

Hand-in Lab02 MLDM

Original Image



Compressed Image



Code

```
# Download image
wiki_image = download_img(img_url)
# Call img2arr
arr_original = img2arr(wiki_image)

# reshape array
arr =
arr_original.reshape(((arr_original.shape[0]*arr_original.shape[1]),arr_original.sh
ape[2]))
print(arr)

#do kmeans with different values and plot the elbow graph
distortions = []
iterations = [2, 4, 6, 8, 16, 32, 64, 128, 256]
for i in iterations:
    k_means = KMeans(n_clusters=i, random_state=0).fit(arr)
    distortions.append(k_means.inertia_)

plt.plot(iterations, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')

#as we can see the fifth value seems to be the best one which is 16. Let's do it
again with 16 clusters.

k_means = KMeans(n_clusters=16, random_state=0).fit(arr)

label = k_means.predict(arr)

# find the centroids
centers = k_means.cluster_centers_
# Call chroma plot to see if we're on the right track
rg_chroma_plot(arr_original, centers)
# call replace nearest color
```

```
new_colors = replace_nearest_color(arr_original, centers)
print(new_colors)
# call arr2img
arr2img(new_colors)
```

Comment

The clustering was verified by using the elbow method. I plotted the distortion using different number of clusters (between 2 and 256) and decided that 16 is the appropriate number.