Image Retrieval Based on Content by using Clustering Algorithms

Dr. Shahbaa I. Khaleel[1], Dr. Nidhal Hussein[2]

[1,2]Software Engineering Dept., College of Computer Science and Mathematics, Mosul University, Iraq

ABSTRACT

In recent years, very large collections of images have grown rapidly. In parallel with this growth, content based retrieval and querying the indexed collections are required to access visual information. Two of the main components of the visual information are texture and graylevel. In this research, a content based image retrieval system by using clustering algorithms is presented that computes texture and graylevel similarity among images. At the first time, we used Grayscale Histogram and Gray Level Distribution Moments, and then an optimal set of five second order texture statistics is extracted from the Spatial Gray Level Dependency Matrix of each image to obtain a low vector dimensionally for efficiency in computation. Instead of using each feature alone, we combined all of these features by a weighted sum of the similarities provided by each of the features; and this is called Integration of Features. Though the Integration of Features is more effective than using each feature alone. The system is also developed by using K-means clustering algorithm to group the images in the database into clusters of images with similar color content. Thus, at the retrieval time, the query image is not compared with all the images in the database, but only with a small subset. The k-means are also improved by using a better similarity distance that found in Modified1 K-means algorithm; this algorithm is also developed by generating the initial clusters center, when used Modified2 Kmeans. This algorithm is an efficient algorithm and produce efficient retrieval over all feature extraction methods.

Keyword: Image Retrieval, feature extraction, grayscal Histogram, distribution moments, co-occurrence matrix, clustering algorithm.

HOW TO CITE THIS ARTICLE

Dr. Shahbaa I. Khaleel, Dr. Nidhal Hussein, “Image Retrieval Based on Content by using Clustering Algorithms”, International Journal of Enhanced Research in Science, Technology & Engineering, ISSN: 2319-7463, Vol. 8 Issue 7, July-2019.

1-INTRODUCTION

Content Based Image Retrieval CBIR is a set of techniques for retrieving semantically-relevant images from an image database based on automatically-derived image features. The main goal of CBIR is efficiency during image indexing and retrieval, thereby reducing the need for human intervention in the indexing process. The computer must be able to retrieve images from a database without any human assumption on specific domain (such as texture vs. non texture).One of the main tasks for CBIR systems is similarity comparison, extracting feature of every image based on its pixel values and defining rules for comparing images. These features become the image representation for measuring similarity with other images in the database. Images are compared by calculating the difference of its feature components to other image descriptors[1]. Image categorization classifies images into semantic databases that are manually precategorized. In the same semantic databases, images may have large variations with dissimilar visual descriptions (e.g. images of persons, images of industries etc.). In addition images from different semantic databases might share a common background (some flowers and sunset have similar colors).In [2], the authors distinguish three type of feature vectors for image description: pixel level features, region level features, and tile level features. Pixel level features store spectral and textural information about each pixel of the image. For example, the fraction of the endmembers, such as concrete or water, can describe the content of the pixels. Region level features describe groups of pixels. Following the segmentation process, each region is described by its boundary and a number of attributes, which present information about the content of the region in terms of the endmembers and texture, shape, size, fractal scale etc [2]. Tile level for image features present information about whole images using texture, percentages of endmembers, fractal scale and others [3].

2-FEATURE EXTRACTION FROM IMAGES

.A variety of image features, such as color, shape, and texture, have been used. In a typical image retrieval model, an image is represented as a feature vector in a n-dimensional vector space as follows: $I = (f_1, f_2, f_3, ..., f_n)$ Where f_i is an element of the image feature vector[4].

The feature extraction process aims to describe each image in the database in terms of low level features. These low level features, known as descriptors, are used to provide similarity measures between different images. Descriptors are typically smaller in size compared to the original image. In this research, used gray-scale histogram, gray-level distribution moments, and gray-level co-occurrence matrices for gray images. Figure (1) shows the feature extraction process.

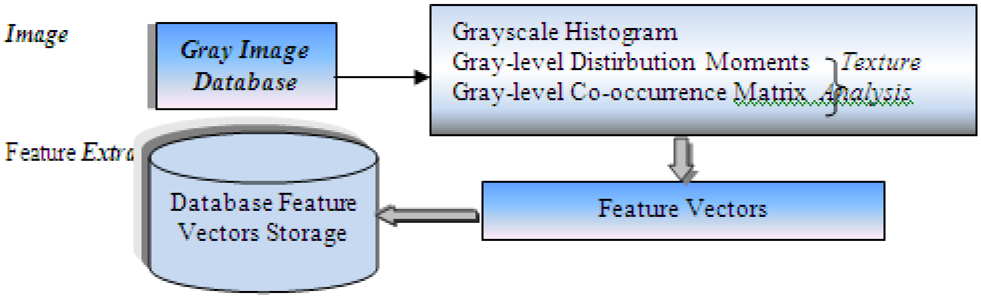


Image
Gray Image Database
Grayscale Histogram
Gray-level Distirbution Moments
Gray-level Co-occurrence Matrix
Texture Analysis
Feature Extra
Database Feature Vectors Storage
Feature Vectors

Figure 1: Gray-level Image

2.1-Grayscale Histogram

A typical gray image downloaded from the Internet normally has only one channel [5]. The normalized grayscale histogram extracted from an image is a 256 dimensional vector that is contained in the histogram space S_H (represented by a unit hypercube), i.e., $H = (h_0, h_1, ..., h_{255})^t, h_i \geq 0 \; , \; 0 \leq i \leq 255$ Where H represents the histogram, $h_i = \frac{B_i}{B_T}$ represents the normalized value of the i^{th} grayscale, B_i represents the number of pixels corresponding to the i^{th} grayscale, B_T is the total number of pixels in an image, and $\sum_{i=0}^{255} h_i = 1$. A histogram is extracted from each image of the database. We treat H as the feature vector [6].

2.2-Graylevel Distribution Moments

One of the simple ways to extract statistical features in an image is to use the first-order probability distribution of the amplitude of the quantized image. They are generally easy to compute. The first order histogram estimate of $p(b)$ is simply : $p(b) = \frac{N(b)}{M}$ Where b is a gray level in an image, M represents the total number of pixels in an image, and $N(b)$ is the number of pixels of gray value b in the image where $0 \leq b \leq L-1$ .Now the following measures have been extracted by using first order probability distribution.

Mean: $S_M = \bar{b} = \sum_{b=0}^{L-1} b \quad p(b)$ Standard deviation: $S_D = \sigma_b = \left[\sum_{b=0}^{L-1} (b - \bar{b})^2 \quad p(b) \right]^{1/2}$

Skew-ness: $S_S = \frac{1}{\sigma_b^3} \sum_{b=0}^{L-1} (b - \bar{b})^3 \quad p(b)$ Energy: $S_N = \sum_{b=0}^{L-1} [p(b)]^2$ Entropy: $S_E = -\sum_{b=0}^{L-1} p(b) \quad \log_2 \{p(b)\}$

The first two features are the mean and standard deviation of pixel intensities within the image[7]. In order to get information on the shape of the distribution of intensity values within image the skewness are determined. The skewness characterizes the degree of a symmetry of the intensity distribution around the mean intensity. If skewness is negative, the data spread out more to the left of the mean than to the right. If skewness is positive, the data are spread out more to the right[7,8]. The energy and entropy are also determined. The energy is useful to examine the power content (repeated transitions) in a certain frequency band. Entropy is a common concept in many fields, mainly in signal processing. [7]

2.3-Graylevel Co-Occurrence Matrix Method

GLCM method is a second order statistical measure, which is based on the spatial gray level dependence. It is a popular statistical technique for extracting textural features from different types of images. It has been used in the analysis and classification of various types of images.

2.3.1-Graylevel Co-Oocurrence Matrices (Glcm) Computation

To extract location-based statistical values, it is necessary to devise a means of describing the location of each pixel, and its relative position to pixels of a certain intensity more accurately. Though known under various names, GLCM is a matrix comparing the intensities of all pixels[9].GLCM has the same size as the number of gray levels in an application. In a case where there exist 64 distinct gray levels, GLCM shall be 64*64 matrix.In addition to the GLCM matrix, a position operator P needs to be defined. The operator P is now passed over the image. For each image pixel, the position operator needs to be evaluated. If the pixel intensity is i and the pixel intensity to which the operator points is j, the matrix element c_{ij} of GLCM is a function of P and P is a function of distance d and angle θ, where the angle could be one specific direction, or a set of directions [9]. The GLCM $P_{\theta,d}(i, j)$ is defined as follows:

$$P_{0^o,d}(i, j) = \#\{((k,l),(m,n)) \in (M \times N) \times (M \times N) : k - m = 0, |l - n| = d, f(k,l) = i, f(m,n) = j\}$$

$$P_{45^o,d}(i, j) = \#\left\{\begin{matrix} ((k, l), (m, n)) \in (M \times N) \times (M \times N) : (k - m = d, l - n = -d) or (k - m = -d, l - n = d), \\ f(k,l) = i, f(m,n) = j \end{matrix}\right\}$$

$$P_{90^o,d}(i, j) = \#\{((k, l), (m, n)) \in (M \times N) \times (M \times N) : |k - m| = d, l - n = 0, f(k,l) = i, f(m,n) = j\}$$

$$P_{135^o,d}(i, j) = \#\left\{\begin{matrix} ((k, l), (m, n)) \in (M \times N) \times (M \times N) : (k - m = d, l - n = d) or (k - m = -d, l - n = -d), \\ f(k,l) = i, f(m,n) = j \end{matrix}\right\}$$

Where d =1 distance between gray value i and j of an image, $i, j = 0 \ldots 255$ number of possible gray-levels, # denotes the number of elements in the set, $k, m = 1 \ldots M$ which represent image width, and $l, n = 1 \ldots N$ represent image height.

2.3.2-Graylevel Co-Occurrence Features

Texture classification can be based on criteria (features) derived from the co-occurrence matrices. In order to use the information contained in the GLCM, used the five features that are used most frequently: energy, entropy, correlation, inertia, and local homogeneity. To simplify and reduce the dimension of a feature vector, quantized the gray levels into 16 steps (textural properties are preserved by this operation), then compute the five features. The compact use of these statistical functions provide efficiency in terms of computational complexity. Thus the following five statistical functions are extracted in the system:[7,9]

Energy: It is also called Angular Second Moment, Homogeneity, and Uniformity. It is a measure of the homogeneity of an image and has the following characteristics: when all the matrix elements are almost equal (i.e., when gray level intensities are very close to each other), the value of the energy is small. When the GLCM is more irregular the value of energy is high.

$$Energy = \sum_i \sum_j P^2_{\theta,d}(i, j)$$

where θ represent the angle and d represent the distance between gray value i and j of an image.

Entropy: It is the opposite of energy; thus it has a lower value when the GLCM is irregular. It has its highest peak when the GLCM is uniform.

$$Entropy = -\sum_{i}\sum_{j} P_{\theta,d}(i,j)\ \log\ P_{\theta,d}(i,j)$$

Contrast: It is also called inertia and measures the difference moment of the GLCM. The value will be high if the image has high local variation.

$$Contrast = \sum_{i}\sum_{j} (i-j)^{2}\ P_{\theta,d}(i,j)$$

Inverse Difference Moment: It is also called local homogeneity and it is the opposite of contrast. If the GLCM has high values at the diagonal, the value of the function is high. The value is also high when similar gray levels are next to each other.

$$Inverse\ \ Difference\ \ Moment = \sum_{i}\sum_{j} \frac{P_{\theta,d}(i,j)}{|i-j|^{2}}\ , i \neq j$$

Correlation: The correlation measures the linear dependency of the gray level values in the GLCM. A high or a low correlation value leads to no immediate conclusion about the image.

$$Correlatio\ \ n = \frac{\sum_{i}\sum_{j} (ij)\ P(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$$

Where means and standard deviations are defined as: $\mu_x = \sum_{i} i \sum_{j} P(i,j)$ $\mu_y = \sum_{j} j \sum_{i} P(i,j)$ $\sigma_x = \sum_{i} (i-\mu_x)^2 \sum_{j} P(i,j)$

$$\sigma_y = \sum_{j} (j-\mu_y)^2 \sum_{i} P(i,j)$$

3-THE NORMALIZATION AND INTEGRATION OF FEATURES (IGM) METHOD

When each feature extraction method was applied alone and the similarity between the query and all images in the database was computed, the results were not effective; therefore here used the (IGM) method that combine the three feature extractions. Figure (2) shows the outline of IGM method. Different features can generate different ranges of values of similarity; a normalization method should be applied to each similarity computation. Here normalize each similarity by the min/max normalization (linear scaling) method as follows:

$$N(I,I') = \frac{D(I,I') - \min\ (D(I,I'))}{\max\ (D(I,I')) - \min\ (D(I,I'))}$$

where $D(I,I')$ represent the difference between the query image and database image.

After normalizing similarity, the total similarity between the query and the image in the data collection is calculated via a weighted sum of the similarities provided by each of the features. The equation for combining the similarities is defined as follows: $D_{combine}(I,I') = W_1 N_{gray-hist}(I,I') + W_2 N_{first-order}(I,I') + W_3 N_{glcm}(I,I')$

here $D_{combine}(I,I')$ is the weighted sum of similarities; $N_{gray-hist}(I,I')$ is the normalized similarity of grayscale histogram; $N_{first-order}(I,I')$ represents the Gray Level Distribution Moments Features; and $N_{glcm}(I,I')$ represents the GLCM features. W_1, W_2, and W_3 are weighting factors to adjust the relative importance of image features. Here choose W_1=1.0, W_2 =0.5, and W_3= 0.02 for our experiments in this research. Combining features with these weights is more effective than using each feature alone.

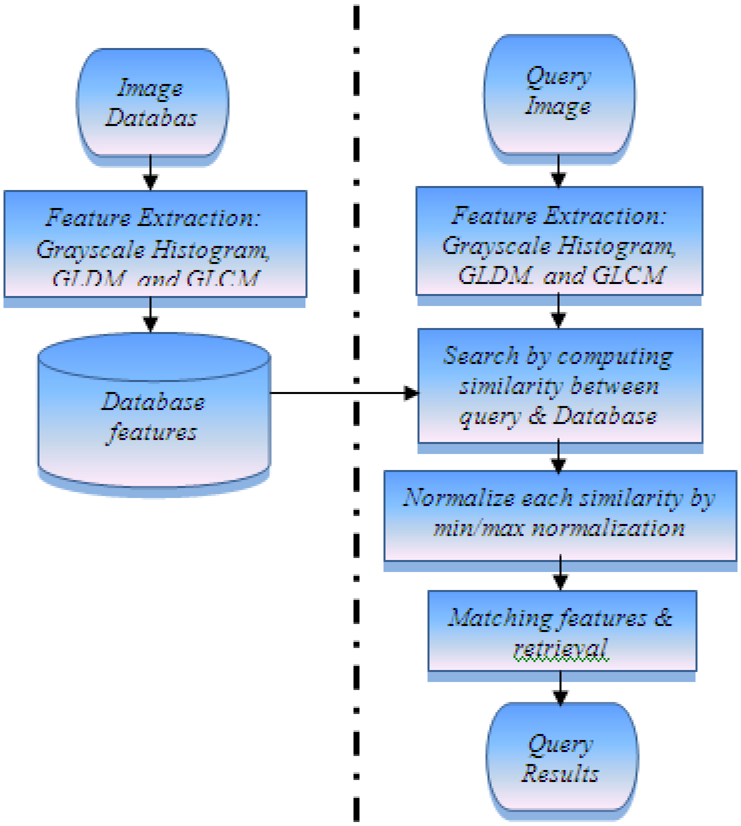


Image Databas
Feature Extraction: Grayscale Histogram, GLDM and GLCM
Database features
Query Image
Feature Extraction: Grayscale Histogram, GLDM and GLCM
Search by computing similarity between query & Database
Normalize each similarity by min/max normalization
Matching features & retrieval
Query Results

Figure 2: Feature Extraction of all Images in Database

4-CLUSTERING

Clustering is an unsupervised learning problem, which tries to group a set of points into clusters such that points in the same cluster are more similar to each other than points in differents clusters, under a particular similarity metric [10].The goal of clustering is to identifying the clusters, which can be considered as classes[11]. Clustering can be used to produce an effective image index as follows: After clustering, each cluster is represented by its centroid or sometimes a single representative data item (i.e. the image lable for that cluster) and, instead of the original data items, the query point is compared to the centroids or the cluster representatives. The best cluster or clusters, according to the used similarity measure, are then selected and the data items belonging to those clusters are retrieved also according to the used similarity measure [12].

The image clustering process aims to decrease the number of image (or feature) vectors compared with the query image. The query is compared to the centroids only; the best clusters are then selected and the image belonging to this cluster are retrieved. Figure (3) shows the clustering algorithms used in this research.

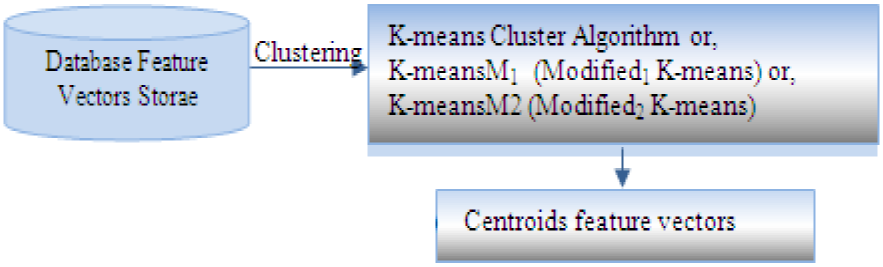

Database Feature Vectors Storae
Clustering
K-means Cluster Algorithm or,
K-meansM1 (Modified1 K-means) or,
K-meansM2 (Modified2 K-means)
Centroids feature vectors

Figure 3: Clustering Algorithms

4.1- K-Means Clustering Algorithm

The k-means algorithm is the most frequently used clustering algorithm due to its simplicity and efficiency. K-means is a partitional clustering algorithm. It performs iterative relocation to partition a dataset into k cluster [10]; and it is based on the minimization of a performance index which is defined as the sum of the squared distances from all points in a cluster domain to the cluster center. This algorithm consists of the following steps: [10,13]

Step1: Choosing K initial cluster centers $z_1(1), z_2(1),..., z_K(1)$. These are arbitrary and are usually selected as the first K samples of the given sample set.

Step2: Distributing the samples $\{x\}$ at the k^{th} iterative step among the K cluster domains, using the relation: $x \in S_j(k) \quad if \quad \|x - z_j(k)\| < \|x - z_i(k)\|$ for all $i = 1,2,..., K, i \neq j,$ where $S_j(k)$ denotes the set of samples whose cluster is $z_j(k)$.

Setp3: Computing the new cluster centers $z_j(k+1), j = 1,2,..., K$, such that the sum of the squared distances from all points in $S_j(k)$ to the new cluster center is minimized. In other words, the new cluster center $z_j(k+1)$ is computed so that the performance index: $J_j = \sum_{x \in S_j(k)} \|x - z_j(k+1)\|^2 \quad , j = 1,2,..., K$ is minimized. The $z_j(k+1)$ which minimizes this performance index is simply the sample mean of $S_j(k)$. Therefore, the new cluster center is given by: $z_j(k+1) = \frac{1}{N_j} \sum_{x \in S_j(k)} x \quad , i = 1,2,..., K$ where N_j is the number of samples in $S_j(k)$. The name "K-means" is obviously derived from the manner in which clusters are sequentially updated.

Sept4: If $z_j(k+1) = z_j(k)$ for $j = 1,2,..., K$, the algorithm has converged and the procedure is terminated. Otherwise one should go to step2.

The behavior of the k-means algorithm is influenced by the number of cluster centers specified, the choice of initial cluster centers, the order in which the samples are taken, and, of course, the geometrical properties of the data. Although no general proof of convergence exists for this algorithm, it can be expected to yield acceptable results when the data exhibit characteristic pockets which are relatively far from each other. In most practical cases the application of this algorithm will require experimenting with various values of K as well as different choices of starting configurations [13].

4.2-Modified$_1$ K-Means Clustering Algorithm

The performance of k-means algorithm is not only dependent on the type of data being analyzed, but is also strongly influenced by the chosen measure of pattern similarity i.e. the measure used for identifying clusters in the data. In the preceding algorithm (i.e. k-means clustering algorithm) we considered the Euclidean distance for comparing two feature vectors. Here improved the k-means by using L_1 norm distance metrics instead of Euclidean distance; in practice, L_1 norm performs better than Euclidean distance since it is more robust and computationally efficient.[14]This algorithm consists of the following steps:

Step1: Choosing K initial cluster center $\mu_1, \mu_2,..., \mu_K$. These are selected as the first K samples of the given sample set.

Step2: Classifying each feature f to the cluster p_s with the smallest distance: $p_s = \arg_{1 \le j \le k} \min \quad D(f, \mu_j)$ this D is a function to measure the distance between two feature vectors and defined as : $D(f, f') = \frac{1}{z(f, f')} \left[\sum_{i=1}^{n} |f(i) - f'(i)| \right]$ where $z(f, f') = \sum_{i=1}^{n} f(i) + \sum_{i=1}^{n} f'(i)$ which is a normalizing function, and n represents number of features in feature vector, where f' is cluster center.

Step3: Based on the classification, update cluster centroids as: $\mu_j = \frac{1}{n_j} \sum_{i=1}^{n_j} f_i^{(j)}$ where n_j is the number of images in cluster j, and $f_i^{(j)}$ is the i^{th} feature vector in cluster j.

Step4: If any cluster centroid changes the value in step3 one should go to step2, otherwise one should stop.

4.3-Modified$_2$ K-Means Clustering Algorithm

In the k-means, the initial cluster assignment is random; different runs of the k-means clustering algorithm may not give the same final clustering solution; or when selected as the first k samples of the sample set the same as modified$_1$ k-means, these two states may not give the good solution. To deal with this, we need to get good starting points for the initial cluster assignment. This leads to develop a modified$_1$ k-means clustering algorithm where an additional step is used to provide the initial cluster centers and L$_1$ norm distance metric when computing the distance between two feature vectors[14]. This algorithm is implemented as:

Step1: The initial centroids are selected in the following way:

1. Given v d -dimensional feature vectors, divide the d dimensions to $p = \frac{d}{K}$. these subspaces are indexed by $[1,2,3,\ldots, p],[p+1, p+2,\ldots, 2p],\ldots, [(k-1)p+1,(k-1)p+2,(k-1)p+3,\ldots, kp]$.
2. In each subspace j of $[(j-1)p+1,\ldots, jp]$, associate a value $f_i^{\,j}$ for each feature vector f_i by : $f_i^{\,j} = \sum_{d=(j-1)p}^{jp} f_i(d)$
3. Choose the initial cluster centroids $\mu_1, \mu_2,\ldots, \mu_K$ by $\mu_j = \arg_{f_i} \min_{1<i<v} f_i^{\,j}$

Step2: Classify each feature f to the cluster p_s with smallest distance. $p_s = \arg_{1\le j\le K} \min D(f, \mu_j)$This D is a function to measure the distance between two feature vectors and defined as: $D(f, f') = \frac{1}{z(f, f')}\sum_{i=1}^{v} \left| f(i) - f'(i)\right|$where $z(f, f') = \sum_{i=1}^{v} f(i) + \sum_{i=1}^{v} f'(i)$ which is a normalizing function.

Step3: Based on the classification, update cluster centroids as: $\mu_j = \frac{1}{v_j}\sum_{i=1}^{v_j} f_i^{(j)}$ where v is the number of images in cluster j , and is the i^{th} feature vector in cluster j .

Step4: If any cluster centroid changes the value in step3, go to step2, otherwise stop.

In this research, we based on Modified$_2$k-means algorithm to cluster all the images in the database into classes.

5-EXPERIMENTAL AND RESULTS

Here, we present a technique for image retrieval based on graylevel from large databases. The goal is to group similar images into clusters and to compute the cluster centers, so that during retrieval, the query image need not be compared exhaustively with all the images in the database. To retrieve similar images for a given query, the query image is initially compared with all the cluster centers. Then a subset of clusters that have the largest similarity to the query image is chosen and all the images in these clusters are compared with the query image.

The Image Retrieval Based on ClusteringIRBC system has a three-step approach to retrieve images from the databases. The first step is *indexing*: for each image in a database, feature vector is computed and stored in feature space. The second step is *clustering*: the images in the database are grouped into clusters of images with similar graylevel content using clustering algorithm. The third step is *searching*: given a query by a user, its feature vector is computed and the system retrieves images having feature vectors with a small subset of clusters that best match the query feature vector. Figure (4) shows the steps of image retrieval based on clustering.

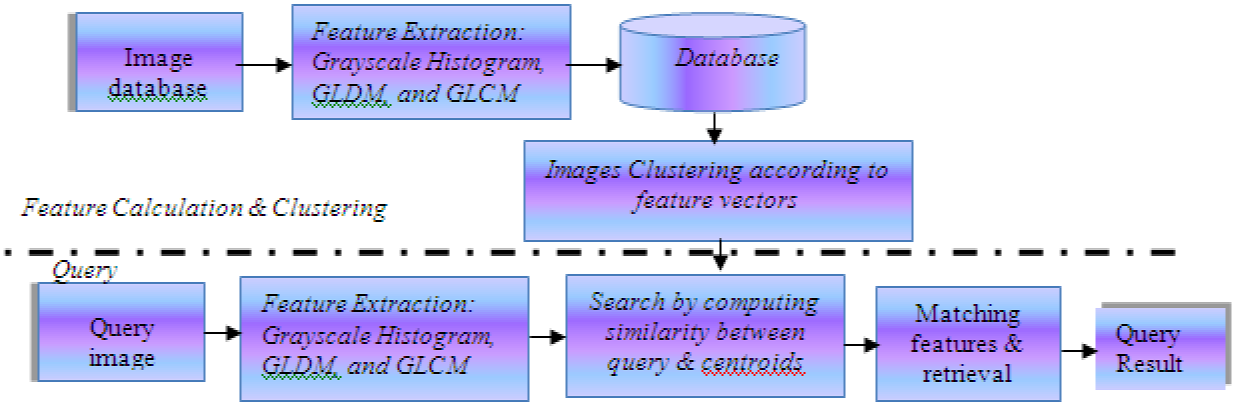


Image database
Feature Extraction: Grayscale Histogram, GLDM, and GLCM
Database
Images Clustering according to feature vectors
Feature Calculation & Clustering
Query
Query image
Feature Extraction: Grayscale Histogram, GLDM, and GLCM
Search by computing similarity between query & centroids
Matching features & retrieval
Query Result

Figure 4: The steps of image retrieval based on clustering

In this research used gray level image and applied three types of feature extraction Grayscale Histogram, Gray level Distribution Moments, Gray level Co-occurrence matrix, and then employed the three clustering method k-means clustering algorithm, Modified$_1$ K-means Clustering Algorithm (K-meansM1), Modified$_2$ K-means Clustering Algorithm(K-meansM2) to decrease the comparison of query.

5.1-Grayscale Histogram Method

Initially used Grayscale Histogram to extract features of gray image, and to retrieve similar images. Here, used k-means, k-meansM$_1$, and k-meansM2 to improve the retrieval images. The results show that the retrieval effectiveness of the method using the k-meansM2 method is better than other methods. This can be clearly noticed in table (1). Rel-Ret means number of relevant retrieved images and Rel-Num, Ret-Num means relevant number of query in database, retrieved number of images from database respectively. GHist refer to Grayscale Histogram method. The result of Grayscale Histogram of query image (49) is 11 relevant image from total retrieved of images 35; while 1 relevant image from 11 image in GHist with k-means clustering algorithm; and 4 relevant image from 6 in GHist with k-meansM1; but 10 relevant image from 16 in GHist with k-meansM2 and the recall is 0.666667 and precision is 0.625000; the final algorithm is the best because both recall and precision are close to 1, (i.e., the ratio of relevant-retrieved from the relevant and relevant-retrieved from the total retrieved is better than the previous algorithms).

Table 1: The Effectiveness of K-meansM2 in Grayscale Histogram Method

Images	Rel-Ret	Rel-Num	Ret-Num	Recall	Precision	Method
Monkey	11	15	35	0.733333	0.314286	GHist
	1	15	11	0.066667	0.090909	GHist-kmeans
	4	15	6	0.266667	0.666667	GHist-kmeansM$_1$
	10	15	16	0.666667	0.625000	GHist-kmeansM2
Rabit	2	11	30	0.181818	0.066667	Ghist
	1	11	20	0.090909	0.050000	GHist-kmeans
	1	11	16	0.090909	0.062500	GHist-kmeansM$_1$
	2	11	8	0.181818	0.250000	GHist-kmeansM2
Livestock	12	14	46	0.857143	0.260870	GHist
	9	14	16	0.642857	0.562500	GHist-kmeans
	10	14	19	0.714286	0.526316	GHist-kmeansM$_1$
	10	14	18	0.714286	0.555556	GHist-kmeansM2

5.2-Graylevel Distribution Moments Method

One of the simple ways to extract features in a gray image is to use gray level distribution moments. The measures that have been extracted by using this method are Mean, Standard deviation, Skewness, Energy and Entropy. This method also proved that k-meansM2 is better than other methods (i.e., k-means, k-meansM$_1$). Some results of this method are shown in table (2). GLDM means Graylevel Distribution Moments method.

Table 2: The Difference between Clustering Algorithms in GLDM Method

Images	Rel-Ret	Rel-Num	Ret-Num	Recall	Precision	Method
Balloon	18	25	67	0.720000	0.268657	GLDM
	3	25	8	0.120000	0.375000	GLDM-kmeans
	10	25	30	0.400000	0.333333	GLDM-kmeansM_1
	10	25	28	0.400000	0.357143	GLDM-kmeansM2
Monkey	10	15	63	0.666667	0.158730	GLDM
	6	15	14	0.400000	0.428571	GLDM-kmeans
	6	15	14	0.400000	0.428571	GLDM-kmeansM_1
	10	15	20	0.666667	0.500000	GLDM-kmeansM2
Livestock	14	14	59	1.000000	0.237288	GLDM
	12	14	23	0.857143	0.521739	GLDM-kmeans
	8	14	8	0.571429	1.000000	GLDM-kmeansM_1
	10	14	11	0.714286	0.909091	GLDM-kmeansM2
Tiger	13	25	61	0.520000	0.213115	GLDM
	4	25	9	0.160000	0.444444	GLDM-kmeans
	5	25	11	0.200000	0.454545	GLDM-kmeansM_1
	8	25	15	0.320000	0.533333	GLDM-kmeansM2

5.3-Graylevel Co-Occurrence Matrix Method

This method can be based on criteria (features) derived from the co-occurrence matrices. In order to use the information contained in the GLCM, we use the five features, Energy, Entropy, Correlation, Inertia and Local Homogeneity. Then we employ the three clustering algorithms to improve this method. Table (3) shows some results of this method and a three clustering algorithms.

Table 3: The Difference between Clustering Algorithms in GLCM Method

Images	Rel-Ret	Rel-Num	Ret-Num	Recall	Precision	Method
Monkey	15	15	106	1.000000	0.141509	GLCM
	7	15	16	0.466667	0.437600	GLCM-kmeans
	7	15	13	0.466667	0.538462	GLCM-kmeansM_1
	6	15	7	0.400000	0.857143	GLCM-kmeansM2
Tiger1	23	25	66	0.920000	0.348485	GLCM
	12	25	14	0.480000	0.857143	GLCM-kmeans
	12	25	14	0.480000	0.857143	GLCM-kmeansM_1
	8	25	8	0.320000	1.000000	GLCM-kmeansM2
Tiger2	20	25	105	0.800000	0.190476	GLCM
	3	25	15	0.120000	0.200000	GLCM-kmeans
	3	25	14	0.120000	0.214286	GLCM-kmeansM_1
	3	25	8	0.120000	0.375000	GLCM-kmeansM2

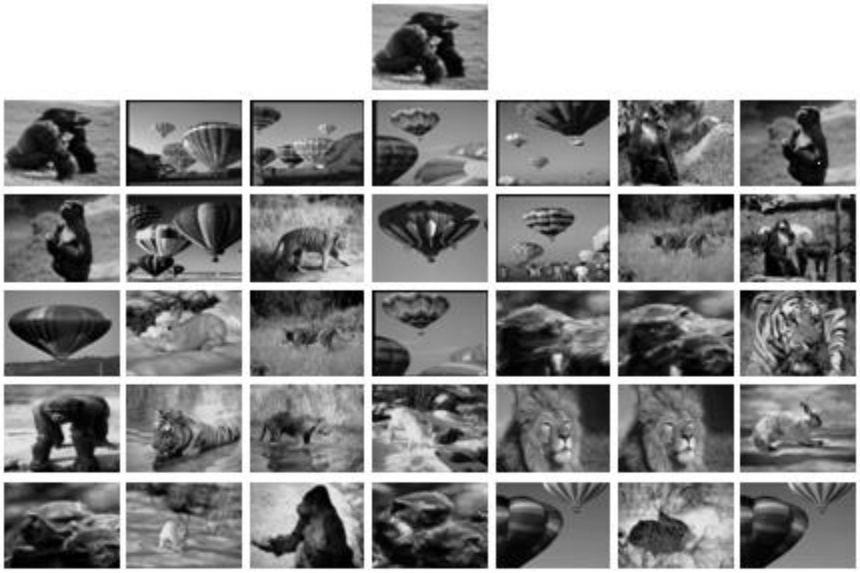
5.4-The IGM Method (Integration of Features)

Here, combined all the previous methods Grayscale Histogram, Gray level Distribution Moments and GLCM in one method by using one equation that includes all the features of gray image. Thus, the total similarity between the query and the image in the data collection is calculated by a weighted sum of the similarities provided by each of the features. Then combining features with one equation is more effective than using each feature alone. Moreover, when the three clustering algorithm is applied, the k-meansM2 is also the best one; see table (4).

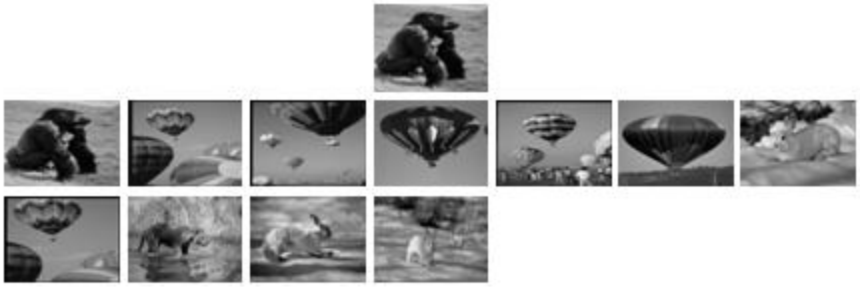
Table 4: The Difference between Clustering Algorithms in IGM Method

Images	Rel-Ret	Rel-Num	Ret-Num	Recall	Precision	Method
Balloon	21	25	63	0.840000	0.333333	IGM
	9	25	14	0.360000	0.642857	IGM-kmeans
	10	25	15	0.400000	0.666667	IGM-kmeansM_1
	9	25	10	0.360000	0.900000	IGM-kmeansM2
Lion	20	21	68	0.952381	0.294118	IGM
	8	21	24	0.380952	0.333333	IGM-kmeans
	9	21	20	0.428571	0.450000	IGM-kmeansM_1
	12	21	23	0.571429	0.521739	IGM-kmeans M2
Monkey	14	15	37	0.933333	0.378378	IGM
	10	15	17	0.666667	0.588235	IGM-kmeans
	7	15	9	0.466667	0.777778	IGM-kmeansM_1
	9	15	9	0.600000	1.000000	IGM-kmeans M2
Livestock	14	14	35	1.000000	0.400000	IGM
	11	14	19	0.785714	0.578947	IGM-kmeans
	13	14	22	0.928571	0.590909	IGM-kmeansM_1
	12	14	12	0.857143	1.000000	IGM-kmeans M2
Tiger1	23	25	81	0.920000	0.283951	IGM
	9	25	26	0.360000	0.346154	IGM-kmeans
	10	25	21	0.400000	0.476190	IGM-kmeansM_1
	14	25	14	0.560000	1.000000	IGM-kmeans M2
Tiger2	23	25	74	0.920000	0.310811	IGM
	12	25	29	0.480000	0.413793	IGM-kmeans
	14	25	25	0.560000	0.560000	IGM-kmeansM_1
	16	25	16	0.640000	1.000000	IGM-kmeans M2
Tiger3	22	25	63	0.880000	0.349206	IGM
	12	25	29	0.480000	0.414793	IGM-kmeans
	13	25	24	0.520000	0.541667	IGM-kmeansM_1
	15	25	15	0.600000	1.000000	IGM-kmeans M2
Tiger4	25	25	70	1.000000	0.357143	IGM
	10	25	27	0.400000	0.370370	IGM-kmeans
	11	25	22	0.440000	0.500000	IGM-kmeansM_1
	15	25	15	0.600000	1.000000	IGM-kmeans M2

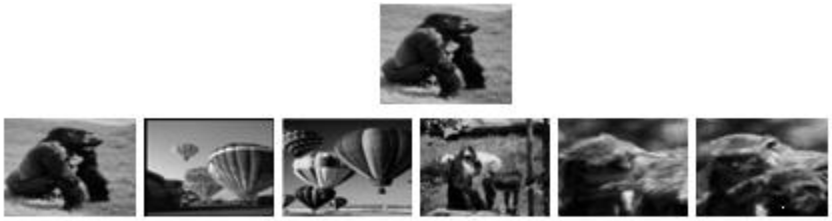
In this work, we applied three types of feature extraction GHist, GLDM, GLCM, and combined the three methods in one method called IGM, and then we employed the three clustering methodsk-means, k-meansM1 and k-meansM2 clustering algorithms to decrease the comparison of query. The results show that the k-meansM2 is better than other algorithms. This can be clearly noticed in the following figures.



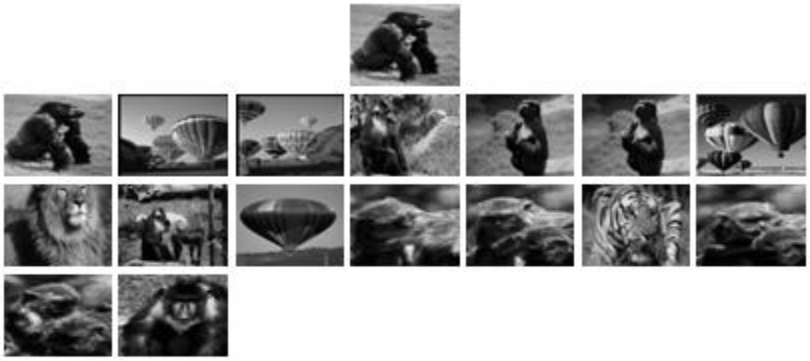
Monkey query, 11 matches from the Top 35 Using GHist Method



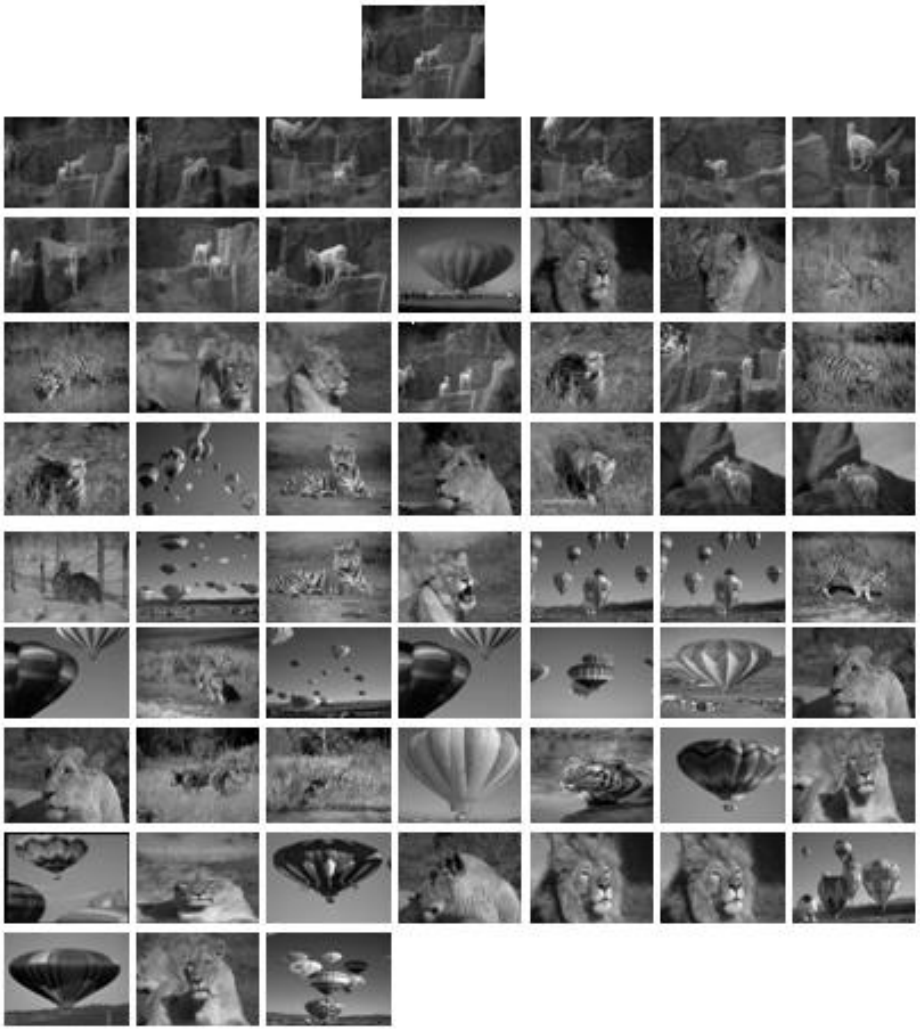
Monkey query, 1 matches from the Top 11 Using GHist Method with K-means Clustering Algorithm



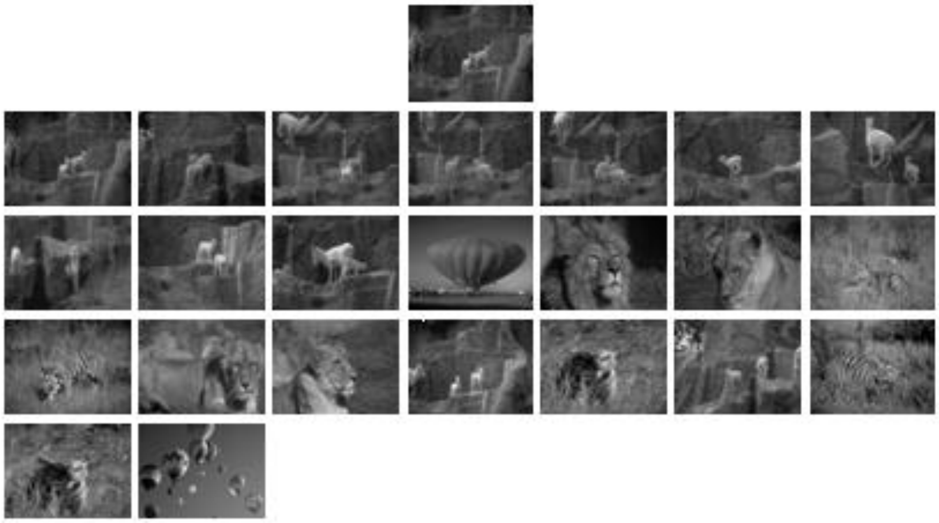
Monkey query, 4 matches from the Top 6 Using GHist Method with K-meansM_1 Clustering Algorithm



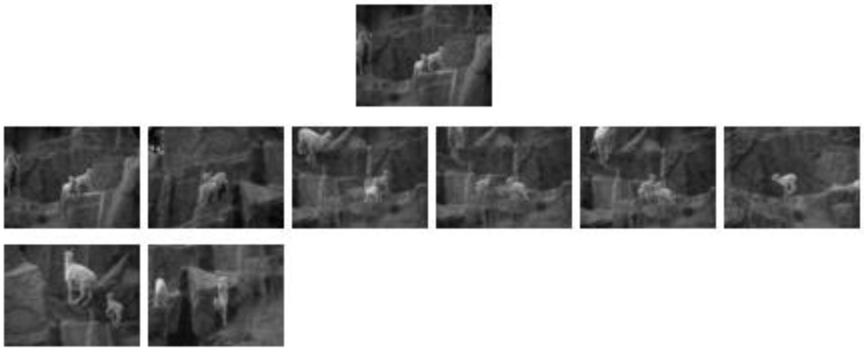
Monkey query, 10 matches from the Top 16 Using GHist Method with K-meansM2 Clustering Algorithm



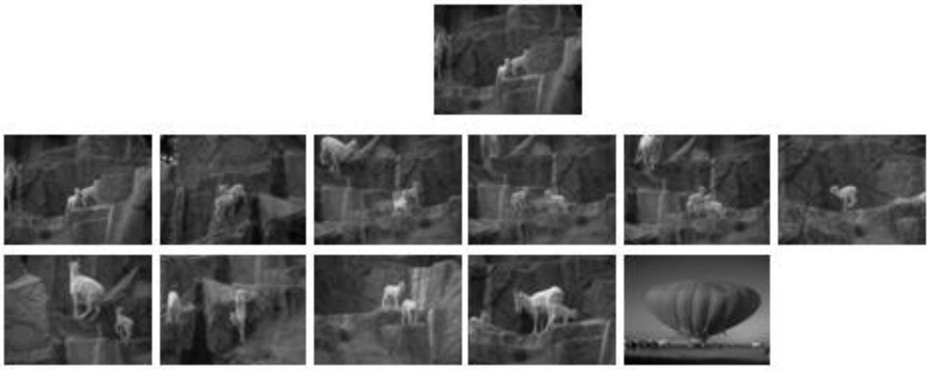
Livestock query, 14 matches from the Top 59 Using GLDM



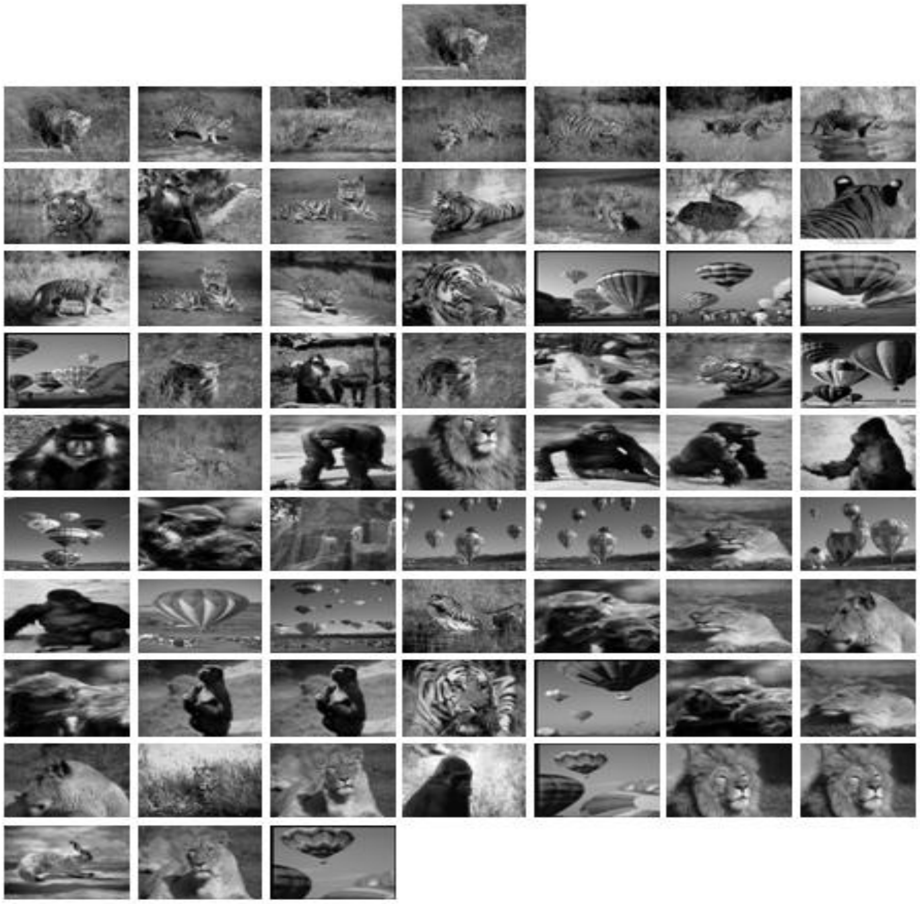
Livestock query, 12 matches from the Top 23 Using GLDM with K-means Clustering Algorithm



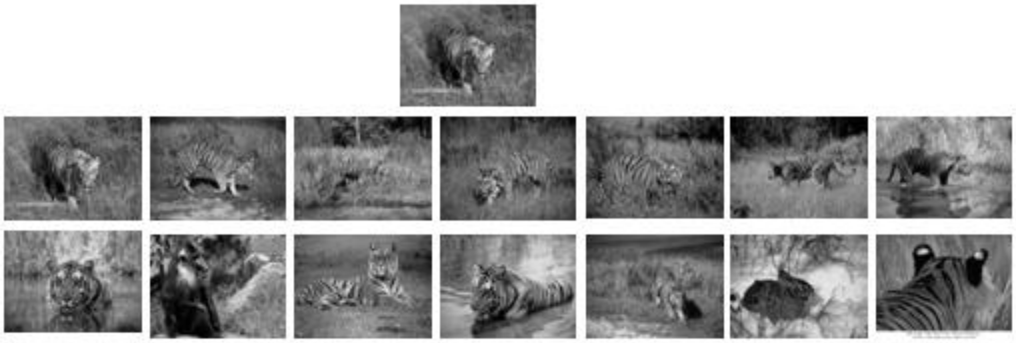
Livestock query, 8 matches from the Top 8 Using GLDM with K-meansM1 Clustering Algorithm



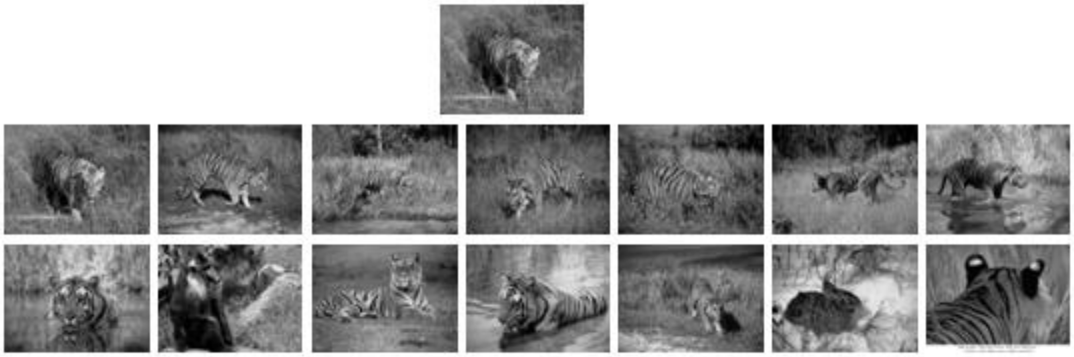
Livestock query, 10 matches from the Top 11 Using GLDM with K-meansM2 Clustering Algorithm



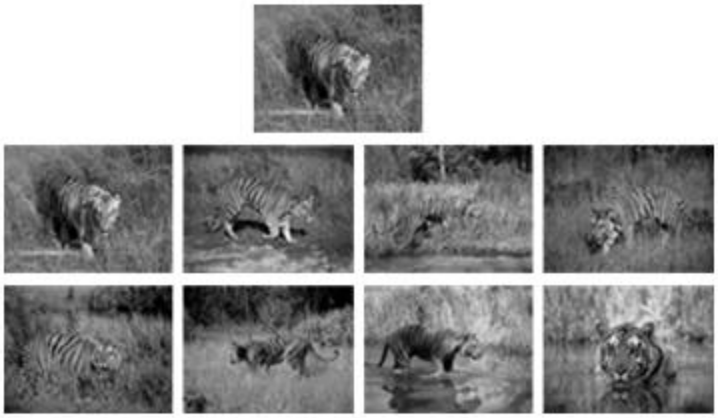
Tiger query, 23 matches from the Top 66 Using GLCM



Tiger query, 12 matches from the Top 14 Using GLCM with K-means Clustering Algorithm



Tiger query, 12 matches from the Top 14 Using GLCM with K-meansM$_1$ Clustering Algorithm



Tiger query, 8 matches from the Top 8 Using GLCM with K-meansM2 Clustering Algorithm

CONCLUSIONS

In this research designed and implemented a content based image retrieval system by using clustering algorithms that evaluates the similarity of each image in its data store to a query image in terms of graylevel and textural characteristics, and returns the images within a desired range of similarity. Here, extract a Grayscale Histogram, Graylevel Distribution Moments, and Graylevel Co-occurrence Matrices of the images. integrated all these features after normalizing similarity; the total similarity between the query and the image in the data collection is calculated via a weighted sum of the similarities provided by each of the features; this method is called IGM. This research introduces Images Retrieval Based on Clustering, an efficient image retrieval scheme, based on a rather simple assumption: semantically similar images tend to be clustered in some feature space. Cluster based retrieval of an image attempts to retrieve semantically coherent image clusters from unsupervised learning of how images of the same semantics are alike. At search time, the query image is not compared with all the images in the database, but only with a small subset. Here we used k-means clustering algorithm and also used Modified1 k-means that improved the k-means by using a better similarity distance to obtain another clustering algorithm called k-meansM_1 and used the developed of this algorithm by generating the initial clusters, and this is called k-meansM2. This algorithm is an efficient algorithm and produce efficient retrieval over all feature extraction methods.

REFERENCES

[1] Aulia E., (2005), Hierarchical Indexing For Region Based Image Retrieval, MSc Thesis, Department of Industrial and Manufacturing Systems Engineering, University of Louisiana State.

[2] Zhang J., Hsu W., Lee M., (2001), Image Mining: Issues, Frameworks and Techniques, In Proc. of the second International Workshop on Multimedia Data Mining (MDM/KDD 2001), San Francisco, CA, USA, pp. 13-20.

[3] Zhang J., Hsu W., Lee M., (2001), An Information-driven Framework for Image Mining, In Proc. of 12th International Conference on Database and Expert System Applications, Munich, pp. 232-242.

[4] Park G., Baek Y., Lee H., (2003), Re-ranking Algorithm Using Post-Retrieval Clustering For Content-Based Image Retrieval, Information Processing and Management.

[5] Tran L.V.,(2003), Efficient Image Retrieval With Statistical Color Descriptors, PhD Thesis, Department of Science and Technology, Linköping University, Sweden.

[6] Iqbal Q., Aggarwal J., (1999), Using Structure In Content-Based Image Retrieval, Signal and Image Processing, pp. 129-133.

[7] Gonzalez R., Woods R., (2002), Digital Image Processing, Prentice-Hall, Inc., USA.

[8] Giudici P., (2003), Applied Data Mining Statistical Methods For Business and Industry, John Wiley & Sons Ltd., The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England.

[9] Konak E., (2002), A content-Based Image Retrieval System For Texture And Color Queries, MSc thesis, Department of Computer Engineering, University of Bilkent.

[10] Basu S., (2003), Semi-Supervised Clustering: Learning With Limited User Feedback, PhD Thesis, University of Texas, Austin.

[11] Aleksandra T., (2003), Genetic Programming in Data Mining : Cellular Approach, MSc Thesis, Institute of Informatics Faculty of Mathematics, Physics and informatics, Comenius University, Slovakia.

[12] Krishnamachari S., Abdel-Mottaleb M., (1999), Hierarchical Clustering Algorithm For Fast Image Retrieval, Conference on Storage and Retrieval For Image and Video Databases, pp. 427-435.

[13] Tou J., Gonzalez R., (1974), Pattern Recognition Principles, Addision-Wesley Publishing Company, USA.

[14] Shailendra S., Prem Narayan A., (2012), Comparison of K-means and Modified K-mean algorithms for Large Data-set , International Journal of Computing, Communications and Networking, Volume 1, No.3, pp. 106-110 .