CHAPTER 4

Image Enhancement Techniques.

4.1. Introduction

Image enhancement is the processing of images to increase their usefulness. Methods and objectives vary with the application. When images are enhanced for human viewers, as in television, the objective may be to improve perceptual aspects: image quality, intelligibility, or visual appearance. In other applications, such as object identification by machine, an image may be preprocessed to aid machine performance. Because the objective of image enhancement is dependent on the application context, and the criteria for enhancement are often subjective or too complex to be easily converted to useful objective measures, image enhancement algorithms tend to be simple, qualitative and ad hoc. In addition, in any given application, an image enhancement algorithm that performs well for one class of images may not perform as well for other classes.

Image Enhancement is closely related to image restoration. When an image is degraded, restoration of the original image often results in enhancement. In image restoration, an ideal image has been degraded, and the objective is to make the processed image resemble the original as much as possible. In image enhancement, the objective is to make the processed image better in some sense than the unprocessed one. In this case the ideal image depends on the problem and is often not well defined. To illustrate the difference, note that an original, un-degraded image cannot be further restored but can be enhanced by increasing the sharpness through high pass filtering.

Image enhancement is desirable in a number of contexts. In one important class of problems, modifying its contrast and/or dynamic range enhances an image. For example a typical image, though un-degraded, will often appear better when the edges are sharpened. Also, if an image with large dynamic range is recorded on a medium with small dynamic range, such as film or paper, the contrast and, therefore, the details of the image are reduced, particularly in the very bright and dark regions.

In another class of enhancement problems, reducing the degradation may enhance a degraded image. When an image is quantized for the purpose of bit rate reduction, it may be degraded by random noise or signal dependent false-contours. When an image is coded and transmitted over a noisy channel or degraded by electrical sensor noise, as in a vidicon TV camera, degradation appears as salt–and–pepper noise. An image may also be degraded by blurring (convolutional noise) due to mis-focus of lenses, to motion, or to atmospheric turbulence. In this case high frequency details of the image are often reduced, and the image appears blurred. An image degraded by one or more of these factors can be enhanced by reducing the degradation.

Thus an image can often be enhanced when one or more of the following objectives is accomplished: modification of the contrast or dynamic range; edge enhancement; reduction of additive, multiplicative and salt and pepper noise; reduction of blurring. The following section discusses a few of these techniques by which the geometric detail in an image may be modified and enhanced. The specific techniques covered are applied to the image data directly and could be called image domain techniques [5].

4.2 Scalar and Vector Images

Two particular image types require consideration when treating image enhancement. The first could be referred to as a scalar image, in which each pixel has only a single brightness value associated with it. Such is the case for a simple black and white image. The second type is a vector image, wherein a vector of brightness values represents each pixel, which might be the blue, green and red components of the pixel in a color scene. Most image enhancement techniques relate to scalar images and also to the scalar components of vector imagery. Enhancement methods that relate particularly to vector imagery tend to be transformation oriented.

4.3 The Image Histogram

Consider a spatially quantized scalar image such as that as shown below. If each pixel in the image is examined and its brightness value noted, a graph of number of pixels with a given brightness versus brightness value can be constructed. This is referred to as the histogram of the image. The tonal or radiometric quality of an image can be assessed from its histogram. An image that makes good use of the available range of brightness values has a histogram with occupied bins over its full range, but without significantly large bars at black or white.

An image has a unique histogram but the reverse is not true in general since a histogram contains only radiometric and no spatial information. A point of importance is that the histogram can be viewed as a discrete probability distribution since the relative height of a particular bar indicates the chance of finding a pixel with that particular brightness value somewhere in the image. It is useful in what is to follow if the histogram is regarded as a continuous function of a continuously varying brightness value. While this is not the case for digital imagery, such a concept simplifies the derivations and leads to results that are easily interpreted in the digital form. A continuous histogram function $h(x)$ is illustrated where x is the continuous variable representing brightness value, and h

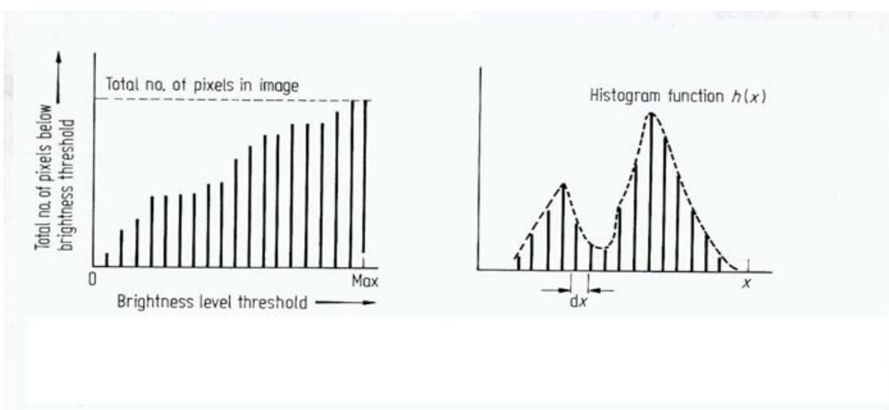


Fig. 4.1 – Illustration of a cumulative histogram and a continuous histogram

represents the value of the discrete histogram at that brightness. Strictly the height of a bar in the discrete histogram will be $h(x)dx$ where dx is the brightness value increment.

When the number of brightness values is very large $dx \cong 1$. However in general it can be shown that

$$dx = \frac{(L-1)}{L}$$

Where L is the total number of brightness values. In statistical terms then $h(x)$ is a probability density function and $\int h(x)\, dx$ is the cumulative probability function [6].

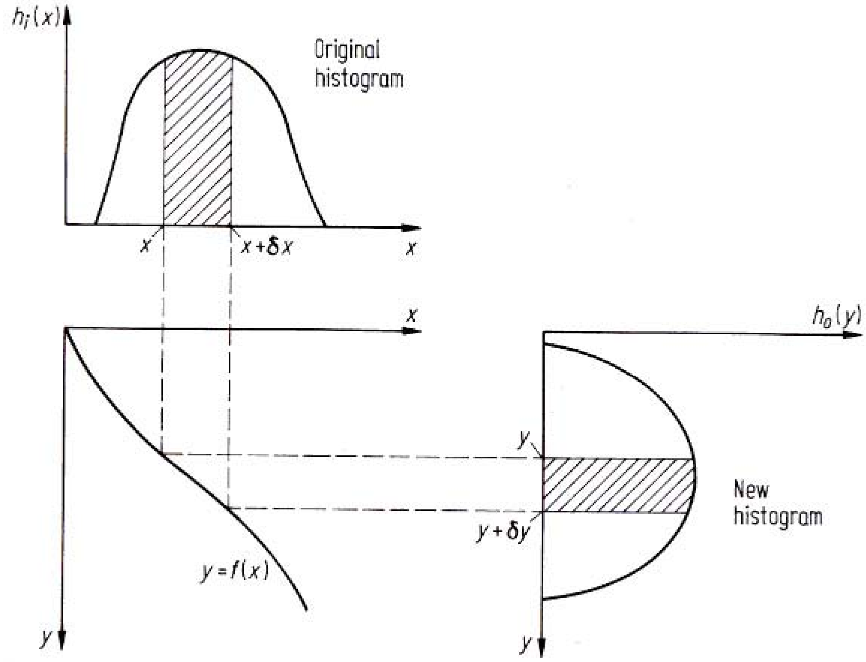
4.4 Contrast Modification in an Image

If we have an image with poor contrast, and it is desired to obtain an image with high contrast, a histogram that has a good spread of bars over the available brightness range, a so-called contrast stretching of the image data is required. Often the degree of stretching desired is apparent. For example the original histogram may occupy brightness values between 40 and 75 and it may be wished to expand the range to the maximum possible, say 0 – 255. Even though the modification is obvious it is necessary to express it in mathematical terms in order to relegate it to a computer. *Contrast modification is a mapping of brightness values, in that the brightness value of a particular histogram bar is respecified more favorably.* The bars themselves are not altered in size, although in some cases some bars may be mapped to the same new brightness value and will be superimposed. In general, however, the new histogram will have the same number of bars as the old one. They will simply be at different locations.

The mapping of brightness values associated with contrast modification can be described as

$$y = f(x)$$

where x is the old brightness value of a particular bar in the histogram and y is the corresponding new brightness value. Let $h_i(x)$ be the histogram function of the original image and $h_o(y)$ be the histogram function of the contrast-modified image. The subscripts i and o here are meant to connote the "input" histogram to a contrast modification process, and the resulting "output" histogram. The design of such a process, and indeed an observation of the results of the radiometric modification of an image, follows from an answer to the query: knowing $h_i(x)$ and $y = f(x)$ what is the form of $h_o(y)$? The answer can be obtained by reference to Fig 4.2, which has been adopted, from Castleman [7].



Diagrammatic representation of contrast modification by the brightness value mapping function $y = f(x)$

Fig. 4.2 Contrast modification by brightness mapping

In Fig.4.2 the number of pixels represented by the range y to $y + \delta y$ in the modified histogram must, by definition in the diagram, be equal to the number of pixels represented in the range x to $x + \delta x$ in the original histogram. Given that $h_i(x)$ and $h_o(y)$ are strictly density functions, this implies

$$h_i(x)\delta x = h_o(y)\delta y$$

So that in the limit as δx, δy --> 0

$$h_o(y) = h_i(x)\frac{dx}{dy}$$

Since $y = f(x)$, $x = f^{-1}(y)$ and the above equation becomes

$$h_o(y) = h_i f^{-1}(y) \frac{d[f^{-1}(y)]}{dy} \qquad (4.1)$$

which is an analytical expression for the output histogram. It should be noted mathematically, that this requires the inverse $x = f^{1}(y)$ to exist. Should the inverse not exist – for example, if $y = f(x)$ is not monotonic – Castleman [7] recommends treating the original brightness value range x as a set of continuous sub ranges over each of which y = f(x) is monotonic. These are then treated separately and the resulting histograms added.

4.4.1 Contrast Enhancement Techniques

Several contrast enhancement techniques are frequently implemented and a few of them are outlined below.

4.4.1.A Linear Contrast Enhancement

As an illustration of contrast modification, consider the simple linear variation described by

$$y = f(x) = ax + b$$

so that

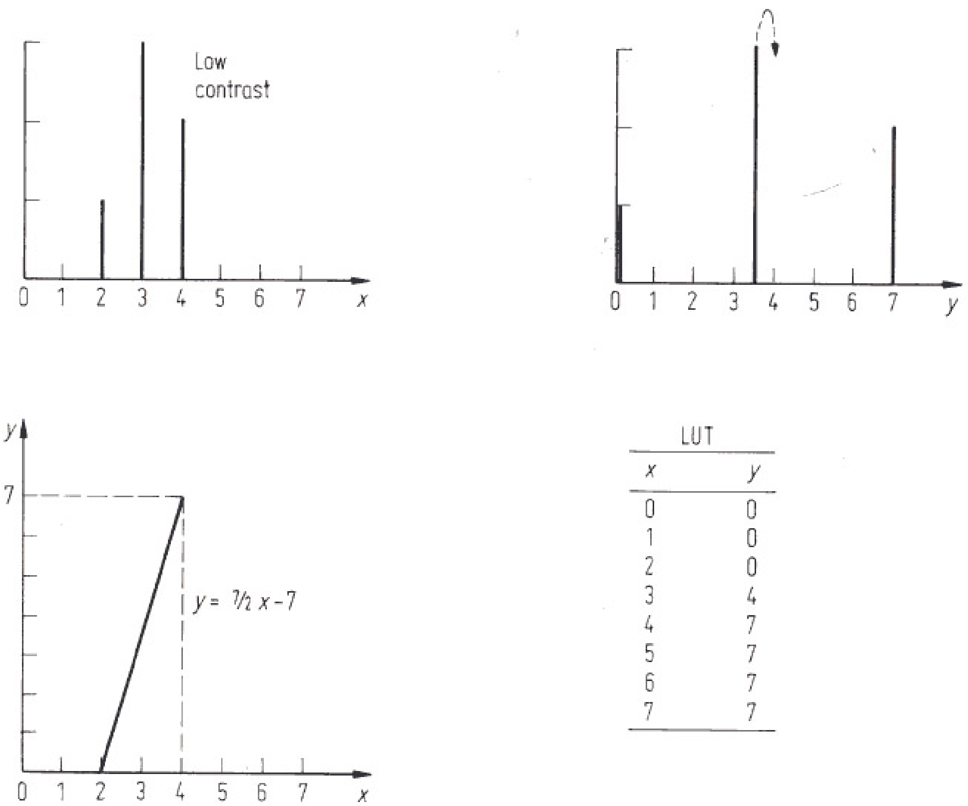
$$x = f^{-1}(y) = \frac{(y-b)}{a}$$

and

$$\frac{df^{-1}(y)}{dy} = \frac{1}{a}$$

The modified histogram therefore from equation (4.1) is

$$h_o(y) = \frac{1}{a} \times h_i \frac{(y-b)}{a}$$



LUT

x	y
0	0
1	0
2	0
3	4
4	7
5	7
6	7
7	7

Fig. 4.3. – Simple numerical example of linear contrast modification.

Relative to the original histogram, the modified version is shifted owing to the effect of *b*, is spread or compressed, depending on whether *a* is greater or less than 1 and is modified in amplitude. The last effect only relates to the continuous histogram function and cannot happen for discrete brightness value data. A numerical example of linear contrast modification is shown in fig.4.3. The look up table (LUT) has been included in the figure. In practice this would be used by a computer routine to produce the new image. This is done by reading the brightness values of the original version, pixel by pixel, substituting these into the left hand side of the table and then reading the new value for a pixel from the right hand side of the table. It is important to note in digital image handling that the new brightness values, just as the old must be discrete, and cover usually the same range of brightnesses. Generally this will require some rounding to integer form of the new brightness values calculated from the mapping function $y = f(x)$. A further point to note in the above example is that the look up table is undefined outside the range 2 to 4 of inputs.

To do so would generate output brightness values that are outside the range valid for this example [6].

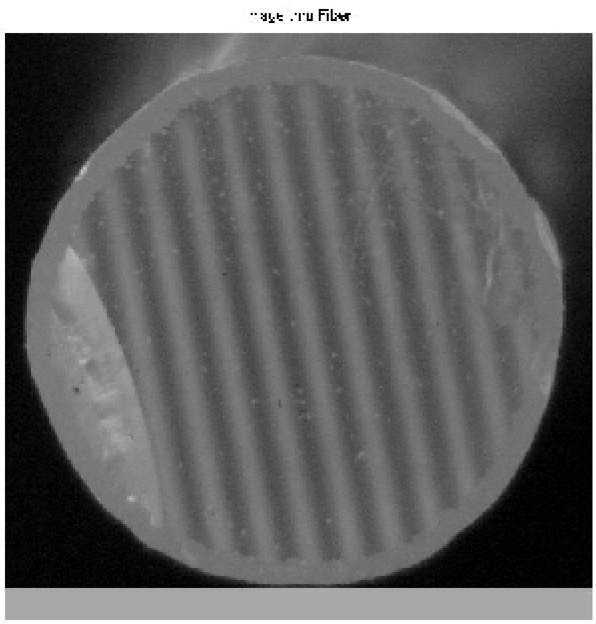


Fig 4.4

Image of the stripe test pattern seen through the fiber in ambient lighting conditions.

Now fig 4.4 shows the image as seen through the optical fiber under ambient lighting conditions. The image is dark and has poor contrast. Hence to improve the overall contrast of the image we use a contrast enhancement technique, specifically linear contrast enhancement. From the image histogram shown above we see that the image through the fiber utilizes a restricted range of brightness values leading to poor contrast

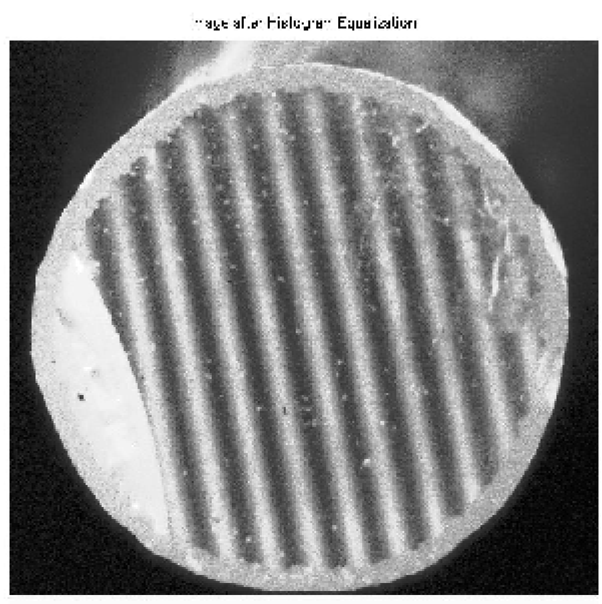


Fig 4.5

Image of the stripe test pattern after linear contrast enhancement.

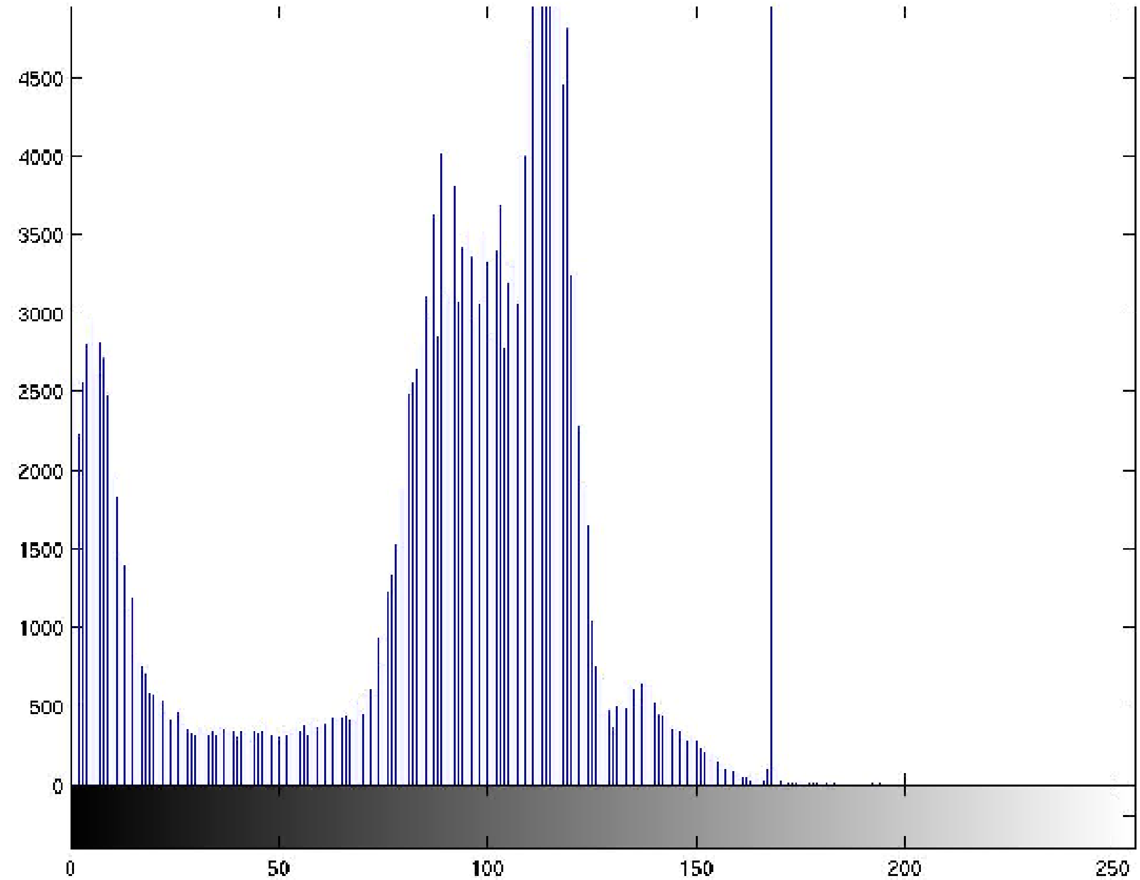


Fig 4.6. Input histogram corresponding to the figure 4.4

Fig 4.7 shows the image histogram of the output after implementing a linear contrast stretch, which distributes the pixel values over the entire available brightness range. The result is as shown in Fig.4.5. You can now see the stripe test pattern very clearly.

Another example of the usefulness of contrast enhancement is shown below. Particles of the sizes of around 2mm were imaged using the fused optical fiber bundle. Fig.4.8 shows the image of the particles under ambient lighting conditions. As seen, the image is dark and has a poor contrast. The image after implementing a linear stretch of the brightness values over the entire ranges which is from 0 – 256 are as shown in fig. 4.9.

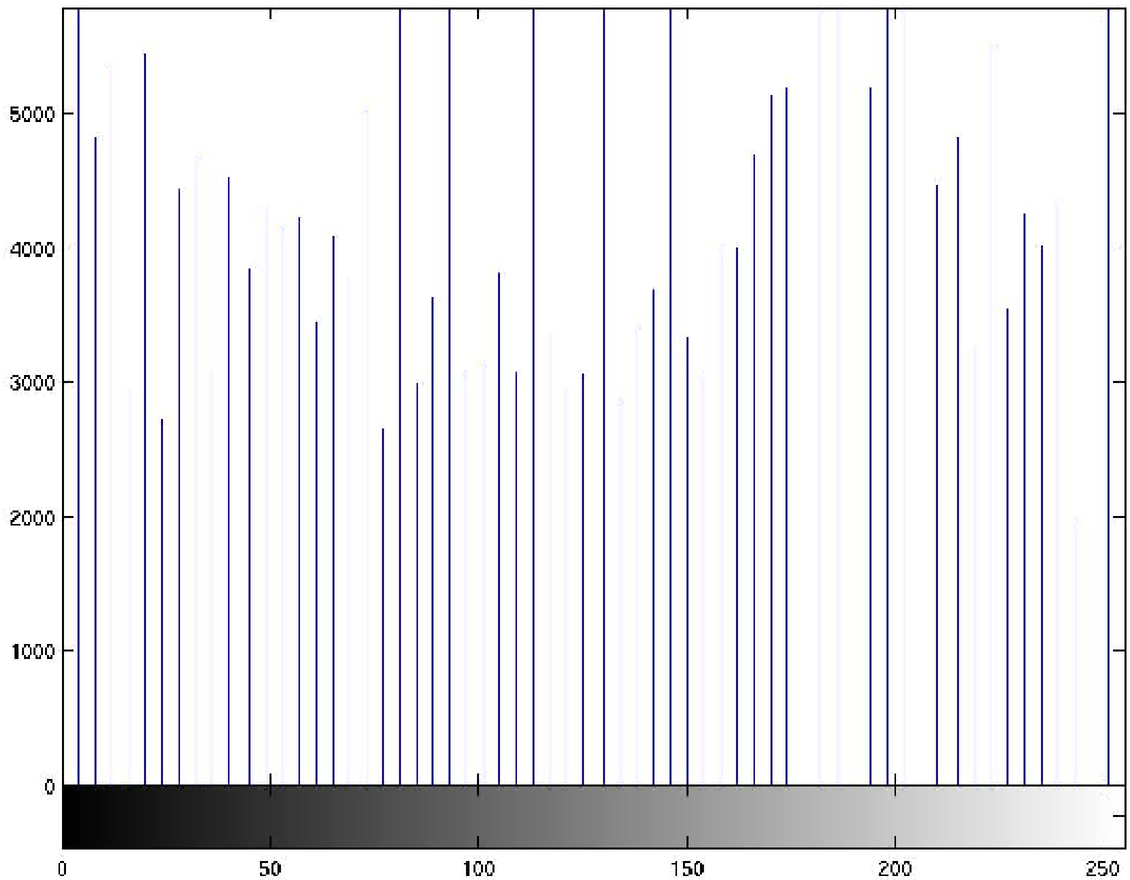


Fig 4.7 – Output histogram corresponding to figure 4.5

The image is clearer but we can see the imperfections on the fiber optical face also get enhanced. These imperfections however can be filtered out using suitable filtering techniques discussed later on in the chapter.

4.4.1.B Saturating Linear Contrast Enhancement

Frequently a better image product is given when linear contrast enhancement is used to give some degree of saturation at the black and white ends of the histogram. Such is the case, for example, if the darker regions in an image correspond to the same background type within which small radiometric variations are of no interest. Similarly, a region of

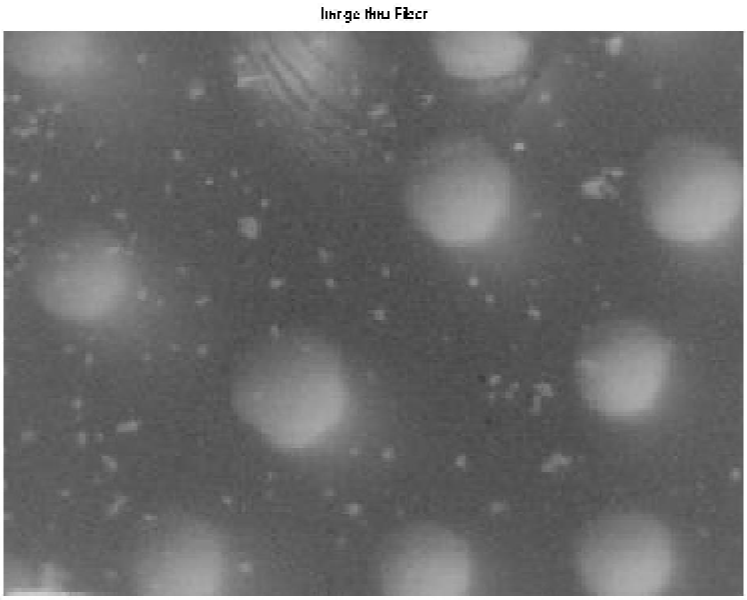


Fig.4.8

Image of white particles as seen through the optic fiber in ambient lighting conditions.

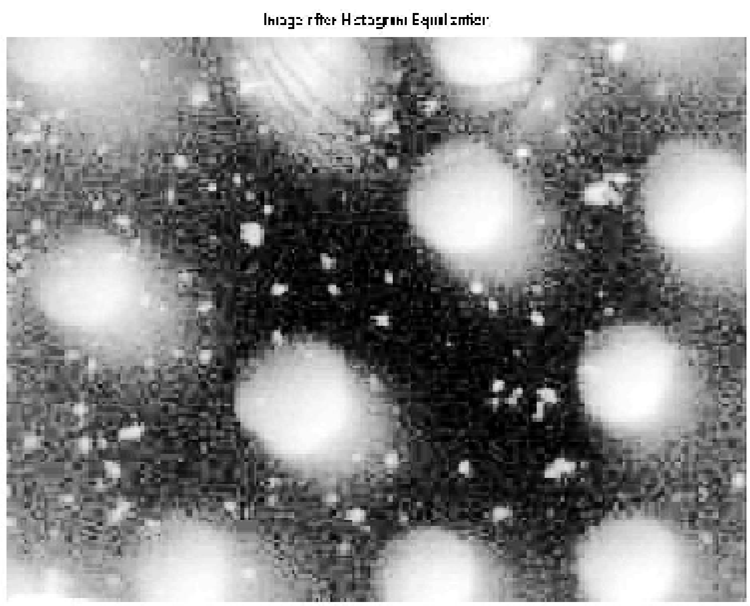


Fig. 4.9.

Image of the particles after histogram equalization

interest in an image may occupy a restricted brightness value range; saturating linear contrast enhancement is then employed to expand that range to the maximum possible dynamic range of the display device with all other regions being mapped to black or white.

The brightness value mapping function y = f(x) for saturating linear contrast enhancement is shown above, in which B_{max} and B_{min} are the user-determined maximum and minimum brightness values that are to be expanded to the lowest and highest brightness levels supported by the display device.

4.4.1.C Automatic Contrast Enhancement

Image display systems frequently implement an automatic contrast stretch on the raw data in order to give an output with good contrast. Such a procedure is also of value when displaying image data on other output devices as well, at least for the first look of the data before further processing.

Typically the auto contrast enhancement procedure is a saturating linear stretch. The cut off and saturation limits B_{min} and B_{max} are chosen by determining the mean brightness of the raw data and its mean standard deviation and then making B_{min} equal to the mean less three standard deviations and B_{max} equal to the mean plus three standard deviations.

4.4.1.D Logarithmic and Exponential Contrast Enhancement.

Logarithmic and exponential mappings of brightness values between original and modified images are useful for enhancing dark and light features respectively. The mapping functions are depicted in Fig.4.10 along with their mathematical expressions. It is particularly important with these that the output values be scaled to lie within the range of the device used to display the image and that the output values be rounded to allow discrete values

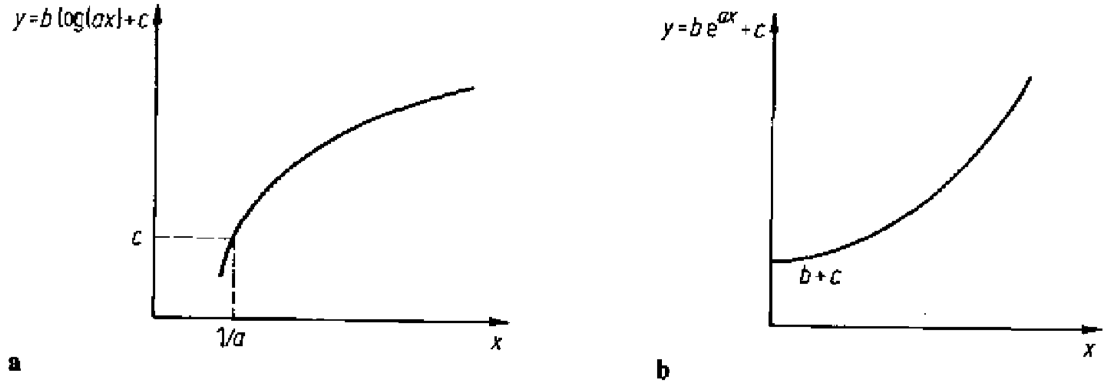


*Fig 4.10 – Logarithmic **a** and exponential **b** brightness mapping functions. The parameters a, b and c are usually included to adjust the overall brightness and contrast of the output.*

4.4.1.E Piecewise Linear Contrast Modification

A particularly useful and flexible contrast modification procedure is the piecewise linear mapping function shown in the Fig. 4.11. This is characterized by a set of user specified break points as shown. Generally the user can also specify the number of break points desired.

In contrast to the point operations used for radiometric enhancement, techniques for geometric enhancement are characterized by operations over neighborhoods. The procedures still determine modified brightness values for an image's pixels; however, the new value for the pixel is derived from the brightness of a set of the surrounding pixels. It is this spatial interdependence of the pixel values that leads to variations in the perceived image geometric detail.

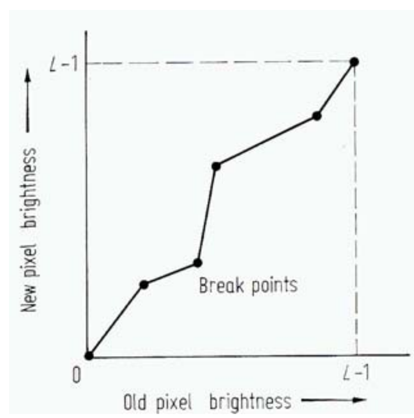


Fig. 4.11.

Piecewise linear contrast modification function, characterized by the break points that are user specified.

4.5 Template Operators

Geometric enhancements of most interest usually relate to smoothing, sharpening, edge or line detection. We have used geometric enhancement techniques for smoothening, sharpening of our images. Most of the methods can be expressed as template techniques in which a template, box or window is defined and then moved over the image row-by-row or column-by-column. The products of the pixel brightness values, covered by the template at a particular position, and the template entries are taken and summed to give a template response. This response is then used to define a new brightness value for the pixel currently at the center of the template. When this is done for every pixel in the image, a radiometrically-modified image is produced that enhances or smoothes geometric features according to the specific numbers loaded into the template. A 3 × 3 template is illustrated in the Fig 4.12.

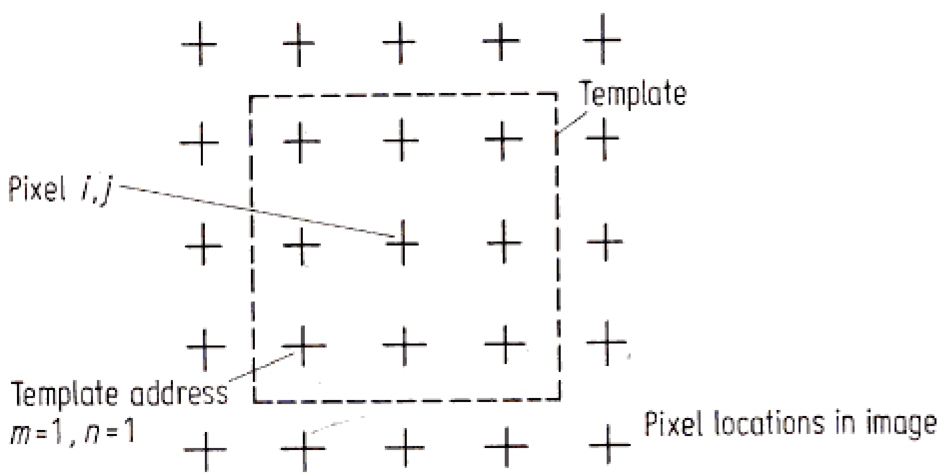


Fig. 4.12 – a 3 × 3 template positioned over a group of nine image pixels, showing the relative locations of pixels and template entry address.

Templates of any size can be defined, and for an M by n pixel sized template, the response for the image pixel i, j is

$$r(i,j) = \sum_{m=1}^{M}\sum_{n=1}^{N} \phi(m,n) t(m,n) \qquad (4.2)$$

Where $\phi(m,n)$ is the pixel brightness value, addressed according to the template position and $t(m,n)$ is the template entry at that position. Often the template entries collectively are referred to as the 'kernel' of the template and the template technique generally is called as convolution, in view of its similarity to time domain convolution in linear system theory.

4.5 Image Domain Versus Fourier Transform Approaches.

Most geometric enhancement procedures can be implemented using either the Fourier transform approach or the image domain procedures of this chapter. Several factors such as available software, familiarity with each method including its limitations and idiosyncrasies, and ease of use determine the option to be used. A further consideration relates to computer processing time.

Both the Fourier transform, frequency domain process and the template approach consist only of sets of multiplication and additions. No other mathematical operations are involved. It is sufficient, therefore from the point of view of cost, to make a comparison based upon the number of multiplications and number of additions necessary to achieve a result. The additions can be ignored, as they are generally faster than multiplication for most computers and also since they are comparable in number to the multiplications involved.

For an image of K × K pixels (a square image is considered for simplicity) and a template of size M × N the total number of multiplications necessary to evaluate (4.2) for every image pixel is

$$N_C = MNK^2$$

The number of complex multiplications required in the frequency domain approach is,

$$N_F = 2K^2 \log_2 K + K^2$$

A cost comparison therefore is

$$\frac{N_C}{N_F} = \frac{MN}{(2\log_2 K + 1)}$$

When this figure is less than 1 it is more economical to use the template operator approach. Otherwise the Fourier transformation procedure is more cost-effective. Clearly this does not take into account program overheads (such as bit shuffling required in the frequency domain approach, how data is buffered into computer memory from the disc for processing) and the added cost of complex multiplications; however is a reasonable starting point in choosing between the methods.

One important point in this comparison is that the frequency domain method is able to implement processes not possible (or at least not viable) with template operators. Removal of periodic noise is one example. This is particularly simple in the spatial frequency domain but requires unduly complex templates or even nonlinear operators (such as median filtering) in the image domain. However the template approach is a popular one since often 3×3 and 5×5 templates are sufficient to achieve desired results [6].

4.7 Image Smoothing (Low Pass Filtering)

Images can contain random noise superimposed on the pixel brightness values owing to noise generated in the transducers that acquire the image data, systematic quantization noise in the signal digitizing electronics and noise added to the video signal during transmission.

4.7.1 Low Pass Filtering

Usually this can be removed by the process of low pass filtering or smoothing, unfortunately sometimes at the cost of high frequency information.

4.7.1.A Mean Value Smoothing.

To smooth an image a uniform template in (4.2) is used with entries

$t(m,n) = \frac{1}{MN}$ for all m,n

so that the template response is a simple average of the pixel brightness values currently within the template, viz

$$r\ (i,j) = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\phi(m,n) \ldots\ldots(4.3)$$

The pixel at the center of the template is thus represented by the average brightness level in a neighborhood defined by the template dimensions. This is an intuitively obvious template for smoothing and is equivalent to using running averages for smoothing time series information. It is evident that high frequency information such as edges will also be averaged and lost. This loss of high frequency information such as edges will also be averaged and lost. This loss of high frequency detail can be circumvented somewhat if a threshold is applied to the template response in the following manner.

Let

$$\rho(i,j) = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\phi(i,j)$$

then

$$r(i,j) = \rho\ (i,j) \text{ if } |\varphi\ (i,j) - \rho\ (i,j)\ | < \mathrm{T}$$
$$= \varphi\ (i,j) \text{ otherwise.}$$

where T is the pre-specified threshold. T could be determined a priori based upon knowledge of or an estimate of the scene signal to noise ratio.

Eliason and McEwan [8] recommend choosing the threshold as a multiple of the standard deviation of brightness within the template window. This provides better noise removal in homogenous regions while allowing preservation of other high frequency spatial detail. A simple illustration of image smoothing by averaging over a template, both with and without the application of a threshold, is given in the fig 4.13 below.

For clarity this is based upon a hypothetical one-dimensional image, or alternatively a single line of image data, with which a 3 $\times$ 1 template is used. In this manner the actual numerical modification of pixel brightness values can be observed.

In principle, templates of any shape and size can be used. Larger templates give more smoothing (and greater loss of high frequency detail) whereas horizontal rectangular templates will smooth horizontal noise but leave noise and high frequency detail in the vertical direction relatively unaffected by comparison.

Commonly, smoothing by template methods is referred to as boxcar filtering. When based upon equation (4.3) it is also called as the mean value smoothing, or averaging

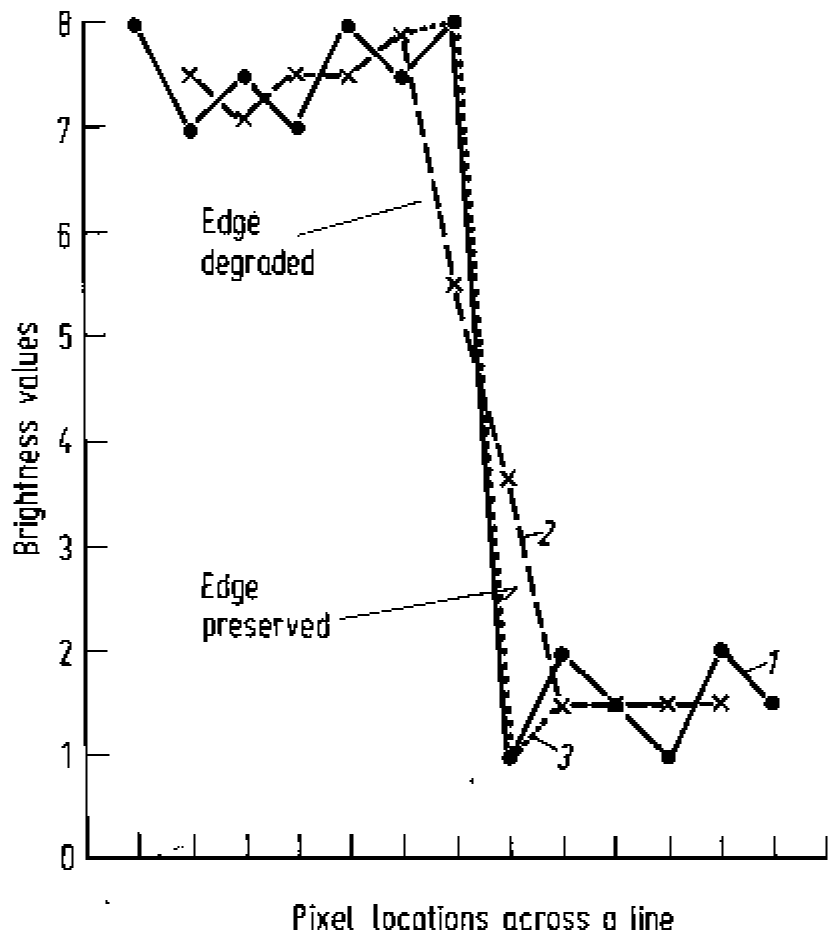


Fig 4.13 – Illustration of the effect of a 3 × 1 averaging across a single line of image data with and without thresholding.

.

As an example of mean value smoothing or averaging is the image shown below in Fig.4.14. This image is the output of our stripe test pattern imaged earlier in Fig.4.4. It can be seen that the image is much smoother than the contrast enhanced image in Fig.4.5. It can also be observed that some of the fiber imperfections have been smoothened out.

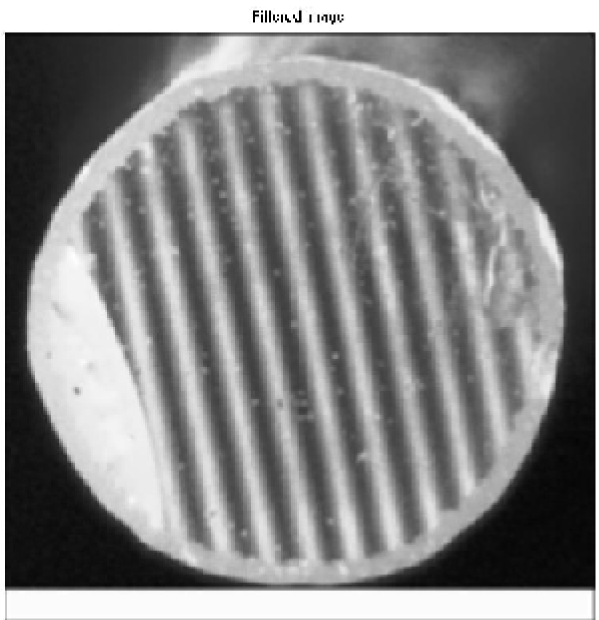


Fig. 4.14

The test strip pattern after using mean or average filtering.

4.7.1.B Median Filtering

Disadvantages of the thresholding method for avoiding high frequency detail deterioration are that it adds to the computational cost of the smoothing operation and T must be determined. An alternative technique for smoothing in which the edges in an image are maintained is that of median filtering. In this the pixel at the center of the template is given the median brightness vale of all the pixels covered by the template – i.e. that value which ahs as many values higher and lower. (For e.g. median of 4,6,3,7,9,2,1,8,8 is 6, whereas the mean is 5.3). Figure shows the effect of median filtering on a single line of image data compared with simple boxcar averaging, which uses the mean of pixel brightness values. Again, it can be seen that most of the original high frequency detail is preserved.

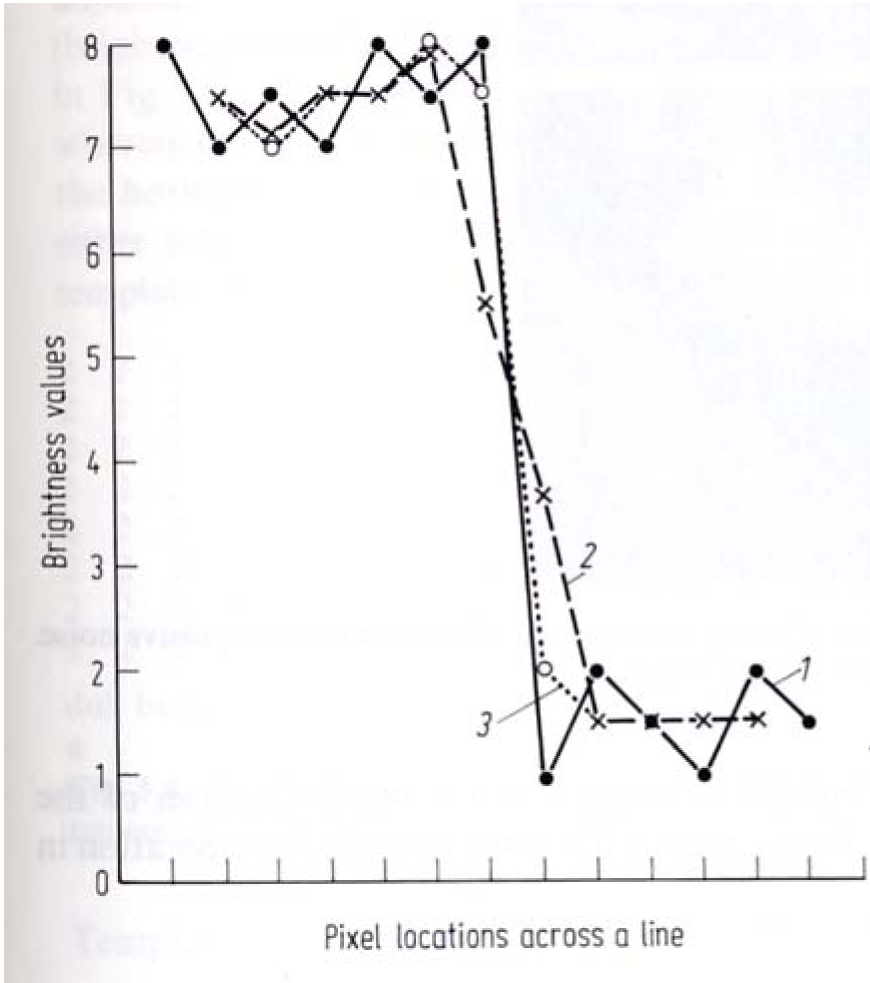


Fig 4.15

Comparison of a simple averaging and median filtering of a single line of image data.
1. Original image 2. 3 x 1 smoothing 3. 3 x 1 median filtering

An application for which the median filtering technique is well suited is the removal of impulse like noise. This is because pixels corresponding to noise spikes are atypical in their neighborhood and will be replaced by the most typical pixel in that neighborhood. Finally it should be noted that median filtering is not a linear function of the brightness values of the image pixels.

Fig. 4.16 and Fig. 4.17 are examples of median filtering implemented on the test stripe pattern and the particles imaged respectively. You can see that the images are now freer of the fiber imperfections that show up as dots or lines in the original unfiltered image.

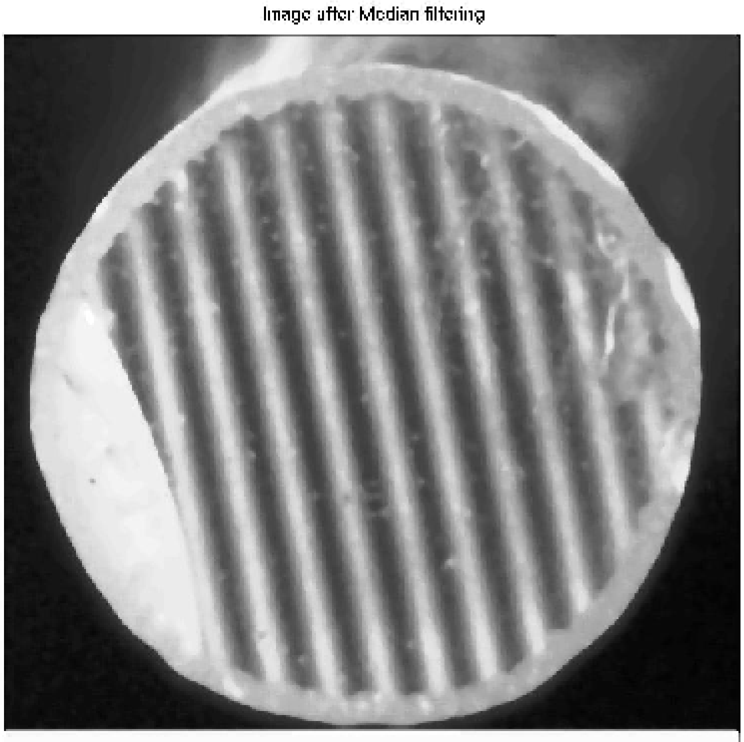


Fig. 4.16

Image of the stripe test pattern after median filtering.

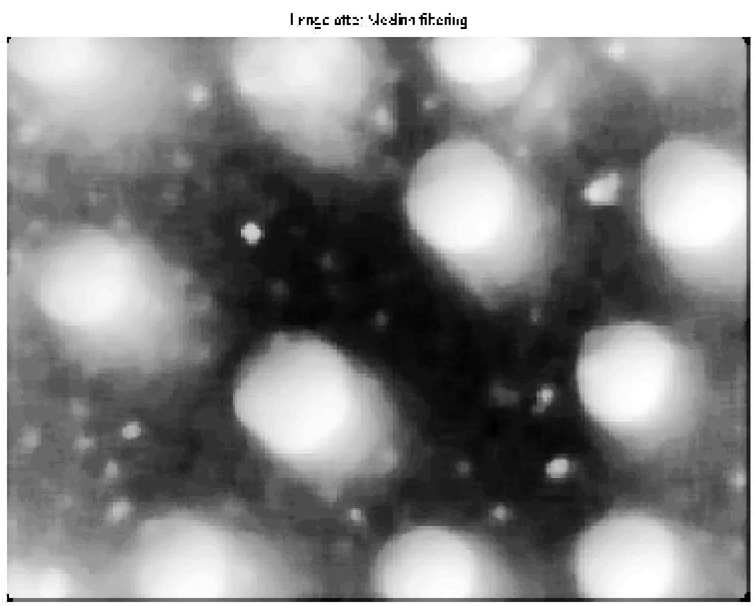


Fig. 4.17

Image of white particles after median filtering.

4.7.2 High Pass Filtering

High Pass Filtering emphasizes the high frequency components of a signal while reducing the low frequency components. Because the high frequency components of a signal generally correspond to edges or fine details of an image, high pass filtering often increases the local contrast and thus sharpens the image. Images with sharper edges are often more pleasant visually.

High pass filtering is also useful in reducing image blur. When an image $x(n1,n2)$ is degraded by blurring due to misfocus of the lens, motion, or atmospheric turbulence, the blurred image $y(n1,n2)$ can be represented by

$$y(n1,n2) = x(n1,n2) * b(n1,n2)$$

or

$$Y(w1, w2) = X(w1, w2)\, B(w1, w2)$$

Where $b(n1, n2)$ is the point spread function of the blur, and $B(w1, w2)$ is its Fourier transform. Because $B(n1, n2)$ is typically of low pass character, one approach of reducing blur is to use a high pass filter.

High- pass filtering can also be used in preprocessing an image prior to its degradation by noise. In applications such as image coding, an original un-degraded image is available for processing prior to its degradation by noise such as quantization noise. In such applications, the undegraded image can be high pass filtered prior to its degradation by noise such as quantization noise. The usual effect of this is an improvement in the quality of the resulting image. For example, when the degradation is due to the wideband random noise, the effective SNR of the degraded image is much lower in the high frequency components than in the low frequency components, due to the low-pass character of a typical image. High pass filtering prior to the degradation significantly improves the SNR in the high frequency components at the expense of a small SNR decrease in the low frequency components. This process typically results in quality improvement of the resulting image. Some typical examples of the impulse response of a high pass filter used for image enhancement are shown in fig.4.18.

0	-1	0
-1	5	-1
0	-1	0

1	-2	1
-2	5	-2
1	-2	1

-1	-2	-1
-2	19	-2
-1	-2	-1

Fig 4.18 Templates for impulse responses of a high pass filter

4.6.4 Adaptive Filtering

In many applications requiring filtering, the necessary frequency response may not be known beforehand, or it may vary with time. (Example; suppression of engine harmonics in a car stereo.) In such applications, an adaptive filter which can automatically design itself and which can track system variations in time is extremely useful. Adaptive filters are used extensively in a wide variety of applications, particularly in telecommunications and image analysis. As an example of adaptive filtering we implement the Wiener filter.

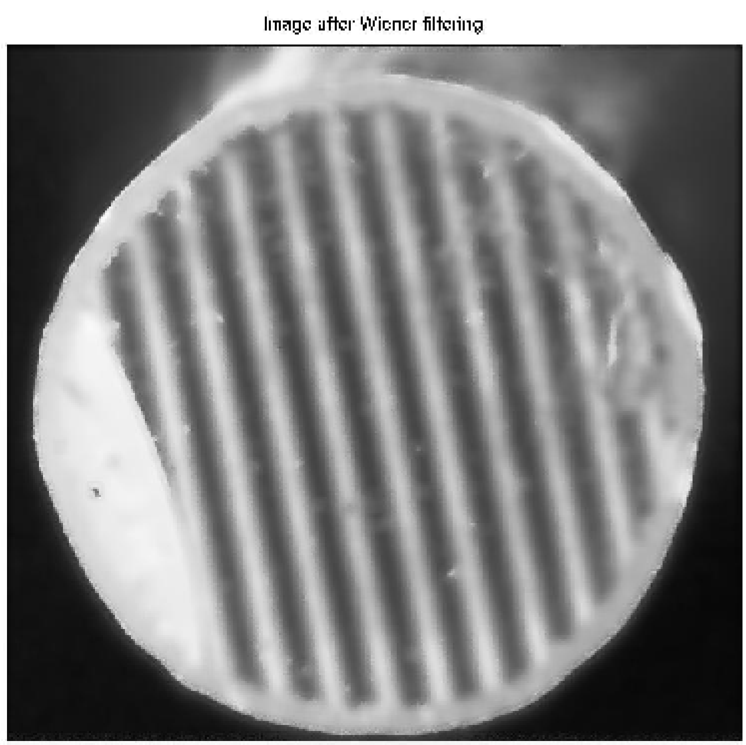


Fig. 4.19

Stripe test pattern after wiener filtering

As we can see from the two figures, the one above and below wiener filtering smoothens out the image and gets rid of almost all the imperfections that are simply treated as noise. Below in the figure we see that some of the fiber imperfections are so large that they are not eliminated. These however can be removed by re-implementing the filters on a filtered image. This would result in a highly smoothened image, with the particles almost smoothened into the image. However, the number of iterations is determined by the application for which the image is being enhanced. In chapter 5 we shall see that the

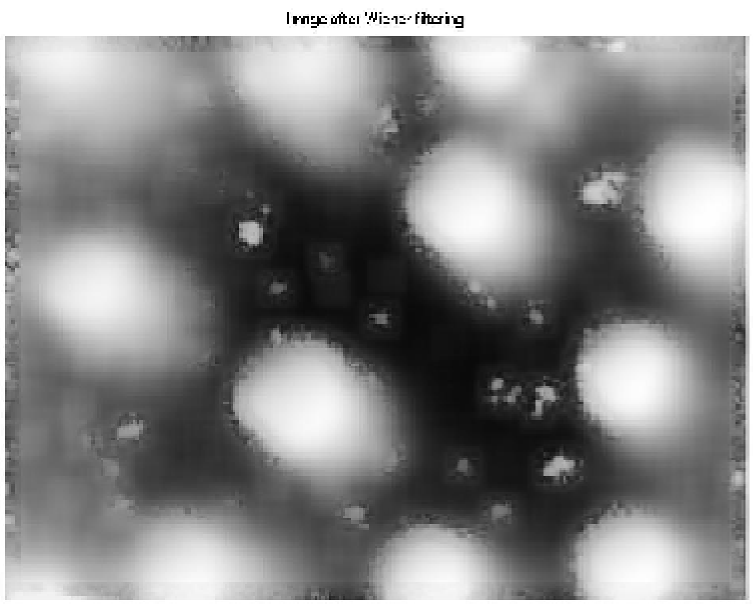


Fig.4.20

Wiener filtering of the white particles.

image is so smooth that the particles almost get smoothened into the image. This process successfully removes all the unwanted noise and the particles are distinct enough to be used to develop a particle counting algorithm.