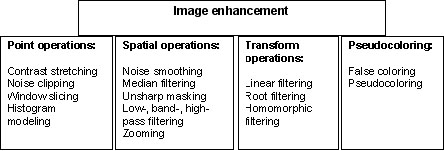
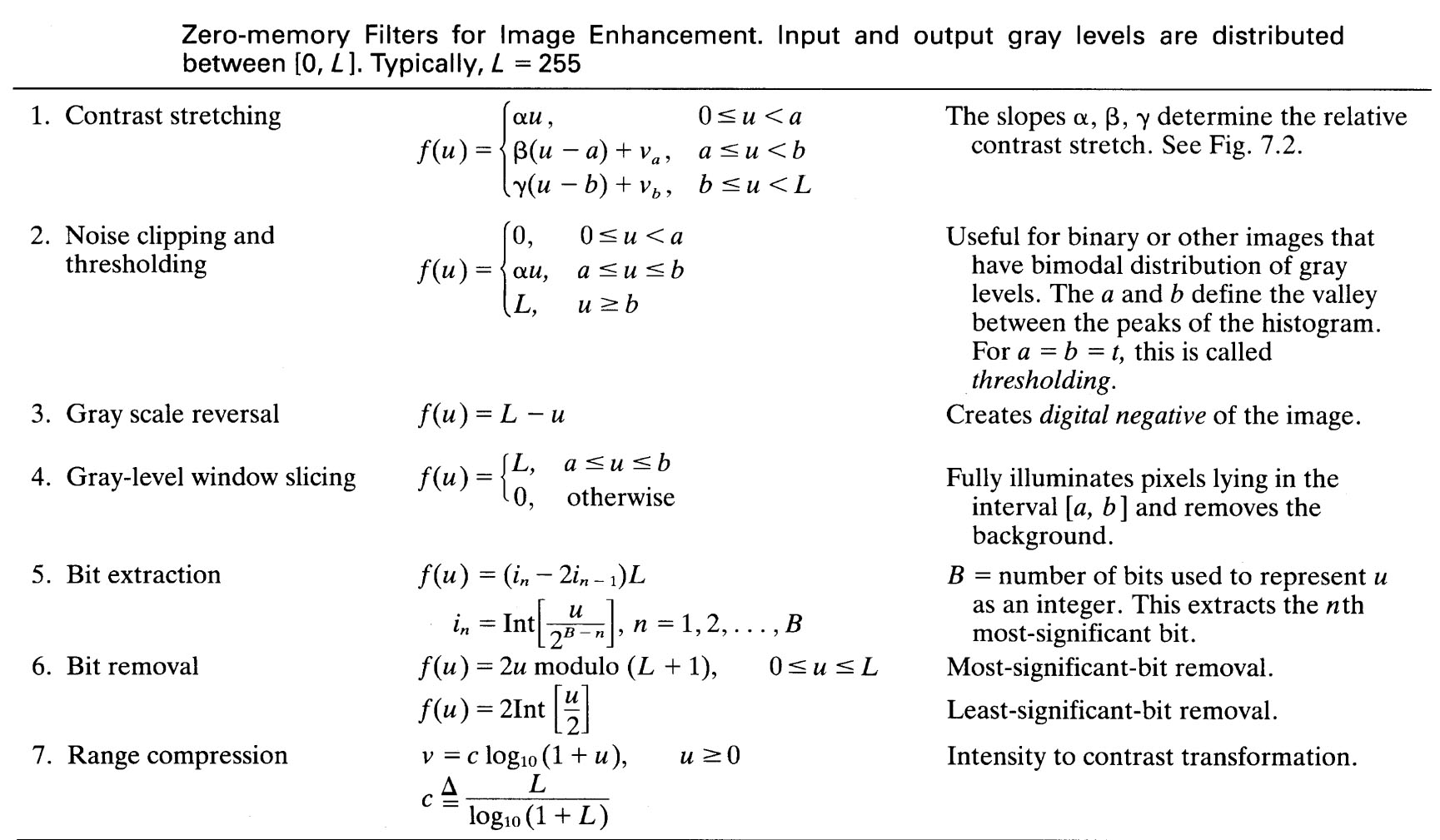
**3. Image Enhancement**

Image enhancement is used in all image processing applications, it refers to accentuation, or sharpening of image features such as edges, boundaries, or contrast to make an image more useful for display and analysis. Image enhancement includes gray level and contrast manipulation, noise reduction, edge crispening and sharpening, filtering interpolation and magnification, pseudocoloring, and so on.



**3.1. Point operations**

Point operations are zero memory operations where a given gray level *uÎ [0,L]* is mapped into a gray level n Î *[0,L]* according to a transformation n *= f(u)*. The following table presents several of these transformations.

****

***(i) Contrast Stretching***

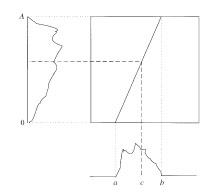
Low contrast images occur often due to poor or nonuniform lighting conditions or due to nonlinearity or small dynamic range of the imaging sensor.

Frequently, the image has weak contrast due to the limited used values of the gray levels. The solution is made by applying the function which extend the used levels to maximum according

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image311_4.gif

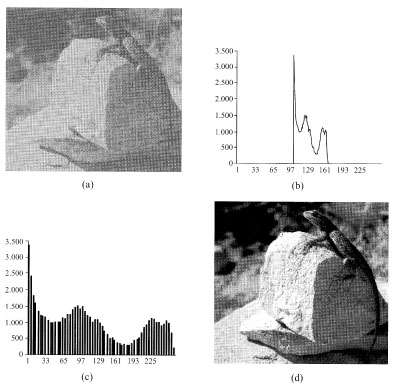
where *a*and*b*are the lowest and highest limits of the gray scale in input image,

*c* is the value of transformed gray level of input image, *A* is the maximum gray level value of the pixels in output image as shown in figure



Amplitude of the scales in input and output images

One example of the scale amplitude extension is shown in following figure:

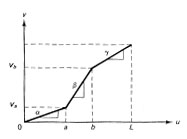


Contrast magnification by the extension of the scale amplitude

and histograms of input and output images

General case of the Contrast stretching can be expressed as

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image311_5.gif

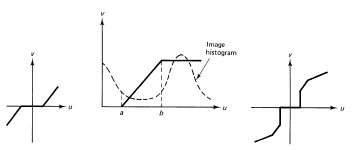


Contrast stretching transformation

The parameters a and b can be obtained by examining the histogram of the image, for example, the gray scale intervals where pixels occur most frequently would be starches most to improve the overall visibility of a scene.

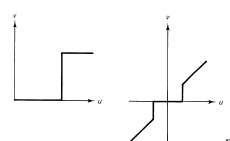
***(iii) Clipping and Thresholding***

The special case of contrast stretching where a *=g =0* is called clipping. This is useful for noise reduction when the input signal is known to lie in the range *[a,b].*



Clipping transformations

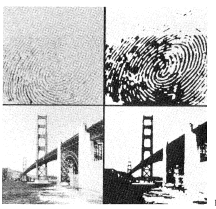
Thresholding is the special case of clipping where *a=b* and output becomes binary.



Thresholding transformations

The thresholding is used to make such an image binary.

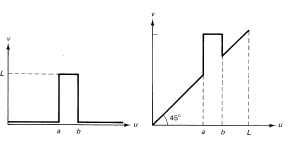
The examples of the clipping and thresholding are shown in the following figure



Clipping and thresholding

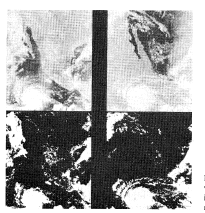
***(iv) Intensity Level Slicing***

There are intensity level slicing without background and with background.



Intensity level slicing a) without background b) with background

These transformations permit segmentation of certain gray level regions from the rest of the image. This technique is useful when different features of an image are contained in different gray levels. The following image shows the result of intensity window slicing for segmentation of low-temperature regions (clouds) of two images where high intensity gray levels are proportional to low temperatures.



Level slicing of intensity window *[175,250]*.

Top row: visual and infrared images, bottom: segmented images

***(v) Bit Extraction***

Suppose each image pixel is uniformly quantized to *B* bits. It is desired to extract the *n-*th most significant bit and display it. Let

H:\lecture\IPPR lecture notes 2068\Clipping and Thresholding_files\Image313_6.gif

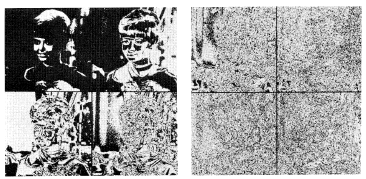
Than we want the output to be

H:\lecture\IPPR lecture notes 2068\Clipping and Thresholding_files\Image313_7.gif

It is easy to show that

H:\lecture\IPPR lecture notes 2068\Clipping and Thresholding_files\Image313_8.gif

In the following figure the first most significant bit of an 8-bits image is shown



8-bits planes of a digital image: a) first four significant bits of image

b) last four significant bits of image

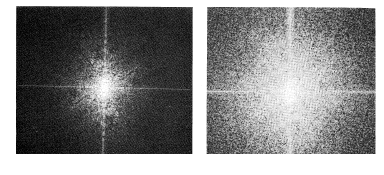
This transformation is useful in determining the number of visually significant bits in an image. As the previous figure shows, the first 6 bits are visually significant, because the remaining bits don not convey any information about the image structure.

***(vi) Range Compression***

Sometimes the dynamic range of the image data may be very large but only a few pixels are visible. The dynamic range may be compressed via the logarithmic transformation.

H:\lecture\IPPR lecture notes 2068\Clipping and Thresholding_files\Image313_10.gif

where c is a scaling constant. This transformation enhances the small magnitude pixels compared to these pixels with large magnitudes



Range compression: a) original, b) logarithmic transformation.

***(vii) Histogram Modeling***

The histogram of an image represents the relative frequency of occurrence of the various gray levels in the image. The histogram modeling techniques modify an image globally so that its histogram has a desired shape.

*1) Histogram Equalization*

The principal goal of the histogram equalization is to obtain the uniform histogram (the function of occurrence must be horizontal, that means that all levels of gray scale have the same occurrence). The histogram equalization is made by using the accumulated histogram represented as:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_1.gif

If the histogram is completely plain, the accumulated histogram for each level of the gray scale may be presented as

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_2.gif

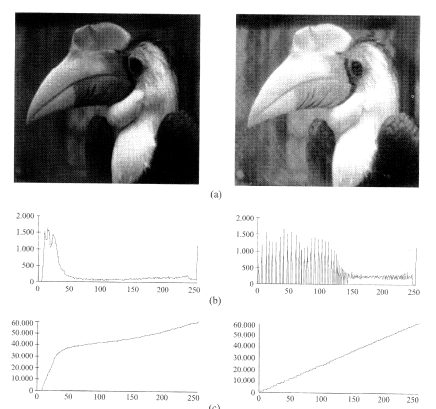
where N and M are the dimensions of an image and 256 is a number of the gray scales. In ideal case it is desired that *G(i')=H(i)*. Therefore

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_3.gif

The gray levels are integer numbers, therefore the following of the gray scales are made

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_4.gif

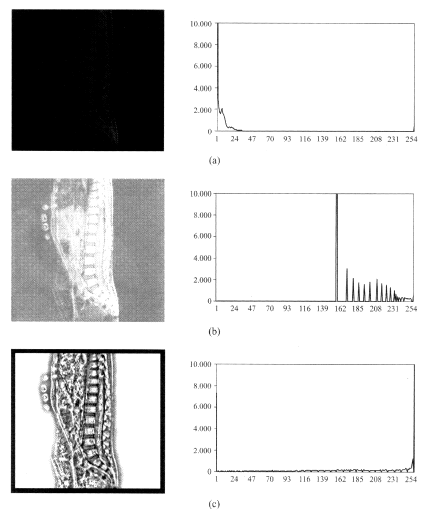
The application of the histogram equalization is shown in the following figure



Histogram equalization: a) input and equalized images,

b) their histograms, c) their accumulated histograms

As it can see in the image the output image has the uniform distribution of the gray scales that sometimes leads to weak contrast images. That is why, usually, the equalization by windows is used as it shown in the following figure.



Histogram equalization: a) input image and its histogram, b) equalized image with histogram, c) equalized image by windows of 15x15 and its histogram

*2) Histogram Modifications*

There are other types of the distributions for obtaining the uniform histograms

***Exponential***

The output image has been obtained by applying

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_7.gif

That gives the new value of pixel as:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_8.gif

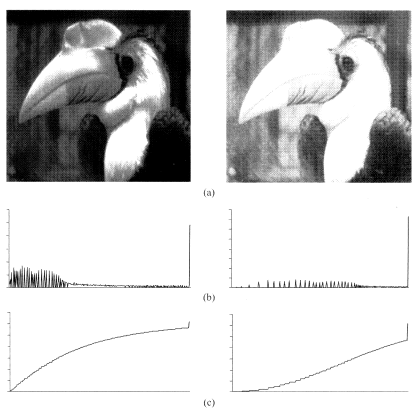
where the factor a is used for exponent variation.

***Rayleigh***

The new histogram and the new values of pixel are obtained by applying the following equations:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image314_9.gif

where the factor a is used for varying the increment of distribution.



Histogram modification: a) the exponential and Rayleigh output images,

b) their histograms, c) their accumulative histograms

***Cubic root distribution***

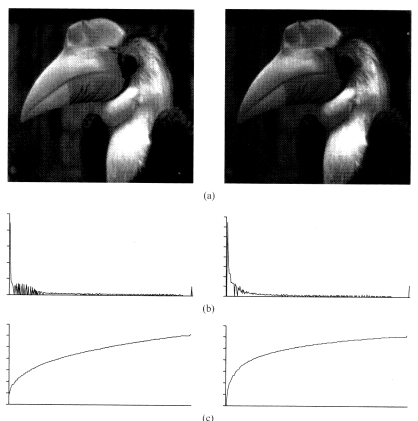
The new histogram and the new values of pixel are obtained by applying the following equations:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/image314_11.jpg

***Logarithmic distribution***

The new histogram and the new values of pixel are obtained by applying the following equations:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/image314_12.jpg



Histogram modification: a) the cubic root and logarithmic distributions output images,

b) their histograms, c) their accumulative histograms

For mathematical transformations a lot of time is necessary to read the pixel, apply the function, normalize the result for output image. For real-time applications the LUT look up table is used (as TLB of the virtual memory systems) . This table consists in the two columns where the first one contains the old values of pixels and another one - the corresponding new pixel values.

***(v) Spatial Low-pass, High-Pass, and Band-Pass Filtering***

*Noise classification*

Each image contains the noise of different types. In general, the four types of noise are classified:

* Gaussian noise produces small variations in an image due to the characteristics of the sensor, digitizing noise, perturbations in transmission, and so on. The value of pixel is composed by ideal value of pixel plus the additional error described by Gaussian variable.
* Impulse noise (known as salt and pepper). The noise takes the either very low or very high values and has no any relationship with the ideal value of the pixel. The sources of the noise are the saturation or loss in the sensor.
* Frequencial noised image is due to the sum of the input image and harmonics of another interfered signal.
* Multiplicative noise as the result of the multiplication of the two signals.

For noise smoothing the filtering techniques are used. There is the frequently used classification of the filters such as :

Spatial Low-pass, High-Pass, and Band-Pass filters.

*The low-pass pass filter* is the mentioned spatial averaging operator:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_1.gif

Typically, *the low-pass filter* would perform a relatively long-term spatial average (for example, on a 5x5, 7x7, or larger window). The low-pass filters are used for noise smoothing and interpolation.

The high-pass filter is defined as simple subtracting the low-pass filter output from filter's input

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_2.gif

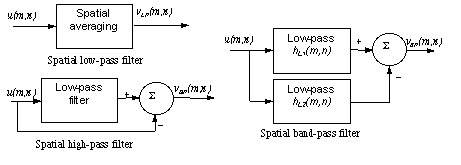
The high-pass filters are useful in extracting edges and in sharpening images.

*The spatial band-pass filter* can be characterized as

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_3.gif

where http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_4.gif denoted the low-pass filters. Typically, they would represent short-term and long-term averages. The band-pass filters are used in enhancement of edges and other high-pass characteristics in the presence of noise.

The block diagrams of these filters can be presented as



Spatial filtering examples: original, high-pass, low-pass, and band-pass filter

***(vi) Magnification and Interpolation (Zooming)***

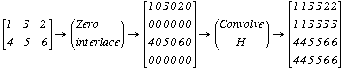
Often the zooming of a given region is necessary to obtain. This requires taking an image and displaying it as a larger image.

*Replication.*

It is a zero-order hold where each pixel along a scan line is repeated once and then each scan line is repeated. It is equivalent to taking an *MxN* image and interlacing it by rows and columns of zeros to obtain a *2Mx2N* matrix and convolving the result with an array***H***, defines as: http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_5.gif

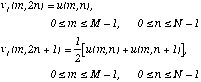
This gives http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_6.gif

The following figure shows the interpolation process and examples.

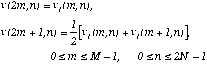


*Linear interpolation*

It is a first order hold where a strait line is first fitted in between pixels along the row. Then pixels along each column are interpolated along a straight line. For example, for *2x2*magnification, linear interpolation along row gives



The interpolation along columns gives the first result as

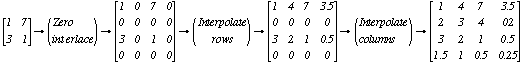


Here it is assumed that the input image is zero outside *[0,M-1]x[0,N-1]*. The above result can also be obtained by convoluting the *2Mx2N* zero interlaced image with the array

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_10.gif

whose origin *(m=0,n=0)* is at the center of the array, that is, the boxed element. High order interpolation (say, *p*) is possible by padding each row and each column of the input image by *p* rows and *p* columns of zeros, and convolving it *p* times with the ***H*** array.

The following figure shows the linear interpolation process. The result is similar to the previous example by replications.

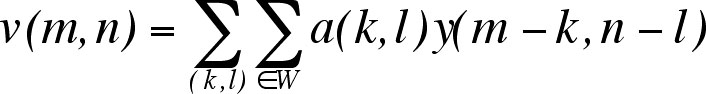


**3.2 Spatial operations**

Many image enhancement techniques are based on spatial operations performed on local neighborhoods of input pixels. Often the image is convolved with a finite impulse response filter called the *spatial mask.*

***(i) Spatial Averaging and Spatial Low-pass Filtering***

The each pixel is replaced by a weighted average of its neighborhood pixel, that is,



where *y(m,n)* an *v(m,n)* ate the input and output images, *W* is a suitably chosen windows, and *a(k,l)* are the filter weights. A common class of spatial averaging filters has all equal weights, giving

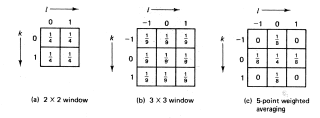
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_1.gif

where *a(k,l)=1/NW* and *NW* is the number of pixels in the window *W*.

Another spatial averaging filter used often is given by

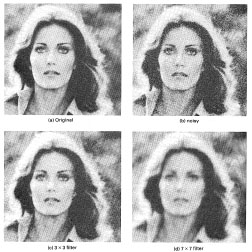
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_2.gif

that is, each pixel is replaced by its average with the average of its nearest four pixels.



Spatial averaging mask *a(k,l)*

Spatial averaging is used for noise smoothing, low pass filtering, and subsampling of images. If the noiseless image *u(m,n)* is constant over the window *W*, then spatial averaging results in a improvement in the output signal-to-noise ration by a factor of *NW*. In practice, the size of window *W* is limited due to the fact that *u(m,n)* is not really constant, so that spatial averaging introduces a distortion in the form of blurring.



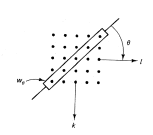
Spatial averaging filters for smoothing images containing the Gaussian noise.

***(ii) Directional Smoothing***

To protect the edges from blurring while smoothing, a directional averaging filter can be used. Spatial averages *v(m,n:q )* are calculated in several directions as

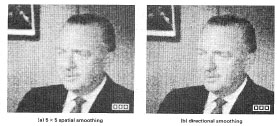
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_4.gif

and a direction * \** is found such that | *y(m,n) - v(m,n: \*) |*is minimum.



Directional smoothing filter

Than *v(m,n)=v(m,n: \*)*gives the desired result.



Comparison of the spatial and directional smoothing

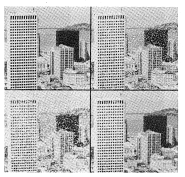
***(iii) Median Filtering***

The input pixels is replaced by the median of the pixels contained in a window around the pixel, that is

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/image32_6.jpg

where*W* is a suitable chosen window. The algorithm for median filtering requires arranging the pixel values in the window in increasing or decreasing order and picking the middle value. For example, if *y(m)={2,3,8,4,2}*and *W=[-1,0,1]*, then the median filter output is given by: *v(0)=2* (boundary value), *v(1)=median{2,3,8}=3*, *v(2)=median{3,8,4}=4*, *v(3)=median {8,4,2}=4*, *v(4)=2*(boundary value).

Typically windows are 3x3, 5x5, 7x7, or 5 point window considered for spatial averaging. (size of window is chosen so that *NW* is odd). The main purpose of the median filter using is elimination of the *binary* noise.

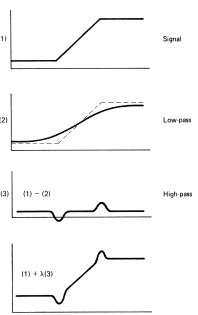


Spatial averaging versus median filtering: (a) original, b) with binary noise,

c) five nearest neighbors spatial average, d) 3x3 median filtered image

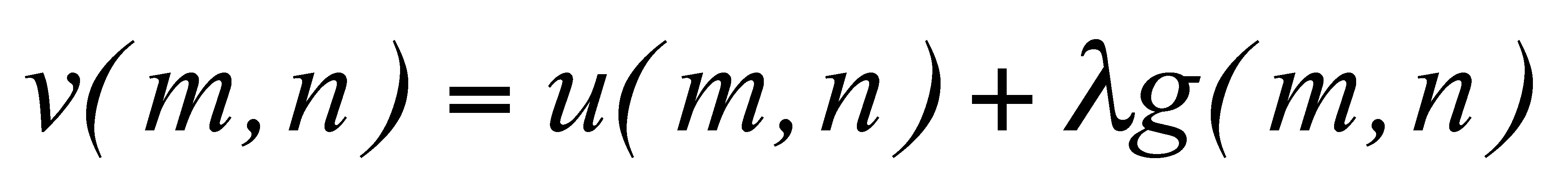
***(iv) Unsharp Masking and Crispening***

The unsharp technique is used for crispening the edges. A singal proportional to unsharp, or low-pass filtered version of the image is subtracted from the image. This is equivalent to adding the gradient, or a high-pass signal, to the image.



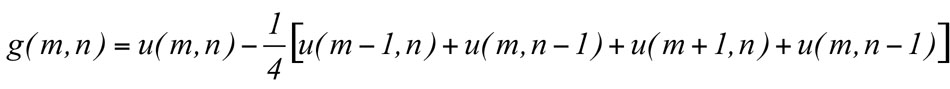
Unsharp masking operation

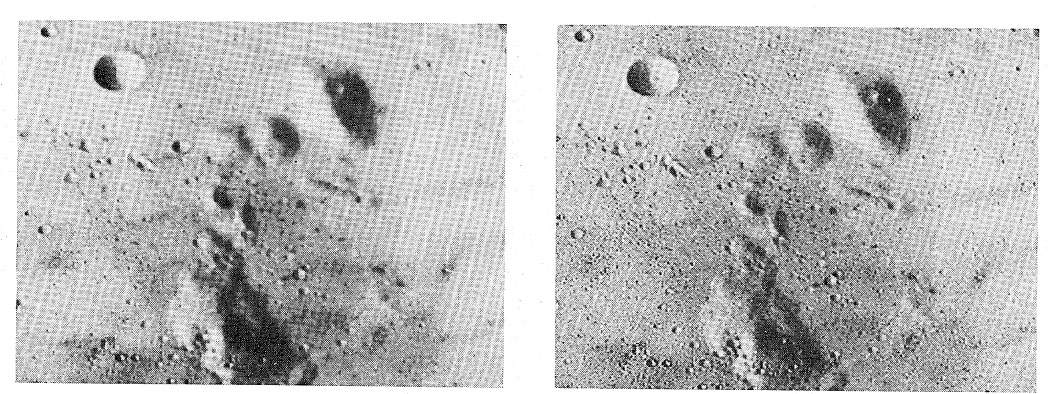
In general, the unsharp masking operation can be represented by



where  *>0* and *g(m,n)* is a suitably defined gradient an *(m,n)*.

A commonly used gradient function is the discrete Laplacian operator (later) :



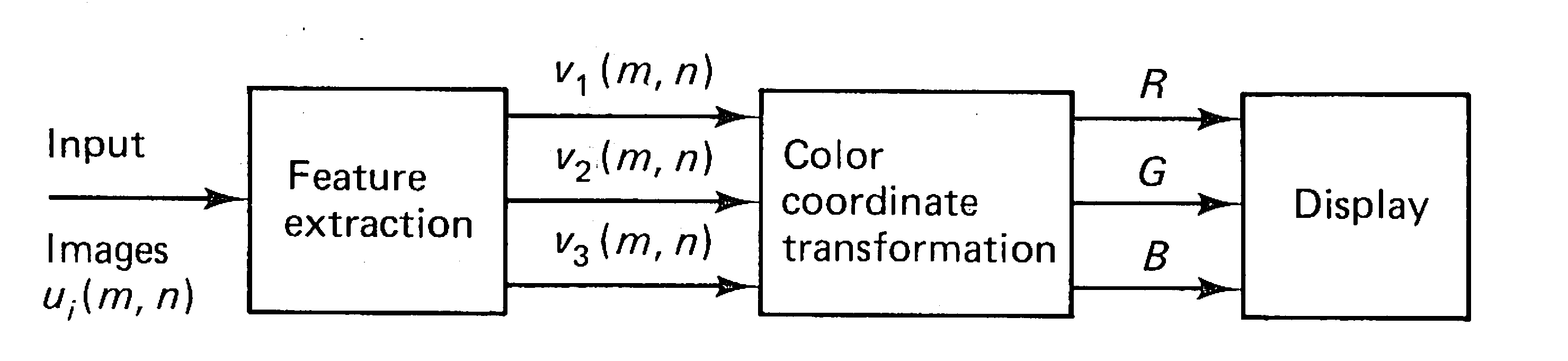


Unsharp masking: original and enhanced images

***3.4 False color and Pseudocoloring***

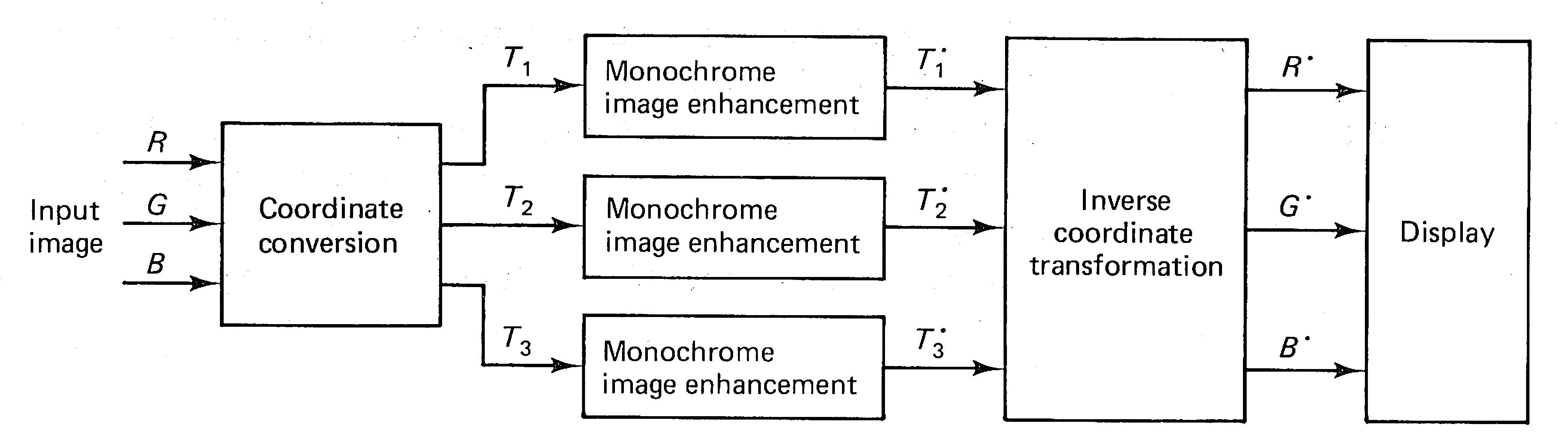
Since we can distinguish many more colors than levels, the perceptual dynamic range of a display can be effectively increased by coding complex information in color. False color implies mapping a color image into another color image to provide a more striking contrast (which may not be natural) to attract the attention of the viewer.

The following figure shows the general procedure for determining pseudocolor mapping where the input image is mapped to three features images, which are then mapped into the three color primaries by any standard coordinate transformation (for example to keep the saturation constant and map the gray level values into brightness and the local spatial averages of gray levels into hue, OR pseudorandom mapping of gray levels into R,G,B coordinates).

**

Pseudocolor image enhancement

Sometimes the color image enhancement may require improvement of color balance or color contrast. The practical approach to develop color image enhancement algorithms is shown in figure.



Color image enhancement

Here the color coordinates of each pixel in input image are independently transformed into the set of the coordinates where the image in each coordinate is enhanced by its own (monochrome) algorithm. The enhanced image coordinates *T\*1, T\*2, T\*3*are inverse transformed to *R', G', B'* for display.