DWT Foveation-Based Multiresolution Compression Algorithm

J. C. Galan-Hernandez, V. Alarcon-Aquino, O. Starostenko, J. M. Ramirez-Cortes[1]

Department of Computing, Electronics, and Mechatronics
Universidad de las Americas Puebla
Sta. Catarina Martir, Cholula, Puebla. C.P. 72810. MEXICO
Email: {juan.galanhz, vicente.alarcon}@udlap.mx

[1]Department of Electronics
Instituto Nacional de Astrofisica, Optica y Electronica
Tonantzintla, Puebla MEXICO

Abstract – **Discrete Wavelet Transform (DWT) foveated compression can be used in real-time video processing frameworks for reducing the communication overhead. Such algorithms lead into high rate compression results due to the fact that the information loss is isolated outside a region of interest (ROI). The fovea compression can also be applied to other classic transforms such as the commonly used discrete cosine transform (DCT). An analysis has then been performed showing different error and compression rates for the DWT-based and the DCT-based foveated compression algorithms. Simulation results show that with foveated compression high ratio of compression can be achieved while keeping high quality over the designed ROI.**

Keywords — **Foveation, wavelets, wavelet transforms, discrete cosine transform, compression.**

I. Introduction

Digital images and video are essentially multi- dimensional signals and are thus quite data intensive [1]. Video and image compression can help in reducing the communication overhead between computing nodes and with wavelet compression, high ratios of compression can be achieved [2]. However, such algorithms also cause loss of information on video frames. In applications where a region of interest (ROI) can be isolated, foveation can be used in order to constraint the information loss only on those areas outside of the ROI. With such model, several applications can classify an area as a ROI such as medical video processing framework searching for melanomas, and perform a lossy compression over the video frames, leaving the ROI intact for later processing. Previous works in [2, 3, 4] show different foveation methods assessing the final quality of the image against the original image using Human Vision System criteria. Foveation also guarantee certain quality from the human system vision perspective granting the output results with enough quality for a user to inspect the ROI without noticing the loss of quality unless a close inspection outside the ROI. In the work reported in this paper, an analysis of DWT-based foveated compression algorithms is carried out. The remainder of this paper is organized as follows. In Section II an overview of foveated compression is given. Section III describes the proposed approach. Section IV presents simulation results, and Section V reports conclusions and future work.

II. Foveated Compression

Wavelet transforms involve representing a general function in terms of simple, fixed building blocks at different scales and positions. These building blocks are generated from a single fixed function called mother wavelet by translation and dilation operations. Thus, wavelet transforms are capable of “zooming-in” on short-lived high frequency phenomena, and “zooming-out” on long-lived low frequency phenomena [5].

A. Wavelets and the Discrete Wavelet Transform

The purpose of wavelet transforms is to represent a signal into the frequency-time domain. To perform this task two functions are required, namely, a wavelet and a scaling function. If a set of mother wavelets and scaling functions is orthonormal it is called an orthonormal bases and is defined as follows [6]

$$\{\varphi_{l_0,n}\}_{0\leq n\leq 2^{l_0}} \cup \{\psi_{j,n}\}_{j<l_0, 0<n<2^{-j}} \tag{1}$$

where each $\psi_{j,n}$ is a translated copy of ψ at scale j:

$$\psi_{j,n}(t) = \sqrt{2^{-j}}\psi(2^{-j}t - n) \tag{2}$$

and each $\varphi_{l_0,n}$ is a translated copy of the scaling function φ at scale l_0:

$$\varphi_{l_0,n}(t) = \sqrt{2^{-j}}\varphi(2^{-j}t - n) \tag{3}$$

In two dimensions three mother wavelets are used [6], $\Psi^d_{j,m,n}$, $\Psi^v_{j,m,n}$ and $\Psi^d_{j,m,n}(x,y)$ defined as follows

$$\Psi^d_{j,m,n}(x,y) = \psi_{j,m}(x)\psi_{j,n}(y) \tag{4}$$

$$\Psi^v_{j,m,n}(x,y) = \psi_{j,m}(x)\varphi_{j,n}(y) \tag{5}$$

$$\Psi^h_{j,m,n}(x,y) = \varphi_{j,m}(x)\psi_{j,n}(y) \tag{6}$$

with the scaling function:

$$\Phi_{j,m,n}(x,y) = \varphi_{j,m}(x)\varphi_{j,n}(y) \tag{7}$$

where $\Psi^d_{j,m,n}(x,y)$ are the diagonal coefficients, $\Psi^v_{j,m,n}(x,y)$ are the vertical coefficients and $\Psi^h_{j,m,n}(x,y)$ are the horizontal coefficients. With the wavelet base defined, the next step is using it to represent a signal. The sum over all time of the signal multiplied by scaled, shifted versions of the mother wavelet Ψ is given by:

$$a_j(n) = \int_{-\infty}^{\infty} f(t)\psi_{j,n}(t)dt \tag{8}$$

where f is the signal to be represented. However, for compression this transform is not suitable because it expands the signal into more coefficients than the samples of the signal itself. A better transform, suited for compression and many other applications, is called the discrete wavelet transform (DWT). In [7], the DWT is calculated through a simple algorithm that applies two filters, a low pass filter and a high pass filter. This algorithm is known as the fast wavelet transform [7]. In wavelet analysis of a signal f, we often speak of approximations and details. The approximations are the low-frequency components of the signal, see (9); whereas the details are the high-frequency components of the signal, see (10).

$$a_j[n] = \langle f, \varphi_{j,n}\rangle \tag{9}$$

$$d_j[n] = \langle f, \psi_{j,n}\rangle \tag{10}$$

B. Discrete Cosine Transform

The discrete cosine transform (DCT) expresses a signal in terms of cosine functions. Such transform is commonly used in the JPEG compression algorithm, the MP3 audio format and the VP8 video format [8]. The discrete cosine transform for a signal $f(x)$ of length N is defined as follows [9]

$$C(u) = \alpha(u)\sum_{x=0}^{N-1} f(x)cos\left[\frac{\pi(2x+1)u}{2N}\right] \tag{11}$$

where $\alpha(u)$ is defined by:

$$\alpha(n) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for} \quad u = 0 \\ \sqrt{\frac{2}{N}} & \text{for} \quad u \neq 0 \end{cases} \tag{12}$$

In particular, $u(0)$ is known as the direct current coefficient (DC) and the remaining coefficients are called the alternating current coefficients [10]. Most of the energy of the signal is packed in the DC coefficient.

C. Wavelet Compression

The objective of data compression is to represent a set of data with less information than the original. There are two types of compression: lossy and lossless compression [10]. In lossy, some of the original data is discarded in order to achieve its goal. When the data is reconstructed it will be slightly different from the original. Using lossy compression can help to achieve a better compression ratio than lossless compression if the losses are acceptable over the result. This is a usual practice on images, such as jpeg and jpeg2000 formats [7, 10], and video such as mpeg4 format. When a wavelet transform is applied on an image, the resultant coefficients can then be compressed more easily because the information is statistically concentrated in just a few coefficients. Wavelet compression can reach higher compression ratio than other transforms such as the discrete cosine transform suggested for foveated compression in [3]. In Wavelet lossy compression, the coefficients that contain the most amount of energy are preserved and the rest are discarded. Selecting such coefficients can be done using the wavelet energy profile and choosing a cutoff frequency.

D. Foveation

Foveated images are images which have a non-uniform resolution [6]. Results reported in [11] have demonstrated that the human eye experiments a form of aliasing from the fixation point to the edges of the image. Such aliasing increases in a logarithmic rate on all directions. This can be seen as concentric cutoff frequencies from the fixation point. When it is used in a wavelet, this can be expressed as a function [2]:

$$I_0(x) = \int I(t)C^{-1}(x)s\left(\frac{t-x}{w(x)}\right)dt \tag{13}$$

where $I(t)$ is a given image, $I_0(x)$ is the foveated image, $w(x)$ is the weight function. The function s is called the weighted translation of s by x. The function C is defined as:

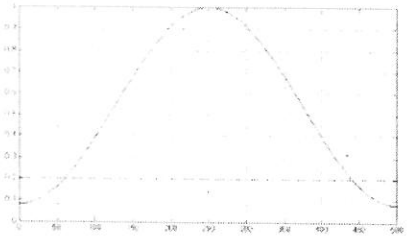
$$C(x) = \left\| s\left(\frac{-x}{w(x)}\right) \right\| \tag{14}$$

The results on a foveated image from [6] are shown in Figure 1. There are several weighted translation functions such as the ones defined in [6]. In [3], the suggested weighted functions are the Hamming window (Figure 2a) and the triangular window (Figure 2b). Such windows offer a smooth degradation from the fixation point. However, in order to preserve a ROI intact, such windows are not useful. A ROI needs to be left with all its coefficients without cutoff. For well defined ROIs, it is proposed to use a Tukey window (Figure 2c) Such windows can be used to define a weighted function where the radio of a fixation point is bigger than one, leaving the coefficients from the ROI untouched and right after the ROI ends the energy begins to decay in a smooth ratio.

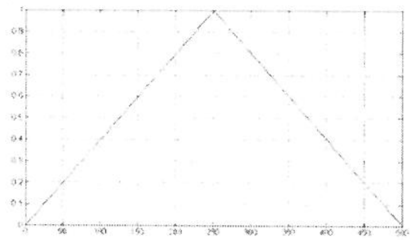
(a) Original gray level image

(b) Foveation point at the right eye

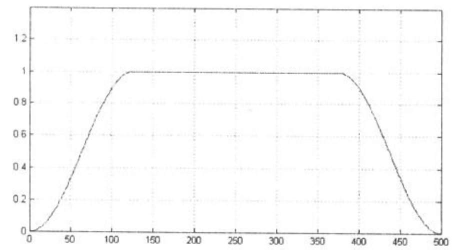
Fig. 1. An image and its wavelet foveated compression



(a) Hamming window



(b) Triangular window



(c) Tukey window

Fig. 2. Different foveating windows

III. Proposed DWT-Based Foveated Algorithm

An implementation of DWT foveated compression was made in order to compare the quality and compression rate of a DWT-based foveated compression using a Tukey window over different wavelets. For compression quantization three different schemes were used, namely, an energy frequency profile cutoff scheme, a Tukey window-based scaling and a JPEG Type 1 quantization based on the Tukey window [8]. Figure 3 shows a foveated image with the different schemes with the fovea point in the eye of the bird and a radius of 60 pixels and four levels of wavelet decomposition using a Daubechies 7 wavelet. Other wavelets may also be considered. For the energy frequency profile cutoff, the algorithm is defined as follows: Given a fixation point, the ROI centroid, and a radius, the algorithm computes the foveated image I of dimension M x N with J levels of decomposition showed in Algorithm 1. The two-dimensional Tukey window, $T2D$, is calculated through interpolation of the one-dimensional Tukey window and the function $x^2 + y^2$. Note that the interpolation method used was linear interpolation [13]. The set L is a decent ordered set of all the coefficients of the wavelet discrete transform. ε is the total energy of the coefficients. The set E is the wavelet energy profile. The set $Cutoff$ is the fovea filter and contains all the thresholds for each pixel of the image. $Cutoff_j$ is the fovea filter weighted for each level of decomposition. The filter is not applied to the approximation DWT coefficients a_j. This creates a coarser effect than in [3] or in [6]. However, with a well defined ROI, this method leads into a better compression ratio with a slightly less quality outside the ROI. For the Tukey window-based scaling, the cutoff is replaced for the next formula using a scaled $T2D_j$ window:

$$d'^{\alpha}_j[m,n] := d^{\alpha}_j[m,n] * T2D_j[m,n] \tag{15}$$

where $\alpha \in \{h, v, d\}$, $T2D_j$ is the $T2D$ window resized by j. In this scheme, the quantization is also applied to the approximation coefficients a_j.

$$a'_K[m,n] := a_K[m,n] * T2D_K[m,n] \tag{16}$$

(a) Original gray level image

(b) Energy Profile Quantization

(c) Tukey window scaling Quantization

(d) JPEG Type 1Quantization

Fig. 3. A Wavelet-based foveated image with different quantizations.

ALGORITHM 1 PROPOSED DWT-BASED FOVEATED ALGORITHM

---

for all $j \in J$ **do**

$d_j^d[m,n] = \langle I, \Psi_{j,n}^d \rangle$

$d_j^v[m,n] = \langle I, \Psi_{j,n}^v \rangle$

$d_j^h[m,n] = \langle I, \Psi_{j,n}^h \rangle$

$a_j[n] = \langle I, \Phi_{j,n} \rangle$

end for

$T2D := Interpolate(Tukey, x^2 + y^2)$

$L: = \{L_i \in d_j^v \cup d_j^h \cup d_j^d \cup a_j \,|\, 0 < i \leq MxN, L_{i-1} \geq Li\}.$

$\varepsilon := \sum_{\forall i} L_i^2$

$E := \{e_i | e_i = \frac{\sum_{j=1}^{i} L_j^2}{\varepsilon}, i = 1 \ldots M \times N\}$

for all x, y **do**

$Cutoff(x,y) := \begin{cases} L_i & \text{if } e_i \neq \emptyset \\ 0 & \text{if } e_i = \emptyset \end{cases}$

where $e_i = max\,(\{e_j | e_j \leq T2D(x,y)\})$

end for

for all x, y **do**

$Cutoff_j := resize(Cutoff, \frac{M}{j}, \frac{N}{j})$

end for

end for

for all x, y **do**

$d'^{\alpha}_j[m,n] := \begin{cases} d_j^{\alpha}[m,n] & \text{if } d_j^{\alpha}[m,n] \leq Cutoff(m,n) \\ 0 & \text{if } d_j^{\alpha}[m,n] > Cutoff(m,n) \end{cases}$

where $\alpha \in \{h, v, d\}$

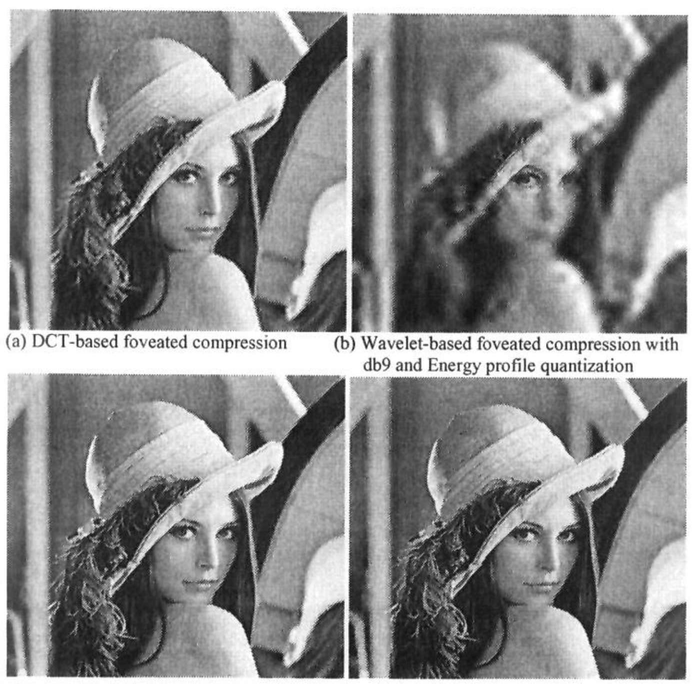
end for

Calculate inverse discrete wavelet transform with the new coefficients d_j^v, d_j^d, d_j^h and the approximation coefficients a_j

where $K = max(J)$. A gray scale image "lena" of dimensions 512x512 was used for assessing the performance of the proposed algorithm. The original gray scale image is shown in Figure 3a.

IV. Simulation Results

Comparisons were carried out with the wavelets Daubechies 7 (db7) and Daubechies 9 (db9) suggested by the JPEG2000 standard [12] with two arbitrary levels of decomposition, $J = 2$ and $J = 4$. To assess the performance of the wavelet-based foveated algorithm the mean squared error (MSE) metric is used. Also, an arbitrary fovea radius used was of 60 pixels. The results are shown in Table 1. The DCT-based foveated compression was also realized and it is shown in Table 1 as dct. The compression was realized using the Type 3 JPEG variable quantization compression [8] and the Tukey window as the quantization weight. Table 1 shows how many coefficients becomes zero after the quantification function is applied. Figure 4 shows the DCT-based foveated image and a wavelet-based foveated image using the Daubechies 9 wavelet with four levels of decomposition with energy profile quantization, Tukey window-based scaling quantization and JPEG Type-1 quantization. It should be noted in Table 1 that the amount of zeros in the coefficients after applying wavelet-based foveated algorithm using energy profile and JPEG Type-1 quantizations schemes is lower than the DCT-based approach. Furthermore, simulation results show that the DCT-based algorithm created artifacts that make the image fuzzier than the wavelet-based algorithm. The performance of the algorithm depends on the quantization method used. Energy quantization gives a good compression ratio at expenses of computational speed, it is expected a complexity of n^2 because of the wavelet coefficient sorting and the calculation of the cutoff window. Scaling quantization is faster with an expected complexity of n but has a low compression ratio.



(a) DCT-based foveated compression

(b) Wavelet-based foveated compression with db9 and Energy profile quantization

(c) Wavelet-based foveated compression with db9 and Tukey window quantization

(d) Wavelet-based foveated compression with db9 and JPEG Type 1 quantization

Fig. 4. Foveated image compression with different methods.

TABLE I

COMPARISONS OF DIFFERENT WAVELET-BASED FOVEATED COMPRESSION AND DCT-BASED FOVEATED COMPRESSION.

Wavelet	J	Zero Coef. Percentage	MSE	Quantization
dct	-	0.9383	2.2402	Type3
db7	2	0.3287	0.2534	Tukey Scaling
db7	4	0.3333	0.2503	Tukey Scaling
db9	2	0.3279	0.3279	Tukey Scaling
db9	4	0.3309	0.2428	Tukey Scaling
db7	2	0.8761	3.8959	JPEG Quantization
db7	4	0.9174	2.4795	JPEG Quantization
db9	2	0.875	3.8167	JPEG Quantization
db9	4	0.9161	2.3974	JPEG Quantization
db7	2	0.9163	2.4192	Energy Profile
db7	4	0.9644	6.0816	Energy Profile
db9	2	0.9153	2.3897	Energy Profile
db9	4	0.9624	6.0116	Energy Profile

V. CONCLUSION

Wavelet foveation compression offers a very good compression ratio at expenses of controlled losses. However, applying foveation with wavelets yields into squared artifacts as mentioned in [3]. These artifacts increase as the decomposition levels increases. However, the DCT also showed a similar behavior in the area outside the ROI as the compression rate increases. With a good model for choosing a ROI, this kind of compression can help in reducing the transmission overhead of video frames, increasing the ability of realizing real-time in video processing. It is intended for future work to compare the wavelet-based foveated compression algorithm with other methods such as the JPEG2000 ROI compression called maxshifts [12].

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support from the Mexican National Council for Science and Technology (CONACYT) and the Puebla State Government under the contract no. 109417.

REFERENCES

[1] N. Kehtarnavaz and M. Gamadia, “Real-Time Image and Video Processing: From Research to Reality”, Morgan and Claypool, University of Texas at Dallas, USA, 2006.

[2] E. C. Chang and C. K. Yap, “A wavelet approach to foveating images”, *In SCG ’97: Proceedings of the thirteenth annual symposium on Computational geometry*, pages 397–399, New York, NY, USA, 1997. ACM.

[3] S. Lee and A. Bovik, “Fast algorithms for foveated video processing”, *Circuits and Systems for Video Technology, IEEE Transactions on*, 13(2):149 – 162, 2003.

[4] Chenlei Guo; Liming Zhang, “A Novel Multiresolution Spatiotemporal Saliency Detection Model and Its Applications in Image and Video Compression”, *Image Processing, IEEE Transactions on* , vol.19, no.1, pp.185-198, Jan. 2010

[5] A. Graps, “An introduction to wavelets”, *IEEE Comput. Sci. Eng.*, 2(2):50–61, 1995.

[6] E.-C. Chang, S. Mallat, and C. Yap, “Wavelet Foveation”, in *Applied and Computational Harmonic Analysis*, vol. 9, issue 3, pp. 312-335, 2000.

[7] S. Mallat, “A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way”, Academic Press, 2008.

[8] Digital compression and coding of continuous-tone still images – Requirements and guidelines, *Recommendation T.81 (09/92) ITU-T*; 1992.

[9] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” *IEEE Transactions on Computers*, vol. C-32, pp. 90-93, Jan. 1974.

[10] A. C. Bovik, “The Essential Guide to Image Processing”, Academic Press, 2009.

[11] B. A. Wandell. *Foundations of Vision*. Sinauer Associates, Inc., 1995.

[12] JPEG 2000 image coding system: Core coding system, *Recommendation T.800 (08/02) ITU-T*, 2002

[13] Meijering, E.; ,"A chronology of interpolation: from ancient astronomy to modern signal and image processing," Proceedings of the IEEE , vol.90, no.3, pp.319-342, Mar 2002, doi: 10.1109/5.993400