SWF Image Compression by Evaluating objects compression ratio

Hind Rostom Mohammed
Assistant professor
Kufa University
College of mathematical and computer sciences
Computer Department

Ameer Abd Al-Razaq
Computer Engineer
Kufa University – Computer Center

Abstract

This work discusses the compression objects ratio for Macromedia Flash File (SWF) Image by Wavelet functions for compression and there effect for Macromedia Flash File (SWF) Images compression . We discusses classification objects in Macromedia Flash (SWF) image in to nine types objects Action, Font, Image, Sound, Text, Button, Frame, Shape and Sprite. The work is particularly targeted towards wavelet image compression best case by using Haar Wavelet Transformation with an idea to minimize the computational requirements by applying different compression thresholds for the wavelet coefficients and these results are obtained in fraction of seconds and thus to improve the quality of the reconstructed image. The promising results obtained concerning reconstructed images quality as well as preservation of significant image details, while, on the other hand achieving high compression rates and better image quality while DB4 Wavelet Transformation higher compression rates ratio without kept for image quality .

Key word: Image Compression Macromedia Flash File, compression objects .

1.Introduction

Data compression techniques help in efficient data transmission, storage and utilization of hardware resources. Uncompressed multimedia requires considerable storage capacity and transmission bandwidth. Despite rapid progress in mass- storage density, processor speeds and digital communication system performance, demand for data storage capacity and data transmission bandwidth continues to outstrip the capabilities of available technologies. The recent growth of intensive digital audio, image and video (multimedia) based applications, have not only sustained the compression of such signals central to signal storage and digital communication technology [3].

JPEG is a commonly used standard to compress digital color images. However, the various settings used during JPEG compression and decompression are not standardized:The following JPEG settings can be chosen by the user such as an imaging device: (1) the color space used to independently compress the image’s three color planes; (2) the sub sampling employed on each color plane during compression and the complementary interpolation used during decompression; and (3) the quantization table
used to compress each color plane[6]. The Macromedia Flash File Format (SWF) was designed as a very efficient delivery format and not as a format for exchanging graphics between graphics editors. There are two categories of tag in Macromedia Flash (SWF):

1- Definition Tags are used to define the content of the Macromedia Flash (SWF) movie – the shapes, text, bitmaps and sounds that are used in the movie. Each definition tag assigns a unique ID called a character to the content it defines. The player then stores the character is a repository called the dictionary.

2-Control Tags are used to manipulate characters in the dictionary, and control the flow of the movie. For example, control tags can be used to place characters on the screen, start sounds, and to tell the player to go to another frame in the movie. Control tags can be divided into three categories: display list, control, and action.

Generally speaking, tags in a Macromedia Flash (SWF) file can occur in any order. However, there are a few rules that must be observed:

1)A given tag should only depend on tags that come before it. A tag should never depend on a tag that comes later in the file.

2)A definition tag that defines a character must occur before any control tag that refers to that character.

3)Streaming sound tags must be in order. Out of order streaming sound tags will result in the sound being played out of order.

4)The End tag is always the last tag in the Macromedia Flash (SWF) file.

A general approach to compression of a compound image will include 3 major steps:

1) image segmentation into the regions of similar data types,

2) selection of the best compression algorithm for each region,

3) bit allocation (quality setting) among various regions/compression algorithms.

Advances in Wavelet Transforms and Quantization methods have produced algorithms capable of surpassing image compression standards, like the Joint Photographic Expert Group (JPEG) algorithm. The recent growth of data intensive multimedia based applications have not only sustained the need for more efficient ways to encode the signals and images but also have made compression of such signals central to storage and communication technology. The JPEG2000 standard employs wavelet for compression due to its merits in terms of scalability, localization and energy concentration .

It also provides the user with many options to choose to achieve further compression[4] .

This paper presents the result of image compression for Haar and DB4 wavelets. It is concluded that selection of proper advanced Haar is one of the important parameters of image compression.

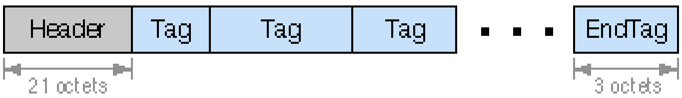
2.Wavelet image compression

Wavelet compression uses band pass filters to separate an image into images with low or high spatial frequencies. Low frequency images are those in which brightness change is gradual, for example, flat or rounded background areas. Such images appear soft and blurry. Higher frequency band images are crisp and sharp edged. Adding the frequency band images back together should reconstruct the original input image; perfectly if the processing is perfect [5]. Object-based segmentation is divided into regions, where each region follows exact object boundaries. An object may be a photograph, a graphical object, a letter, etc. In principle, this method may provide the best compression, since it provides the best match between a data type and the compression method most suitable for this data type. In reality, the best compression may not be achievable for the following reasons. Coding the object boundaries requires extra bits, and the typical algorithms, used for lossy image compression, are designed to operate on rectangular objects. They can operate on objects with nonrectangular boundaries, but the compression performance will suffer. Complexity is another drawback of this method, since precise image segmentation may require the use of very sophisticated segmentation algorithms. A pixel data stream from an input image is divided into several sub-bands by a tree of band pass filters. Each filter allows only a specific band of frequencies to pass. The filters may be analog or digital, but since neither kind is perfect some image distortion can be expected even at this stage [6].

3.Test, Verification and Efficiency Structure for Macromedia Flash File compression.

We can classified nine types objects in Macromedia Flash (SWF) image:
1-Action 2-Font 3-Image 4-Sound 5-Text 6-Button 7-Frame 8-Shape 9-Sprite

Following the header are a series of tagged data blocks. All tags share a common format, so any program parsing a Macromedia Flash (SWF) file can skip over blocks it does not understand. Data inside the block can point to offsets within the block, but can never point to an offset in another block. This enables tags to be removed, inserted, or modified by tools that process a SWF file.



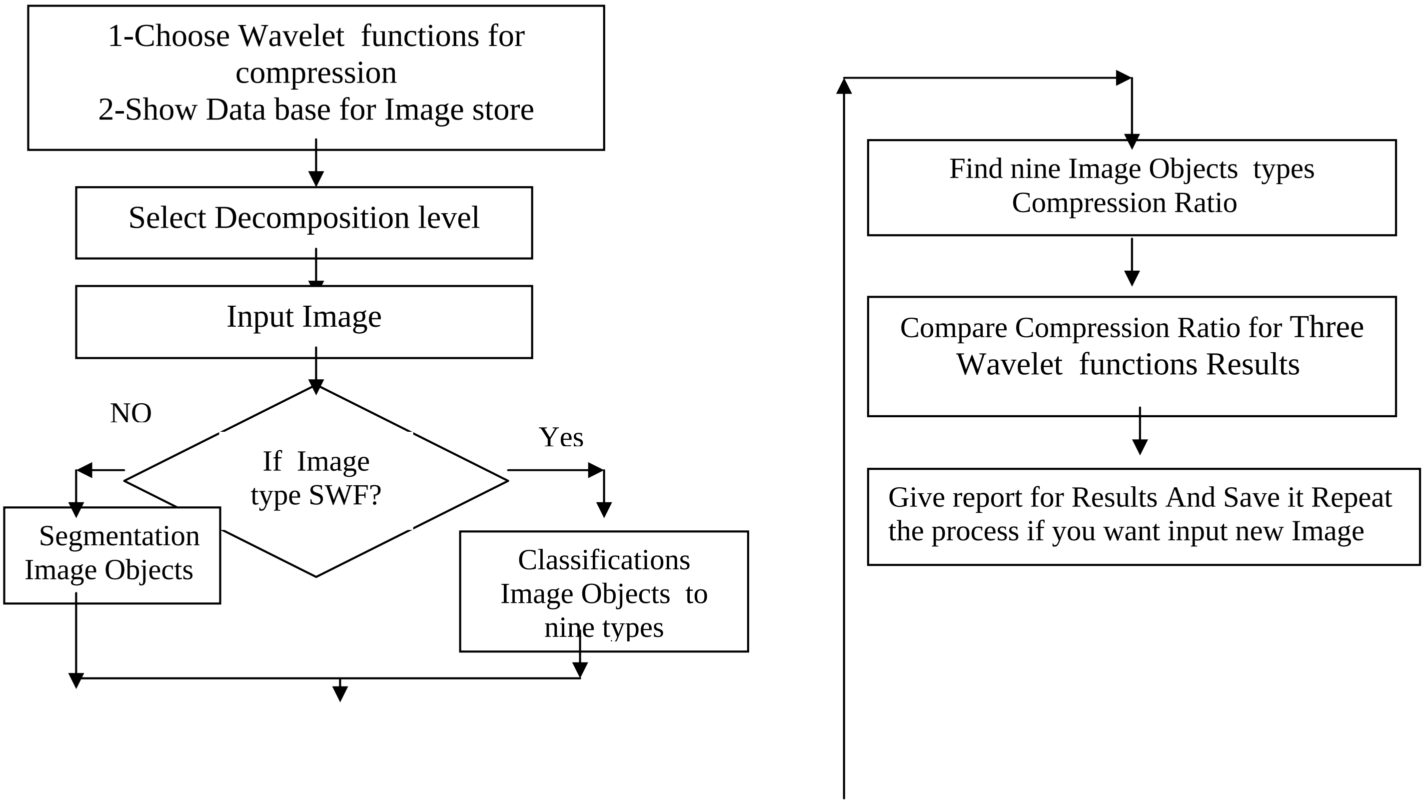
figure(1): Macromedia Flash (SWF) File Structure

These operations correspond to the following filtering processes:

1) Upper left: 2-D low pass filter (Lo-Lo).
2) Upper right: horizontal high pass and vertical low pass filter (Hi-Lo).
3) Lower left: horizontal low pass and vertical high pass filter (Lo-Hi).
4) Lower right: 2-D high pass filter (Hi-Hi).

To apply this transform to a complete image, we group the pixels into 8 × 8 blocks. Figure(2) show the two images type Macromedia Flash (SWF), To view the result, all the upper left components in Figurer (3.a) of the 8 × 8 blocks in y were grouped together to form the upper left sub image and the same for the components in the other three positions.

It is clear from Figurer (3.c) that the most of the energy is contained in the upper left (Lo-Lo) sub image and the least energy is in the lower right (Hi-Hi) sub image. The upper right (Hi-Lo) and the lower left (Lo-Hi) sub image contains the edges. The system implementation that used in this study is shown in Figure(2).



Figure(2): The system implementation that used in this study

SWF1 SWF2

Figure(3): Macromedia Flash (SWF)

(a) (b) (c)

Figure(4): Original image (a) was decomposed using wavelet image decomposition by the Haar transform and result (b) Synthesized Image(c) Image after Matlab Haar wavelet decomposition algorithm by Matlab wavelet decomposition function. The algorithm of image features extraction involves.

a) decomposition, using one level DWT with the Haar transform, of each sub image Ci,j of size 8 × 8 taken from the upper left corner

b) computation of the co-occurrence matrix features energy and contrast
obtained from each sub image Ci,j
c) forming new feature matrices

$$\text{Contrast}=\sum_{i,j=1}^{n}(i-j)^2\,C_{ij} \qquad \ldots\ldots(1)$$

An image histogram is type of histogram which acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance.

Building a wavelet-based histogram is a three-step procedure:

1. Preprocessing In a preprocessing step, we form the extended cumulative data distribution T^{C+} of the attribute X, from the original data or from a random sample of the original data.

2. Wavelet Decomposition We compute the wavelet decomposition of T^{C+}, obtaining a set of N wavelet coefficients.

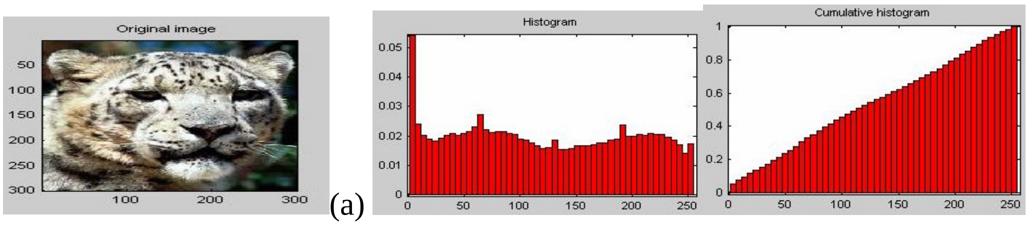
3. Pruning We keep only the m most significant wavelet coefficients, for some m that corresponds to the desired storage usage. The choice of which m coefficients we keep depends upon the particular pruning method we use.

After applying the above algorithm, we obtain m wavelet coefficients. The values of these coefficients, together with their positions (indices), are stored and serve as a histogram for reconstructing the approximate data distribution in the on-line phase (query phase). The cumulative histogram is a variation of the histogram in which the vertical axis gives not just the counts for a single bin, but rather gives the counts for that bin plus all bins for smaller values of the response variable[1]. The advantages of Haar Wavelet transform compare with Other Wavelet transform as follows:

1. Best performance in terms of computation time.

2. Computation speed is high. 3. Simplicity
4. HWT is efficient compression method.

5. It is memory efficient, since it can be calculated in place without a temporary array.



(a) (b) (c)

Figure(5): (a)Original image (b) Histogram   (c) Cumulative histogram

4. Results and Discussions

The project deals with the implementation of the objects SWF image by haar wavelet compression techniques and a comparison over various SWF input images. We first look in to algorithm, Information for two Macromedia Flash shown in table(1), information for two sound files in Macromedia Flash (SWF) File and compression ratio shown in Table(3), results of wavelet compression technique by calculating their comparison ratios and then find comparison ratios results for each objects SWF image which is shown in figure(10,11).We Evaluation objects compression ratio Information in Macromedia Flash (SWF) File and compression ratio shown in table(4) and higher Compression ratio to Macromedia Flash (SWF) File by compression ratio for objects shown in Table(5). The Algorithm for our system is:

1. Read the image from the user.
2. Apply 2D DWT using haar wavelet over the image
3. For the computation of haar wavelet transform, set the threshold value 25, 10, 5 and 1% ie., set all the coefficients to zero except for the largest in magnitude 25, 10, 5 and 1%. And reconstruct an approximation to the original image by apply the corresponding inverse transform with only modified approximation coefficients.
4. This simulates the process of compressing by factors of ¼, 1/10, 1/20, 1/100 respectively.
5. Segmentation the Macromedia Flash File (SWF) Image by its objects and Display the resulting objects and comment on the quality of the images.
6. Calculate Compression Ratios for each objects values in Macromedia Flash File (SWF) Image of different Compression Ratios for corresponding Reconstructed images.
7. To compute the haar wavelet transform, set all the approximation coefficients to zero except those whose magnitude is larger than 3 sigma.
8. This same case is applicable to detail coefficients that is horizontal, vertical & diagonal coefficients.
9. Reconstruct an estimate of the original image by applying the corresponding inverse transform.
10. Display and compare the results by find some factor for all image with each objects and find Compression Ratio(CR) for SWF image and its Compression Ratio(CR segmentation objects.
11. The same process is repeated for various images and compare its performance.

The compression ratio is equal to the size of the original image divided by the size of the compressed image. This ratio gives an indication of how much compression is achieved for a particular image. Our algorithm have a typical range of compression ratios that they can achieve over a variety of images and sound in Macromedia Flash (SWF). Because of this, it is usually more useful to look at an average compression ratio for a particular method. The compression ratio typically affects the picture quality. Generally, the higher the compression ratio, the poorer the quality of the resulting image. The tradeoff between compression ratio and picture quality is an important one to consider when compressing images. Furthermore, some compression schemes produce compression ratios that are highly dependent on the image content.

The compression efficiency is defined by the parameter compression ratio, CR and is given by,

Compression Ratio(CR)=Uncompressed file size /Compressed file size ……(2)

Or, CR = Actual bpp / Reduced bpp ……………………………………….(3)

If k=Image information then we can computed three factors:

1) IB=k.Width*k.Height*k.BitDepth/8

2) CB=k.FileSize

3) CR=IB/CB

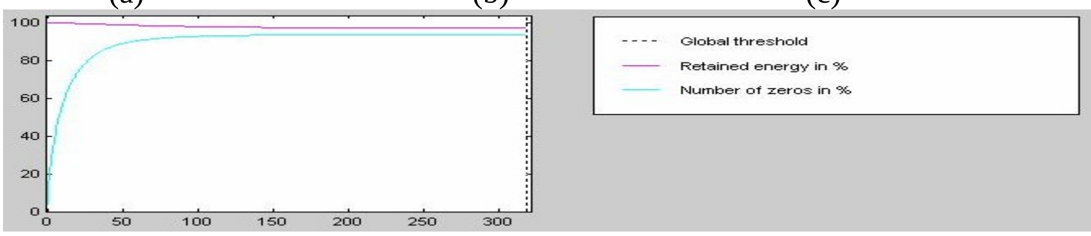
Or, For example, if actual bpp = 8 and reduced bpp = 0.5 then CR = 16:1. If original data =512 x 512 x 8 = 1.497152 Bits and compressed data = 1.497152 Bits then CR = 1000:1.

Table(1):Information for two Macromedia Flash (SWF) SWF1 and SWF2

SWF Type	File Size	Width	Height	Bit Depth	IB	CB	CR
SWF1	17105	300	300	24	270000	17105	15.7849
SWF2	16241	300	300	24	270000	16241	16.6246

The synthesized image is a restored version of good quality of the common underlying original image.

(a) (b) (c)



(d)

Figure(6): (a) Original image, (b),(c) two ratio compression for Original image by Haar and DB4 wavelet ,(d)Graph for energy and number of zeros for figure(7.c)

Table (2): Information for figure (7 .c)

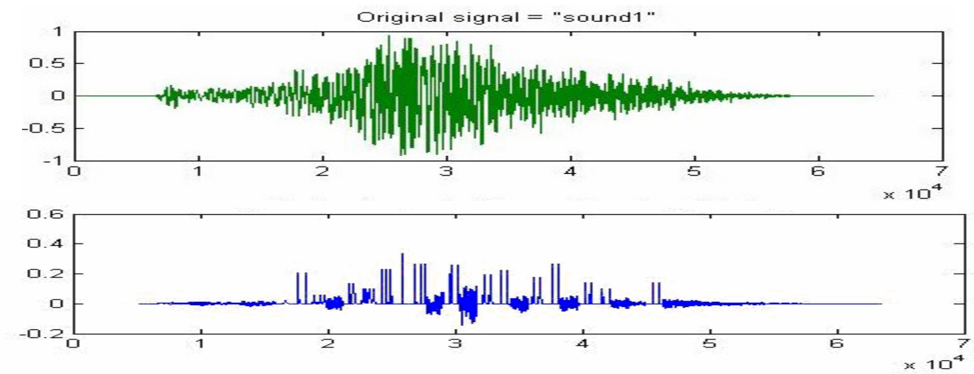
Number of Zeros	Threshold	Energy
93.75%	317.8	97.23%

Speech compression may mean different things:

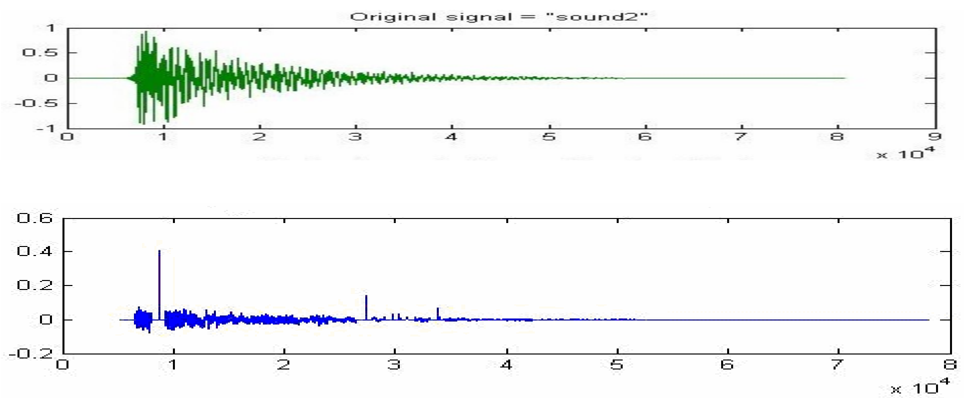
1) Speech encoding refers to compression for transmission or storage, possibly to an unintelligible state, with decompression used prior to playback.

2)Time-compressed speech refers to voice compression for immediate playback, without any decompression (so that the final speech sounds faster to the listener).

For a given memory size, compression allows longer messages to be stored than otherwise. Digital speech signals are sampled at a rate of 8000 samples/sec. Typically, each sample is represented by 8 bits (using mu-law). This corresponds to an uncompressed rate of 64 kbps (kbits/sec). With current compression techniques, it is possible to reduce the rate to 8 kbps with almost no perceptible loss in quality. Further compression is possible at a cost of lower quality.



(a)

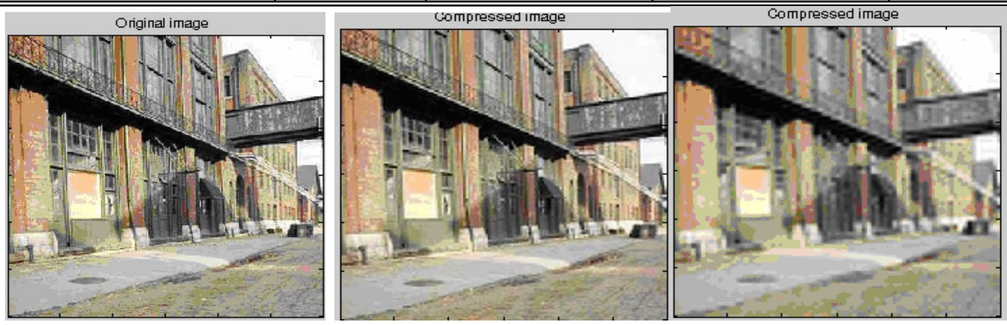


(b)

Figure(7): (a) Sound1 file and its compression in Macromedia Flash (SWF) File, (b) Sound2 file and its compression in Macromedia Flash (SWF) File

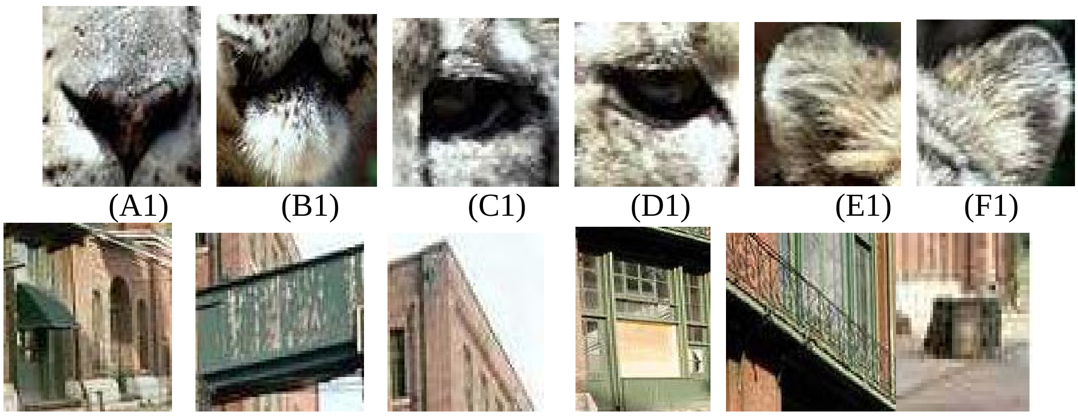
Table(3): Information for two sound files in Macromedia Flash (SWF) File and compression ratio

Object Name	File Size	Time	Number of bytes	CR
Sound1	252 KB	1.46 seconds	258.110 bytes	2.0322580645
Compression Sound1	124KB	1.44 seconds	127.052 bytes	
Sound2	315 KB	1.83 seconds	322.622 bytes	2.0322580645
Compression Sound2	155 KB	1.80 seconds	158.804 bytes	



a-Original image b- compression by Haar c-compression by DB4

Figure(8): Greatest SWF2 and it's compression

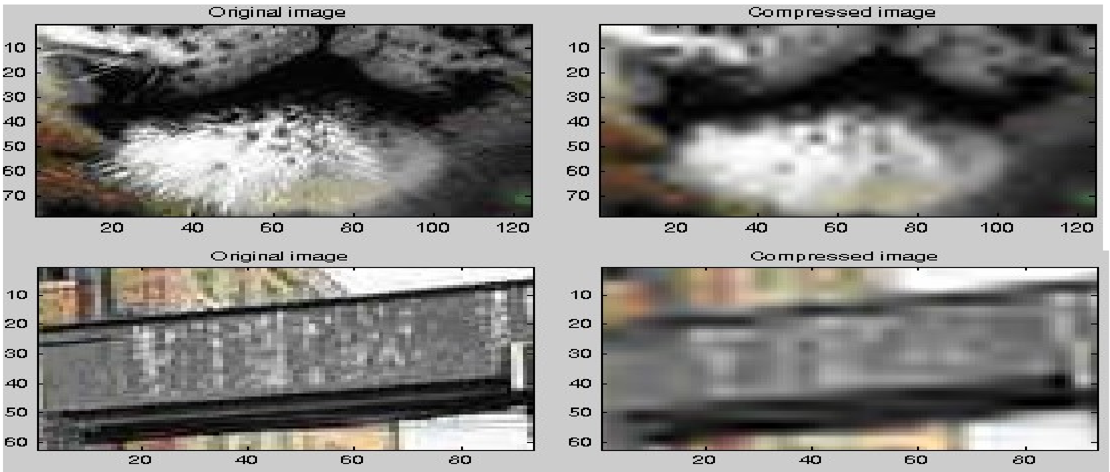


(A1) (B1) (C1) (D1) (E1) (F1)

(A2) (B2) (C2) (D2) (E2) (F2)

Figure(9):Objects in Macromedia Flash (SWF) File

Motion detection results in SWF file containing objects with sizes ranging from 44*44 to 125*80 for high Width and 67*166 for high Height ,Bit Depth =24 objects are presented in table(4).



Figure(10):Objects in Macromedia Flash (SWF) File and there compression

Table(4): Evaluation objects compression ratio Information in Macromedia Flash (SWF) File and compression ratio

Object Name in SWF	File Size	Width	Height	Bit Depth	IB	CB	CR
(A1)	2.32 KB	70	80	24	16800	2378	7.0648
(B1)	3.18 KB	125	80	24	30000	3258	9.2081
(C1)	1.80 KB	60	60	24	10800	1847	5.8473
(D1)	1.86 KB	60	60	24	10800	1905	5.6693
(E1)	1.92 KB	65	65	24	12675	1974	6.4210
(F1)	2.03 KB	65	65	24	12675	2082	6.0879
(A2)	2.54 KB	63	94	24	17766	2602	6.8278
(B2)	2.45 KB	92	62	24	17298	2514	6.8807
(C2)	1.54 KB	64	57	24	10944	1581	6.9222
(D2)	3.75 KB	67	166	24	33366	3841	8.6868
(E2)	3.30	113	79	24	26781	3380	7.9234
(F2)	1.23 KB	44	44	44	5808	1264	4.5949

To Evaluation objects compression ratio Macromedia Flash (SWF) File with compression ratio Macromedia Flash (SWF) File (compare table(1) with table(4)).

We obtained higher Compression ratio to Macromedia Flash (SWF) File by using total compression ratio for objects Table(5) show the results.

Table(5): higher Compression ratio to Macromedia Flash (SWF) File by compression ratio for objects

SWF Type	Compression ratio for SWF type	Object Name in SWF	File Size	Compression ratio for object	Total Compression ratio for objects
SWF1	15.7849	(A1)	2.32 KB	7.0648	40.2984
		(B1)	3.18 KB	9.2081	
		(C1)	1.80 KB	5.8473	
		(D1)	1.86 KB	5.6693	
		(E1)	1.92 KB	6.4210	
		(F1)	2.03 KB	6.0879	
SWF2	16.6246	(A2)	2.54 KB	6.8278	41.8358
		(B2)	2.45 KB	6.8807	
		(C2)	1.54 KB	6.9222	
		(D2)	3.75 KB	8.6868	
		(E2)	3.30	7.9234	
		(F2)	1.23 KB	4.5949	

5.Conclusion

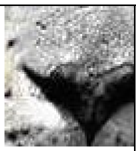
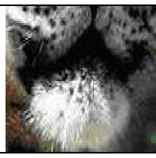
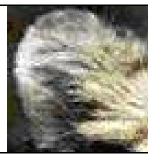
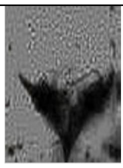
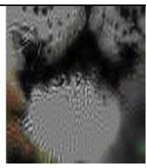
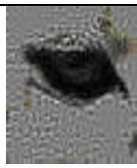
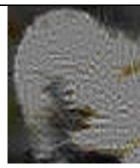
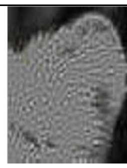
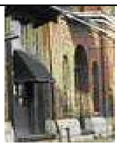
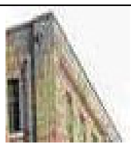
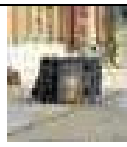
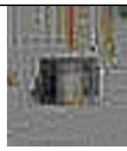
This paper reported is aimed at developing computationally efficient and effective algorithm for objects SWF image compression using Haar wavelet techniques. So this proposed algorithm developed to compress the SWF image so faster . The algorithm find Evaluation objects compression ratio performance by testing 14 Macromedia Flash (SWF) images and two sound files in each one ie. Macromedia Flash (SWF) information(269 KB (275.838 bytes)) .

The promising results obtained concerning reconstructed SWF image quality as well as preservation of significant image details, while on the other hand achieving high compression rates. Results shows that reduction in encoding time with little degradation in image quality compared to existing methods. While comparing the developed method with other methods compression ratio is also increased. Some of the applications require a fast image compression technique but most of the existing technique requires considerable time.

So this proposed algorithm developed to compress the objects SWF image so fastly. The main bottleneck in the compression lies in the search of domain, which is inherently time expensive. This leads to excessive compression time.

The experimental results indicate its correctness and efficiency , best performance in terms of computation time, Computation speed is high, Simplicity, it is memory efficient, since it can be calculated in place without a temporary array and a compromise between high compression ratio and good Evaluation objects compression ratio is well obtained.

Table(6) :higher objects compression ratio with out kept image quality by DB4 Wavelet Transformation

Original Objects For SWF1						
Higher compression ratio	42.4105	50.0700	38.8587	38.5386	47.5824	43.1576
Objects After compression						
Original Objects For SWF2						
Higher compression ratio	44.1224	43.9813	58.0420	54.8400	50.6834	49.4135
Objects After compression						

References

[1] David Klein K. Fritz Thompson," Accessible Video in a Diverging Web Environment" CSUN Conference: Technology and Persons with Disabilities Los Angeles, CA March 16, 2005.

[2] D. Gall, "MPEG: A video compression standard for multimedia applications, " Commun. ACM, Vol(34), pp. 46-58, Apr. 1991.

[3] G. Sadashivappa, and K. V. S. AnandaBabu,, Evaluation of Wavelet Filters for Image Compression World Academy of Science, Engineering and Technology,Vol(51), No.(1), pp . 131-137, 2009.

[4] G. K. Kharate,V. H. Patil and N. L. Bhale ,"Selection of Mother Wavelet For Image Compression on Basis of Nature of Image,", JOURNAL OF MULTIMEDIA, Vol(2), NO.(6), pp. 44-51, 2007.

[5] Kamrul Hasan TalukderI and Koichi HaradaII," Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image",IAENG International Journal of Applied Mathematics, 36:1, IJAM_36_1_9,2007.

[6] Robert L. Cook and Tony DeRose, "Wavelet Noise, ACM Transactions on Graphics", July 2005, Vol(24), No. (3), 2005.

[7] Z. Fan and R. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," IEEE Trans. Image Processing, Vol(12), pp. 230–235, 2003.

الخلاصة

ناقش العمل نسبة ضغط الكيانات في الصورة الفلاشية(البطاقات الفلاشية نوع Macromedia Flash File (SWF)) باستخدام تحويلات wavelet وتأثيرها في ضغط الصور الفلاشية Macromedia Flash File (SWF). ناقش العمل عملية تصنيف تسع أصناف من الكيانات داخل الصورة الواحدة SWF Image قسمت كالاتي:
الفعالية ، الخط ،الصورة، الصوت،النص،Button ،الإطار ،الشكل وSprite اظهرت النتائج العملية ان طريقة Haar Wavelet Transformation هي الأفضل لضغط الصور من نوع SWF Image حيث طبقت فيها مختلف أنواع قيم العتبة تم الحصول عليها بوقت اقل بمعنى أخر تكون فيها نسبة عالية من الضغط مع الحفاظ على نوعية الصورة بينما أعطت طريقة DB4 Wavelet Transformation نسب عالية ايضا لضغط الصور بدون الحفاظ على نوعية الصورة.