International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 5, No. 1, 2018, pp. 1-11.

ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Fuzzy Content Based Image Retrieval Using Software Agents

Monireh Azimi hemat

Department of Computer, Faculty of Engineering ,Payame Noor University, P.O. Box 19395-3697, Tehran, IRAN,

Abstract

Retrieve images in large data base, often is time consuming. In this paper we present a parallel image retrieval system implemented on software Agents. This method is based on multi- agent paradigm which is suitable for various parallelisms within applied problems. This paper focuses on how to implement a fuzzy content based image retrieval system, in the form of a multi agent model. The Implemented Results shown that the speed of this new system increase greatly.

Keywords: Image Retrieval, Multi Agents, Parallelism

Introduction:

Since 1970, image retrieval is an active field of research and researchers in two major areas are working in this field: database management and machine vision. The view of first group is Text-based and the vision of second group is based on visual characteristics. Text-based images retrieval began in early 1970, in which a general framework of image retrieval was presented first by indexing images with keywords, and then using database management systems for image retrieval.

There are two major problems in this kind of retrieval, especially when the size of database is large. The first one is the time spent for indexing and the second one is the large size of images and the different human perceptions of the same images.

In early 1990, due to rapid increase in high-volume collections of images and lack of accountability of systems based on text, image retrieval based on visual feature was introduced (liu, 2007 & Smeulders, 2000).

In content based image retrieval, user describes an arbitrary picture in the form of desired visual features and the image retrieval system retrieves closest image to what the user describes (Mafi et al., 2017). In these systems, visual features such as color, shape and texture are indexed (wang, 2011& yue, 2011& wang, 2012).

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Generally visual features are of a few tens or hundreds of orders, so in the large data base, searching in low-level features to find the nearest images would be very time-consuming. With growing database images, database search time increases linearly and at large databases, spending this time makes the system slow. To date, many studies have been done to solve time-consuming search problem. In the source (amato, 2008), by using feedback and fuzzy clustering methods, a knowledge base is designed. This knowledge base is used to make users aware of the overall organization of the image database during the query process. In another research, feature vectors in a given image are ordered by the principal axis analysis to speed up the similarity search in a high-dimensional image database. This work is done with using k nearest neighbor searches (cheng, 2006). Yildizer and his Cooperators (2012) propose an extremely fast CBIR system which uses Multiple Support Vector Machines Ensemble. They have used Daubechies wavelet transformation for extracting the feature vectors of images and present an efficient system .

In this paper a different method is presented for solving time-consuming search problem. In this method, we design a new distributed image retrieval system, and use of parallelism techniques.

In the past, large computers with multiple processors were used in large parallel processing systems (Mafi et al., 2012). But today, with the advent of intelligent systems, multi agent systems provide a suitable and general template for modular parallel design. Using multi agent systems the complex problems are broken to easier and smaller components and implemented by different agents (wooldrideg, 2002& vidal, 2001).

These systems are in fact distributed systems in which the agents do things. There is little research on the application of these systems to improve the efficiency of image retrieval systems, but they are applied in a variety of ways (Vitor, 2017& Esmaeili, 2017& Wang, 2017).

In the following, an image retrieval system is described, and after describing a fuzzy image retrieval system, the concepts of multi-agent systems and parallel processing are expressed. In the final sections of the paper, the sequential and parallel multi-agent models of the fuzzy image retrieval system, and how it is implemented, and the calculation of the speed improvement rate are described.

Theoretical basis of research:

Content Based Image Retrieval

Structure of a content based image retrieval system is shown in Figure 1. Image database is including images among which retrieval is done.

Appropriate visual features extracted from these images and features are kept on the feature database. Query processing subsystem extracts the appropriate features of the query image kept from users and sends it to the similarity measurement unit. In this unit, the similarity of query image features with features of each image in database is calculated and most similar images to query image are extracted and sent to the user interface.

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

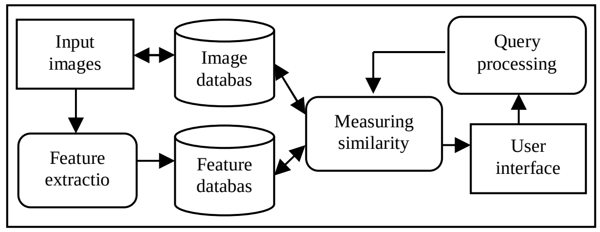


Figure 1: A system for content based image retrieval

Feature extraction is the key part of a content based image retrieval system. The main features are divided into two general and specific categories. General features usually are included color, texture, shape, and spatial relationships. These features in a general retrieval system which is applied for different images, is not so applicable. General features are divided into four general categories include: color, texture, shape and color layout (wang, 2011& yue, 2011& wang, 2012). Texture analysis algorithms are classified into statistical or structural methods, that segment images in regions or blocks of interest (Nilizadeh, 2014).

In this paper, a method is presented which is based on the color features and HSI (Hue-Saturation-Intensity) color space is used instead of RGB (Red-Green-Blue) which is usually used for color image modeling. Our reason to use HIS color space is its ability to design a retrieval system using only two Hue and Intensity components of space instead of using all three components of RGB space. Therefore we reduce the dimensions of feature vectors and then the time consumed by system is reduced.

In similarity measurement unit, the similarity of two images is defined as the distance between the feature vectors extracted from them. In other words, the similarity function is mapping the distance between two feature vectors to a positive real number which makes it possible to compare the visual features of the two images. Here we use fuzzy modeling and each of the two vectors as feature vectors of each image are expressed in the form of a fuzzy set. Therefore we are using fuzzy similarity measures to recognize similar images.

Fuzzy Color Image Retrieval

Suppose our image database is contained of n digital color images and we also have a query image which is called A_0. In order to retrieve the most similar images to the query image A_0, we must measure similarity between A_0 and any image A_j that is in the image database. Then the amounts of similarities are sorted in a descending form and only the images which are similar to query image, are retrieved. Therefore, the correct calculation of the similarity between color images is critical to the success of this system. Thus, we first describe the fuzzy modeling of color images and then we investigate the similarity of these fuzzy sets.

Fuzzy Partitioning of HSI Color Space

As it is described in the previous section, we use HSI (Hue, Saturation and Intensity) color space for representation of images. This space describes the color intuitively, and is based on the direct perception (Russ, 1999& Sharma, 2003). Hue component describes the color according to wavelength. Saturation component is the color value which is displayed. This component describes the differences between each

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

specific color with white light. Intensity component shows the amount of brightness of color. HSI color space can be modeled to a cylindrical coordinates. Hue component is represented with an angle that is variable from 0 to 360 degree. Saturation component is also in accordance with the radius of the cylinder which is variable from 0 to 1 (or from 0 to 255). Intensity component along the z axis can change, and therefore 0 is black and one (or 255) is white.

In order to extract the dominant color sets, we limit ourselves into eight main colors where each of them is modeling with trapezoidal fuzzy numbers (Chamorro-Martnez, 2005). A trapezoidal fuzzy number A (Fuller, 1991), is a fuzzy set with trapezoidal membership function. The trapezoidal membership function usually is depending on four scalar parameters a, b, c and d, as follow:

$$\mu_A(x\,;a,b,c,d) = \begin{cases} 0 & if\ x \le a, \\ \frac{x-a}{b-a} & if\ a < x < b, \\ 1 & if\ b \le x \le c, \\ \frac{d-x}{d-c} & if\ c < x < d \\ 0 & if\ d \le x \end{cases} \tag{1}$$

Thus we obtain a fuzzy classification for the hue component (Dubios, 2000) which is shown in Figure 2.

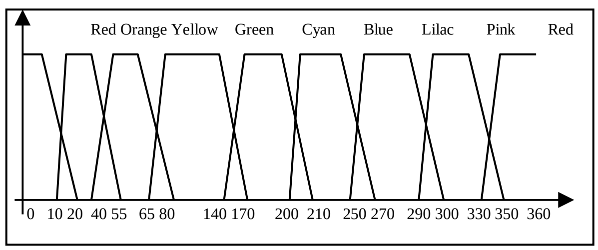


Figure 2: fuzzy partitioning of hue component

Also in order to increase the precision and to interference the amount of enlightening and darkening, we have used the intensity component. The fuzzy classification of this component is shown in Figure 3.

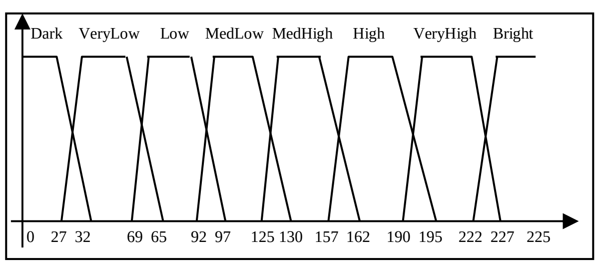


Figure 3: fuzzy partitioning of intensity component

Image Retrieval Based on Fuzzy Similarity Measures

Hue Component Similarity in Two Images

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

In this section, we compute the membership degree in each of eight main colors modeled in Figure 2, for each pixel in image. Then, we obtain for each hue value, the number of pixels with this hue value and a frequency chart is created for them. Here we consider a separate histogram for each color according to a range of hue values which has been covered.

Then each histogram is normalized by dividing all of its values to the total number of pixels, and thus these histograms are converted to fuzzy sets. In the following, we change the vertical and horizontal axis, so for each color, by having a degree of membership as input, which has a value between zero and one, we will have the number of pixels of that membership degree. Finally, for each image we have eight fuzzy sets that represent the membership degree distribution frequency in terms of eighth main colors. We express these fuzzy sets with their membership functions. The fuzzy sets for each image A_j and a color c is called $\mu^c_{A_j}$.

To calculate similarity between two fuzzy sets $\mu^c_{A_0}$ and $\mu^c_{A_j}$, $S\left(\mu^c_{A_0}, \mu^c_{A_j}\right)$, we use the measures which are defined to investigate the similarity between two sets (Van der Weken, 2003):

$$S\left(\mu^c_{A_0}, \mu^c_{A_j}\right) = \frac{\left|\mu^c_{A_0} \cap \mu^c_{A_j}\right|}{\left|\mu^c_{A_0} \cup \mu^c_{A_j}\right|} = \frac{\sum_{x \in S_c} \min\left(\mu^c_{A_0}(x), \mu^c_{A_j}(x)\right)}{\sum_{x \in S_c} \max\left(\mu^c_{A_0}(x), \mu^c_{A_j}(x)\right)} \tag{2}$$

Where S_c is the range [0, 1]. The same action for other seven sets is done to get the similarities between A_0 and A_j based on eight main colors. We have obtained the eight numbers and merge them as follow to calculate the overall similarity of two images for the components of hue:

$$S^h\left(A_0, A_j\right) = \frac{\sum_c S\left(\mu^c_{A_0}, \mu^c_{A_j}\right)}{8} \tag{3}$$

Intensity Components Similarity of Two Images

As previously mentioned, considering the intensity component is essential because in certain cases where we do not using of any color in the image, black and white are considered much like together. In This section, by having intensity component value for each pixel of image, we do the process the same as hue component. For each image A_j and intensity value d (instead of color c), we get eight $\mu^d_{A_j}$ fuzzy sets. The Intensity component similarity between the images A_0 and A_j in each of the eight $S\left(\mu^d_{A_0}, \mu^d_{A_j}\right)$ set, is obtained with the same method that utilize hue component. The overall similarity of two images in their intensity component is obtained from the following equation:

$$S^i\left(A_0, A_j\right) = \frac{\sum_d S\left(\mu^d_{A_0}, \mu^d_{A_j}\right)}{8} \tag{4}$$

The Overall Similarity of Two Images

Overall similarity of two images is obtained in terms of similarity of the two images in hue and intensity components as follows:

$$\left(A_0, A_j\right) = \frac{S^h\left(A_0, A_j\right) + S^i\left(A_0, A_j\right)}{2} \tag{5}$$

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Multi Agent Systems

An agent is an autonomous and independent entity which has the ability to process. It receives data from the environment through sensors, and by using some actuators, does the work according to its program in the environment (Weiss, 1999) Moreover, these agents have the ability to achieve their goals; they have interaction with environment and other agents in the environment.

Agents should not be confused with the objects or entities. An agent, especially an intelligent agent, is more than one object. An agent, unlike an object, is autonomous. It means that the autonomous agent follows series of actions which should reach its goals, and is capable of making decisions independently. The agents are integral component in processing environments. They receive information from their environment continuously and always carry out the actions whose result is reforming and changing the environment in order to achieve their goals.

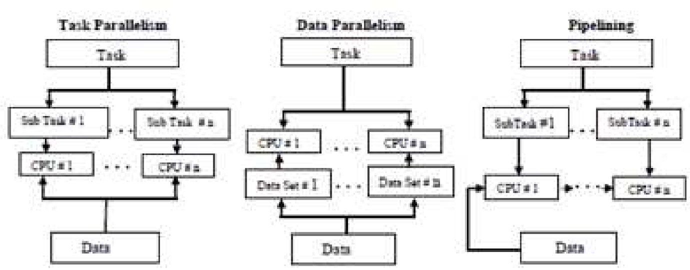
Intelligent agents have an important feature, the adaptability to environment. They can contribute with other agents to increase the probability of events that bring them to their goals continually. In traditional software environment, multi agent structures are used as a method for implementing adaptable and flexible modular system (Hale, 1994). In such systems, software agents are highly autonomous and they react based on received input and often are capable of learning and inference.

Most of these agents have been designed to react in unpredictable events and they are doing this action via comparison their operation with environmental changes (Jennings, 1995& Weiss, 1999). Such intelligent software agents have their own goals and have the ability to do a series of actions independently to reach these goals. They can observe environment and use the perimeter events to learn, inference and decision making.

In addition, these agents are designed someway that they can work together well. In other words, they are able to cooperate with other agents. Often, this cooperation is based on breaking large and complex problems into smaller and simpler tasks. Each of agents is able to do these tasks easily and quickly. As mentioned before, a multi agent system is a distributed system in which several agents are responsible for doing their duty. The implementation of such agent based systems can be done in the software distributed processing environment.

Parallel Processing Using Agents

In distributed systems, processes are assigned to different systems in order to run. One of the main advantages of distributed systems is their high speed which can be achieved by performing tasks in parallel. The first step of processing tasks within a distributed system is to choose the method of parallelism. In general three methods for basic parallelism can be used, tasks parallelism, data parallelism, pipelining and combination of these three methods. These methods are shown in Figure 4 (Naji, 2002).



Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Figure 4- parallel processing method

Research methodology

Multi Agent Image Retrieval Model

In the system that was introduced in section 3, in the first phase of image retrieval, after receiving the three RGB of color space components for each pixel two feature vectors (hue and intensity) for each picture are extracted.

In this phase separate operations are done on the hue and intensity components. These operations are independent from each other and can be done in parallel. On the other hand, each of these two independent processes consists of two parts that must be done consecutively. Multi agent model of this phase is shown in Figure 5.

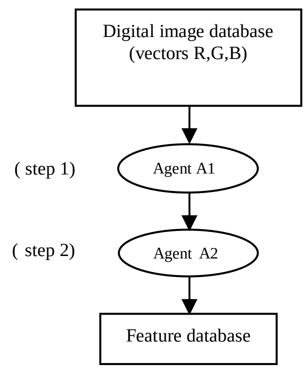


Figure 5-A: sequential multi agent model for first phase of image retrieval

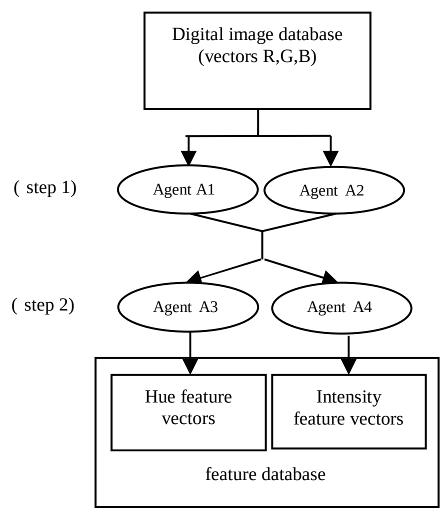


Figure 5-B: parallel multi agent model for first phase of image retrieval

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

In this model, agent A1 determines hue value for each pixel and calculates the total number of pixels with this hue value. Agent A2 is doing the same process for the intensity component. These two agents work in parallel and their output is as input of agents in the second step. Agent A3 normalizes the outputs of agent A1 according to the method described in the section 3-2. Then for each eight divided fuzzy sections of hue component and each membership value in the fuzzy interval [0,1], the number of pixels which their hue and membership value is the same will be calculated.

Thus there is a vector that representing the frequency of pixels with the same membership value in each of eight colors of image. This vector is hue feature vector of image and is stored in visual features database for indexing images.

Agent A4 also is doing the above operations on intensity component and stores a vector for that component in the features database. These two agents A3 and A4 also work in parallel. So far, we have presented the model of multi agent system using task parallelism that can reduce the load of work in the system at least to the half. It should be noted that this method of parallel processing could be used in different image retrieval systems which are based on multiple feature vectors. we intend to use pipelining method to increase the speed further.

Pipelining in the first phase of image retrieval can be used when features of all images in a large image data base should be extracted. In this case, after first image passes from step1 to step 2, simultaneously the second image enters into step 1. After ending these two stages, second image is entering into step 2 and simultaneously third image is entering into step 1 to be processed.

So far in the first retrieval phase we had a good improvement and now we intend to study the second phase. In this phase feature vectors to the query image should be extracted in a way that in Figure 7 is presented. Then these vectors must be compared with each feature vector in database. This comparison of vectors and applying similarity measure on them is done by agent A5. As we mentioned before, usually image databases have a high number images and comparing the query image with all of them is very time consuming. We have used another method of parallelism described in previous sections, data parallelism, to improve the system timing.

By the method of parallelism in this section images in the database are divided into two parts and two agents of type A5 compare images in this two sections with query image. These two agents A5-1 and A5-2 work in parallel and the time consumed in this section decreases at least to the half. In this section, according to the number of images in data base and the available hardware resources, the images can be divided into further two parts and also use more than two agents. This phase of image retrieval, using two agents, is shown in Figure 6. In this multi- agent modelling, first phase uses four agents, and second phase uses six agents.

The Analysis of Data

To prove the speed improvement of this system, we assume the database contains 1000 images that their features should be extracted and indexed. Also we have 100 images that most similar images to each of them in the database should be detected. If the first phase if 1000 images in a sequential and traditional form entered to system, the time is calculated as follows:

T1 = 1000 × (sequential running time of step1+ sequential running time of step2)

If they enter to system in parallel and pipelining form, the time is calculated as follow:

T1’ = parallel running time of step1 + 1000 × (maximum (parallel running time of step1 , parallel running time of step2)) + parallel running time of step2

The time to extract query image in traditional and sequential form in the second phase is calculated as

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

follow:

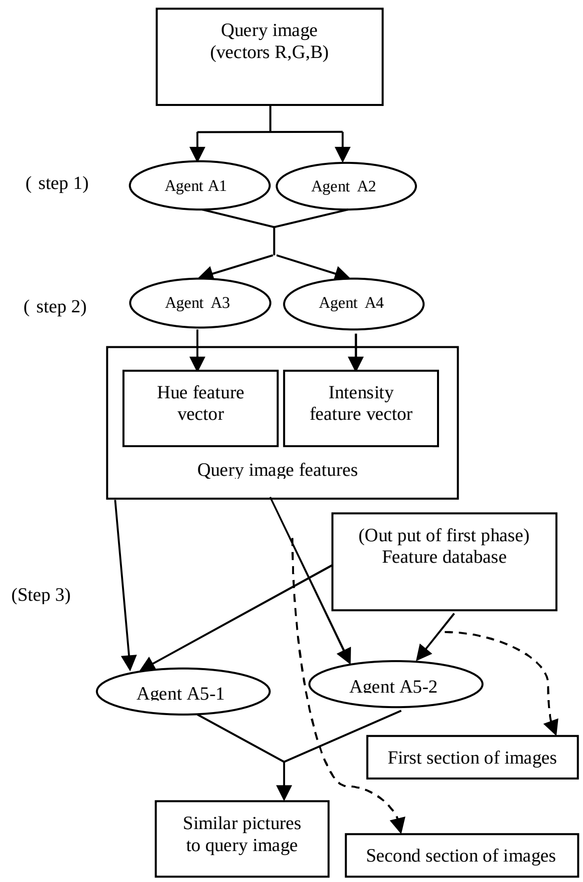


Figure 6- parallel multi agent model for second phase of image retrieval

T2 = sequential running time of step1 + sequential running time of step2 + sequential running time of step3

If this process is done in parallel based on model 2 timing is calculated as follow:

T2'= parallel running time of step1 + parallel running time of step2 + parallel running time of step3

And finally increase in speed of system is calculated as following:

$$\text{Speed up} = \frac{\text{sequentiol time}}{\text{paralel time}} = \frac{T1 + T2}{T1' + T2'} \quad (6)$$

Research Findings

We have implemented five agents in software on a 1.8 GHZ Dual Core system in both sequential and parallel method. The result is shown in table 1.

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

	Step 1	Step 2	Step 3	First Phase	Second phase
Sequential Time(ms)	14	13	9	27000	36
Parallel Time(ms)	8	7	5	8015	20

Table 1 – Results of software implementation

Speed up = 3.36

Which means parallel implementation by multi-agent system is three times faster.

Conclusion:

In this study, the general method for increasing speed and efficiency of content based image retrieval systems is presented. This method is based on multi agent distributed system. First, considered retrieval system is distributed in the form of multi agent model. Then concerning the nature of software agents, we exert different parallelism techniques on them.

Parallelism techniques with respect to availability of multi nuclear processors, cause the optimal use of these sources and decrease systems consumption time with simultaneous processing considerably. In this study, we use these methods on a sample image retrieval system and achieving the increase of almost 3-times speed. Although this method is implemented on a special system, but is generalizable on each image retrieval system which is used several image content features.

References:

Amato, A, and Lecce, V.D. (2008), "A knowledge based approach for a fast image retrieval system", Image and Vision Computing, 11 (26), pp. 1466-1480.

Chamorro-Martnez, J., Medina, J.M., Barranco, C., Galn-Perales, E., and Soto-Hidalgo, J.M. (2005), An approach to image retrieval on fuzzy objectrelational database using dominant Logic and Technology, EUSFLAT, pp. 676–684.

Cheng, S.C. and Wu, T.L. (2006), "Fast indexing method for image retrieval using k nearest neighbors searches by principal axis analysis", Journal of Visual Communication and Image Representation, 1 (17), pp. 42-56.

Coelho, V. N., Weiss Cohen, M., Coelho, I. M., Liu, N., Guimarães, F. G.(2017). Multi-agent systems applied for energy systems integration, State-of-the-art applications and trends in microgrids Applied Energy, Volume 187, , Pages 820-832.

Dubois, D. and Prade, H. (editors), (2000), Fundamentals of fuzzy sets, Kluwer Academic Publishers.

Esmaeili, A., Mozayani, n., Jahed Motlagh, M. R., Matson, E. T. (2017), A socially-based distributed self-organizing algorithm for holonic multi-agent systems, Case study in a task environment Cognitive Systems Research, Volume 43, Pages 21-44.

Fuller, R. (1991), On product-sum of triangular fuzzy numbers, Fuzzy Sets and Systems, vol. 41(1), pp. 83–87.

Hale, M.A., Craig, J. (1994), Preliminary development of agent technologies for a design integration framework, in: Proceedings of the Fifth Symposium on Multidisciplinary Analysis and Optimization, Panama City, FL.

Jennings, N. and Wooldridge, M. (1995), Intelligent Agents: Theory and Practice, The Knowledge Eng. Rev., vol. 10, no. 2 , pp. 115–152.

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018

Liu, Y., Zhang, D., Lu G.,, Ma, W.Y. (2007). A survey of content-based image retrieval with high-level semantics, Pattern Recognition, Vol. 40, No. 1, pp. 262-282.

Mehdi Mafi, Harold Martin, Malek Adjouadi, "HIGH IMPULSE NOISE INTENSITY REMOVAL IN MRI IMAGES", IEEE Signal Processing in Medicine and Biology Symposium (SPMB17)," Philadelphia, USA, Dec. 2017.

Mehdi Mafi, "Integration of Mobile Ad hoc and WIMAX Networks with Approach of Admission Control and Hand off Combination Applied in Telemedicine Services," American Journal of Scientific Research, vol. 83, 2012, pp. 14-24.

Mehdi Mafi, "A Hierarchical Model of ICT in Digital Society to Access Information," Canadian Journal on Electrical and Electronics Engineering, vol. 3, issue 7, 2012, pp. 366-374.

Mehdi Mafi, "Integration of Mesh WLAN and WiMAX Networks Applied in Telemedicine services to Transfer Fixed and Mobile User`s Data," Second Conference on Telemedicine, Tehran, Iran, Jun. 2014.

Naji, H. R., Etzkorn, L., Adhami, R., Earl Wells, B. (2002), Parallel Image Processing with Agent-based Reconfigurable Hardware, Proceedings of the 15th International Conference on Parallel and Distributed Computing Systems (PDCS 2002), Louisville, KY.

Nilizadeh. A. F., Naghsh Nilchi, A. R. (2014). Block texture pattern detection based on smoothness and complexity of neighborhood pixels, International Journal of Image, Graphics and Signal Processing (IJIGSP), Vol. 6, No 5, pp. 1-9.

Russ, J.C. (1999) The image processing handbook, CRC Press.

Reza Sarraf Shirazi, Mehdi Mafi, Habib Azizi, "A Low Noise PLL Frequency Synthesizer in 2.4 GHz with 1MHz Frequency Step," International Organization of Scientific Research Journal of Engineering, vol. 2, issue 8, 2012, pp. 196-200.

Sharma, G. (2003) Digital color imaging handbook, CRC Press.

Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R. (2000). Contentbased image retrieval at the end of the early years, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 12, pp. 1349– 1379.

Van der Weken, D., Nachtegael, M., and Kerre, E.E. (2003), Using similarity measures for histogram comparison, Lecture Notes in Artificial Intelligence, vol. 2715, pp. 396–403.

Vidal, Jose M., Bohlor Paul A., Huhns, Michael N. (2001). Inside an Agent, IEEE Internet Computing; Jan-Feb.

Wang, X.Y. and Yu, Y.J and Yang, H.Y (2011), An effective image retrieval scheme using color, texture and shape features, Computer Standards & Interfaces, 1 (33), pp. 59-68.

Wang, X.Y. and Chen, Z.F. and Yun, J.J. (2012), An effective method for color image retrieval based on texture, Computer Standards & Interfaces, 1 (34), pp.35-31

Wang, W., Wen, C., Huang, J. (2017), Distributed adaptive asymptotically consensus tracking control of nonlinear multi-agent systems with unknown parameters and uncertain disturbances Automatica, Volume 77, Pages 133-142.

Weiss, G. (1999), Multiagent Systems—A Modern Approach to Distributed Artificial intelligence, MIT Press, Cambridge, MA.

Wooldridge M. (2002). An introduction to MultiAgent Systems, 1st edition, John Wiley & Sons.

Yildizer, E. and Balci, A.M. and Hassan, M. and Alhajj, R. (2012), "Efficient content-based image retrieval using Multiple Support Vector Machines Ensemble", Expert Systems with Applications, 3 (39), pp.2385-2396

Yue, J. and Li, Z. and Liu, L. and Fu, Z. (2011), Content-based image retrieval using color and texture fused features, Mathematical and Computer Modelling, 3-4 (54), pp. 1121-1130.

Received:8 January 2018/Accepted:12 April 2018/Published:1 June 2018