Point (Pixel) Processing in Images

Pixel processing is about the alteration of a pixel's color or intensity values. The alteration can be applied unconditionally to all pixels of an image when all values need to be changed the same way without regard to content, e.g. to brighten the image where all of the values are increased by 10 percent. It can also be applied conditionally, based on the values of pixels, which will allow different operations to be carried out based on the content, e.g. to "replace the red pixels with yellow" we would only change the value of pixels that are in a specific intensity or color range.

The intensity histogram

The histogram of a grayscale image shows the distribution of intensity levels for all pixels of that image. It can be useful in adjusting intensity levels.

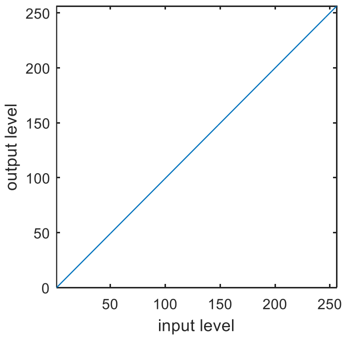
You can experiment with images interactively using the imtool, e.g. imtool('moon.tif')

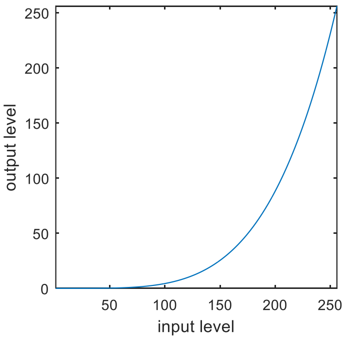
Simple brightness adjustment

The brightness of an image can be increased or decreased simply by multiplying all pixels with a scaler. That is, A = A * 1.2; However, this will saturate many pixel values near the upper limit and will cause loss of detail. A better way to adjust brightness is by using nonlinear mappings.

Altering intensities with nonlinear mappings

A transform can be defined to map intensities of the input image to the intensities of the output. We can simply define a mapping with a vector of 256 elements. Here are plots of two mapping vectors:





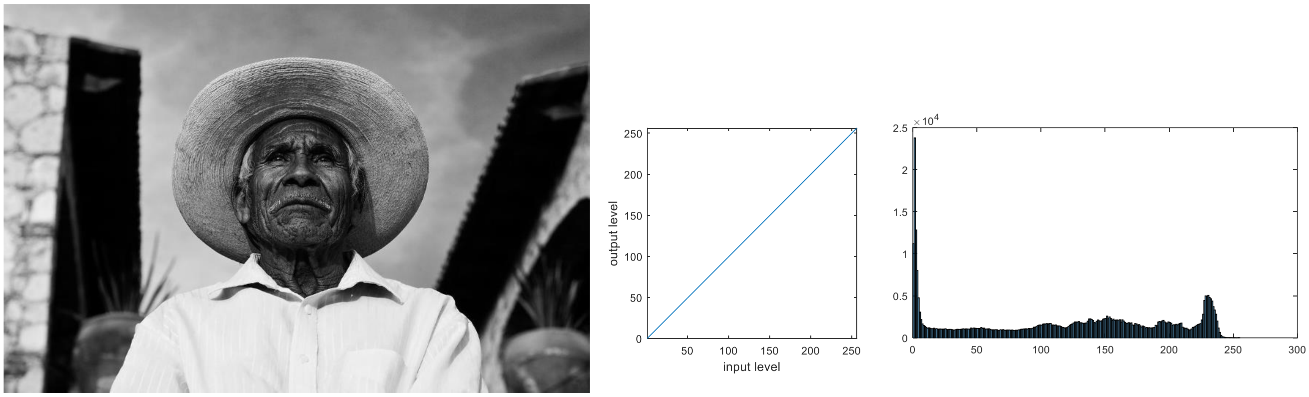
The one on the left is an identity mapping which leaves the values unchanged. The one on the right will suppress darker pixels further and enhance the difference between lighter pixels.

These transformations are also used in display devices to correct for the human eyes' intensity nonlinearity. One such transformation is gamma correction where

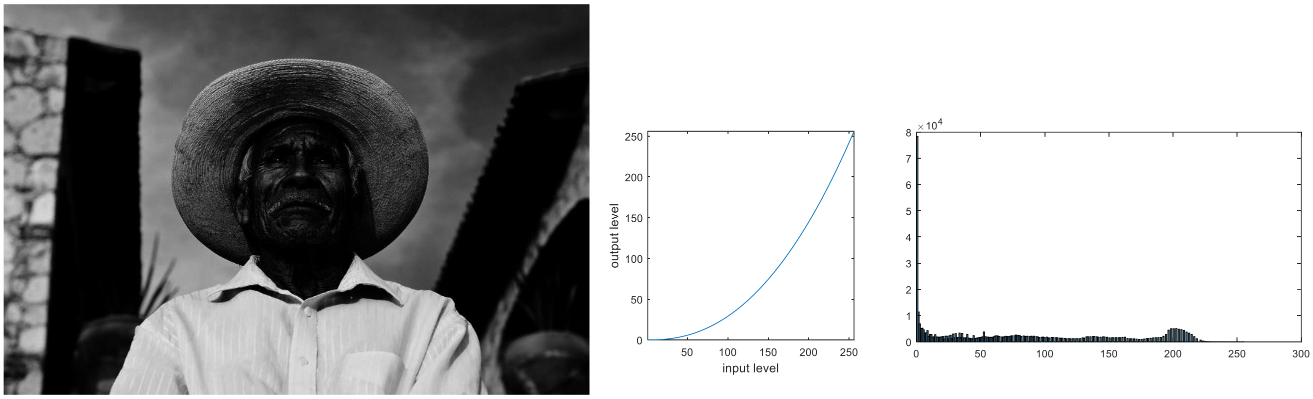
$$V_{out} = V_{in}^{\ gamma}$$

Below are examples of different levels of gamma applied to the same image. The images are shown on the left, the transform is shown in the middle and the histogram is shown on the right for each case.

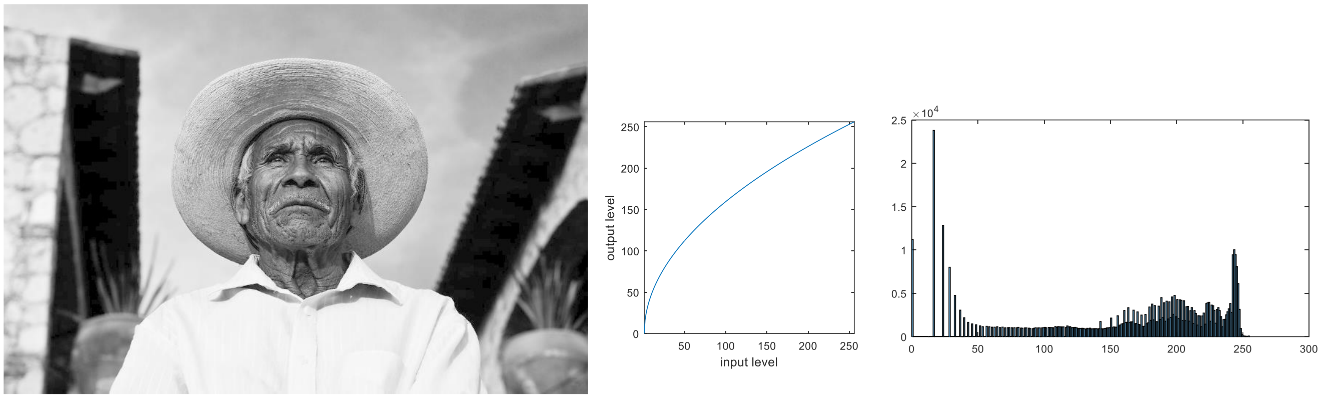
original



gamma = 1.0



gamma = 2.3



gamma = 0.5

Checking color ranges in RGB

We can check for color ranges using the RGB color model by specifying color intensity ranges on the individual color channels. For example, R > 100 and G < 100 and B < 100 would specify a red dominant color with intensity at least 100. This will work if all the pixels of interest have red intensities greater than 100. But, there might also be pixels of the same color that are not as bright that we would want to target. In that case, we can use a color ratio condition such as:

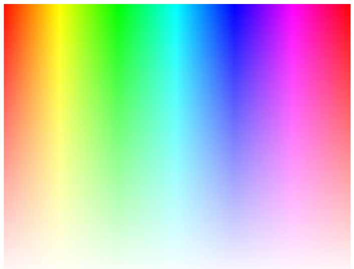
$$R / (R + G + B) > XR$$

where XR is a fixed threshold between 0 (no red) and 1 (pure red).

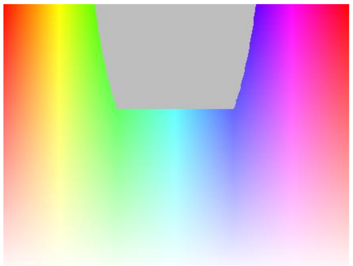
In general, we can combine conditions (with logical AND) for all color channels to look for more specific color ranges. E.g.

$$R / (R + G + B) > XR$$
$$G / (R + G + B) < XG$$
$$B / (R + G + B) < XB$$

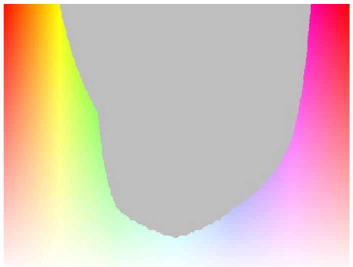
Here is an example demonstrating the difference between an absolute intensity threshold and the above conditions on a color spectrum (left figure). The first condition is R > 100 (lower left figure) and only the pixels that meet the condition are displayed. The second condition uses R > 100, XR= 0.3, XG=0. 5, XB=0.4 (lower right figure). It is easier to target a particular color using the second method.



(original)



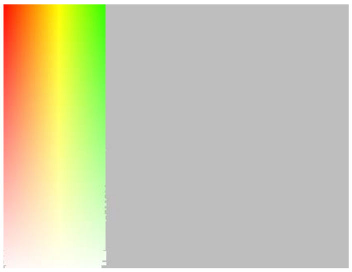
R > 100



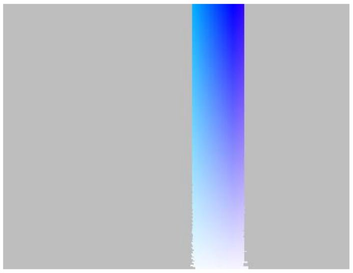
R > 100, XR= 0.3, XG=0. 5, XB=0.4

Checking ranges in HSV

Once an RGB image is converted to HSV we can target color ranges using the hue values. Two examples are given below on the same image as above. Hue from 0 to 0.3 (left) and 0.55 to 0.7 (right).



0 < hue < 0.3



0.55 < hue < 0.7

Content based processing

If we need to change pixels based on the content then we can make a copy of the original image, analyze the original image and make changes on the second image based on the analysis. For example, to detect a line or curve drawn with a different color along a certain direction we can follow paths and look for color changes.

Example: To replace a region with a particular color we can check the range and generate a new color in the new image only for those pixels. In the image below, the red color has been altered with random intensities. Other pixels remain unaltered.