1. Source Code is attached.
   * 1. Description: Project1Q1.py is program 1, and Project1Q2.py is program 2.
2. Explanation:

Program1:

I use OpenCV to manipulate the image.

The image data read by OpenCV is BGR, and in range [0, 255]. As I try to convert it to LUV color space, OpenCV provide a function cvtColor to do so. However, OpenCV does not the the BGR is 8bits or 16bits integer, so I have to normalize them to [0, 1], and convert them.

In LUV color space, I find the smallest and largest L value in the windows, and applied linear scaling to it. So for L values inside the window, they will be scaled between the smallest and largest value. For any L value outside the window, map the value to 0 or 100 if they are smaller of larger than the range inside. To produce the new image, I convert it back to RGB and change to range back to [0, 255].

Program2:

Do the same to convert the image to LUV color space, this time, I first calculate the histogram, round the min and max value and change them to integer. After that, calculate the cumulative value of the histogram from min to max L value inside the window. And then applied the histogram equalization formulas, I get a mapping from the original L value to equalized L value in range of min to max. For L values inside the min and max range, map them to new value, for L values outside the range, map them to either 0 or 100.

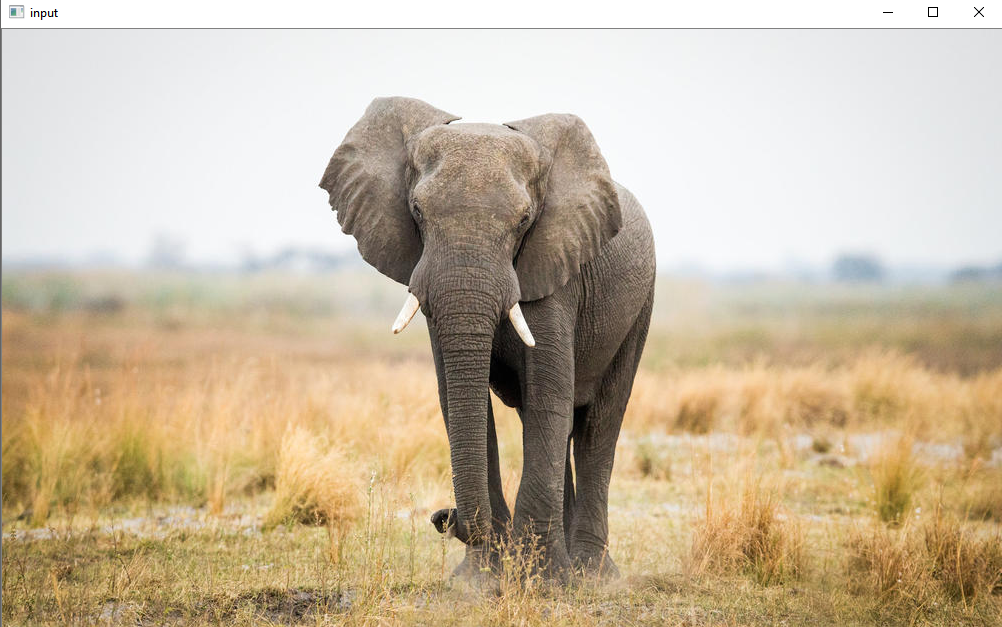
Finally, do the same as program 1 to get the new image.

1. Result.

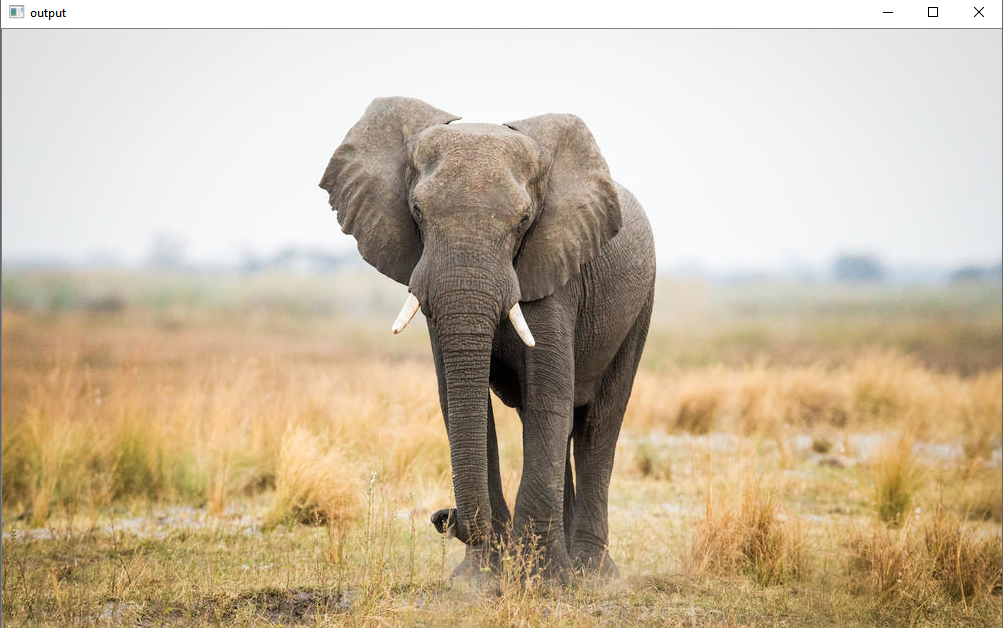
Program1:

Window: 0.3 0.3 0.7 0.7

input



Output

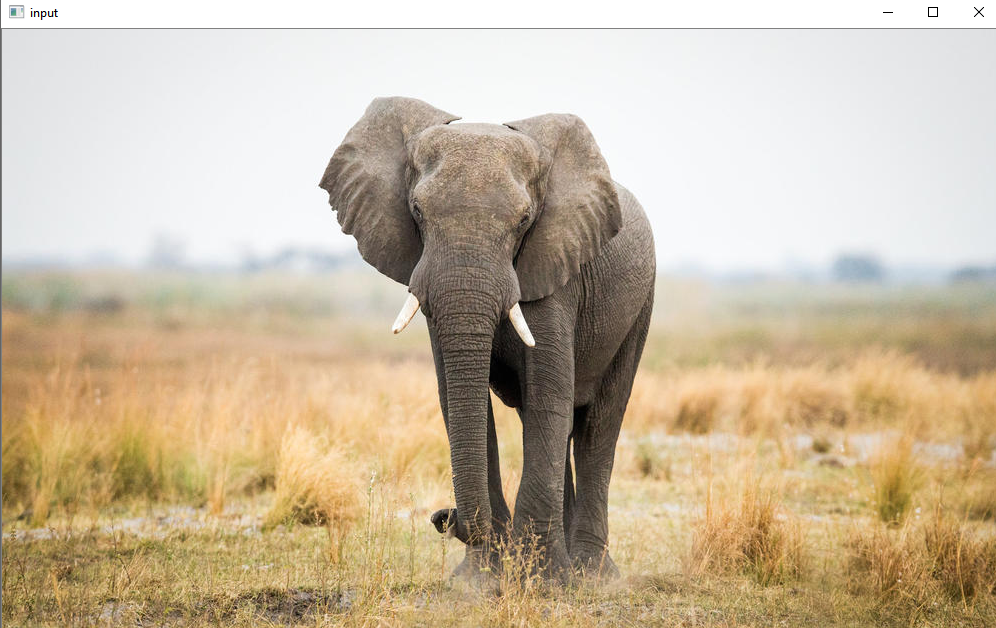


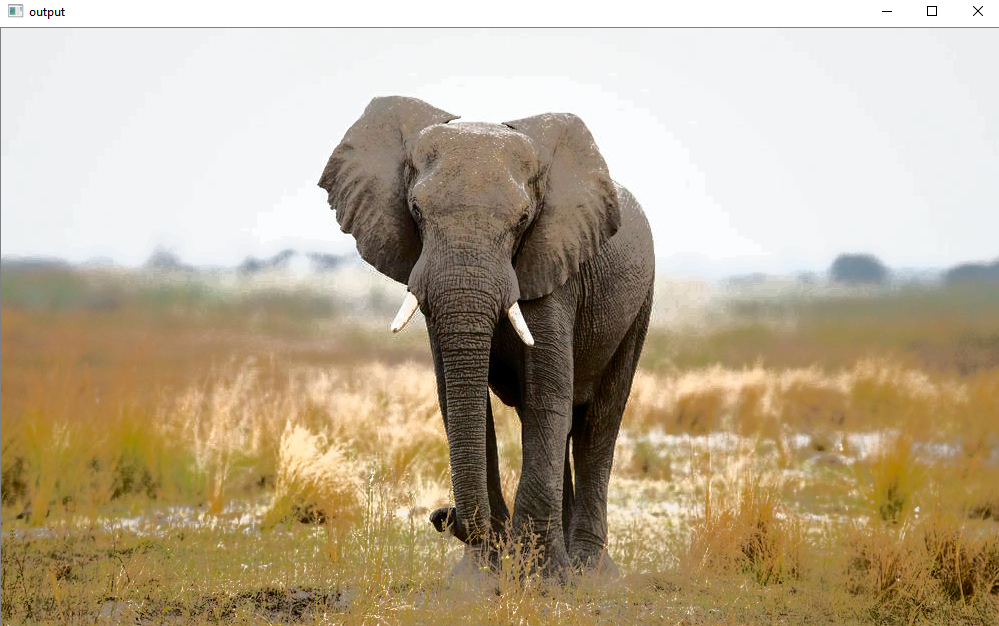
The light in middle is slightly improved, but that is very marginal.

Program2:

Window: 0.3 0.3 0.7 0.7

Input:





This program improves a lot. The middle part of the image is very clear in light.

In the scenario that either the selected window is too small, or the image is too bright or dark in some part, and the range between min and max is too small. Then the image will look “bad”. Because the light will be scaled too much.

Input:



Output:

