A Survey on Transform based Compression

Dipsi Dave

U & P U. Patel Department of Computer Engineering, C.S.P.I.T., Changa, India

Abstract: On-board image compression systems aim to augment the amount of data that can be stored using on-board mass memory and reduces the bandwidth prerequisite for transmission to ground station. in general, Imaging instruments are major sources of data generation because of the increase in resolution and dynamic range which is handled by Onboard Computer (OBC). OBC has the limited: hardware, bandwidth, memory and communication time for transmission. To convene these constraints there is always a requirement of onboard compression. Lossy/Lossless data compression with low complexity algorithms is then required while compression ratio must extensively ascend. Various image compression algorithms have been used to increase their compression performance while complying with image quality necessities.

Keywords: Transform, Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Fast Fourier Transform (FFT), Karhunen-Loeve Transform (KLT), Quantization, Block Coding, EBCOT

INTRODUCTION

Space missions are designed to leave Earth ambiance and operate in the outer space. Sensed and reflected or emitted energy is processed, analyzed and applied that information in Remote Sensing. Store-and-forward mechanism is operated in satellite imaging payloads, whereby the captured images are stored on board and transmitted to ground later on. Increase of spatial resolution, an extensive amount of imaging data need to be handled. There exists a direct connection between Satellite Imagery and Image Processing techniques are utilized to augment the image for visual interpretation, and to reinstate the image. The compression of remote sensing imagery includes a diversity of system issues like the scene, sensor and resulting data characteristics. The best choice and impact of various compression approaches can be very reliant upon these system issues. One latent approach to reconstruct an optimal image using image processing techniques from observed high resolution images is called Image Compression. The philosophy behind image compression is to diminish the amount of redundant data/information and compress the high data/information contents of an image. Image compression algorithms are classified as lossy, or lossless. Lossy compression algorithms permit some degradation in the image, with aim of achieving a higher compression ratio. The critical facet of lossy algorithms is the trade-off between image quality and compression rate. Lossless compression algorithms achieve bit-identical reconstruction of the input image. High-tech lossless algorithms typically achieve around 50% compression ratio. Lossy compression is limited only by the acceptable image quality with compression ratios of between 5 and 10%.

TRANSFORMS

I. DISCRETE COSINE TRANSFORM (DCT) : The power of digital images are concerted in low frequency, so we can use DCT transform to part low and high frequencies and then preserve the low frequency constituent as far as possible and also subtract the high frequency constituent to diminish compression rate. The encoding part performs the DCT on macro-blocks to achieve decorrelation and frequency analysis because the transforms are performing on the 8x8 blocks. The forward 2D DCT formula is defined in equation (1). p[x,y] and D[i,j] represent the input signal and the DCT coefficients respectively.

$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j)$$

$$\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x,y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right] \qquad (1)$$

Where, $C(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u = 0\\ 1 & \text{if } u > 0 \end{cases}$

Fig 1 shows that DCT co-efficient values are the relative amount of the 64 spatial frequencies that presented in the original 64-point input. The "DC coefficients" are the rudiments in uppermost left equivalent to zero frequency in both directions and rest is called "AC coefficients."



Fig 1. 64 Two-dimensional spatial frequencies

The FDCT processing step lays the base for achieving data compression by focusing most signal in the lower spatial frequencies as the pixel values usually change vary slowly from point to point across an image. For a distinctive 8x8 sample block from a distinctive source image, most of the spatial frequencies need not to be encoded because they have zero or near-zero amplitude.

After quantization, more than half of the DCT coefficients are equal to zero. JPEG incorporates run-length encoding to take gain from this. For each non- zero DCT coefficient, JPEG records three parameters like number of zeros (that preceded the number), number of bits (needed to represent the amplitude of number), and amplitude itself.

Fig 2.shows that JPEG processes DCT coefficients in the zigzag pattern to merge the run of zeros.



Fig 2. Zigzag Scanning of DCTMatrix

Inverse DCT reverses this processing step at the decoder. It takes 64 DCT coefficients and reconstructs 64-point output image signal by summing the base signals. Mathematically, the DCT is one-to-one mapping for 64-point vectors between frequency domains and image. In standard, the DCT introduces no thrashing to the source image samples It simply transforms samples to a domain in which they can be more efficiently encoded.

II. DISCRETE WAVELET TRANSFORM (DWT): Before entropy coding the wavelet transform is applied to tiles. The transformed data usually exhibits lower entropy and is thus more "compressible" this is the benefit of employing the wavelet transform. In particular, wavelet transform separates a tile into four sub-bands; source modeling is customized to each sub- band. Thit is exactly what JPEG2000 does. In fact, since wavelet filters are designed to store different frequencies into each sub-band, sub-bands exhibit peculiar features which are "captured" by the JPEG2000 source modeler.^{[5][6]}

Fig 3 shows the effect of frequency filtering on a grey scale image. A variety of wavelet filters are allowed by the standard for both lossy and lossless compression. Lossy filters usually give better results but they involve floating point operations. The correct reconstruction of the input signal is not guaranteed using these filters due to floating point approximation. Filters involving only integer operations are allowed to overcome this problem.



Fig 3. Wavelet transform: Sub-band division

- 1. **Reversible Wavelets:** Uniform dead zone scalar quantization with a step size of Δb =1is used when reversible wavelets are utilized in JPEG 2000.
- e.g. CDF 5/3wavelet.
- 2. **Irreversible Wavelets:** The step size selection is restricted only by the signaling syntax itself when irreversible wavelets are utilized in JPEG 2000.

e.g. CDF 9/7 wavelets. The step size is specified in terms of an exponent $\epsilon b,$ $0{\leq}\epsilon b$ ${<}25,$ and a mantissa $\mu b\,$, $0{\leq}\mu b\,$ ${<}211$

In lossy compression, wavelet transform can be followed by a quantization step. Quantization reduces the bit depth of wavelet coefficients at the expense of precision i.e. coefficient ab(u,v) of sub band b is quantized to value qb(u,v).

$$q_b(u,v) = sign[a_u(u,v)] \cdot floor\left[\frac{|a_b(u,v)|}{\Delta b}\right]$$
(2)

Where quantization step size is $\Delta b = 2^{R_b - \varepsilon_b} \left(1 + \frac{\mu_b}{2^{11}} \right)$

III. FAST FOURIER TRANSFORM (FFT): The frequency components of a linear system are recognized and characterized by using Fourier transform to make a continuous waveform. To compute Discrete Fourier Transform (DFT), Fast Fourier transform (FFT) is an efficient method. Discrete Fourier Transform (DFT) is formed by analyzing the system on a digital computer i.e. sampling of waveform [8][9]. Two finite sequences of length N can be related to Discrete Fourier Transform. Given a sequence with components x[k] for k= 0,1...N-1 the Discrete Fourier Transform of this sequence is a sequence X[r] for r=0,1...N-1 defined by

$$X[r] = \frac{1}{N} \sum_{k=0}^{N-1} x[k] exp^{\left(-\frac{j2\pi rk}{N}\right)}$$
(3)

IV. KARHUNEN-LOEVE TRANSFORM (KLT): The Karhunen–Loeve (KL) transform is a well-known signal processing technique for data compression and filtering. A simple description of the KL transform is as follows. Given a (complex) random vector of dimension nx1, the KL transform of is represented by a square matrix T_{KL} of maximum possible rank that minimizes

$$J_{KL}(T) = E\{\|X - T_x\|^2\}$$
(4)

Where, E denotes expectation and |||| the frobenius norm. The matrix T_{KL} is known to be the projection matrix onto the rank-m principle sunspace of the covariance matrix = $E(XX^H)$. Where, the superscript H denotes conjugate transpose. More specifically, if the Eigen composition of R_8 is expressed as,

$$R_{\aleph} = \sum_{i=1}^{n} \lambda_i e_i e_i^H$$

q

Where, $\lambda_1 \ge \lambda_1 \ge \cdots \lambda_n$ then $T_{KL} = PP^H$ where $P = e_1e_1 \dots e_n$. The matrix T_{KL} implies two companions operations: compression i.e. $X_c = Q^{-1}P^H x$ and reconstruction i.e. $X_r = QPx$ where Q can be chosen to be any nonsingular matrix but Q=I is the most popular. The data compression ratio is given by m/n.

QUANTIZATION

Quantization is instrumental in enabling the rich feature set of JPEG 2000. Quantization is the element of lossy compression systems responsible for reducing the precision of data in order to make them more compressible. JPEG 2000 offers several different quantization options. Only uniform scalar (fixed-size) dead-zone quantization is included in Part I of the standard. Part II of the standard generalizes this quantization method to allow more flexible dead-zone selection. Furthermore, trellis coded quantization (TCQ) is offered in Part II as a value-added technology.

I. Scalar Quantization: The simplest form of quantization is scalar quantization. JPEG 2000 employs a dead-zone uniform scalar quantizer to coefficients resulting from the wavelet transform of image samples. A scalar quantizer (SQ) can be described as a function Q that maps each element in a subset of the real line to a particular value. For a given wavelet coefficient x, the quantizer produces a signed integer q given by

$$=Q(x) \tag{6}$$

where, q is quantization index.

$$q = Q(x)\sin(x)\left|\frac{|x|}{\Delta}\right|$$
(7)

The wavelet coefficients inside the interval $(-\Delta, \Delta)$ are quantized to zero for the quantizer. Thus, the interval $(-\Delta, \Delta)$ is called the "dead-zone". The width of this interval is 2Δ , while all other intervals are of width Δ .

BLOCK CODING

I. JPEG: Lossy/lossless data compression in image processing try to eliminate the spatial redundancies. Example of coding technique is:

Huffman coding: This method was first introduced by david A. Huffman in a 1952, this was attracted an overwhelming amount of research and has been adopted in many important and/or commercial application, such as fax machines, jpeg, and mpeg.

The encoding steps of the huffman coding described in bottom-up manner.

Step 1

- 1. Sort the gray levels by decreasing probability.
- 2. Add the two smallest probabilities.
- 3. Sort the new value into the list.
- 4. Repeat until only two probabilities remain.

Step 2

- 1. Give the code 0 to the highest probability, and the code 1 to the lowest probability in the present node.
- 2. Go backwards through the tree and add 0 to the highest and 1 to the lowest probability in each node until all gray levels have a unique code

II. JPEG2000: The entropy coding and generation of compressed bit stream in JPEG2000 is divided into two coding steps: Tier-1 and Tier-2 coding.

- Tier-1 (EBCOT coder and Binary Arithmetic Coding-MQ-Coder.
- Tier-2 (Organization of the bit-stream).

Divided into Blocks: After quantization, each sub-band is divided into rectangular blocks, called code-blocks (see Fig); these code-blocks are encoded independently. The code-block is decomposed into *P* bit-planes and they are encoded from the most significant bit-plane to the least significant bit- plane sequentially (Fig). Each bit-plane is f first encoded by a *fractional bit-plane coding* (BPC) mechanism to generate intermediate data in the form of a *context* and a binary *decision* value for each bit position. In JPEG2000 the *embedded block coding with optimized truncation* (EBCOT) algorithm has been adopted for theBPC.



Fig 4. Partitioning into Code Blocks

Tier-1 Coding: In the chain coding JPEG 2000 entropy coding of information is carried by the EBCOT algorithm. This algorithm has been created by David Taubman, The version proposed for JPEG 2000 has very interesting characteristics, especially adapted to the new compression standard.



Fig 5. Representation in bit plane

The basic principle of EBCOT is: when coding, EBCOT block receives a set of quantization coefficients within a code block. Later it is driven bit-plane by bit-plane, starting with the MSB by three coding passes.

Each bit of the code-block is supported by one of the three passes; it sends data to MQ pair to encode the bit. This pair consists of the context determined by the neighbors of the bit, and the value of the symbol to encode. This process is illustrated in Fig 6.



Fig 6. Flow between MQ and EBCOT

EBCOT Coder: EBCOT encodes each bit-plane in three coding passes. The three coding passes in the order in which they are performed on each bit-plane are significant propagation pass, magnitude refinement pass, and cleanup pass.



Fig 7. Scan order in Code Block

All three types of coding passes scan the samples of a code block in the same fixed order shown in Fig 7. The code block is partitioned into horizontal stripes, each having a nominal height of four samples. As shown in the diagram, the stripes are scanned from top to bottom. Within a stripe, columns are scanned from left to right. Within a column, samples are scanned from top to bottom.

Each coefficient bit in the bit plane is coded in only one of the three coding passes and for each coefficient in a block is assigned a binary state variable called its significance state that is initialized to zero (insignificant) at the start of the encoding. The significance state changes from zero to o ne (significant) when the first nonzero magnitude bit is found. The context vector for a given coefficient is the binary vector consisting of the significance states of its eight immediate neighbor coefficients for each pass, contexts are created which are provided to the arithmetic coder.

In the following each coding pass is described:

- i. Significance propagation pass: During the significance propagation pass, a bit is coded if its location is not significant, but at least one of its eightconnect neighbors is significant.
- ii. Magnitude refinement pass:

During this pass, all bits that became significant in a previous biplane are coded. The magnitude refinement pass includes the bits from coefficients that are already significant.

iii. Clean-up pass:

The clean-up pass is the final pass in which all bits not encoded during the previous passes are encoded (i.e., coefficients that are insignificant and had the context value of zero during the significance propagation pass). The very first pass in a new code block is always a clean-up pass.

Binary Arithmetic Coding-MQ-Coder

As explained previously, the fractional bit- plane coding (EBCOT) produces a sequence of symbols, pairs of context and decision (CX, D), in each coding pass. The context-based adaptive binary arithmetic MQ-coder that is used in JBIG2 is adapted in JPEG2000 standard to encode these symbols.



Fig 8. Input, Output: Arithmetic Encoder

Comparision of transforms:

Transform				
	DCT	DWT	FFT	KLT
Parameter				
Name	Discrete Cosine	Discrete Wavelet	Fast Fourier	Karhunen-
	Transform	Transform	Transform	Loève Transform
		O(N)		
Complexity	$O(2N \log_2 N-N+2)$		O(Nlog ₂ N)	O(Nlog ₂ N)
		(In certain cases)		
Main task	fewer cosine	captures both	Manages to reduce	Minimizes MSE
	functions are needed	frequency and location	complexity of	
	to approximate a	information (location in	computing	
	typical signal	time)		
Application	JPEG image	signal coding	Fast algorithms for	Signal detection in
	compression,		discrete cosine or	white noise
		preconditioning for data	sine transforms	
	MJPEG, MPEG, DV,	compression (Signal detection in
	Daala,		Fast Chebyshev	colored noise
	and Theora video	signal processing of	approximation	
	compression	accelerations for gait		Prewhitening
		analysis,	Fast	
			Discrete Hartley	
		JPEG2000	Transform	
			Solving Difference	
			Equations	

CONCLUSION

JPEG2000 image compression standard uses DWT for transformation of the data has very less complexity as compared to other transformation techniques.JPEG2000 algorithm for compression includes DWT, EBCOT (Embedded Block Coding with Optimized Truncation). EBCOT includes Arithmetic Coding and Mq Coder to generate compressed output.It decreases the original file size by given decomposition levels which depends original file size.

REFERENCES

- [1] Xavier Delaunay- "EBCOT coding passes explained on a detailed example"
- [2] ManjunathGangadhar, Dinesh Bhatiya- 2005- "FPGA based EBCOT architecture for JPEG 2000"
- [3] CSPIT- 12 October, 2001- "An overview of Quantisation in JPEG2000"
- [4] Majid Rabbani, Rajan Joshi- 2002- "An overview of JPEG2000 still image compression standard"
- [5] Chun-Liu, Liu- 23 February, 2010- "A tutorial on Wavelet transform"
- [6] Qiu-yuan CAI, Geng-sheng WANG, Yun-xin YU- 2010- "Research of Still Image Compression Based on Daubechies 9/7 Wavelet Transform"