CSCI 478/578 Multimedia Data Processing/Multimedia Systems

Fall, 2013

Assignment 3, 100 possible points (12%)

Preliminary Demo to Peer: in class of October 16 (Wednesday)
Final Demo to Peer: in class of October 30 (Wednesday)
Program & Report Due by: November 3 (Sunday)

Assignment Description:

This project is to implement a Content-Based Image Retrieval system 1) using three different image texture features and 2) using relevance feedback.

1. Test Image Database
   This test image database includes 100 true-color images in .jpg format (the same set of images as in Project 1).

A. Texture Features

(1) Co-Occurrence Matrix

A *co-occurrence* matrix is a two-dimensional array *C* in which both the rows and the columns represent a set of possible image values *V*. In this project, again the intensity values for a given RGB color image are used:

$$I = \lfloor 0.299R + 0.587G + 0.114B \rfloor \quad (1)$$

In this case, $V \subseteq \{0, 1, \ldots, 255\}$. The value of $C_d(i, j)$ indicates how many times value *i* co-occurs with value *j* in some designated spatial relationship *d* where *d* is a displacement vector [*dr*, *dc*] specifying the displacement between the pixel having values *i* and the pixel having value *j*. *dr* is a displacement in rows (downward) and *dc* is a displacement in columns (to the right). The gray-tone co-occurrence matrix C_d for image *I* is defined by:

$$C_d[i, j] = \left| \{[r,c] \mid I[r,c] = i \ \ and \ \ I[r+dr, c+dc] = j\} \right| \quad (2)$$

where $i, j \in V$

2	2	0	0
2	2	0	0
0	0	3	3
0	0	3	3

Image *I*

$C_{[1, 1]}$

i \ *j*	0	2	3
0	2	0	2
2	2	1	1
3	0	0	1

In this project, you need to compute the **gray-tone co-occurrence matrix** $N_{[1, 1]}$ (***d* = [1, 1]**) for all the images in the database. The Figure above gives some sample data. You may use it to check the correctness of your program. For this 4×4 image *I*, *V* = {0, 2, 3}. The co-occurrence matrix $C_{[1, 1]}$ is obtained with *d* = [1, 1], indicating that value *j* occurs immediately to the bottom-right of value *i*.

(2) Normalized Gray-Tone Co-Occurrence Matrix N_d

The normalized gray-tone co-occurrence matrix N_d is defined by:

$$N_d[i, j] = \frac{C_d[i, j]}{\sum_i \sum_j C_d[i, j]} \tag{3}$$

In this project, you need to compute the **normalized gray-tone co-occurrence matrix** $N_{[1, 1]}$ (***d* = [1, 1]**) for all the images in the database.

(3) Three Texture Features

The following are three standard features derived from a **normalized co-occurrence matrix**:

$$Energy = \sum_i \sum_j N_d^2[i, j] \tag{4}$$

$$Entropy = \sum_i \sum_j N_d[i, j] \log_2 (N_d[i, j]) \tag{5}$$

$$Contrast = \sum_i \sum_j (i - j)^2 N_d[i, j] \tag{6}$$

(4) Still use **Manhattan Distance** as distance metric

B. Relevance Feedback

Implement system based on the RF algorithm (will discussed in next class). Detailed requirements:

1. Use color-code based histogram together with 3 texture features as the feature set for each image.
2. Use Gaussian normalization for feature normalization
3. Use simplified RF version (i.e., different from the original RF framework discussed in the reference paper [Rui98] that will be distributed in next class)
    a. Use the normalized feature matrix and initial weights (no-bias weights) to return initial query results
    b. On the GUI, only two levels of relevance are required: relevant and non-relevant
    c. Collect user's feedback, update the feature weights
    d. Return updated query results and go through iterations (step c & d)
    e. Distance metric:
    $$D(I, J) = \sum_i \omega_i \cdot |V_i(I) - V_i(J)|$$

NOTE: In updating the weights, if standard deviation st_i for a feature i of all the relevant images is 0

a. and its mean value m_i is not 0, set st_i to be 0.5*min(non-zero standard deviations of all the features). Then calculate the feature weight W_i
b. and m_i is 0, set $W_i = 0$

Preliminary Demo to Peer:

1. User query interface

Similar to Project 1 except for more retrieval options and user feedback. Retrieved images should be presented in a decent way: the entire image should be visible to users.

2. Implementation of texture based retrieval: demo the query results using the three different texture feature methods (separately) described above.
3. Any other items you like.

Final Demo to Peer:

1. Demo the query results using relevance feedback for several iterations.
2. Users are allowed to switch among different query methods (intensity, color-code, texture, and RF) within the same application.
3. Any other items you like.

Submission Requirements:

The **softcopy of well-commented code (& an executable ready for testing) and report** (how to run the program and how to use the system, screen dumps to show the first page of retrieval results for **query image 1.jpg** using 3 texture feature methods and using relevance feedback with **first 3 iterations**, and corresponding **precision values for first page** of results in each case) needed to be submitted to Moodle by the due time. Also a **hardcopy of the report** needs to be submitted in the first class after the due time.

You will also be asked to do a preliminary demo and final demo to your peers (see **Preliminary Demo and Final Demo** sections).

Extra requirement for CS 595:

Implement another 2 simplified RF versions, using 1) color code + intensity 2) color code + intensity + 3 texture features. Provide users the option to pick any one of these 3 RF methods. **In your report**, discuss how your system is well designed to handle the change of features.

Evaluation Criteria: - 100 points

Correctness (testing):	60 pts
User friendly:	20 pts
Report and comments:	20 pts