VIETNAM NATIONAL UNIVERSITY
UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY

REPORT

Course: Applied Mathematics and Statistics

Project 1: Color Compression

Class: 22CLC08

Student: Đinh Vũ Gia Hân
ID: 22127098

Intructors: Vũ Quốc Hoàng
Phan Thị Phương Uyên
Nguyễn Văn Quang Huy
Nguyễn Ngọc Toàn

Ho Chi Minh, 2024

Contents

I. Student's information

- ID: 22127098.
- Name: Đinh Vũ Gia Hân.
- Class: 22CLC08.

II. Implementation idea and function description

1. Implementation idea

This implementation idea for K-means function is opitmized by using algorithm of Dr. J Rogel-Salazar. [1]

- Step 1: Define the number of clusters.
- Step 2: Inititate the centroids.
- Step 3: Calculate distance.
- Step 4: Assign centroids.
- Step 5: Update centroid location.
- Step 6: Repeat steps 3-5.

2. Function description:

2.1. Library:

- Mumpy is used to calculate matrices.
- PIL is used to read and write image.
- Matplotlib is used to display image.

2.2. Read image function:

- Input: Iamge path.
- Description: This function is used to read an image. First, it opens the image using the `Image.open` function from PIL library. If successful, a 2D image object (`img_2d`) is created. If the image path doesn't exist, or image format is not supported by Matplotlib or PIL, or any other exceptions, print error messages. Add the first subplot to the figure using `plt.subplot(1,2,2)`. Set the title of the subplot to "Orginal Image" and show it. Turn off the axes (x and y labels) of the plot and adjust the spacing between subplots. After successfully reading the image, this function return a 2D Numpy array.
- Output: 2D image

2.3. Show image function:

- Input: 2D Image.
- Description: This function is used to show an image. It adds the second subplot to the figure using `plt.subplot(1,2,2)`. Set the title of the subplot to "Compressed Image" and show it using `imshow()`. Turn off the axes (x and y labels) of the plot and adjust the spacing between subplots.
- Output: No output.

2.4. Save image function:

- Input: 2D Image, Image path.
- Description: This function supports 3 saving formats 'JPG', 'PNG' and 'PDF'. First, convert 2D image into a Pilow image object assuming the format is RGB. Split the image path at the dot to separate the original image name and create a new name for new image. Save the new image right in the same folder as the original image with the selected format.
- Output: No output.

2.5. Convert image to 1D image function:

- Input: 2D Image.
- Description: This function converts the 2D image object (`img_2d`) into a Numpy array (`img_1d`). Extracts the image dimensions from the `img_1d.shape`. Rreshape `img_1d` from it's original 2D shape in to a 1D array and return 1D image.
- Output: 1D Image.

2.6. Kmeans function:

- Input: 1D Image, Number of clusters, Max iterator, Method used to initialize the centroids.
- Description: This function is used to group the pixels. First, it creates a random array with each element in this array representing a random color value within the range [0,255] or random pixels from the image. The loop iterates for a maximum of `max_iter` times. Calculate the distance between each pixel in image and each centroid. `distances` is a 3D array created using `centroids[:, np.newaxis]` where `distance[i,j]` the distance between the i-th pixel and the j-th centroid. Then, it finds and stores the index of

the closest centroid for each pixel in labels. Loop to calculate cluster means. Calculate the average color value for the selected pixels in each cluster. If thera are no pixels assigned to a particular cluster, it appends a zero matrix to means. Check if the labels haven't changed. If there's no change, it means the algorithm has converged and the loop can be stopped. Update centroids by looping through each cluster. If the mean for cluster i is not empty, it update the corresponding centroid with the calculated mean value. Then store the previous labels and countinue the loop until it done. Finally, this function return `centroids` – the K-means cluster centroid and `labels` – the cluster labels for each pixel in the image.

- Output: Centroids array, Labels array.

2.7. Generate image function:

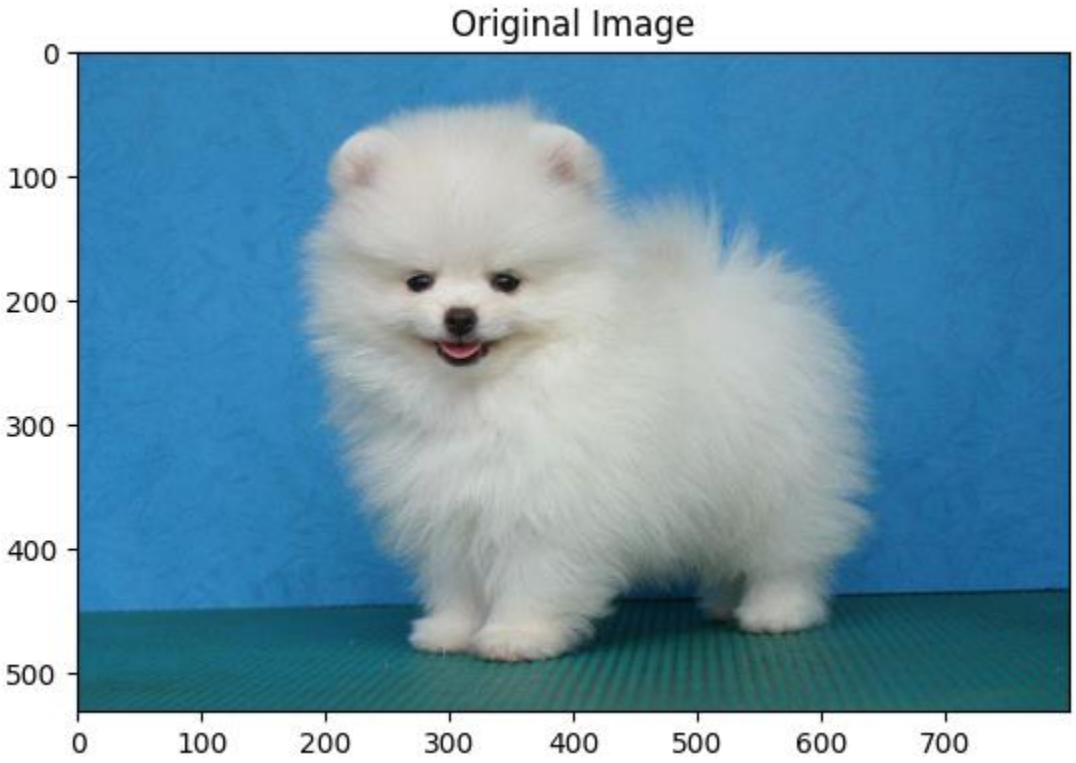
- Input: Shape of image, Centroids array, Labels array.
- Descripton: This function is used to create a image. First, it extracts the image dimensions from the `img_2d_shape` and create a zero matrix to hold the final image data. Loop through each centroid and reassign labels to the pixels. Convert the data type to "uint8" (data type for image pixel values). Reshapes the flattened array back to the 2D image and return it.
- Output: 2D Image.

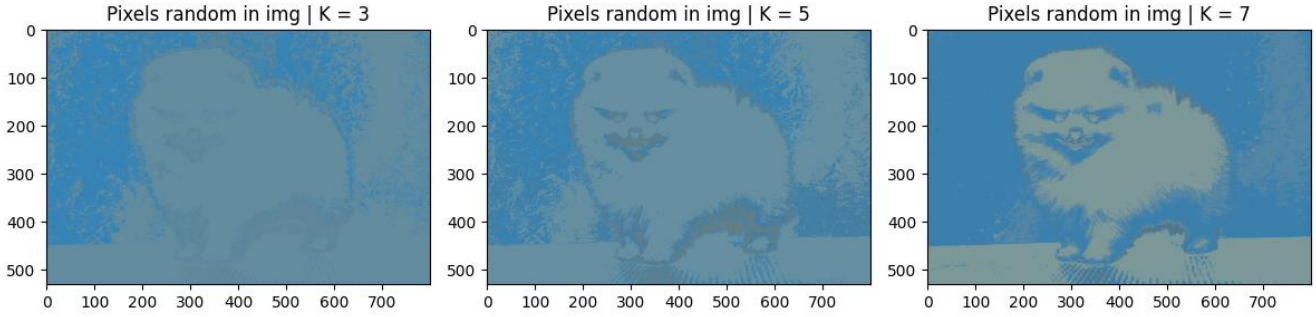
2.8. Color compression function:

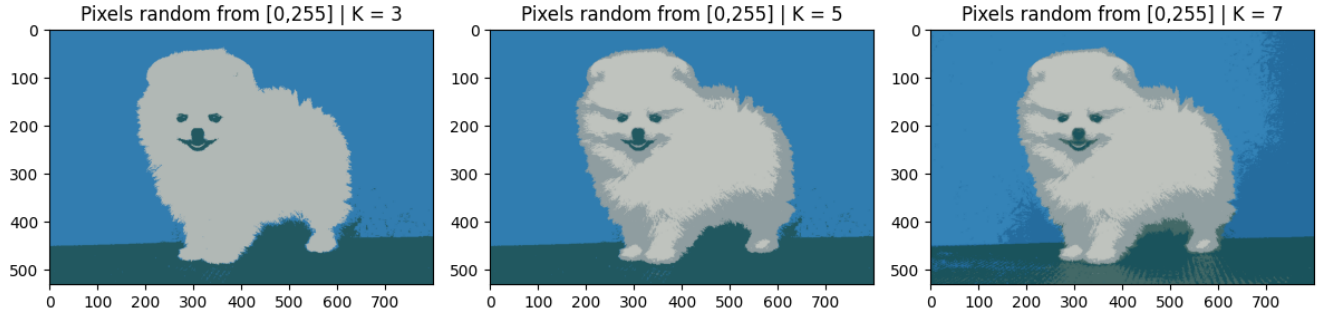
- Input: Image path, Number of clusters, Max iterator, Method used to initialize the centroids.
- Description: This function is used to compress color. It checks input validation. Read image and convert it to 1D image. Group the pixels and convert it into image. Show image and save it in the same path with original image.
- Output: No output.

III. Demo result and Comment:

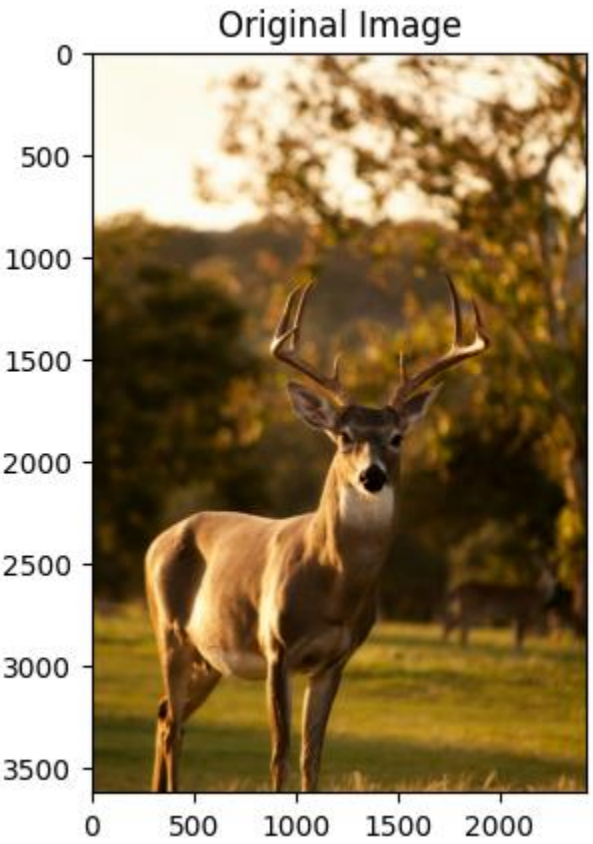
1. Example image 1:

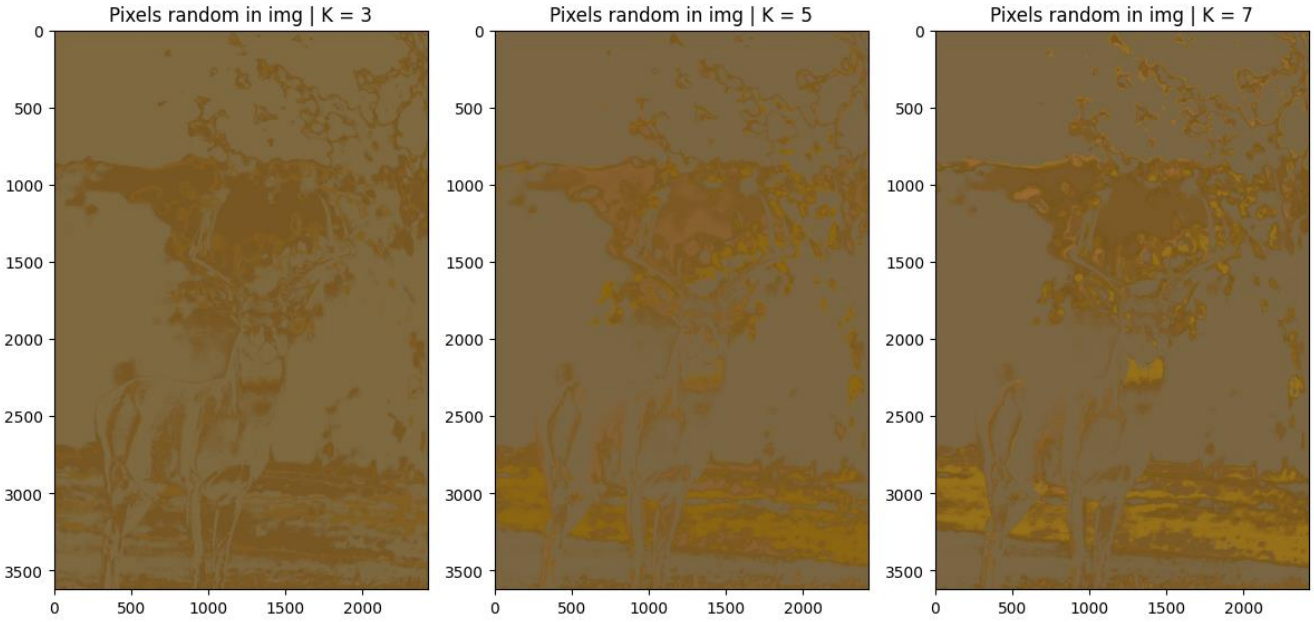


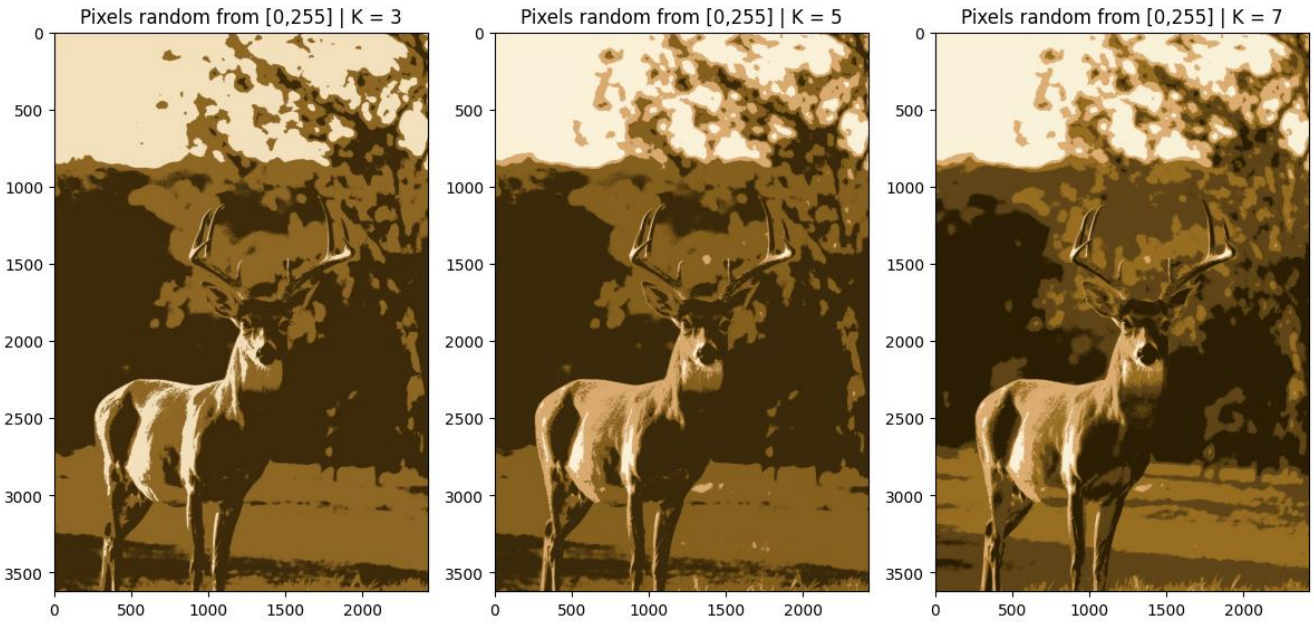




2. Example image 2:







3. Comment:

- In general, the demo indicate a good result. They are approximately as effective as scikit-learn's K-means. However, image compression performance for small images (compressed - color reduced with other software) or monochrome images often gives poor results with small max_iter.
- When you plot the convergence of Kmean, when init randomly in range [0, 255], the convergence speed will be faster than init with random pixels in image. In case the image has many colors, randomization within pixels will bring better results and be more similar to the original image. In another hand, when the image has few colors, init centroids with random value will bring better results and as same as the original image.

VI. References:

[1]: https://domino.ai/blog/getting-started-with-k-means-clustering-in-python, 16/6/2024

[2]: https://github.com/NgocTien0110/Applied-Mathematics-and-Statistics, 16/6/2024

[3]: https://github.com/kieuconghau/color-compression/blob/master/Source, 16/6/2024

[4]: https://www.reddit.com/r/learnpython/comments/13vs2fu/need_help_to_understand_the_notation_x_npnewaxis/?rdt=44936, 16/6/2024