QUESTION FOR FINAL EXAM

1. Histograms

Texture is made up of repeated local patterns, so to represent texture we need to:

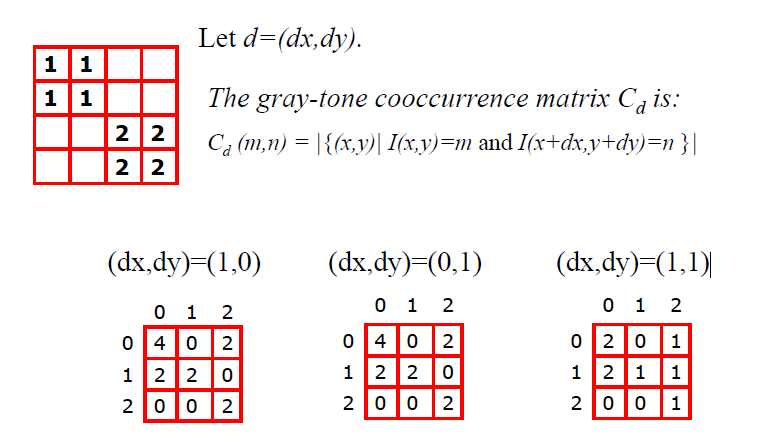
* Use filters that look like patterns and consider magnitude of response to find the patterns.
* Use mean, standard deviation, histogram and histogram of “prototypical” feature occurrences to describe their statistics.

*Comparing histograms is the simplest texture discrimination.*

Dividing intensities into discrete ranges and counting how many pixels in each range are the steps to compare histograms. One may calculate chi square distance between texton histograms. Distance reveals dissimilar texture.

*Texture representation:*

Original image is filtered, squared and statistics to summarize patterns in small windows. Mean d/dx and d/dy values in statistics table are used to group consistent pixels in windows with primarily horizontal and vertical edges. Finally, compare dissimilarity texture by computing distance between pixels in groups.

1. Co-occurrence Matrices and Features
2. 
3. Filter banks

Before, one can used two filters, and result in a 2-dimensional feature vector to describe texture in a window. Besides, we can generalize to apply a collection of multiple (d) filters: a “filter bank” and result in d-dimensional feature vectors.

We want to have filters with combination of scales and orientations, different types of patterns and put them into bank.

1. Gabor Filters(chứng minh )

From Multivariate Gaussion,

…..

Have Gabor Filters: at different scales and spatial frequencies.

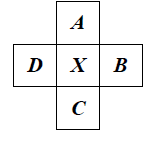
1. Markov Radom Field

A Markov random field (MRF)

•generalization of Markov chains to two or more dimensions.

First-order MRF:

•probability that pixel X takes a certain value given the values of neighbors A, B, C, and D:

1. Markov Chain – Transition Table

Markov Chain

•a sequence of random variables x1,… , xn

• is the state of the model at time t: xt

1. Texture synthesis:

***Goal: create new samples of a given texture***

***Most basic algorithm***

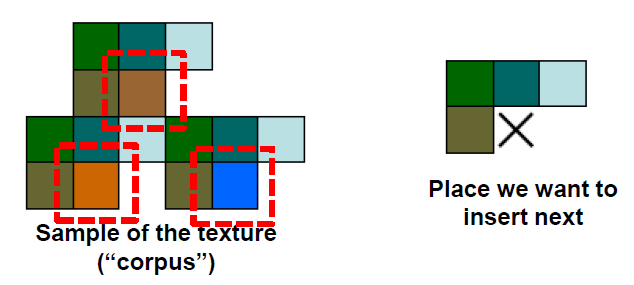
Build probability histogram

* find all blocks of N consecutive words/letters in training documents
* compute probability of occurrence

***Texture synthesis: intuition***

Before, we inserted the next word based on existing nearby words…

Now we want to insert pixel intensities based on existing nearby pixel values.



***Synthesizing One Pixel***

- What is P (x| neighborhood of pixels around x)?

- Find all the windows in the image that match the neighborhood

- To synthesize x

+ pick one matching window at random

+ assign x to be the center pixel of that window

An exact neighborhood match might not be present, so find the best matches using SSD error and randomly choose between them, preferring better matches with higher probability

1. Unit of synthesis – block

Image Quilting

Observation: neighbor pixels are highly correlated

Idea: unit of synthesis = block

•Exactly the same but now we want P (B|N(B))

•Much faster: synthesize all pixels in a block at once

Input image with block form are performed random placement of blocks. Neighboring blocks constrained by overlap and then, minimal error boundary cut

1. How to detect circle? (transform)
2. How to transform general line?
3. Prove : = tan-1()
4. Derivation theorem of convolution
5. How to transform continue Gaussion fuction( Line problem )
6. Sharpening