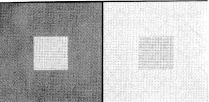
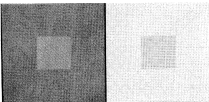
**What is simultaneous contrast? How Mach band effect appears?**

The luminance or intensity of an object is independent of the luminances of the surrounding objects. The brightness (also called apparent brightness) of an object is the perceived luminance and depends on the luminance of the surround. Visual phenomena: *Two objects with different surroundings could have identical luminance but different brightness.* Our perception is sensitive to luminance contrast rather than the absolute luminance values.



Simultaneous contrast: small squares in the middle have equal luminance

but do not appear equally bright.



Simultaneous contrast: small squares in the middle appear almost equally bright,

but their luminances are different.

According to Weber's law : if the luminance *f0* of an object is just noticeably different from the luminance *fs* of its surround, then their ratio is

C:\Documents and Settings\khec\Desktop\chap3\1.4_files\Image14_13.gif

Writing *f0 = f , fs + f* where * f* is small for just noticeably different luminances, previous equation can be shown as

C:\Documents and Settings\khec\Desktop\chap3\1.4_files\Image14_14.gif

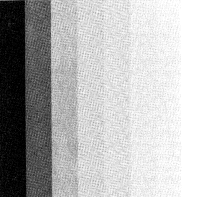
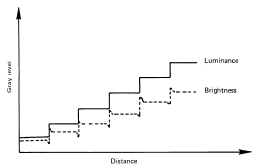
The value of the constant has been found experimentally to be *0.02*, which means that at least *50* levels are needed for the contrast on the scale of *0* to *1*. Previous equation says equal increment in the log of the luminance should be perceived to be equally different, i.e. * (log f) is proportional to  c,* the changes in contrast. Accordingly, the quantity : *c=a1 + a2 log10 f* where *a1 , a2* are constants, is called the contrast.

There are other models of contrast **" Luminance to Contrast Models"**

|  |  |  |
| --- | --- | --- |
| 1 | Logarithmic law | c=50log10f, 1  f  100 |
| 2 | Power law | *[C:\Documents and Settings\khec\Desktop\chap3\1.4_files\eq2.jpg](http://ict.udlap.mx/people/oleg/docencia/IMAGENES/eq)*, n=2,3,..., 2 = 10,  3 = 21.9 |
| 3 | Background law | C:\Documents and Settings\khec\Desktop\chap3\1.4_files\Image14_15.giffB background luminance |

The luminance *f* usually lies in the interval *[0,100]* except in the logarithmic law. Contrast scale is over *[0,100]*.

The spatial interaction of luminances from an object and its surround creates a phenomenon called the *Much band effect*. This effect shows that brightness is not a monotonic function of luminance. Consider the gray level bar chart with constant luminance within each bar. Transition an each bar appear BRIGHTER on the right side and DARKER on the left side (it is shown by dashed line with undershoots and overshoots which illustrate the Much band effect)

Gray-level bar chart and Much band effect.

**What do you mean by image fidelity?**

Image fidelity are the criteria useful for measuring image quality and for rating the performance of a processing technique or a vision system. There are two types of criteria that are used for evaluation of image quality, subjective and quantitative.

**a)** **Subjective criteria** uses the rating scales such as goodness scales and impairment scale. The training set of images is used to calibrate such scale. The group goodness scale is based on comparison within a set of images.

**Image goodness scales**

|  |  |
| --- | --- |
| **Overall goodness scale** | **Group goodness scale** |
| Excellent (5) | Best (7) |
| Good (4) | Well above average (6) |
| Fair (3) | Slightly above average (5) |
| Poor (2) | Average (4) |
| Unsatisfactory (1) | Slightly below average (3) |
|  | Well below average (2) |
|  | Worst (1) |

The numbers in parenthesis indicate a numerical weight attached to the rating.

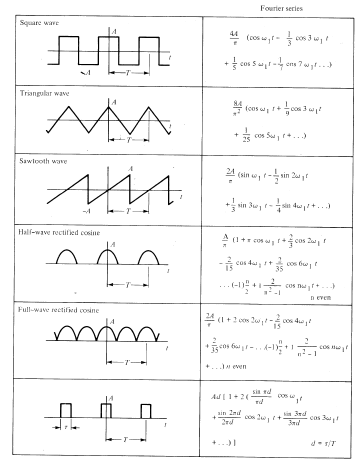
**b) There is an *impairment scale*** which rates an image on the basis of the level of DEGRADATION present in an image when compared with an ideal image.(useful in applications such as image coding, restoration, projection...)

**Impairment scale**

|  |  |
| --- | --- |
| Not noticeable | 1 |
| Just noticeable | 2 |
| Definitely noticeable but only slight impairment | 3 |
| Impairment not objectionable | 4 |
| Somewhat objectionable | 5 |
| Definitely objectionable | 6 |
| Extremely objectionable | 7 |

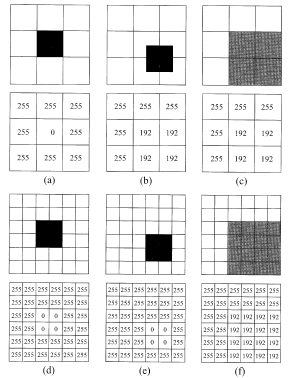
**c) Among the quantitative measures,** a class of criteria used often is calls the *mean square criterion*. It refers to some sort of average or sum (or integral) of squares of the error between two images. For *MxN* images *u(m,n)* and *u'(m,n)*, (or *v(x,y)* and *v'(x,y)* in the continuous case ), the quantity:

C:\Documents and Settings\khec\Desktop\chap3\1.4_files\Image14_18.gif

where *R* is the region over which the image is given, is called the *average least squares* (or i*ntegral square*) *error*

**Why FT is used in image processing?**

Fourier transform is used in image processing because of the simplification of the operation, manipulation with the principal characteristics of the signals in frequency domain, and for solution of the specific problems, which are difficult in time domain representation. One example is following: the digital camera captures the object, which occupies the area of exactly one pixel (or the set of the pixels). In this case representation of this object is simple and accurate. But usually the object takes only one part of the area defined by pixel. That is why the representation will be not correct (extension of the object size with the lost of contrast and generation of the weak boundaries)



Digitizing the objects with respect of the their position

b), c) the object covers 1/3 part of the pixel's area and is represented

as 192 (3/4 of 255 of white)

**What is histogram modelling? Illustrate histogram equalization to enhance low contrast image.**

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

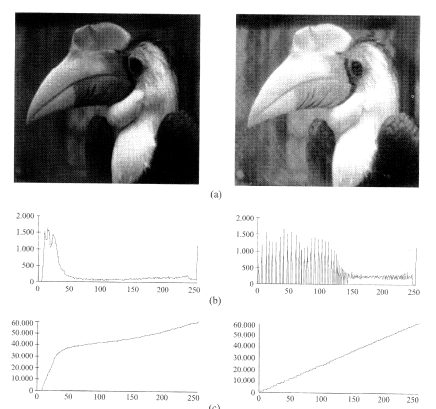
where N and M are the dimensions of an image and 256 is a number of the gray scales. In ideal case it is desired that *G(i')=H(i)*. Therefore

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

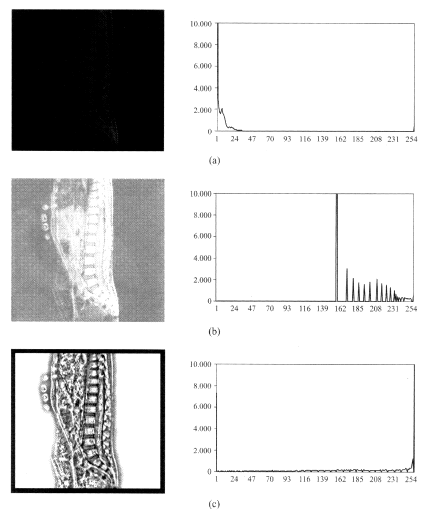
The application of the histogram equalization is shown in the following figure



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

**What is image magnification? Explain two different techniques.**

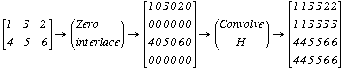
Often the zooming of a given region is necessary to obtain. This requires taking an image and displaying it as a larger image.

*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

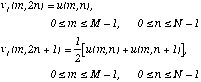
This gives http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image321_6.gif

The following figure shows the interpolation process and examples.

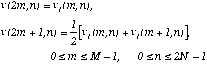


*Linear interpolation*

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

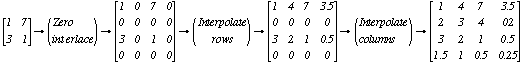


Here it is assumed that the input image is zero outside *[0,M-1]x[0,N-1]*. The above result can also be obtained by convoluting the *2Mx2N* zero interlaced image with the array

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.

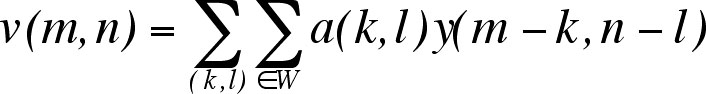


**Discuss image sharpening & smoothing filters.**

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

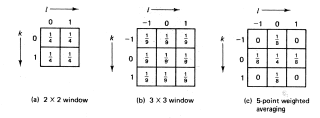
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_1.gif

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

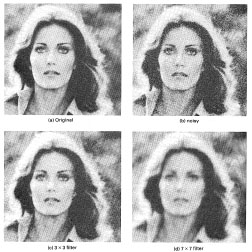
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_2.gif

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.



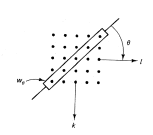
Spatial averaging filters for smoothing images containing the Gaussian noise.

***(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

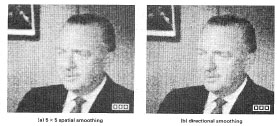
http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter3/Image32_4.gif

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

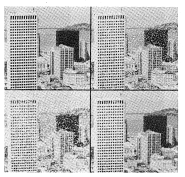
***(iii) Median Filtering***

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.

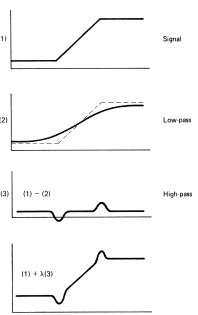


Spatial averaging versus median filtering: (a) original, b) with binary noise,

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

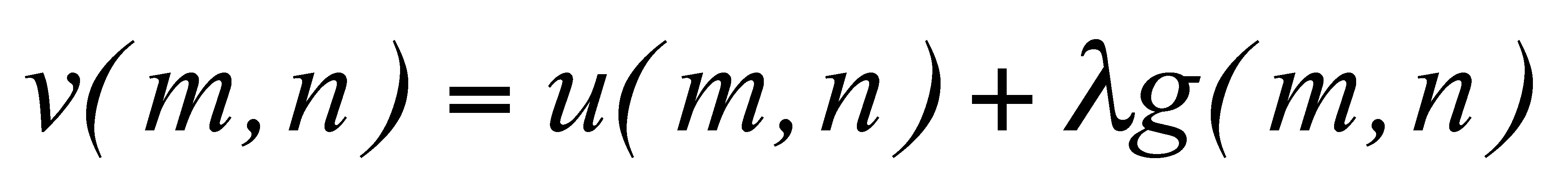
***(iv) Unsharp Masking and Crispening***

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.



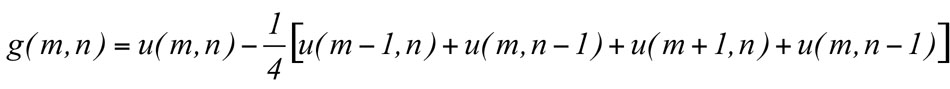
Unsharp masking operation

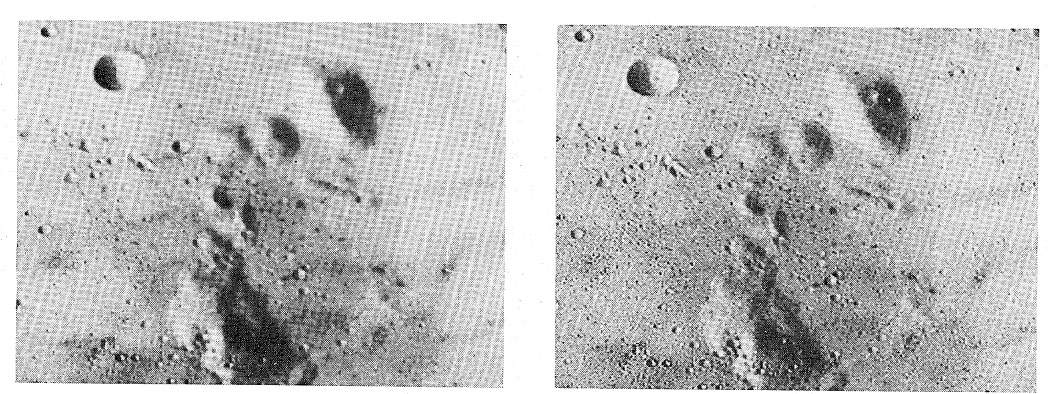
In general, the unsharp masking operation can be represented by



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

**Define data redundancy, code redundancy & image entropy.**

**Data Redundancy**

* Control issues in digital image compression
* If n1 and n2 denote the no of information carrying units in two data sets that represent same information
* Then Relative Data Redundancy (RD) of first data set with respect to second one is given by
* Rd=1-1/CR (CR is compression ratio)

CR= n1/n2

* if n1=n2 , CR =1, and RD =0 the first representation of information contains no redundant data (Relative to second data)
* when n2<<n1 , CR→∞ and RD→1 ( significant compression and highly redundant for n1)
* A practical compression ration such as 10(or 10:1) => First data set hs 10 information carrying units (bits) for every 1 unit in second or compressed data set .
* Compression ratio =0.9=90% of first data is redundant.

**Coding Redundancy**

* Let us assume a discrete variable rk represents the gray levels of an image
* Each rk occurs with a probability Pr(rk ) is given by Pr(rk )=nk/n ( histogram normalization)
* L is no of gray levels
* nk => no of times that kth grey level appears in the image
* n=> Total no of pixels in the image
* if the no of bits used to represent each value of rk  is l(rk ) => average no of bits required to represent each pixels is given by

(Where Lavg is the average length of code assigned for variable grey level values)

e.g. consider 8 level image has grey level distribution as shown

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **rk** | **Pr(rk )** | **Code1** | **L1(rk )** | **Coder 2** | **L2(rk )** |
| r0 | 0.19 | 000 | 3 | 11 | 2 |
| r1 | 0.25 | 001 | 3 | 01 | 2 |
| r2 | 0.21 | 010 | 3 | 10 | 2 |
| r3 | 0.16 | 011 | 3 | 001 | 3 |
| r4 | 0.08 | 100 | 3 | 0001 | 4 |
| r5 | 0.06 | 101 | 3 | 00001 | 5 |
| r6 | 0.03 | 110 | 3 | 000001 | 6 |
| r7 | 0.02 | 111 | 3 | 000000 | 6 |

* if a natural 3bit binary code is used to represent 8 possible grey level then Lavg=3
* if code 2 is used then
* Lavg= 2\*0.19+2\*0.25+2\*0.21+3\*0.16+4\*0.08+5\*0.06+6\*0.03+6\*0.02

=2.7 bits

* compression ratio = CR=n1/n2= 3/2.7= 1.11
* Approximate 10% of data resulting from code1 is redundant
* Exact level of redundancy RD=1-1/1.11= 0.099
* Assigning code of fewer bits to the more probable grey levels than to the less probable ones => archives data compression= > variable length coding

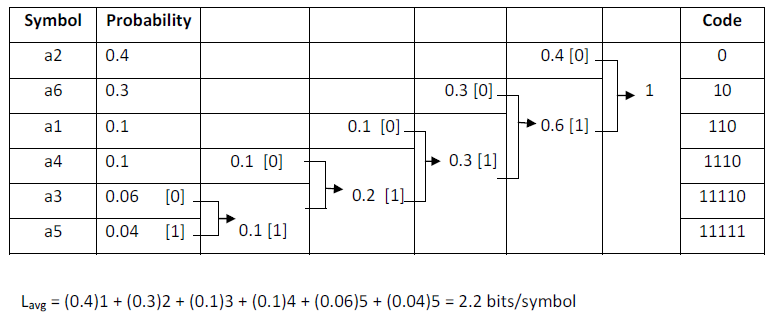
**Illustrate Huffman Coding.**

**Huffman Coding**

* Most popular technique for removing coding redundancy is due to Huffman coding.
* Consider the symbols with different probabilities.

|  |  |
| --- | --- |
| Symbol | Probability |
| a1 | 0.1 |
| a2 | 0.4 |
| a3 | 0.06 |
| a4 | 0.1 |
| a5 | 0.04 |
| a6 | 0.3 |

* Symbol should be sorted in descending order according to its Probability

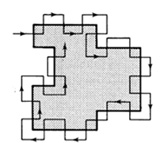


* The First step in Huffman’s approach => Create a series of source reductions by ordering the probabilities. [ Descending order ]
* The Second step => Code each reduced source starting with the smallest source and working back to the original source.

**Differentiate inter-frame & intra-frame coding.  
What do you mean by morphological processing? Define dilation & erosion.  
Write boundary extraction algorithm with example.**

**Contour following**

The contour-following algorithms trace boundaries by ordering successive edge points. A simple algorithm for tracing closed boundaries in binary images is shown in figure.



Finding the boundary in a binary image.

This can be done simply by a procedure that functions like Papert's turtle:

* Scan the image until a region pixel is encountered
* If it is a region pixel, turn left and step, else turn right and step
* Terminate upon return to the starting pixel.

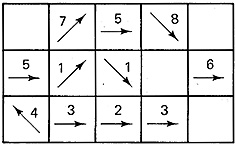
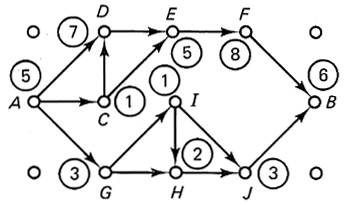
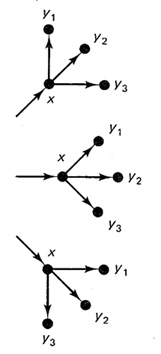
This procedure requires the region to be four-connected for a consistent boundary. Parts of the eight-connected regions can be missed. Also, some bookkeeping is necessary to generate an exact sequence of boundary pixels without duplication.

This method may be extended for gray scale images by searching for edges in the 45° to 135° direction from the direction of the gradient to move from the inside to the outside of the boundary and vise-versa.

**Edge linking as heuristic graph searching**

A boundary can also be viewed as a pass through a graph formed by linking the edge segments together. Linkage rules give the procedure for connecting the edge elements. If each pass has its weight or cost, the lowest-cost path between two nodes of weighted graph is found.

Assume *3x5* array of edges in image where a gradient operator is applied to this gray level image, creating the magnitude image *g(x)* and direction image (tangential contour direction) *(x)* (! the contour direction is *90°* to the gradient direction) as nodes in a graph, each with the weighting factor *g(x)* as it shown in following figure.

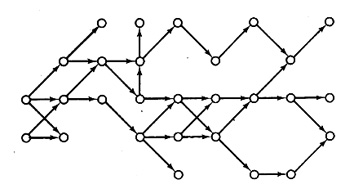
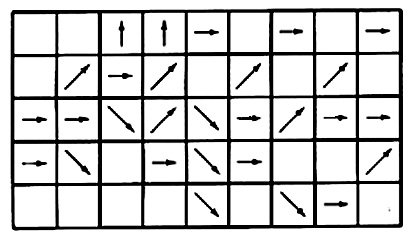
  

gradient magnitudes *g(x)* --------linkage rules ----------------graph interpretation

with contour directions *(x)*

Heuristic graph search method for boundary extraction

Another example of the linkage rules application for gradient of image is shown as:



Interpreting a gradient image as a graph.

The pixel *x* is considered to be linked to *y* if the latter is one of the three eight-connected neighbors (*y1, y2, y3* in figure) in front of the contour direction and if |*(x)-(y)|<90°* . According the these linkage rules the graph is obtained [heuristic search by Nilsson, Martelli].

As an example, *S(xk)* is the sum of edge gradient magnitudes along the path from A to *xk*. At A, the successor nodes a D,C, and G with *S(D)=12, S(C)=6, and S(G)=8*. Therefore the node D is selected , and C and G are discarded. From here on nodes E,F, and B provide the remaining path (boundary path is ADEFB).

**Define pruning operation. How skeleton of an object is obtained?**

Skeleton of an object is obtained from thining algorithm.

Concept: 1. Do not remove end points

2. Do not break connectivity

3. Do not cause excessive erosion

Apply only to contour pixels: pixels “1” having at least one of its 8 neighbor pixels valued “0”

**Notation:** Let is equal to Neighborhood arrangement for the thinning algorithm

Let 

T(*p*1) = the number of transition 0-1 in the ordered sequence *p*2, *p*3, … , *p*8, *p*9, *p*2.

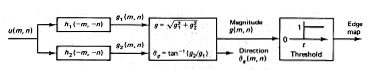
**What is image segmentation? Discuss discontinuity base image segmentation.**

Image segmentation is used to separate an image into constituent parts based on some image attributes. Image segmentation is an important step in image analysis.

Discontinuity of pixel properties at the boundary between object and background is used to distinguish between pixels belonging to the object and those of background.

**What are different approaches of segmentation based on region grouping?  
What is gradient operator? Discuss various gradient operator to detect edges.**

Gradient operators are represented by a pair of masks *H1, H2*, which measure the gradient of the image *u(m,n)* in two orthogonal directions as it shown in figure



Edge detection via gradient operators

Defining the bi-directional gradients

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_14.gif

the gradient vector magnitude and direction are given by

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_15.gif

Often for computation cost reduction, the magnitude gradient is calculated as

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_16.gif

rather than as in equation with square root. This calculation is easier to perform and is preferred especially when implemented in digital hardware.

As the images are not continuous signals the gradient is done by

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_17.gif

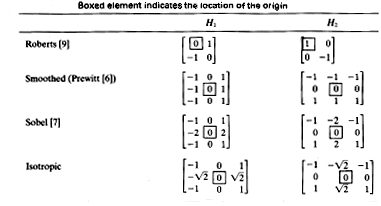
In the digital form it may be represented by masks:

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_18.gif

But in practice, these masks are not used because they are too sensible to noise due to operations on level of the one pixel. There are some developed well-known operators which are not so sensitive to noise.

Some common gradient operators.

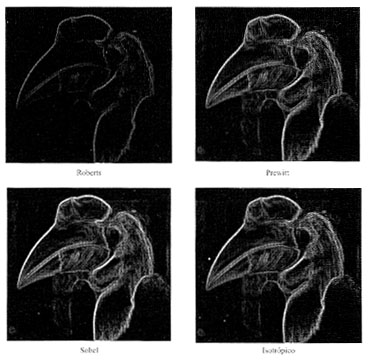
Boxed element indicates the location of the origin



The Prewitt (Smoothed), Sobel, and Isotropic (Frei-Chen) operators compute horizontal and vertical differences of local sums. This reduces the effect of noise in the data. The pixel location *(m,n)*is declared an edge location if *g(m,n)* exceeds some threshold *t*. The locations of edge points constitute an edge map  *(m,n)* which is defined as

http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image41_19.gif

The edge map gives the necessary data for tracing the object boundaries in an image. Typically, *t*may be selected using the accumulative histogram of *g(m,n)* so that 5 to 10 % of pixels with largest gradients are declared as edges.

The Prewitt operator detects better the vertical borders, Sobel does it good in diagonals, Isotropic operator is the is an equilibration between two previous ones.

Comparison between distinct border detection operators

**What is feature thresholding?**

It is the simplest method of image segmentation. Consider an image that contains two types of regions and the distinctness of the regions is reflected by the feature value at the pixel belonging to them. Suppose there exists a threshold (t) such that feature values of all pixels that actually belongs to region of first type are less than or equal to (t) . The gray level values of all pixels that actually belong to regions of the second type are greater than (t) . Hence, the segmented image is represented as

Where p(r,c) is the feature value at pixel (r,c ). If gray level is the feature then p(r,c)= g(r,c) for all (r,c). Note that values 1 and 0 represent labels and not the values => They can be replaced by other symbols. The threshold can be treated as a class boundary. Use of thresholding techniques for image segmentation needs to solve the following two problems.

o Choice of features/ properties to achieve desired segmentation.

o Selection of optimum threshold that would incur the least classification error. The simplest feature may be gray level. Other features => Average gray level over a neighborhood, Average gradient over a neighborhood. Example : Consider three images with disc at the center. In first figure => All pixels of central disk have gray level 255 ( Easy ) In second figure => Central disk contains 25% pixels with gray level 127, 50% pixels with gray level 191 and 25% pixels with gray level 255. Applying mean filter to the image and obtain the threshold. In third case => Central disk contains 50% pixels with gray level 127, 25% pixels with gray level 254 and 25% pixels with gray level 0. Apply gradient operator followed by mean filter => Obtain threshold.

**What are principle of Neuro-computing? Discuss single perceptron network as a classifier.**

The principles of neuro computing are given below

**Connectionizm:**NN is a highly interconnected structure in such a way that the state of one neuron affects the potential of the large number of another neurons to which it is connected according to weights of connections

**Not Programming but Training**:NN is trained rather than programmed to perform the given task since it is difficult to separate the hardware and software in the structure. We program not solution of tasks but ability of learning to solve the tasks

**Distributed Memory** :NN presents an distributed memory so that changing-adaptation of synapse can take place everywhere in the structure of the network.

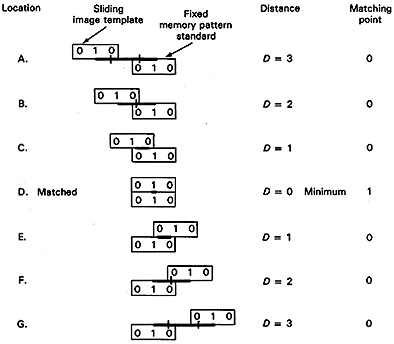
**Learning and Adaptation** :NN are capable to adapt themselves (the synapses connections between units) to special environmental conditions by changing their structure or strengths connections.

**Non-Linear Functionality** :Every new states of a neuron is a nonlinear function of the input pattern created by the firing nonlinear activity of the other neurons.

**Robustness of Assosiativity** :NN states are characterized by high robustness or insensitivity to noisy and fuzzy of input data owing to use of a highly redundance distributed structure

**Short notes**  
**Edge detection using Template**

Cross correlation is determined by calculating the sum of the distance between corresponding pattern points in the two images (one is the original image and another is predefined template. The following figure shows the pattern matching for *3x1*image: the distance D and the number of matching corresponding reference elements is determined for all locations. The matching of *3x1*pattern in the *x* direction can be achieved by shifting the template from right to left one pixel at a time and calculating the sum of the distances between key registration points. In the following example the pixel containing *1* is selected as the registration point.



Summary of results of the template shifting routine for the seven locations:

Matching point\_\_\_\_\_\_ 0 0 0 1 0 0 0

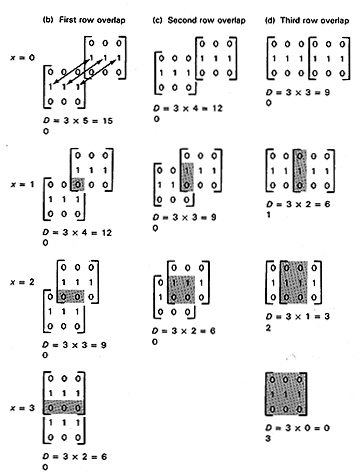
Location\_\_\_\_\_\_\_\_\_\_ A B C D E F G

The best fit is indicated to be location D where there is the only matching point condition of all seven locations.

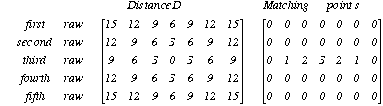
The presence of the noise or distortion will result in the cross-correlation factor not going to zero, but the minimum value is the best estimate of optimum location of a match.

The method may be extended to a *3x3* image pattern as it shown in the following figure. The template is first moved in the *x* direction and then the process is repeated after stepping one pixel in the*y* direction. The output is the expanded matrix containing the number of matching points.

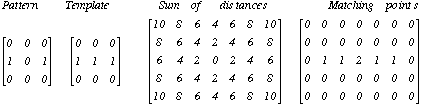
pattern:http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image412_42.gif template: http://ict.udlap.mx/people/oleg/docencia/IMAGENES/chapter4/Image412_42.gifCriteria: distance=*D* units, matching elements



Resulting pattern for distance between the three reference points as *x=0* to *6*



Thus, the pattern location is fixed and template is moved horizontally one unit at a time, and the process repeated for each of the possible rows analyzing the best matching position indicated by the lowest *D* and highest mating point location is the third row, forth position.The presence of the noise (element in pixel (2,2)) in the next example will not affect significantly the edge detection, only the number of the matching points is reduced



**Hopfield network**

* A paper by John Hopfield in 1982 was the catalyst   
  in attracting the attention of many physicists to   
  "Neural Networks".
* In a network of McCulloch-Pitts neurons

whose output is 1 iff Σwij sj ≥ qi and is otherwise 0,

neurons are updated synchronously: every neuron processes its inputs at each time step to determine a new output.

* AHopfield net(Hopfield 1982) is a net of such units subject to the asynchronous rule for updating one neuron at a time:

"Pick a unit i at random.

If Σwij sj ≥ qi, turn it on.

Otherwise turn it off."

* Moreover, Hopfield assumes symmetric weights:

wij= wji

**FFT**

The principal objective of the FFT is the reduction of number of operations necessary for transform from time to frequency domain. The restriction for FFT is that the dimension of the transformed discrete set must be power of 2. That is why

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_24.gifwhere C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_25.gif

Now it is possible to rewrite

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_26.gif

and according the definition

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_27.gif

the FFT is given as

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_28.gif

Using the definitions as

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_29.gif

The FFT finally is defined as

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_30.gif

Taking into account that the

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_31.gif

the following equation is obtained

C:\Documents and Settings\khec\Desktop\chap3\23_files\Image232_32.gif

That means the computing of the first half of the points of array *N* for *F(u)* is obtained with the first expression for *Feven* and the second half - with the expression for *Fodd* . The FFT is applied first for columns and than for rows.

*Another example of the program for FFT*

The following routine computes the discrete Fourier transform of a one-dimensional complex array *XIn* of the length *N=2 Power (logN*) and produces the one-dimensional complex array *Xout*. It uses an array W of the *N* complex *Nth* roots of unity, computed as shown, and an array *Bits* containing a bit-reversal table of length *N. N, LogN, W*, and *Bits* are all global to the subroutine as written. If the logical variable *Forward* is TRUE, the FFT is performed, if *Forward* is FALSE, the inverse FFT is performed.

**Entropy Coding**

* Fundamental concept => Generation of information can be modeled as a probabilistic process.
* A random event E that occurs with probability
* I(E) = log(1/P(E))= *- log P(E)* units of information.
* The quantity *I(E)* is called as the *Self-Information* of E
* If P(E) = 1 ( Event always occurs ) => I(E) = 0 ( No information is attributed to it ).
* In this case, no uncertainty is associated with the event
* If P(E) = 0.99 => E has occurred conveys some small amount of information.
* The base of the logarithm determines the unit used to measure information
* If the base m logarithm is used =>
* If log base 2 is used , P(E) =1/2. => I(E) =-log2(1/2) =1 bit (Amount of Info. conveyed)
* Suppose the gray level of pixels is generated by random variable
* I(rk) = - log (P(rk))
* Entropy => Average information content of an Image.
* It is given as
* Redundancy *R = Lavg – H* Where L
* If Lavg = H then R =0 => No Redundancy