CHAPTER 4

SCALAR PROCESSING

Scalar image processing is a natural extension of grayscale image processing techniques to colour images. The three R, G and B components result in three image *planes* each represented as an array of $M \times N$ values ranging from 0 to 255. M and N represent the number of rows and columns respectively. There are broadly three types of operations in grayscale image processing: *point*, *spatial domain* and *frequency domain* operations. In this chapter, we describe the colour versions of these three types.

4.1 Point Operations

Point operations modify the value of a pixel independently of other pixels. These operations are also known as *global operations* because the modification remains the same across the entire image. These operations are typically used for image *enhancement*. A particularly important set of operations modify *contrast* in grayscale images.

Any point operation may be expressed as

$$I'(x, y) = f(I(x, y)) \tag{4.1}$$

where $I'(x, y)$ is the modified value of the pixel at (x, y), $I(x, y)$ is its original value and $f(\cdot)$ is function defining the modification operation. Note that the modified value depends *only* on the input value.

Figure 4.1: Colour Negative operation on RGB Colour Components.

4.1.1 Negative

The simplest point operation is *negative* where the function $f(x) = 255 - x$. We apply this operation on the three RGB components. As a result, all the bright areas become dark and vice-versa. Colours change towards their complementaries, e.g., yellow colour containing high values of R and G a low value of B turns blue after applying the negative operations. Greens become purples and reds become greenish-blue or bluish green (see Figure 4.1).

4.1.2 Ranging

Ranging operation retains values within a specified range and makes the other values black. This operation is normally used to extract or emphasise objects based on their colours. The range limits are specified independently for the three components as R_l, R_h, G_l, G_h, B_l and B_h where the subscripts l and h indicate the lower and upper limits of the ranges. As a point operation, it is expressed by the function

$$f(x) = \begin{cases} x & \text{if } x_l \leq x \leq x_h \\ 0 & \text{otherwise} \end{cases}$$

where x_l and x_h are the lower and upper limits of the range on x. Figure 4.2 shows how the light blue pants and dresses can be extracted from the image using a ranging operation. The parameters chosen are: $R_l = 0, R_h =$

Figure 4.2: An example of image ranging: it is possible to extract the blue pants and dresses from the colourful image by choosing appropriate ranges of R, G and B values.

$100, G_l = 100, G_h = 200, B_l = 128$ and $B_h = 255$. Note how the choice of limits allows the extraction of only a particular shade of blue. Other darker blue shades (the stripes on the tent, the pants worn by Goofy or the vest worn by Donald Duck) as well as the lighter shades in the vertical stripes on the tent are not extracted.

4.1.3 Threshold

In an image threshold operation, all the values above a certain specified value, called the *threshold*, are retained in the output image while those below it are set to black. When this operation is applied to the R, G and B components, it allows separation of brightly coloured regions from others. There may be a single threshold that is applied to all components or different thresholds for different components.

The threshold operation is given by the function

$$f_\theta(x) = \begin{cases} x & \text{if } x > \theta \\ 0 & \text{otherwise} \end{cases}$$

where θ is the threshold value.

There are several variants of the threshold operation on colour images. In the simplest case, the colour component value is set to 255 if it is above the specified threshold, or to 0 otherwise. The output value may be set to 0 if the input value is below the threshold and unaltered otherwise. A third variant sets the output value to white if the input is above the threshold and black otherwise. This variant results in a binary image. All these variants are shown in Figure **??** for a threshold of 164.

4.1.4 Contrast Enhancement

Contrast is a measure of the difference between the bright and dark areas in an image. *Colour contrast*, by extension, is a measure of the difference between the brightly coloured and dark regions of an image. The higher the contrast, the higher the difference. Generally, people find a higher contrast image as having better quality (see Figure 4.4).

There are two main techniques for contrast modification: transfer functions and histograms. Transfer function maps input gray level to an output gray level. The simplest transfer functions are linear. A line with a slope of $\pi/4$ is the *identity* function mapping input values to identical output values. Any line with a slope greater than $\pi/4$ and less than $\pi/2$ increases contrast while slopes less than $\pi/4$ reduce contrast. More localised contrast enhancements may be obtained using piecewise linear functions.

Figure 4.3 shows the effect of contrast changes using lines with different slopes as transfer functions. The first image shows three transfer functions: $H(x), L(x)$ and $P(x)$. The second image is the original. The third image is obtained by using $H(x)$ as the transfer function. The slope of $H(x)$ is greater than $\pi/4$ and thus increases contrast. The third image is with $L(x)$ and the fourth, with $P(x)$ as transfer functions. $L(x)$ has a slope less than $\pi/4$ and reduces overall contrast. $P(x)$ is piecewise linear with less slope and contrast reduction between 0 and 64, and also from 160 to 255. It increases contrast between the values 64 and 160 as can be seen from the higher slope.

Contrast is hard to define uniquely for colour images. One definition is to measure the difference between the bright and dark values of each component. Contrast enhancement then refers to increasing the differences between bright and dark reds, bright and dark greens, and bright and dark blues. There will almost certainly be other artifacts because of the correlations between the different layers.

Figure 4.4 shows the effect of not taking correlations into effect while modifying contrast. The image shows yellow flowers surrounded by green leaves. Yellow colour is composed of high R and G, and low B values. It implies that the image may have fewer or no pixels with high B values. When

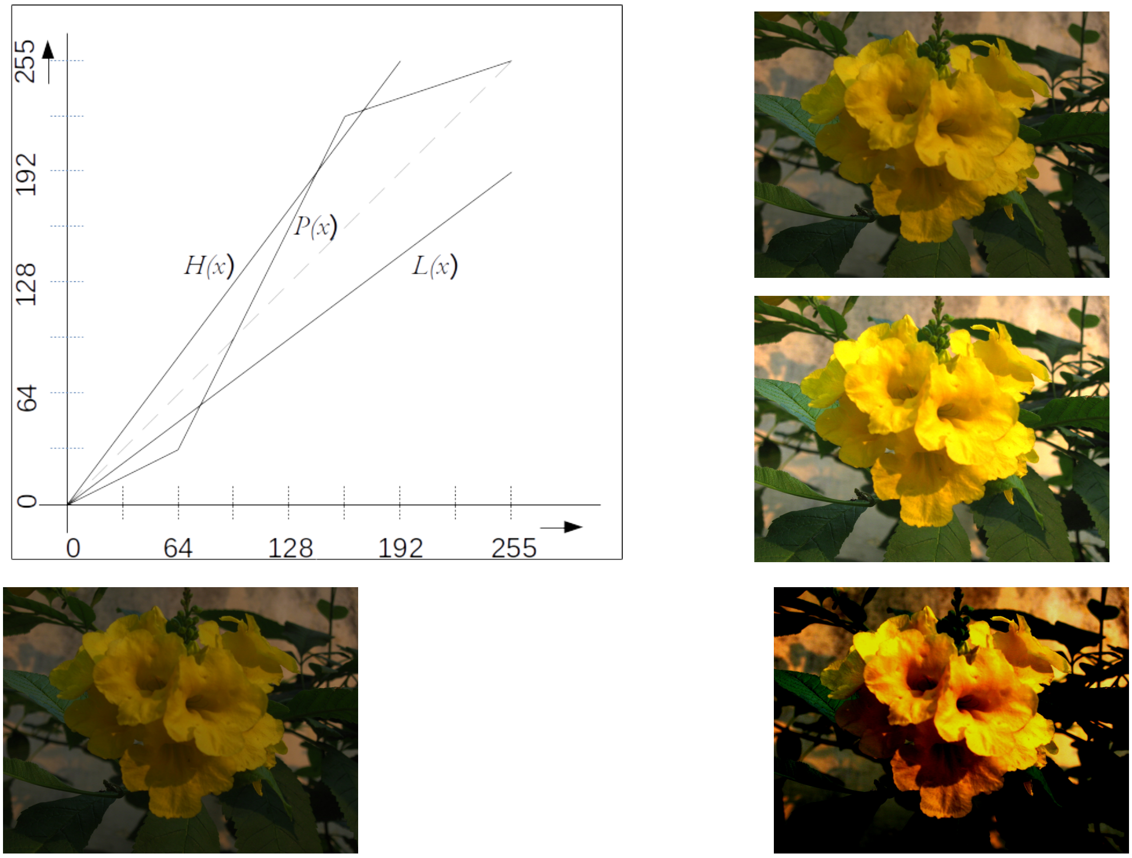

Figure 4.3: Contrast modification using transfer functions. (a) Transfer functions used and (b), (c), (d) and (e) resulting images.

we stretch the components to increase their ranges, R and G components do not get highly stretched while the B component does (see Section **??**). As a result, the output contains an abnormal number of high B value pixels which give a bluish tinge to the entire image (Figure 4.4(b)).