Editor in Chief Professor Hu, Yu-Chen

International Journal of Image Processing (IJIP)

Book: 2009 Volume 3, Issue 1
Publishing Date: 28-02-2009
Proceedings
ISSN (Online): 1985-2304

IJIP Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers

Table of Contents

Volume 3, Issue 1, January/February 2009.

Pages

Study and Comparison of Various Image Edge Detection Techniques

Raman Maini research_raman@yahoo.com
Reader
Punjabi University
Patiala-147002(Punjab), India

Dr. Himanshu Aggarwal himagrawal@rediffmail.com
Reader,
Punjabi University
Patiala-147002(Punjab), India

ABSTRACT

Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Image Edge detection significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. Since edge detection is in the forefront of image processing for object detection, it is crucial to have a good understanding of edge detection algorithms. In this paper the comparative analysis of various Image Edge Detection techniques is presented. The software is developed using MATLAB 7.0. It has been shown that the Canny's edge detection algorithm performs better than all these operators under almost all scenarios. Evaluation of the images showed that under noisy conditions Canny, LoG(Laplacian of Gaussian), Robert, Prewitt, Sobel exhibit better performance, respectively. 1. It has been observed that Canny's edge detection algorithm is computationally more expensive compared to LoG(Laplacian of Gaussian), Sobel, Prewitt and Robert's operator.

Keywords: Edge Detection, Noise, Digital Image Processing

1. INTRODUCTION

Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Classical methods of edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. There are an extremely large number of edge detection operators available, each designed to be sensitive to certain types of edges. Variables involved in the selection of an edge detection operator include Edge orientation, Noise environment and Edge structure. The geometry of the operator determines a characteristic direction in which it is most sensitive to edges. Operators can be optimized to look for horizontal, vertical, or diagonal edges. Edge detection is difficult in noisy images, since both the noise and the edges contain high-frequency content. Attempts to reduce the noise result in blurred and distorted edges. Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. This

[1] Raman Maini is corresponding author with e-mail address research_raman@yahoo.com , Mobile no +91-9779020951 and address as Reader, University College of Engineering, Punjabi University, Patiala-147002 (India), Fax No. +91-1753046324

results in less accurate localization of the detected edges. Not all edges involve a step change in intensity. Effects such as refraction or poor focus can result in objects with boundaries defined by a gradual change in intensity [1]. The operator needs to be chosen to be responsive to such a gradual change in those cases. So, there are problems of false edge detection, missing true edges, edge localization, high computational time and problems due to noise etc. Therefore, the objective is to do the comparison of various edge detection techniques and analyze the performance of the various techniques in different conditions.

There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

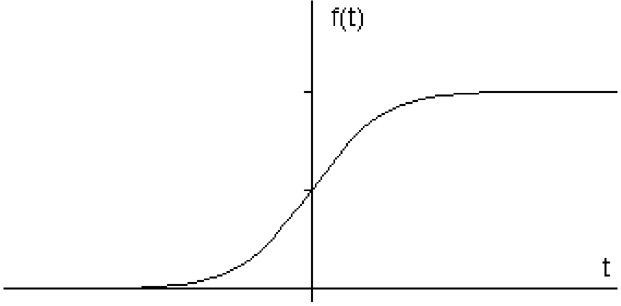
Gradient based Edge Detection:

The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

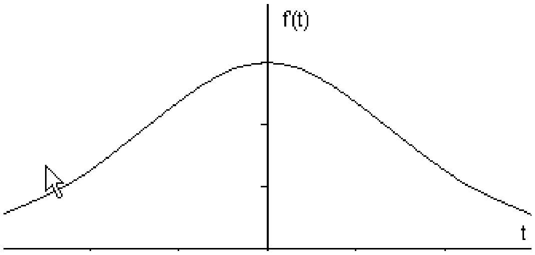
Laplacian based Edge Detection:

The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location. Suppose we have the following signal, with an edge shown by the jump in intensity below:

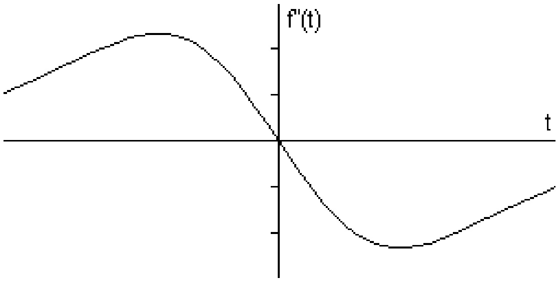
Suppose we have the following signal, with an edge shown by the jump in intensity below:



If we take the gradient of this signal (which, in one dimension, is just the first derivative with respect to t) we get the following:



Clearly, the derivative shows a maximum located at the center of the edge in the original signal. This method of locating an edge is characteristic of the “gradient filter” family of edge detection filters and includes the Sobel method. A pixel location is declared an edge location if the value of the gradient exceeds some threshold. As mentioned before, edges will have higher pixel intensity values than those surrounding it. So once a threshold is set, you can compare the gradient value to the threshold value and detect an edge whenever the threshold is exceeded [2]. Furthermore, when the first derivative is at a maximum, the second derivative is zero. As a result, another alternative to finding the location of an edge is to locate the zeros in the second derivative. This method is known as the Laplacian and the second derivative of the signal is shown below:



In this paper we analyzed and did the visual comparison of the most commonly used Gradient and Laplacian based Edge Detection techniques. In section 2 the problem definition is presented. In section 3 the various edge detection techniques have been studied and analyzed. In section 4 the visual comparisons of various edge detection techniques have been done by developing software in MATLAB 7.0. Section 5 discusses the advantages and disadvantages of various edge detection techniques. Section 6 discusses the conclusion reached by analysis and visual comparison of various edge detection techniques developed using MATLAB 7.0.

2. PROBLEM DEFINITION

There are problems of false edge detection, missing true edges, producing thin or thick lines and problems due to noise etc. In this paper we analyzed and did the visual comparison of the most commonly used Gradient and Laplacian based Edge Detection techniques for problems of inaccurate edge detection, missing true edges, producing thin or thick lines and problems due to noise etc. The software is developed using MATLAB 7.0

3. Edge Detection Techniques

3.1 Sobel Operator

The operator consists of a pair of 3×3 convolution kernels as shown in Figure 1. One kernel is simply the other rotated by 90°.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

FIGURE 1: Masks used by Sobel Operator

These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these *Gx* and *Gy*). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient [3]. The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

Typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy|$$

which is much faster to compute.

The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$$\theta = \arctan(Gy / Gx)$$

3.2 Robert’s cross operator:

The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. The operator consists of a pair of 2×2 convolution kernels as shown in Figure 2. One kernel is simply the other rotated by 90°[4]. This is very similar to the Sobel operator.

+1	0
0	-1

Gx

0	+1
-1	0

Gy

FIGURE 2: Masks used for Robert operator.

These kernels are designed to respond maximally to edges running at 45° to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these *Gx* and *Gy*). These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

although typically, an approximate magnitude is computed using:

$$|G| = |Gx| + |Gy|$$

which is much faster to compute.

The angle of orientation of the edge giving rise to the spatial gradient (relative to the pixel grid orientation) is given by:

$$\theta = \arctan(Gy / Gx) - 3\pi / 4$$

3.3 Prewitt’s operator:

Prewitt operator [5] is similar to the Sobel operator and is used for detecting vertical and horizontal edges in images.

-1	0	+1
-1	0	+1
-1	0	+1

Gx

+1	+1	+1
0	0	0
-1	-1	-1

Gy

FIGURE 3: Masks for the Prewitt gradient edge detector

3.4 Laplacian of Gaussian:

The Laplacian is a 2-D isotropic measure of the 2nd spatial derivative of an image. The Laplacian of an image highlights regions of rapid intensity change and is therefore often used for edge detection. The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian Smoothing filter in order to reduce its sensitivity to noise. The operator normally takes a single gray level image as input and produces another gray level image as output.

The Laplacian *L(x,y)* of an image with pixel intensity values *I(x,y)* is given by:

$$L(x,y)= \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian[5]. Three commonly used small kernels are shown in Figure 4.

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

FIGURE 4. Three commonly used discrete approximations to the Laplacian filter.

Because these kernels are approximating a second derivative measurement on the image, they are very sensitive to noise. To counter this, the image is often Gaussian Smoothed before applying the Laplacian filter. This pre-processing step reduces the high frequency noise components prior to the differentiation step.

In fact, since the convolution operation is associative, we can convolve the Gaussian smoothing filter with the Laplacian filter first of all, and then convolve this hybrid filter with the image to achieve the required result. Doing things this way has two advantages: Since both the Gaussian and the Laplacian kernels are usually much smaller than the image, this method usually requires far fewer arithmetic operations.

The LoG (`Laplacian of Gaussian')[6] kernel can be pre-calculated in advance so only one convolution needs to be performed at run-time on the image.

The 2-D LoG function [7] centered on zero and with Gaussian standard deviation σ has the form:

$$\mathrm{LoG}(x,y) = -1/\pi\sigma^4\left[\ 1-\ \left(\frac{x^2+y^2}{2\sigma^2}\right)\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

and is shown in Figure 5.

0	1	0
1	-4	1
0	1	0

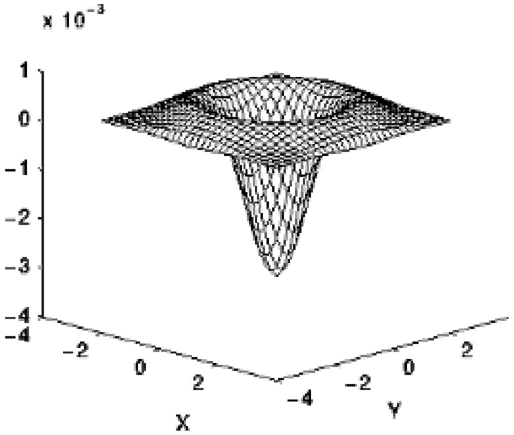


FIGURE 5. The 2-D Laplacian of Gaussian (LoG) function. The *x* and *y* axes are marked in standard deviations (σ).

A discrete kernel that approximates this function (for a Gaussian σ = 1.4) is shown in Figure 6.

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

FIGURE 6. Discrete approximation to LoG function with Gaussian σ = 1.4.

Note that as the Gaussian is made increasingly narrow, the LoG kernel becomes the same as the simple Laplacian kernels shown in figure 4. This is because smoothing with a very narrow Gaussian (σ < 0.5 pixels) on a discrete grid has no effect. Hence on a discrete grid, the simple Laplacian can be seen as a limiting case of the LoG for narrow Gaussians [8]-[10].

3.5 Canny Edge Detection Algorithm

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's intentions were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection"[11]. In his paper, he followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be no responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge. This was implemented because the first two were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (nonmaximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2.

Step 1:-

In order to implement the canny edge detector algorithm, a series of steps must be followed. The first step is to filter out any noise in the original image before trying to locate and detect any edges. And because the Gaussian filter can be computed using a simple mask, it is used exclusively in the Canny algorithm. Once a suitable mask has been calculated, the Gaussian smoothing can be performed using standard convolution methods. A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise. The localization error in the detected edges also increases slightly as the Gaussian width is increased.

Step 2:-

After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image. The Sobel operator performs a 2-D spatial gradient measurement on an image. Then, the approximate absolute gradient magnitude (edge strength) at each point can be found. The Sobel operator [3] uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown below:

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

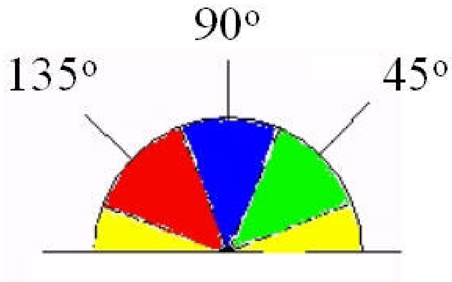
The magnitude, or edge strength, of the gradient is then approximated using the formula:
$|G| = |Gx| + |Gy|$

Step 3:-
The direction of the edge is computed using the gradient in the x and y directions. However, an error will be generated when sumX is equal to zero. So in the code there has to be a restriction set whenever this takes place. Whenever the gradient in the x direction is equal to zero, the edge direction has to be equal to 90 degrees or 0 degrees, depending on what the value of the gradient in the y-direction is equal to. If GY has a value of zero, the edge direction will equal 0 degrees. Otherwise the edge direction will equal 90 degrees. The formula for finding the edge direction is just:
Theta = invtan (Gy / Gx)

Step 4:-
Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image. So if the pixels of a 5x5 image are aligned as follows:

x	x	x	x	x
x	x	x	x	x
x	x	a	x	x
x	x	x	x	x
x	x	x	x	x

Then, it can be seen by looking at pixel "a", there are only four possible directions when describing the surrounding pixels - 0 degrees (in the horizontal direction), 45 degrees (along the positive diagonal), 90 degrees (in the vertical direction), or 135 degrees (along the negative diagonal). So now the edge orientation has to be resolved into one of these four directions depending on which direction it is closest to (e.g. if the orientation angle is found to be 3 degrees, make it zero degrees). Think of this as taking a semicircle and dividing it into 5 regions.



Therefore, any edge direction falling within the yellow range (0 to 22.5 & 157.5 to 180 degrees) is set to 0 degrees. Any edge direction falling in the green range (22.5 to 67.5 degrees) is set to 45 degrees. Any edge direction falling in the blue range (67.5 to 112.5 degrees) is set to 90 degrees. And finally, any edge direction falling within the red range (112.5 to 157.5 degrees) is set to 135 degrees.

Step 5:-
After the edge directions are known, non-maximum suppression now has to be applied. Non-maximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.

Step 6:-
Finally, hysteresis[12] is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold. If a single threshold, T1 is

applied to an image, and an edge has an average strength equal to T1, then due to noise, there will be instances where the edge dips below the threshold. Equally it will also extend above the threshold making an edge look like a dashed line. To avoid this, hysteresis uses 2 thresholds, a high and a low. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T2 to start but you don't stop till you hit a gradient below T1.

4. Visual Comparison of various edge detection Algorithms

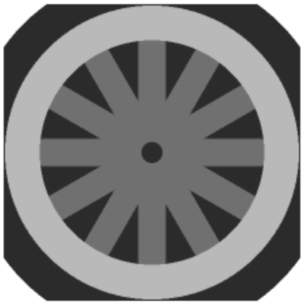


FIGURE 7: Image used for edge detection analysis (wheel.gif)

Edge detection of all four types was performed on Figure 7[13]. Canny yielded the best results. This was expected as Canny edge detection accounts for regions in an image. Canny yields thin lines for its edges by using non-maximal suppression. Canny also utilizes hysteresis with thresholding.

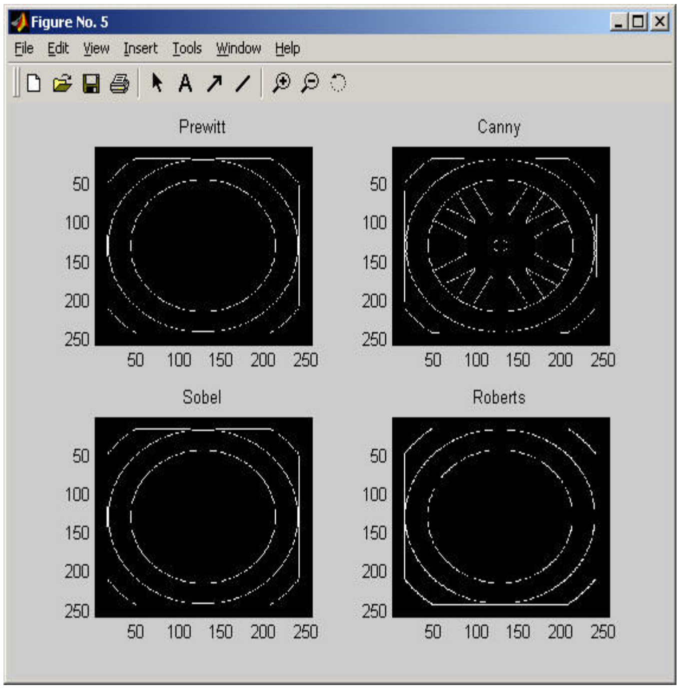


FIGURE 8: Results of edge detection on Figure 7. Canny had the best results

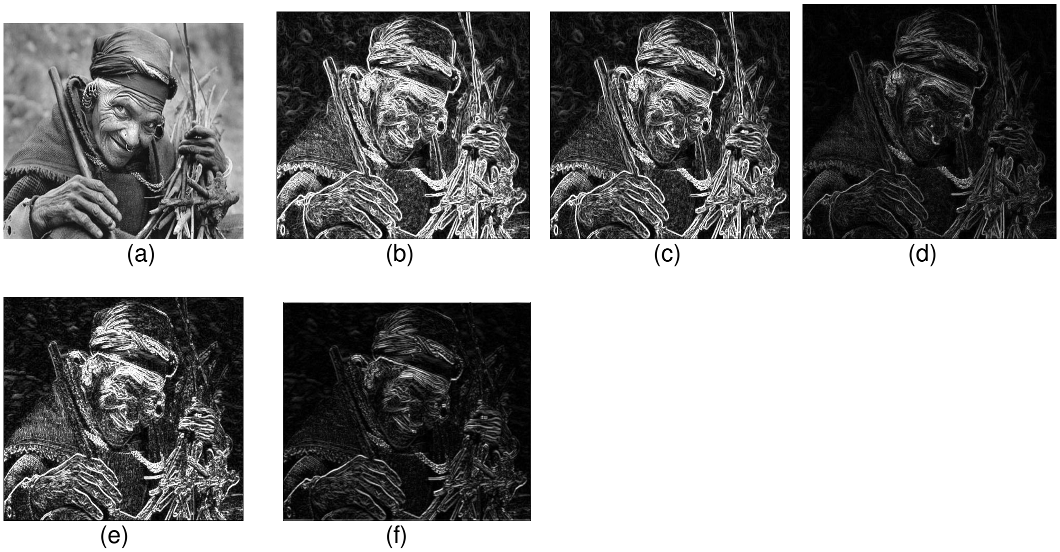


FIGURE 9: Comparison of Edge Detection Techniques Original Image (b) Sobel (c) Prewitt (d) Robert (e) Laplacian (f) Laplacian of Gaussian

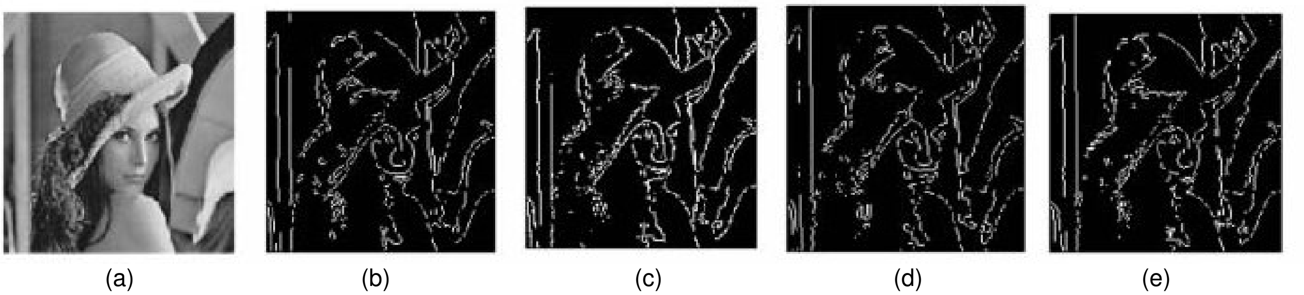


FIGURE 10: Comparison of Edge Detection Techniques on Lena Image Original Image (b) Canny Method (c) Roberts Edges (d) LOG edges (e) Sobel

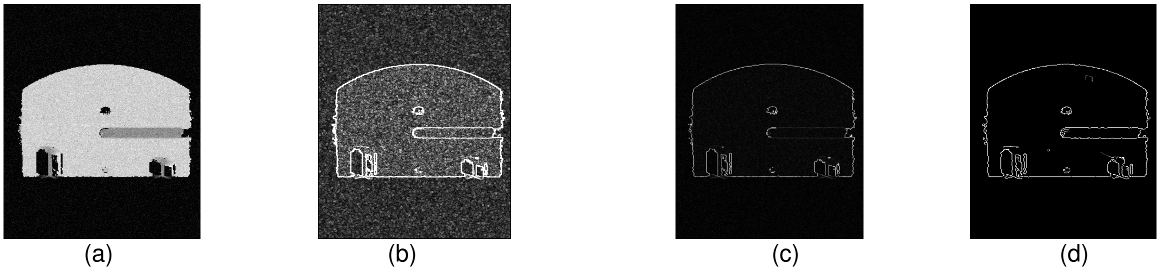


FIGURE 11: Comparison of Edge Detection technique on Noisy Image (a) Original Image with Noise (b) Sobel (c) Robert (d) Canny

5. Advantages and Disadvantages of Edge Detector

As edge detection is a fundamental step in computer vision, it is necessary to point out the true edges to get the best results from the matching process. That is why it is important to choose edge detectors that fit best to the application. In this respect, we first present some advantages and disadvantages of Edge Detection Techniques [13]-[21] with in the context of our classification in Table 1.

Operator	Advantages	Disadvantages
Classical (Sobel, prewitt, Kirsch,…)	Simplicity, Detection of edges and their orientations	Sensitivity to noise, Inaccurate
Zero Crossing(Laplacian, Second directional derivative)	Detection of edges and their orientations. Having fixed characteristics in all directions	Responding to some of the existing edges, Sensitivity to noise
Laplacian of Gaussian(LoG) (Marr-Hildreth)	Finding the correct places of edges, Testing wider area around the pixel	Malfunctioning at the corners, curves and where the gray level intensity function varies. Not finding the orientation of edge because of using the Laplacian filter
Gaussian(Canny, Shen-Castan)	Using probability for finding error rate, Localization and response. Improving signal to noise ratio, Better detection specially in noise conditions	Complex Computations, False zero crossing, Time consuming

Table 1: Some Advantages and Disadvantages of Edge Detectors

6. CONCLUSIONS

Since edge detection is the initial step in object recognition, it is important to know the differences between edge detection techniques. In this paper we studied the most commonly used edge detection techniques of Gradient-based and Laplacian based Edge Detection. The software is developed using MATLAB 7.0.

Gradient-based algorithms such as the Prewitt filter have a major drawback of being very sensitive to noise. The size of the kernel filter and coefficients are fixed and cannot be adapted to a given image. An adaptive edge-detection algorithm is necessary to provide a robust solution that is adaptable to the varying noise levels of these images to help distinguish valid image contents from visual artifacts introduced by noise.

The performance of the Canny algorithm depends heavily on the adjustable parameters, σ, which is the standard deviation for the Gaussian filter, and the threshold values, ‘T1’ and ‘T2’. σ also controls the size of the Gaussian filter. The bigger the value for σ, the larger the size of the Gaussian filter becomes. This implies more blurring, necessary for noisy images, as well as detecting larger edges. As expected, however, the larger the scale of the Gaussian, the less accurate is the localization of the edge. Smaller values of σ imply a smaller

Gaussian filter which limits the amount of blurring, maintaining finer edges in the image. The user can tailor the algorithm by adjusting these parameters to adapt to different environments.

Canny's edge detection algorithm is computationally more expensive compared to Sobel, Prewitt and Robert's operator. However, the Canny's edge detection algorithm performs better than all these operators under almost all scenarios. Evaluation of the images showed that under noisy conditions, Canny, LoG, Sobel, Prewitt, Roberts's exhibit better performance, respectively.

• Acknowledgement

The authors are greatly indebted to the Department of Computer Engineering, University College of Engineering, Punjabi University, Patiala for providing excellent lab facilities that make this work possible.

• References

1. E. Argyle. "Techniques for edge detection," Proc. IEEE, vol. 59, pp. 285-286, 1971

2. F. Bergholm. "Edge focusing," in Proc. 8th Int. Conf. Pattern Recognition, Paris, France, pp. 597- 600, 1986

3. J. Matthews. "An introduction to edge detection: The sobel edge detector," Available at http://www.generation5.org/content/2002/im01.asp, 2002.

4. L. G. Roberts. "Machine perception of 3-D solids" ser. Optical and Electro-Optical Information Processing. MIT Press, 1965 .

5. R. C. Gonzalez and R. E. Woods. "Digital Image Processing". 2nd ed. Prentice Hall, 2002.

6. V. Torre and T. A. Poggio. "On edge detection". IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-8, no. 2, pp. 187-163, Mar. 1986.

7. E. R. Davies. "Constraints on the design of template masks for edge detection". Partern Recognition Lett., vol. 4, pp. 11 1-120, Apr. 1986.

8. W. Frei and C.-C. Chen. "Fast boundary detection: A generalization and a new algorithm ". IEEE Trans. Comput., vol. C-26, no. 10, pp. 988-998, 1977.

9. W. E. Grimson and E. C. Hildreth. "Comments on Digital step edges from zero crossings of second directional derivatives''. IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-7, no. 1, pp. 121-129, 1985.

10. R. M. Haralick. "Digital step edges from zero crossing of the second directional derivatives," IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-6, no. 1, pp. 58-68, Jan. 1984.

11. J. F. Canny. "A computational approach to edge detection". IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-8, no. 6, pp. 679-697, 1986

12 J. Canny. "Finding edges and lines in image". Master's thesis, MIT, 1983.

13. R. A. Kirsch. "Computer determination of the constituent structure of biomedical images". Comput. Eiorned. Res., vol. 4, pp. 315-328, 1971.

14. M. H. Hueckel. " A local visual operator which recognizes edges and line". J. ACM, vol. 20, no. 4, pp. 634-647, Oct. 1973.

15. Y. Yakimovsky, "Boundary and object detection in real world images". JACM, vol. 23, no. 4, pp. 598-619, Oct. 1976

16. A. Yuille and T. A. Poggio . "Scaling theorems for zero crossings". IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-8, no. 1, pp. 187-163, Jan. 1986.

17. D. Marr and E.Hildreth. "Theory of Edge Detection". Proceedings of the Royal Society of London. Series B, Biological Sciences,, Vol. 207, No. 1167. (29 February 1980), pp. 187-217

18. M. Heath, S. Sarkar, T. Sanocki, and K.W. Bowyer. "A Robust Visual Method for Assessing the Relative. Performance of Edge Detection Algorithms". IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 12, pp. 1338-1359, Dec. 1997

19. M. Heath, S. Sarkar, T. Sanocki, and K.W. Bowyer. "Comparison of Edge Detectors: A Methodology and Initial Study ". Computer Vision and Image Understanding, vol. 69, no. 1, pp. 38-54 Jan. 1998

20. M.C. Shin, D. Goldgof, and K.W. Bowyer ."Comparison of Edge Detector Performance through Use in an Object Recognition Task". Computer Vision and Image Understanding, vol. 84, no. 1, pp. 160-178, Oct. 2001.

21. T. Peli and D. Malah. "A Study of Edge Detection Algorithms". Computer Graphics and Image Processing, vol. 20, pp. 1-21, 1982.

Soft Decision Scheme for Multiple Descriptions Coding over Rician Fading Channels

A. H. M. Almawgani and M. F. M. Salleh almawgani2003@hotmail.com
School of Electrical and Electronic,
Universiti Sains Malaysia, 14300 Nibong
Tebal,PulauPinang,Malaysia fadzlisalleh@eng.usm.my

ABSTRACT

This paper presents a new MDC scheme for robust wireless data communications. The soft detection making of the MDC scheme utilises the statistical received data error obtained from channel decoding. The coded bit stream in the system is protected using either the Reed Solomon (RS) or Low Density Parity Check Codes (LDPC) channel coding scheme. Simulation results show that this system has some significant performance improvements over the single description or single channel transmission systems in terms of symbol error rate and peak signal-to-noise ratio PSNR. The system with RS codes is 2 to 5 dB better than single description. The system with LDPC channel codes is 6 to10 dB better than the single description.

Keywords: Multiple Descriptions Coding, Rician fading channel and Image transmission.

1. INTRODUCTION

In recent years, many researchers have involved in the development of Multiple Descriptions Coding (MDC) techniques which improves the robustness of wireless data transmission as presented in [1-9]. In MDC technique, retransmission of the lost information is not required since the receive data are sufficient for quality reconstruction. Retransmission often incurs unacceptable delay. This makes MDC particularly appealing for real-time interactive multimedia applications such as multimedia communication for mobile and video conferencing.

There have been lot of research activities done for applications in multimedia communication. For example, in [1-2] present the works done for image coding, in [3-4] present the works done for video coding, and in [5-7] present the works done for audio as well as speech coding. Other related works to MDC technique include the use of quantization such as MD scalar quantizers (MDSQ) as presented in [8] and the work that involves transformation technique, such as MD Transform Coding (MDTC) as presented in [9].

In this paper, a new MDC coding scheme is introduced where the soft decision making of the reconstructed image is based on the statistical data error channel decoding. An image transmission system that consists of the new MDC scheme and channel coding scheme is simulated over wireless channels. The coded bit stream is protected either using the RS or LDPC as well as the hybrid of RS and LDPC codes. Results for the system with RS codes are 2 to 5 dB better than single description. Results for the system with LDPC codes are 6 to 10 dB better than the single description.

a. Proposed MDC scheme

The proposed image transmission system that consists of the new MDC scheme and channel coding is illustrated in Fig. 1.

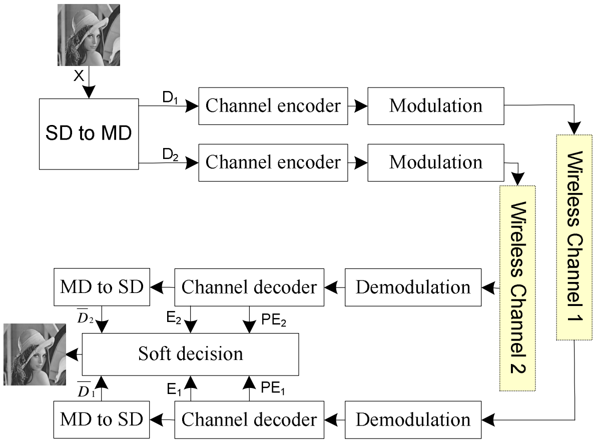


FIGURE 1: A Multiple Description System

An image is first divided into several packets, where the size of each packet is determined by size of the image row. At this stage the data are considered single description (SD) data. Then, each element $X(i, j)$ in SD packet is converted to a pair of elements ($D_1(i, j)$, $D_2(i, j)$). At this stage the data are considered as multiple descriptions (MD) data. The conversion formula used is as the following;

$$D_1(i, j) = \frac{X(i, j)}{2} + 1 \quad 1$$

$$D_2(i, j) = \frac{X(i, j) + 1}{2} \quad 2$$

This can be considered as combining the neighbouring data (pixels) of the original data as shown in Table 1.

Table 1: Single to Multiple Descriptions

$X(i, j)$	1	2	3	4	5	6	7	8	9	10	…
$D_1(i, j)$	1	2		3		4		5		6	…
$D_2(i, j)$	1		2		3		4		5		…

The MD data sequences are then coded using either the RS codes or using the LDPC codes. Both schemes use the same code rate of 1/3. Then, the encoded data are transmitted over wireless channels after BPSK modulation process.

At the receiver, the data are first demodulated by the BPSK demodulator. Then, the data are decoded using either the RS or LDPC decoder. The channel decoder has three outputs i.e. the decoded MD data, sum of error in the packet (channel 1 error $E_1(i)$ or channel 2 error $E_2(i)$), and symbol pairs $R_k(i, j)$. Then the MD data are converted to a SD data. The next process is the conversion of MD to SD data. This is the inverse of SD to MD process given by equation 3 and 4.

$$\overline{D}_1(i, j) = 2*(R_1(i, j) - 1) \quad 3$$

$$\overline{D}_2(i, j) = 2*R_1(i, j) - 1 \quad 4$$

$$\overline{D}_0(i,:) = \overline{D}_1(i,:) \qquad if \quad E_1(i) = 0 \quad 5$$

$$\overline{D}_0(i,:) = \overline{D}_2(i,:) \qquad if \quad E_2(i) = 0 \quad 6$$

The final process is the soft decision algorithm, where it takes the best data out of the two channels. There are three possible inputs to this process;

The best packet will be received from channel 1, if no error occurs during transmission of the whole packet as stated by equation 5.

The best packet will be received from channel 2, if no error occurs during transmission of the whole packet as stated by equation 6.

If there are errors in each channel, the following steps are taken;

a. If $E_1(i) > E_2(i)$: Choose packet from channel 2.
b. If $E_1(i) < E_2(i)$: Choose packet from channel 1.
c. If $E_1(i) = E_2(i)$: Compare each component from both packets, and the check the pixel error PE_1 and PE_2 values. Choose the component where the PE_i of zero value.

The entire process is summarised by the flow chart shown in Fig. 2.

Simulation Results

This section presents the simulation results of the system using two different channel coding schemes. The first scheme uses RS channel coding and the transmission is carried out over AWGN channel. The channel condition is varied by changing the signal to noise ration (SNR) for each transmission. Then, a different channel i.e. the Rician fading channel is used. The second scheme uses LDPC codes as channel coding. The standard image "Lena" is used as the test image. The performance is measured based on the Peak Signal to Noise Ration (PSNR) that serves as the quantitative measure of the quality for the reconstructed images.

	SNR(dB)
	PSNR(dB)
		1	4	7	10

Use RS channel codes	Channel 1	10.37	16.31	34.08	∞
	Channel 2	10.56	16.87	31.59	∞
	Combine between 1&2	10.56	18.81	41.56	∞
Use LDPC channel codes	Channel 1	∞	∞	∞	∞
	Channel 2	∞	∞	∞	∞
	Combine between 1&2	∞	∞	∞	∞

FIGURE 2: Performance in AWGN channel

Table 2 shows the results obtained for image transmission system over AWGN channel. The system consists of the new MDC algorithm and two channel coding schemes. The PSNR performance of the reconstructed image is obtained for channel SNR from 1 dB to 10dB. This results show that the system with LDPC performs better than RS codes at SNR lower than 10 dB.

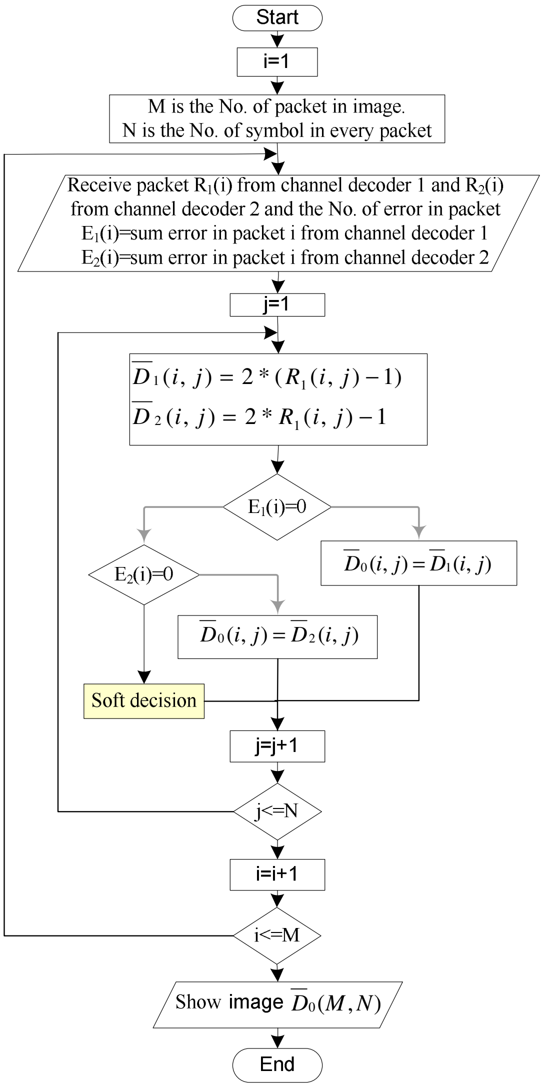

Start
i=1
M is the No. of packet in image.
N is the No. of symbol in every packet
Receive packet R_1(i) from channel decoder 1 and R_2(i) from channel decoder 2 and the No. of error in packet
E_1(i)=sum error in packet i from channel decoder 1
E_2(i)=sum error in packet i from channel decoder 2
j=1
$\overline{D}_1(i, j) = 2*(R_1(i, j)-1)$
$\overline{D}_2(i, j) = 2*R_1(i, j)-1$
E_1(i)=0
E_2(i)=0
$\overline{D}_0(i, j) = \overline{D}_1(i, j)$
$\overline{D}_0(i, j) = \overline{D}_2(i, j)$
Soft decision
j=j+1
j<=N
i=i+1
i<=M
Show image $\overline{D}_0(M, N)$
End

FIGURE 3: flow chart of MD to SD

Fig. 3 shows the results obtained for image transmission system over Rician fading channel. In this simulation, a mobile system with the source rate of 100 kbps and carrier frequency of 900 MHz are assumed. The mobile speed of 120 km/h is used that gives the Maximum Doppler shift as 100 Hz. The AWGN environment is assumed to be with every path in the Rician fading channel. In the simulation, the frequency-selective “multiple paths” Rician channel uses the K-factor equal 10 (K-factor defined as the ratio of signal power for line of sight (LOS) over the scattered, reflected power).

This results show the performance of image transmission system with LDPC codes is better than RS codes. The performance of MD system always outperforms the SD system. For example, looking at 8 dB channel SNR in Fig. 3, for the systems with RS codes, there is almost 2 dB gain obtained by comparing the dotted lines. Similarly, for the systems with LDPC codes, there is almost 5 dB gain obtained by comparing the continuous lines.. The better performance of the system that uses LDPC codes is due to its excellent error correction capability as compared to RS codes over the Rician fading channel.

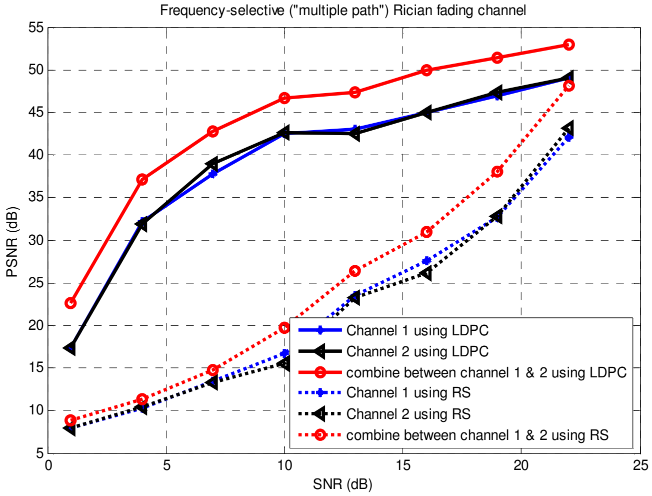


FIGURE 4: PSNR (dB) via frequency-selective “multiple path” Rician channel

1. Conclusions

In this paper the proposed MDC scheme for image transmission system together with two FEC schemes are analysed. The technique has significantly improves the performance of image transmission system in wireless channels. The proposed MDC scheme is an alternative technique for image transmission in wireless channel where methods that use error control schemes such as ARQ are not suitable due to the introduction of delay.

The MDC scheme increases robustness of data transmission. If a receiver gets only one description (other descriptions is lost), it can still reconstruct image with lower but acceptable quality. Simulation results have shown that the image transmission system with LDPC codes performs better than RS codes for both simulation via AWGN and Rician fading channels.

Acknowledgements

This work is funded in part by MOSTI Science Fund with Grant Number 6013353, USM Fellowship, and USM RU grant with Grant Number 814012.

References

1. V. A. Vaishampayan, “Application of multiple description codes to image and video transmission over lossy networks," Proc. 7th Int. Workshop Packet Video, Brisbane, Australia, pp. 55-60, 18-19/3/1996.

2. Y. Wang, M. T. Orchard and A. R. Reibman, “Optimal pairwise correlating transforms for multiple description coding", Proc. Int. Conf. Image Process, vol. 1, pp. 679-683, Chicago, IL, Oct. 1998.

3. Y. Wang, A. R. Reibman, and S. Lin, “Multiple description coding for video delivery,” Proceedings of the IEEE, vol. 93, no. 1, pp. 57-70, Jan.2005.

4. D. Comas, R. Singh and A. Ortega, “Rate-distortion optimization in a robust video transmission based on unbalanced multiple description coding," IEEE Workshop Multimedia Signal Process., MMSP01, Cannes, France, Oct. 2001.

5. G. Schuller, J. Kova·cevic, F. Masson, and V. K. Goyal, “Robust low-delay audio coding using multiple descriptions," IEEE Trans. Speech and Audio Process., vol. 13, pp. 1014-1024, Sept. 2005.

6. M. Kwong, R. Lefebvre and S. Cherkaoui, “Multiple description coding for audio transmission using conjugate vector quantization," Proc. 122th AES Convention, Vienna, Austria, May 2007.

7. J. Balam and J. D. Gibson, “Multiple descriptions and path diversity using the AMR-WB speech codec for voice communication over MANETs," Proc. IWCMC, Vancouver, Jul. 2006.

8.V. Vaishampayan, “Design of multiple description scalar quantizers,”1EEE Trans. on Info. Theory, vol. 39, no. 3, pp. 821-834, May 1993.

9. Y. Wang, M. T. Orchard, and A. Reibman. “Multiple description image coding for noisy channels by pairing transform coefficients”, Proceedings of the 1st Workshop IEEE MMSP. Princeton, NJ, 1997.

Content Modelling for Human Action Detection via Multidimensional Approach

Lili N. A. liyana@fsktm.upm.edu.my
Department of Multimedia
Faculty of Computer Science & Information Technology
University Putra Malaysia
43400 UPM Serdang
Selangor, Malaysia

Fatimah K. fatimahk@fsktm.upm.edu.my
Department of Multimedia
Faculty of Computer Science & Information Technology
University Putra Malaysia
43400 UPM Serdang
Selangor, Malaysia

Abstract

Video content analysis is an active research domain due to the availability and the increment of audiovisual data in the digital format. There is a need to automatically extracting video content for efficient access, understanding, browsing and retrieval of videos. To obtain the information that is of interest and to provide better entertainment, tools are needed to help users extract relevant content and to effectively navigate through the large amount of available video information. Existing methods do not seem to attempt to model and estimate the semantic content of the video. Detecting and interpreting human presence, actions and activities is one of the most valuable functions in this proposed framework. The general objectives of this research are to analyze and process the audio-video streams to a robust audiovisual action recognition system by integrating, structuring and accessing multimodal information via multidimensional retrieval and extraction model. The proposed technique characterizes the action scenes by integrating cues obtained from both the audio and video tracks. Information is combined based on visual features (motion, edge, and visual characteristics of objects), audio features and video for recognizing action. This model uses HMM and GMM to provide a framework for fusing these features and to represent the multidimensional structure of the framework. The action-related visual cues are obtained by computing the spatio-temporal dynamic activity from the video shots and by abstracting specific visual events. Simultaneously, the audio features are analyzed by locating and compute several sound effects of action events that embedded in the video. Finally, these audio and visual cues are combined to identify the action scenes. Compared with using single source of either visual or audio track alone, such combined audio-visual information provides more reliable performance and allows us to understand the story content of movies in more detail. To compare the usefulness of the proposed framework, several experiments were conducted and the results were obtained by using visual features only (77.89% for precision;

72.10% for recall), audio features only (62.52% for precision; 48.93% for recall) and combined audiovisual (90.35% for precision; 90.65% for recall).

Keywords: audiovisual, semantic, multidimensional, multimodal, hidden markov model.

1. INTRODUCTION

Video can transfer a large amount of knowledge by providing combination of text, graphics, or even images. Therefore, it is necessary to analyze all types of data: image frames, sound tracks, texts that can be extracted from image frames and spoken words. This usually involves segmenting the document into semantically meaningful units, classifying each unit into a predefined scene type, and indexing and summarizing the document for efficient retrieval and browsing. In this research, recent advances in using audio and visual information jointly for accomplishing the above tasks were reviewed. Audio and visual features that can effectively characterize scene content, present selected algorithms for segmentation and classification, and reviews on some test bed systems for video archiving and retrieval will be described. To date, there is no “perfect” solution for a complete video data-management and semantic detection technology, which can fully capture the content of video and index the video parts according to the contents, so that users can intuitively retrieve specific video segments.

However, without appropriate techniques that can make the video content more accessible, all these data are hardly usable. There are still limited tools and applications to describe, organize, and manage video data. Research in understanding the semantics of multiple media will open up several new applications (Calic et al. 2005). Multimodality of the video data is one of the important research topics for the database community. Videos consist of visual, auditory and textual channels, (Snoek et al. 2005). These channels bring the concept of multimodality. Definition of the multimodality given by Snoek et. al. as the capacity of an author of the video document to express a predefined semantic idea, by combining a layout with a specific content, using at least two information channels. The visual part of the video is used to represent something happening in the video. Most probably, one or more entities in the video are acting. The audio part of the video is used to represent things heard in the video. Most probably, some sound is created by some entities either saying or making some sound. The textual part of the video is used to represent something either written on the screen or on some entity in video. All of these visual, audio and textual modalities should be considered in a multimodal video data model.

Modeling and storing multimodal data of a video is a problem, because users want to query these channels from stored data in a video database system (VDBS) efficiently and effectively. Video databases can be better accessed if the index generated contains semantic concepts. So, semantic-based retrieval becomes a proper solution to handling the video database. However, there is only few approaches use all of the information. When either audio or visual information alone is not sufficient, combining audio and visual clues may resolve the ambiguities in individual modalities and thereby help to obtain more accurate answers. In this research, an idea of integrating the audio features with visual features for video detection and retrieval was presented. For feasible access to this huge amount of data, there is a great need to annotate and organize this data and provide efficient tools for browsing and retrieving contents of interest. An automatic classification of the movies on the basis of their content is an important task. For example, movies containing violence must be put in a separate class, as they are not suitable for children. Similarly, automatic recommendation of movies based on personal preferences will help a person to choose the movie of his interest and leads to greater efficiency for indexing, retrieval, and browsing of the data in the large video archives. Beside visual, audio and textual modalities, video has spatial and temporal aspects. In this research, these aspects are also considered.

Spatial aspect is about position of an entity in a specific frame through the video. Spatial position in a specific frame can be given by two-dimensional coordinates. Temporal aspect is about time of a specific frame through the video. Hence a video element can be identified in a video with its frame position(s), X coordinate in frame(s), Y coordinate in frame(s). Specific events that occur in a certain place during a particular interval of time are called video events. Video events occur in a particular shot of the video. As a result, particularly, every event belongs to directly to some specific shot and indirectly to some specific sequence. But they still suffer from the following challenging problems: semantic gap, semantic video concept modelling, semantic video classification, semantic detection and retrieval, and semantic video database indexing and access.

Semantic context detection is one of the key techniques to facilitate efficient multimedia retrieval (Chu et al. 2004). Semantic context is a scene that completely represents a meaningful information segment to human beings. In this research, a multidimensional semantic detection and retrieval approach that models the statistical characteristics of several audiovisual events, over a time series, to accomplish semantic context detection was proposed. This semantic information can be used to produce indexes or tables-of-contents that enables efficient search and browsing of video content.

Action is the key content of all other contents in the video. Action recognition is a new technology with many potential applications. Recognizing actions from videos is important topic in computer vision with many fundamental applications in video surveillance, video indexing and social sciences (Weindland et al. 2006). Actions are described by a feature vector comprising both trajectory information (position and velocity), and a set of other descriptors (Robertson, N & Reid, I. 2006). During the last few years, several different approaches have been proposed for detection, representation and recognition, and understanding video events (Yilmaz, A. & Shah, M. 2005).

Understanding activities of objects, especially humans, moving in a scene by the use of video is both a challenging scientific problem and a very fertile domain with many promising applications. The important question in action recognition is which features should be used? Use of both audio and visual information to recognize actions of human present might help to extract information that would improve the recognition results. What to argue is that action is the key content of all other contents in the video. Just imagine describing video content effectively without using a verb. A verb is just a description (or expression) of actions. Action recognition will provide new methods to generate video retrieval and categorization in terms of high-level semantics.

When either audio or visual information alone is not sufficient, combining audio and visual features may resolve the ambiguities and to help to obtain more accurate answers. Unlike the traditional methods (Chen et al. 2003; Xiong et al. 2003; Divakaran et al. 2005; Zhang et al. 2006; Lu et al. 2001; Li 2000; Zhang & Kuo 2001) that analyze audio and video data separately, this research presents a method which able to integrate audio and visual information for action scene analysis. A multidimensional layer framework was proposed to detect action scene automatically. The approach is top-down for determining and extract action scenes in video by analyzing both audio and visual data. The first level extracts low level features such as motion, edge and colour to detect video shots and next we use Hidden Markov model (HMM) to detect the action. An audio feature vector consisting of *n* audio features which is computed over each short audio clip for the purpose of audio segmentation was used too. Then it is time to decide the fusion process according to the correspondences between the audio and the video scene boundaries using an HMM-based statistical approach. Results are provided which prove the validity of the approach. The approach consists of two stages: audiovisual event and semantic context detections. HMMs are used to model basic audio events, and event detection is performed. Then semantic context detection is achieved based on Gaussian mixture models, which model the correlations among several action events temporally. It is the interest of this research to investigate, discover new findings and contribute the idea to the domain knowledge. This research is a fundamental research with experimental proving. The experimental evaluations indicate that the approach is effective in detecting high-level semantics such as action scene. With this framework, the gaps between low-level features and the semantic contexts were bridged.

2. PREVIOUS WORKS

Human tracking and, to a lesser extent, human action recognition have received considerable attention in recent years. Human action recognition has been an active area of research in the vision community since the early 90s. The many approaches that have been developed for the analysis of human motion can be classified into two categories: model-based and appearance-based. A survey of action recognition research by Gavrila, in [5], classifies different approaches into three categories: 2D approaches without shape models, 2D approach with shape models and 3D approaches; the first approach to use 3D constraints on 2D measurements was proposed by Seitz and Dyer in [7].

Many approaches have been proposed for behaviour recognition using various methods including Hidden Markov Model, finite state automata, context-free grammar, etc. [8] made use of Hidden Markov models to recognize the human actions based on low-resolution image intensity patterns in each frame. These patterns were passed to a vector quantizer, and the resulting symbol sequence was recognize using a HMM. Their method did not consider the periodicity information, and they have no systematic method for determining the parameters of the vector quantization. [9] presented a method to use spatio-temporal velocity of pedestrians to classify their interaction patterns. [10] proposed probabilistic finite state automata (FA) for gross-level human interactions.

Previous works on audio and visual content analysis were quite limited and still at a preliminary stage. Current approaches for audiovisual data are mostly focused on visual information such as colour histogram, motion vectors, and key frames [1, 2, 3]. Although such features are quite successful in the video shot segmentation, scene detection based on the visual features alone poses many problems. Visual information alone cannot achieve satisfactory result. However, this problem could be overcome by incorporating the audio data, which may have additional significant information. For example, video scenes of bomb blasting should include the sound of explosion while the visual content may vary a lot from one video sequence to another. The combination of audio and visual information should be of great help to users when retrieving and browsing audiovisual segments of interest from database. Boreczky and Wilcox [4] used colour histogram differences, and cepstral coefficients of audio data together with a hidden Markov model to segment video into regions defined by shots, shot boundaries and camera movement within shots.

3. METHODOLOGY

Unlike previous approaches, our proposed technique characterizes the scenes by integration cues obtained from both the video and audio tracks. These two tracks are highly correlated in any action event. We are sure that using joint audio and visual information can significantly improve the accuracy for action detection over using audio or visual information only. This is because multimodal features can resolve ambiguities that are present in a single modality.

Besides, we modelled them into multidimensional form. The audio is analysed by locating several sound effects of violent events and by classifying the sound embedded in video. Simultaneously, the action visual cues are obtained by computing the spatio-temporal dynamic activity signature and abstracting specific visual events. Finally, these audio and visual cues are combined to identify the violent scenes. The result can be seen in Table 6 on comparison recognition percentage with using single source either visual or audio track alone, or combined audio-visual information.

The framework contains two levels. The first level is to compute features and segmentation. The second level is to apply the computation algorithms to the classification, detection and recognition function to obtain the desired action. See Figure 1.

In particular, this part of elaboration will be concerning with the process of action detection and classification, the framework and structure of building the HMM based framework, and with the automatic recognition and retrieval of semantic action detection, as in Figure 1. Through techniques to be proposed, scenes, segments and individual frames can be characterized in video. This research is concerning on the development of HMM based framework for semantic interpretation of video sequences using:

- Edge feature extraction
- Motion
- Colour feature extraction
- Audio

3.1 The First Layer

Basically the task of the first level is to split the image/video data into several regions based on colour, motion, texture or audio information. The goal of the feature extraction process is to reduce the existing information in the image into a manageable amount of relevant properties. First, what we require is the shot boundary between two different frames is clearly detected. We use a pixel-based approach for conducting the shot detection. Assume X and Y are two frames, and d(X,Y) is the difference between the two frames. $P_X(m,n)$ and $P_Y(m,n)$ represent the values of the (m,n)-th pixels of X and Y, respectively. The following equation shows the approach to do shot change detection. below graphical objects, as in Figure 1.

$$d_{XY}(m,n)=\begin{cases}1, & |P_X(m,n)-P_Y(m,n)|>T_1\\ 0, & else\end{cases}$$

$$and \qquad (1)$$

$$d(X,Y)=\frac{1}{m*n}\sum_m\sum_n d_{XY}(m,n) \qquad (2)$$

If d(X,Y) is larger than a threshold T_1, a shot change is detected between frames X and Y.

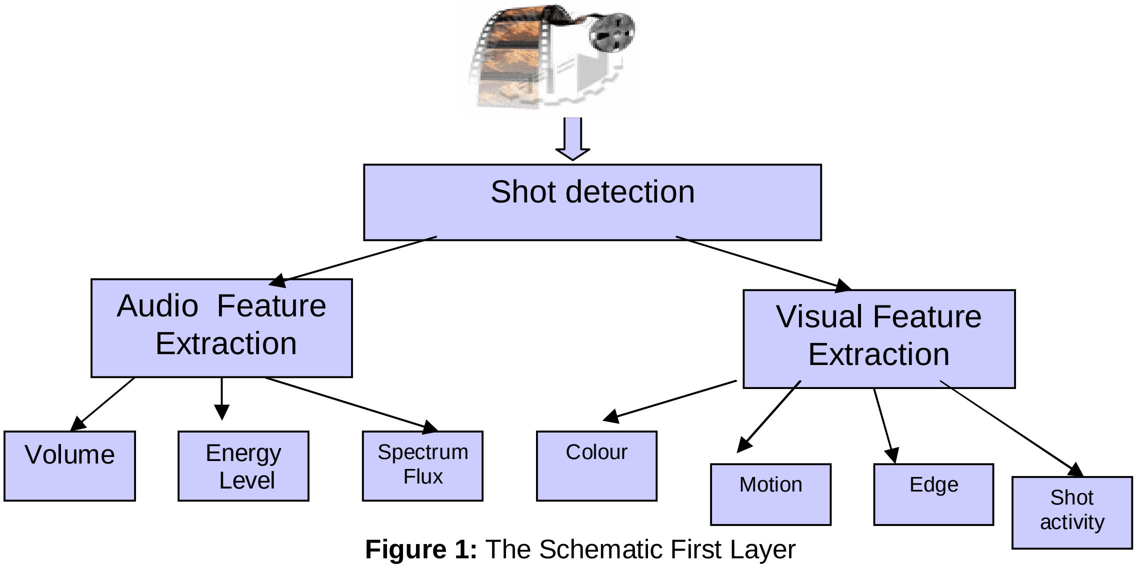


Figure 1: The Schematic First Layer

For extracting audio features, we use an audio feature vector consisting of n audio features which is computed over each short audio clips (100ms duration). The extracted features include: volume, band energy ratio, zero-crossing rate, frequency centroid and bandwidth. These adopted audio features have been widely used in many audio applications, e.g, [2,9] and are known to perform reasonably well.

Volume is the total spectrum power of an audio signal at a given time and is also referred as loudness. It is easy to compute for each frame and is a useful feature to detect silence or to distinguish speech from non-speech signals. The definition of volume is:

$$Vol = \frac{\int_{0}^{\omega_s} |SF(\omega)|^2 d\omega}{Vol_{\max}} \qquad (3)$$

Where $SF(\omega)$ denotes the short-time Fourier Transform coefficients and ω_S is the sampling frequency.

The band energy (BE) is defined as the energy content of a signal, in a band of frequencies:

$$BE = \int_{\omega_L}^{\omega_U} |SF(\omega)|^2 d\omega \qquad (4)$$

In order to model the characteristics of spectral distribution more accurately, the band energy ratio is considered in the system. The frequency spectrum is divided into sub-bands with equal frequency intervals, then the sub-bands energy are computed and normalized by the frame volume as:

$$BER = \frac{\int_{\omega_L}^{\omega_U} |SF(\omega)|^2 d\omega}{Vol} = \frac{BE}{Vol} \qquad (5)$$

where ω_U and ω_L are the upper and the lower bound frequencies of a sub-band, respectively.

Zero crossing rates has been extensively used in many audio processing algorithms, such as voiced and unvoiced components discrimination, end-point detection, audio classification, etc. This feature is defined as the average number of signal sign changes in an audio frame:

$$ZCR = \frac{1}{2N} \sum_{i=1}^{N-1} |\, sign(x(i)) - sign(x(i-1)) \,| \qquad (6)$$

where *x(i)* is the input audio signal, *N* is the number of signal samples in a frame, and *sign()* is the sign function.

Frequency centroid (FC) is the first order statistics of the spectrogram, which represents the power-weighted median frequency of the spectrum in a frame. It is formulated as follows:

$$FC = \frac{\int_0^{\omega_s} \omega |SF(\omega)|^2 \, d\omega}{\int_0^{\omega_s} |SF(\omega)|^2 \, d\omega} \qquad (7)$$

Bandwidth (BW) is the second-order statistics of the spectrogram, which represents the power-weighted standard deviation of the spectrum in a frame. The definition of BW is as follows:

$$BW = \sqrt{\frac{\int_0^{\omega_s} (\omega - FC)^2 \; |SF(\omega)|^2 \; d\omega}{\int_0^{\omega_s} \{SF(\omega)|^2 \; d\omega}} \qquad (8)$$

Frequency centroid and bandwidth are usually combined to describe statistical characteristics of the spectrum in a frame, and their reliability and effectiveness have been proved in previous work [6]. The video feature extraction, mainly based on the low level visual features, which characterize colours, shapes, textures or motion. Colour breaks are detected by comparing the colour histograms between adjacent video frames. Colour histogram, which represents the colour distribution in an image, is one of the most widely used colour features. The histogram feature measures the distance between adjacent video frames based on the distribution of luminance levels. It is simple, easy to compute, and works well for most types of video [8].

The luminance of a pixel *Lpixel* is computed from the 8-bit red (R), green (G), and blue (B) components as

$$Lpixel = 0.30008(R) + 0.5859(G) + 0.1133(B) \qquad (9)$$

H is a 64 bin histogram computed by counting the number of pixels in each bin of 4 gray levels, thus

$$H[k] = \#\text{ of pixels where } k = Lpixel/4,\ 0<=k<=63 \qquad (10)$$

As for the texture, we use three values: coarseness, contrast and direction as defined by Tamura [3] to represent its feature.

Time series motion data of human's whole body is used as input. Every category of target action has a corresponding model (action model), and each action model independently calculates the likelihood that the input data belongs to its category. Then the input motion is classified to the most likely action category. The feature extraction (position, direction, movement) focuses attention on the typical motion features of the action, and a model of the features' behaviour in the form of HMM.

A motion data is interpreted at various levels of abstraction. The HMM expresses what the action is like by symbolic representation of time-series data. In this work, we combine information from features that are based on image differences, audio differences, video differences, and motion differences for feature extraction. Hidden Markov models provide a unifying framework for jointly modeling these features. HMMs are used to build scenes from video which has already been segmented into shots and transitions. States of the HMM consist of the various segments of a video. The HMM contains arcs between states showing the allowable progressions of states. The parameters of the HMM are learned using training data in the form of the frame-to-frame distances for a video labeled with shots, transition types, and motion.

3.2 The Second Layer

The semantic concepts process is performed in a hierarchical manner. In hierarchical recognition, a motion data is interpreted at various levels of abstraction. At first, rough recognition is performed and then more detailed recognition is carried out as the process goes down to the lower level. The advantages of using hierarchy are as follows: recognition of various levels of abstraction, simplification of low level models and response to data easily.

At the semantic context level, the proposed fusion schemes that include feature construction and probabilistic modeling take the result from the first level as a basis for characterizing semantic context. The characteristics of each event are then modeled by an HMM in terms of the extracted features from the first level. The results from the event detection are fused by using statistical models: HMM. The fusion work is viewed as a pattern recognition problem, and similar features (detection result of audio events) would be fused to represent a semantic context. See Figure 2.

We use HMMs to characterize different events. For each event, 100 short video clips each 5 – 10s in length are selected as the training data. Based on the results extracted from the training data, a complete specification of HMM with two model parameters (model size and number of mixtures in each state) would be determined.

The HMM-based fusion scheme constructs a general model for each semantic context and tackles different combinations of relevant events. A hidden Markov model $\lambda = (A, B, \pi)$ consists of the following parameters:

1. N, the number of states in the model
2. M, the number of distinct observation symbols in all states
3. the state transition probability distribution
4. the observation probability distribution
5. the initial state distribution

For each semantic context, the parameters of HMM are estimated from the Baum-Welch algorithm by giving sets of training data. The state number N is set at four, and the number of distinct observation symbols M is also four in here.

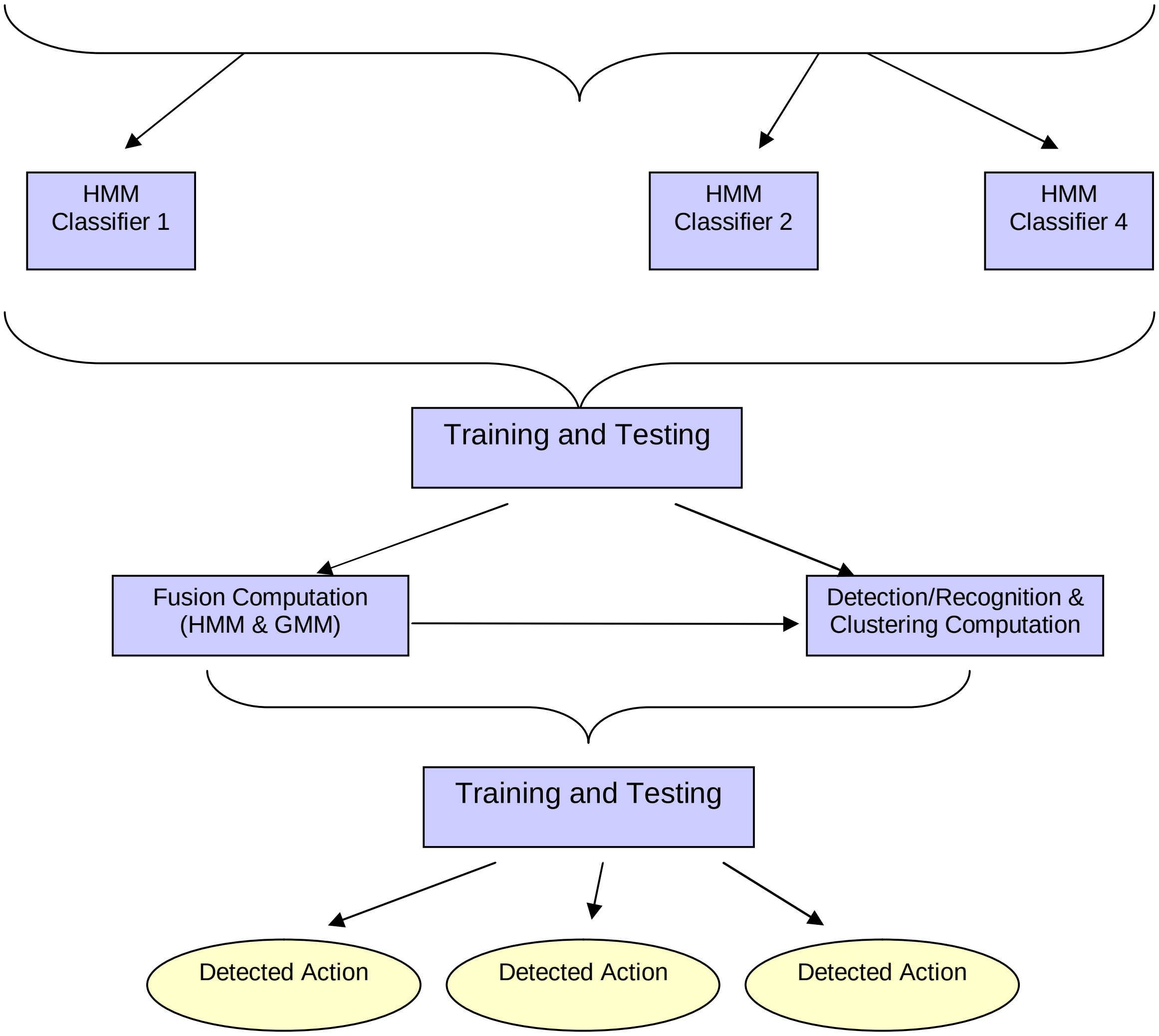


Figure 2: The Schematic Second Layer

Seven HMMs were trained with continuous data extracted from video sequences, one HMM for each action that would be recognized. The HMMs are trained with the data set using the EM algorithm. The parameters of the HMM (including the distribution of each state and transition probabilities between states) are learned using training data of a video manually labelled with shots, transition types, and motion types. Note that in fact, this model is not a “hidden” one, as the states are pre-labelled, and the probability distribution of each state is trained using training segments with the corresponding label. Once the HMM is trained, a given video is segmented into its component shots and transitions, by applying the EM algorithm, to determine the most likely sequence of states. These HMMs are then form the basis for automatically recognize test video sequences into blocks that each relate to one particular action. And since the domain-specific classes jump/run/walk/kick/stand/punch/sit in action movies are very diverse in themselves, a set of video shots for each class were used to capture the structure variations. K=6 with 3-state HMMs topology were trained for jump/run/walk/kick/stand/punch, respectively. The observations are modelled as a mixture of Gaussians. Once HMM models are learned, they can be used to parse new video clips into jump/run/walk/kick/stand/punch/sit action. The data likelihood was evaluated for each of the set of pre-trained model, e.g.

jump models $\theta_j = \{\theta_j^1, ..., \theta_j^K\}$ against each feature chunk *F(t)*, to get likelihood values

$$\overline{Q_j^k}(t), k = 1,...,K; t = 1,...,T_\omega. \qquad (11)$$

And similarly, for *run* models $\theta_r = \{\theta_r^1, ..., \theta_r^K\}$, the likelihood values are

$$\overline{Q_r^k}(t), k = 1,...,K; t = 1,...,T_\omega. \qquad (12)$$

The maximum-likelihood values were taken into decision among all HMMs as the label for the current feature chunk. HMM likelihood represents the fitness of each model for every short segment. Although exist a huge body of literature about models of human and biological motion dynamics including data from physiological studies, it is believed that the parameters of the dynamical representations should be estimated from example data. Given the number of linear dynamical systems *M* and the HMM topology, an iterative nonlinear training technique was presented here which enable to estimate the system parameters of each of the action model $\phi_m := [A0_m, A1_m, B_m]$.

4. EXPERIMENTAL RESULTS

The application is implemented using Mathlab Tools with Microsoft Visual C++. All of the tests are performed using Microsoft Windows XP Professionals v.2002 operating system running on HP Workstation xw4100 Pentium ® IV CPU 2.40GHz, 512MB of RAM. In order to determine how well each action is recognized as well as overall detection performance, a measurement method is needed. To evaluate such performance and effectiveness of detection, the following rules and metrics are used:

- Correct and false detection: consider any candidate detection as a correct detection if more than 10% of the actual transition ranges in included in the associated candidate detection range. That is, each correct detection must hit any portion of actual transition with at least 10% temporally overlapped frames.
- Precision and Recall: given a set of correct and false detection, calculate the detection performance by two traditional metrics, precision and recall, which are widely used for performance evaluation in detection and recognition process.

$$\mathrm{Re}call = \left(\frac{\#\ \ of\ \ correct\ \mathrm{det}ections}{\#\ \ of\ \ actual\ \ action} \right)$$

or

$$\mathrm{Re}call = \frac{CorrectDetection}{TotalShots} \qquad (13)$$

$$\mathrm{Pr}ecision = \left(\frac{\#\ \ of\ \ correct\ \mathrm{det}ections}{\#\ \ of\ \ \mathrm{det}ections} \right)$$

or

$$\mathrm{Pr}ecision = \frac{CorrectDetection}{CorrectDetection + FalseDetection} \qquad (14)$$

Note that these two metrics “globally” indicate how effectively algorithms detect the presence of violence action. An ideal retrieval system is one that retrieves all of the items or information appropriate to the user's need and which retrieves nothing else; furthermore, it must do so on every occasion. Retrieval effectiveness may be quantified.

This work focuses on using produced data, specifically VCD/DVD movies as a dataset. The rich nature of the VCD/DVD content allows users to use the result of this work in other domains that have domain specific grammar and have specific structural elements (e.g. baseball videos etc.) present.

Movies of different genres were digitized including action, adventure, horror and drama to create a database of a few hours of video. Data from seven movies (Ghost Rider, Matrix, Charlie's Angel, 2 Fast 2 Furious, I Robot, Terminator and Rush Hour) has been used for the experiments. The feature movies cover a variety of genres such as horror, drama, and action. Each input scene contains approximately 20 – 30 shots. To prove the usefulness of the proposed method, few experiments were performed to evaluate the detection performance with several video scenes. For the experiment, ten samples scenes with multi-modal features, *i.e.*, *Smultimodal = {S*1, *S*2, *S*3, *S*4, *S*5, *S6, S*7, *S*8, *S9*, *S10}*, were selected to consider dependency among features. The experimental results show that the proposed method to detect simple violence action gives high detection rate and reasonable processing time. The action detection time was calculated for the case with the multimodal feature set. It takes about 102 seconds to detect action within single video clip with PC (Pentium IV CPU 2.40GHz). The overall process time depends on various factors: CPU clock speed, the type of language used in system implementation, optimization scheme, the complexity and the number of processing steps, etc. Because the proposed action detection is performed with unit of short video clip, the detection time is not affected by the length of entire video.

$$Detection\_rate = \sum_{i=r}^{M} \binom{M}{i} P_g^i \left((1-P_g\right)^{(M-i)} \qquad (15)$$

where *r = Round(M/2),M = 1, 3, 5, . . . (odd numbers).*

Action detection rate is obtained by equation (15), *M* is the number of video clips chosen from a video and P_g is the probability to classify one action for single video frame. By using processes for motion computation, the result of successful detected action is depicted in Table 1. As is seen, sitting is characterized as the most significant motion with 80% of success rate. This is due to less motion activities involved. These actions were taken from the dataset itself. For example, most of the standing and walking scenes were collected from movie I, Robot. Most of the falling scenes were captured from Rush Hour, sitting and punching from Charlie's Angel, and kicking and running from Matrix.

Type of Sequence	Total Number	Correctly Classified	% Success
Standing	4	3	75
Sitting	5	4	80
Walking	3	2	67
Punching	5	3	60
Falling	4	3	75
Kicking	6	3	50
Running	2	1	50

TABLE 1: Classification of the individual action sequences

The audio test-data set contains six test sets for each event. The database currently contains data on 10 cries, 10 shots, 10 screams, 10 engines and 10 explosions. This experiment series contained a total of 30 tests. By analyzing and compute the specified audio information (amplitude, frequency, pitch, etc.) needed for classifying audio, Table 2 shows the classification rate. It showed that the classification of audio within a shot was 77.2% in average in the experiment. From table, it shows explosion scored the highest accuracy due to the loudness of it. Screams stands second place as they have the loudness and identified characteristics. Gunshots might have integrated sound so they were not highly recognized. As engines usually combined with fire visual, engine sound is only substitute to this experiment and it always comes in a package of sound (explosion, screeching, bang, etc). These were captured from the movie *2 Fast 2 Furious.*

Audio		**Results in Percent (%)**		
	Correctly classified	**No recognition possible**	**Falsely classified**	**Σ**
Gunshot	81	10	9	100
Cry	51	32	17	100
Explosion	93	7	0	100
Scream	85	10	5	100
Engine	76	9	15	100
Average	**77.2**	**13.6**	**9.2**	**100**

TABLE 2: Audio classification results

Performances were compared over each features, i.e. visual, audio and motion, and performances obtained when features are combined. To compare the usefulness of the proposed multimodal features in this multidimensional framework for action detection, the classification performances of these three cases were evaluated. One case with audio features only, another case with visual features, and the other with audiovisual features. Refer Table 3, Table 4, and Table 5. Table 6 demonstrates a summary of the results for the different methods used in these experiments. Overall, from the table, both the precision rate and recall rate are satisfactory. This experiment gives the best results when audiovisual features are included in the detection process.

SampleVideo	**Correct**	**Miss**	**Fault**	**Precision**	**Recall**
S1	55	15	22	55/77 = 0.714	55/70 = 0.786
S2	14	1	2	14/16 = 0.875	14/15 = 0.933
S3	2	8	3	2/5 = 0.400	2/10 = 0.200
S4	15	5	7	15/22 = 0.682	15/20 = 0.750
S5	11	30	12	11/23 = 0.478	11/41 = 0.268
S6	24	43	19	24/43 = 0.558	24/67 = 0.358
S7	11	17	4	11/15 = 0.733	11/28 = 0.393
S8	10	6	3	10/13 = 0.769	10/16 = 0.625
S9	8	18	3	8/11 = 0.727	8/26 = 0.307
S10	6	16	13	6/19 = 0.316	6/22 = 0.273

TABLE 3: Performance based on audio features

SampleVideo	Correct	Miss	Fault	Precision	Recall
S1	64	6	22	64/86 = 0.744	64/70 = 0.914
S2	10	5	3	10/13 = 0.769	10/15 = 0.667
S3	6	4	2	6/8 = 0.750	6/10 = 0.600
S4	18	2	12	18/30 = 0.600	18/20 = 0.900
S5	28	13	6	28/34 = 0.823	28/41 = 0.683
S6	47	20	9	47/56 = 0.839	47/67 = 0.701
S7	18	10	9	18/27 = 0.678	18/28 = 0.643
S8	13	3	0	13/13 = 1.000	13/16 = 0.813
S9	17	9	4	17/21 = 0.809	17/26 = 0.653
S10	14	8	4	14/18 = 0.777	14/22 = 0.636

TABLE 4: Performance based on colour and motion features (visual feature)

SampleVideo	Correct	Miss	Fault	Precision	Recall
S1	55	15	24	67/79 = 0.848	67/70 = 0.957
S2	15	0	1	15/16 = 0.938	15/15 = 0.100
S3	8	2	1	8/9 = 0.888	8/10 = 0.800
S4	20	0	7	20/27 = 0.741	20/20 = 1.000
S5	34	7	1	34/35 = 0.971	34/41 = 0.829
S6	61	6	2	61/63 = 0.971	61/67 = 0.910
S7	27	1	3	27/30 = 0.900	27/28 = 0.964
S8	13	3	2	13/15 = 0.866	13/16 = 0.812
S9	23	3	1	23/24 = 0.958	23/26 = 0.884
S10	20	2	1	20/21 = 0.952	20/22 = 0.909

TABLE 5: Performance based on audiovisual features

	By audio only (%)		By visual only (%)		By both audio visual (%)	
SampleVideo	Precision	Recall	Precision	Recall	Precision	Recall
S1	71.4	78.6	74.4	91.4	84.8	95.7
S2	87.5	93.3	76.9	66.7	93.8	100
S3	40.0	20.0	75.0	60.0	88.9	80.0
S4	68.2	75.0	60.0	90.0	74.1	100
S5	47.8	26.8	82.3	68.3	97.1	82.9
S6	55.8	35.8	83.9	70.1	97.1	91.1
S7	73.3	39.3	67.8	64.3	90.0	96.4
S8	76.9	62.5	100	81.3	86.7	81.2
S9	72.7	30.7	80.9	65.3	95.8	88.4
S10	31.6	27.3	77.7	63.6	95.2	90.9
Average	**62.52**	**48.93**	**77.89**	**72.10**	**90.35**	**90.65**

TABLE 6: Action detection results by audio only, audiovisual combination

From Table 6, the average recall rate was 90.65% and the average precision rate was 90.35%.**.** As shown in the table, the average recall rate from using audio feature only (48.93%) is lower than those from using visual features only (72.10%) because video data has the complex structure which combines dialogs, songs, colours, and motion activities. The results clearly indicate the effectiveness and the robustness of the proposed algorithm and framework for multidimensional and multimodal styles. Using the audio and visual feature together yielded the highest classification accuracy. As for *S1*, the highest recall rate (95.7%) gained by using both the audio and visual features. *S1* consists of scenes with fire and action, taken from the movie Ghost Rider.

5. CONCLUSION

In this paper, we present a recognition method of human action that utilizes hierarchical structure. By utilizing hierarchical structure, recognition of various levels of abstraction for one action data, simplification of low level models and response to data by decreasing the level of details become possible. The hierarchical approach will also bridges the gaps between low level features and high level semantics to facilitate semantic indexing and retrieval.

We addressed the problem of modeling and recognizing actions, proposing a two layer HMM framework to decompose action analysis problem into two layers. The first layer maps low level audio visual features into one confidence score. The second layer uses results from the first layer as input to integrate and fusing multiple media clues (audio, visual and motion) to recognize actions.

Now the prototype system is under development. Full experiments results will be put out soon. This work is to believe would be very useful for achieving semantic multimedia retrieval.

6. REFERENCES

1. S. W. Smoliar and H. Zhang, “Content-based Video Indexing and Retrieval”. IEEE Multimedia, pp.62 – 72. 1994.

2. W. Niblack, et al., “Query by Images and Video Content: The QBIC System”. Computer, vol. 28 no. 9, pp. 23 – 32, 1995.

3. S. F. Chang, W. Chen and H.J. Meng, et al., “A Fully Automated Content-based Video earch Engine Supporting Spatio-temporal Queries”, IEEE Trans. Circuits System Video Technology, vol. 2, pp. 602 -615, 1998.

4. J. S. Boreczky and L.D. Wilcox, “A Hidden Markov Model Framework for Video Segmentation using Audio and Image Features”, in Proceedings of the International Conference Acoustics, Speech, Signal Processing, pp. 3741 – 3744, 1998.

5. D M Gavrila. “The Visual Analysis of Human Movement: A Survey”, Computer Vision and Image Understanding, vol. 3 no.1, pp.82 - 98, 1999.

6. S. Fischer, R. Lienhart, and W. Effelsberg, “Automatic Recognition of Film Genres”, Proceedings of ACM Multimedia, pp. 295 – 304, 1995.

7. S. Seitz and C.R. Dyer, “View MorthiMorphing: Uniquely Predicting Scene Appearance from Basis Images”. Proceedings on Image Understanding Workshop, pp. 881 – 887, 1997.

8. J. Yamato, J. Ohya, and K. Ishii, “Recognizing Human Action in Time-Sequential Images using Hidden Markov Models”. Proceedings of Computer Vision and Pattern Recognition, pp. 379 – 385, 1992.

9. K. Sato, J. K. Aggarwal, “Tracking and Recognizing Two-Person Interactions in Outdoor Image Sequences”. Proceedings of IEEE Workshop on Multi Object Tracking, pp. 87 – 94, 2001.

10. S. Hongeng, F. Bremond and R. Nevatia, “Representation and Optimal Recognition of Human Activities”. IEEE Proceedings of Computer Vision and Pattern Recognition, pp. 818 – 825, 2000.

Empirical Evaluation of Decomposition Strategy for Wavelet Video Compression

Rohmad Fakeh rohmad@rtm.gov.my
Engineering Division
Department of Broadcasting
Radio Television Malaysia

Abdul Azim Abd Ghani azim@fsktm.upm.edu.my
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
43400 Serdang, Selangor, Malaysia

ABSTRACT

The wavelet transform has become the most interesting new algorithm for video compression. Yet there are many parameters within a wavelet analysis and synthesis which govern the quality of a decoded video. In this paper different wavelet decomposition strategies and their implications for the decoded video are discussed. A pool of color video sequences has been wavelet-transformed at different settings of the wavelet filter bank and quantization threshold and with decomposition of dyadic and packet wavelet transformation strategies. The empirical evaluation of the decomposition strategy is based on three benchmarks: a first judgment regards the perceived quality of the decoded video. The compression rate is a second crucial factor, and finally the best parameter setting with regards to the Peak Signal to Noise Ratio (PSNR). The investigation proposes dyadic decomposition as the chosen decomposition strategy.

Keywords: Wavelet Analysis, Decomposition Strategies, Empirical Evaluation

1. INTRODUCTION

Wavelet technology has provided an efficient framework of multi-resolution space-frequency representation with promising applications in video processing. Discrete wavelet transform (DWT) is becoming increasingly important in visual applications because of its flexibility in representing non-stationary signals such as images and video sequences.

Applying 3D wavelet transform to digital video is a logical extension to the 2D analysis [20]. Most video compression techniques use 2D coding to achieve spatial compression and motion compensated difference coding in the time domain. Most of these techniques involved complicated and expensive hardware. By applying the wavelet transform in all the three dimensions, the computational complexity of coding while achieving high rates of compression can be reduced, depending on the coding strategy [1].

The choice of coding strategy has been reported by many researchers [19], [20], [22], [6], [5], [1]. Basically the three-dimensional wavelet decomposition can be performed in three ways: temporal filtering followed by two-dimensional spatial filtering known as (t+2D) [19], [20], [27], [22], [6], two-dimensional spatial filtering followed by temporal filtering (2D+t) [30]. The third approach is introduced for scalable video coding (SVC) is 2D+t+2D uses a first stage DWT to produce reference video sequence at various resolution; t+2D transform are then performed on each resolution level of the obtained spatial pyramid [1].

The research in t+2D orientation of the 3D spatio-temporal wavelet video coding one important advantage is that it avoids motion estimation and motion compensation which are generally very difficult task where the motion parameters are usually sensitive to transmission errors, include low computational complexity [22]. The research on t+2D was on low bit rate wavelet image and video compression with adaptive quantization, coding and post processing. A video signal is decomposed into temporal and spatial frequency sub bands using temporal and spatial bandpass analysis filter-banks. The computational burden of the 3D sub band video coding is minimized by decomposing the video signal of temporal decomposition based on 2-tap Haar filter-bank basically the difference and average between frames [24], [28].

However this research uses traditional multi-tap FIR filterbanks such as the 9-tap QMF filter of Adelson [2] and perfect-reconstruction FIR filters of Daubechies [9], [10], [11]. This spatio-temporal wavelet transformation which produces fixed and limited to 11 sub-bands tree-structured spatio-temporal decomposition [22] as not optimum and not flexible since the limited level of penetration is limited to two for low-pass-temporal and one level for high-pass temporal.

Along the same t+2D orientation using Haar filter for the temporal sub band filtering, research by Ashourian et al. [6] on Robust 3-D sub band video coder has followed the same orientation as done by Luo [22]. The research uses traditional multi-tap FIR filterbanks such as the 9-tap QMF filter of Adelson [2] and spatio-temporal wavelet transformation produces 11 sub-bands. The research also applies different types of quantization depends on the statistics of each of the 11 sub bands which is not optimum in video coding performance, and the high-pass sub bands are quantized using a variant of vector quantization.

The rest of the paper is organized as follows: Section 2 discusses related works. Section 3 presents the proposed 3D wavelet video compression scheme. Section 4 contains performance measures, and Section 5 presents results of decomposition strategy. Section 6 discusses the outcomes from decomposition strategy. Finally, the conclusions are mentioned in Section 7.

2. RELATED WORKS

Research reported by Luo [22] was on low bit rate wavelet image and video compression with adaptive quantization, coding and post processing. A video signal was decomposed into temporal and spatial frequency subbands using temporal and spatial bandpass analysis filter-banks. According to Karlsson et. al [19], the computational burden of the 3D sub band video coding is minimized by decomposing the video signal of temporal decomposition based on 2-tap Haar filter-bank, basically the difference and average between frames [24], [28]. This also minimizes the number of frames needed to be stored and the delay caused by the analysis and synthesis procedures. In the case of spatial decomposition, longer length of filters can be applied since these filters can be operated in parallel.

The research uses traditional multi-tap FIR filterbanks such as the 9-tap QMF filter of Adelson [2] and perfect-reconstruction FIR filters of Daubechies [9], [10], [11]. This spatio-temporal wavelet transformation produces a fixed and limited an 11 sub-bands tree-structured spatio-temporal decomposition as in Figure 1. The template for displaying the 11-band decomposition is as in Figure 2. The sub bands produced correspond to penetration depth or decomposition level of two to the Temporal Low-pass and one level to the Temporal High-Pass sub-bands.

The general strategies for the quantization and coding algorithm on the characteristics of the 11 sub bands are as follows:

1. Sub bands at courser scale levels with small index in Figure 2 with most significant energy and higher visual significance requires relatively higher quality coding and finer quantization.
2. Sub bands at finer scale levels are quantized more coarsely, or can be discarded.

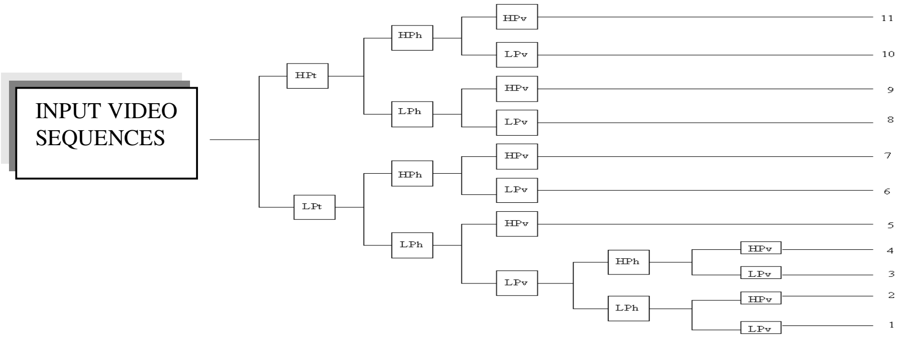


FIGURE 1: An 11 sub-bands tree-structured spatio-temporal decomposition

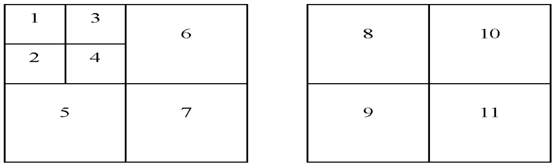


FIGURE 2: Template for displaying the 11-band decomposition

Since images are of finite support, the research by Luo [22] used several applicable extension methods, including zero padding and symmetric extension. Although the best extension method may be image dependent, in most cases, appropriate symmetric extension yield optimum result. The video sequences used are in CIF formats (360x288), with frame rate of 15fps, which are typically used in videoconferencing through ISDN channels. There is no direct comparison to the performance of the 3D methods used since the experimental results are on the entropy reduction for Lena image and the extent of the adaptive quantization with results in comparison to EZW coding. The report proposes a fixed number of sub bands with represents a decomposition of level 2 and this is might not be optimal for rate-distortion point of view since the lowest frequency sub bands can further be decomposed in a tree-structured fashion to achieve higher compression ratio.

Research by Ashourian [6] on robust 3-D sub band video coder has followed the same orientation as done by Luo [22]. The research also applies different types of quantization depends on the statistics of each of the 11 sub bands and the high-pass sub bands are quantized using a variant of vector quantization. The results of the simulation over un-noisy channels are as in Table 1.

	Claire	Miss-America	Salesmar	Suzie	Carphone
Bitrate (kbits/s)	Average PSNR [dB				
62.0	37.0	38.5	30.2	33.5	30.0
113.0	39.5	42.0	32.8	34.9	32.2
355.0	42.1	43.8	37.6	38.6	36.9

TABLE 1: Performance comparison (PSNR, [dB] of research by Ashourian [6] at frame rate of 7.5fps, and video bitrate of 62, 113 and 355 kbps.

3. PROPOSED 3D WAVELET VIDEO COMPRESSION SCHEMEThe proposed wavelet 3D video coding methods is outlined in Figure 3. The original input color video sequences in

QCIF, CIF and SIF formats are used in the simulations. The video sequences are fed into the coder either in RGB, YUV or YCbCr color space first by temporal filtering without motion compensation and followed by spatial filtering. The color-separator signal is transformed into frequency space yielding a multi-resolution representation or wavelet tree of image with different levels of detail.

Separable 3D dyadic wavelet transformation from the four families of filter banks are used and applied to each of the luminance and chrominance components of the video sequences frame by frame. The boundary extensions can be applied here from the proposed boundary treatment strategies. The appropriate level of decomposition depth or penetration depth can be applied to the compression scheme employing either global or level-dependent thresholds. The compressed video sequences of every color components are reconstructed and the objective evaluation of MSE, MAE, PSNR, Bit- rate and compression ratio are then calculated.

The input color sequences are first temporally decomposed into Low-Pass-Temporal (LPT) and High-Pass-Temporal (HPT) sub bands. For example for YUV color space of the input video sequences received, for SIF resolution of 352x240 pixels, the corresponding functions to read the .sif video yielding individual Y,U and V image components using level dependent threshold, each of the Y,U,V components are fed into the function of "comp_3dy_qsif_lvd.m" .

```
 [Y,U,V]=read_sif('football',i).
By initializing  A = zeros(size(352,240));
B = Y;  % B is also assigned to values of U and V for the chrominance components
 L_B = plus(A,B)/2;       % For average temporal subbands of the same size
 H_B = minus(A,B);       % For difference temporal subbands of the same size
```

The various related information and explanation steps used in the proposed algorithm are explained in the proceeding sections.

3.1 Input Sequences

The fundamental difficulty in testing an input image and video compression system is how to decide which test sequences to use for the evaluations. Monochrome, color images and color video sequences are to be selected and evaluated.

A) Monochrome Images

A digital grayscale image is typically represented by 8 bits per pixel (bpp) in its uncompressed form. Each pixel has a value ranging from 0 (black) to 255 (white). Transform methods are applied directly to a two dimensional image by first operating on the rows, and then on the columns. Transforms that can be implemented in this way are called separable. Figure 4: (a) shows the original Lena image which is then wavelet transformed to the fifth decomposition levels, as in (b). The corresponding histogram plot and the frequency response are as in Figure 4: (c) and (d) respectively.

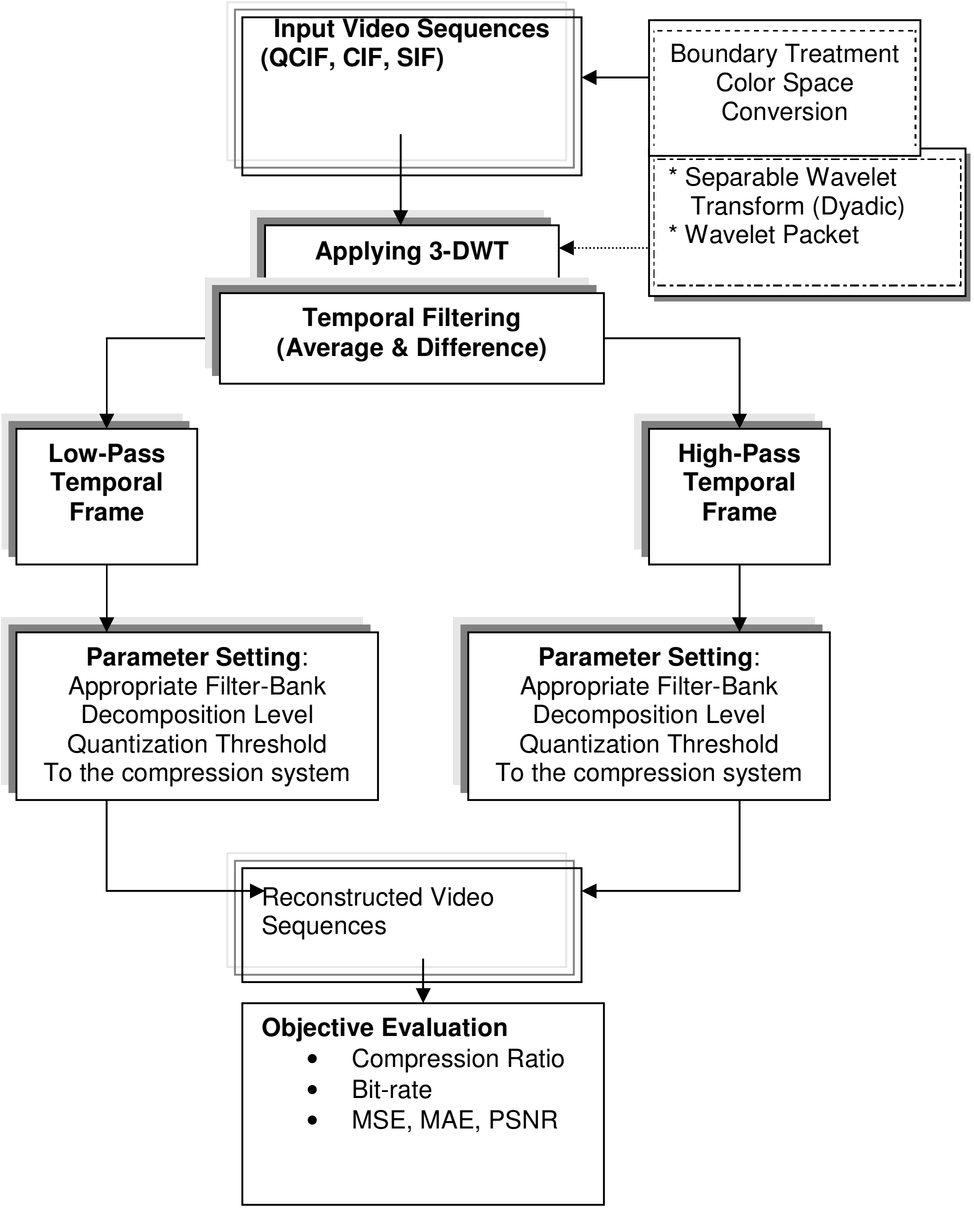


FIGURE 3: Proposed 3D Wavelet Video Compression

B) Color Images

A digital color image is stored as a three-dimensional array and uses 24 bits to represent each pixel in its uncompressed form. Each pixel contains a value representing a red (R), green (G), and blue (B) component scaled between 0 and 255–this format is known as the RGB format. Image compression schemes first convert the color image from the RGB format to another color space representation that separates the image information better than RGB. In this thesis the color images are converted to the luminance (Y), chrominance-blue (Cb), and chrominance-red

(Cr) color space. The luminance component represents the intensity of the image and looks like a grey scale version of the image. The chrominance-blue and chrominance-red components represent the color information in the image. The Y, Cb, and Cr components are derived from the RGB space.

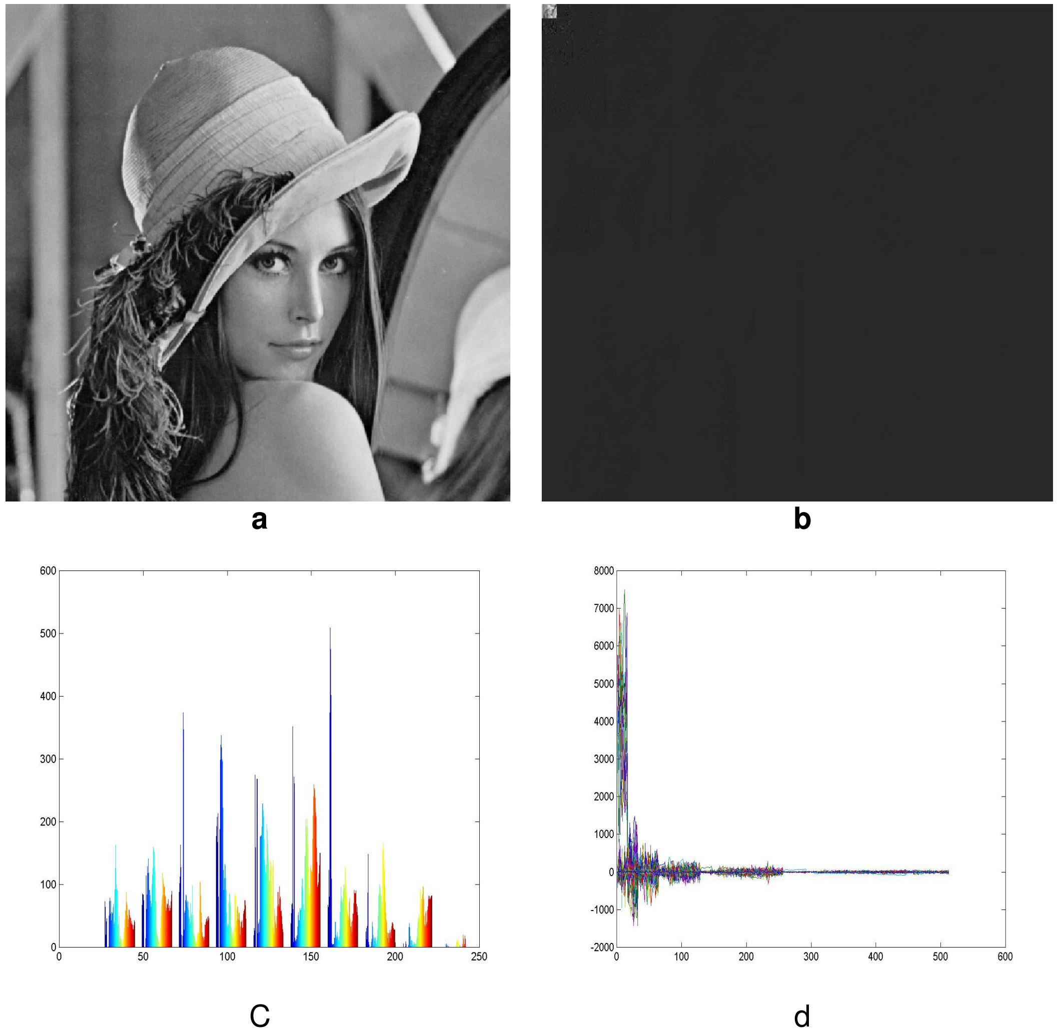


FIGURE 4: (a) The original Lena Image, (b) Level 5 decomposition of Lena image, (c) Histogram plot of the original Lena image, (d) Frequency response after level 5 decomposition of Lena image.

C) Source Picture Formats

To implement the standard, it is very important to know the picture formats that the standard supports and positions of the samples in the picture. Table 2 shows the different kinds of motion of QCIF video sequences. The samples are also referred to as pixels (picture elements) or pels. Source picture formats are defined in terms of the number of pixels per line, the number of lines per picture, and the pixel aspect ratio. H.263 allows for the use of five standardized picture formats.

These are the CIF (common intermediate format), QCIF (quarter-CIF), sub-QCIF, 4CIF, and 16CIF. Besides these standardized formats, H.263 allows support for custom picture formats that can be negotiated. Details of the five standardized picture formats are summarized in Table 3.

Since human eyes are less sensitive to the chrominance components, these components typically have only half the resolution, both horizontally and vertically, of the Y components, hence the term "4:2:0 format."

No	Video Sequences in QCIF Formats	Kinds of Motion
1	Carphone	Fast object translation
2	Claire	Slow object translation
3	Foreman	Object translation and panning. Medium spatial detail and low amount of motion or vice versa.
4	News	Medium spatial detail and low amount of motion vice versa.
5	Akiyo	Low spatial detail and low amount of motion
6	Mother &Daughter	Low spatial detail and low amount of motion
7	Grandmother	Low spatial detail and low amount of motion
8	Salesman	Low spatial detail and low amount of motion
9	Suzie	Low spatial detail and low amount of motion
10	Miss America	Low spatial detail and low amount of motion

TABLE 2: Different Kinds of Motion of QCIF Video Sequences

	Format				
Property	Sub_QCIF	QCIF	CIF	4CIF	16CIF
Number of pixels per line	128	176	352	704	1408
Number of lines	96	144	288	576	1152
Uncompressed bit rate (at 30 Hz), Mbit/s	4.4	9.1	37	146	584

TABLE 3: Standard Picture Format Supported by H.263

D) Original Sequences

Original test video sequences of Quarter Common Intermediate Format (QCIF) with each frame containing 144 lines and 176 pixels per line, Common Interface Format (CIF) with each frame containing 288 lines and 352 pixels per line and SIF with each frame containing 240 lines and 352 pixels per line. The QCIF test video sequences typically have different kinds of motion, such as fast object translation, for the case of Carphone. Foreman has slow object translation and panning. Claire sequence shows slow object translation and low motion activity. Suzie and Miss America show stationary, small displacement and slow motion.

The original test sequences in CIF sized progressive digital sequences, are originally stored in YUV format or YCrCb, with the U or Cr and V or Cb components sub-sampled 2:1 in both horizontal and vertical directions.

E) Color Space Conversion

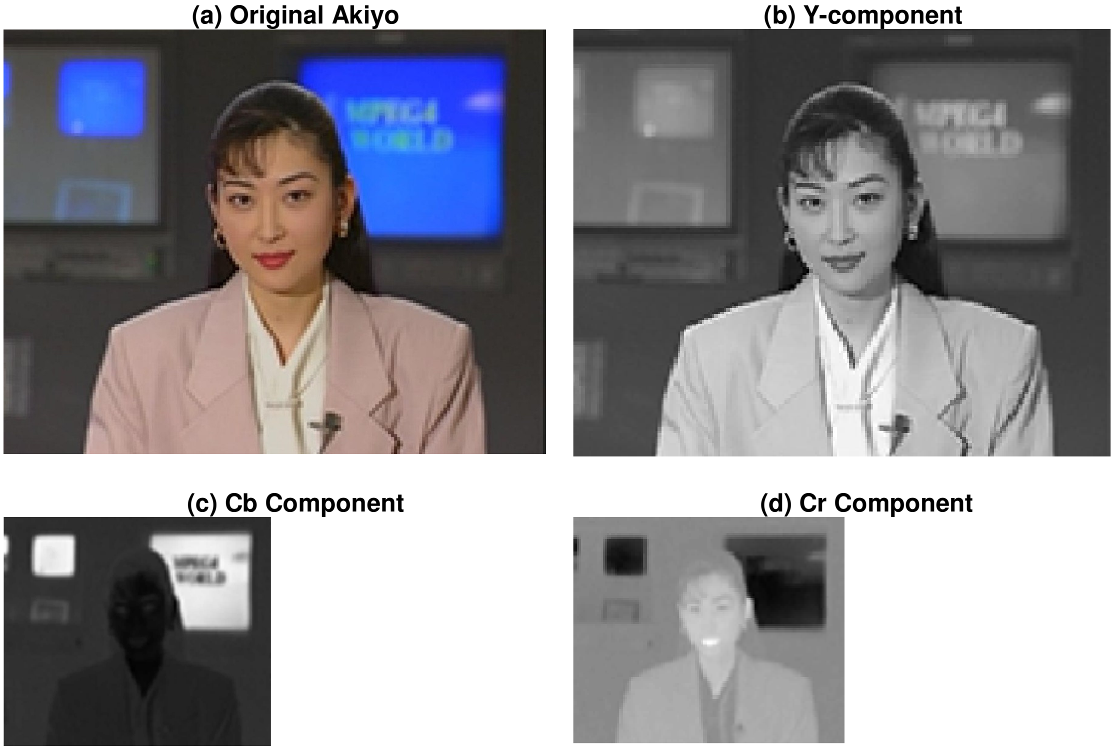
The video sequences of QCIF, CIF and SIF sizes are converted into individual YCbCr format so that the data is represented in a form more suitable for compression, and wavelet decomposition of each color component. In order to eliminate spectral redundancies, color space is changed from RGB to YCbCr as the first step of compression. Changing the color space does not introduce any error. The following equations transform RGB components into YCbCr of ICT (Irreversible component transformation, used for lossy image compression):

$$\begin{bmatrix} \mathbf{Y} \\ \mathbf{Cb} \\ \mathbf{Cr} \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ \mathbf{G} \\ \mathbf{B} \end{bmatrix}$$

TABLE 4: Matrix of RGB to YCbCr Color Conversion

The first component Y, or luminance represents the intensity of the image. Cb and Cr are the chrominance components and specify the blueness and redness of image respectively. Figure 5, Figure 6, and Figure 7 shows the Akiyo, News and Stefan video image respectively, in the YCbCr color space and in each of the three components. This illustrates the advantage of using the YCbCr color space–most of the information is contained in the luminance. Each of the three components (Y, Cb, and Cr) is input to the coder. The PSNR is measured for each compressed component (Yout, Cbout, and Crout) just as for grayscale images. The three output components are reassembled to form a reconstructed 24-bit color image.

As can be seen from the pictures of figures, the Y-component contributes the most information to the image, as compared to the other two components of Cb and Cr. This makes it possible to get greater compression by including more data from the Y-component than from the Cb and Cr components.



(a) Original Akiyo

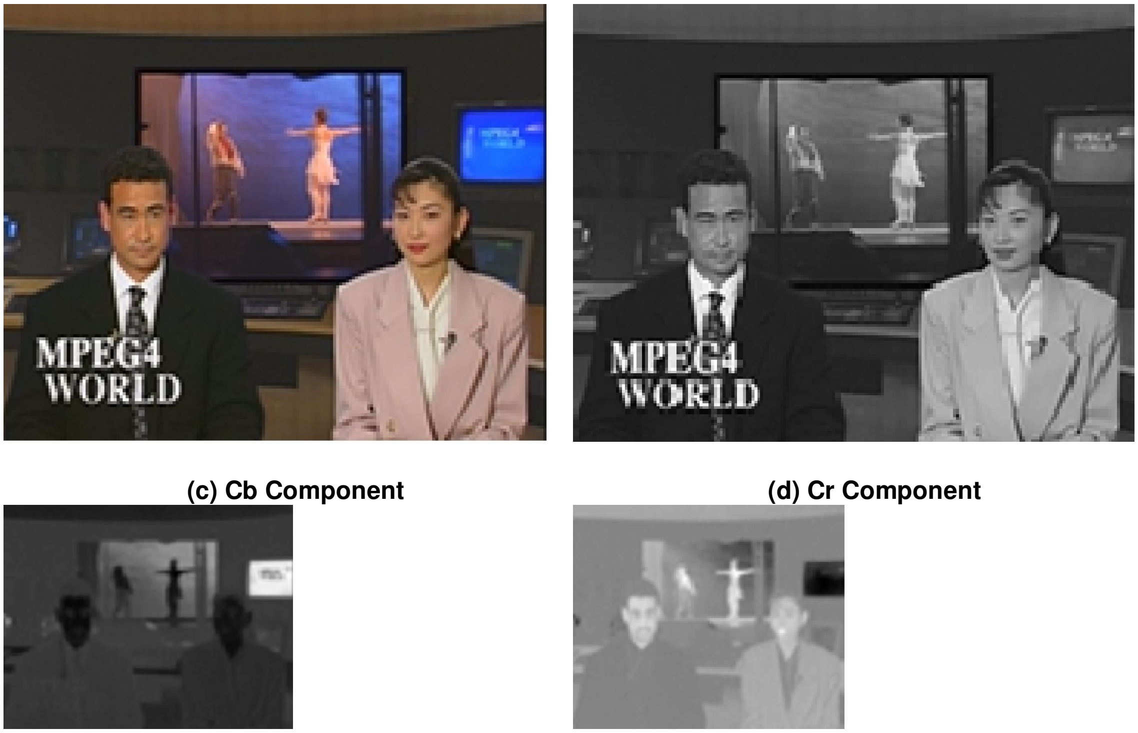
(b) Y-component

(c) Cb Component

(d) Cr Component

FIGURE 5: (a) The Original, (b) Y, (c) Cb and (d) Cr components for the Akiyo QCIF Sequence

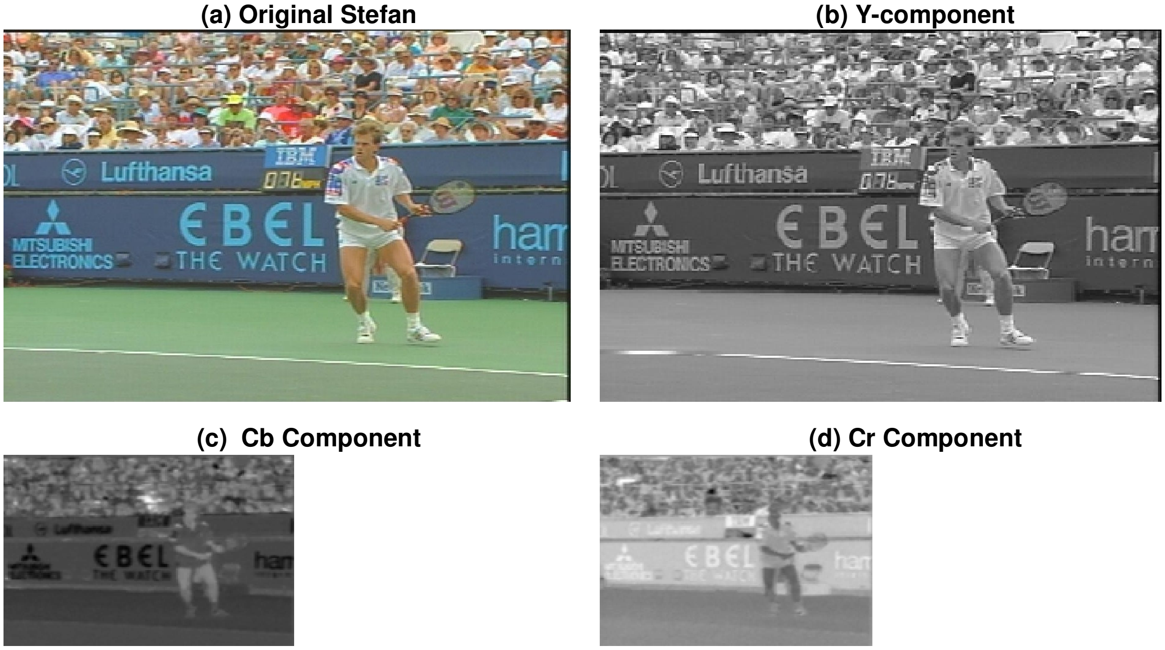
(a) Original News **(b) Y-component**



(c) Cb Component

(d) Cr Component

FIGURE 6: a) The Original, (b) Y, (c) Cb and (d) Cr components for the News CIF Sequence



(a) Original Stefan

(b) Y-component

(c) Cb Component

(d) Cr Component

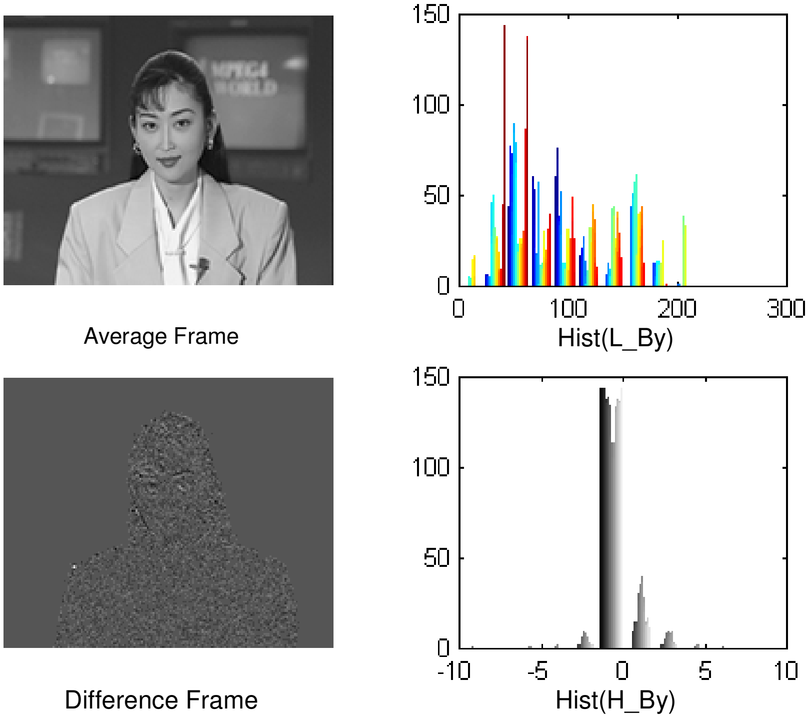
FIGURE 7: a) The Original, (b) Y, (c) Cb and (d) Cr components for the the Stefan SIF Sequence

3.2 Wavelet Transformation

Video compression techniques, which only remove spatial redundancy, cannot be highly effective. To achieve higher compression ratios the similarity of successive video frames has to be exploited.

A) Temporal Compression

The color source video sequences in QCIF, SIF and CIF formats are used producing sequences in Y, U and V color components as the input to the compression scheme. The average and difference between frames producing two temporal low-pass and high-pass sub-bands are calculated. For example the average and difference frame and the corresponding histogram plot for the 1st. frame of Akiyo video sequence is given as in Figure 8.


Average Frame

Hist(L_By)

Difference Frame

Hist(H_By)

FIGURE 8: Wavelet decomposition of Average and Difference frame and the corresponding histogram plot for the 1st. frame of Akiyo video sequence

B) Spatial Compression

For the application of the wavelet transform to images, cascaded 1D filters are used. The 1D discrete wavelet transformation step is calculated by using Mallat's pyramid algorithm [23]. First, the forward transformation combined with a down-sampling process of the image is performed, once in the horizontal and twice in the vertical direction. This procedure produces one low- and three high pass components. This procedure is applied recursively to the low pass output until the resulting low pass component reaches a size small enough to achieve effective compression of the image. In general, additional levels of transformation result in a higher compression ratio.

Spatial compression attempts to eliminate as much redundancy from single video frames as possible without introducing degradation of quality. This is done by first transforming the image from spatial to frequency domain and secondly by applying quantizing threshold the transformed coefficients. A two dimensional discrete wavelet transformation (DWT) is applied spatially to the images producing one level of decomposition into LL, LH, HL and HH sub-bands as in Figure 9. Figure 10 shows the High-pass and Low pass sub-bands with the low-pass sub-bands further decomposed and iterated. Figure 11 further illustrated the decomposition steps up to level three.

Both the Low-Pass-temporal and the High-Pass-Temporal sequences are shown to be spatially decomposed into a number of sub bands as in Figure 12.

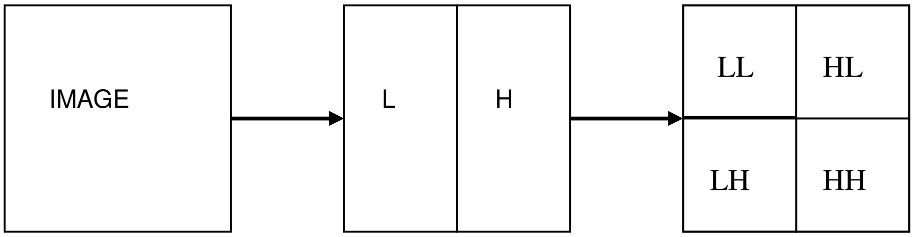


FIGURE 9: Level one of 2-D DWT applied on an image

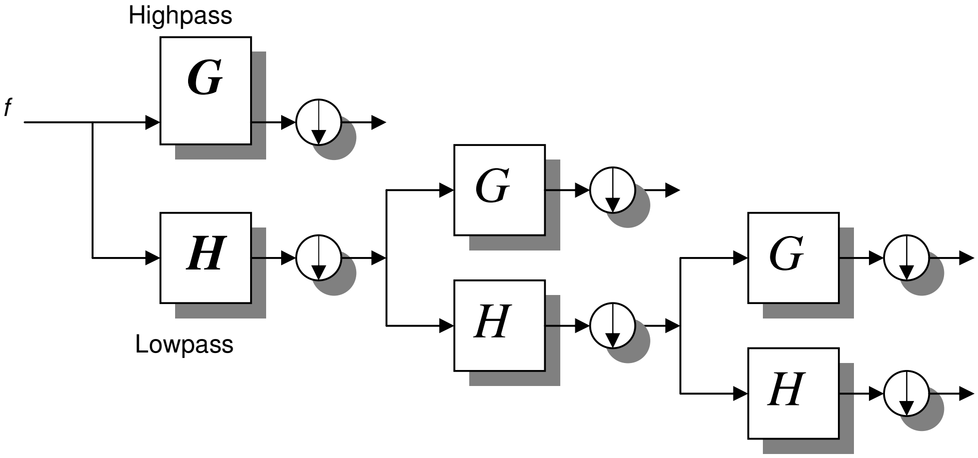


FIGURE 10: Level one of 2-D DWT of Highpas and Lowpass

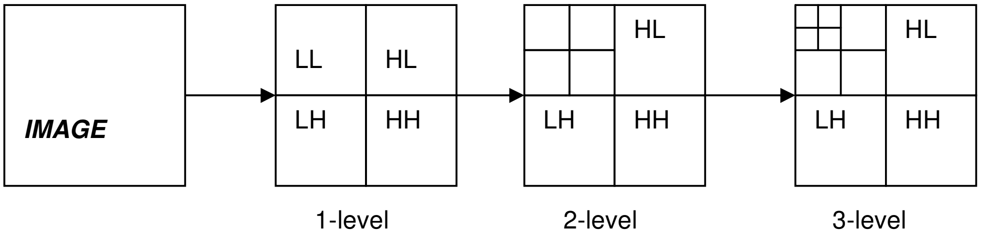


FIGURE 11: Level Three Dyadic DWT scheme used for Image Compression

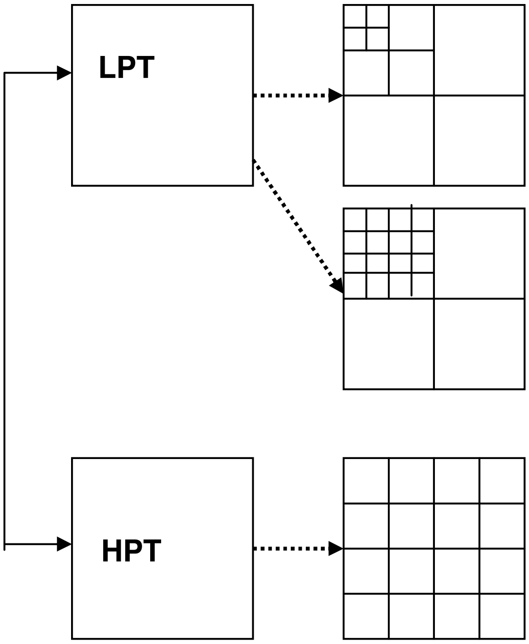


FIGURE 12: High-Pass Temporal and Low-Pass Temporal for Video sequences Compression

Given a signal *s* of length *N*, the DWT consists of $\log_2 N$ stages. The first step produces, starting from *s*, two sets of coefficients: approximation coefficients cA_1 and detail coefficients cD_1. These vectors are obtained by convolving *s* with the low pass filter LoF_D for approximation, and with the high-pass filter HiF_D for detail, followed by dyadic decimation. For images, an algorithm similar to the one-dimensional case is possible for two-dimensional wavelets and scaling functions obtained from one-dimensional wavelets by tensor product. The two-dimensional DWT leads to a decomposition of approximation coefficients at level j in four components: the approximation at level $j+1$ and the details in three orientations (horizontal, vertical and diagonal).

The multilevel 2-D decomposition in the MATLAB Environment, Wavelet Toolbox is a two-dimensional wavelet analysis function, of [c,s] = wavedec2(x,n,'wname'), which returns the wavelet decomposition of the input image of matrix x, at n decomposition levels, using the wavelet named in the string 'wname'. The output wavelet 2-D decomposition structure [C,S] contains the wavelet decomposition vector C and the corresponding book-keeping matrix S. Vector C is organized as: C = [A(N) | H(N) | V(N) | D(N) | ... H(N-1) | V(N-1) | D(N-1) | ... | H(1) | V(1) | D(1)], where A, H, V, D, are row vectors such that: A = approximation coefficients, H = horizontal detail coefficients, V = vertical detail coefficients, D = diagonal detail coefficients, each vector is the vector column-wise storage of a matrix. Matrix S is such that: S(1,:) = size of approximation coefficient(N), S(i,:) = size of detail coefficient(N-i+2) for i = 2,...,N+1 and S(N+2,:) = size(X).

C) The choice wavelet filter banks

The choice of wavelet basis for video compression was based on reconstruction properties and runtime complexity [16]. Generally, complexity for wavelet filter is $O(n)$, where n is the number of filter taps. The one-dimensional n-tap filter pair is applied as follows:

$$l_k = \sum_{i=0}^{n=1} \tilde{L}_i x_{2k(i_0+i)} \quad \text{and} \quad h_k = \sum_{i=0}^{n=1} \tilde{H}_i x_{2k(i_0+i)}$$

$\widetilde{L}$ and $\widetilde{H}$ are the low- and high-pass filters, x the pixel values with row- or column-index i, and k is the index of filter output. Iterating with step $2k$ automatically introduces the desired down-sampling by 2. Filter coefficients are real numbers in the range [-1,1].

Much research effort has been expended in the area of wavelet compression, with the results indicating that wavelet approaches outperform DCT-based techniques [3], [4], [21], [25]. However, it is not completely clear which wavelets are suitable for video compression. Wavelets implemented using linear-phase filters are generally advantageous for image processing because such filters preserve the location of spatial details. A key component of an efficient video coding algorithm is motion compensation. However at the present time it is too computationally intensive to be used in software video compression. A number of low end applications therefore use motion-JPEG, in essence frame by frame transmission of JPEG images [29] with no removal of inter-frame redundancy.

The choice of filter bank in wavelet image and video compression is a crucial issue that affects both image and video quality and compression ratio. A series of bi-orthogonal, and orthogonal wavelet filters of differing length were evaluated by compressing and decompressing a number of standard video test sequences, using different quantization thresholds. In this section, the selection of wavelet filter-banks of QCIF, CIF and SIF video sequences and their implications for the decoded image are discussed. A pool of color video sequences has been wavelet - transformed with different settings of the wavelet filter bank, boundary selection, quantization threshold and decomposition method. The reconstructed video sequences of QCIF sizes are evaluated using an objective quality of peak signal to noise ratio (PSNR).

This section investigates how wavelet filter banks affect the subsequent quality and size of the reconstructed data, using a wavelet based video codec developed. Test video sequences were compressed with the codec, and the results obtained indicate that the choice of wavelet greatly influences the quality of the compressed data and its size.

D) The Wavelet Filter-Banks

The DWT is implemented using a two-channel perfect reconstruction linear phase filter bank [26]. Symmetric extension techniques are used to apply the filters near the frame boundaries; an approach that allows transforming images with arbitrary dimensions.

3.3 Wavelet Threshold Selection

This section describes wavelet thresholding for image compression under the framework provided by Statistical Learning Theory aka Vapnik-Chervonenkis (VC) theory. Under the framework of VC-theory, wavelet thresholding amounts to ordering of wavelet coefficients according to their relevance to accurate function estimation, followed by discarding insignificant coefficients. Existing wavelet thresholding methods specify an ordering based on the coefficient magnitude, and use threshold(s) derived under gaussian noise assumption and asymptotic settings. In contrast, the proposed approach uses orderings better reflecting statistical properties of natural images, and VC-based thresholding developed for finite sample settings under very general noise assumptions.

The plot of wavelet coefficients in Figure 4: (d) Frequency response after level 5 decomposition of Lena image in wavelet domain, suggests that small coefficients are dominated by noise, while coefficients with a large absolute value carry more signal information than noise. Stated more precisely, the motivation to this thresholding idea based on the following assumptions:

- The de-correlating property of a wavelet transform creates a sparce signal: most untouched coefficients are zero or close to zero.
- The noise level is not too high so that the signal wavelet coefficients can be distinguished from the noisy ones.

As it turns out, this method is indeed effective and thresholding is a simple and efficient method for noise reduction.

A) Global thresholding

Wavelet thresholding for image denoising involves taking the wavelet transform of an image (i.e., calculating the wavelet coefficients discarding setting to zero) the coefficients with relatively small or insignificant magnitudes. By discarding small coefficients one actually discard wavelet basis functions which have coefficients below a certain threshold. The denoised signal is obtained via inverse wavelet transform of the kept coefficients. One global threshold derived by Donoho [13], [14], [15] under gaussian noise assumption. Clearly, wavelet thresholding can be viewed a special case of signal/data estimation from noisy samples, which can be addressed within the framework of VC-theory. The original wavelet thresholding technique is equivalent to specifying a structure that uses only a magnitude ordering of the wavelet coefficients. Obviously, this is not the best way of ordering the coefficients.

B) Level dependent threshold

Level-dependent thresholding has been proposed to improve the performance of wavelet thresholding method. Instead of using a global threshold, level-dependent thresholding uses a group of thresholds, one for each scale level. It can be interpreted as the ordering of the wavelet coefficients with respect to their magnitudes adjusted by scale level.

This suggests that the level-dependent thresholding be viewed as a special case of more sophisticated importance ordering in model selection based denoising method. A number of different structures (ordering schemes) can be specified on the same set of basis functions. A good ordering should reflect the prior knowledge about the signal/data being estimated. Similarly, 2-D image signal estimation with VC approach may require more complicated ordering scheme.

4. PERFORMANCE MEASURES

Although there are several metrics that tend to be indicative of image quality, each of them has situations in which it fails to coincide with an observer's opinion [26]. However, since running human trials is generally prohibitively expensive, a number of metrics are often computed to help judge image quality. Some of the more commonly used "quality" metrics are given below, $x(m,n)$ stands for the original data sized M by N, and $\hat{x}(m,n)$ the reconstructed mage.

- **MAE (Mean Absolute Error)**

MAE: One quantity, often computed in conjunction with other metrics, is *maximum absolute error*. Since this metrics measure error, it is regarded as inversely proportional to image quality.

$$MAE = \text{Max}\, |\, x(m,n) - \hat{x}(m,n)\, | \quad (1)$$

- **MSE (Mean Square Error)**

MSE, RMS: Two other quantities that appear frequently when comparing original and reconstructed or approximated data are mean square error, and root mean square. These metrics attempt to measure an inverse to image quality.

$$MSE = \frac{1}{N.M}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left[x(m,n) - \hat{x}(m,n)\, |^{2}\right] \quad (2)$$

$$RMS = \sqrt{MSE} \quad (3)$$

- **SNR: Signal to noise ratio**

$$SNR = 10\,\log_{10}\left[\frac{\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}x(m,n)^{2}}{\sum_{i=1}^{N-1}\sum_{j=0}^{M=1}\left[x(m,n) - \hat{x}(m,n)\right]^{2}}\right] \quad (4)$$

- **Peak Signal-to-Noise Ratio (PSNR)**

Peak signal-to-noise ratio (PSNR) is the standard method for quantitatively comparing a compressed image with the original. For an 8-bit grayscale image, the peak signal value is 255. Hence the PSNR of an M×N 8-bit grayscale image x and its reconstruction ˆx is calculated. PSNR: Peak signal-to-noise ratio. The MSE and PSNR are directly related, and one normally uses PSNR to measure the coder's objective performance.

$$PSNR = 10\,\log_{10}\left(\frac{255^{2}}{MSE}\right) \quad (5)$$

At high rate, images with PSNR above 32 dB are considered to be perceptually lossless. At medium and low rates, the PSNR does not agree with the quality of the image. For color images, the reconstruction of all three color spaces must be considered in the PSNR calculation. The MSE is calculated for the reconstruction of each color space. The average of these three MSEs is used to generate the PSNR of the reconstructed RGB image (as compared to the original 24-bit RGB image).

$$PSNR = 10\,\log_{10}\left(\frac{255^{2}}{MSE_{RGB}}\right) \quad (6)$$

Where MSE_{RGB} is:-

$$MSE_{RGB} = \frac{MSE_{red} + MSE_{green} + MSE_{glue}}{3} \quad (7)$$

- **Compression ratio (CR)**

Compression ratio is the relation between the amount of data of the original signal compared to the amount of data of the encoded signal [8]:

$$CR = \frac{Amount\ of\ data(original\ signal)}{Amount\ of\ data(encoded)} \qquad (8)$$

5. RESULTS OF DECOMPOSITION STRATEGY

To evaluate the decomposition strategy, three types of video sequences in QCIF resolutions with varying types of motion such as Miss America, Foreman and Carphone are used in the simulation. To compare the performance of the decomposition strategy between the dyadic discrete wavelet transform (DWT) and wavelet packet (WP), an empirical evaluation is conducted using three test video sequences of Miss America, Foreman and Carphone. Nine types of wavelet filter-banks s are used for the evaluation. They are Bior-22, Bior-2.6, Bior-4.4, Bior-6.8, Coif-2, Coif-3, Sym-4, Sym-5 and Sym-7. Global threshold is used in this evaluation, with values of threshold (thr) ranging from 10 to 125. For every filter-bank used for decoding the three video sequences, the average PSNR values for both the DWT and WP are calculated and tabulated as in Table 5 to Table 10. The corresponding plots of PSNR values Vs the filter-banks used are as in Figure 13 to Figure 18.

	Thr=10							
Filter Name	**Miss America**		**Foreman**		**Carphone**		**Average PSNR [dB]**	
	DWT	**WP**	**DWT**	**WP**	**DWT**	**WP**	**DWT**	**WP**
Bior-2.2	42.09	40.90	38.16	32.85	39.99	33.77	**40.08**	35.84
Bior-2.6	42.40	40.88	38.34	32.78	40.18	33.18	**40.31**	35.61
Bior-4.4	41.80	40.73	38.28	33.38	39.71	34.37	**39.93**	36.16
Bior-6.8	41.96	40.86	38.33	33.33	39.63	34.07	**39.98**	36.09
Coif-2	41.97	30.33	38.42	21.99	39.87	22.30	**40.08**	24.88
Coif-3	41.89	40.06	38.39	30.18	39.74	31.47	**40.01**	33.90
Sym-4	41.98	30.02	38.41	22.36	39.88	21.34	**40.09**	24.57
Sym-5	42.03	41.86	38.39	36.10	39.85	37.86	**40.09**	38.60
Sym-7	41.86	39.96	38.34	28.47	39.62	29.90	**39.94**	32.78

TABLE 5: Average in PSNR for Wavelet Packet and Discrete Wavelet Transform of three video sequences of Miss America, Foreman, and Carphone , using the best nine types filters and Global threshold of 10

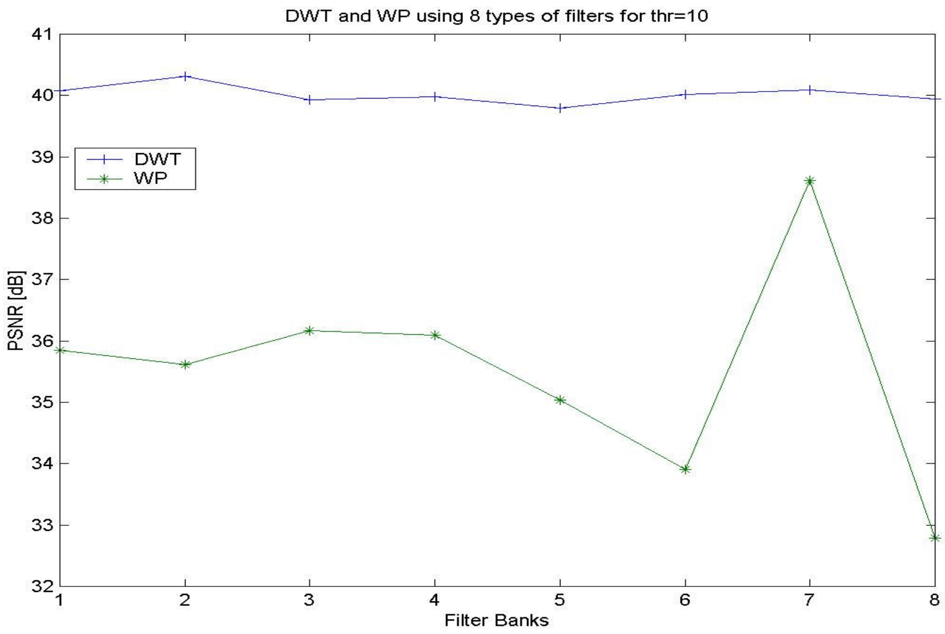


FIGURE 13: The Plot for Average PSNR for Wavelet Packet and Discrete Wavelet Transform of using the best nine types of wavelet filters and Global threshold of 10

	Thr=20							
Filter Nar	**Miss America**		**Foreman**		**Carphone**		**Average PSNR [dB]**	
	DWT	**WP**	**DWT**	**WP**	**DWT**	**WP**	**DWT**	**WP**
bior-2.2	38.32	37.99	33.45	30.98	35.32	32.28	**35.70**	33.75
bior-2.6	38.69	38.17	33.69	31.08	35.47	32.07	**35.95**	33.77
bior-4.4	37.74	37.47	33.32	31.54	34.85	32.78	**35.30**	33.93
bior-6.8	38.02	37.73	33.47	31.60	34.88	32.73	**35.45**	34.02
coif-2	38.09	30.60	33.46	23.03	34.97	22.61	**35.51**	25.42
coif-3	38.06	37.59	33.43	29.94	34.86	31.28	**35.45**	32.94
sym-4	37.97	30.77	33.48	23.61	35.00	22.79	**35.48**	25.72
sym-5	38.04	38.09	33.44	32.88	34.76	34.31	**35.42**	35.09
sym-7	37.94	37.09	38.34	29.59	34.61	30.11	**36.96**	32.27

Table 6: Average in PSNR for Wavelet Packet and Discrete Wavelet Transform of three video sequences of Miss America, Foreman, and Carphone, using the best nine types filters and Global threshold of 20

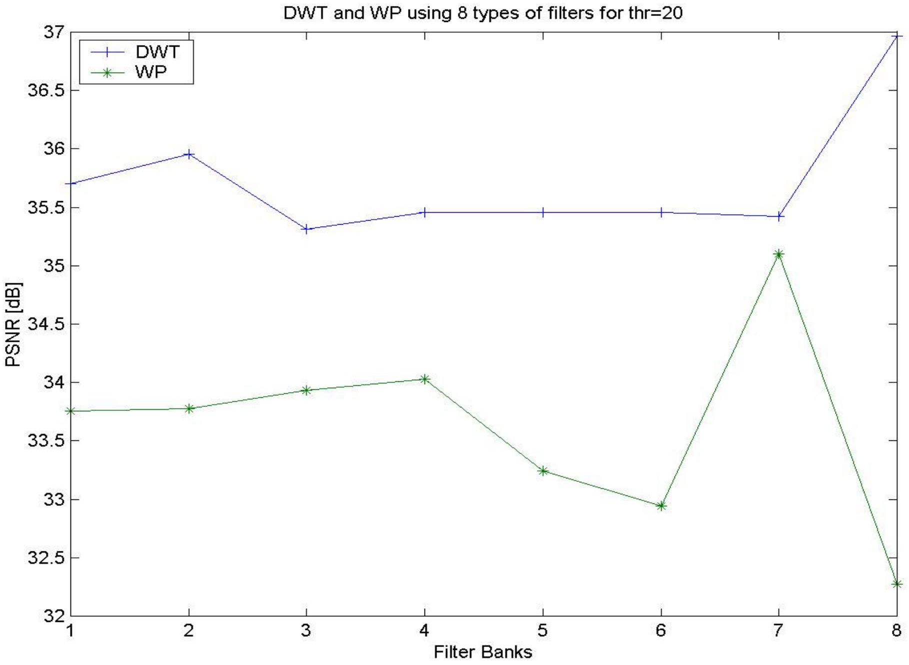


FIGURE 14: The Plot for Average PSNR for Wavelet Packet and Discrete Wavelet Transform of using the best nine types of wavelet filters and Global threshold of 20

				Thr=45				
Filter Name	**Miss America**		**Foreman**		**Carphone**		**Average PSNR [dB]**	
	DWT	**WP**	**DWT**	**WP**	**DWT**	**WP**	**DWT**	**WP**
bior-2.2	34.70	34.34	29.09	28.07	29.67	28.99	**31.15**	30.47
bior-2.6	35.06	34.69	29.37	28.26	30.02	28.99	**31.48**	30.64
bior-4.4	33.92	33.86	28.43	28.01	29.20	29.14	**30.52**	30.34
bior-6.8	34.23	34.18	28.66	28.32	29.28	29.22	**30.72**	30.57
coif-2	34.24	31.72	28.50	24.06	29.38	27.83	**30.70**	27.87
coif-3	34.22	34.04	28.48	27.30	29.10	29.18	**30.60**	30.17
sym-4	34.06	31.75	28.44	25.08	29.34	26.99	**30.61**	27.94
sym-5	33.90	33.97	28.51	28.67	29.44	29.76	30.62	**30.80**
sym-7	34.07	33.94	28.45	28.69	29.10	28.70	**30.54**	30.44

TABLE 7: Average in PSNR for Wavelet Packet and Discrete Wavelet Transform of three video sequences of Miss America, Foreman, and Carphone , using the best nine types filters and Global threshold of 45

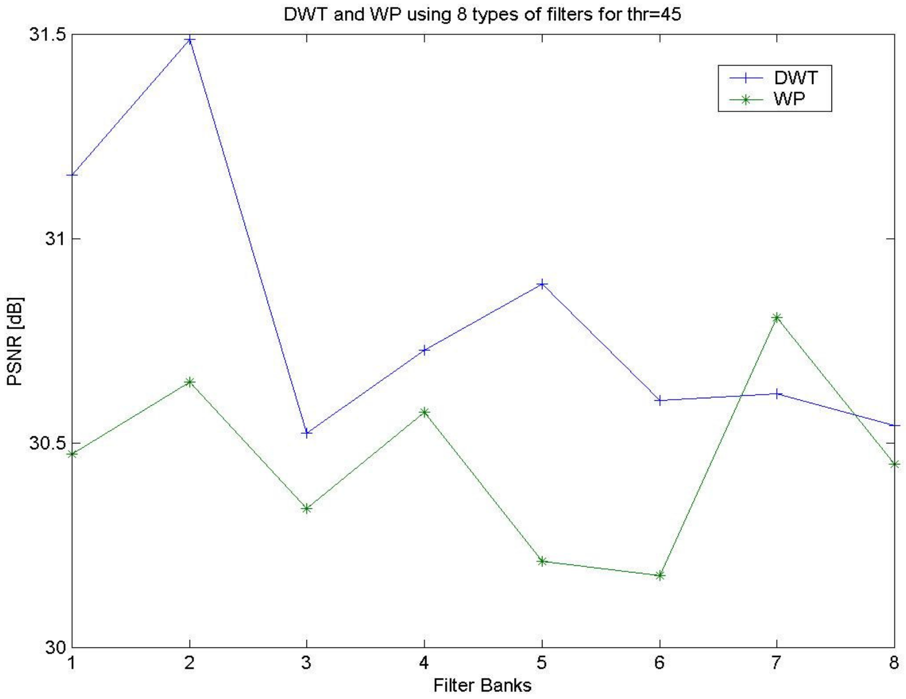


FIGURE 15: The Plot for Average PSNR for Wavelet Packet and Discrete Wavelet Transform of using the best nine types of wavelet filters and Global threshold of 45

			Thr=85					
Filter Name	**Miss America**		**Foreman**		**Carphone**		**Average PSNR [dB]**	
	DWT	**WP**	**DWT**	**WP**	**DWT**	**WP**	**DWT**	**WP**
bior-2.2	32.10	32.09	26.35	25.83	26.58	26.48	**28.34**	28.13
bior-2.6	32.46	32.44	26.73	26.04	26.99	26.08	**28.72**	28.19
bior-4.4	31.41	31.40	25.56	25.41	25.93	26.09	27.63	**27.64**
bior-6.8	31.76	31.74	25.91	25.75	26.20	26.36	**27.96**	27.95
coif-2	31.51	31.37	25.75	23.00	26.26	23.79	**27.84**	26.05
coif-3	31.45	31.54	25.80	24.99	26.11	25.77	**27.79**	27.43
sym-4	31.60	31.30	25.58	25.05	26.08	25.37	**27.75**	27.24
sym-5	31.65	31.68	25.81	25.77	26.27	26.41	27.91	**27.95**
sym-7	31.35	31.44	25.82	25.84	26.22	25.82	**27.80**	27.70

TABLE 8: Average in PSNR for Wavelet Packet and Discrete Wavelet Transform of three video sequences of Miss America, Foreman, and Carphone, using the best nine types filters and Global threshold of 85

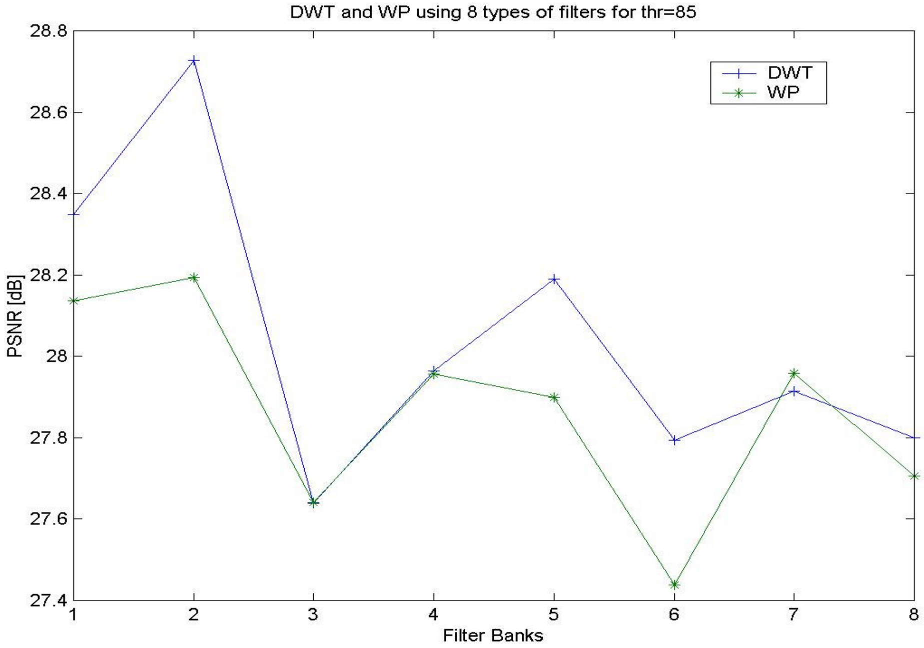


FIGURE 16: The Plot for Average PSNR for Wavelet Packet and Discrete Wavelet Transform of using the best nine types of wavelet filters and Global threshold of 85

Filter Name	Miss America		Thr=125 Foreman		Carphone		Average PSNR [dB]	
	DWT	**WP**	**DWT**	**WP**	**DWT**	**WP**	**DWT**	**WP**
bior-2.2	30.95	30.85	24.71	24.48	24.58	24.48	**26.75**	26.60
bior-2.6	31.34	31.30	25.04	24.83	24.91	24.82	**27.10**	26.98
bior-4.4	30.15	30.38	23.86	23.87	23.91	24.26	25.98	**26.17**
bior-6.8	30.40	30.65	24.17	24.20	24.13	24.27	26.24	**26.37**
coif-2	30.31	30.29	24.15	22.01	24.42	23.05	**26.30**	25.12
coif-3	30.36	30.40	24.19	23.89	24.27	24.54	26.27	**26.28**
sym-4	30.34	30.40	24.08	24.09	24.18	24.56	26.20	**26.35**
sym-5	30.62	30.69	24.04	24.26	24.55	24.72	26.40	**26.56**
sym-7	30.40	30.47	24.07	24.34	24.39	24.70	26.29	**26.50**

TABLE 9: Average in PSNR for Wavelet Packet and Discrete Wavelet Transform of three video sequences of Miss America, Foreman and Carphone, using the best nine types of filters and Global threshold of 125

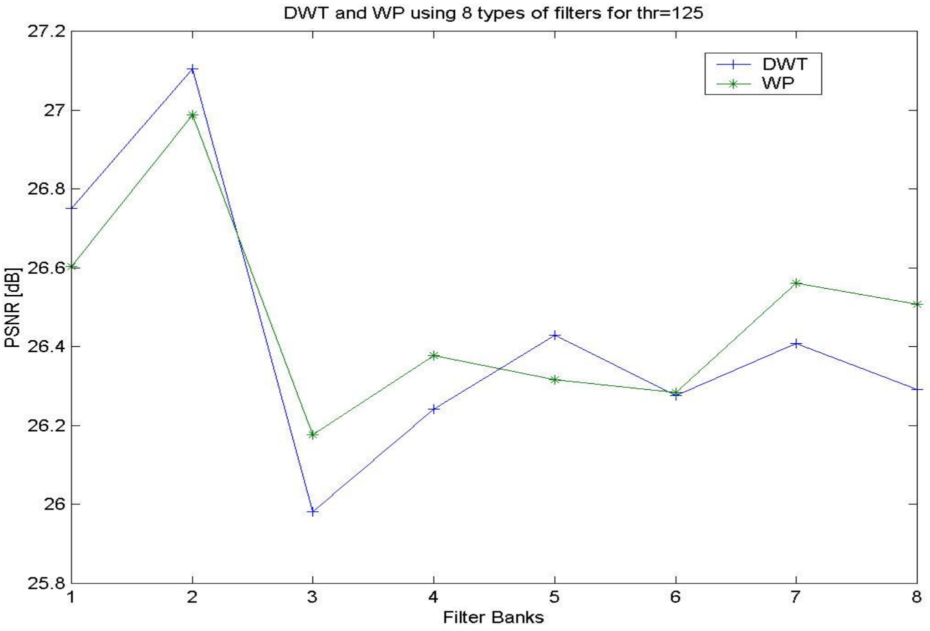


FIGURE 17: The Plot for Average PSNR for Wavelet Packet and Discrete Wavelet Transform of using the best nine types of wavelet filters and Global threshold of 125

No.	Video Sequences	(WP) dB	(DWT) dB	Difference (WP-DWT) - dB
1	Miss America	38.09	38.03	0.05
2	Suzie	34.59	34.58	0.01
3	Claire	37.21	37.26	-0.05
4	Mother & Daughter	33.77	33.69	0.07
5	Grandma	33.86	33.80	0.05
6	Carphone	34.88	34.53	0.34
7	Foreman	33.22	33.02	0.19
8	Salesman	32.38	32.19	0.18

TABLE 10: Average and difference in PSNR for Wavelet Packet and Discrete Wavelet Transform of eight video sequences, using Sym5 Filter for Low-Pass-Temporal and Haar Filter for High-Pass-Temporal Frequencies, at two Decomposition Levels, and Global threshold of 20

6. DISCUSSION

The choice of decomposition strategy between dyadic transformation of Discrete Wavelet Transform (DWT) and Wavelet Packet (WP) Transform has been examined and found that DWT decomposition is preferred over the WP due to the superior values of PSNR generated throughout the simulations for all the lower values of global threshold from thr=10 and thr=20. For thr=45, wavelet filter of sym-5 resulted the WP outperforms DWT, and further for thr=85 and thr=125. At thr=85, bior-4.4 and sym-5 resulted the WP to outperform DWT. Only for higher value of global threshold of 125, the WP shows better PSNR values compared to DWT. However for the results of Table 10, the average difference of WP and DWT is only marginal when global threshold value of 20 and wavelet filter bank of sym-5 is used.

7. CONCLUSION

This paper has addressed a number of challenges identified in 3D video compression based on wavelet transformation technique. The challenges appear when having to develop the coding scheme of 3D wavelet video coding, the question arises whether to transform the inter-frame followed by spatial filtering or vice versa. The substantial contribution on 3D wavelet coding is utilizing the spatio-temporal t-2D scheme, which has successfully exploited the wavelet transform technology and the best parameter strategies produces video output of high performance evaluated objectively using PSNR. It has shown that the selection of the parameters within the wavelet transform of periodic symmetric extension as the border distortion strategy, dyadic DWT, and level dependent thresholds have resulted to produce resulted in superior quality of the decoded video sequences. These parameters has been identified and further used in the proposed wavelet video compression (WVC) for the overall objective performance of the decoded video sequences. The spatio-temporal inter-frame temporal analysis of video sequences has been extended using Birge-Massaart strategy of wavelet shrinkage employing level-dependent quantization threshold. The objective evaluation pointed out that the PSNR values correlates better to different types of motion of the test video sequences especially to the Carphone sequence and using newly found Sym-5 filter-bank.

8. REFERENCES

1. Adami, N., Michele, B., Leonardi, R., and Signoroni, A. "*A fully scalable wavelet video coding scheme with homologous inter-scale prediction*". ST Journal of Research, 3(2):19-35, 2006
2. Adelson, E. H., and Simoncelli, E. "*Orthogonal pyramid transforms for image coding*". In Proceedings of SPIE Visual Communications and Image Processing II, 845:50-58, 1987
3. Albanesi, M. G., Lotto, I., and Carrioli, L. "*Image compression by the wavelet decomposition*". European Transactions on Telecommunication, 3(3):265-274, 1992.
4. Antonini, M., Barlaud, M., Mathieu, P., and Daubechies, I. "*Image coding using wavelet transform*". IEEE Transactions on Image Processing, 1(2):205-220, 1992
5. Antonio, N. "*Advances in Video Coding for hand-held device implementation in networked electronic media*". Journal of Real-Time Image Processing, 1:9-23, 2006
6. Ashourian, M.; Yusof, Z.M.; Salleh, S.H.S.; Bakar, S.A.R.A. "*Robust 3-D subband video coder*". Sixth International,Symposium on Signal Processing and its Applications, Vol 2:549–552, 2001
7. Brislawn, M. "*Classification of non-expansive symmetric extension transforms for multirate filter banks*". Applied and Computational Harmonic Analysis, 3(4): 337-357,1996.
8. Claudia, S. "*Decomposition strategies for wavelet-based image coding*". IEEE International Symposium on Signal Processing and its Applications (ISSPA), Vol. 2: 529-532, Kuala Lumpur, Malaysia, 2001.
9. Daubechies, I. "*Orthonormal bases of compactly supported wavelets*". Commun. Pure Applied. Math, 41:909-996, 1988.
10. Daubechies, I. "*The wavelet transform, time-frequency localization and signal analysis*". IEEE Trans. on Information Theory, 36(5):961-1005, 1990.
11. Daubechies, I. "*Ten Lectures on Wavelets*". Philadelphia, Pennsylvania: Society for Industrial and Applied Mathematics (SIAM), 1992

12. Daubechies, I. and Sweldens, W. "*Factoring wavelet transforms into lifting steps*". Journal of Fourier Analysis and Applications, 4(3):245-267, 1998.

13. Donoho, L. "*De-noising by soft-thresholding*". IEEE Transactions on Information Theory, 41(3):613-627, 1995.

14. Donoho, L. and Johnstone, I. M. "*Ideal spatial adaptation via wavelet shrinkage*". Biometrika, 81(3):425-455, 1994.

15. Donoho, L. and Johnstone, I. M. "*Minimax estimation via wavelet shrinkage*". The Annals of Statistics, 26(3):879-921, 1998.

16. George, F., Dasen, M., Weiler, N., Plattner, B., and Stiller, B. "*The wavevideo system and network architecture: design and implementation*". Technical report No. 44. Computer Engineering & Networks Laboratory (TIK), E7H, Zurich, Switzerland, 1998.

17. Golwelkar, A. V. and Woods, J. W. "*Scalable video compression using longer motion-compensated temporal filters*". VCIP 2003: 1406-1416, 2003

18. Hsiang, S. T. and Woods, J. W. "*Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank*". Journal of Signal Processing: Image Communication, Vol. 16: 705-724, 2001.

19. Karlsson, G. and Vetterli, M. "*Three dimensional subband coding of video*".In Proceedings of IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP), II:1100-1103, 1988.

20. Lewis, A. S. and Knowles, G. "*Video compression using 3D wavelet transforms*". Electronic Letters, 26(6):396-398, 1990.

21. Lewis, A. S. and Knowles, G. "*Image compression using the 2-D wavelet transform*". IEEE Trans. Image Processing, 1:244-250, 1992.

22. Luo, J. "*Low bit rater wavelet-based image and video compression with adaptive quantization, coding and post processing*". Technical Report EE-95-21. The University of Rochester, School of Engineering and Applied Science, Department of Electrical Engineering, Rochester, New York, 1995.

23. Mallat, S. G. 1989. "*A theory for multiresolution signal decomposition: the wavelet representation*". IEEE Transactions on Pattern Analysis and Machine Intelligence*,* 11(7):674-693, 1989.

24. Podilchuk, Jayant, N. S., and Farvardin, N. "*Three-dimensional subband coding of video*". IEEE Translation of Image Processing, 4(2): 125-139, 1995.

25. Shapiro, J. M. "*Embedded image coding using zerotrees of wavelets coefficients*". IEEE Transactions on Signal Processing, 41(12):3445-3462, 1993.

26. Strang, G., and Nguyen, T. "*Wavelets and Filter Banks*". Wellesley-Cambridge Press, Wellesley, MA, USA, 1997.

27. Taubman, D., and Zakhor, A. "*Multirate 3-D subband coding of video*". IEEE Transactions on Image Processing, 3(5): 572-588, 1994

28. Vass, J., Zhuang, S., Yao, J., and Zhuang, X. "*Mobile video communications in wireless environments*". In Proceedings of IEEE Workshop on Multimedia Signal Processing, Copenhagen, Denmark: 45-50, 1999

29. Wallace, G. K. "*The JPEG still picture compression standard*". Comm. ACM, 34(4):30-44, 1994.

30. Wang, X., and Blostern, S. D. 1995. "*Three–dimensional subband video transmission through mobile satellite channels*". In Proceedings of International Conference on Image Processing, Vol. 3, pp. 384-387, 1995.

Use of Wavelet Transform Extension for Graphics Image Compression using JPEG2000 Standard

Singara Singh singara@thapar.edu
Lecturer, School of Mathematics
& Computer Applications
Thapar University, Patiala-147004-India

R. K. Sharma rksharma@thapar.edu
Professor, School of Mathematics
& Computer Applications
Thapar University, Patiala-147004-India

M.K. Sharma mksharma@thapar.edu
Assistant Professor, School of Mathematics
& Computer Applications
Thapar University, Patiala-147004-India

ABSTRACT

The new image compression standard JPEG2000, provides high compression rates for the same visual quality for gray and color images than JPEG. JPEG2000 is being adopted for image compression and transmission in mobile phones, PDA and computers. An image may contain the formatted text and graphics data. The compression performance of the JPEG2000 behaves poorly when compressing an image with low color depth such as graphics images. In this paper, we propose a technique to distinguish the true color images from graphics images and to compress graphics images using a wavelet transform extension under JPEG2000 standard that will improve the compression performance. This method can be easily adapted in image compression applications without changing the syntax of compressed stream of JPEG2000.

Keywords: JPEG2000, JPEG, Entropy, Discrete Wavelet Transform (DWT), Down-sampling Factor Style (DFS)

1. INTRODUCTION

JPEG2000 is state of the art image coding standard that resulted from the joint efforts of International standard Organization (ISO) and International Telecommunications Union (ITU). The Part 1 of the JPEG2000 standard describes core coding system [1]. But, when an image contains graphics type data, such as clips or logos, the performance of the JPEG2000 degrades due to the fact that these graphics images are either using color palette with low color depth or containing objects with solid areas and a limited number of colors. A general lossless image compression of JPEG2000 standard contains two steps. The first step, image de-correlation step is used to reduce the spatial redundancy of an image. This step provides the more compact representation of the image. The second step is the entropy encoding in which the de-correlated image is processed by an entropy encoder using some variable length coding techniques. But in case of graphics type images, the de-correlation step of JPEG2000 is actually degrading the image compression performance.

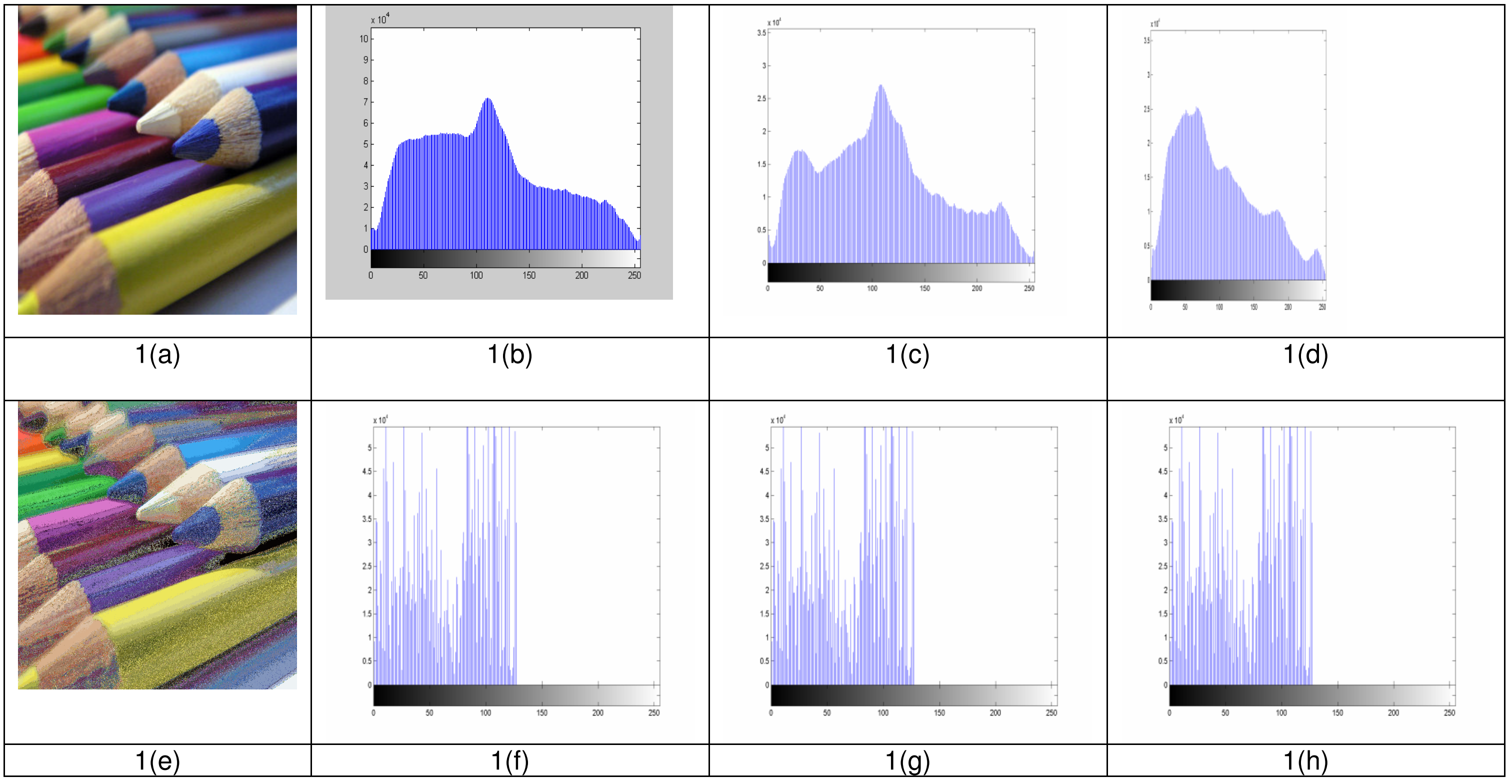
Some techniques have been proposed to compress the graphics images. Banerje *et. al.*[8] proposed a post processing technique to minimize the ringing distortions at low bit rates. Although post-processing technique gives better result but it adds to the complexity to the JPEG2000 codec. Tsai *et al.* [6] used the concept of entropy to distinguish the graphics images and true color images, and then compress the graphics images by bypassing the DWT step of the JPEG2000 codec.

In this paper, we proposed a simple method which still works under the framework of the JPEG2000 to improve the compression performance for the graphics type images. The idea is to use the different wavelet decomposition structure than the normal dyadic decomposition structure for the graphics images. The entropy is used to distinguish a graphics type images and true color images.

The rest of the paper is organized as follows. In Section 2, we present our key observation regarding the compression performance issue and review the concept of entropy of an image. The proposed method will be described in section 3. Simulation results of the proposed method are shown in section 4. Conclusion and future work will be discussed in section 5.

2. KEY OBSERVATIONS

Fig. 1(a) shows a true color image with 256 colors per channels. The histograms of Red, Green and Blue components are shown in Fig. 1(b) - (d). The image has very nice color distributions. In order to observe the behavior of a graphics image with low colors, we convert the pencil image into an image, as shown in Fig. 1(e), with 128 colors (as described in [5]). As shown in Fig. 1(f) – (h), the RGB histograms are very discrete. This observation inspired the idea of using the "entropy" to distinguish the true color image and graphics images and compress the graphics images using an wavelet transform extension method under JPEG2000 standard that will improve the compression performance. In the next sub-section, we will provide the quick overview for JPEG2000 standard.



1(a) 1(b) 1(c) 1(d)

1(e) 1(f) 1(g) 1(h)

FIGURE 1: (a) Original Pencil image with 256 colors (b)- (d) Histograms of (a)'s color channels; (e) image with reduced color depth (128 RGB colors). (f) – (h) Histograms of (e)'s color channels.

2.1 OVERVIEW OF JPEG2000 STANDARD

In this section, we present an overview of JPEG2000 with emphasis on concepts related to the ideas presented in this paper. In JPEG2000 [1-4], the first stage consists of (optionally) dividing the input image into non-overlapping rectangular tiles. For multi-component images, an optional component transform can be applied to decorrelate the components. These transformed components are known as tile components. An irreversible or reversible wavelet transform is then applied to each tile component to decorrelate the samples of the image, which is to be compressed. The irreversible transformation is used for lossy compression and reversible transformation for lossless compression. Each component of a tile is independently transformed by the DWT [11]. For lossy compression, 9/7 irreversible wavelet transformation is used and for lossless compression, 5/3 reversible wavelet transformation is used. The wavelet transform creates the decomposition levels. These decomposition levels are subbands of coefficients that characterize the local frequency of the tiles. For lossy compression, these subbands are then quantized. After quantization, each subband is divided into non overlapping rectangular blocks, called code blocks. Code blocks are the basic coding unit for entropy coding. Encoding of the blocks is done independently and the size of the block is typically 32 x 32 or 64 x 64. The entropy encoding in JPEG2000 consists of a fractional bit plane coding (BPC) and binary arithmetic coding (BAC). The combination of both coding is also known as Tier-1 coding in the standard. BPC has three passes in each bit plane: Significance Propagation Pass, Magnitude Refinement Pass, and Cleanup Pass. Each of the pass generates context models and the corresponding binary data. The output of the BPC and BAC produces the compressed bit stream. So an independent bit stream is generated for each code block. All these bit streams are combined into a single bit stream using Tier-2 coding, which is based on the output of the rate distortion optimization [4].

The structure of a simple JPEG2000 codestream contains the precincts and packets. A precinct is formed by grouping together the codeblocks that corresponding to a particular spatial location at a given resolution. Compressed data from each precinct are arranged to form a packet. Each packet contains a header and a body. The packet header contains information about the contribution of each codeblock in the precinct to the packet, while the body contains compressed coding passes from the codeblocks. Packets that belong to a particular tile are grouped together to form a tile stream, and tile streams are grouped together to form the JPEG2000 codestream. Similar to packets, tile streams are composed of a header and a body. The EOC marker indicates the end of the codestream.

2.2 ENTROPY OF AN IMAGE

In Information theory [5], entropy is the expected length of a binary code over all possible samples of a source. The entropy is defined as

$$E= - \sum_{i=0}^{N} p(a_i)\; log_2\; p(a_i)$$

Where N is the no of samples of the image and $p(a_i)$ is the probability of occurrence of samples a_i in the image.

The entropy provides a bound for compression that can be achieved. The entropy of an image will be a good indication for distinction of true color images or graphics images.

3. PROPOSED METHOD

JPEG2000 coding standard uses the DWT transform for the purpose of de-correlation of image pixels. We applied JPEG2000 part-2 downsampling factor styles (DFS) [2] features to implement the extension of wavelet decomposition structure. In our method, the first DWT transform level splits the image components only in vertical direction against the JPEG2000 decomposition in which the DWT transform decomposes the image in horizontal and vertical directions.

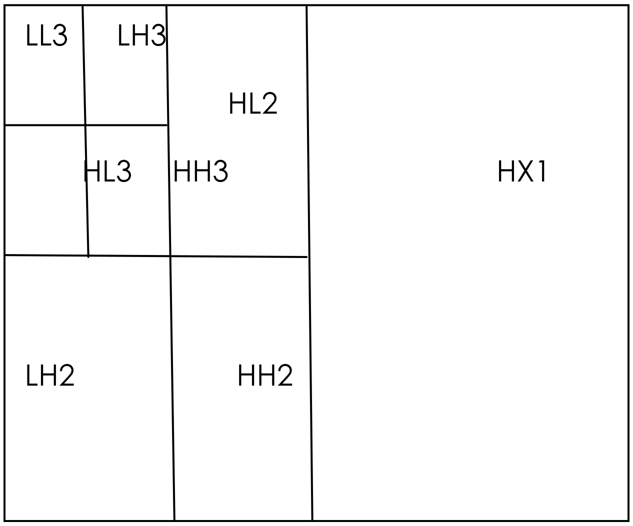


FIGURE 2: Non Dyadic wavelet decomposition

Subsequent DWT levels use full horizontal and vertical splitting for all image components, as shown in Figure 2. Here, the first level DWT transform is performed only on rows of the input image and it decomposes image into LX1 and HX1, where H stands for higher subimage , L stands for lower subimage and X denotes that no transform was performed on the columns. The second level decomposes LX1 into LL2, LH2, HL2 and HH2. No operation is carried on HX1. Then subsequent wavelet transformations split the LL portion into four subbands, as like a dyadic decomposition. The information of this non dyadic decomposition is stored into SIZ marker of the JPX files. In this method, there will be no extra overhead at the decoder side.

4. EXPERIMENTAL RESULTS

To observe the impact of different color depths, we gradually reduced the color depth from 256 colors to 4 colors per channel. We first applied the DWT decomposition method of JPEG2000 with five levels of using the kakadu [7] software. Then we applied the extension of DWT decomposition under the JPEG2000 standard. In both methods, the code blocks are 64x64 sizes. The proposed method outperforms the JPEG2000 standard, when the input image has less no. of colors.

Table1: Compression size comparisons and entropy values

Image No of Colors			256 colors	128 colors	64 colors	32 colors	16 colors	8 colors	4 colors
Pencils.tif (3.12 MB) [9]	Entropy		7.82	6.66	5.78	3.80	3.83	2.87	1.96
	Compressed Size	JPEG2000	2189	1532	1054	689	455	249	142

	(Kilo Bytes)	Proposed Method	2250	1382	953	627	416	230	140
Icon.tif (509 KB) [10]	Entropy		4.92	4.05	3.53	2.89	2.70	2.22	1.68
	Compressed Size(Kilo Bytes)	JPEG2000	120	61	49	36	24	16	16
		Proposed Method	124	59	46	33	21	14	14

It is clear that for the images having very limited number of colors, the proposed method performs better than the JPEG2000 standard.

5. CONCLUSION & FUTURE WORK

Based on the simulation results, it is clear that the performance of JPEG2000 degrades when it compresses the images having low color depth such as graphics images. The performance of JPEG2000 improves significantly using JPEG2000 Part-2 downsampling factor styles. We used the entropy to distinguish the true color image or graphics images. The proposed method can be easily adapted in compression applications for graphics or drawing type images without changing the syntax of compressed bit stream of JPEG2000. This method can be extended to compress the video sequences under the JPEG2000 standard.

6. REFERENCES

1. JPEG2000 Part1: Core Coding System, Final Committee Draft (ISO/IEC FCD 15444-1), ISO/IEC JTC1/SC29/WG1 N11855, March 2000.

2. JPEG2000 Part2: JPEG2000 Extension, Final Committee Draft (ISO/IEC FCD 15444-2),November 2001.

3. D. Taubman and M. Marcellin, JPEG2000: Image Compression Fundamentals, Standards and Practice, Boston: Kluwer Academic Publisher, 2002.

4. D. Taubman, “High performance scalable image compression with EBCOT”, IEEE Transaction on Image Processing, Vol. 9, No. 7, pp. 1158-1170, July 2000.

5. R C Gonzalez, and R.E. Woods, “Digital Image Processing”, 2nd Edition, Pearson Education.

6. Ping Sing Tsai, and Ricardo Suzuki, “Graphics Image Compression Using JPEG2000”, IEEE 2008 Congress on Image and Signal Processing, pp. 603-607, 2008.

7. www.Kakadusoftware.com

8. Serene Banerjee and Brian L Evans, “Tuning JPEG2000 Image Compression for Graphics Region”, Fifth IEEE Southwest Symposium on Image Analysis and Interpretation, pp 1- 5, 2002.

9. Pencil Image (http://www.stpaulcareers.umn.edu/img/assets/16141/Graphic%20Design145x100.jpg).

10. Icon Image (http://graphics.cs.brown.edu/games/G3D/icon.jpg).

11. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transform”, IEEE Transaction on Image Processing, Vol. 1, pp. 205-220, April 1992.

COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA