CHAPTER 3

SPATIAL IMAGE ENHANCEMENT

3.1 INTRODUCTION

The basic aim of enhancement is to make the image look better so that the resulting image is more suitable than the original image for road traffic analysis.

Enhancement techniques are problem oriented i.e. the enhancement technique used to enhance an X ray image may not be the best technique to enhance a space image. Enhancement techniques cannot be generalized because viewer is the ultimate judge as to how well a particular technique works. Visual evaluation is totally subjective. Hence, enhancement process is highly subjective.

There are two categories of image enhancement, namely,

a) Spatial Domain Method: Direct manipulation of the pixels in the image is done and

b) Frequency Domain Method: Fourier transform of the image is modified.
Spatial Image Processing Algorithm

3.4
Spatial Image Enhancement

Function without losing any information.

Representation allows one to work and remain in the original domain of
completely via an inverse process, without loss of information. This
expressed in another Fourier transformation or series can be reconstructed.

Functions way in study and improvement image enhancement approaches. Functions
Fourier transformation. Fourier transformation provides a meaningful and practical
image enhancement can be carried out in frequency domain using

\[ f(x, y) \] is the input image.

\[ \hat{f}(\xi, \eta) \] is the Fourier transform of \( f(x, y) \).

\[ \hat{f}(\xi, \eta) \] is the frequency domain representation of \( f(x, y) \).

\[ f(x, y) = \mathcal{F}^{-1} \{ \hat{f}(\xi, \eta) \} \]

Where

\[ \mathcal{F}^{-1} \{ \hat{f}(\xi, \eta) \} \]

domain process is denoted by the expression

computing an image. These methods operate directly on pixels. Spatial

The term spatial domain refers to the aggregate of pixels:

3.2 Spatial Domain Method

104
1. The desired range of gray levels is highlighted background
2. The rest of gray values are displayed by low value
3. Gray levels in the range of interest are displayed with a high value

Image is highlighted. Two algorithms are used:

- Conversion to gray scale images and the specific range of gray levels in an
- Selective conversion color images permitting to road network are

![Gray level Shading](image)

Figure 3.1 Image enhancement algorithm

5. Cropped images are enhanced using median filter
Histograms are used in real-time image processing and are easy to calculate in software.

where \( L \) is the grey level and \( n_k \) is the number of pixels in the image.

\[
H(L) = \frac{n_k}{L}
\]

The histogram of a digital image with different grey levels are represented as a discrete function.

3.4.2.2 Histogram Processing

Background details, level shifting, in which needed grey levels are increased to preserve the enhanced image in the Figure 3.2(b), the input images are enhanced using gray level shifting.

The Figure 3.2(a) shows the spatial initial image before enhancement.

Figure 3.2: Spatial Image enhancement using gray level shifting

(a) Before enhancement

(b) After enhancement
(a) Before histogram processing    (b) After histogram processing

Figure 3.3 Spatial Image enhancement using histogram processing

The Figure 3.3(a) shows spatial image before histogram processing. The Figure 3.3(b) shows spatial image after histogram processing, that is, certain gray levels in the histogram are increased to increase the quality of the image.

3.4.1.3 Spatial image resizing

To resize an image, the image resize function is used. To resize an image, the image to be resized and the magnification factor are specified as input parameter. To enlarge an image, magnification factor greater than 1 is specified. To reduce an image, specify a magnification factor between 0 and 1 is specified. The figure 3.4(a) shows a spatial image before resizing. In Figure 3.4(b) the image is resized to 1.25 times , so that the image details appears to be large and clear in the output.
3.4.1.4 Spatial image rotation

To rotate an image, the `imrotate` function is used. To rotate an image, the image to be rotated and the rotation angle are specified in degrees. If the specification is a positive rotation angle, `imrotate` rotates the image counterclockwise and if it negative rotation angle, `imrotate` rotates the image clockwise.

By default, `imrotate` creates an output image large enough to include the entire original image. Pixels that fall outside the boundaries of the original image are set to 0 and appear as a black background in the output image. One can, however, specify that the output image be the same size as the input image, using the 'crop' argument. Similarly, `imrotate` uses nearest-neighbor interpolation by default to determine the value of pixels in the output image, but one can specify other interpolation methods.

With the help of spatial image rotation the traffic patterns can be placed either horizontal or vertical.
Figure 3.5 Spatial image rotation

Figure 3.5(a) shows the original image before rotation. Figure 3.5(b), the original image has been rotated at an angle 95° counterclockwise.

3.4.1.5 Spatial image cropping

To extract a rectangular portion of an image, imcrop function is used. Using imcrop, the crop region is specified interactively using the mouse or programmatically by specifying the size and position of the crop region.

This example in Figure 3.6 illustrates an interactive syntax. The example reads an image into the MATLAB workspace and calls imcrop specifying the image as an argument. The imcrop displays the image in a figure window and waits one draws the crop rectangle on the image. When
one moves the pointer over the image, the shape of the pointer changes to cross hairs +. Click and drag the pointer to specify the size and position of the crop rectangle., one can also move and adjust the size of the crop rectangle using the mouse. When one is satisfied with the crop rectangle, double-click to perform the crop operation.

(a) Image before cropping           (b) Image after cropping

Figure 3.6 Spatial image cropping

The Figure 3.6(a) shows images before cropping. In Figure 3.6(b), the traffic portion of the image along the road alone is cropped from the original image.

3.4.1.6 Spatial filtering

Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and
preserve edges. \( B = \text{medfilt2}(A, [m \ n]) \) performs median filtering of the matrix \( A \) in two dimensions where \( A \) is the gray level matrix. Each output pixel contains the median value in the \( m \)-by-\( n \) neighborhood around the corresponding pixel in the input image. The \text{medfilt2} pads the image with 0s on the edges, so the median values for the points within \([m* n]/2\) of the edges might appear distorted.

(a) Image before median filtering

(b) Image after median filtering

Figure 3.7 Median Filtering

The Figure 3.7(a) shows traffic image before median filtering. In the Figure 3.7(b) shows traffic image after median filtering.
Figure 3.8 Enhanced and Filtered images for dimension d1

The Figure 3.8, road traffic images collected for dimension d1, near to location Solinganalur and Kellambakkam, Chennai east central zone.
Figure 3.9  Enhanced and filtered images for dimension d2
The Figure 3.9 shows images collected for dimension d2 near to location Solinganalur and Kellambakkam, Chennai east central zone.

Figure 3.10 Enhanced and Filtered images for dimension d3
The Figure 3.10, shows images collected for dimension d3 near to location Thuraipakkam, ECR road Chennai west zone.

Figure 3.11 Enhanced and filtered images for dimension d4
The Figure 3.11, shows images collected for dimension d4 near to location Injambakkam, ECR road Chennai west zone.

Figures 3.8 to 3.11 shows the results of algorithm of Figure 3.1 applied to different road traffic images collected from dimensions d1, d2, d3 and d4. Images are collected from Chennai east, where the Chennai east is divided into four zones namely north east, south east, central east, east west (or) west zone and each zone is again divided into further four zones forming quad tree. The dimensions d1, d2 corresponds to images collected from Chennai east central zones and the dimensions d3 and d4 corresponds to images collected from Chennai west zone. These images are collected from google map and road accident data management system. The captured images along with their dimensions are given as input to further chapters.

3.5 SUMMARY

In this chapter, the satellite captured road traffic spatial images for different dimensions are converted to gray scale images and the needed portion of traffic regions are extracted by using the cropping function and finally extracted road traffic images are enhanced using median filter to increase the visual quality. This enhanced spatial traffic images are used as an input to falling chapter “An efficient approach to project clusters in high dimensional road traffic images”.
