

**LAB MANUAL**

**Image Processing and Machine Vision Lab**

# ECL604

**Semester-VI**

**Third Year Engineering**

**Department of Electronics and Telecommunication.**

**SIES GST, Nerul, Navi Mumbai**

**VISION OF SIESGST**

"To be a centre of excellence in Education and Technology committed towards Socio-Economic advancement of the country"

**MISSION OF SIESGST**

1. To impart advanced knowledge in Engineering and Technology.
2. To transform young minds towards professional competence by inculcating values and developing skills.
3. To promote research and ensure continuous value addition among students and employees.
4. To strengthen association with industry, research organizations and alumni to enhance knowledge on current technologies.
5. To promote next generation technocracy and nurture entrepreneurial culture for social-economic growth.

**VISION OF EXTC Department**

" To be a Premier Department in Electronics & Telecommunications Engineering. "

**MISSION OF EXTC Department**

M1: To provide quality education satisfying the requirements of corporate world across diverse fields.

M2: To develop life-long learning skills to cater to the socio-economic needs.

M3: To strengthen Industry-Institute Interaction to bridge the gap between academic and industrial requirements.

M4: To equip students with leadership and entrepreneurial skills.

## Program Education Objectives:

## Graduates will be able to

PEO1: Identify, formulate and solve engineering problems in the Industry, complying with ethical standards and societal needs.

PEO2: Pursue higher studies and professional development courses leading to significant advancement in the field of specialization.

PEO3: Apply technical concepts to develop applications and design products.

PEO4: Exhibit leadership and entrepreneurial acumen in career.

**Program Specific Outcome**:

Students will be able to

1. Achieve eminence in domains like signal processing, VLSI, embedded IOT, RF & microwave.
2. Become technocrats capable of working in multi disciplinary fields.

**Program Outcomes**:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science,    engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**Course Outcomes:**

|  |  |
| --- | --- |
| CO1 | Implement and compare enhancement of image through various techniques. |
| CO2 | Implement segmentation techniques on images and compare the performances of techniques. |
| CO3 | Implement and compare restoration techniques for noisy grayscale images |
| CO4 | Apply the transforms for different applications in image processing |
| CO5 | Apply different morphological techniques on different types of binary images |
| CO6 | Implement advanced methods for the project in a team using their knowledge of different steps of image processing |

**Experiment List**

|  |  |  |
| --- | --- | --- |
| **Expt.No.** | **Experiment Title** | **CO** |
| 1 | To perform spatial and tonal resolution | CO1 |
| 2 | To perform and compare histogram equalization and contrast stretching on an image. | CO1 |
| 3 | To apply DCT transform to an image and study its applications | CO4 |
| 4 | To implement ideal, butterworth, Gaussian low pass filters in frequency domain and compare their performances. | CO1 |
| 5 | To implement median filter for removal of salt and pepper noise and arithmetic mean filter for removal of Gaussian noise and compare their performances for different noise intensity and with different filter sizes. | CO3 |
| 6 | To perform erosion and dilation on a black and white image and its application for boundary extraction | CO5 |
| 7 | To detect edges in the image with the help of Prewitt’s and Sobel’s edge detectors and compare their performances. | CO2 |
| 8 | To obtain co-occurrence matrix of an image | CO2 |
| 9 | Project | CO6 |

**EXPERIMENT NO. 1**

**SPATIAL AND TONAL RESOLUTION**

**EXPERIMENT NO. 1**

**AIM**: Spatial and tonal resolution

**OBJECTIVES:**

1. To understand concept of spatial resolution.
2. To understand concept of tonal resolution.

**EQUIPMENTS/SOFTWARE:** SCILAB orMATLAB 7.0

**THEORY:** **Spatial and Gray level resolution:**Sampling is the principal factor determining the spatial resolution of an image. Basically spatial resolution is the smallest discernible detail in an image.

As an example suppose we construct a chart with vertical lines of width W, and with space between the lines also having width W. A line-pair consists of one such line and its adjacent space. Thus width of line pair is http://nptel.tvm.ernet.in/courses/Webcourse-contents/IIT-KANPUR/Digi_Img_Pro/chapter_1/images/image056.gifand there are http://nptel.tvm.ernet.in/courses/Webcourse-contents/IIT-KANPUR/Digi_Img_Pro/chapter_1/images/image058.gif line-pairs per unit distance. A widely used definition of resolution is simply the smallest number of discernible line pairs per unit distance; for es 100 line pairs/mm.

Gray level resolution: This refers to the smallest discernible change in gray level. The measurement of discernible changes in gray level is a highly subjective process.

We have considerable discretion regarding the number of Samples used to generate a digital image. But this is not true for the number of gray levels. Due to hardware constraints, the number of gray levels is usually an integer power of two. The most common value is 8 bits. It can vary depending on application. When an actual measure of physical resolution relating pixels and level of detail they resolve in the original scene are not necessary, it is not uncommon to refer to an L-level digital image of size http://nptel.tvm.ernet.in/courses/Webcourse-contents/IIT-KANPUR/Digi_Img_Pro/chapter_1/images/image062.gifas having a spatial resolution of http://nptel.tvm.ernet.in/courses/Webcourse-contents/IIT-KANPUR/Digi_Img_Pro/chapter_1/images/image062.gif pixels and a gray level resolution of L levels.

**ALGORITHM SPATIAL RESOLUTION:**

1. Read the image.
2. Select alternate pixels for spatial resolution.
3. Show the output image.
4. Read the image.
5. Copy the rows and columns pixels to increase the size of the image.
6. Show the output image.

**ALGORITHM TONAL RESOLUTION:**

1. Divide the complete grey resolution in N no of tones. N=2n n-no of bits required to represent the grey color.
2. Assign one grey color for each of N bands.
3. Show the images in different tonal resolutions.

**CONCLUSION:**

**EXPERIMENT NO. 2**

**PERFORM AND COMPARE CONTRAST STRETCHING AND HISTOGRAM EQUALIZATION**

**EXPERIMENT NO. 2**

**AIM**: To perform and compare contrast stretching and histogram equalization

**OBJECTIVES:**

1. To understand concept of contrast enhancement
2. To perform enhancement of image using contrast stretching.
3. To perform enhancement of image using histogram equalization.
4. To compare performance of both the methods.

**EQUIPMENTS/SOFTWARE:** SCILAB 6.0.0

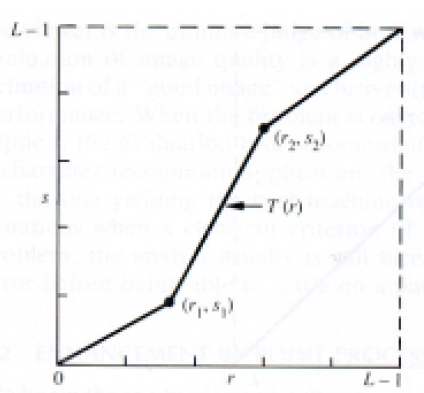
**THEORY:**

**Contrast Stretching/Compression**

Stretch gray-level ranges where we desire more information.

Low-contrast images can result from poor illumination, lack of dynamic range in the image sensor, or even wrong setting of a lens aperture during image acquisition.

The idea behind contrast stretching is to increase the dynamic range of the gray levels in the image being processed.



**Steps:**

1. Read the input image and its size.
2. Obtain values of a, b, v, w.
3. For every pixel of input image check its value and modify to new value accordingly.
4. Display input and output images with title.

**Histogram Equalisation**

Histogram of a digital image with gray levels in range [0,L-1] is a discrete function h(rk) = nk where rk -kth gray level and nk = no. of pixels of an image having gray level rk.

In histogram there are 3 possibilities as follows,

1. For a dark image the components of histogram on the low (dark) side.

2. For a bright image the component are on high ( bright ) side

3. For an image with low contrast they are in the middle of gray side.

Histogram equalization is done to spread there component uniformly over the gray scale as far as possible.

This is obtained by function

Where

Thus processed image is obtained by mapping each pixel with level rk into a corresponding pixel with level Sk in output image. This transformation is called Histogram equalization

**ALGORITHM:**

1. Read the input image and its size.
2. Obtain probability of each the gray level values of each pixel from the image.
3. Compute CDF for each gray value.
4. Compute new value for each input grey level by multiplying its CDF by 255.
5. Replace the input gray values with corresponding new values Sk.
6. Plot the equalized histogram and original histogram
7. Display the original and the equalized image.

**CONCLUSION:**

**EXPERIMENT NO. 3**

**PERFORM AND COMPARE FREQUENCY DOMAIN FILTERS**

**EXPERIMENT NO. 3**

**AIM:** To implement ideal, butterworth, Gaussian low pass filters in frequency domain and compare their performances

**OBJECTIVES:**

1. To apply and compare performance averaging filters of various size and cutoff frequency
2. To understand to convert image from spatial domain to frequency domain.
3. To see the frequency spectrum of the image.
4. To understand the concept of frequency domain filtering.

**EQUIPMENTS/SOFTWARE:** SCILAB 6.0.0

**THEORY:** Using low pass filters -

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  | | --- | --- | --- | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | |  | |  |  |  | | --- | --- | --- | | 1 | 2 | 1 | | 2 | 4 | 2 | | 1 | 2 | 1 | |
|  |  |  |  |

The second mask, shown in Figure, yields a so-called weight average, thus giving more importance (weight) to some pixels at the expense of others.

Frequency domain filtering-

The reason for doing the filtering in the frequency domain is generally because it is computationally faster to perform two 2D Fourier transforms and filter multiplication in this domain than to perform convolution in the image (spatial) domain. Also convolution becomes more complex in spatial domain as filter size increases.

The transfer function of a Butterworth LPF of order ‘n’, and with cutoff frequency at a distance Do from its origin is defined as

The transfer function of a Gaussian LPF with cutoff frequency at a distance Do from its origin is defined as

**ALGORITHM:**

**Spatial domain filtering-**

1. Read the image.
2. Define LPF masks
3. Run the mask on the image.
4. See result of the filtering

**Frequency domain filtering-**

1. Read the input image and its size.
2. Obtain the padding parameters P and Q. Typically, we select P=2M and Q=2N
3. Form a padded image, fp(x,y) of size PxQ by appending the necessary number of zeroes to f(x,y)
4. Multiply fp(x,y) by (-1)x+y to center its transform
5. Obtain the Fourier transform of the image
6. Generate a Butterworth and Gaussian filter function, H1 and H2, the same size as the image (PxQ)
7. Multiply the transformed image by the filter:  
   G1=H1.\*F; G2=H2.\*F;
8. Obtain the real part of the inverse FFT of G.

**FUNCTIONS USED (MATLAB / SCILAB):**

1. imread
2. double
3. fft2
4. ifft2
5. real
6. imshow

**CONCLUSION:**

**EXPERIMENT NO. 4**

**TO APPLY DCT TRANSFORM TO AN IMAGE AND STUDY ITS APPLICATIONS EXPERIMENT NO 4**

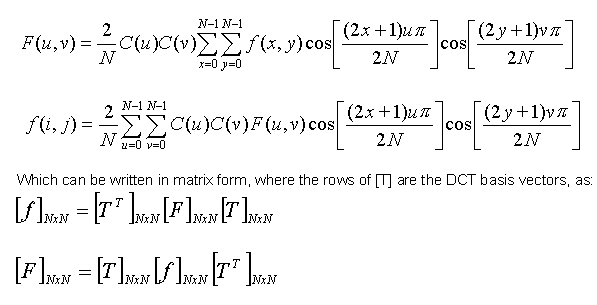
**AIM**:- To apply DCT transform to an image and study its applications

**OBJECTIVES:**

1. To obtain DCT of size equal to size of image
2. To apply the transform to the image
3. To obtain inverse DCT of the transformed image

**SOFTWARE used**:- Scilab

**Theory:**- DCT has found application in image compression. All JPEG images uses discrete cosine transform as initial stage of compression .Just as fourier transform uses sine and cosine waves to represent signal DCT uses only cosine waves. Hence DCT is purely real unlike DFT which is complex (has magnitude as well as phase). DCT equation is given by



Sinusoidal transforms, like the Discrete Cosine Transforms (DCT) and Discrete Fourier

Transforms (DFT) use image-independent transformations. It is seen that DCT’s energy compaction performance closely resembles that of KLT. Moreover, fast algorithms and architectures are available for DCT and DFT. As compared to DFT, application of DCT results in less blocking artifacts due to the even symmetric extension properties of DCT. Also, DCT uses real computations, unlike the complex computations used in DFT. This makes DCT hardware simpler, as compared to that of DFT. These advantages have made DCT-based image compression a standard in still-image and multimedia coding standards.

**Limitations of DCT:-**

Despite excellent energy compaction capabilities, mean-square reconstruction error performance closely matching that of KLT and availability of fast computational approaches, DCT offers a few limitations which restrict its use in very low bit rate applications. The limitations are listed below:

(i) Truncation of higher spectral coefficients results in blurring of the images, especially wherever the details are high

(ii) Coarse quantization of some of the low spectral coefficients introduces graininess in the smooth portions of the images.

(iii) Serious blocking artifacts are introduced at the block boundaries, since each block is independently encoded, often with a different encoding strategy and the extent of quantization.

Of all the listed problems, as above, blocking artifact is the most serious and objectionable one at low bit rates. Blocking artifacts may be reduced by applying an overlapped transform, like the Lapped Orthogonal Transform (LOT) or by applying post-processing. At lower bit rates, Discrete Wavelet Transforms (DWT) avoid the blocking artifacts of DCT and present better coding performance

**Conclusion:-**

Students are expected to write conclusion by their own.

**EXPERIMENT NO. 5**

**TO PERFORM EROSION AND DILATION ON A BLACK AND WHITE IMAGE AND ITS APPLICATION FOR BOUNDARY EXTRACTION**

**EXPERIMENT NO. 5: Morphological Processing**

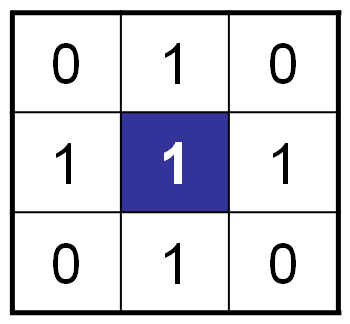
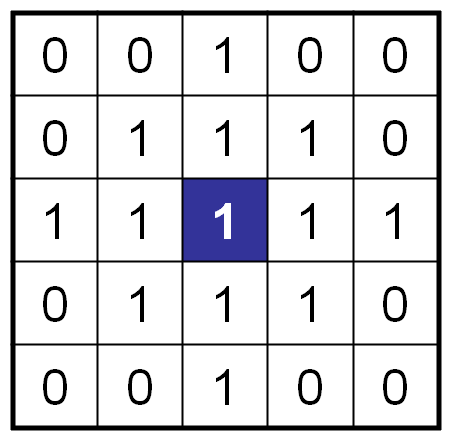
**AIM: -** To perform erosion and dilation on a black and white image and its application for boundary extraction

**THEORY: -**

Once segmentation is complete, morphological operations can be used to remove imperfections in the segmented image and provide information on the form and structure of the image. Morphological image processing (or *morphology*) describes a range of image processing techniques that deal with the shape (or morphology) of features in an image

Morphological operations are typically applied to remove imperfections introduced during segmentation, and so typically operate on bi-level images.

Structuring elements can be any size and make any shape. However, for simplicity we will use rectangular structuring elements with their origin at the middle pixel.



**Morphological Operations**

Fundamentally morphological image processing is very like spatial filtering

The structuring element is moved across every pixel in the original image to give a pixel in a new processed image

The value of this new pixel depends on the operation performed

There are two basic morphological operations: erosion and dilation

**Dilation** of image *f*  by structuring element *s* is given by *f*   *s*

The structuring element s is positioned with its origin at *(x, y)* and the new pixel value is determined using the rule:



**Original Image Processed Image with Dilated Pixels Structuring Element**

****

**Original image Dilation by 3\*3 square Dilation by 5\*5 square**

**structuring element structuring element**

**Erosion** Erosion of image *f* by structuring element *s* is given by *f* ⊖ *s*

The structuring element s is positioned with its origin at *(x, y)* and the new pixel value is determined using the rule:

**Compound Operations**

More interesting morphological operations can be performed by performing combinations of erosions and dilations

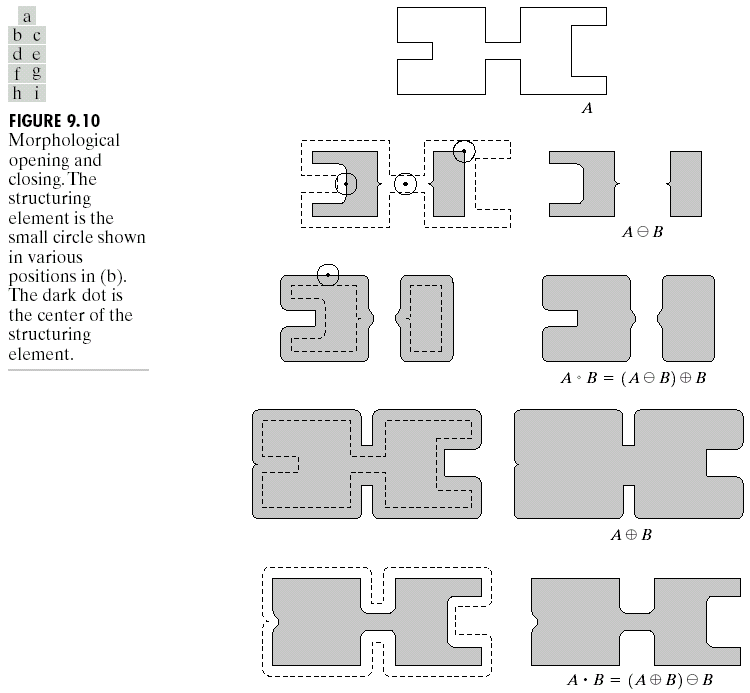
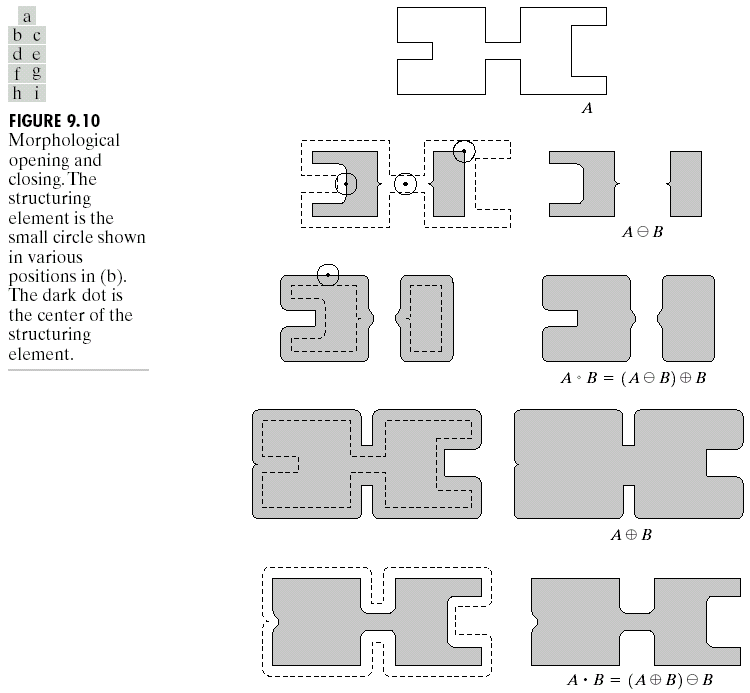
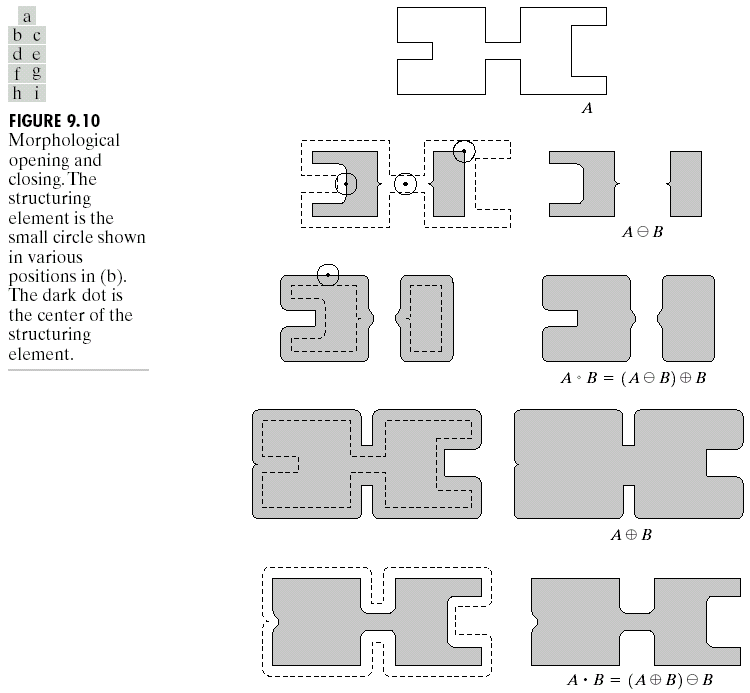
The most widely used of these *compound operations* are:

* + Opening
  + Closing

**Opening**

The opening of image *f* by structuring element *s,* denoted *f* ○ *s* is simply an erosion followed by a dilation

***f* ○ *s = (f* ⊖*s)***  ***s***

****

**Original Image Processed Image**

**Closing**

The closing of image f by structuring element s, denoted f • s is simply a dilation followed by an erosion

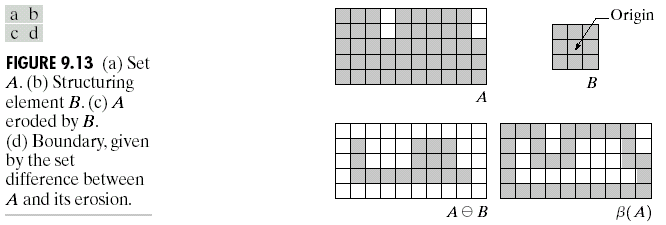
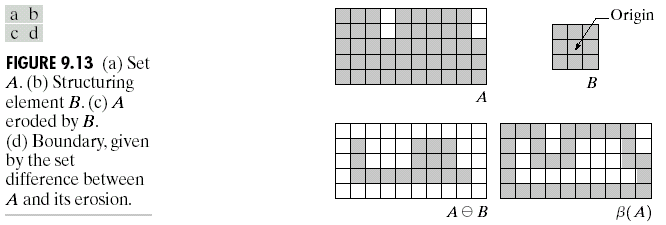
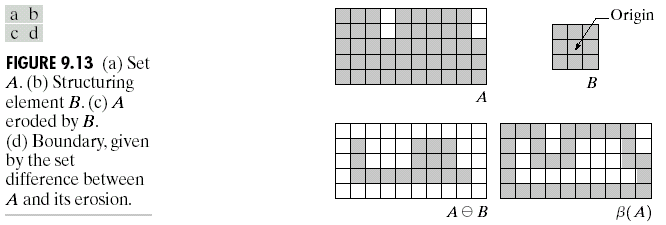
f • s = (f s)⊖s



**Boundary Extraction**

Extracting the boundary (or outline) of an object is often extremely useful

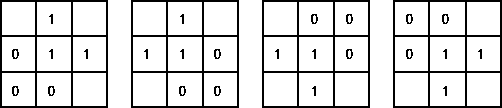
The boundary can be given simply as  *β(A) = A – (A⊖B)*

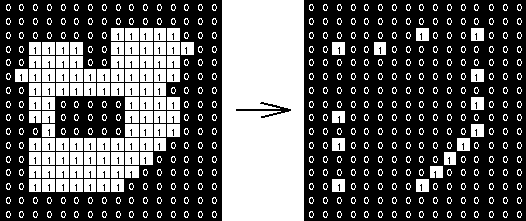
****

**Hit or Miss Transform**

The hit-and-miss transform is a general binary morphological operation that can be used to look for particular patterns of foreground and background pixels in an image. It is actually the basic operation of binary morphology since almost all the other binary morphological operators can be derived from it. As with other binary morphological operators it takes as input a [binary image](http://homepages.inf.ed.ac.uk/rbf/HIPR2/binimage.htm) and a [structuring element](http://homepages.inf.ed.ac.uk/rbf/HIPR2/strctel.htm), and produces another binary image as output.

[structuring element](http://homepages.inf.ed.ac.uk/rbf/HIPR2/strctel.htm)s





**CONCLUSION** :-

**EXPERIMENT NO. 6**

**TO IMPLEMENT AND COMPARE PERFORMANCE OF FILTERS FOR REMOVING SALT AND PEPPER NOISE AND GAUSSIAN NOISE**

**EXPERIMENT NO. 6**

**AIM:** To implement median filter for removal of salt and pepper noise and arithmetic mean filter for removal of Gaussian noise and compare their performances for different noise intensity and with different filter sizes.

**OBJECTIVES:**

1. To understand concept of impulse noise, gaussian noise
2. To understand process of restoration.
3. To compare the results of restoration by geometric mean and median filter for Gaussian and salt and pepper noise

**EQUIPMENTS/SOFTWARE:** SCILAB or Matlab 7.0

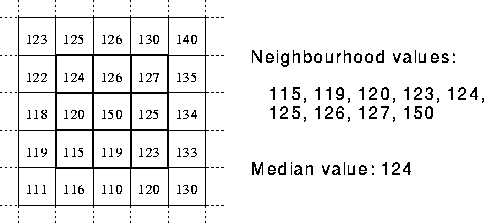
**THEORY:** Digital images are frequently affected by impulse noise during their acquisition or transmission in a noisy environment. Therefore, to efficiently remove noise from an image while preserving its features is a fundamental problem of image processing. The impulse noise can be classified either as salt-and-pepper with noisy pixels taking either maximum or minimum value, or as random valued impulse noise. The removal of fixed-valued impulse noise has been widely studied and a large number of algorithms have been proposed. The median filter is the most popular choice for removing the impulse noise from images because of its effectiveness and high computational efficiency.

The median filter is normally used to http://homepages.inf.ed.ac.uk/rbf/HIPR2/mote.gifreduce noise in an image, somewhat like the [mean filter](http://homepages.inf.ed.ac.uk/rbf/HIPR2/mean.htm). However, it often does a better job than the mean filter of preserving useful detail in the image.

Like the [mean filter](http://homepages.inf.ed.ac.uk/rbf/HIPR2/mean.htm), the median filter considers each pixel in the image in turn and looks at its nearby neighbors to decide whether or not it is representative of its surroundings. Instead of simply replacing the pixel value with the mean of neighboring pixel values, it replaces it with the median of those values. The median is calculated by first sorting all the pixel values from the surrounding neighborhood into numerical order and then replacing the pixel being considered with the middle pixel value. (If the neighborhood under consideration contains an even number of pixels, the average of the two middle pixel values is used.)

Figure 1 illustrates an example calculation.

As can be seen, the central pixel value of 150 is rather unrepresentative of the surrounding pixels and is replaced with the median value: 124. A 3×3 square neighborhood is used here --- larger neighborhoods will produce more severe smoothing.



**Gaussian noise** represents [statistical noise](http://en.wikipedia.org/wiki/Statistical_noise) having [probability density function](http://en.wikipedia.org/wiki/Probability_density_function) (PDF) equal to that of the [normal distribution](http://en.wikipedia.org/wiki/Normal_distribution), which is also known as the [Gaussian distribution](http://en.wikipedia.org/wiki/Gaussian_distribution). In other words, the values that the noise can take on are Gaussian-distributed.

The probability density function pof a Gaussian random variable zis given by:

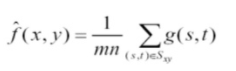
p_G(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{ -\frac{(z-\mu)^2}{2\sigma^2} }

where z represents the grey level, \muthe [mean](http://en.wikipedia.org/wiki/Mean) value and \sigmathe [standard deviation](http://en.wikipedia.org/wiki/Standard_deviation).

Principal sources of [Gaussian noise](http://en.wikipedia.org/wiki/Gaussian_noise) in [digital images](http://en.wikipedia.org/wiki/Digital_image) arise during acquisition eg. [sensor noise](http://en.wikipedia.org/wiki/Sensor_noise) caused by poor illumination and/or high temperature, and/or transmission eg. [electronic circuit noise](http://en.wikipedia.org/wiki/Circuit_noise_level).

**Arithmetic mean filter:-**

In this filter centre pixel of g(x,y) is replaced



x,y belong to subimage.

**ALGORITHM:**

1. Read the image.
2. Add impulse noise to the image
3. Add Gaussian noise to the image.
4. Perform arithmetic mean and median filter on both the noisy images.
5. Compare the results of both the filtering

**FUNCTIONS USED (MATLAB / SCILAB):**

1. imread
2. imnoise
3. sort(MATLAB)/gsort(SCILAB)
4. imshow

**CONCLUSION:**

**EXPERIMENT NO. 7**

**TO DETECT EDGES IN THE IMAGE WITH THE HELP OF PREWITT’S AND SOBEL’S EDGE DETECTORS AND COMPARE THEIR PERFORMANCES**

**EXPERIMENT NO.:7**

**AIM: -** To detect edges in the image with the help of Prewitt’s and Sobel’s edge detectors and compare their performances.

**OBJECTIVE:**

1. Tounderstand concept of edge detection operators.

**EQUIPMENTS/SOFTWARE: Scilab6.0.0**

**THEORY: -**

Technically speaking, image segmentation refers to the decomposition of a scene into different components (thus to facilitate the task at higher levels such as object detection and recognition)

Segmentation subdivides an image to regions or objects

Two basic properties of intensity values

* Discontinuity

– Edge detection

* Similarity

– Thresholding

– Region growing/splitting/merging

**Detection of Discontinuities: Point Detection**

**Mask operation**

• Point detection  

Isolated point  whose gray value is significantly different from its background



**Edge Detection: Gradient Operators**

Gradient

– Vector pointing to the direction of maximum rate of change of *f* at coordinates (*x*,*y*)

– Magnitude: gives the quantity of the increase (sometimes referred to as *gradient* too)

– Direction: perpendicular to the direction of the edge at (*x*,*y*)

– Partial derivatives computed through 2x2 or 3x3 masks

•Sobel operators introduce some smoothing and give more importance to the center point



 (Magnitude)  (Direction)  

**Detecting diagonal edges**

****

**ALGORITHM**

1. Read input image
2. Define mask for particular operator
3. Move mask over image and calculate horizontal and vertical edges from the image
4. Display all detected edges.

**FUCTIONS**

1. **imread()**
2. **imshow()**
3. **double()**

**CONCLUSION**:-

**EXPERIMENT NO. 8**

**REGION REPRESENTATION USING CO-OCCURRENCE MATRICES**

**EXPERIMENT NO. 8**

**AIM:** To represent the image regions using co-occurrence matrix

**OBJECTIVES:**

1. To implement method of co occurrence matrix for representation of region.

**THEORY:** A co-occurrence matrix is a 2-D array C in which both the rows and the columns represent a set of possible image values V. The value of C(i, j) denotes how many times value I co-occurs with value j in some designated spatial relationships R. V can be the set of possible gray tones for gray-level images or the set of possible colours for colour images.

Let d = (dr, dc) be a displacement vector (or position vector) where dr is a displacement in rows and dc is a displacement in columns. Let V be a set of gray tones. The gray-level co-occurrence matrix Cd for image I is defined by



Any matrix or pair of matrices can be used to generate a co-occurrence matrix, though their most common application has been in measuring [texture](https://en.wikipedia.org/wiki/Texture_(computer_graphics)) in images, so the typical definition, as above, assumes that the matrix is an image.

**ALGORITHM**

1. Read input image
2. Find the co occurrence matrix of the image.

**CONCLUSION**: