GNR602
Advanced Methods in Satellite Image Processing

Instructor: Prof. B. Krishna Mohan

CSRE, IIT Bombay

bkmohan@csre.iitb.ac.in

Slot 13

Lecture 22-23 Texture Segmentation Methods

Contents of the Lecture

Concept of Texture

Co-occurrence matrix principle

- Texture features

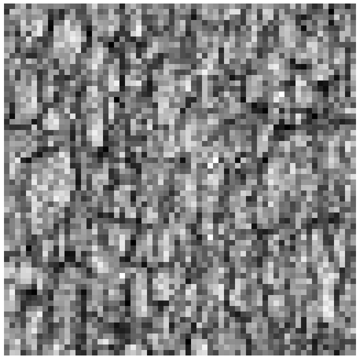
Gabor Texture Feature Extraction Method

Concept of Texture

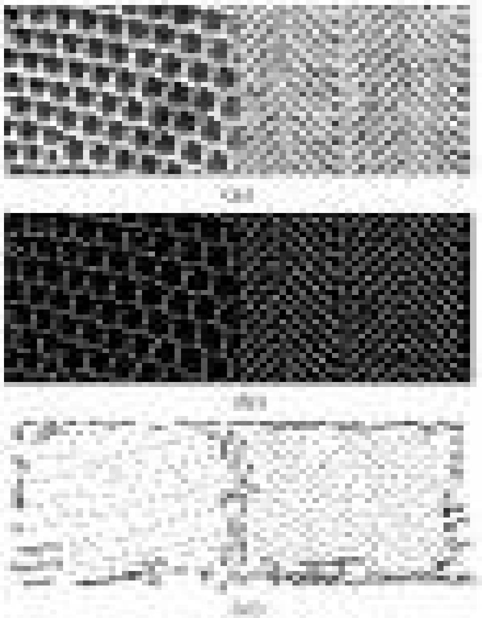
- Texture is an important visual cue
- What does texture mean? Formal approach or precise definition of texture does not exist!
- Texture discrimination techniques are for the part ad hoc.

Concept of Texture

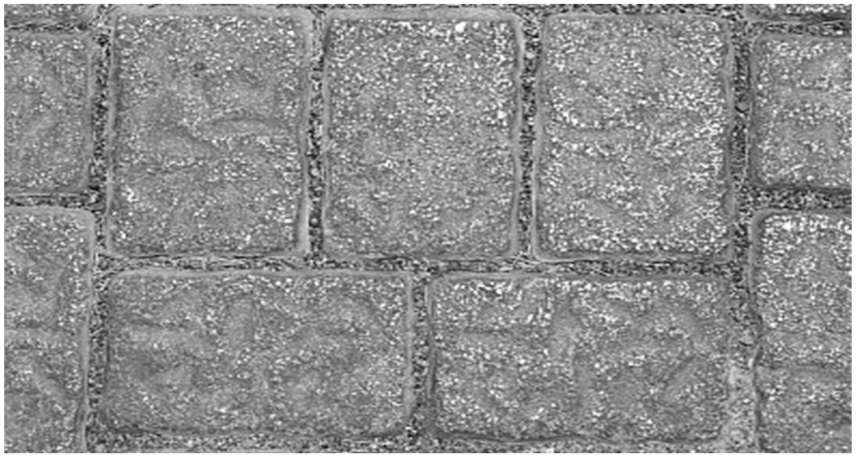
- Perception of texture is dependent on the spatial organization of gray level or color variations.
- Manmade features have a repetitive pattern, where a basic pattern or primitive is replicated over a region
- Large variation within the pattern leads to a textured appearance, while flat regions lead to a smooth appearance

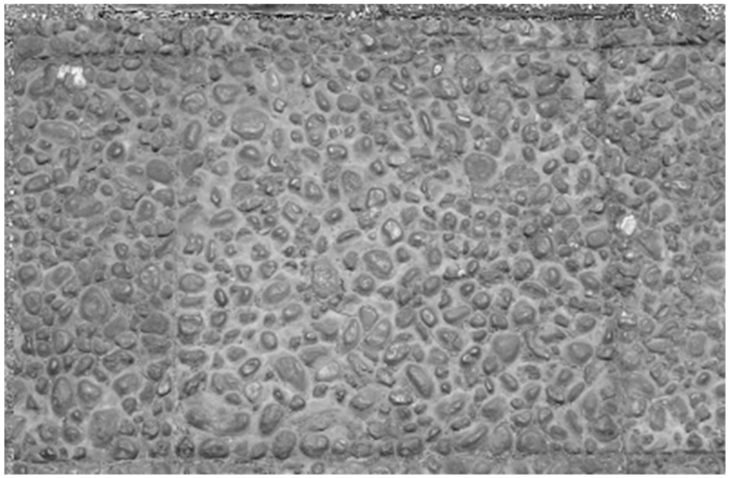


Sample Textures



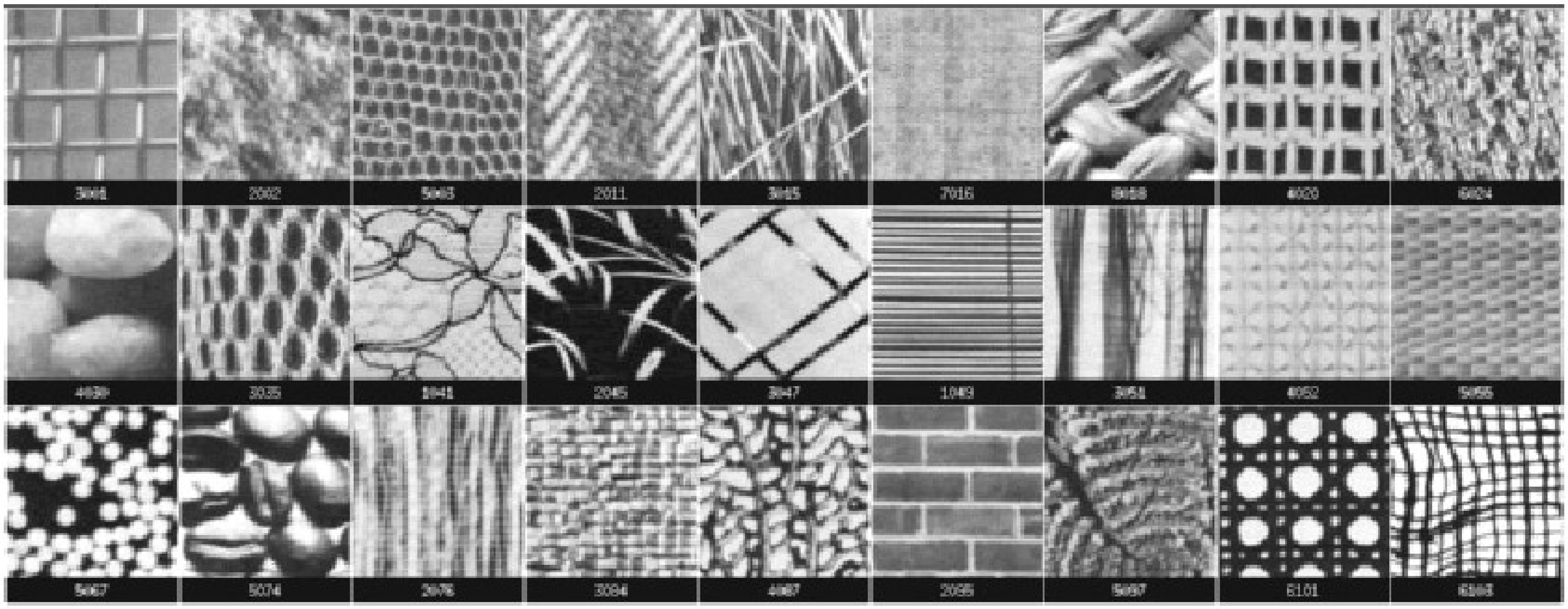
Sample Textures

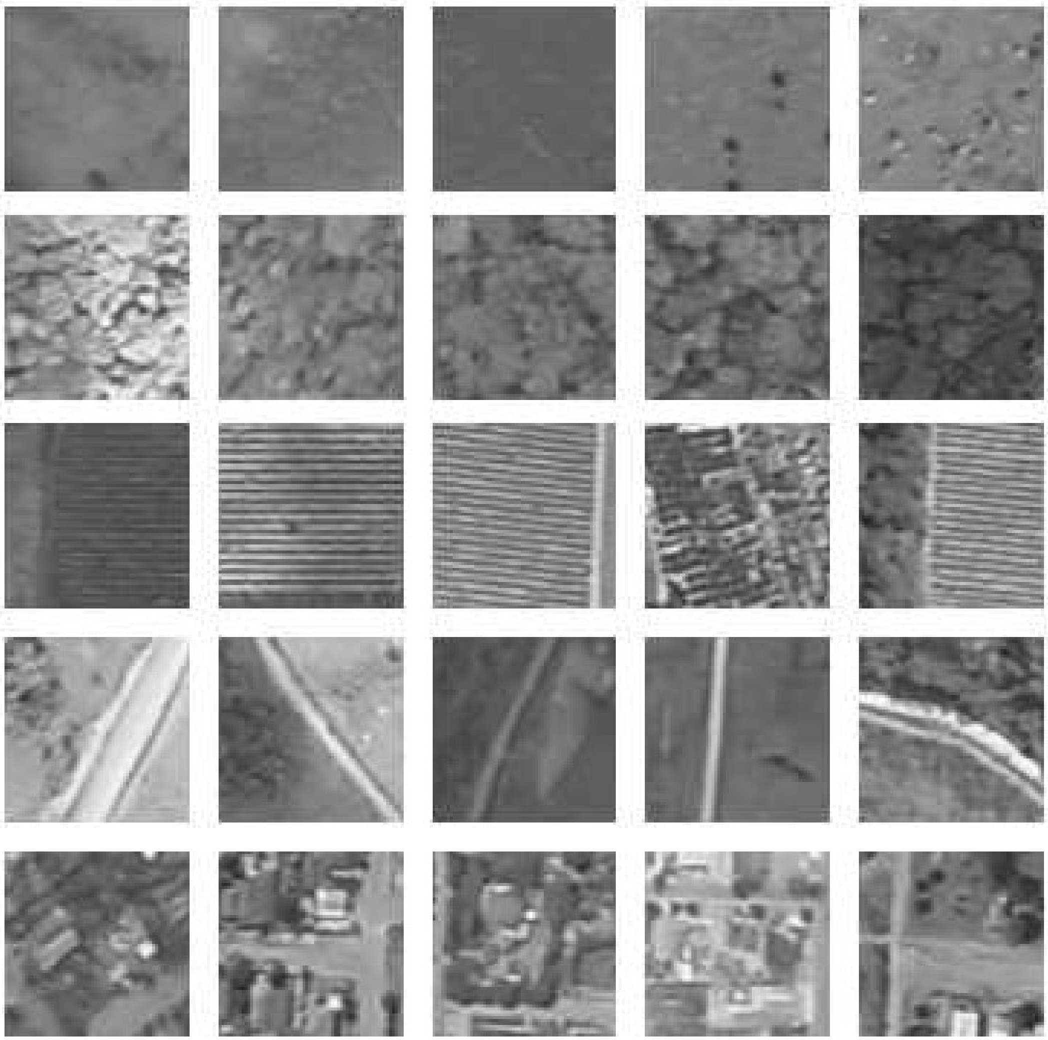




Source: www.pepfx.net

More Examples of Texture



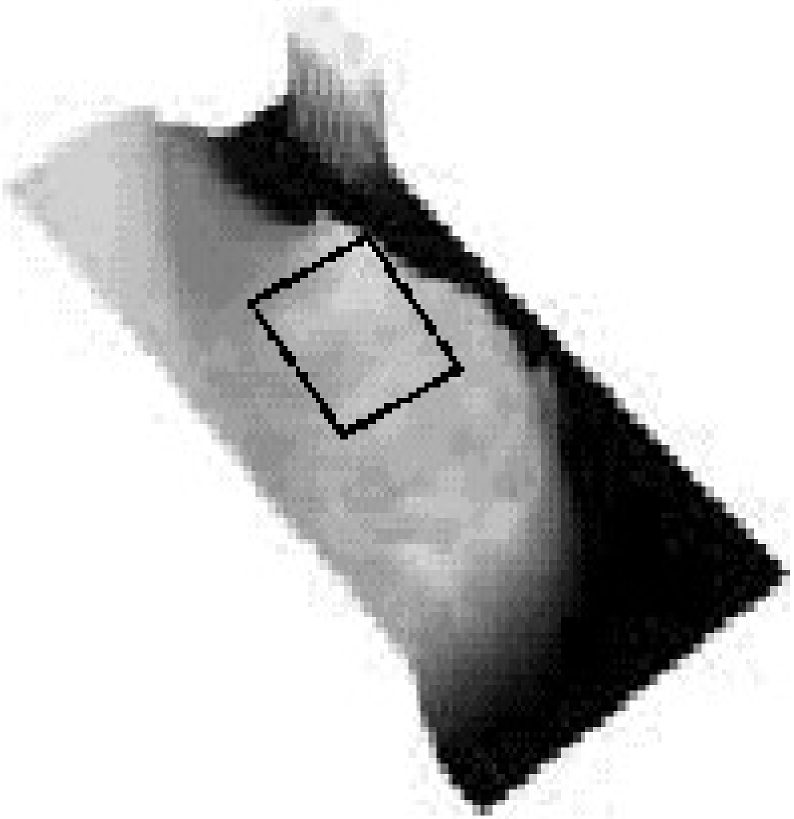


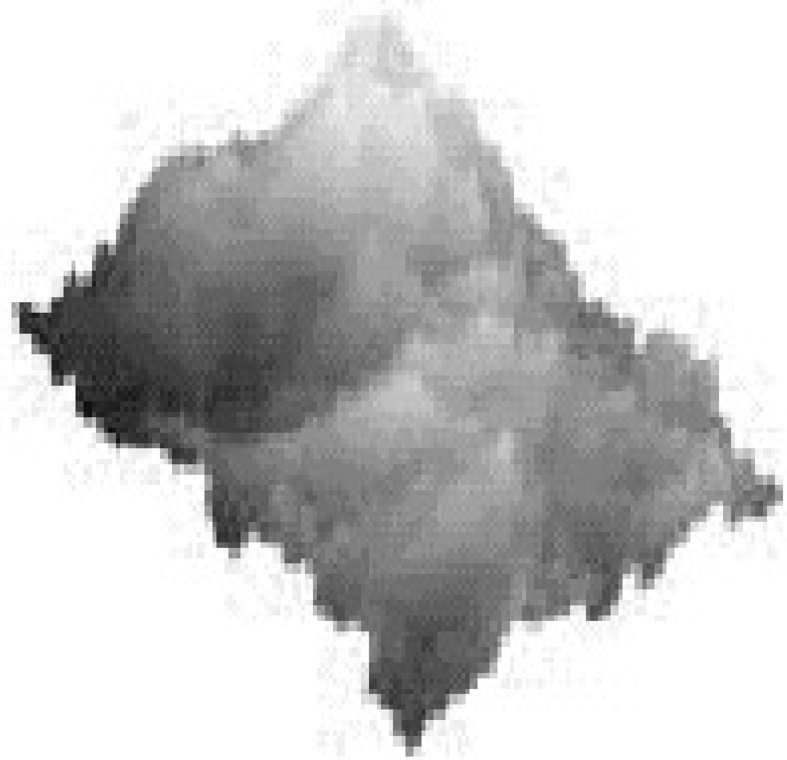
From Remotely Sensed Images

What is Texture?

- A feature used to partition images into regions of interest and to classify those regions
- Spatial arrangement of colours or intensities in an image
- Characterized by the spatial distribution of intensity levels in a neighbourhood
- A repeating pattern of local variations in image intensity
- An area attribute, not defined at a point

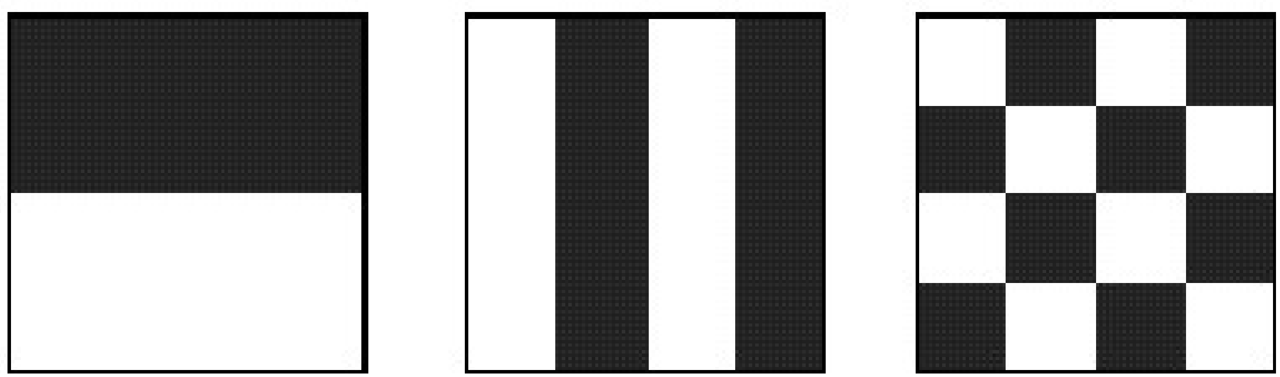
What is Texture?





Notion of Texture

- Suppose an image has a 50% black and 50% white distribution of pixels.
- Three different images with the same intensity distribution, but with different textures.



Composition of Texture

- Made up of texture primitives, called ***texels***.
- Can be described as fine, coarse, grained, smooth, etc.
- Tone is based on pixel intensity properties in the ***texel***, while structure represents the spatial relationship between ***texels***.
- If ***texels*** are **small** and tonal differences between texels are large a fine texture results.
- If ***texels*** are **large** and consist of several pixels, a coarse texture results.

Notion of Texture

- Statistical methods are particularly useful when the texture primitives are small, resulting in ***<u>microtextures</u>***.
- When the size of the texture primitive is large, first determine the shape and properties of the basic primitive and the rules which govern the placement of these primitives, forming ***<u>macrotextures</u>***.

Example of micro- and macro-texture

Description/Definition of Texture

- Non-local property, characteristic of region more important than its size
- Repeating patterns of local variations in image intensity which are too fine to be distinguished as separated objects at the observed resolution

Definition of Texture

- There are ***three approaches*** to describing what texture is:
- ***Structural*** : texture is a set of primitive texels in some regular or repeated relationship.
- ***Statistical*** : texture is a quantitative measure of the arrangement of intensities in a region.

  This set of measurements is called a *feature vector*.
- ***Modeling*** : texture modeling techniques involve constructing models to specify textures.

Texture Analysis

- Two primary issues in texture analysis:

  - ***texture classification***
  - ***texture segmentation***

- ***Texture classification*** is concerned with identifying a given textured region from a given set of texture classes.

  Each of these regions has unique texture characteristics.

  Statistical methods are extensively used.

- ***Texture segmentation*** is concerned with automatically determining the boundaries between various texture regions in an image.

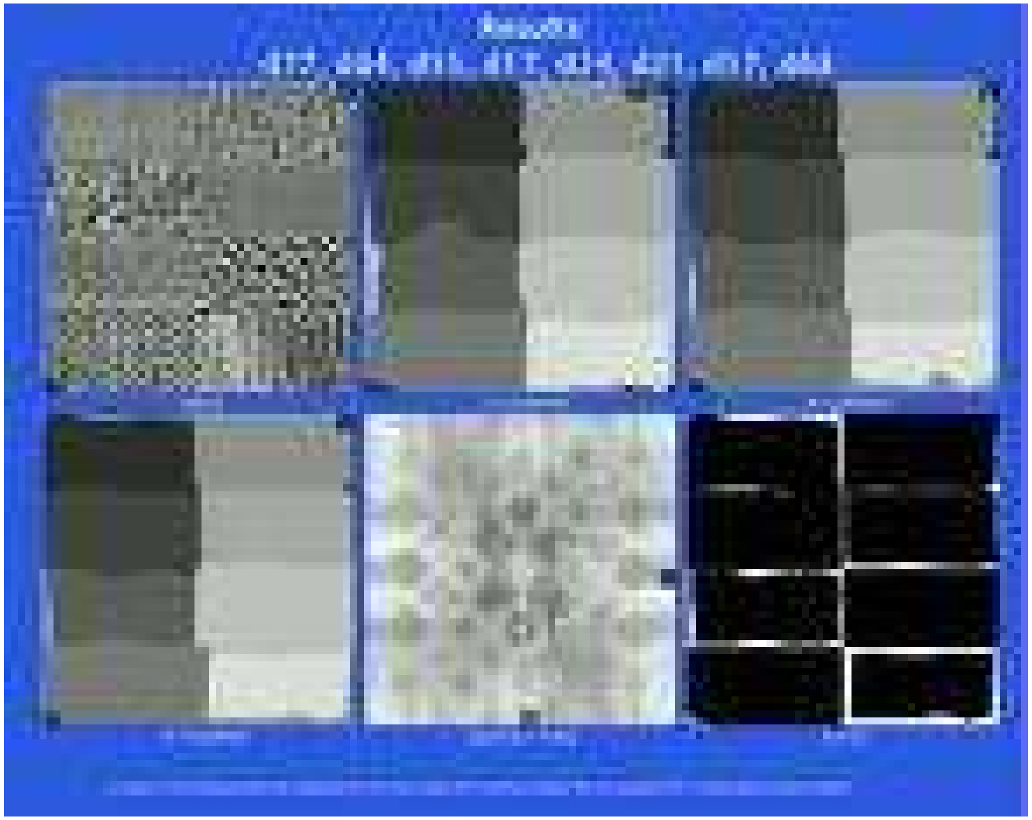
Texture Classification

- ***Texture classification*** is concerned with identifying a given textured region from a given set of texture classes.
- Each of these regions has unique texture characteristics.
- Statistical methods are extensively used.

Texture Segmentation

- ***Texture segmentation*** is concerned with automatically determining the boundaries between various texture regions in an image.
- Texture segmentation also results in regions homogenous with respect to texture property

Example



Approaches to Measuring Texture

Edge per unit area

First Order Statistics

- Mean / average, Standard deviation
- Mean Deviation, Range, Median,
- Skewness

Higher order statistics

Measuring energy in various frequency sub-bands

Fractal modeling

Geostatistical methods

Wavelet transform approaches

…

Edge per unit area

Textured areas are seen to be *rough* – spatial intensity variations over small patches

Gradient operators produce moderate edge magnitudes

Measuring the average edge magnitude over an area (e.g., over 11x11 or 15x15) can help separate textured areas from non-textured areas

Variance

In textured areas both high and low intensity pixels can be found

Variance of the pixel intensities over an area will be higher for textured areas compared to non-textured areas

Variance image (e.g., as in ERDAS software) can be used to represent texture

Directionality of Texture

Texture is a strong directional feature

e.g., horizontal stripes and vertical stripes are clearly perceived separately

Some texture features can provide directional information

Features like edge per unit area or variance cannot handle texture orientation

Gray Level Co-occurrence Matrix Approach

- GLCM is based on second order statistics (2-D histogram)
- It is conjectured (B. Jules, a psychophysicist) that textures differing in second order statistics are indeed different. ***(counter-examples provided later)***
- Therefore numerical features were extracted from the image in terms of the second-order statistics that were a measure of the underlying texture.

Construction of GLCM

- A co-occurrence matrix is a two-dimensional array, **P**, in which both the rows and the columns represent a set of possible image values.
- A GLCM $\mathbf{P_d}[i,j]$ is defined by first specifying a displacement vector **d**=(dx,dy) and counting all pairs of pixels separated by **d** having gray levels i and j.
- (dx,dy) define the directionality of texture; dx=1,dy=0 represents horizontal direction;dx=1,dy=1 represents diagonal direction

Definition of GLCM

- The **GLCM** is defined by:

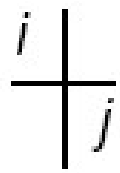
$$P_d(i,j) = n_{i,j} = \#\{f(m,n) = i,\ f(m+dx,\ n+dy) = j;\ 1 \le m \le M;\ 1 \le n \le N\}$$

– where $\mathbf{n_{ij}}$ is the number of occurrences of the pixel values **(i,j)** lying at distance **d** in the image.

– The co-occurrence matrix $\mathbf{P_d}$ has dimension **n× n**, where n is the number of gray levels in the image.

Example

2	1	2	0	1
0	2	1	1	2
0	1	2	2	0
1	2	2	0	1
2	0	1	0	1



$$P_d = \begin{array}{c} \\ \begin{bmatrix} 0 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 3 & 2 \end{bmatrix} \\ \begin{matrix} 0 & \;1 & \;2 \end{matrix} \\ j \end{array} \begin{array}{c} 0 \\ 1 \\ 2 \end{array} \quad i$$

There are 16 pairs of pixels in the image which satisfy this spatial separation. Since there are only three gray levels – 0,1,2, **P[i,j]** is a **3×3** matrix.

Algorithm to construct GLCM

Count all pairs of pixels in which the first pixel has a value ***i***, and its matching pair displaced from the first pixel by **d** has a value of ***j***.

This count is entered in the $\mathbf{i^{th}}$ row and $\mathbf{j^{th}}$ column of the matrix $\mathbf{P_d[i,j]}$

Note that $\mathbf{P_d[i,j]}$ is not symmetric in **this** form of counting, since the number of pairs of pixels having gray levels **[i,j]** does not necessarily equal the number of pixel pairs having gray levels **[j,i].**

Normalized GLCM

The elements of $\mathbf{P_d[i,j]}$ can be normalized by dividing each entry by the total number of pixel pairs.

Normalized GLCM **N[i,j],** defined by:

$$N[i,j] = \frac{P[i,j]}{\sum_i \sum_j P[i,j]}$$

which normalizes the co-occurrence values to lie between 0 and 1, and allows them to be thought of as probabilities.

Numeric Features from GLCM

Gray level co-occurrence matrices capture properties of a texture but they are not directly useful for further analysis, such as the comparison of two textures.

Numeric features are computed from the co-occurrence matrix that can be used to represent the texture more compactly.

Haralick Texture Features

Haralick et al. suggested a set of 14 textural features which can be extracted from the co-occurrence matrix, and which contain information about image textural characteristics such as homogeneity, linearity, and contrast.

Haralick, R.M., K. Shanmugam, and I. Dinstein, "Textural features for image classification" IEEE Transactions on Systems, Man and Cybernetics: pp. 610-621. 1973.

Features from GLCM: Angular Second Moment (ASM)

- Angular Second Moment ASM
- ASM = $$\sum_{i=1}^{K}\sum_{j=1}^{K} P_d^2(i,j)/R$$
- R is a normalizing factor
- ASM is large when only very few gray level pairs are present in the textured image
- K is the number of gray levels

Contrast (CON)

- Contrast CON
- CON = $$\sum_{i=1}^{K}\sum_{j=1}^{K}(i-j)^2 P_d(i,j)/R$$
- This feature highlights co-occurrence of very different gray levels

Entropy (ENT)

- ENT = $$\sum_{i=1}^{K}\sum_{j=1}^{K} P[i,j]\ln\left(\frac{1}{P[i,j]}\right)$$
- ENT emphasises many different co-occurrences
- P(i,j) is the normalized co-occurrence matrix, each entry indicating probability of occurrence of that gray level combination

Inverse Difference Moment (IDM)

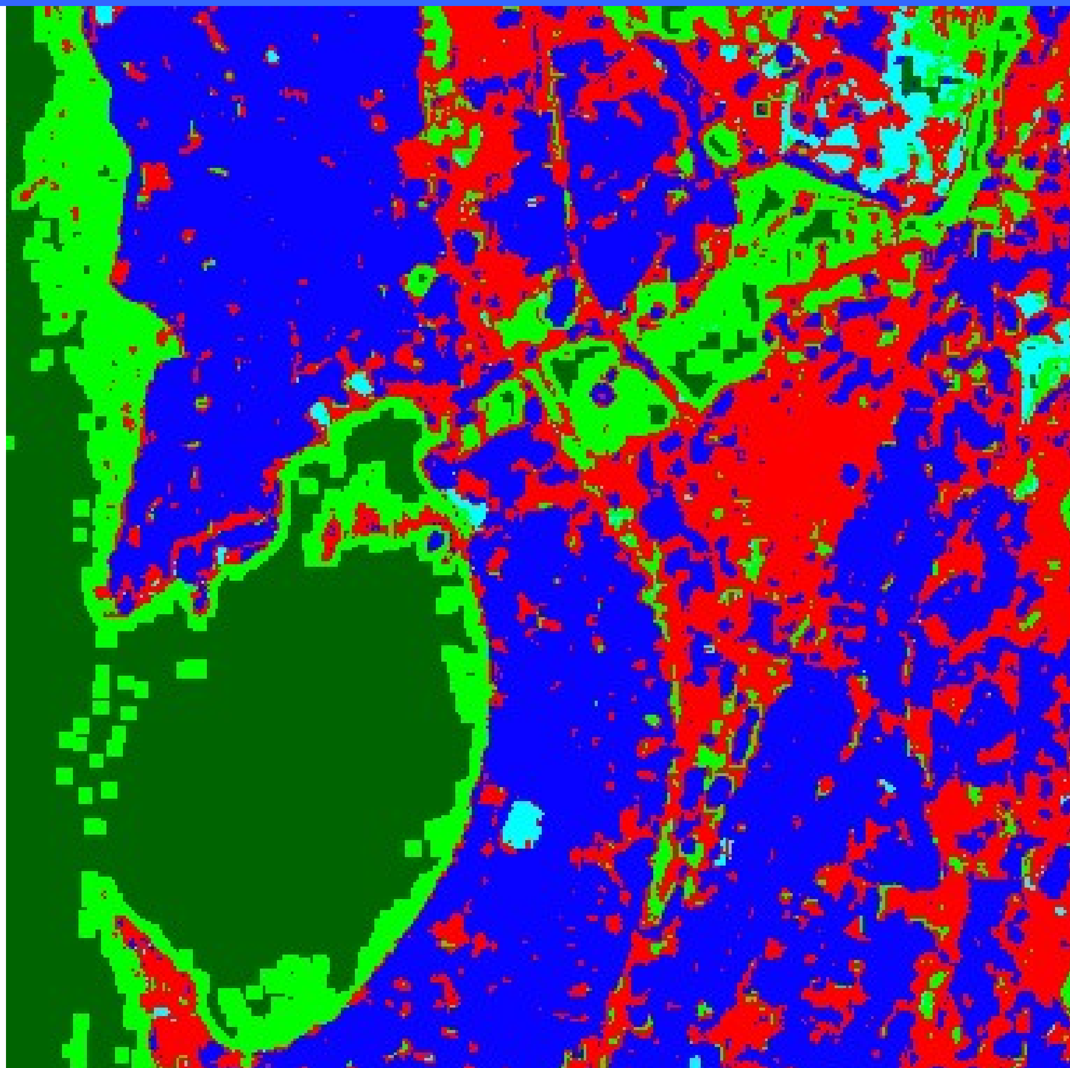
- Inverse Difference Moment IDM
- IDM = $$\sum_{i=1}^{K}\sum_{j=1}^{K}\frac{P_d^{\,r}[i,j]}{|(i-j)^m|_{i\neq j}}$$
- IDM emphasises co-occurrence of close gray levels compared to highly different graylevels. m and r can user specified

Algorithm for image segmentation

- Specify a window of size wxw
- For the pixels in the window, compute the co-occurrence matrix
- Derive the texture features from the co-occurrence matrix
- Move the window by 1 pixel, and repeat the procedure
- The procedure leads to texture images that may be treated like additional bands, equal to the number of features computed.

Input Image

IDM Feature



CLASSIFIED IMAGE (Mumbai)

CLASSIFIED IMAGE (Mumbai)

LEGEND

- WATER
- MARSHY LAND / SHALLOW WATER
- HIGHLY BUILT-UP AREA
- PARTIALLY BUILT-UP AREA
- OPEN AREAS/ GROUNDS

Other Features from GLCM

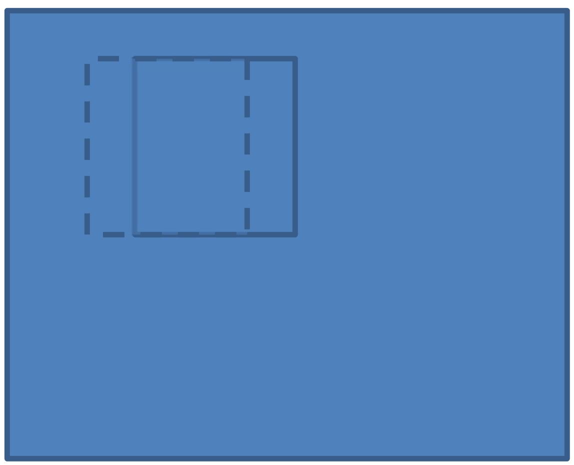
- More features are defined from GLCM by Haralick et al. and by other researchers
- e.g., Sahasrabudhe and Nageswara Rao used eigenvalues of GLCM as texture features
- 1st and 2nd eigenvalues of GLCM were shown to be capable of good texture discrimination
- Limited utility due to computational intensive nature of features
- Haralick et al. defined 28 features of which ASM, ENT, CON, IDM were most effective

Fast Computation of GLCM

- Fast Computation of GLCM useful for efficient application
- Basis for fast computation – number of pixel pairs that are common to computation of GLCM / features at successive positions of the window
- Significant savings possible when window size is large, and many features are computed

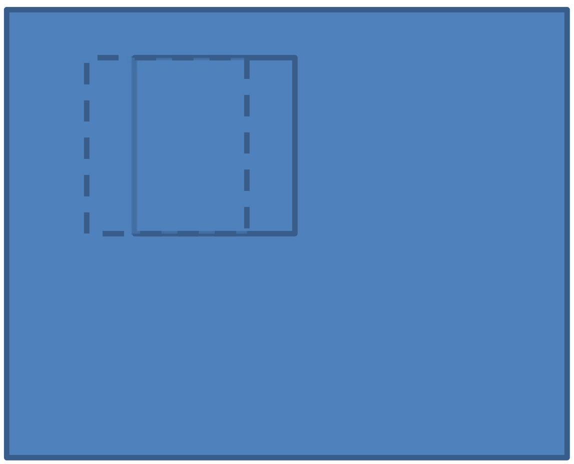
Redundant Computations

- When texture window moves by 1 pixel right,
  - The first column moves out of the computation
  - The last column enters the computation
  - Many pixel pairs remain unchanged



Efficiency Considerations

- Incremental Adjustments
  - Deduct the pairs formed with elements of first column
  - Add pairs formed with elements of last column
  - New matrix is ready



Efficiency Considerations

- Direct computation of features
  - Examine each feature
  - Make modifications to the feature directly instead of to the GLCM
- ASM = $$\sum_{i=1}^{K}\sum_{j=1}^{K} P_d^2(i,j)/R$$
- Deduct from P_d (i,j) for column moving out
- Add to P_d (i,j) for column coming in

Fast Computation

- ASM and CON can easily be adjusted for changes in GLCM due to window shift
- ENT and IDM may be difficult

Laws' texture energy measures

- **Laws' Texture description uses**
  - average gray level
  - edges
  - spots
  - ripples
  - waves

Laws Texture Energy Approach

- Some of the masks
- L5 = [1 4 6 4 1];
- E5 = [-1 -2 0 2 1]
- S5 = [-1 0 2 0 -1];
- W5 = [-1 2 0 -2 1]
- R5 = [1 -4 6 -4 1]

Texture Energy

2-dimensional masks are created by cross-products of the one-dimensional masks

The masks are used as filters to convolve with the input image

Energy within specified window sizes WxW is computed

Energy and Texture

- E5L5 L5= [1 4 6 4 1]^t; E5 = [-1 -2 0 2 1]^t

E5L5

-1	-4	-6	-4	-1
-2	-8	-12	-8	–2
0	0	0	0	0
2	8	12	8	2
1	4	6	4	1

Cross product of the two vectors of size 5x1 resulting in a mask of size 5x5

Laws Filter Masks

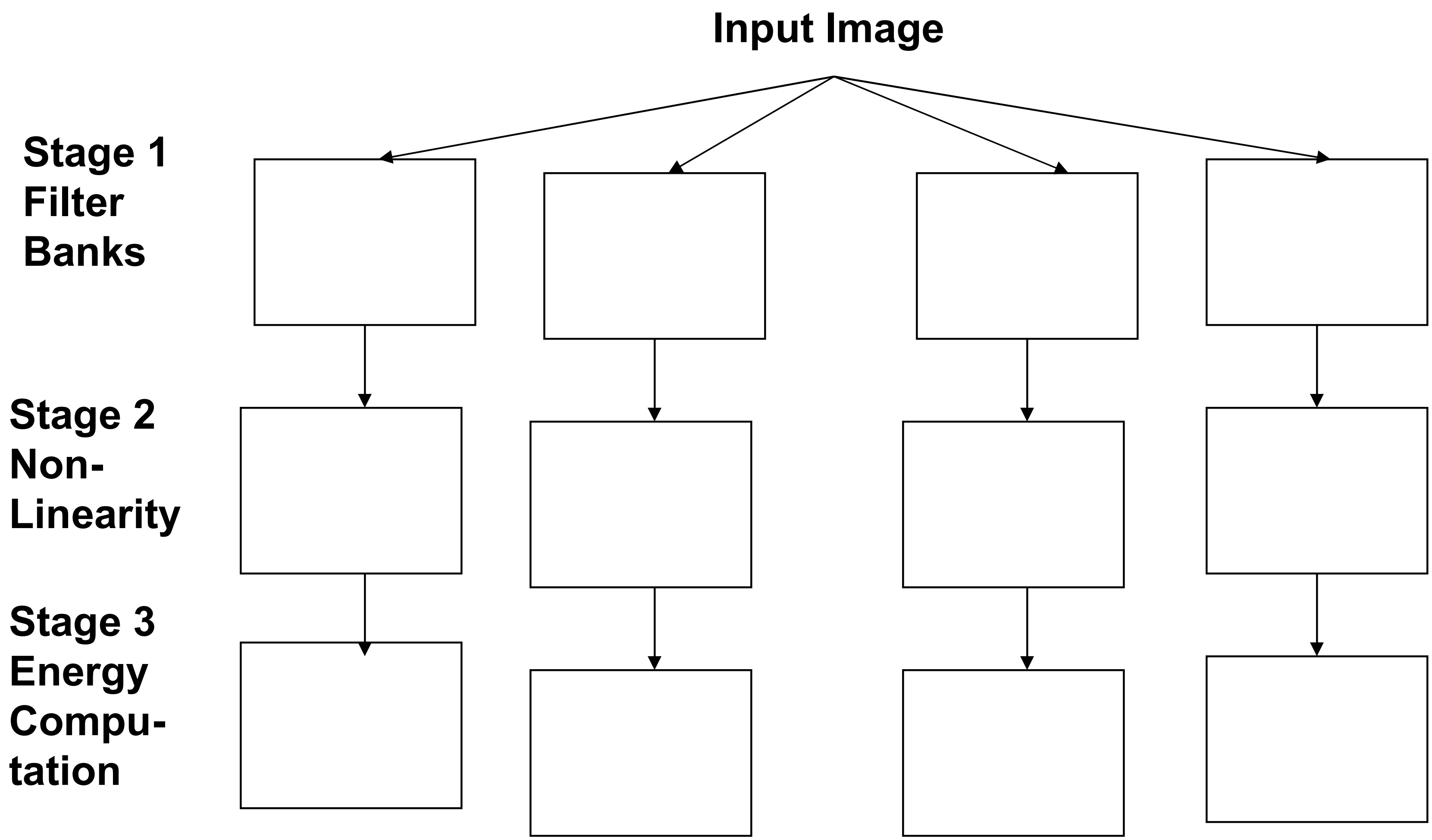
- Another filter mask:

$$L_5^T \times S_5 = \begin{bmatrix} -1 & 0 & 2 & 0 & -1 \\ -4 & 0 & 8 & 0 & -4 \\ -6 & 0 & 12 & 0 & -6 \\ -4 & 0 & 8 & 0 & -4 \\ -1 & 0 & 2 & 0 & -1 \end{bmatrix}$$

Texture Based Analysis

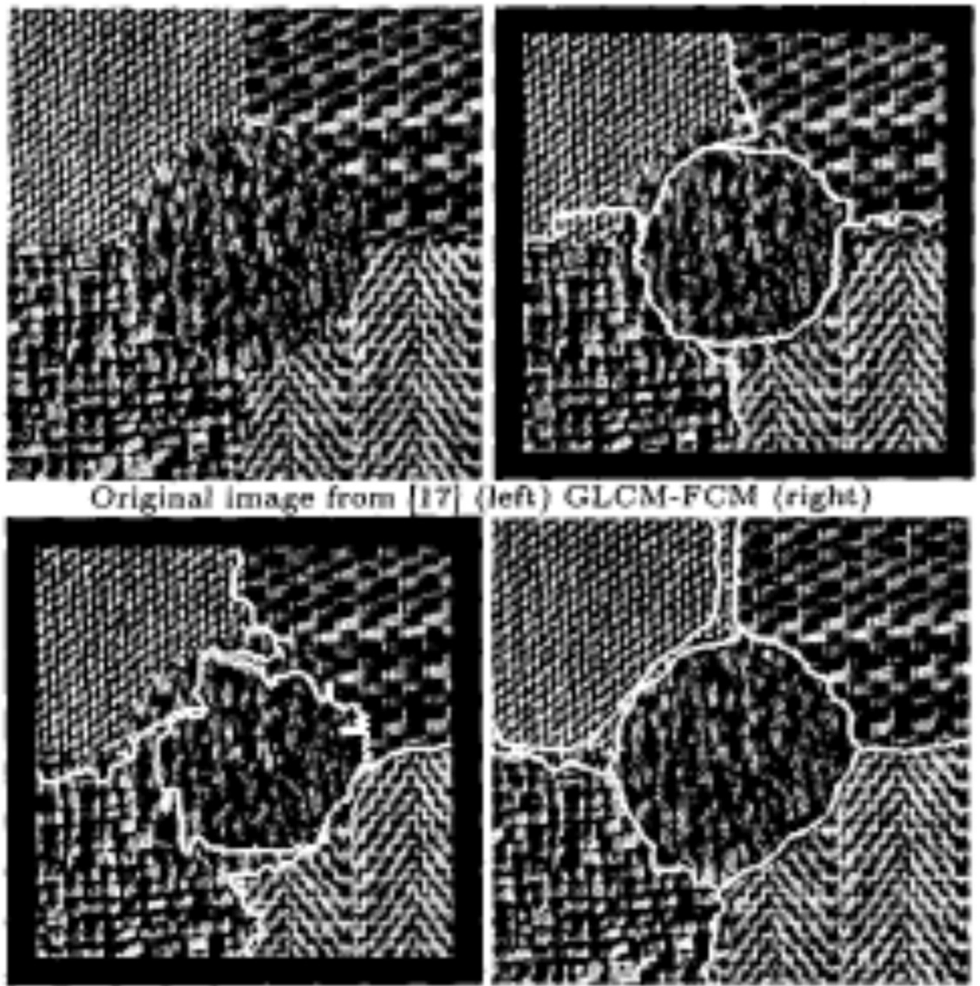
Steps

- Convolve input image with Laws' masks
- Nonlinearly transform Laws' filter outputs using sigmoid or similar function
- Using a moving window (or size 15x15 or similar), find energy in each window position
- Generate texture energy images
- Classify using standard classifiers



Gabor filter bank for texture analysis

Illustration



Original image from [17] (left) GLCM-FCM (right)

LAWS-FCM (left) GABOR-CLST (right)

Transform Domain Texture Analysis

- Rapid or slow changes in local intensity or color in the spatial domain corresponding to high or low frequency components in the transform domain
- Orientation in one domain is reflected in the other domain in a perpendicular direction
- Texture therefore can also be analyzed efficiently in transform domain as well

Transform Domain Texture Analysis

- Among the transform domain approaches, Gabor functions and wavelets are popularly used.
- Essentially the image is decomposed into different frequency bands. Rapid local tonal variations correspond to higher frequencies, while slow variations correspond to lower frequencies

Gabor Filters for texture analysis

- Multi-channel filtering approach to texture analysis are:
- Functional characterization of the channels and the number of channels
- Extraction of appropriate texture features from the filtered images
- The relationship between channels (dependent vs independent)
- Integration of texture features from different channels to produce a segmentation

Gabor Textures

Different multi-channel filtering techniques that are proposed in the literature differ in their approach to one or more of the above issues.

Gabor Filters

- The frequency spectrum of the image is divided into radial and angular ranges such that there are N_r x N_o windows in the frequency domain
- Retaining each such window, the inverse transform is computed to generate a bandpass filtered input image
- This process is based on the early processing of visual information in the human visual cortex

Gabor Filter Bank

Impulse response and Fourier transform of a real-symmetric Gabor filter

$$h(x,y) = \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\}\cos(2\pi u_0 x).$$

$$H(u,v) = A\left(\exp\left\{-\frac{1}{2}\left[\frac{(u-u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\} + \exp\left\{-\frac{1}{2}\left[\frac{(u+u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\}\right).$$

where $\sigma_u = 1/2\pi\sigma_x$, $\sigma_v = 1/2\pi\sigma_y$, and $A = 2\pi\sigma_x\sigma_y$.

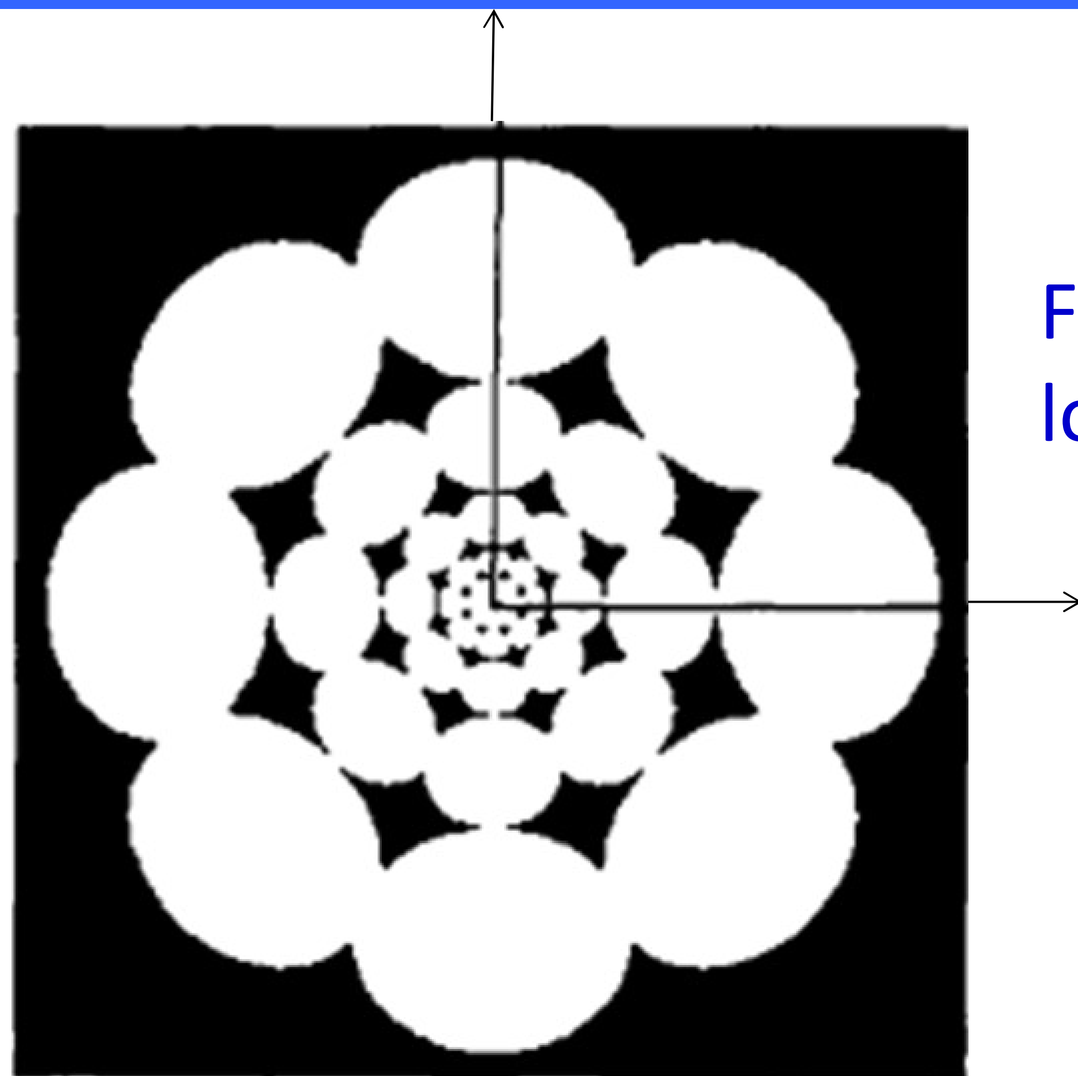
Gabor Filter Bank

Instead of using ideal low/high/bandpass filters, Gaussian shaped filters are used for stability

$$h(x, y) = \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\}\cos(2\pi u_0 x).$$

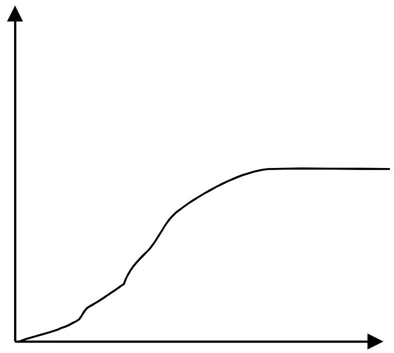
$$H(u, v) = A\left(\exp\left\{-\frac{1}{2}\left[\frac{(u-u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\} + \exp\left\{-\frac{1}{2}\left[\frac{(u+u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\}\right).$$

where $\sigma_u = 1/2\pi\sigma_x$, $\sigma_v = 1/2\pi\sigma_y$, and $A = 2\pi\sigma_x\sigma_y$.



Frequency lobes

Stages in Gabor Filter Banks Approach



Nonlinear Transformation

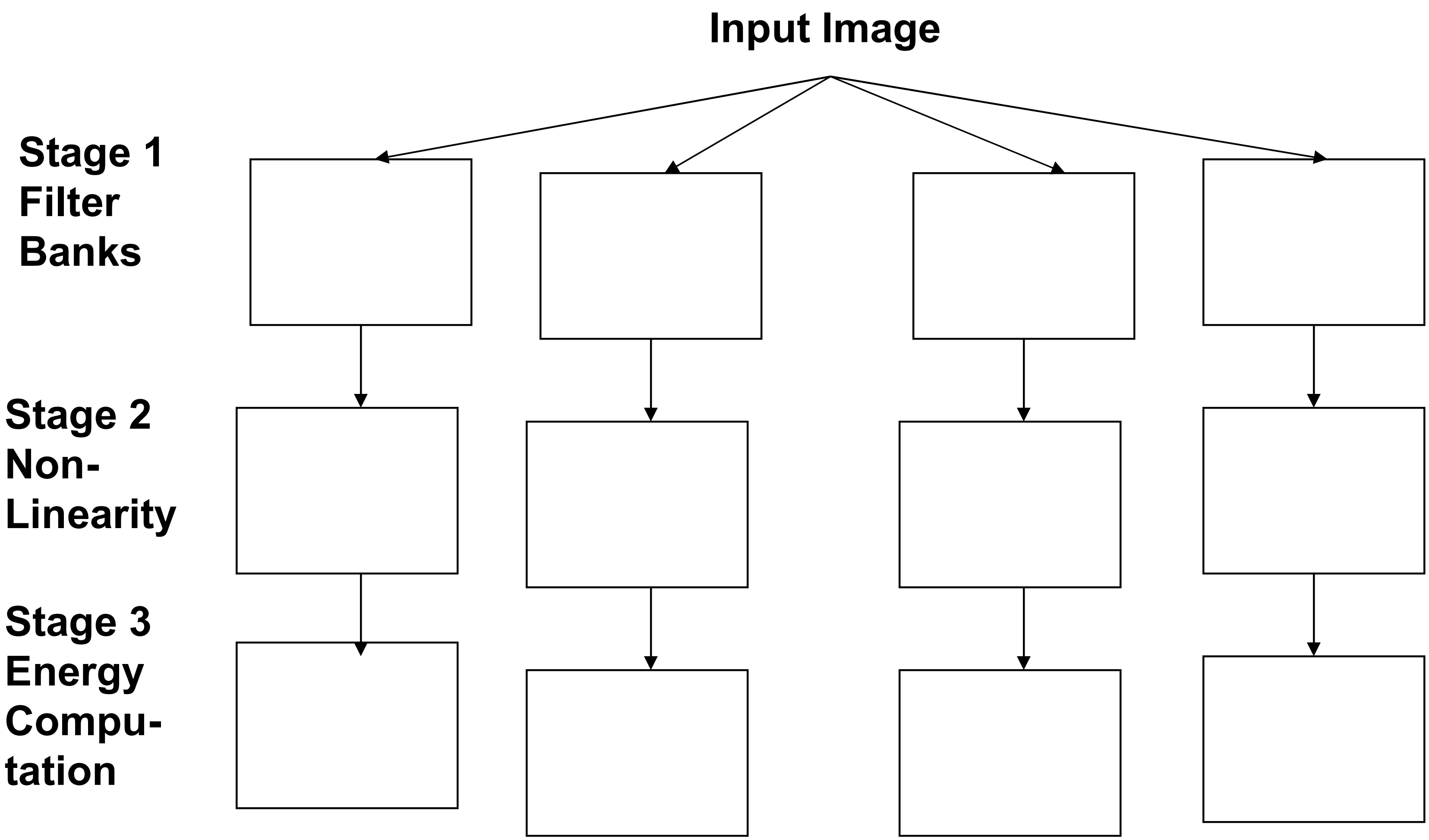
-Bandpass filtering using radial and angular sector basis

-Nonlinear transformation on the output of bandpass filtering

-Energy computation in moving window of size like 11x11 or 15x15 etc.

-Classification of texture images

-Number of texture images = number of bandpass filters



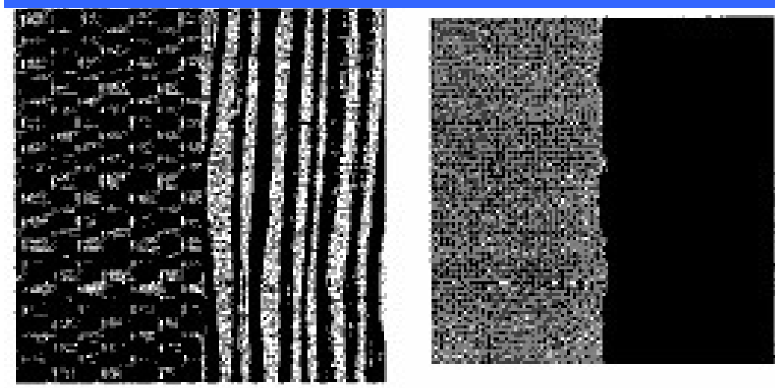
Gabor filter bank for texture analysis

Nonlinear transformation

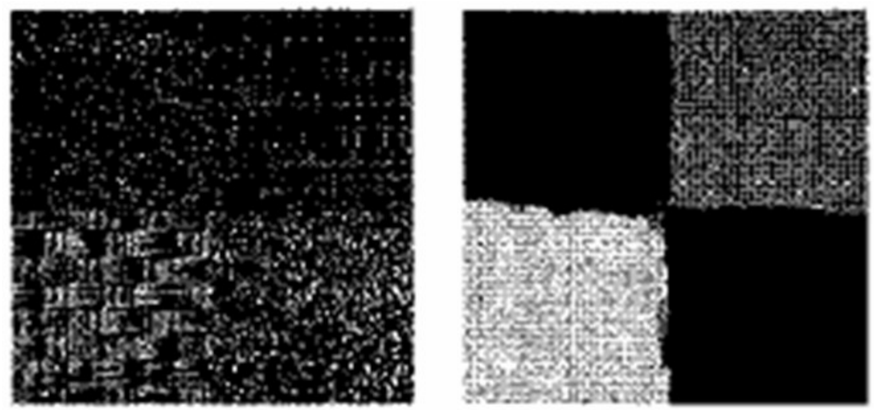
- The inverse Fourier transformed images are passed through a nonlinear function, similar to the way the neural computation happens in the human system
- Typical nonlinearity is a hyperbolic tangent function
- $\tanh(\alpha(t)) = [(1 - \exp(-2\ \alpha t)) / (1 + \exp(-2\ \alpha t))]$

Energy Computation

- From the nonlinear transformed images, texture energy is computed over local windows of user specified size
- Jain and Farrokhnia, who first proposed this approach, used 4 intervals in the radial direction, 4 angular sectors for orientation preference, one lowpass and one highpass filter, in all 18 texture images derived from the input image
- Ref: Pattern Recognition, vol. 24, pp.1167-1186, 1991



Illustration



Improved version can be found at **"Integrating Region and Edge Information for Texture Segmentation …", Image and Vision Computing, Vol. 26, No. 8, pp. 1106-1117, August 2008.**

Contd…