

Project 1 Report

Anuar Maratkhan

I. INTRODUCTION

This is the report for ROBT 310 Image Processing course project 1. Main deliverable of the project is to write a MATLAB program that will create a mosaic from an image assembled from images in the dataset.

II. THE WORK DONE

I have decided to compare means of whole image with batch (first method) rather than comparing pixel by pixel which require more computational resources. The comparison was done using Euclidean distance formula:

$$\text{dist}(P, Q) =$$

$$\sqrt{\sum_{i=0}^n (P_i - Q_i)^2} (1)$$

First, I started by generating batches of certain size without changing the size of actual image so that size of rectangles perfectly fit to the image. The generated batches were then checked if they were computed right by generating colored batches that actually just lowered the image resolution. Further, I have been using two small datasets for replacing those colored batches by rectangular images. Later, after I have finished that part I started adjusting the size of the rectangle so that the user can input the size of batch at runtime. At the same time I have been refactoring my code such that it produces the other image of different size if size of all batches combined does not produce the same sized image. That took me a lot for computing and understanding math and linear algebra in particular. As soon, as I have finished the adjustable size part, I have started thinking of how to produce the different shape and reminded myself the method from lecture 7 where we used masks to produce desired image. So based on that I have taken a mask image and converted it to logical matrix which produced an array of 0s and 1s. Further, I typecasted it to 8-bit integer values to element-wise multiplication to batch image. That produced me a circled image in rectangle where the image was inside the circle and the rest was black. Moreover, I had to fill that black space with something. I had an idea to use transparent image but it required me to call `imshow` function each time when the batch image is chosen. Thus, I have chosen to fill it with the right RGB colors that are actual mean RGB for that batch. As a consequence, this produced me a better mosaic with well noticed circles (due to little gradients). Next, I had a problem of circles with wrong quality. That was due to low resolution of database of images which I have resized after importing them for lower computational memory usage. After playing with the size of image to be resized I have decided to use size [100 NaN] because of golden middle quality and memory usage. However, I have many unnecessary variables that I could prevent using, which may then require more RAM. Even though, the program works fast

enough (because of computing mean of whole image rather than of each pixel). After I have finished all of that stuff, I started refactoring my code so that anyone who will use the program, will have to call just one function with parameters provided. In addition, the way I have produced the dataset is using `ffmpeg` program which is also easy to use. I have chosen to use movie 'Zootopia' which I thought would be colorful enough for the mosaic. I have generated more than 700 images (one frame each 8 seconds). Something Linky

III. USAGE

The usage of a function I have implemented is simple enough. We just need to provide the name of a file (full, with extension like .jpg) that stores an image we want to convert to mosaic, the path to a directory with dataset of images we want to use as batches, the size of each batch (e.g. 16 - single integer less than size of image), the shape of each batch like 'rectangular' or 'circular' which I have chosen to implement. The dataset I have used is stored in Google Drive. The actual large dataset is named 'dataset', and the other smaller two datasets are 'logos' and 'single color'. Full codes for the project is available in my Github repo.