CLASS : T.E. E &TE **SUBJECT: DIP**

EXPT. NO. : 3 **DATE:**

TITLE: PIECEWISE LINEAR TRANSFORMATION ON AN IMAGE

CO 1:	Apply the fundamentals of digital image processing to perform various image-enhancement and image segmentation operations on gray scale image.

AIM :

To implement:

1. Contrast stretching
2. Bit Plane slicing
3. Gray level slicing

SOFTWARES REQUIRED: Jupyter Notebook / Google Colaboratory.

THEORY:

3.1 Gray-level Slicing:

Highlighting a specific range of gray levels in an image often is desired. Applications include enhancing features such as masses of water in satellite imagery and enhancing flaws in X-ray images. There are several ways of doing level slicing, but most of them are variations of two basic themes. One approach is to display a high value for all gray levels in the range of interest and a low value for all other gray levels. This transformation, shown in Fig. 3.4(a), produces a binary image.

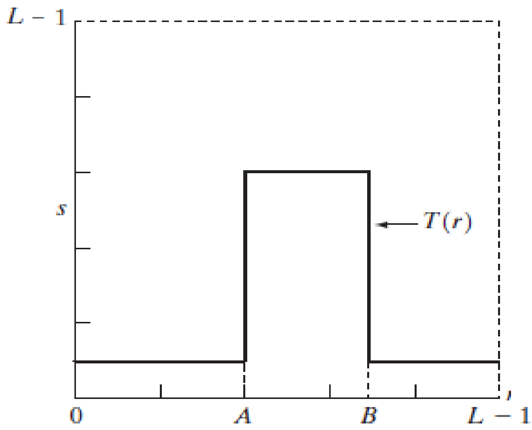

Fig 3.1(a)

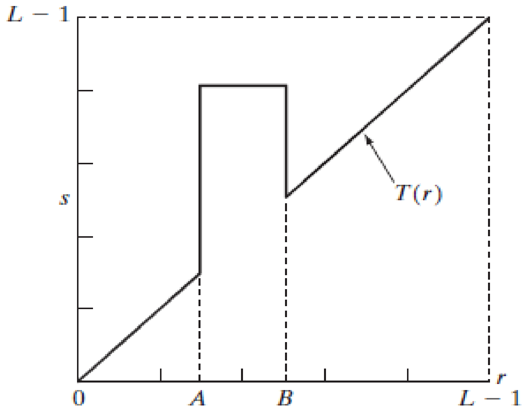

Fig 3.1(b)

The second approach, based on the transformation shown in Fig. 3.4(b), brightens the desired range of gray levels but preserves the background and gray-level tonalities in the image

3.2 Bit Plane Slicing:

Instead of highlighting gray-level ranges, highlighting the contribution made to total image appearance by specific bits might be desired. Suppose that each pixel in an image is represented by 8 bits. Imagine that the image is composed of eight 1-bit planes, ranging from bit-plane 0 for the least significant bit to bit-plane 7 for the most significant bit. In terms of 8-bit bytes, plane 0 contains all the lowest order bits in the bytes comprising the pixels in the image and plane 7 contains all the high-order bits. Figure 3.5 illustrates these ideas. Note that the higher-order bits (especially the top four) contain the majority of the visually significant data. The other bit planes contribute to more subtle details in the image. Separating a digital image into its bit planes is useful for analyzing the relative importance played by each bit of the image, a process that aids in determining the adequacy of the number of bits used to quantize each pixel. Also, this type of decomposition is useful for image compression.

In terms of bit-plane extraction for an 8-bit image, it is not difficult to show that the (binary) image for bit-plane 7 can be obtained by processing the input image with a thresholding gray-level transformation function that (1) maps all levels in the image between 0 and 127 to one level (for example, 0); and (2) maps all levels between 129 and 255 to another (for example, 255).

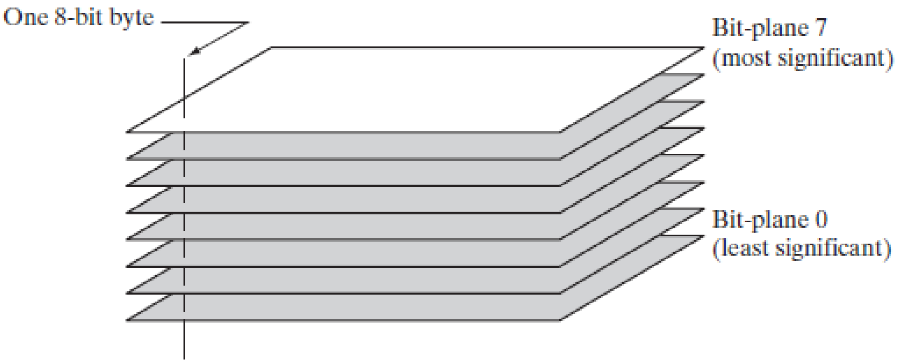

Figure 3.2: Bit Plane Representation of 8 bit Image.

3.3 Contrast Stretching:

One of the simplest piecewise linear functions (function whose pieces are linear) is a contrast stretching transformation. The form of the piecewise function can be arbitrary complex. A practical implementation of some important transformation can be formulated only as piecewise function. It increases the dynamic range of the gray levels in the image being processed. Low contrast images are resulted from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture of image acquisition.

Transformation function is shown in Fig. 3.6. Locations of the points (r1, s1) and (r2, s2) used to control the shape of the transformation function. If r1=s1 and r2=s2 transformation is linear function and produces no changes in the gray level. If r1=r2, s1=0 and s2=L-1, transformation becomes a thresholding function that results a binary image.

Intermediate values of (r1, s1) and (r2, s2) produce various degrees of spread in the gray levels of the output image, thus affecting its contrast.

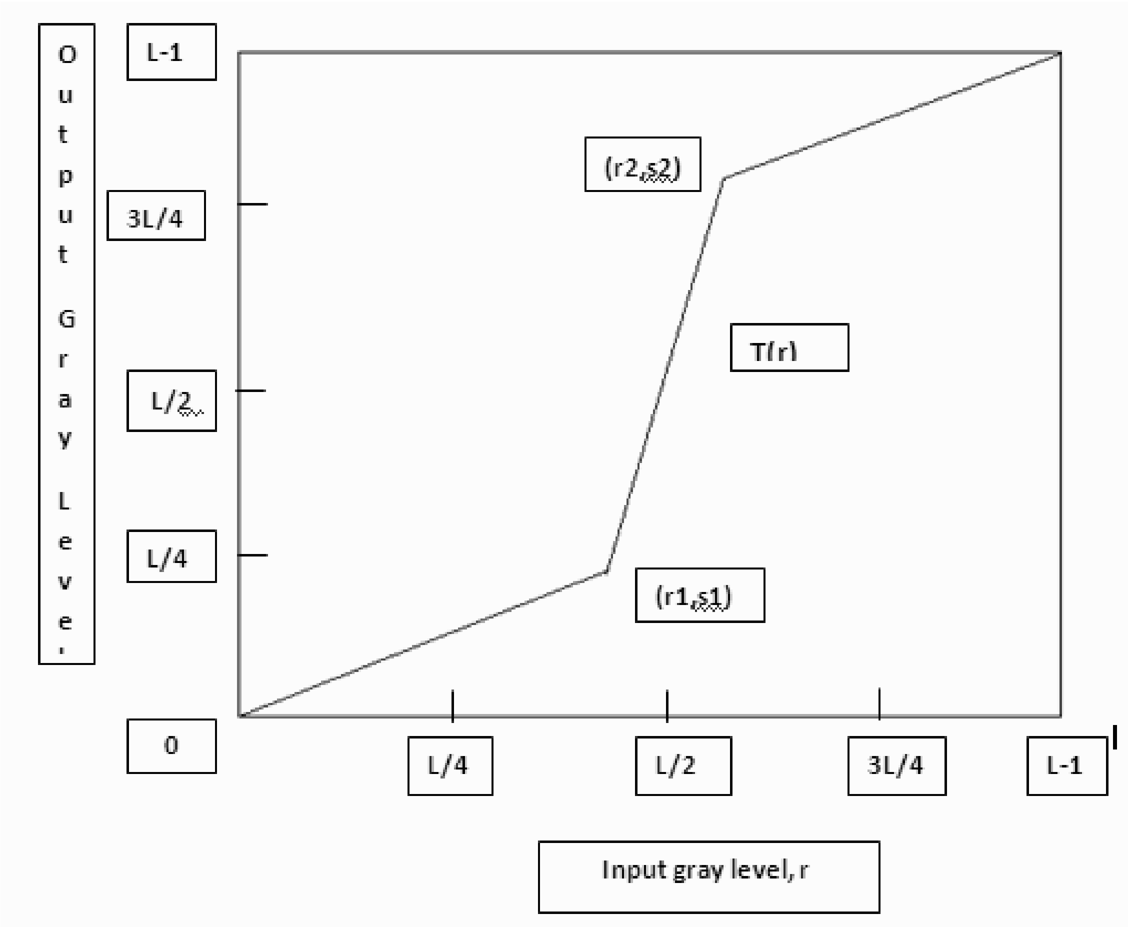

Fig 3.3: Contrast Stretching

3.4 Algorithms for Gray level slicing on Image:

3.4.1 With Background

3.4.2 Without Background

3.5 Algorithms for Bit plane slicing on Image:

3.5.1 Bit Plane Extraction

3.5.2 Effect of different bit plane concatenation:

3.6 Algorithms for Contrast stretching on Image:

3.7 Conclusion:

___

___

___

___

___

___

___

___

___

3.8 References:

i. Gonzalez R, Woods R, “Digital image processing”, Pearson Prentice Hall, 2008.

ii. Gonzalez R, Woods R, Steven E, “Digital Image Processing Using MATLAB®”, McGraw Hill Education,2010.

iii. Jayaraman S, Esakkirajan S and Veerakumar T,“Digital Image Processing” Tata McGraw Hill, 2010

iv. Joshi, Madhuri A. “Digital Image Processing: an algorithm approach”, PHI Learning Pvt. Ltd., 2006.

v. Pictures taken from: http://www.imageprocessingplace.com/root_files_V3/image_databases.html

___

(Course Teacher)