

Technical task

My task was to implement an image processing program, so that a user could type several requests to process at a time.

These are the options you can choose to convert the picture:

- Mirror
- Rotate
- Inverse
- Lighten
- Darken
- Sharpen
- Convert to black&white

My implementations

I was using only numpy for processing pictures

- 1) For Mirror I applied `numpy.flip()`, that flips the matrix
- 2) For Rotate I used `np.rot`, and because I was supposed to rotate picture to the right and `np.rot` rotates to the left, I passed a 3 as a second parameter
- 3) For Inverse I simply equated value of every pixel to $255 - \text{old_value}$, because value of pixel in the picture can only be from 0 to 255
- 4) For Lighten and Darken I was inspired by OpenCV Gamma Correction (<https://www.pyimagesearch.com/2015/10/05/opencv-gamma-correction/>). I used this formula $O = I ^ (1 / G)$, where I is input image, G is gamma and O is output image scaled back to $[0, 255]$. If Gamma is 1, then output == input. Gamma values < 1 will shift the image towards the darker end of the spectrum while gamma values > 1 will make the image appear lighter.
- 5) For sharpen I used padding method with kernel
- 6) For converting to black and white `np.mean`, regarding the color axis and I get a 2D image at the end

Last thoughts

As I found out during making this program, for some actions, such as mirror, rotate and inverse, there is only one way possible to implement them, while others have several solutions. I found out three possible ways to black&white picture on this site

https://brohrer.github.io/convert_rgb_to_grayscale.html for example. For darkening and lightening I also found a lot, but I really liked a gamma correction idea, because it is kinda simple and work fast with numpy array.