THE JPEG IMAGE COMPRESSION ALGORITHM

THE JPEG IMAGE COMPRESSION ALGORITHM

Muzhir Shaban AL-Ani, Fouad Hammadi Awad
College of Computer, Anbar University, Anbar, Iraq

ABSTRACT

The basis for the JPEG algorithm is the Discrete Cosine Transform (DCT) which extracts spatial frequency information from the spatial amplitude samples. These frequency components are then quantized to eliminate the visual data from the image that is least perceptually apparent, thereby reducing the amount of information that must be stored. The redundant properties of the quantized frequency samples are exploited through quantization, run-length and huffman coding to produce the compressed representation. Each of these steps is reversible to the extent that an acceptable approximation of the original space-amplitude samples can be reconstructed from the compressed form. This paper examines each step in the compression and decompression.

KEYWORD: *Image Compression, JPEG, DCT, Quantization, Run-Length Coding.*

I. INTRODUCTION

In the late 1980's and early 1990 a joint committee known as the Joint Photographic Experts Group (JEGP) of the International Standards Organization (ISO) and the Comitte Consultatif International Telephonique et Telegraphique (CCITT) developed and established the first in ternational compression standard for continuous tone images [1]. In June 1987, conducted a selection process based on behind assessment of subjective picture quality and narrowed 12 proposed method to three. Three information working group formed to refine them and in January 1988 a second more rigorous selection process revealed that the best on the 8x8 DCT, had produced the best picture quality [2] . Devices for image acquisition, data storage, and bitmapped printing and display have brought about many applications of digital imaging. However, these applications tend to be specialized due to their relatively high cost .With the possible exception of facsimile digital images are not commonplace in general purpose computing systems the way text and geometric graphics are. The majority of modern business and consumer usage of photographs and other types of images takes place through more traditional analog means [10]. The key obstacle for many applications is the vast amount of data required to represent a digital image directly. A digitized version of a single, color picture at TV resolution contains on the order of one million bytes 35mm resolution requires ten times that amount Use of digital images often is not viable due to high storage or transmission costs, even when image capture and display devices are quite affordable. Modern image compression technology offers a possible solution. State-of-the-art techniques can compress typical images from 1/10 to 1/50 their uncompressed size without visibly affecting image quality. But compression technology alone is not sufficient. For digital image applications involving storage or transmission to become widespread in today's marketplace, a standard image compression method is needed to enable interoperability of equipment from different manufacturers [2].Nowadays, the size of storage media increases day by day. Although the largest capacity of hard disk is about two Terabytes, it is not enough video file without compressing it [3].

The rest of the paper is structured in following manner. In section 2 a brief explained about image compression. Section 3 a brief background to the related work is provided. Section 4 Architecture of the proposed system. Section 5 Result and analysis. The conclusion and future work appear in the final section of the paper.

II. IMAGE COMPRESSION

Image compression addresses the problem of reducing the amount of data required to represent a digital image. It is a process intended to yield a compact representation of an image, thereby reducing the image storage/transmission requirements. Compression is achieved by the removal of one or more of the three basic data redundancies, Coding Redundancy, Interpixel Redundancy, Psychovisual Redundancy .Coding redundancy is present when less than optimal code words are used. Interpixel redundancy results from correlations between the pixels of an image. Psychovisual redundancy is due to data that is ignored by the human visual system (i.e. visually non essential information). Image compression techniques reduce the number of bits required to represent an image by taking advantage of these redundancies. An inverse process called decompression (decoding) is applied to the compressed data to get the reconstructed image. The objective of compression is to reduce the number of bits as much as possible, while keeping the resolution and the visual quality of the reconstructed image as close to the original image as possible. Image compression systems are composed of two distinct structural blocks: an encoder and a decoder [9] [10].

III. RELATED WORK

There are several systems that implemented for image compression during previous years. In 1991 Gregory K. Wallace shows JPEG features a simple lossy technique known as the Baseline method, a subset of the other DCT-based modes of operation. The Baseline method has been by far the most widely implemented JPEG method to date and is sufficient in its own right for a large number of applications. This article provides an overview of the JPEG standard and focuses in detail on the Baseline method [1].

In 2005 John W. O'Brien introduce the JPEG Algorithm ,The basis for the JPEG algorithm is the Discrete Cosine Transform (DCT) which extracts spatial frequency information from the spatial amplitude samples .that examines each step in the compression sequence with special emphasis on the DCT [2]. Sonal, Dinesh Kumar in 2005 explain a study of various image compression techniques that presents the Principal Component Analysis approach applied to image compression. PCA approach is implemented in two ways – PCA Statistical Approach & PCA Neural Network Approach. It also includes various benefits of using image compression techniques [3].

In 2007 Jacques Levy Vehel, Franklin Mendivil and Evelyne Lutton introduce ovecompressing JPEG images with Evolution Algorithms this Evolutionary strategies are used in order to guide the modification of the coefficients towards a smoother image and the result was three compression ratios have been considered: The compressed images are obtained by using the quantization values in table 1 multiplied by 5, 10, and 15. [4].

In 2008 Jin Li, Jarmo Takala, Moncef Gabbouj and Hexin Chen used a detection algorithm for zero quantized DCT coefficients in jpeg show Experimental results show that the proposed algorithm can significantly reduce the redundant computations and speed up the image encoding. Moreover, it doesn't cause any performance degradation. Computational reduction also implies longer battery lifetime and energy economy for digital applications [5].

In 2012 Bheshaj Kumar, Kavita Thakur and G. R. Sinha introduce performance evaluation of JPEG image compression using symbol reduction technique. In this paper, a new technique has been proposed by combining the JPEG algorithm and Symbol Reduction Huffman technique for achieving more compression ratio. The symbols reduction technique reduces the number of symbols by combining together to form a new symbol. As a result of this technique the number of Huffman code to be generated also reduced. The result shows that the performance of standard JPEG method can be improved by proposed method. This hybrid approach achieves about 20% more compression ratio than the Standard JPEG [6].

IV. Architecture of the Proposed System

JPEG algorithm has four modes and many options. It is more like a shopping list than a single algorithm. For our purposes, though only the lossy sequential mode is relevant, and that one is illustrated in Figure.1 steps of JPEG algorithm Furthermore, we will concentrate on the way JPEG is normally used to encode 24-bit RGB images.

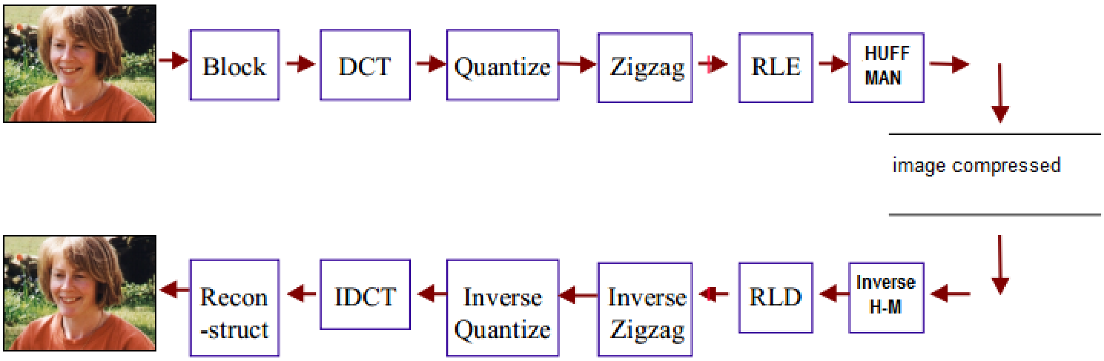

Figure.1 The steps of JPEG algorithm in lossy sequential mode.

Step 1 of encoding an image with JPEG is block preparation. For the sake of specificity we assume that the JPEG input is a 640×480 RGB image with 24 bits/pixel, as shown in Figure 2(a). Since using luminance and chrominance gives better compression, we first compute the luminance Y and the two chrominances Cb, Cr and the inverse, according to the following equations1, 2:

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ Cb &= 0.564(B - Y) \\ Cr &= 0.713(R - Y) \end{aligned} \qquad \text{---------(1)}$$

$$\begin{aligned} R &= Y + 1.402Cr \\ G &= Y - 0.344Cb - 0.714Cr \\ B &= Y + 1.772Cb \end{aligned} \qquad \text{---------(2)}$$

Separate matrices are constructed for Y, Cb, and Cr, each with elements in the range 0 to 255. Next, square blocks of four pixels are averaged in the Cb and Cr matrices to reduce them to 320×240. This reduction is lossy, but the eye barely notices it since the eye responds to luminance more than to chrominance. Nevertheless, it compresses the total amount of data by a factor of two .Now 128 is subtracted from each element of all three matrices to put 0 in the middle of the range.

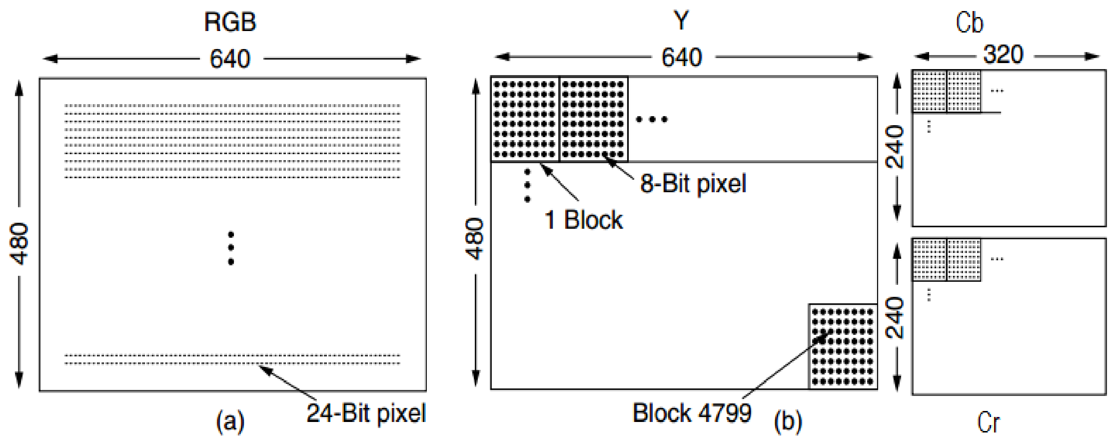

Figure. 2(a) RGB input data. (b) After block preparation.

Finally, each matrix is divided up into 8×8 blocks. The Matrix has 4800 blocks; the other two have 1200 blocks each, as shown in Figure 2(b).

Step 2 of JPEG is to apply a DCT (Discrete Cosine Transformation) according to equation 3 and 4 to each of the 7200 blocks separately. The output of each DCT is an 8×8 matrix of DCT coefficients. DCT element (0, 0) is the average value of the block. The other elements tell how much spectral power is present at each spatial frequency. In theory a DCT is lossless but in practice using floating-point numbers and transcendental functions always introduces some round off error that results in a little information loss. Normally, these elements decay rapidly with distance from the origin (0, 0) as shown in Figure 3.

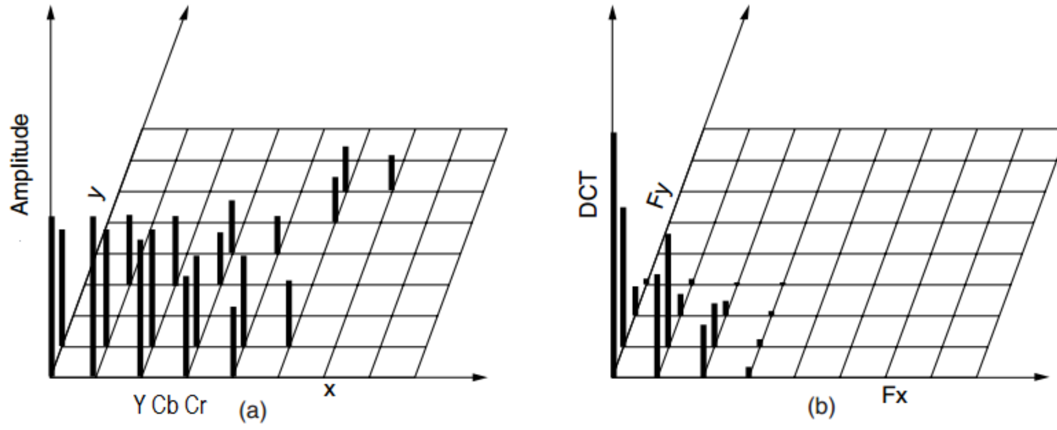

Figure .3 (a) One block of the Y matrix. (b) The DCT coefficients.

$$F(u,v) = \frac{1}{4} C(u)C(v)\left[\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) * \cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16}\right] \quad \textbf{(3)}$$

$$f(x,y) = \frac{1}{4}\left[\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u,v) * \cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16}\right] \quad \textbf{(4)}$$

where: $C(u),\ C(v) = 1/\sqrt{2}$ for $u,v = 0$;

$C(u),\ C(v) = 1$ otherwise.

Once the DCT is complete, JPEG moves on to **step 3**, called quantization.In which the less important DCT coefficients are wiped out. This lossy transformation is done with table quality 50 by dividing each of the coefficients in the 8×8 DCT matrix by a weight taken from a table. If all the weights are 1, the transformation does nothing. However, if the weights increase sharply from the origin, higher spatial frequencies are dropped quickly.

An example of this step is given in Figure 4, in which we see the initial DCT matrix, the quantization table and the result obtained by dividing each DCT element by the corresponding quantization table element. The values in the quantization table are not part of the JPEG standard. Each application must supply its own allowing it to control the loss-compression trade-off.

DCT Coefficients

150	80	40	14	4	2	1	0
92	75	36	10	6	1	0	0
52	38	26	8	7	4	0	0
12	8	6	4	2	1	0	0
4	3	2	0	0	0	0	0
2	2	1	1	0	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Quantization table

1	1	2	4	8	16	32	64
1	1	2	4	8	16	32	64
2	2	2	4	8	16	32	64
4	4	4	4	8	16	32	64
8	8	8	8	8	16	32	64
16	16	16	16	16	16	32	64
32	32	32	32	32	32	32	64
64	64	64	64	64	64	64	64

Quantized coefficients

150	80	20	4	1	0	0	0
92	75	18	3	1	0	0	0
26	19	13	2	1	0	0	0
3	2	2	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure. 4 Computation of the quantized DCT coefficients.

Step 4 reduces the (0, 0) value of each block (the one in the upper-left corner) by replacing it with the amount it differs from the corresponding element in the previous block. Since these elements are the averages of their respective blocks, they should change slowly, so taking the differential values should reduce most of them to small values. No differentials are computed from the other values. The (0,0) values are referred to as the DC components the other values are the AC components.

Step 5 linearizes the 64 elements and applies run-length encoding to the list. Scanning the block from left to right and then top to bottom will not concentrate the zeros together, so a zigzag scanning pattern is used as shown in Figure 5.

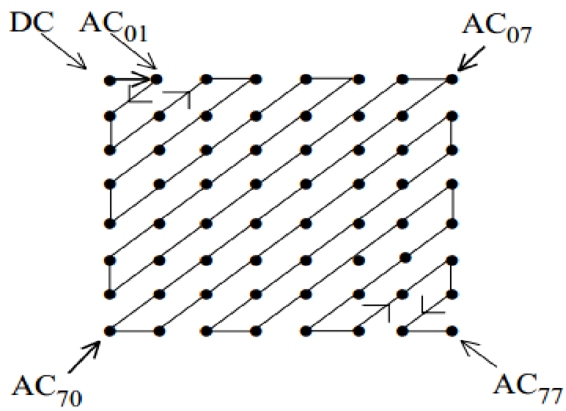

Figure 5 sequence of zigzag.

In this example, the zigzag pattern produces 38 consecutive 0s at the end of the matrix. This string can be reduced to a single count saying there are 38 zeros, a technique known as run-length encoding as following

Example data: 20,17,0,0,0,0,11,0,-10,-5,0,0,1,0,0,0, 0 , 0 ,0 , only 0,..,0

RLC for JPEG compression (0,20) ; (0,17) ; (4,11) ; (1,-10) ; (0,-5) ; (2,1); EOB.

Step 6 Huffman-encodes the numbers for storage or transmission assigning common numbers shorter codes that uncommon ones. The details of Huffman table for luminance DC in table .1 and Huffman table for chrominances AC in table .2

Table. 1 Number of codeword v s. code length for Huffman table for luminance DC.

Codelength	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Number of codeword	0	1	5	1	1	1	1	1	1	0	0	0	0	0	0	0

Algorithm 1: Generate Huffman table

```
i=0 , code value =0;
for(k=1;k<=16;k++)
{
For(j=1;j<=number_of_codeword[k];j++)
Codeword[i]=codevalue;
Codelength[i]=k;
Codevalue++;  i++;
   { Codevalue*=2;  }
```

Table.2 Number of codeword vs. code length for Huffman table for luminance AC.

Codelength	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Number of codeword	0	2	1	3	3	2	4	3	5	5	4	4	0	0	1	125

V. Results and Analysis

We take five images for test, in different size and different quality, the size of the first image is 192 Kb the result shows after compression become 7.4 Kb, the peak signal to noise ratio 31.9 while the size of image .2 is 74.3 Kb become 2.4 Kb ,the peak signal to noise ratio 34.8, and the size of image .3 is 74.3 Kb when compressed become 2.8 Kb while PSNR 32.3,the size of image .4 is 74.3 Kb become 5.4 Kb , PSNR is 28.9 at the end size of image .5 is 858 Kb when compressed become 83.3Kb and the units measures PSNR is 36.5 the Fig.6 show input images and Fig.7 show output images as shown in table 3.

image 1 image2 image3 image4 image5

Figure.6 the input images

Table.3 the obtained results

No.	Size of original image	Size after image compression	PSNR
1	192 Kb	7.4 Kb	31.9
2	74.3 Kb	2.4 Kb	34.8
3	74.3 Kb	2.8 Kb	32.3
4	74.3 Kb	5.4 Kb	28.9
5	858 Kb	83.3 Kb	36.5

image 1 image 2 image 3 image 4 image 5

Figure.7 decompressed images

VI. CONCLUSION

Image compression is an extremely important part of modern computing. by having the ability to compress images to fraction of their original size valuable and expensive disk space can be saved .In addition, transportation of images from one computer to another becomes easier and (which is why image compression has played such as important role in the development of the internet). The JPEG image compression algorithm provides a very effective way to compress images with minimal loss in quality .Although the actual implementation of the JPEG algorithm is more difficult than other image format (such as png) and the actual compression of image is expensive computationally, the high compression ratios that got attained using the JPEG algorithm easily compensate for the amount of time spent implementation the algorithm and compressing an image that give good result that indicate 36.5 of PSNR , when JPEG-compressed digital images come to be regarded and even taken for granted as just another data type as text and graphics are today.

VII. FUTURE WORK

There are several suggestions given below that could be implemented in the future to make the project more optimal:

- An exploration of Huffman coding in the context of probability, information theory and can be apply shift coding instead of huffman coding.
- A review of the other modes of operation of the JPEG algorithm.
- Applications of the DCT or similar transforms to the compression and manipulation of other kinds of data.

REFERENCE

[1]. Gregory K. Wallace," The JPEG Still Picture Compression Standard", Submitted in December 1991 for publication in IEEE Transactions on Consumer Electronics.
[2]. John W. O'Brien ," The JPEG Image Compression Algorithm" , APPM-3310 FINAL PROJECT, DECEMBER 2, 2005.
[3]. Sonal, Dinesh Kumar," a study of various image compression techniques" , Guru Jhambheswar University of Science and Technology, Hisar , 2005.
[4]. Jacques Levy Vehel , Franklin Mendivil and Evelyne Lutton ," Overcompressing JPEG images with Evolution Algorithms" , Author manuscript, published in "EvoIASP2007, Valencia : Spain (2007
[5]. Jin Li , Jarmo Takala , Moncef Gabbouj and Hexin Chen ," detection algorithm for zero quantized DCT coefficients in jpeg" , Authorized licensed use limited to: Tampereen Teknillinen Korkeakoulu Downloaded on February 8, 2009 at 05:16 from IEEE Xplore.
[6]. Bheshaj Kumar , Kavita Thakur and G. R. Sinha ," introduce performance evaluation of JPEG image compression using symbol reduction technique", Natarajan Meghanathan, et al. (Eds): ITCS, SIP, JSE-2012, CS & IT 04, pp. 217–227, 2012.
[7]. Lian E. G. Richardson , " video codec design developing image and video compression systems" , JOHN WILEY & SONS Ltd, Baffins Lane chichester , West Sussex PO19 IUD, England , 2002.
[8]. William B. pennebaker ,Joan L. Mitchell , "JPEG still image data compression standard " , Library of congress cataloging- in -publication, Eight printing 2004 by kluwer Acadimic publishers.
[9]. Pao-Yen Lin ," Basic Image Compression Algorithm and Introduction to JPEG Standard" , National Taiwan University, Taipei, Taiwan, ROC 2009.

[10]. G.M.Padmaja and P.Nirupama ," Analysis of Various Image Compression Techniques", ARPN Journal of Science and Technology 2011- 2012. All rights reserved.
[11]. Iain E. G. Richardson," H.264 and MPEG-4 Video Compression", John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester West Sussex PO19 8SQ, England ,2003.
[12]. James Rosenthal, " JPEG Image Compression Using an FPGA ", A Thesis submitted in partial satisfaction of the requirements for the degree Master of Science in Electrical and Computer Engineering, December 2006.
[13]. Muzhir Shaban Al-Ani and Talal Ali Hammouri," Frame Selection Key to Improve Video Compression", 1 Amman Arab University, Department of Computer Science, Amman-Jordan, 11953,2010.
[14]. Muzhir Shaban Al-Ani and Talal Ali Hammouri ,"Video Compression Algorithm Based on Frame Difference Approaches ", International Journal on Soft Computing (IJSC) Vol.2, No.4, November 2011.
[15]. Colin Doutre, *Student Member*, *IEEE*, Panos Nasiopoulos, *Member*, *IEEE*, and Konstantinos N. Plataniotis, *Senior Member*, *IEEE*," H.264-Based Compression of Bayer Pattern Video Sequences", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 18, NO. 6, JUNE 2008.
[16]. Chien- Ting Kuo, I- Iming Chen, and Feng- Li Lian ," Intelligent video transmission control for mobile Cameras ", 2009 IEEE International Symposium on Intelligent Control Part of 2009 IEEE Multi-conference on Systems and Control Saint Petersburg, Russia, July 8-10, 2009.
[17]. Mohammed Ghanbari ," Standard Codecs: Image Compression to Advanced Video Coding" , ISBN:0852967101,2003.
[18]. Wei- Yi Wei ," An Introduction to Image Compression " , National Taiwan University, Taipei, Taiwan, ROC,2008.
[19]. S. T. Klein,. Y. Wiseman, “Parallel Huffman Decoding with Applications to JPEG Files“2003.
[20]. B. Fang. G. Shen Techniques for Efficient DCT / IDCT Implementation on Generic. May 2005.

AUTHORS

[1]**Muzhir Shaban Al-Ani** has received Ph. D. in Computer & Communication Engineering Technology, ETSII, Valladolid University, Spain, 1994. Assistant of Dean at Al-Anbar Technical Institute (1985). Head of Electrical Department at Al-Anbar Technical Institute, Iraq (1985-1988), Head of Computer and Software Engineering Department at Al-Mustansyria University, Iraq (1997-2001), Dean of Computer Science (CS) & Information System (IS) faculty at University of Technology, Iraq (2001-2003). He joined in 15 September 2003 Electrical and Computer Engineering Department, College of Engineering, Applied Science University, Amman, Jordan, as Associated Professor. He joined in 15 September 2005 Management Information System Department, Amman Arab University, Amman, Jordan, as Associated Professor, then he joined computer science department in 15 September 2008 at the same university. He joined in August 2009 Computer Science Department, Anbar University, Anbar, Iraq, as Professor.

[2]**Fouad Hammadi Awad** has received B.Sc. in Computer Science, Al-Anbar University, Iraq, (2007-2011). M.Sc student (2012- till now) in Computer Science Department, Al - Anabar University. Fields of interest: Adapting of video signal over cellular mobile devices. He taught many subjects such as cryptography, operation system, computer vision and image processing.