

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/220595166>

Image Clustering using Color, Texture and Shape Features

Article in *KSII Transactions on Internet and Information Systems* · January 2011

DOI: 10.3837/tis.2011.01.012 · Source: DBLP

CITATIONS

16

READS

2,048

6 authors, including:



Azzam Sleit

University of Jordan

131 PUBLICATIONS 490 CITATIONS

[SEE PROFILE](#)



Abdel Latif Abu Dalhoum

University of Jordan

28 PUBLICATIONS 150 CITATIONS

[SEE PROFILE](#)



Mohammad Qatawneh

University of Jordan

60 PUBLICATIONS 277 CITATIONS

[SEE PROFILE](#)



Ola Karajeh

Jordan University of Science and Technology

2 PUBLICATIONS 16 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Knapsack problem [View project](#)



Call for papers [View project](#)

Image Clustering using Color, Texture and Shape Features

Azzam Sleit, Abdel latif Abu dalhoum, Mohammad Qatawneh, Maryam Al-Sharief, Rawa'a Al-Jabaly and Ola Karajeh

Computer Science Department, King Abdulla II School for Information Technology
P.O. Box 13898

University of Jordan, Amman 11942, Jordan

[e-mail: {azzam.sleit, a.latif, mohd.qat, maryams, rawaj, olak}@ju.edu.jo]

*Corresponding author: Azzam Sleit

*Received October 11, 2010; revised November 8, 2010; accepted December 3, 2010;
published January 31, 2011*

Abstract

Content Based Image Retrieval (CBIR) is an approach for retrieving similar images from an image database based on automatically-derived image features. The quality of a retrieval system depends on the features used to describe image content. In this paper, we propose an image clustering system that takes a database of images as input and clusters them using k-means clustering algorithm taking into consideration color, texture and shape features. Experimental results show that the combination of the three features brings about higher values of accuracy and precision.

Keywords: Content-based image retrieval, HSV color histogram, Gabor filter, Fourier descriptor, k-means clustering

1. Introduction

Search in image data sets is currently a very active research area. The amount of available image data is growing rapidly due to the availability of inexpensive storage devices, digital cameras and video recorders. For this reason, an increasing need to efficiently access this information has developed. Information in visual form differs from traditional textual databases in many important ways. It is highly unstructured, and needs some kind of decoding to determine its semantic content. Several techniques have been proposed, which largely fall into two categories.

Text-based image retrieval - textual descriptions of the images (metadata) are stored in the database. User posts query using either keywords or natural languages.

Content-based image retrieval (CBIR) - images are organized according to their visual content. It is a pixel-based approach that allows user to post query by example or image characteristics such as color, shape, or texture.

Text-based image retrieval methods suffer from the problem that when extracting meaning from images in database applications, it is sometimes difficult to derive an image which captures the meaning in the context that the database user desires. In CBIR, the visual contents of a query image are extracted which are then compared with visual contents of other images in the database in order to detect similarity between them. The most relevant images are returned as a result of the image query. The visual contents of images are represented as sets of features that can be either globally or locally extracted from images. Global features are fetched from the entire image while local features are fetched from a small group of pixels. In order to avoid the tedious task of manually annotating the images, good content-based search solutions for searching and classifying image content need to be developed. Analysis of image content has been worked on since the introduction of digital images, mostly for very specific problems with very controlled image data. There are three main directions of CBIR-related research [1].

Feature extraction - Color, texture, and shape of objects are classified as general characteristics. They are used in the majority of CBIR systems and are convenient for retrieval from heterogeneous image collections. Feature extraction algorithms and similarity measures for image comparison are usually the basis of most CBIR systems. In the framework of this research direction, feature vectors and methods to extract them are suggested. New metrics are introduced on the corresponding vector spaces.

Image indexing - This research direction focuses on the development of multidimensional indexing algorithms to facilitate fast search in large-scale collections of high dimension data. Clustering is one type of image indexing where images are categorized into different groups based on their features, such as shape, color, or texture. An image is included into a feature cluster only if the image contains all the features under the same cluster. Distance metrics for image content can be used as a basis for both classification and search algorithms. Calculating image distance metrics can be done in many different ways and is dependent on several factors. Some distance metrics will, for instance, be limited to Boolean or binary evaluation. Furthermore, different distance metrics can be combined in order to create flexible models for information importance.

Query Answering - This area focuses on the user interface of a CBIR system to help the user formulate queries and refine them on subsequent stages of communication with the system.

While feature extraction algorithms affect the quality of the retrieved answer set, indexing makes search faster. The human-engineered interface of a query answering system helps in retrieving relevant images. Several techniques for content based image retrieval are described in the literature [1][2][3]. In this paper, we propose a CBIR scheme that extracts color, texture, and shape features of images. Then, we group similar images together using k-means clustering. We use the color histogram, Gabor filters, and Fourier transformation for color, texture, and shape feature extraction, respectively. Finally, a feature vector that combines all three features is used to cluster images into different groups. Sections 2, 3, and 4 discuss related work in color, texture and shape feature extraction, respectively. Sections 5 and 6 propose a CBIR k-means cluster based scheme. Section 7 describes experimental results. The paper concludes with section 8.

2. Color Feature Extraction

Color feature is one of the most widely used visual features in image retrieval systems. Color can be easily extracted, and is invariant with respect to image size and orientation. In a color-based image retrieval approach, a user specifies the color of regions in a query picture and the images that partially or exactly satisfy the user-specified query are retrieved. Color feature extraction methods can be classified into two categories: color histograms and statistical methods [4].

The color histograms feature extraction category is the simplest and most commonly used to extract the color feature vector. It indicates the frequency of occurrences of every color in an image, and can be defined as a mass function of the image intensities. It represents the joint probability of the intensity values for the three color channels [4]. The second category is based on statistical methods such as color moments. In [5], Stricker *et al.* proposed color moments based on the color distribution of the image. Assuming that the distribution of color in an image can be considered as probability distribution, if the colors in an image follow a certain probability distribution, the moments of that distribution can be used as a color feature vector. This scheme uses three central moments of image color distribution; namely, the Mean, Standard Deviation and Skewness. They are computed for the three color bands: Red, Green, and Blue (RGB). The feature vector for color contains nine moments; i.e., three moments for each of the three bands.

Our work is based on the HSV color histogram feature extraction. HSV color histogram is a color model that represents, with equal emphasis, the three color variants that characterize color: Hue (H), Saturation (S) and Value of intensity (V). Using Eq. (1), (2), and (3), we can convert from the RGB space to the HSV space [6].

$$H = \cos^{-1} \left\{ \frac{1/2 [(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right\} \quad (1)$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \quad (2)$$

$$V = \frac{1}{3} (R + G + B) \quad (3)$$

Computationally, the color histogram is formed by counting the number of pixels of each color in the image. For image I , the color histogram (CH_I) can be represented as a vector as per Eq. (4).

$$CH_I = \{ch_1, ch_2, ch_3, \dots, ch_n\} \quad (4)$$

ch_i denotes the number of pixels of color i in the image. Typically, images are represented in the RGB color space with some of the most significant bits per color channel.

H , S and V are used to compute each ch_i as shown in the following equations:

$$L(x, y) = \alpha \times H(x, y) + \beta \times S(x, y) + \gamma \times V(x, y) + \theta \quad (5)$$

$$ch_i = F(i) \quad (6)$$

where F represents the frequency of i in L . ch_i is normalized as follows:

$$\text{Normalized } ch_i = ch_i / \sum_{j=0}^N ch_j \quad (7)$$

In our experiment, we use, $\alpha = 4$, $\beta = 2$, $\gamma = 1$, $\theta = 8$ and $N = 36$.

3. Texture Feature Extraction

Texture can be defined as periodically repeated local patterns in an image [7][13]. Texture features extraction was introduced based on the Human Visual System (HVS). It was observed that some cells in human eyes are sensitive to channels of combinations of different orientations and spatial frequencies. Moreover, it was found that the repetition of texture can be characterized by the spatial frequency while the directionality can be characterized by orientation. This fact caused texture analysis methodologies to be invented based on the HVS model [7]. Texture-based image retrieval methods allow users to retrieve the images from a database which are most similar to a sample query by utilizing texture primitives and their displacement vectors for the query image and the database images.

Gabor filters are among the well known efficient methods for texture feature extraction. The filters represent a group of wavelets with each of which capturing energy at different frequency and direction which makes Gabor filters useful in texture analysis. A Gabor function in the spatial domain is represented as a complex modulated sinusoidal [8]. The Gabor function is characterized as follows:

$$f_{mn}(x, y) = a_{mn}(x, y) c_{mn}(x, y) \quad (8)$$

where the Gaussian component is:

$$a_{mn}(x, y) = (1 / 2\pi\sigma_m^2) \exp[(x^2 + y^2) / 2\sigma_m^2] \quad (9)$$

and the sinusoidal component is:

$$c_{mn}(x, y) = \cos(2\pi(U_m x \cos \theta_n + U_m y \sin \theta_n)) \quad (10)$$

We used 4 scales, and 8 orientations, where the orientation bandwidth B_θ is 22.5° and the frequency bandwidth is 1 octave. The highest spatial frequency is:

$$U_I = 0.5 / (1 + \tan(B_\theta / 2)) \quad (11)$$

After calculating f , the Gabor discrete wavelet transform for a specific image I is given as follows:

$$G_{mn}(x, y) = I(x, y) * f_{mn}(x, y) \quad (12)$$

When applying Gabor wavelet transform with a specific frequency and orientation to an image, the energy content is calculated as follows:

$$E(m, n) = \sum_x \sum_y |G_{mn}(x, y)| \quad (13)$$

Finally, the texture feature vector of an image of size $P \times Q$ consists of the Median (μ) and Standard Deviation (SD) of the calculated energy at each scale and orientation:

$$\mu_{mn} = E(m, n) / (P \times Q) \quad (14)$$

$$SD_{mn} = \sqrt{(\sum_x \sum_y |G_{mn}(x, y)| - \mu_{mn}) / (P \times Q)} \quad (15)$$

4. Shape Feature Extraction

The shape feature is one of the most important low level visual features of images. Rui et al. proposed a system that uses a modified geometric hashing technique to retrieve similar shaped images from the image database [1]. It retrieves similar images from the database by using scaled, rotated or mirrored sketched query images. Shape feature extraction methods can be contour-based or region-based. Contour-based feature extraction methods are used when the shape information of an object is contained in its boundary. These methods extract the pixels existing on the boundary of an object. However, region-based feature extraction methods are used when the shape information of an object is contained in the interior of the object. These methods extract shape information from all pixels existing within a specific object [9].

There are many methods that can be used in contour-based feature extraction such as Fourier descriptors, wavelet descriptors, curvature scale space descriptors, shape signatures, etc. The Fourier descriptors-based method is the most widely used contour-based shape feature extraction scheme, where the shape feature in spectral domain is analyzed to reduce noise and variations. Additionally, the Fourier descriptors method is used to represent the shape feature in a compact and computationally light way [9].

The Fourier descriptors-based method starts by computing the boundary pixels using the shape signature function. For the computation of boundary pixels, the edge detector and boundary tracer are used to determine the boundary coordinates. Thus, a set of boundary pixels can be formed and denoted as follows:

$$P = \{(x(t), y(t)) \mid t \in [1, N]\} \quad (16)$$

where N is the number of boundary pixels.

The next step is to compute the shape signature function from the set of boundary pixels P . The most widely used shape signature functions are complex coordinates, curvature function, cumulative angular function and centroid distance. It has been shown that the Fourier descriptors method using centroid distance as a shape signature function outperforms Fourier descriptors methods using other shape signature functions. In the centroid distance method, we compute the centroid of the shape of the set P , denoted by (X_c, Y_c) as follows:

$$X_c = \frac{1}{N} \sum_{t=0}^{N-1} X(t) \quad , \quad Y_c = \frac{1}{N} \sum_{t=0}^{N-1} Y(t) \quad (17)$$

Then, we compute the distance between each pixel in the set P and the centroid point, denoted by $r(t)$ as follows:

$$r(t) = \sqrt{[x(t) - x_c]^2 + [y(t) - y_c]^2}, \quad t = 1, 2, \dots, N \quad (18)$$

Next, the discrete Fourier transform is computed as follows.

$$a_n = \frac{1}{N} \sum_{t=1}^N r(t) \exp\left(\frac{-2j\pi n X t}{N}\right), \quad n = 0, 1, 2, \dots, N-1 \quad (19)$$

where a_n denotes the Fourier transformed coefficients of $r(t)$.

Finally, the Fourier transformed coefficients are standardized by the first Fourier transformed coefficient a_0 . The standardized Fourier transformed coefficients, denoted as b_n are computed as follows.

$$b_n = |a_n / a_0| \quad (20)$$

The Fourier descriptor is used to compute shape features and the shape feature vector FD.

$$FD = \{b_i \mid i \in [0, N/2 - 1] \} \quad (21)$$

5. k-Means Clustering Algorithm

The k-means algorithm is one of the simplest unsupervised learning algorithms used for clustering, where the number of clusters k is previously known [10][11][14]. The following steps describe the k-means clustering algorithm.

Step1: Let k be the number of clusters.

Step2: Randomly initialize the centers of clusters to values which belong to the range of the dataset.

Step3: Use the Euclidean distance to associate each sample in the dataset with one of the k clusters (This depends on the shortest distance between sample and the clusters centers).

Step4: Find the new centers of the k clusters by taking the average of all samples in each cluster.

Step5: If none of the centers has changed STOP
Else go to **Step3**.

6. Proposed CBIR System

In order to achieve high accuracy of the retrieving process, we propose a novel algorithm in which a combination of color, texture, and shape features is used to form the feature vector. We use the k-means clustering algorithm to group images into different categories according to their feature vectors. **Fig. 1** displays the block diagram of the proposed system.

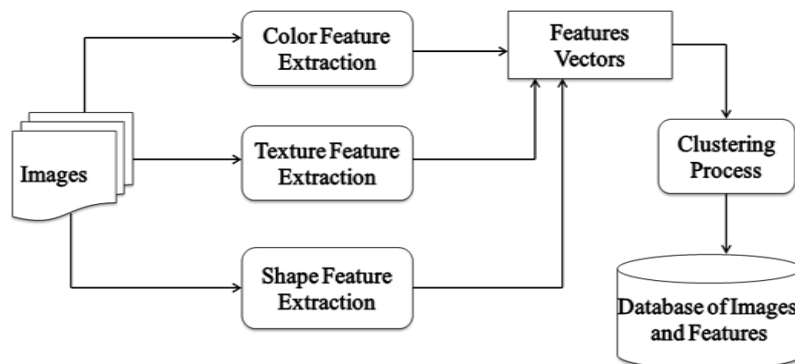


Fig. 1. Proposed System Architecture.

The following steps describe the proposed algorithm.

Step 1: For every image in the database repeat step 2, 3, 4, and 5.

Step 2: Extract the color visual feature using color histogram. Then apply the color histogram on Hue, Saturation and Value to obtain the following color feature vector for each band:

$$CH = \{ch_1, ch_2, ch_3, \dots, ch_n\}$$

Step 3: Extract the texture visual feature using Gabor filter. Compute the mean μ_{mn} and the standard deviation SD_{mn} of the magnitude of the transformed coefficients which can be used to represent the homogenous texture feature of the region. The Gabor measurements are computed for each of the three color components (RGB). The feature vector for m scales and n orientations for each band is given by:

$$F_{\text{Texture}} = (\mu_{00}^i, SD_{00}^i, \mu_{01}^i, SD_{01}^i, \dots, \mu_{mn}^i, SD_{mn}^i)$$

Step 4: Extract the shape visual feature using the Fourier descriptors with brightness. The shape feature vector is extracted for the three color components, and can be denoted as follows:

$$FD = \{b_i \mid i \in [0, N/2 - 1]\}$$

where b_i denotes the Fourier transformed coefficients, N the number of boundary pixels, and i the color band.

Step 5:

5.1: Let the feature vector for each color band be denoted as (RGB) $F = \{F_{\text{Texture}}, F_{\text{Shape}}\}$. For each component (RED, GREEN, BLUE), the extracted feature vectors are:

$$F_R = \{\mu_{00}^R, SD_{00}^R, \mu_{01}^R, SD_{01}^R, \dots, \mu_{mn}^R, SD_{mn}^R, b_0^R, b_1^R, \dots, b_{N/2}^R\}$$

$$F_G = \{\mu_{00}^G, SD_{00}^G, \mu_{01}^G, SD_{01}^G, \dots, \mu_{mn}^G, SD_{mn}^G, b_0^G, b_1^G, \dots, b_{N/2}^G\}$$

$$F_B = \{\mu_{00}^B, SD_{00}^B, \mu_{01}^B, SD_{01}^B, \dots, \mu_{mn}^B, SD_{mn}^B, b_0^B, b_1^B, \dots, b_{N/2}^B\}$$

5.2: Integrate the color feature vector with the averages of the texture feature vector and shape feature vector as follows:

$$V_{\text{Image}} = [CH_{\text{color}} (FR + FG + FB) / 3 \text{ Texture} \\ (FR + FG + FB) / 3 \text{ Shape}]$$

Step 6: Run the following steps of the k-means Clustering algorithm.

6.1: Input the feature vector for each image in the database.

6.2: Randomly initialize the centers of the K clusters centers to feature vectors of images in the database.

6.3: For every image in the database,

6.3.1 Use the Euclidian distance to find the cluster C_i whose center is closest to the feature vector of the image.

6.3.2 Add the image to the cluster C_i

6.3.3 Re-Calculate the new clusters centers by taking the average of all samples in the same cluster.

6.4: If the centers of the clusters have not changed from their previous values STOP

Else GO TO step 6.3

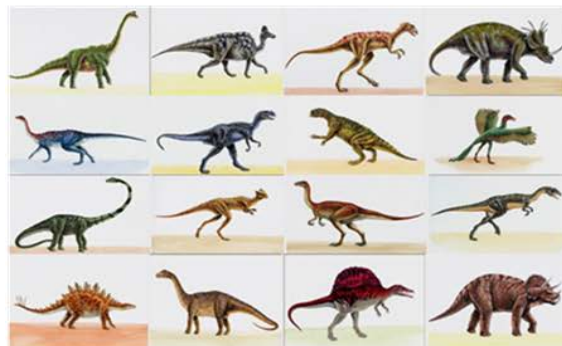
When a test image is to be classified, the system extracts the feature vector as per steps 1-5. Then, the Euclidian distance is computed between the feature vector of the image and the center of each cluster. The images of the cluster with the minimum distance from the test image are considered to be the most similar images to the test image.



Fig. 2. Sample of unclustered data.

7. Experimental Results

Our proposed system combines color, texture and shape feature extraction techniques for content-based image clustering. We conducted experiments using Matlab on an image database that is publicly available from wang.ist.psu.edu/iwang/test1.tar. The images of the database belong to four different groups; namely, Dinosaurs, Flowers, Busses and Elephants as shown in **Fig 2**. The spatial resolution of each image is of size 384×256 pixels. We ran our system on the image data set using the combination of the three features which returned the four clusters shown in **Fig 3**. The generated busses cluster included fourteen busses and two flowers images.

**Cluster1****Cluster2****Cluster3****Cluster4****Fig. 3.** k-means clustering based on color, texture and shape features.

**Cluster1****Cluster2****Cluster3****Cluster4****Fig. 4.** k-means clustering based on shape feature.

**Cluster1****Cluster2****Cluster3****Cluster4****Fig. 5.** k-means clustering based on texture features.

**Cluster1****Cluster2****Cluster3****Cluster4****Fig. 6.** k-means clustering based on color feature.

The flowers and dinosaurs clusters correctly included sixteen flower and dinosaur images, respectively. The elephants cluster included thirteen elephant images but also included three busses images.

In order to evaluate the benefit of combining the three features, we applied the k-means clustering algorithm using a single feature. **Fig 4**, **Fig 5**, and **Fig 6** show the result of generating the four clusters using only a single feature; shape, texture or color, respectively. We assessed our experiments using the recall and precision measures. The recall measure is calculated by dividing the number of relevant images in a cluster by the number of relevant images in the whole database. However, the precision measure is the number of relevant images in a cluster divided by the number of images in the whole cluster.

Table 1 displays the values for recall and precision calculated as a result of using the combination of the three features as well as each of the three features (shape, texture, color) independently. It is clear that combining the three features brought about higher precision and recall values for k-means clustering. The texture feature seems to contribute the most to the accuracy of clustering followed by color and shape.

Table 1. Recall and precision values.

Cluster	Feature	Recal	Precision
Dinosaurs	Color, Texture & Shape	100%	100%
	Color	97%	94%
	Texture	100%	97%
	Shape	94%	94%
Flowers	Color, Texture & Shape	88%	100%
	Color	83%	63%
	Texture	95%	80%
	Shape	40%	84%
Busses	Color, Texture & Shape	81%	86%
	Color	93%	74%
	Texture	75%	60%
	Shape	77%	50%
Elephants	Color, Texture & Shape	100%	85%
	Color	94%	92%
	Texture	61%	88%
	Shape	55%	57%

8. Conclusion

Search in image content uses low-level image features such as shape, color, and texture. A content-based image retrieval system uses information from the content of images for retrieval and helps the user in retrieving database images relevant to the contents of a query

image. During retrieval, one image can be chosen from the database as query, which can also be composed from images in the database using copy and paste. The system retrieves images with similar features.

This article presents a k-means clustering system which utilizes color, texture and shape features to group images. We use the color histogram, Gabor filters, and Fourier transformation for color, texture, and shape feature extraction, respectively. The combination of the three features experimentally demonstrated higher recall and precision values for the generated clusters.

The proposed scheme can be extended for clustering pixel level encrypted images such as medical image database where pixel level encryption is mandatory. The idea is to cluster sensitive images first using the proposed scheme which will then generate a composite feature vector for each cluster, then, scrambling the medical images within a particular cluster. Finally, the composite feature vector from unencrypted images can be used to represent the cluster with encrypted images. As the composite feature vector does not disclose the raw images, it achieves the goal of clustering encrypted images where no other methods have addressed this issue before [15].

References

- [1] Y. Rui, T.S. Huang and S. F. Chang, "Image retrieval: past, present and future," *Journal of Visual Communication and Image Representation*, pp. 1-23, 1997.
- [2] Y. Cai-xiang and Q. Shu-bo, "Image retrieval algorithm based on texture and color feature," *WASE International Conference on Information Engineering*, vol. 1, pp. 125-128, 2009. [Article \(CrossRef Link\)](#).
- [3] P. Androutsos, A. Kushki, K. N. Plataniotis and A. N. Venetsanopoulos, "Aggregation of color and shape features for hybrid query generation in content based visual information retrieval," *Signal Processing*, vol. 85, no. 2, pp. 385–393, 2005. [Article \(CrossRef Link\)](#).
- [4] N.S. Vassilieva, "Content-based image retrieval methods," *Programming and Computer Software*, vol.35, no. 3, pp. 158-180, 2009. [Article \(CrossRef Link\)](#).
- [5] M. Stricker and M. Orengo, "Similarity of color images," in *Proc. of the SPIE Conference on Storage and Retrieval for Image and Video Databases III*, vol. 2420, pp.381-392, 1995.
- [6] Marius Tico, Taneli Hayerinen and Pauli Kuosmanen, "A method of color histogram creation for image retrieval," in *Proc. of Nordic Signal Processing Symposium (NORSIG)*, pp.157-160, 2000.
- [7] Khaled Hammouda and Ed Jernigan, "Texture segmentation using gabor filters," *Thesis, University of Waterloo, Ontario, Canada*, 2008.
- [8] Manish Maheshwari, Sanjay Silakari and Mahesh Motwani, "Image clustering using color and texture," in *Proc. of the 1st International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 403-408, 2009. [Article \(CrossRef Link\)](#).
- [9] Gang Zhang, Z.M. Ma, Qiang Tong, Ying He and Tienan Zhao, "Shape feature extraction using Fourier Descriptors with brightness in content-based medical image retrieval," in *Proc. of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp.71-74, 2008. [Article \(CrossRef Link\)](#).
- [10] Priti Maheswary and Namita Srivastava, "Retrieval of remote sensing images using colour & texture attribute," *International Journal of Computer Science and Information Security*, vol. 4, no. 1, pp. 19-23, 2009.
- [11] P.S. Hiremath and J. Pujari, "Content based image retrieval using color, texture and shape features," in *Proc. of the 15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, pp.780-784, 2007. [Article \(CrossRef Link\)](#).

- [12] M. Kokare, P.K. Biswas and B.N. Chatterji, "Texture image retrieval using rotated wavelet filters," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1240-1249, 2007. [Article \(CrossRef Link\)](#).
- [13] E. Saykol, U. Gudukbay and O. Ulusoy, "A histogram-based approach for object-based query-by-shape-and-color in image and video databases," *Image and Vision Computing*, vol. 23, no. 13, pp. 1170–1180, 2005. [Article \(CrossRef Link\)](#).
- [14] W. Raffe, J.Hu, F. Zambetta, and K. Xi, "A dual-layer clustering scheme for real-time identification of plagiarized massive multiplayer games (MMG) assets," in *Proc. of the 5th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Taiwan, June 2010. [Article \(CrossRef Link\)](#).
- [15] J. Hu and F. Han, "A pixel-based scrambling scheme for digital medical images protection," *Journal of Network and Computer Applications*, vol. 32, no. 4, pp. 788-794, 2009. [Article \(CrossRef Link\)](#).



Azzam Sleit is an Associate Professor with the Computer Science Department, University of Jordan, where he also functioned as the Director of the Computer Center from 2007 through 2009. Before joining the University of Jordan in 2005, Dr. Sleit was the Chief Information Officer at Hamad Medical/ Ministry of Public Health, where he was responsible for developing and overseeing the execution of the information technology strategy of healthcare for State of Qatar. Dr. Sleit has over twenty years of experience and leadership in the information technology field including work at all levels of government, private and international sectors. Before joining Hamad Medical, Dr. Sleit was the Vice President of Strategic Group & Director of Professional Services of Triada, USA. Dr. Sleit also served as the Director of Professional Services at Information Builders, USA. His research interest includes imaging databases, information retrieval, and distributed systems. Dr. Sleit holds B.Sc., M.Sc. and Ph.D. in Computer Science. He received his Ph.D. in 1995 from Wayne State University, Michigan.



Abdel latif Abu dalhoum received his PhD degree in computer science from the University Autonoma De Madrid, Spain in 2004. He is currently an Associate Professor of evolutionary algorithms and complex systems at the University of Jordan. He has published about 25 papers in evolutionary algorithms, cellular automata, Fractals and DNA computing. He is a member of GHIA research group at the University Autonoma De Madrid.



Mohammed Qatawneh is an Associate Professor and Chairman of the Computer Science Department at the University of Jordan. He received his Ph.D. in Computer Engineering from Kiev University in 1996. Dr. Qatawneh published several papers in the areas of parallel algorithms, Networks and Embedding systems. His research interests include Parallel Computing, Embedding System, and Network Security.



Maryam Al-Sharief received her B.S. and M.S. degrees in Computer Science from King Abdulla II School for Information Technology, University of Jordan, Amman, Jordan, in 2008, and 2010 respectively. Her research interests include pattern recognition and image features extraction.



Ola Karajeh received her B.S. and M.S. degrees in Computer Science from King Abdulla II School for Information Technology, University of Jordan, Amman, Jordan, in 2008, and 2010 respectively. Her research interests include image processing and securing wireless networks.



Raw'aa Al-Jabaly received the B.S. degree from King Abdulla II School for Information Technology, University of Jordan, Amman, Jordan, in 2008. She is currently a master student in Computer Science at the University of Jordan, Amman, Jordan. Her research interests include image processing and cross layer optimization in Wireless Networks.