Histogram Equalization: Histogram Equalization (HE) is a statistical approach for spreading out intensity values. In image processing, HE is used for improving the contrast of any image, that is- to make the dark portion darker and the bright portion brighter.

Simple Histogram Equalization adjusts the global contrast of an image by updating the image histogram's pixel intensity distribution.

1) Script:Import necessary packages and declare image path

```
# import the necessary packages
import argparse
import cv2

img_path = "images/einstein.jpg"
```

2) Script:Load image and convert to grayscale

```
# load the input image from disk and convert it to grayscale
print("[INFO] loading input image...")
image = cv2.imread(args["image"])
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
```

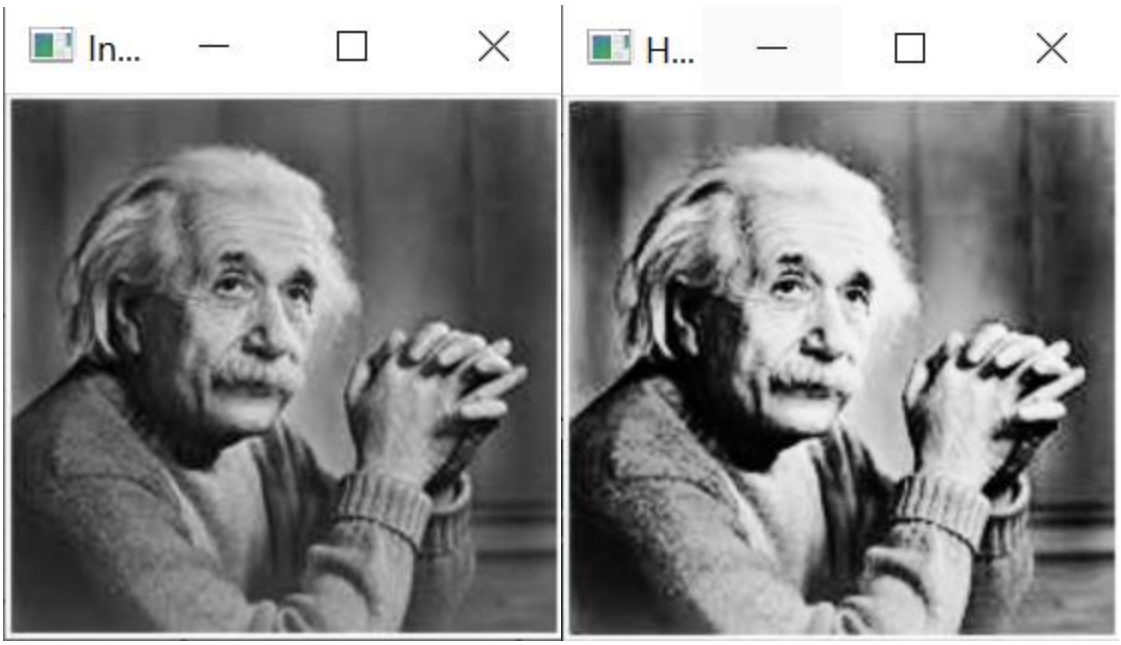
3) Script :Perform simple histogram equalization

```
# apply histogram equalization
print("[INFO] performing histogram equalization...")
equalized = cv2.equalizeHist(gray)
```

4) Script:show the normal image and the image formed after histogram equalization

```
# show the original grayscale image and equalized image
cv2.imshow("Input", gray)
cv2.imshow("Histogram Equalization", equalized)
```

Result:



Grayscale Image Equalized Image

Adaptive Histogram Equalization:it improves the local contrast and enhances the definitions of edges in each region of an image.

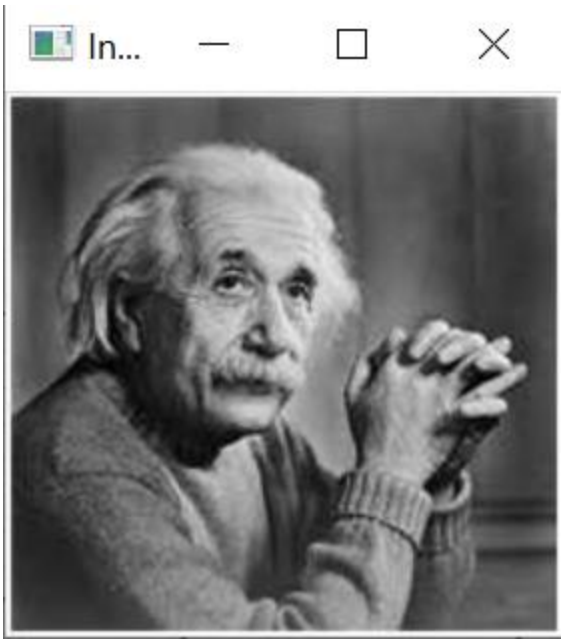
Starting from the end of script 2 in simple histogram equalization

3) Script:Instantiating the adaptive histogram equalization function and applying it to grayscale image

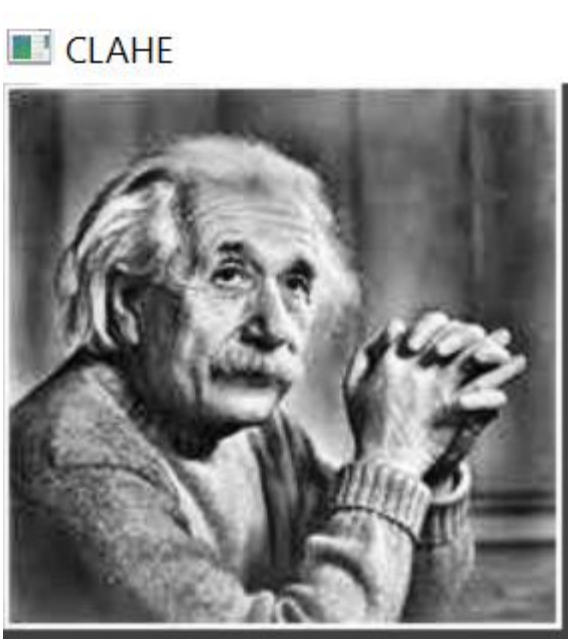
```
# apply CLAHE (Contrast Limited Adaptive Histogram Equalization)
print("[INFO] applying CLAHE...")
clahe = cv2.createCLAHE(clipLimit=args["clip"],
    tileGridSize=(args["tile"], args["tile"]))
equalized = clahe.apply(gray)
```

4) Script:

```
# show the original grayscale image and CLAHE output image
cv2.imshow("Input", gray)
cv2.imshow("CLAHE", equalized)
```



Grayscale Image



CLAHE Image