# Multimedia Database

**Name: Aparajita Sahay**

**Date: 4/14**

**Assignment 1:**

1. Coding Environment used: Visual Studio Community 2013. I have developed a Windows Presentation Foundation (WPF) application.
2. Technology used: C# and XAML.
3. Steps to run the program: **Create a “images” folder at the same location where WpfApplication4.exe is located or saved. “images” folder consist 100 images provided as the sample image.**
4. Step by step method to use the system:

* **WpfApplication4.exe will take around 1-2 minutes to load**. Because every time the program is loaded each image pixel value is computed and is kept inside the histogram-bins.
* Once program is loaded, a user interface is displayed. It will show list of images. Each page will show 20 images.
* System will have radio buttons for Intensity method and color-code method, a reset button and page navigation button next and previous.
* Select an image then choose radio button which method you want to see, once the method is selected, result will be shown.
* If user selects a radio button or method type without selecting the image, a message box will be displayed to select image before moving ahead.
* Once any radio button is selected, relevant images are shown from page 1 to page 5. Each page has 20 images. Page navigation can be done using next and previous button. Next and previous button will be automatically enabled and disabled depending upon the available images. Read image from left to right.

1. Development:

* I implementing assignment1, I have developed all the code by myself. I have designed user interface using XAML. I have used the concept of data binding to display images in UI. For layout I have used Grid and stack panel. To display images I have used ListBox control. Added items to the list and have binded the ItemsSource property to a data source.
* In the c# code, I have used bitmap GetPixel() method (Sytem.Drawing) to get pixel value of an image.
* Used Dictionary to store histogram value for a method.

1. Advantage/Limitation of using the System:

**User:**

* As a user I can find images similar to my query image from large database of images in one click

**Limitation as a user:**

1. Performance: As a user this system takes longer time to load and compute pixel value for the first time.
2. Semantic gap: As a user I get the results but few results are not what I was expecting to see in the top rankings.
3. In my view, for images 33.jpg , color code method gives more relevant result as the top result than Intensity method. But it is my perception to relate images with more orange (similar color) to be similar and relevant. But every user has different perception about the image.

Advantage as a system Designer:

1. As a system designer, I don’t have to auto tag the image. Since every user have different perception about the image, tags can vary depending upon the user perception about the image. So, by using methods such as intensity and color code, now we can use the algorithm to compute similarity between the query image and rest of the image.
2. Manual image labeling, is practically difficult for exponentially increasing image database, therefore using these method we can retrieve relevant image using relevant algorithm and techniques

**Limitation as a system designer**:

1. As a system designer, I am using color feature of the image (rgb value of pixel) to create relation between query image and data base image. But there can be other attributes of image feature such as texture and other relevant characteristics. By computing image feature, texture and other characteristics we can suggest more relevant result.
   1. weight co-occurrence based integrated color and intensity matrix (WCICIM) for content-based image retrieval. The proposed image indexing technique introduced a feature that utilizes color and texture properties of the image. Hue, saturation and intensity (HSV) color space is the good representation to extract color and texture information [3].
2. Histogram is one of the simplest image features. Despite being invariant to translation and rotation about viewing axis, lack of inclusion of spatial information is its major drawback. Many totally dissimilar images may have similar histograms as spatial information of pixels is not reflected in the histograms.
   1. Solution: Histogram intersection based method for comparing model and image histograms was proposed in for object identification [1]. Histogram refinement based on color coherence vectors was proposed [2]. The technique considers spatial information and classifies pixels of histogram buckets as coherent if they belong to a small region and incoherent otherwise. Though being computationally expensive, the technique improves performance of histogram based matching.

References:

[1] M. Swain and D. Ballard, “Color indexing”, International Journal of Computer Vision, 7(1), 1991, pp. 11–32

[2] G. Pass and R. Zabih, “Histogram Refinement for Content Based Image Retrieval”, 3rd IEEE Workshop on Applications of Computer Vision, WACV , 1996 , pp. 96- 10.

[3] Megha Agarwal, R.P. Maheshwari ,“Weight Co-occurrence based Integrated Color and Intensity Matrix for CBIR”, July. 2011







