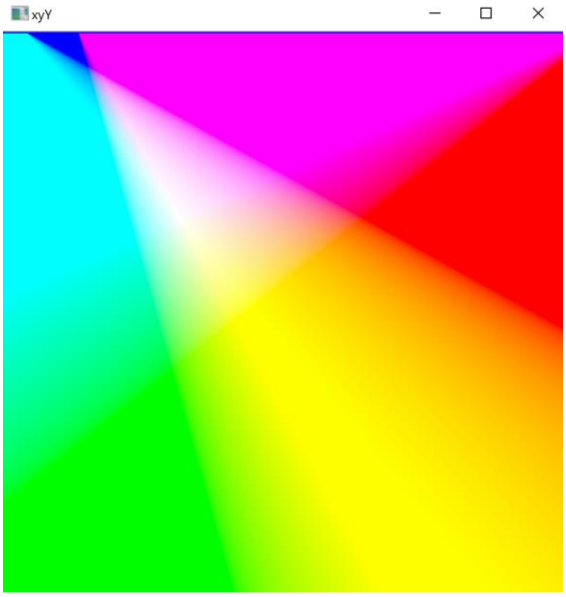
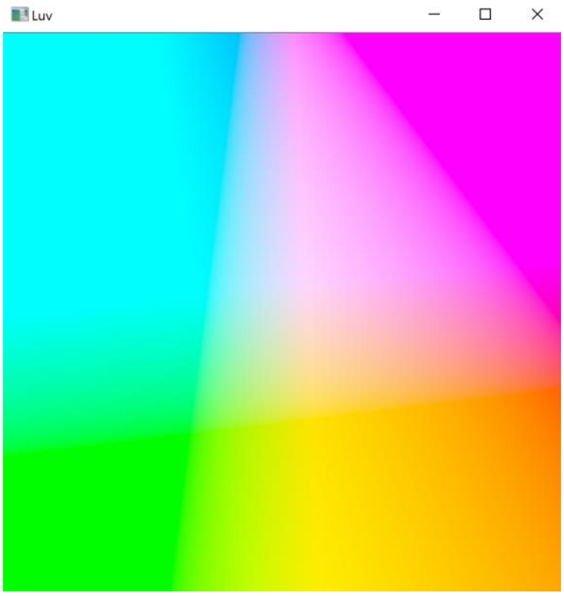
CS 6384.002 Computer Vision Project-1 Report

Xiangru Zhou

Question-1

When the width = 500, height = 500, the results show as:

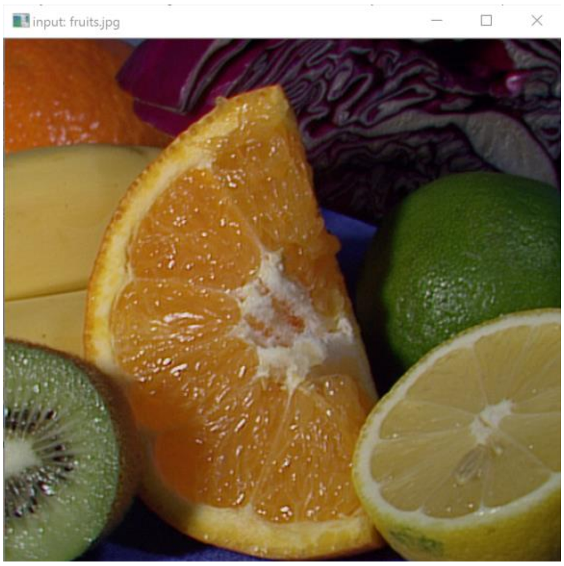


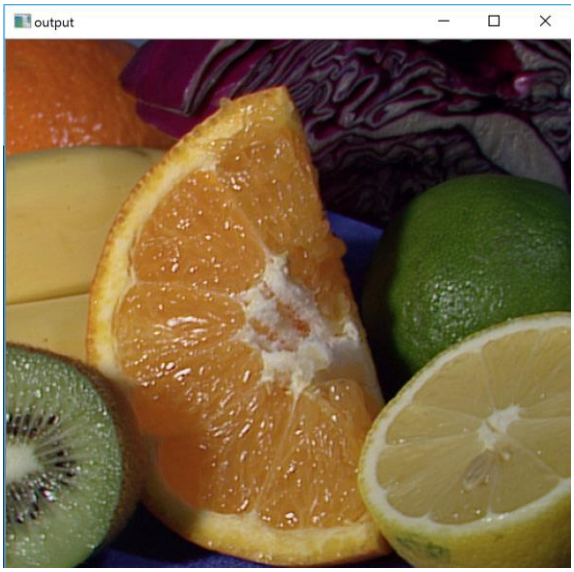


For the xyY image, my workflow is: xyY -> XYZ -> linear sRGB -> non-linear sRGB -> write back to r,g,b
Fot the Luv image, my workflow is: Luv -> XYZ -> linear sRGB -> non-linear sRGB -> write back to r,g,b
Because the range of linear sRGB and non-linear sRGB should be in [0,1], I write a function to check theirs range. If one RGB > 1.0, this RGB is set as 1.0. If one RGB < 0, this RGB is set as 0.
I also write a function to apply gamma correction when dealing with the conversion from linear sRGB to non-linear sRGB. The function is exactly the same as the formula in professor's handout.
When writing a non-linear sRGB back to a pixel, each non-linear sRGB is multiplied by 255 and converttted as an interger.

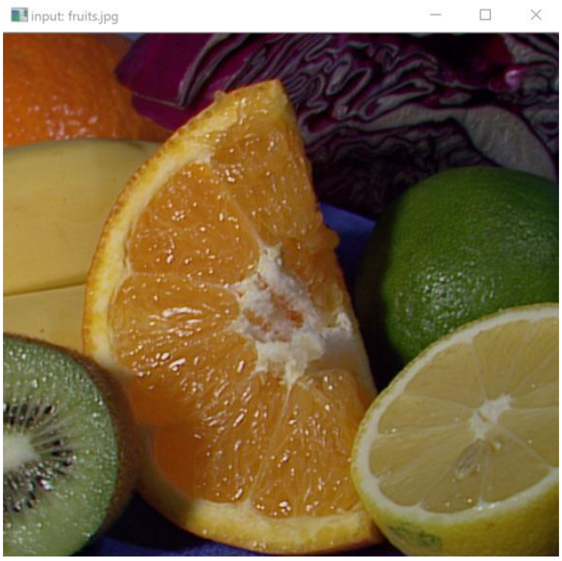
Question-2

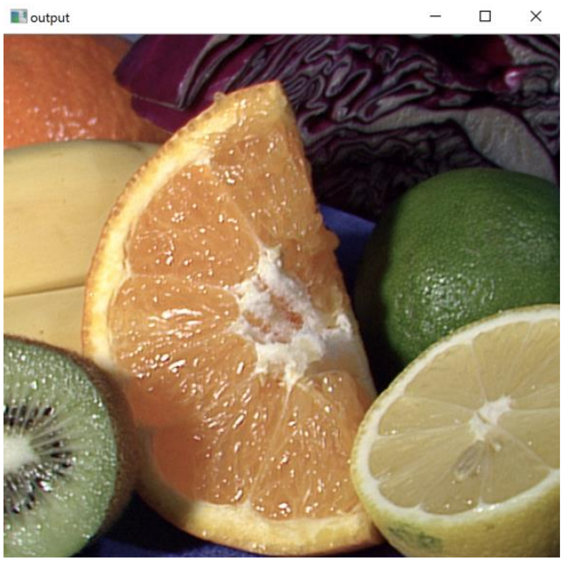
When the window size: W1 = 0, H1 = 0, W2 = 1, H2 = 1, the results show as:



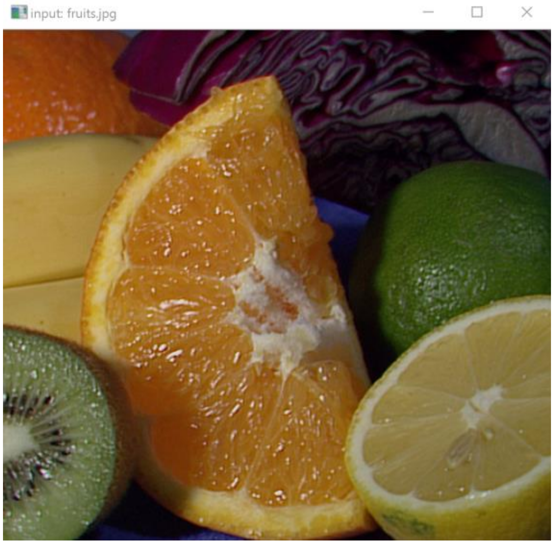


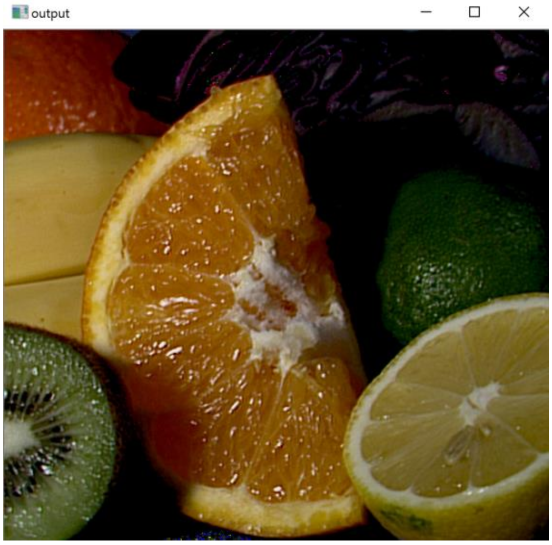
When the window size: W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the results show as:





When the window size: W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the results show as:





My workflow for question-2 is: create and initialize 2D arrays for L value, u value and v value -> calculate all L for a specified window -> find the max and min L for this window -> calculate all L, u, v for the whole image -> apply the linear stretching on every L in the whole image, get new L -> XYZ -> linear sRGB -> non-linear sRGB -> write back to R, G, B

Because the range of linear sRGB and non-linear sRGB should be in [0,1], I write a function to check theirs range. If one RGB > 1.0, this RGB is set as 1.0. If one RGB < 0, this RGB is set as 0.

I write a function to apply gamma correction when dealing with the conversion from linear sRGB to non-linear sRGB. The function is exactly the same as the formula in professor's handout.

I also write a function to apply invgamma when dealing with the conversion form non-linear sRGB to linear sRGB. The function is exactly the same as the formula in professor's handout.

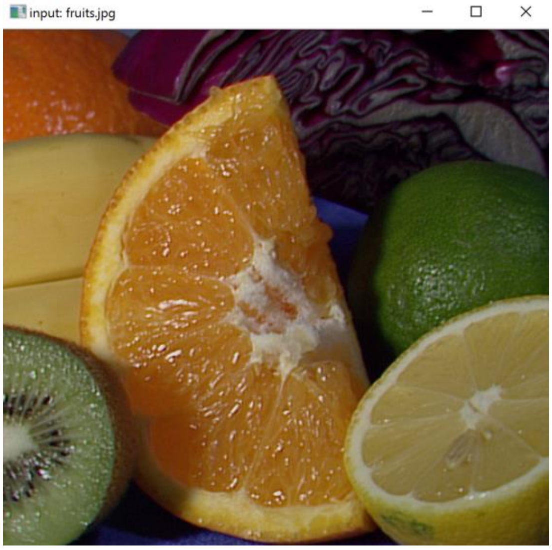
When applying linear stretching on L of the whole image, the min and max L from the specified window are needed for the formula. Each new L should be checked in range [0, 100]. If new L < 0, just set it as 0. If new L > 100, just set it as 100.

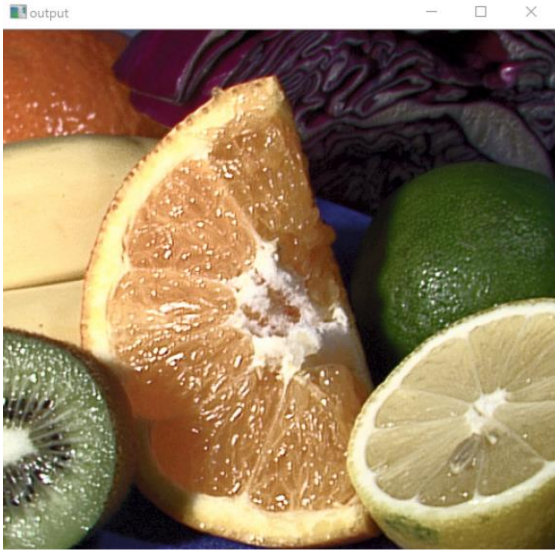
When writing a non-linear sRGB back to a pixel, each non-linear sRGB is multiplied by 255 and converttted as an interger.

From some results, when the specified window size is W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the out image is brighter than the original image. When the specified window size is W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the out image is darker than the original image.

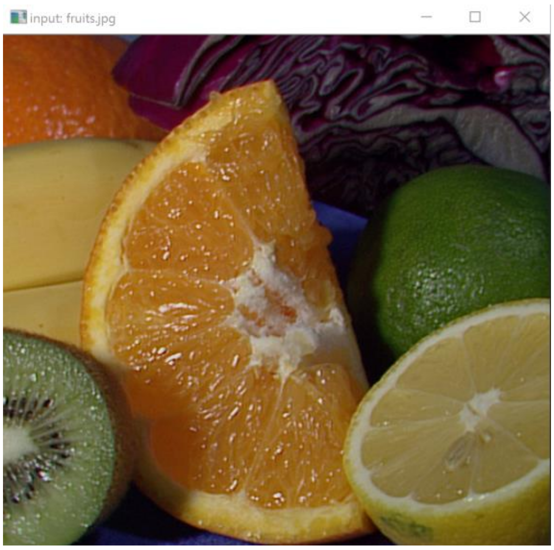
Question-3

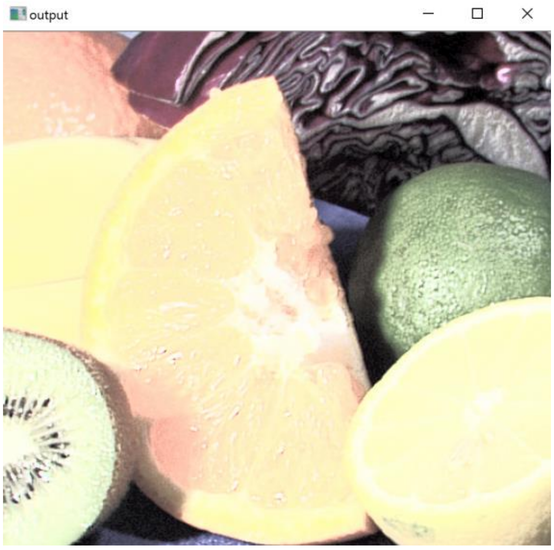
When the window size: W1 = 0, H1 = 0, W2 = 1, H2 = 1, the results show as:



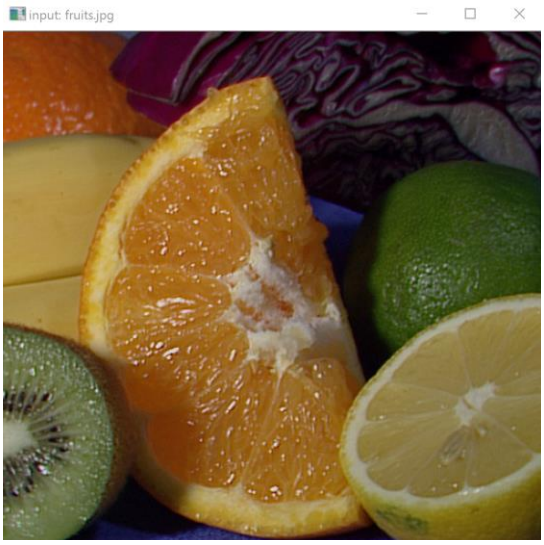


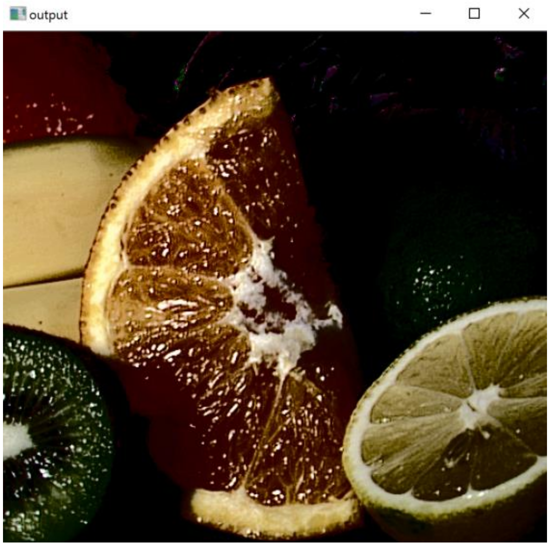
When the window size: W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the results show as:





When the window size: W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the results show as:





My workflow for question-3 is: create and initialize 2D arrays for L value, u value and v value -> calculate all L for a specified window -> find the max and min L for this window -> calculate histogram equalization on L for the specified window, get a mapping from old L value to new L value-> calculate all L, u, v for the whole image -> apply the mapping on every L in the whole image, get new L -> XYZ -> linear sRGB -> non-linear sRGB -> write back to R, G, B

Because the range of linear sRGB and non-linear sRGB should be in [0,1], I write a function to check theirs range. If one RGB > 1.0, this RGB is set as 1.0. If one RGB < 0, this RGB is set as 0.

I write a function to apply gamma correction when dealing with the conversion from linear sRGB to non-linear sRGB. The function is exactly the same as the formula in professor's handout.

I also write a function to apply invgamma when dealing with the conversion form non-linear sRGB to linear sRGB. The function is exactly the same as the formula in professor's handout.

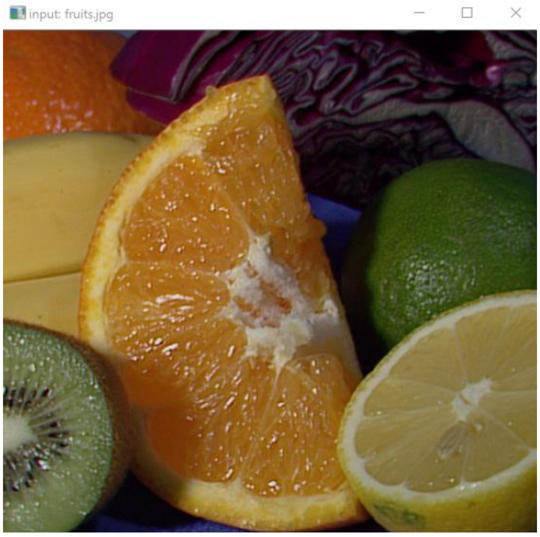
When calculating histogram equalization on L of the specified window, the total number of L is 101. Before applying mapping on L of the whole image, the range of the L need to be checked into [0, 100].

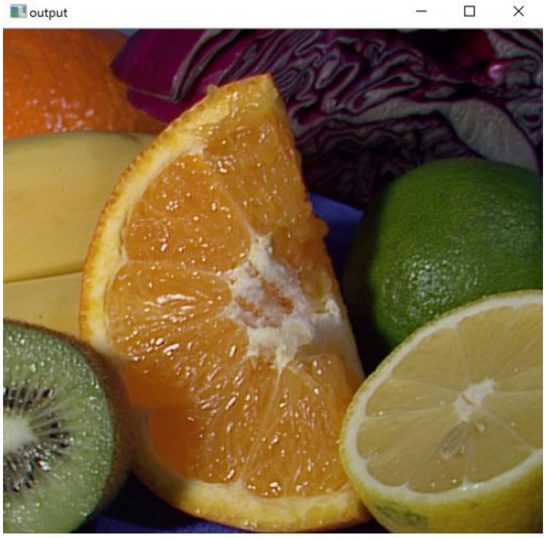
When writing a non-linear sRGB back to a pixel, each non-linear sRGB is multiplied by 255 and converted as an interger.

From some results, when the specified window size is W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the out image is too much brighter than the original image. When the specified window size is W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the out image is too much darker than the original image.

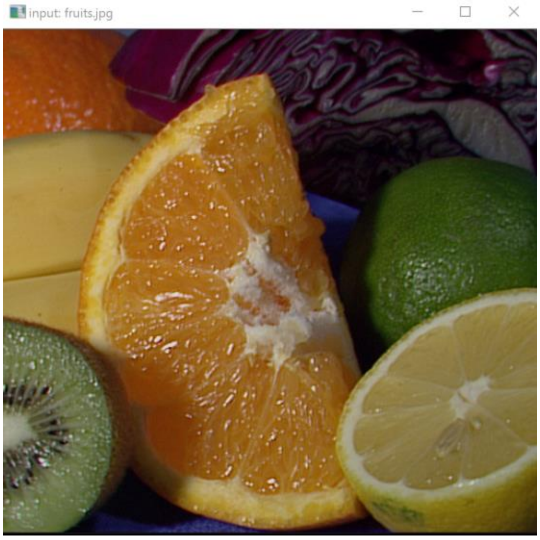
Question-4

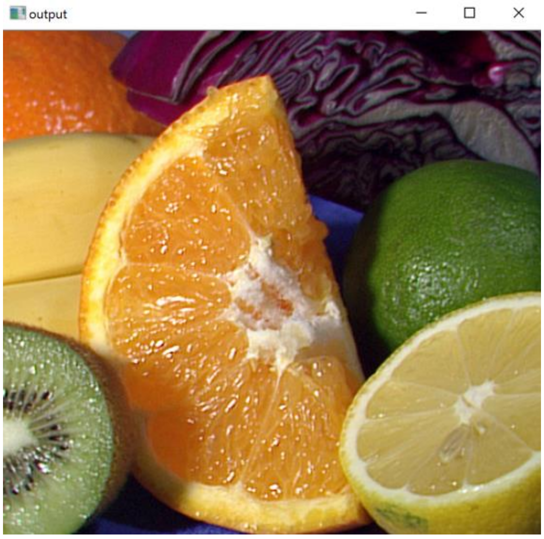
When the window size: W1 = 0, H1 = 0, W2 = 1, H2 = 1, the results show as:



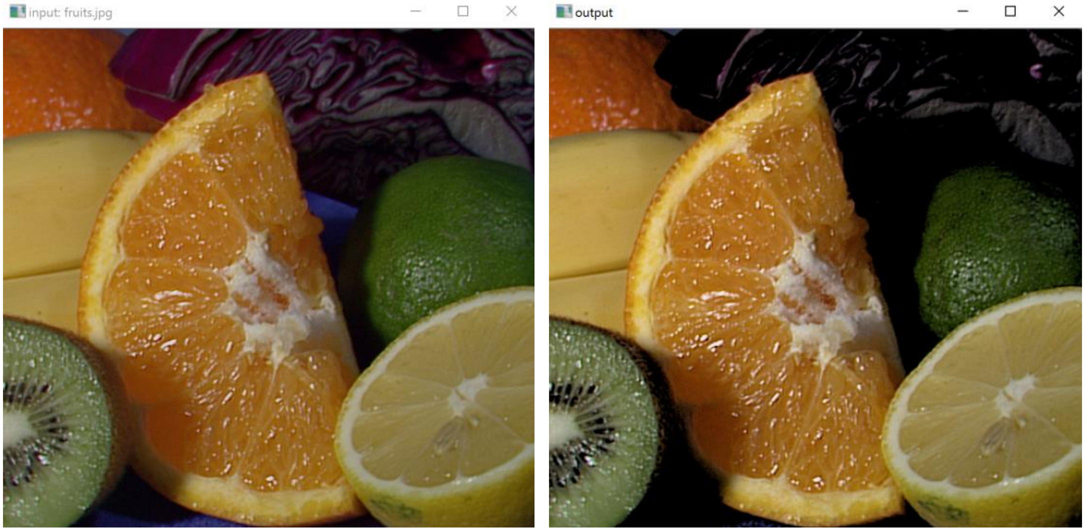


When the window size: W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the results show as:





When the window size: W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the results show as:



My workflow for question-4 is: create and initialize 2D arrays for Y value, x value and y value -> calculate all Y for a specified window -> find the max and min Y for this window -> calculate all Y, x, y for the whole image -> apply the linear stretching on every Y in the whole image, get new Y -> XYZ -> linear sRGB -> non-linear sRGB -> write back to R, G, B

Because the range of linear sRGB and non-linear sRGB should be in [0,1], I write a function to check theirs range. If one RGB > 1.0, this RGB is set as 1.0. If one RGB < 0, this RGB is set as 0.

I write a function to apply gamma correction when dealing with the conversion from linear sRGB to non-linear sRGB. The function is exactly the same as the formula in professor's handout.

I also write a function to apply invgamma when dealing with the conversion form non-linear sRGB to linear sRGB. The function is exactly the same as the formula in professor's handout.

When applying linear stretching on Y of the whole image, the min and max Y from the specified window are needed for the formula. Each new Y should be checked in range [0, 1]. If new Y < 0, just set it as 0. If new Y > 1, just set it as 1.

When writing a non-linear sRGB back to a pixel, each non-linear sRGB is multiplied by 255 and converttted as an interger.

From some results, when the specified window size is W1 = 0.5, H1 = 0, W2 = 1, H2 = 0.5, the out image is brighter than the original image. When the specified window size is W1 = 0.2, H1 = 0.2, W2 = 0.4, H2 = 0.4, the out image is darker than the original image.