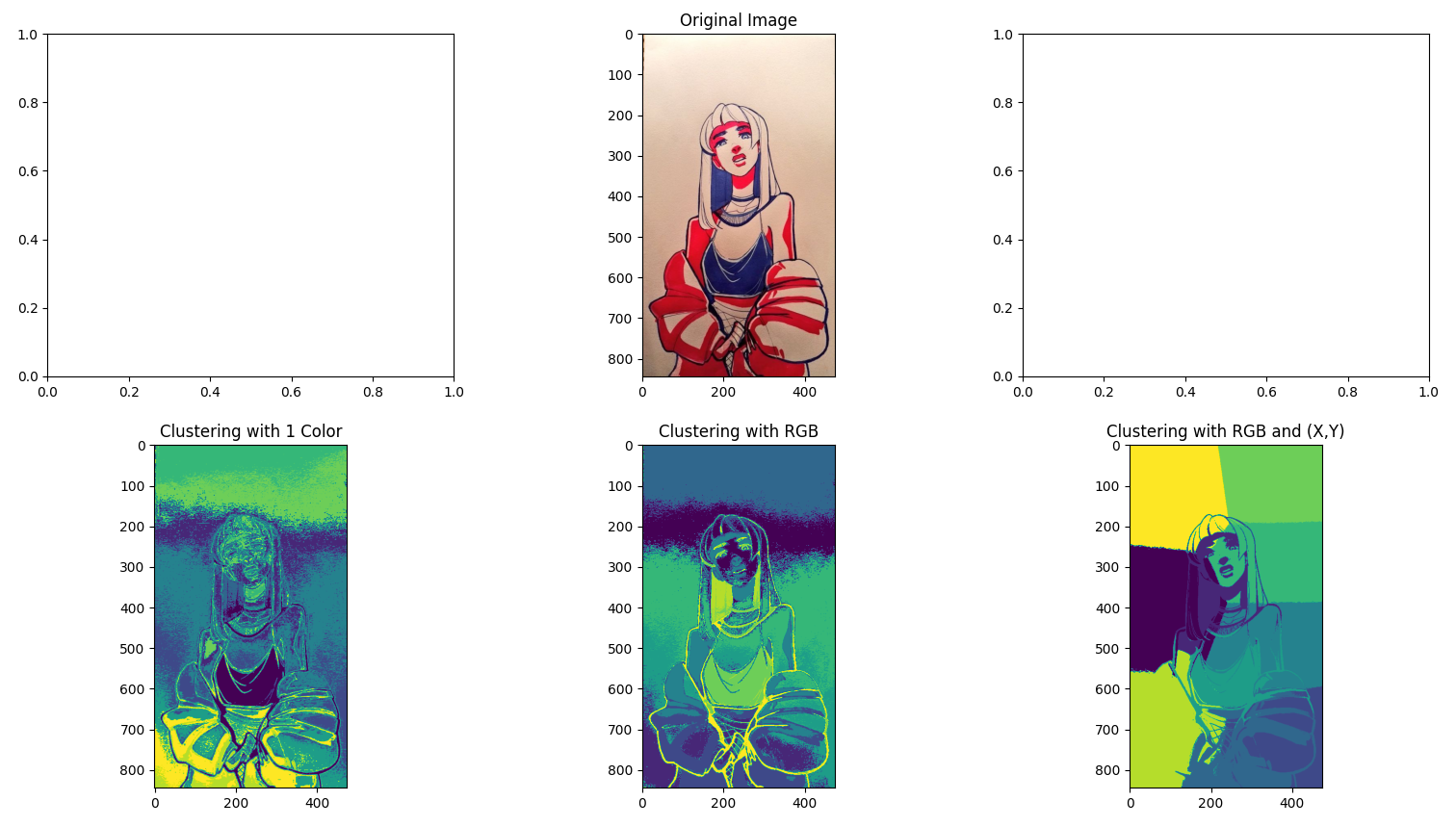
**1D, 3D, and 5D clustering with 3 clusters**

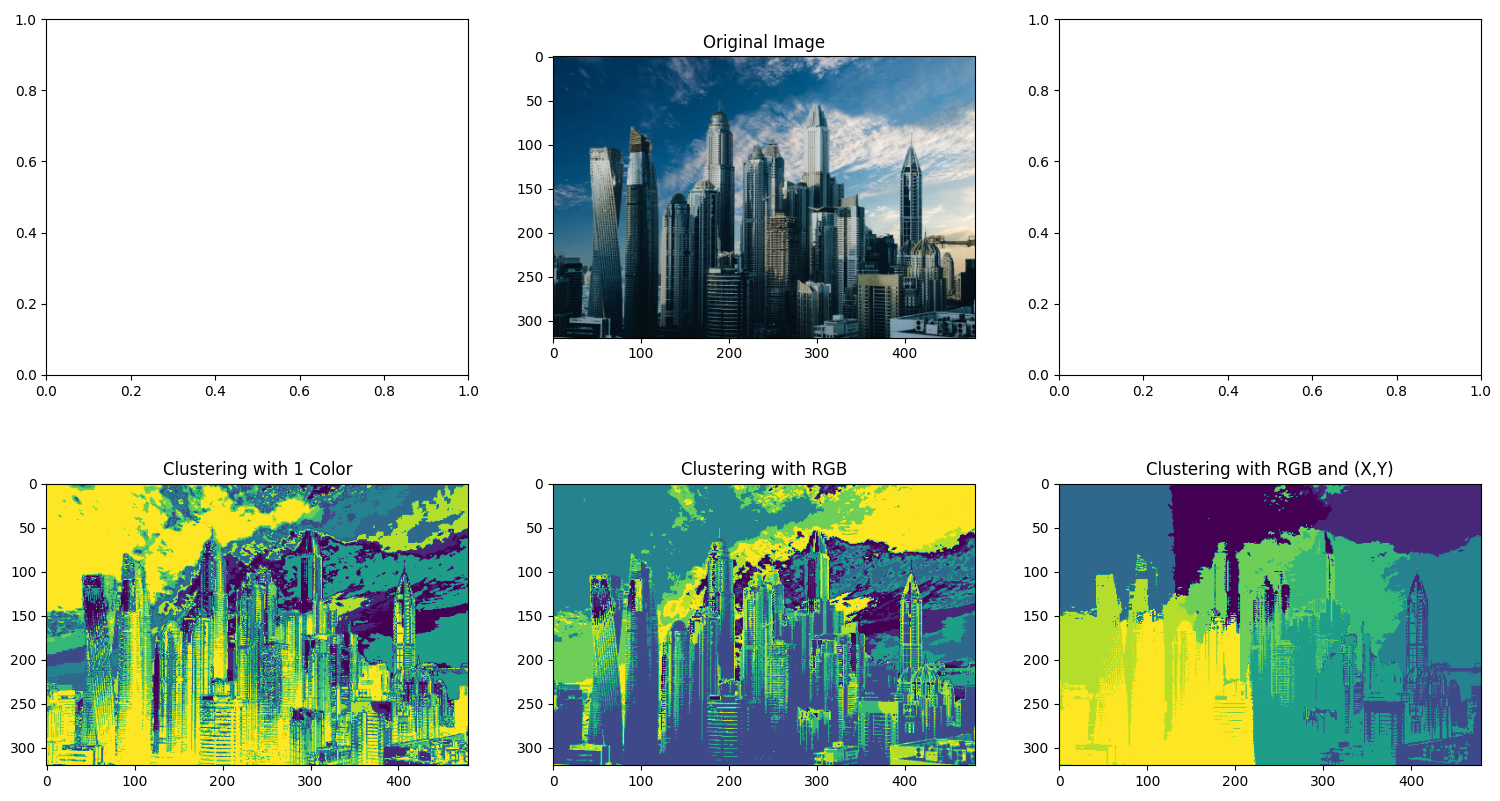


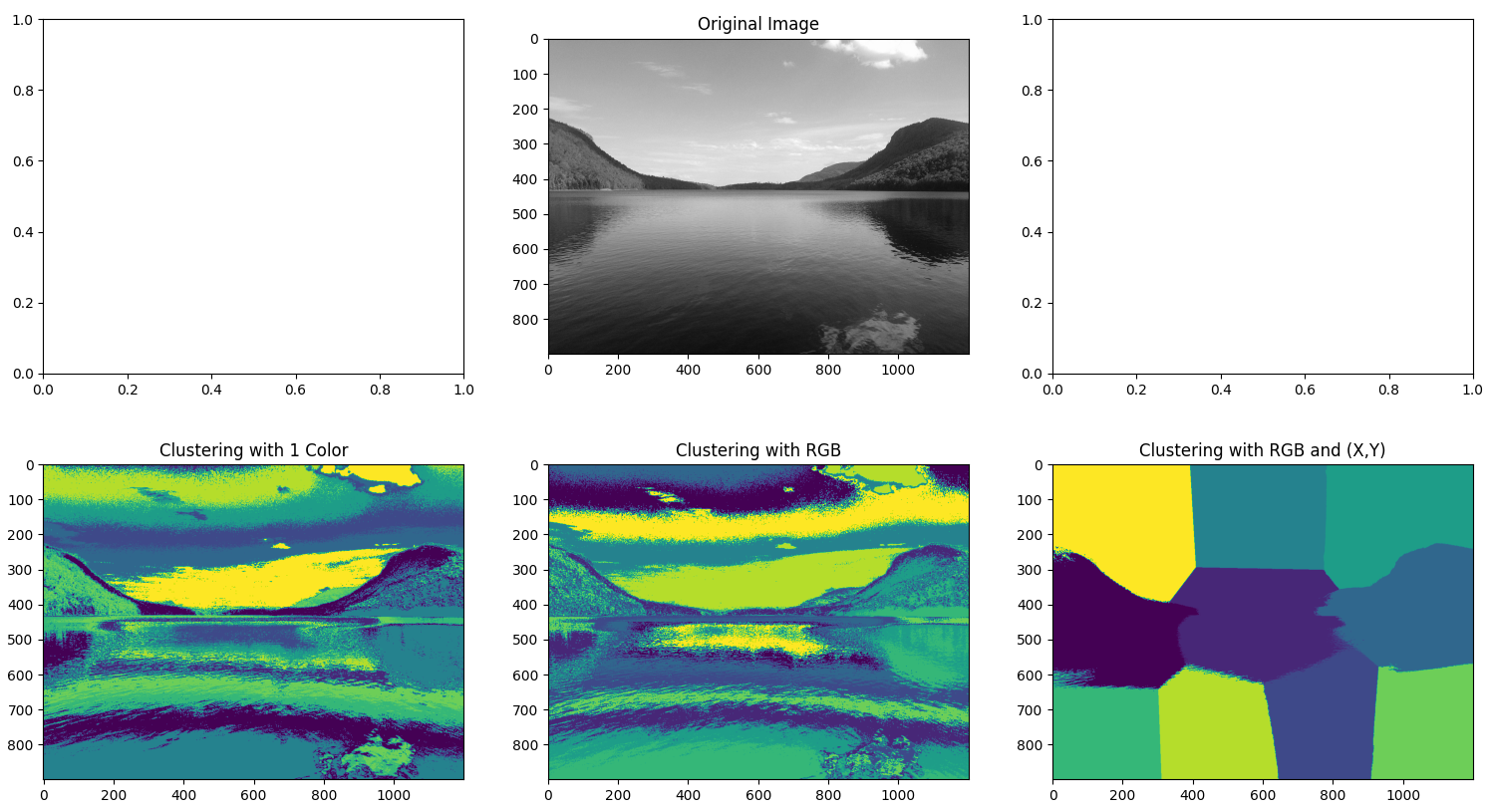




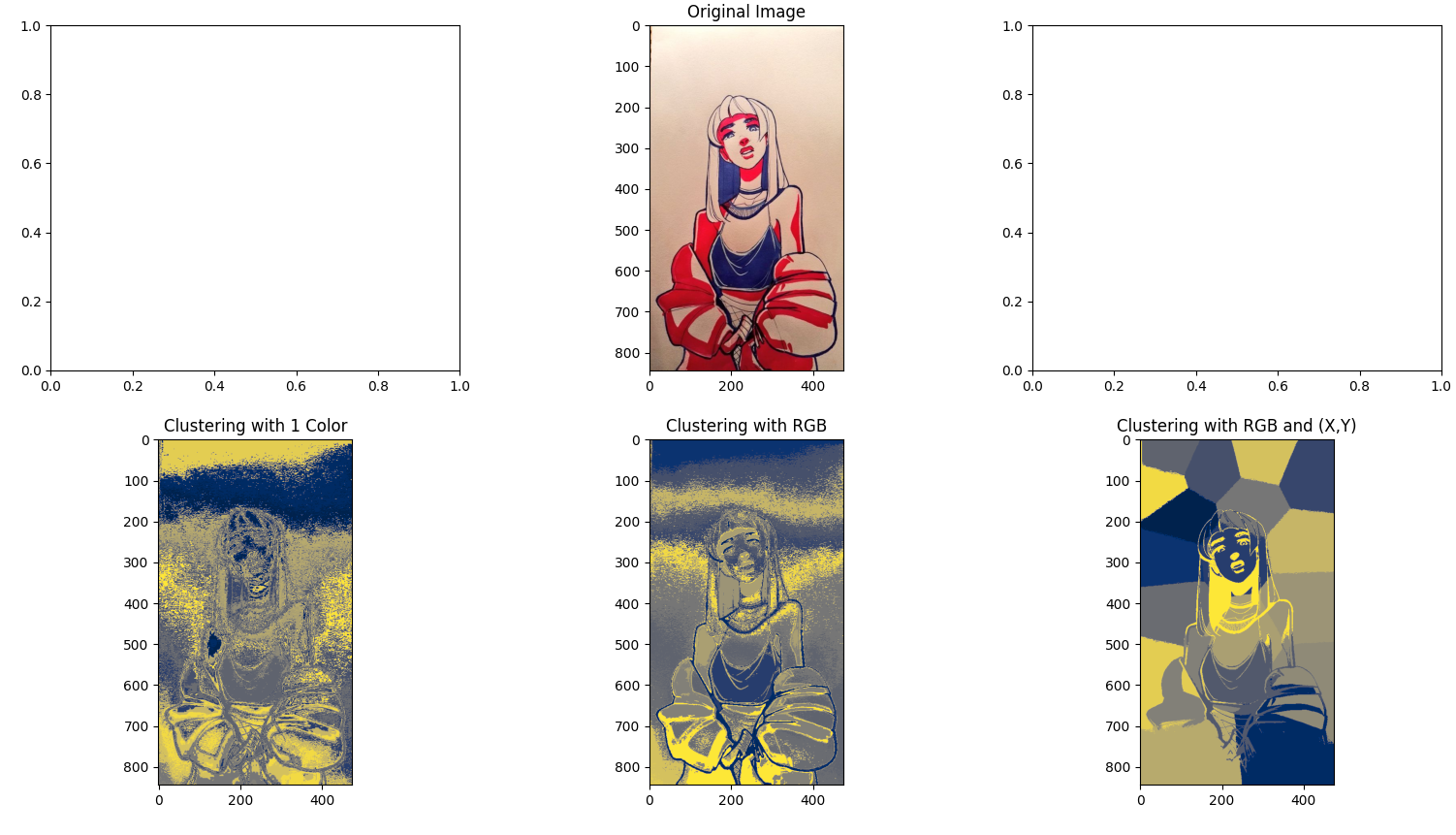
**1D, 3D, and 5D clustering with 10 clusters**

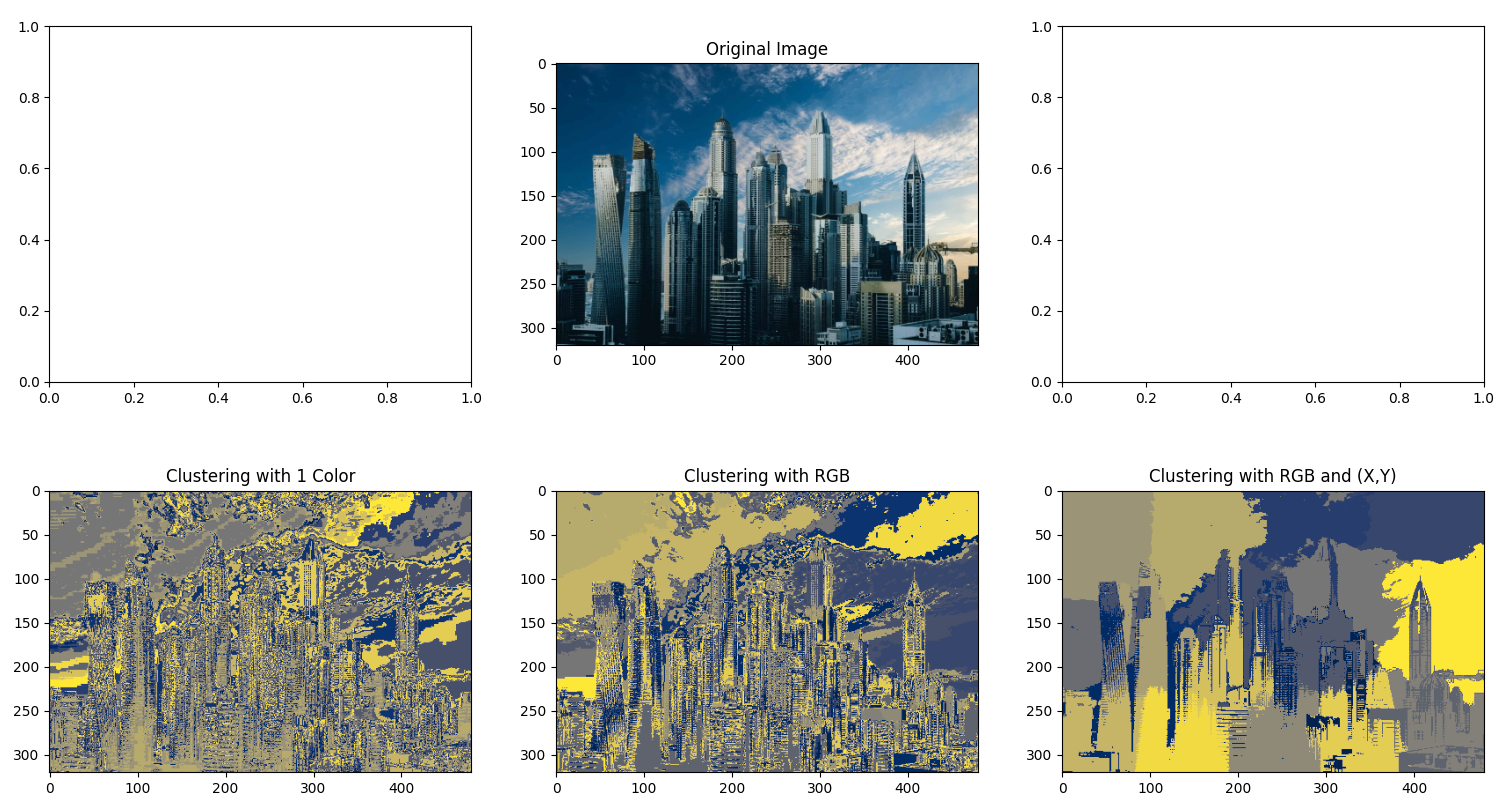


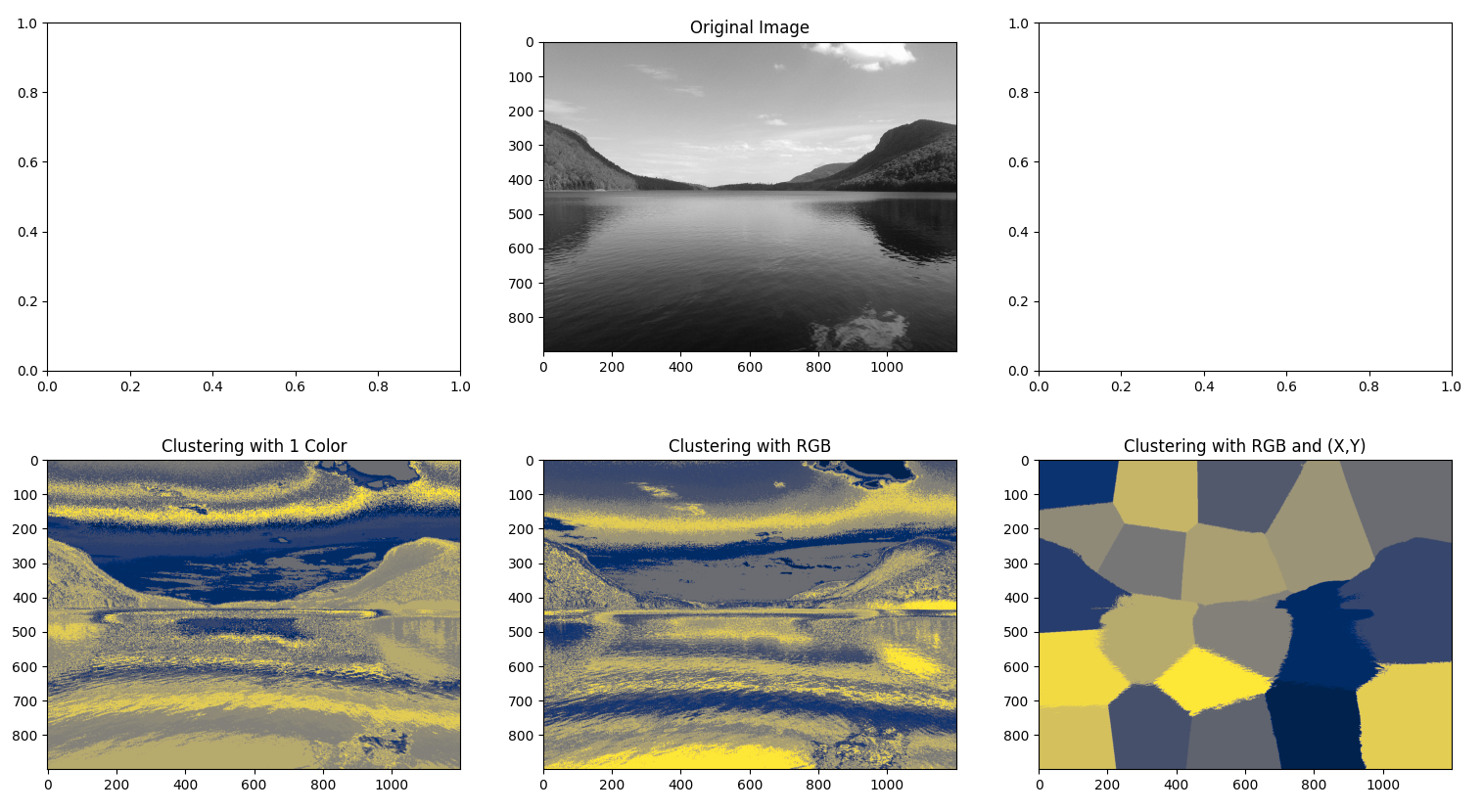




**1D, 3D, and 5D clustering with 20 clusters**







**Discussion about K-means convergence**

K-means will diverge and fail to cluster data if the number of clusters (K) is larger than the unique number of data points in the data.