Fast Intra-frame Compression for Video Conferencing using Adaptive Shift Coding

Ali Makki Sagheer
Information System Department
College of Computer
University of Anbar, Ramadi, Iraq

Ahmeed Suliman Farhan
Computer Science Department
College of Computer
University of Anbar, Ramadi, Iraq

Loay E. George
Computer Science Department
College of Science
University of Baghdad, Iraq

ABSTRACT

Overhead of video transmission over the Internet is increasing the exponentially every day. Optimization of natural bandwidth is the basic motive by compressing video to the maximum extend. This paper introduces a new idea for intra-frame compression. Most standard compression algorithms depended on Huffman coding technique. The adaptive shift coding technique reduces the number of bits which required to implement the data depending on the value. The result shows that the new technique more efficient for the video conference system because they're depending on real time and the adaptive shift coding technique reduce the compression time.

General Terms

Video compression, image compression, video conference.

Keywords

Video Conferencing, Adaptive Shift Coding, Discrete Cosine Transform, Run-Length-Encoding, Quantization.

1. INTRODUCTION

Multimedia is simply generated using various forms such as video, audio, image, text, etc. As the availability of inexpensive and powerful cameras and video recorders increases, the need for new techniques in processing these media has further grown [1]. The data is compressed prior to their storage and/or transmission. The purpose of data compression is to remove the unnecessary redundancy from the original signal. Therefore, the number of binary bits required to represent the data contained within the original signal will be reduced. To do the best compression ratio, it is not only required to understand the nature of the original signal in its binary form, but also knowing how humans interpret that binary information that the signal represents [2]. Video and image compression is usually done in a different representation other than their original domains. These domains are spatiotemporal for videos and images, respectively [1]. Schemes for digital video compression are paramount importance in today’s wireless telecommunication and multimedia system, where bandwidth is a valuable commodity. Therefore, video compression schemes are ultimate important so as to reduce the number of bits needed to represent a frame sequence through transmission. This operation must be done without affecting the subjective quality of the transmitted frame sequence [2]. The quality is required to reconstruct image and video depending on its application such as medical diagnosis and some other scientific fields. The reconstructed image and video are needed to be a perfect copy of the original image and video, this scheme must be reversed as an information-preserving schemes. Such compression is called lossless compression. Other application may allow a certain amount of information loss such as motioned picture, video conference and television (TV). These types of compression refer to lossy compression [3].

2. RELATED WORKS

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 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 [4].

In 2008 Aree A. Mohammad and Loay E. George proposed a new scheme for image compression; this method involves two steps. The first step uses a lifting scheme wavelet-based transform. Second step develops a modified entropy coding algorithm. The tested result shows that the quality maintains the same value for different coding techniques. The compression factor reaches a good ratio with block sublevel coding algorithm but the computational time is costly [5].

In 2008 Andreoli et al produced two new algorithms for the reuse of H.263+ prediction modes in H.264 Intra mode prediction. The performances of these techniques compare to the full transcoding approach in term of PSNR. Tests reveal a high correlation between PSNR scores with a residual error up to 0.1 dB. A 10% increase of the overall bitrate is revealed, but a 27% reduction of computational burden is also measured [6].

In 2012, Kekre et al proposed a method for video conferencing, storage of movies and CCTV footage that combines both lossy half tone and lossless Wavelet Transform techniques so as to obtain low-bit rate video data transmission. In their work decimal values of bitmapped image are to be converted into either 1 or 0 in the half toning process that incur pictorial loss and gives 8:1 compression ratio (CR) irrespective of image while Wavelet Transform is applied on halftone image for higher compression for various levels. They applied their technique on ten sample images of different people captured by Nikon camera are used for experimentation. All images are bitmap (.BMP) 512 X 512 in size, and the experimental result shows the higher CR, minimum Mean Square Error (MSE) [7].

3. VIDEO COMPRESSION FOR VIDEO CONFERENCING

Video conferencing applications have been extensively utilized in recent years in order to provide real-time video communication. The need for highly efficient video compression algorithms that can be used successfully in the casual computer devices with limited bandwidth comes mainly from the cost of computing and network resources limitations. As a result, the most desirable algorithm is the one that achieves good compression while maintaining a satisfactory visual quality. Low computational complexity and real-time performance would be an advantage because it will allow the algorithm to operate on a wide variety of less powerful computers [13].

Video encoder combines intra-frame and inter-frame to provide fast processing for video conference. The algorithm creates two types of frames compression. First; intra-frames is compressed using DCT, quantization, and entropy variable-length coding similarly to JPEG. Second; predictive inter-frames, compressed using Differential Pulse Code Modulation (DPCM) and motion estimation [8].

4. FIDELITY CRITERIA

Quantifying the video quality is a challenging task and often imprecise art, because numerous factors may affect the results [11]. To determine, evaluate and compare video communication systems it is necessary to specify the quality of the video images showed to the user. Since there are many factors that can affect the results, therefore, measuring the quality of vision is difficult and inaccurate [9]. Two of the most evaluation criteria for video and image compression are Peak Signal to Noise Ratio (PSNR) and Compression Ratio (CR).

4.1 Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio (PSNR) (Equation 1) can be calculated using a logarithmic scale and this calculation depends on the mean squared error (MSE) of between an original and an impaired image or video frame, relative to (2n −1)2 (the square of the highest-possible signal value in the image, where n is the number of bits per image sample).

$$PSNR_{dB} = 10\, log_{10} \frac{(2^n-1)^2}{MSE} \qquad (1)$$

$$MSE = \frac{1}{N}\sum_{n=1}^{N}(x_n - y_n)^2 \qquad (2)$$

PSNR is a very popular quality measure because of its easy and quick calculations. It is widely used in comparing the quality of a compression and decompression of video and images [9].

4.2 Compression Ratio (CR)

It is crucial to know how much detail coefficient can be discarded from the input data in order to preserve critical and necessary information of the original data in the process of image compression. The Compression Ratio (CR) can be defined as:

$$CR = \frac{Compression\ data}{Original\ data} \qquad (3)$$

CR is a measure of the reduction of the detail coefficient of the data. It can be modified to get various qualities for the compressed image. The higher the CR can be applied, the more the details coefficients are discarded which means lower reconstruction quality of the image [12].

5. THE PROPOSED INTRA-FRAME COMPRESSION

The main operations of Intra-frame compression are similar to JPEG image compression. The real time coding is required for video conferencing compression; therefore the time of execution must be as faster as. To achieve that, we produced Intra-frame compression based on proposed adaptive shift coding instead of Huffman coding which is previously used in JPEG and costly in execution.

In this paper we introduce an idea for frame compression depending on DCT and Adaptive Shift Coding as illustrated in Fig 1.

5.1 Ycbcr Frame Representation

The purpose of this step is to convert a frame from RGB to Ycbcr. The input frame is a 176×144 RGB frame with 24 bits/pixel, as shown in Fig 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 equations 4, 5:

$$\begin{aligned} &Y=0.299R+0.587G+0.114B \\ &Cb=0.564(B-Y) \qquad (4) \\ &Cr=0.713(R-Y) \end{aligned}$$

$$\begin{aligned} &R=Y+1.402Cr \\ &G=Y-0.344Cb-0.714Cr \qquad (5) \\ &B=Y+1.772Cb \end{aligned}$$

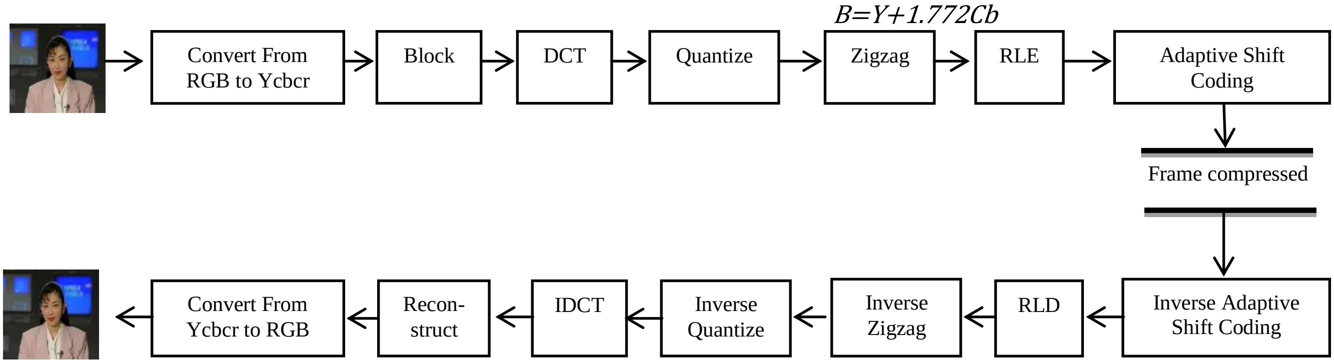


Fig 1: The operation of The Compression Algorithm.

Separate matrices are constructed for Y, Cb , and Cr, each with elements in the range 0 to 255. Next, blocks of four pixels are averaged in the Cb and Cr matrices to reduce them to 88×72. This reduction is lossy, but it is 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 .

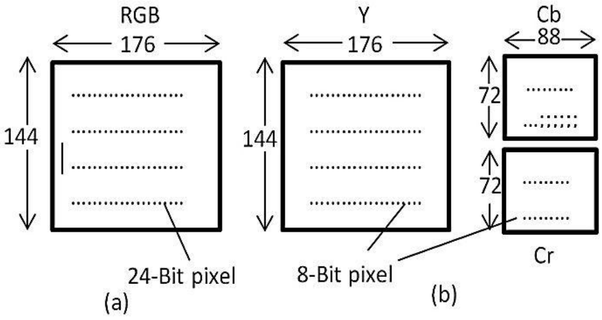


Fig 2: (a) RGB input data. (b) After convert to YCbCr.

5.2 Discrete Cosine Transformation(DCT)

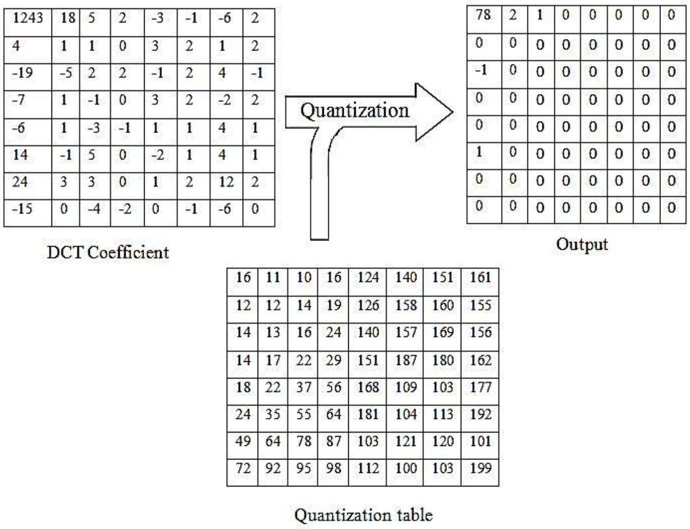
Apply a DCT according to equation 6 and 7 on each 8*8 blocks in the frame. The output matrix consists of DCT coefficients, which is arranged in a way that coefficients containing useful and important data for representation of the frame be in the upper left of the matrix and in the lower right coefficients containing less useful information. The DC|T coefficient is at position (0, 0) in the upper left-hand corner of the matrix and it represents the average of the other 63 value in the matrix.

5.3 Quantization

Quantization is the step where we throw away data. The DCT is a lossless procedure. The data can be precisely recovered through the IDCT (this isn't entirely true, in reality the DCT and IDCT contain transcendental functions which no physical implementation can be computed with perfect accuracy). During Quantization every element in the 8x8 DCT matrix is divided by a corresponding element in a quantization matrix Q to yield a matrix QDCT according to equation 8:

$$QDCT = round\left(\frac{DCT(u,v)}{Q(u,v)}\right) \qquad (8)$$

The goal of quantization is to reduce most of the less important high frequency coefficients to zero, we can generate the better the image will compress. the quantization matrix enable the user to decide on quality levels ranging from 1 to 100, where 1 gives the poorest frame quality and highest compression, while 100 gives the best quality and lowest compression. The quantization matrix we used is of quality 50. An example of this step is given in Fig 3, 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.



1243	18	5	2	-3	-1	-6	2
4	1	1	0	3	2	1	2
-19	-5	2	2	-1	2	4	-1
-7	1	-1	0	3	2	-2	2
-6	1	-3	-1	1	1	4	1
14	-1	5	0	-2	1	4	1
24	3	3	0	1	2	12	2
-15	0	-4	-2	0	-1	-6	0

DCT Coefficient

78	2	1	0	0	0	0	0
0	0	0	0	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
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Output

16	11	10	16	124	140	151	161
12	12	14	19	126	158	160	155
14	13	16	24	140	157	169	156
14	17	22	29	151	187	180	162
18	22	37	56	168	109	103	177
24	35	55	64	181	104	113	192
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	199

Quantization table

Fig 3: Computation of the quantized DCT coefficients.

5.4 Reorder The QDCT Block in a Zigzag Sequence

The zigzag is applied to convert the 8x8 block from two dimension into one dimension by taking the value in the upper-left corner of location (0,0) and(0,1) then(1,0) and continue until the location (7,7) from 8x8 block as Fig 4.

5.5 Run-Length Encoding

RLE works by reducing the physical size of a repeating zeros. This repeating zeros, called a run. Each time a zero is encountered in the input data, two values are written to the output file. The first of these values is a zero, a flag to indicate that run-length compression is beginning. The second value is the number of zeros in the run.

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

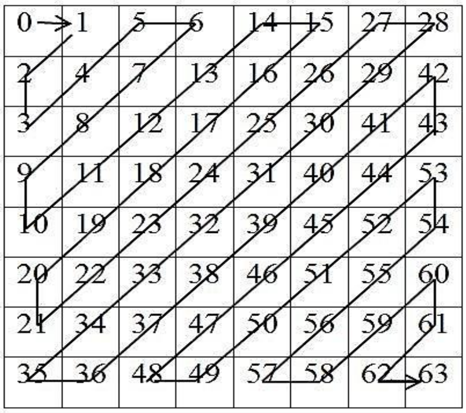
RLC compression: 20,17,0,10,11,-10,-5,0,2,1 ; EOB.

$$DCT(u,v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(\,x,y)\, cos\left[\frac{\pi(2x+1)u}{2N}\right] cos\left[\frac{\pi(2y+1)v}{2N}\right] \qquad (6)$$

$$IDCT(x,y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)DCT(u,v) cos\left[\frac{\pi(2x+1)u}{2N}\right] cos\left[\frac{\pi(2y+1)v}{2N}\right] \qquad (7)$$

for u=0,…..,N-1 and v=0,…,N-1

$$where\ N = 8\ and\ C(k) = \begin{cases} \frac{1}{\sqrt{2}} & for\ k = 0 \\ 1 & otherwise \end{cases}$$



0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

Fig 4: Sequence Zigzag.

5.6 Adaptive Shift Coding

Before shift coding begins, All values of data must be converted to positive. The negative values are multiplied by -2 to become even and positive. While the positive values are multiplied by 2 and then 1 is subtracted from the multiplied value to become odd in order to distinguish them from negative values in the decompression process.

The shift coding step reduces the number of bits required to represent the data after run-length. After run-length, most of the values are between 0 and 15. These values need 4 bits instead of 8 bits to represent them. Therefore, applying shift coding reduces that waste by a measurable amount. According to input values, we used two methods to apply adaptive shift coding.

The first method is applied when most of the values are less than 15. These values will be represented using 4 bits while other values that are larger than 15 will take 10 bits (the first 4 bits is the integer 15 and the other 6 bits will take the value minus 15).

The second method is applied when the number of values larger than 15 is more than the number of values less than 15. In this method, the values that are larger than 15 take 8 bits and the values that are less than 15 take 5 bits. It sets the most significant bit to 1 if the value is less than 15 and gives this value 4 bits and if the value is larger than 15 it sets the most significant bit to 0 and gives 7 bits to the value. The example below shows the shift coding process.

Table 1. Example1 of Shift Coding Methods.

Method 1		Method 2	
Number	**coding**	**Number**	**Coding**
1	0001	1	10001
7	0111	7	10111
0	0000	0	10000
4	0100	4	00100
6	0110	6	10110
70	1111110111	70	01000110
3	0011	3	10011
11	1011	11	11011
2	0100	2	10100
1	0001	1	10001
Total no. of bits	46		53

Table 2. Example2 of Shift Coding Methods.

Method 1		Method 2	
Number	**coding**	**Number**	**Coding**
32	1111010001	32	00100000
7	0111	7	10111
18	1111000011	18	00010010
20	1111000101	20	00010100
6	0110	6	10110
70	1111110111	70	01000110
17	1111000010	17	00010001
11	1011	11	11011
40	1111100011	40	00101000
1	0001	1	10001
Total no. of bits	76		68

The results of applying the first method of shift coding as shown in Table 1 are better than those of the second method because most of the values are less than 15. In table 2, the results of applying the second method was better than those obtained from applying the first method, because the values that are less than 15 are not the most.

According to the input values to the adaptive shift coding, the method to be used is then determined. The criteria of choosing which method to apply is represented as follows:

N = number of values less than 15

M= number of values larger than 15

If (N*4+M*10) is less than (N*5+M*8) then

Select the first method and store a bit value 1 at the most significant bit as an indicator.

Otherwise, select the second method and store a bit value 0 at the most significant bit as an indicator.

6. RESULTS AND DISCUSSIO

The proposed algorithm is tested on nine selected frames as shown in Fig 5. Table 3 shows the results of applying our algorithm compared with the results of applying the standard JPEG method on the same set of frames. Fig 6 shows the decompression for the proposed algorithm. The proposed algorithm was tested on a PC with a 2.50 GHz Core i5 CPU and 4 GB of RAM. Visual Basic 2010 programming language was used to implement the proposed algorithm.

Table 3. Evaluation criteria of the proposed

	Proposed algorithm			JPEG		
no	**CR**	**PSNR**	**Time**	**CR**	**PSNR**	**Time**
1	1:17.6	31.8	0.0086	1:23.25	33.9	0.711
2	1:17.12	32.81	0.0077	1:22	33.82	0.077
3	1:14.12	31.41	0.0073	1:18.7	32.6	0.176
4	1:15.44	32.84	0.0099	1:19.5	35.66	0.291
5	1:15.38	32.65	0.0042	1:21.7	33.51	0.167
6	1:18	35.24	0.0038	1:24.5	35.36	0.099
7	1:10	25.7	0.0098	1:9.2	26	0.399
8	1:10.3	28.5	0.0025	1:8.7	29.4	0.579
9	1:10.8	29.1	0.0074	1:9.1	30.2	0.678

The results in Table 3 indicate a success of our proposed algorithm in the terms of time for all the nine selected frames, this is because we used an adaptive shift coding and this led to a decrease the time comparison with the JPEG method. For CR the JPEG technique produces the best ratio comparing with our technique for the first six frames but when applying the two techniques on frame 7, 8 and 9 the adaptive shift coding give better CR from Huffman coding as showing in Table 3 and the cause for this difference because the Huffman coding depending on the redundancy and the last three frames contain a less ratio of redundancy. It is worth to mention that it was not our intention to produce a high quality frame as we focused on the compression ratio in order to gain fast transmission for the real time video conferencing environment.

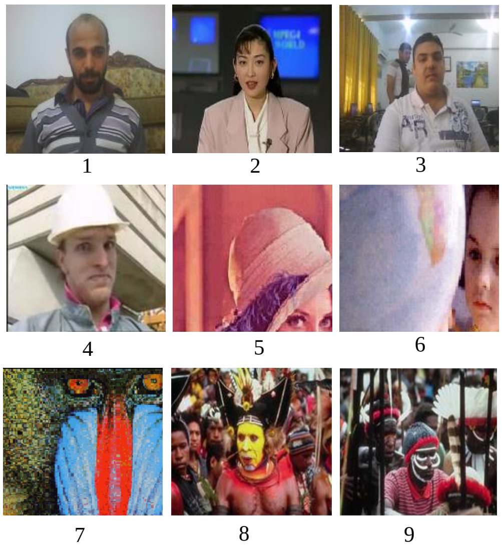


Fig 5: Input Frame.

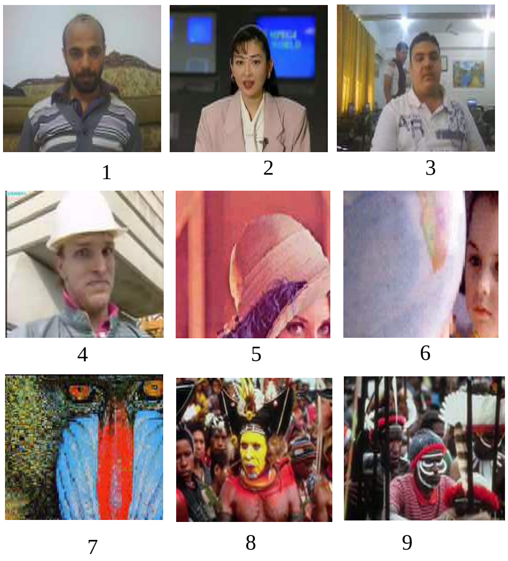


Fig 6: Decompressed Frame of Proposed Algorithm

7. CONCLUSION

In this paper a new intra frame compression method is introduced using proposed adaptive shift coding. For video conference the time is one of the most important issue. The proposed technique reduced the compression and decompression time comparing with Huffman coding which is used in standard JPEG also introduced a good compression ratio with the frames which contain less ratio of redundancy. The compression ratio of the proposed technique will be better with the frames that contain high redundancy and that make it more appropriate for video conference. Also the quality is good comparing with JPEG.

8. REFERENCES

[1] Jayanta Mukhopadhyay, "Image and Video Processing in the Compressed Domain", by Taylor and Francis Group, LLC,2011.

[2] L. Hanzo, P. J. Cherriman and J. Streit, "Video Compression and Communications" , IEEE Communications Society, Sponsor,2011.

[3] Yun Q. Shi and Huifang Sun, " Image and Video Compression for Multimedia Enginerring ", Second Edition `, by Taylor and Francis Group, LLC,2008.

[4] Gregory K. Wallace ," The JPEG Still Picture Compression Standard", Consumer Electronics, IEEE Transactions on, vol. 38, 1992.

[5] Aree A. Mhammad and Loay E. George, "Intraframe Compression Using Lifting Scheme Wavelet-Based Transformation", (JZS) Journal of Zankoy Sulaimani, December2008, 11(1) Part A (53-60).

[6] Alessandro Andreoli, Folco Fioretti, Samuele Pasqualini, Paola Pierleoni and Sandro Tumini, "A Fast Method for H.263 to H.264 IntraSmart-Transcoding",Riunione Annuale 2008 GTTI – Firenze, June 16-18. 2008.

[7] H.B.Kekre, Sanjay R. Sange and Dr. Tanuja K. Sarode, "Data Compression for Video-Conferencing using Halftone and Wavelet Transform", (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 3, pp. 123-130, 2012.

[8] Raymond Westwater and Borko Furht ," Real-Time Video Compression ", Kluwer Academic Publishers, 1997.

[9] Iain E. G. Richardson , " H.264 and MPEG-4 Video Compression", John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester,2003.

[10] David Salomon, "Data Compression", Third Edition, Springer-Verlag New York, Inc,2004.

[11] L. Hanzo, P. J. Cherriman and J. Streit," Video Compression and Communications",Second Edition, IEEE Communications Society, Sponsor,2007.

[12] Suchitra Shrestha, "Hybrid DWT-DCT Algorithm for Image and Video Compression Application", Master thesis, 2010.

[13] S. Katsigiannis, D. Maroulis, and G. Papaioannou, "A GPU based real-time video compression method for video conferencing " 18th IEEE/EURASIP International Conference on Digital Signal Processing (DSP2013), 2013.