

PHOTOMOSAIC WITH PYTHON AND OPENCV

Arturo Lopez, arlof95@gmail.com

04/20/2022

Description

The goal of this project is to generate a photomosaic of a given target picture by utilizing a set of pictures.

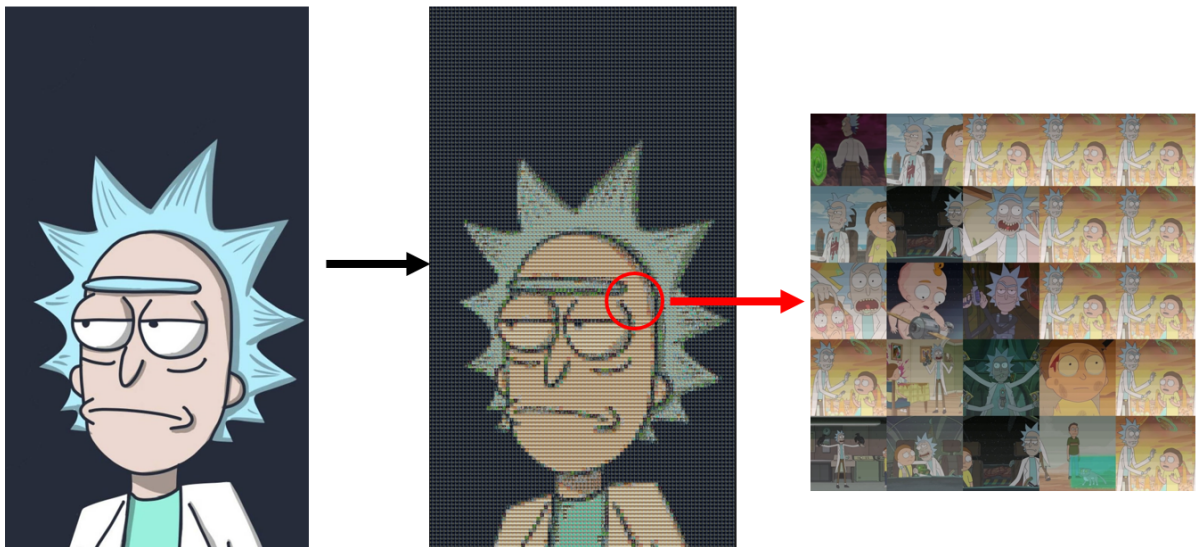


Figure 1: A photographic mosaic of a Rick Sanchez, created using pictures of frames of the TV show "Rick and Morty".

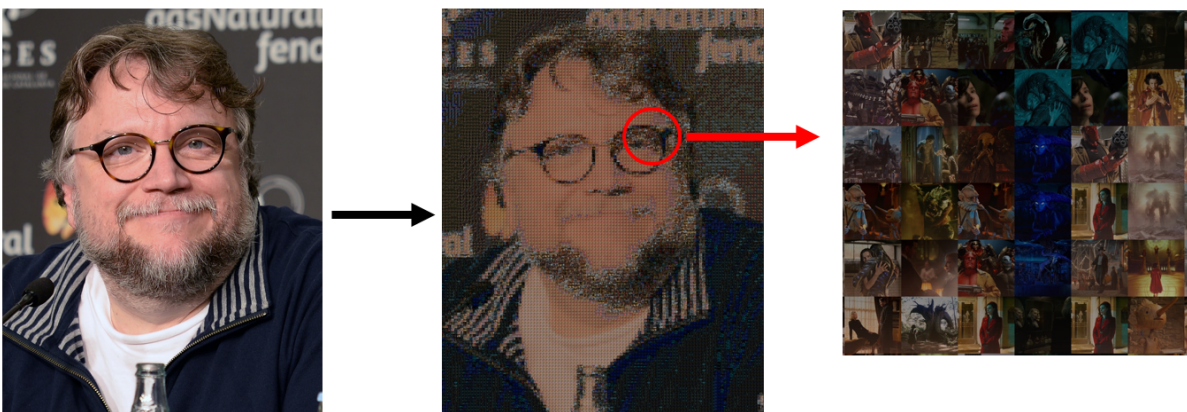


Figure 2: A photomosaic of Academy Award-winning film director Guillermo del Toro, created using pictures of frames from del Toro's filmography.

What is a Photomosaic?

In the field of photographic imaging, a photographic mosaic, also known under the term Photomosaic, is a picture (usually a photograph) that has been divided into (usually equal sized) tiled sections, each of which is replaced with another photograph that matches the target photo. When viewed from a distance, the photomosaic appears to be a single image, but upon closer inspection, the individual tiles become visible, see Figure 1, see Figure 2.

Creating a Photomosaic

Each part of the target image is averaged down to a single color. Each of the library images is also reduced to a single color. Each part of the target image is then replaced with one from the library where these colors are as similar as possible. In effect, the target image is reduced in resolution (by downsampling), and then each of the resulting pixels is replaced with an image whose average color matches that pixel.

Example

For creating a photomosaic, we need a target image, for this example, we are using a picture of Rick Sanchez from the TV show "Rick and Morty", see Figure 3, and a set of images, also known as tiles, for this example in particular we are using 60 frames the episodes of the TV show "Rick and Morty", see Figure 4.

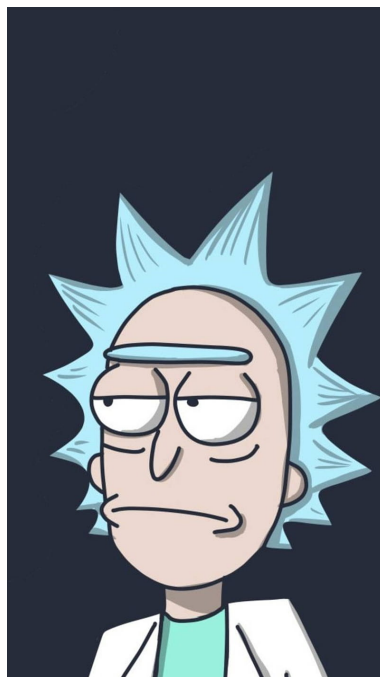


Figure 3: Target image.



Figure 4: Set of images that are going to be used as tiles to create the target image (60 different images in total).

First, we need to get the dominant color on the for each of the tiles images. For doing this, we created a couple of functions, function `getAvgColorSet` will return a list

```

1 # Return a list including all the average colors of the set of pictures
def getAvgColorSet(totalNumberSet):
3     colorsListSet = []
    colorsListSet.append(0)
5     dirSet = "E:/Projects/photoMosaicGenerator/set_Images_rickandmorty/r_m_"
    for i in range (1, totalNumberSet+1):
7         img = cv2.imread(dirSet+ str(i) + ".jpg")
        img = squareImg(img)
9         colorsListSet.append(getAverageColor(img))
    return colorsListSet
11
# Return the dominant color in rgb_to_hex of a single image
13 def getAverageColor(img):
    avg_color_per_row = np.average(img, axis=0)
15     avg_color = np.average(avg_color_per_row, axis=0) # blue, green, red
    blue = int(round(avg_color[0]))
17     green = int(round(avg_color[1]))
    red = int(round(avg_color[2]))
19     color = rgb_to_hex((red, green, blue))
    return color

```

Listing 1: `getAllDominantColots` function.

Image	Dominant RGB
Image 1	RGB 1
Image 2	RGB 2
Image 3	RGB 3
...	...
Image N	RGB N

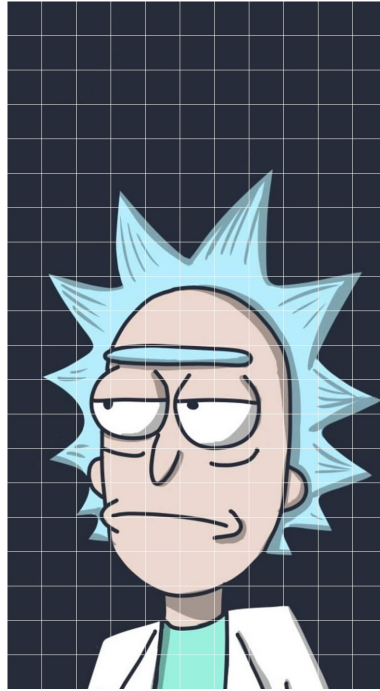


Figure 5: Set of images that are going to be used as tiles to create the target image (60 different images in total).