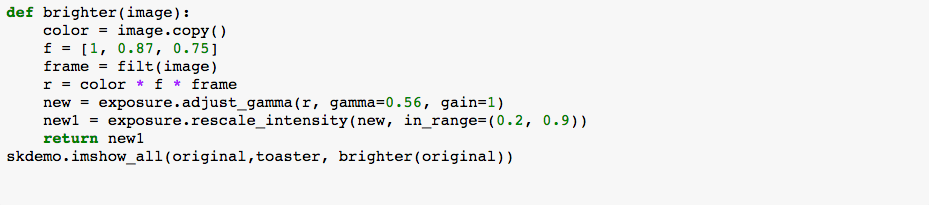
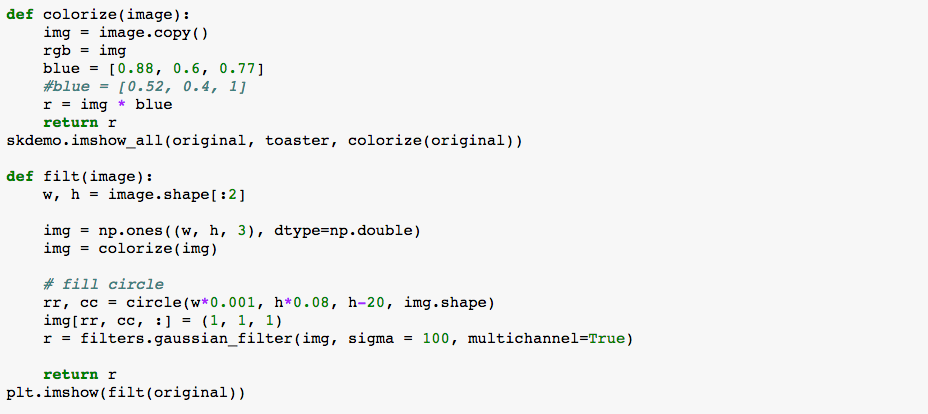
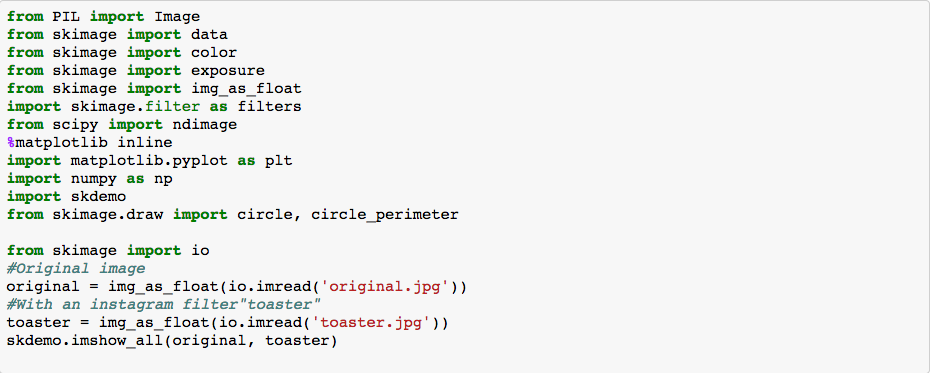
Assignment 3

Analyses and Re-creation of Toaster Instagram filter

Anh Tran

**Abstract**

Filters are one of the main highlights of Instagram. No matter who you are, an amateur iPhone photographer or a professional, you can always enlighten your photos using them. Due to their popularity, a lot of people are curious how these filters were actually created, what filters actually do to an image and what is the secret “recipe” each filter has. This paper is going to analyze the “Toaster” instagram filter and try to propose a code that replicates that filter. A lot of analyzes were made throughout the research process, such as comparing histograms, threshold, lightness, edge, red intensity. After comparing those features between original and filtered images, I found out that the filter has a red vignette effect on the right half. It also makes the image more yellow and brighter in the center. Based on those observations, I re-created the filter using python library Scipy. The replicated filter is not as contrast and bright as the Toaster filter itself, although it’s very close to it.

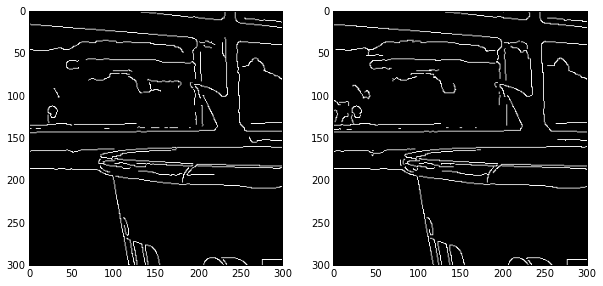
**Code**

**Methods and Results**

*Lightness:* In order to determine how lighter the filtered image was comparing to the original one, I created a new image of differences in pixels of original and filtered images. The darker part of a new image describes when the filtered image is lighter, while the brighter part means the opposite. To replicate that, I adjusted the gamma correction, setting it to 0.56, and increased the contrast with rescale\_intensity in brighter() method. In order to get the vignette effect, I create a circled frame and multiplied picture’s channels with it in brighter() method

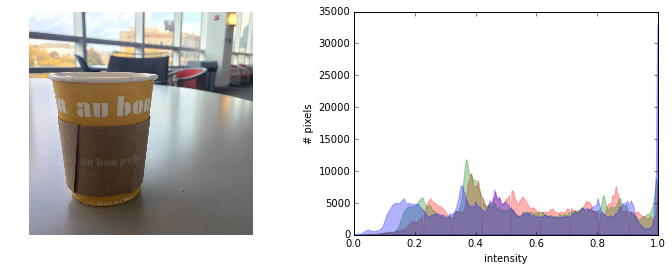
****

*Edge detection:* in order to determine if the filter makes the image blurrier or not, I compared edge detections of both pictures using Canny operation and zoom it to a certain part of the photo in order to see the edges better. Based on the comparison, we can see that the filtered image has slightly more edges that the original one. This means that the filter sharpen an image as well. Although I did not use sharpen operation directly, I did add contrast in brighten() method because I only wanted the picture to be sharpened only for a small amount

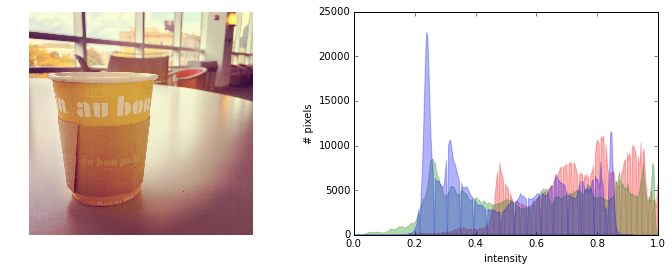
****

*Original Filtered*

*Histograms:* histograms provide a lot of information about the instensity of each channel in a picture. In my case, I plotted the RGB chacnnels of each picture and compared them. The graph shows that the filtered image has much less blue pixels with high intensity while it has a lot of green and red high intensity ones. In order to create this effect, I multiplied RGB channels of an original image with [0.88, 0.6, 0.77] in colorize() for the vignette filter, and [1, 0.87, 0.775] in brighten() for the entire image. As the red multiplier is biggest here, the red intensity will be increased much more than other channels.

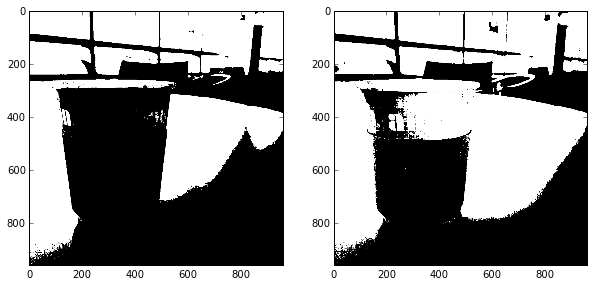
**

*Original*

**

*Filtered*

*Threshold:* threshold is a good tool to compare the brightness of two images. In order to see the which picture is lighter, I took the treshold of the original and applied it on both images. Based on the result, we can see that the filtered image has a bit more white color, especially in the center of the picture.



*Original Filtered*

*Red Intensity:* just in order to see how much the filtered image is redder comparing to the original one, I decided to draw the first channel of each picture. The filtered image is much lighter, which means it has more red pixels comparing to the original one.



*Original Filtered*

**Discussions**

Due to analyses I managed to re-create the filter. Even though it’s not contrast and saturated enough, the new one is pretty close to the original filter

****

*Original Instagram Filter Replicated Filter*

**References**

Burger, Wilhelm, Burge, Mark J. (2009) Principles of Digital Image Processing - Fundamental Techniques**,** p.57.Retrieved from<http://link.springer.com/book/10.1007/978-1-84800-191-6/>

Skimage Documentation. Retrieved from <http://scikit-image.org/docs/dev/>