International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Implementation of Adaptive Video Compression using Hybridized Wavelet Transform

Anurag Srivastava[1], Puneet Sharma[2]

[1]Department of Computer Science & Engineering, Amity School of Engineering and Technology, Amity University, Uttar Pradesh, India

[2]Assistant Professor, Department of Computer Science & Engineering, Amity School of Engineering and Technology, Amity University, Uttar Pradesh, India

Abstract: ***Videos have been used for years but in current scenario video is one of the most use methods to represent information. In current time videos are not only used for communication and entertainment but also for education point of view, tutorials and lectures are also present in the form of video files. But difficulty with such a powerful media is its huge size, and it contains very large quantity of redundant data. Video holds memory in storage media and bandwidth over a communication medium (wireless or wired). The compression techniques are execute to shrink the size of the video but high compression ratio compromises the quality of the video (low PSNR value) when decompressed and it's also a very lengthy procedure. The challenge is to propose an approach which gives a satisfactory high value of compression ratio and Peak Signal to Noise Ratio (PSNR). The wavelet transform is broadly applied compression technique but it gives a low PSNR value if we use an adaptive hybrid wavelet transformation system. The system uses the 3D-SPIHT (3D Set Partitioning in Hierarchical Trees) which uses the properties of wavelet transformed frames of the video to increase the efficiency, performance. And after that, the proposed system uses the RLE (Run Length Encoding) to add more compression ratio without and gives a better PSNR value which makes it more capable than DCT technique. To examine the values of compression ratio and PSNR can be calculated by simulating the system using MATLAB.***

Keywords: DWT (Discrete Wavelet Transform), SPIHT (Set Partitioning in Hierarchical Trees), 3D-SPIHT, RLE (Run Length Encoding), PSNR (Peak Signal to Noise Ratio)

1. Introduction

A video is organized sequence of video frames or we can say that images, which is an essential part of multimedia it provide entertainment and education both. We can learn things from videos there are tutorials of different courses offered in form of video and now a day's there are online lectures offered by different institutions using video streaming.

But there if a problem with videos is that, it occupies very large amount of bandwidth and storage. Video compression is the process to decrease the size of the video so requirement of storage space can be lower. This huge size of video is because of redundancies present in the data. Video Compression basically reduces the redundancies from the data. Compression means the trimming of data. If after compression we achieve the data without any loss then it is the lossless compression otherwise it is the lossy compression. The wavelet transform is a lossy transformation technique but we can use it in this kind of media (videos) because video is the huge collection of data and some small losses don't affect the overall video or if it affect then this tiny effect over such media can be neglect.

The video compression is entirely different from image compression because a video can be consider as an organized sequence of frames (images), we perform compression on a image straight away but we cannot compress a video without braking it into frames. a variety of compression techniques are present for image compression like EZW, SWT, LZW etc [1], but for video compression there are very few and they gives low compression ratio and low video quality (low PSNR). The broadly applied algorithm used in all video compression techniques is DCT [2].

Unlike image compression in video compression we use steps like motion estimation and video compensation to reduce redundancies and irrelevancies (perceptually unimportant information) and to reduce time complexity. The steps of motion estimation and compensation make compression more efficient and accurate. DCT is the most widely used compression algorithm which is used in most techniques for compression like MPEG and H.246 [1] [6]. Here wavelet transformation [3] is used, which is a multi resolution transformation technique whose properties are used at multiple levels by 3D-SPIHT algorithm which is capable of compress the all 3 color planes of a video frame whether RBG or YCbCr unlike simple SPIHT which can only compress grayscale images. And after that RLE (run length encoding) is used which is a simple compression algorithm which adds efficiency and accuracy to this system, providing more compression ratio without adding complexity and decreasing PSNR value of the video. As the result the values of compression ratio and PSNR can be calculated by simulating the system using MATLAB.

2. DWT

Discrete Wavelet Transform is a multi resolution transform which was developed to outcome the shortcomings of S.T.F.T. (Short Time Fourier Transform). The Discrete Wavelet Transform (DWT) has high energy compaction property which makes it suitable for compression. DWT is the implementation of wavelet transform where signal and wavelets are discrete in time. DWT passes the signal through a low pass filter which yields low resolution signals and a high pass filter which yields difference signals. The outputs

Licensed Under Creative Commons Attribution CC BY

are down-sampled by a fix amount mostly 2 which have the same number of bits as the input signal. And we up-sampled the output of both filters and add them to get reproduced signal.

3. SPIHT

The Embedded Zero-tree Wavelet (EZW) coding is very effective algorithm for compression but it is very complex and difficult to understand. Set Partitioning in Hierarchical Trees (SPIHT) algorithm is an alternative to EZW algorithm which is similar to it but more easy and efficient than it when it comes to compression [4]. SPIHT algorithm uses the sub partitioning sorting algorithm; the coefficients are divided into 4 sub-bands into a multi-resolution pyramid. Then the bands further divided into sub-bands and this decomposition continues until a desired scale is reached.

4. 3D-SPIHT

The existing simulation of SPIHT algorithm in MATLAB is only for greyscale (black & white) image. The 3D-SPIHT algorithm is an extended form of SPIHT which is capable of compressing all plane of a frame of the video, in other words this modified technique can compress a colour video with a good compression ratio and high PSNR. This adapted version of SPIHT algorithm gives more effectiveness, high performance and precise rate control. The implication of precise rate control is that we can control the rate of compression and we can stop the compression at the desire level at selected threshold level.

5. RLE

Run-length encoding is a simple algorithm of compression, which adds more efficiency without raising the complexity. In RLE algorithm we change the successively repeating occurrences of an integer (or any other symbol) by that integer followed by the number of occurrence of the repeating data. First we scan each matrix row wise to identify the repeating integer then group of repetitions replaced by the integer and the frequency of duplicate data. This is done for the whole plane.

6. Methodology

The video compression is the most useful and comparatively new field in data compression if we talk about image compression then we found a numerous research paper and new techniques for it but may be due to complexity, video compression techniques are few and many of them are not so useful as they give a low compression ratio or low video quality (low PSNR value). The proposed methodology gives a high compression ratio with a high PSNR value. As same as any other compression algorithm, the value of PSNR and compression ratio can be different for dissimilar videos. In the paper, rhinos.avi video is use to demonstrate the working of compression system. In the proposed system we hybridize the properties of DWT with the help of 3D-SPIHT and RLE which in future can use in more advance techniques to give more Compression without any loss in the quality of reconstructed (decompressed) video.

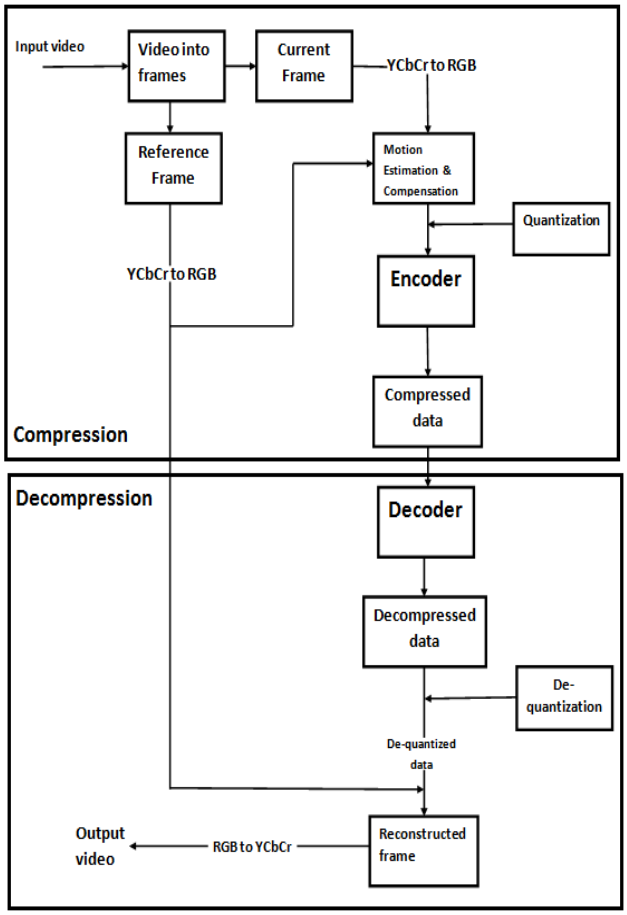

Figure 1: The Block Diagram of the Proposed System

6.1 RGB to YCbCr

RGB image stores the colour in form of RBG that is, Red, Blue and Green planes but YCbCr stores colours in form of Luminance (luma) or brightness and Chrominance (croma) or hue. The human eye is more sensitive to luma information than croma information. The additional compression can be achieved than RBG in YCbCr when we perform more compression on luma plane than on croma planes. Therefore to offer improved compression ratio we convert the RBG planes of the video frames into YCbCr planes. The simulation shows us that the Cb and Cr planes give more compression ratio than Y plane.

Figure 2: Difference between RGB frame and YCbCr frame of video rhinos.avi [5]

6.2 Reference Frame and Current Frame

To perform motion compensation and estimation and compression the video frames can be divided into 3 kinds I, B and P. The 'I' frame called also known as Reference frame which is the first frame of the video and 'B' frame

Licensed Under Creative Commons Attribution CC BY

also known as current frame. For compression the current frame is processed by the reference of Reference Frame, the position of current frame is changing during the compression process. In other words the frame on which compression performed is the Current Frame. To increase performance, the position of reference frame can also be change after a number of frames because after a number of frames the difference of pixels between reference frame and current frame becomes very high that it starts to affects the PSNR value. The new reference frame is the 'P' frame.

6.3 Motion Estimation and Compensation

The objective of Motion Estimation and Motion Compensation is to calculate the motion vectors or in simple words moving pixels, between reference frame and current frame. The video is an organized sequence of frames which are usually alike, means in a video the two or more successive frames can be similar, or all frames of the whole video can be similar to each other. So if we compress the whole frames of the video at once we have to compress redundant data over and over again which is unnecessary and lengthy process. In motion guesstimate and reparation we remove this redundancy so we have to compress only those data which is not similar or we can say that we have to compress only those pixels which move to their previous place. In frame based motion estimation and compensation we calculate difference between reference frame and current frame. Motion guesstimate and reparation is use in both compression and decompression.

A. In Compression: In compression we calculate difference between reference frame and current frame and send the difference matrix or difference frame for compression.

B. In Decompression: After decompression the resultant data is not a video frame it still a difference frame to achieve reconstructed frame we add the reference frame to that difference frame. In following figure the difference frame is shown as a result of motion estimation and motion compensation between Y (luma) plane (1st plane from YCbCr planes) of reference frame and current frame. The current frame is always changing during the entire procedure. This difference frame is the frame on which we perform compression. The advantage of motion estimation and motion compensation step is that it reduces the time complexity of the system and also reduces the size of data even before the compression algorithm applies.

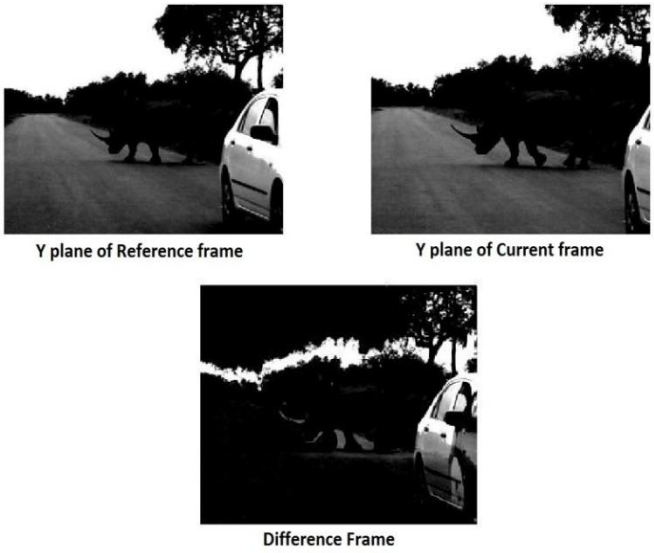

Figure 3: The Difference Frame result of motion estimation and compensation between Y plane of reference and current frame

6.4 Quantization and De-quantization

It is a lossy process because some values cannot recover as previous. In Quantization we perform some mathematical operation on the frame matrices to lower the value in the matrix and to achieve more compression ratio and PSNR. In this system to perform quantization on the difference frame the round value of difference frame is divided by the quantization-scale, it is depend upon planes (for Y its 50% of quantization-scale and for Cb and Cr its double).This quantization-scale if a predefined value (here quantization-scale=2). The quantization decreases the weight of values in the matrix in compression process to save time. Quantization reassigns the values into a direction.

In De-quantization the reverse process of quantization is performed means if we follow the above process of quantization then in de-quantization we have to multiply the quantization-scale into decoded frame which is the result of decompression algorithm.

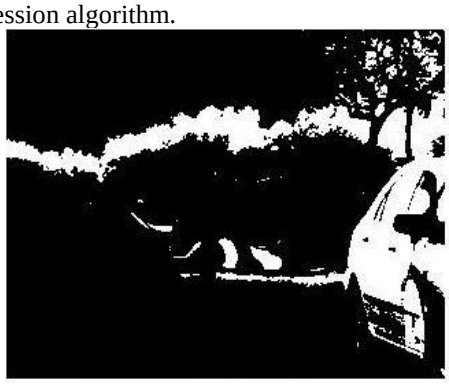

Figure 4: Quantized Frame

In above figure a difference frame is shown after quantization. The black areas in quantized frame have value zero and the other areas (white and shaded) have some value. The intensity of colour white shows the weight of the value of the pixel in the matrix at that area. Quantization decreases the size of difference frame and makes it easy to process.

Licensed Under Creative Commons Attribution CC BY

117	68	0	0	14	0
73	12	146	86	11	0
90	17	111	155	231	140
135	94	11	2	38	54
0	7	46	46	0	0
14	87	124	72	6	44

(a). The part of difference frame matrix from rows 305 to 310 and column 315 to 320, before quantization.

2	1	0	0	0	0
1	0	2	1	0	0
1	0	2	2	4	2
2	1	0	0	1	1
0	0	1	1	0	0
0	1	2	1	0	1

(b). The part of quantized difference frame matrix from rows 305 to 310 and column 315 to 320, after quantization.

Figure 5: Figure shows the difference between frame matrices before and after quantization

In the Figure 5 we can see that how quantization reduce the weight of values, the less the value is, the less number of bits it requires to store.

6.5 Encoder

The Encoder compresses the quantized frame and saves in the form of data (not a video). The output from encoder finally is the compressed video which is sends to decompress in decoder. The proposed system uses a hybrid system 3D-SPIHT and RLE which utilize the properties of wavelet transform. In encoder first the quantized data passes through the 3D-SPIHT algorithm and then through the RLE at different rate of threshold. As the value of Threshold is increases the quality of compressed data is increases but the compression ratio can decrease but it can vary from plane to plane and frame to frame. In the simulation of proposed system the value of threshold is 7. The data have to pass every time through 3D-SPIHT+RLE when threshold changes and it should be done for all planes of every frame of the video.

6.6 Decoder

The decoder decompresses the compressed data and saves in the form of frame. The output from decoder sends to de-quantization and inverse motion estimation and motion compensation. Then the output of motion estimation and compensation further convert to the reconstructed frame. The decoder performs inverse 3D-SPIHT and inverse RLE which uses the properties of inverse wavelet transform. In encoder first the compressed data passes through the inverse RLE algorithm and then through the inverse 3D-SPIHT at different rate of threshold. As the value of Threshold is increases the quality of compressed data is increases but the compression ratio can decreases but it can vary from plane to plane and frame to frame. In other words if we increase the value of threshold the PSNR value increases but the compression ratio of that frame may reduce. The data have to pass through inverse of 3D-SPIHT+RLE every time when threshold changes. We have to remember that the output from decoder is not the final reconstructed frame which we can use as a decompressed video it have to be quantize and added to the reference frame and it has to be done for all frames of the video.

6.7 Overview of the System

The input video breaks into frames and we have to perform the whole compression process on each and every plane of the frame of the video here I explain the working of the propose system only for one plane of the frame of video. The planes of the first frame of video is saves as a reference frame after converting it into YCbCr rest of the planes are compressed with reference of the corresponding planes of reference frame. The plane to be transformed is called current frame and it should be in YCbCr format.

After that we execute motion estimation and compensation. The result of that process is called difference frame. After quantization that difference frame is called quantized frame which is sends to encoder to compression. The output of encoder is a data which cannot be seen as a frame. To perform decompression we pass that data throughout decoder. Afterward we perform de-quantization and add the reference frame to that de-quantized frame in order to perform inverse motion estimation and compensation. Then further transforming the reconstructed fame into RBG we get the decompressed video frame.

7. Result

The proposed system can be simulated on MATLAB. The decompressed video cannot be of same quality as the original there always some quality loss in the frames, which is calculated by PSNR. Fig 6 shows the difference between the original frames and decompressed.

Figure 6: Shows the quality difference between decompressed frame (on left) and original frame (on right) of rhinoes.avi video[5]

For estimating the performance of a compression algorithm we use mainly following matrices which can be calculated with the process of compression simulation on MATLAB.

7.1 PSNR

Peak Signal to Noise Ratio (PSNR) is measure of quality of decompressed frame [2]. It shows the measure of peak error. It tells us how much the video losses its quality after compression and decompression. The PSNR is calculated between the original frame and reconstructed frame. PSNR can be defined as the ratio between maximum size of the frame (R) and Mean Square Error (MSE). PSNR can calculate by the following equation [5].

$$PSNR=10Log_{10}(R^2/MSE)\ldots (1)$$

Mean Square Error (MSE) is the average of cumulative squared error, which can calculated by following equation[5] where $I_1(m,n)$ is the original frame data and $I_2(m,n)$ is the compressed frame data.

$$MSE=\frac{\sum_{M,N}[I_1(m,n)-I_2(m,n)]^2}{M*N}\ldots (2)$$

The value of PSNR is different for every plane of a frame. Following figure is a graph which shows the PSNR value of the respective frame; in it the PSNR values of 15 frames are

Licensed Under Creative Commons Attribution CC BY

demonstrated, the mean value of PSNR of Y, Cb and Cr planes is the PSNR of the frame.

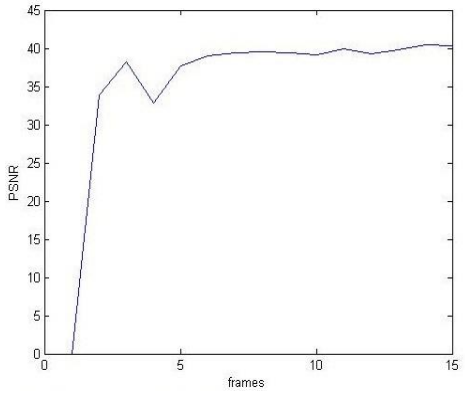

Figure 7: Graph between PSNR and Frames

7.2 Compression Ratio

Compression Ratio tells us that how much the compression performed on the data and the storage space needed by the data is how much reduce. Compression Ration can be defined as the ratio between original frame size and compressed frame size [2].

$$\text{Compression Ratio}=\frac{\text{original frame}}{\text{compressed frame}} \quad \ldots (3)$$

The value of Compression Ratio is different for every plane of a frame. Following figure is a graph which shows the value of the Compression Ratio of the respective frame; in it the compression ratio of 15 frames are demonstrated, the mean value of compression ratio of Y, Cb and Cr planes is the value of compression ratio of the frame. The y axis shows the value of compression ratio and x axis shows the number of frames.

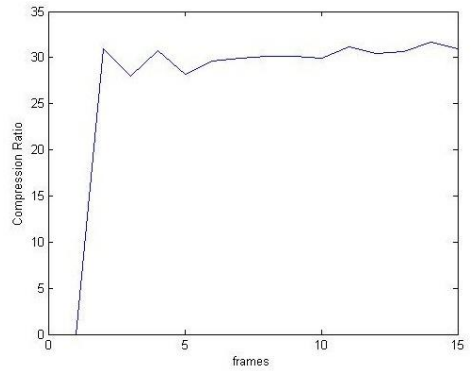

Figure 8: Graph between compression ratio and frames

The proposed system gives compression ratio=28.17(mean of all values of compression ratio) and PSNR=35.93(mean of all values of PSNR) at quantization-scale=2 and threshold=7.

The values presented in above graph can be vary from video to video. In this paper to demonstrate the working of the proposed system we input rhinos.avi video and all values are for this video, if any other video is used then these values may be different.

These values are also change when the quantization-scale and threshold change; high quantization-scale means high Compression ratio but low PSNR and high threshold means high PSNR but low compression ratio. The quantization-scale and threshold are 2 and 7 respectively in simulation of proposed system in MATLAB gives best results and all the values showed in above results are calculated in these values (quantization-scale=2 and threshold=7).

Table 1: Values of Compression ratio with PSNR at different values of quantization-scale and threshold

Values		Compression Ratio	PSNR
Quantization-scale	Threshold		
2	8	22.4745	37.0536
3	8	24.0759	35.5927
2	7	28.1680	35.9252
7	5	29.411	34.3410
8	5	52.0979	25.2963

Above table shows the different values of Compression ratio and PSNR at different values of quantization-scale and threshold. These are the best of various values. The PSNR and Compression ratio, in the table, are the average (mean) values of PSNR and Compression ratio of all planes of the frames.

8. Future Scope

The propose system can work on colour videos and can be use not only on .avi but also on other video file formats. The future scope of this system is that in future this algorithm can use in the video compression techniques as DC is used in MPEG and JPEG etc. This algorithm can design to give more compression ratio in future. And as a future scope this algorithm is not design to work on live video streaming so in future this algorithm could use on live video streaming (video conferences).

9. Conclusion

The video is the most popular form of multimedia but the large size of such media is a difficulty. To decrease the size of video we perform video compression on the video. There are execute compression techniques but most of them is only for image or provide low PSNR and compression ratio. Here in this paper we present a hybrid system. The process is stated as follows: Firstly, to compress a video, input it. And split it into frames. Then in next step the motion estimation and compensation is performed. In succeeding step we perform quantization. And afterward we use wavelet transform as the part of 3D-SPIHT. At last in then we perform additional RLE coding to enhance compression. Then finally we havea compressed video with higher compression ratio. And then we decompress (with inverse motion compensation and de-quantization) and calculate the PSNR value to judge the quality difference between original video and decompressed video. This algorithm can work on colour image and various video file formats.

References

[1] M. Atheeshweri and K. Mahesh, *"Video Compression Techniques- A Comprehensive Survey"* International journal of advance research in Computer Science and Software Engineering, ISSN 2277-128X, volume 4, issue 1, Jan 2014.

Licensed Under Creative Commons Attribution CC BY

[2] L. EscalinTresa and M. Sundararajan, *"Video Compression using Hybrid DWT-DCT algorithm"* International journal of emerging technology and advance engineering, ISSN 2250-2459, volume 4, issue 7, July 2014.
[3] Reza Adhami, *"Video Compression Technique using Wavelet Transform"* IEEE 0-7803-319 6-6, 1996.
[4] Amir Said and William A. Pearlman, *"A New, Fast and Efficient Image Codec based on Set Partitioning in Hierarchical Trees"* IEEE Transactions on circuits and system for video technology, volume 6, issue 3, June 1996.
[5] Source, MATLAB R2010a, version 7.10.0.499, 32 bit, Feb. 2010
[6] ParamjeetKaur, Sugandha Sharma and Satindar Pal Singh Ahuja, *"Latest Video Compression Standard H.264 within Video Survillance",* International journal of advance research in Computer Science and Software Engineering, ISSN 2250-2459, volume 2, issue 1, Jan 2012.
[7] Muhammad AkifShaikh andSagar S. Badnerkar, *"Video Compression Algorithm Using Motion Compensation Technique: A Survey",* International journal of advance research in Computer Science and Management studies, ISSN 2321-7782, volume 2, issue 3, March 2014.
[8] Rajeshwar Das, Lalit Singh andSandeepKaushik, *"Video Compression Technique",* International journal of Scientific & Technology Research, ISSN 2277-8616, volume 1, issue 10, Nov 2012.
[9] K. S. Thyagarajan, *"Still Image and Video Compression Using MATLAB"* John Wiley & Sons Publications, 2011, pp. 359-411
[10] Sangeeta Mishra andSudheerSavarkar, *"Video Compression using EZW and FBSM"* International journal of scientific and research publication, ISSN 2250-3153, volume 2, issue 10,October 2012.
[11] Komal Gupta and Ram LautanVerma, *"Minimum Entropy Base Lossless Image Compression using Predictive coding and Integer Wavelet Transform"* International journal of engineering science and innovative technology, ISSN 2319-5967, volume 2, issue 4, July 2013.
[12] Thazni Aziz andD.Raveena Judie Dolly, *"Motion Estimation and Motion Compensated Video Compression using DCT and DWT"* International Journal of Emerging Technology and Advance Engineering, ISSN 2250-2459, volume 2, issue 12, December 2012.
[13] Deepa Mary Thomas, SubhaVarier, *"A Novel Based Approach for Finding Motion Estimation in Video Compression"* International Journal of Advance Research in Computer and Communication Engineering, ISSN 2250-2459, volume 1, issue 8, October 2012.
[14] A.M. Raid, W.M. Khedr and Wesam Ahmed *"Image Compression using Embedded ZeroTree Wavelet"* Signal &Image processing: An International Journal (SIPIJ), volume 5, issue 6, Dec 2014.
[15] E.Kannan andG.Murugan, *"Lossless Image Compression Algorithm for Transmitting over Low Bandwidth Line"* International Journal of Advance Research in Computer Science and Software Engineering, ISSN 2277-128x, volume 2, issue 2, February 2012.
[16] PallaviM.Sune and VijayaK.Shandilya, *"Image Compression Technique based on Wavelet and Huffman coding"* International Journal of Advance Research in Computer Science and Software Engineering, ISSN 2277-128x, volume 3, issue 4, April 2013.
[17] DalvirKaurandKamaljitKaur,*"Huffman Based LZW Lossless Image Compression using Retinex Algorithm",* International Journal of Advance Research in Computer and Communication Engineering, volume 2, issue 8, August 2013.
[18] K.R. Namuduri and V.N. Ramaswamy, "*Feature preserving image compression*" in Pattern Recognition letters, vol. 24, no 15, pp. 2767-2776, Nov.2003.
[19] D.Khosla and A.Kaur, "*Design of Hybrid Compression Model using DWT-DCT-Huffman Algorithms for Compression of Bit Stream*", International Journal of Engineering Research &Technology (IJERT) Vol. 1 Issue 5, July – 2012
[20] Huang Ke-kun, "*Improved Set Partitioning in Hierarchical Trees Algorithm Based on Adaptive Coding Order*", Journal of ComputerApplication-china, 2012, 32(3):732-735, March 2012.

Licensed Under Creative Commons Attribution CC BY