Survey of Image Compression Algorithms in Wireless Sensor Networks

Li Wern Chew
University of Nottingham
43500 Selangor, Malaysia.
eyx6clw@nottingham.edu.my

Li-Minn Ang
University of Nottingham
43500 Selangor, Malaysia.
kezklma@nottingham.edu.my

Kah Phooi Seng
University of Nottingham
43500 Selangor, Malaysia.
kezkps@nottingham.edu.my

Abstract

The implementation of image processing engines in visual sensor nodes has been a major concern in the development of wireless multimedia sensor networks in a hardware constrained environment. In this paper, a review on eight popular image compression algorithms is presented. After conducting a comprehensive evaluation, it is found that Set-Partitioning in Hierarchical Trees (SPIHT) wavelet-based image compression is the most suitable hardware implemented image compression algorithm in wireless sensor networks due to its high compression efficiency and its simplicity in coding procedures.

1. Introduction

Wireless sensor network (WSN) is a network that consists of many sensing devices that communicate over a wireless channel with the capability of performing data processing and computation at the sensor nodes. WSNs are being developed for a wide range of applications such as environmental monitoring, habitat studies, object tracking, video surveillance, satellite imaging as well as in military applications [1]-[4].

Recently, there has been a growing interest in the research and development for a reliable and efficient wireless multimedia sensor network (WMSN). Multimedia data such as images and video frames collected from the camera nodes require extensive processing and this makes the implementation of a WMSN difficult especially in a hardware constrained environment. The challenges and constraints that influence the development of an efficient and flexible WMSN include high power consumption, limited bandwidth and memory limitation [1] [2].

Power consumption is a fundamental concern in the development of many WMSNs. Most sensing devices are self-powered by batteries. However, the need for long-term data collection and extensive processing consumes a large amount of energy. Therefore, a proper design is needed to maximize the lifetime of the wireless sensing networks [1] [2].

Also, multimedia contents especially high resolution images require extensive bandwidth for transmission [1]. Due to the limited bandwidth available, an image captured by the sensor nodes needs to be processed and compressed before it is transmitted. With image compression, a more efficient method of transmission can be obtained by removing the redundant information from the raw data.

Recent technologies have made possible the production of micro sensing devices with embedded processing capabilities. Due to space restrictions and the high cost of providing large amounts of memory storage, on-chip memory available is therefore limited and has become another major constraint in the processing of large images. Hence a simpler and more cost efficient system will need to be developed in order to meet the high memory storage demands in image processing.

For data processing in WMSNs, it is desirable to maintain a high compression ratio while at the same time, providing a reliable compression performance. This is what makes image processing so challenging and why image compression has continued to be one of the most popular research topics today.

In this paper, eight reliable and efficient coding techniques (four each from both the first and second generation image coding) are evaluated to determine which technique is the most suitable for implementation in a WSN.

2. Image compression algorithms

In view of the fact that neighbouring pixels in an image are highly correlated, this redundant information can be discarded by finding a less correlated representation of the image. This is the basic idea

978-1-4244-2328-6/08/$25.00 © 2008 IEEE

behind the image compression theory. Figure 1 shows the basic components of an image coding process which is performed in two stages, namely the image transformation stage followed by entropy coding stage. Image coding can be categorized under first generation and second generation image coding.

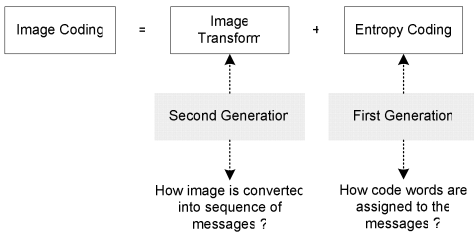

Figure 1. Process of image coding.

First generation image coding emphasize more on how well the information contained in a transformed image is efficiently encoded whereas the second generation places more importance on how we can exploit and extract useful information from the image. The second generation makes use of available techniques developed in the entropy coding stage to encode the sequence of information obtained from the image transform stage [10].

In this section, a literature review on both the first and second generation image compression algorithms will be presented. Four of the most popular transform-based image compression algorithms – Joint Photographic Experts Group (JPEG), Embedded Zerotree Wavelet (EZW), Set-Partitioning in Hierarchical Trees (SPIHT) and Embedded Block Coding with Optimized Truncation (EBCOT) algorithms are described under the first generation image coding. Pyramidal coding, directional decomposition based coding, segmentation based coding and vector quantization are then described under the second generation image coding.

2.1. First generation image coding

2.1.1. Discrete cosine transform based image compression. A well-known image compression standard JPEG uses the discrete cosine transform (DCT) based image compression technique. In DCT-based image compression, the image source is first partitioned into blocks of sub-images with a typical size of 8 x 8 pixels and each image block is coded independently [29].

DCT causes no loss to the original image data as it merely transforms the image into a domain in which they can be encoded more efficiently [29]. After the discrete cosine transformation, each of the 64 DCT coefficients is uniformly quantized. Zig-zag scanning is then applied to rearrange the coefficients prior to entropy coding. The low frequency non-zero components are placed at the beginning of the bit stream followed by the high frequency components. Figure 2 shows the process of zig-zag scanning. The DC component which contains important information about the image is coded differently. Finally, the output stream after the zig-zag scanning is entropy encoded.

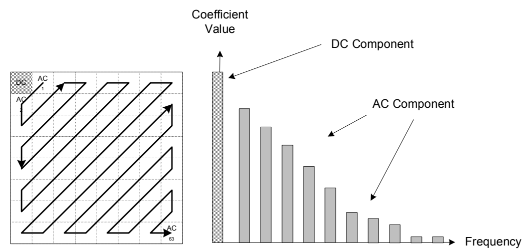

Figure 2. Zig-zag scanning.

DCT-based image compression provides satisfactory compression efficiency and it gives a low memory implementation since the encoding is done on small individual image blocks [17]. However, the tiling of the blocks causes blocking artifacts which lead to a degradation in performance especially at very low bit rates [17] [29].

2.1.2. Embedded zerotree wavelet-based image coding. Embedded coding started to gain attention in the field of image coding with the introduction of the popular wavelet-based image coding technique, EZW [6].

EZW coding involves the coding of the position of those wavelet coefficients that will be transmitted as a non-zero value. A wavelet coefficient, x is said to be insignificant with respect to a given threshold T if $|x| < T$. However, if $|x| \geqslant T$, then the coefficient is said to be significant with respect to T. The algorithm is developed based on the following hypothesis quoted from [6] - "if a wavelet coefficient at a higher scale is insignificant with respect to a given threshold T, then all the wavelet coefficients of the same orientation in the same spatial location at the lower scales are likely to be insignificant with respect to T."

In EZW coding, a tree is referred to as the zerotree if the parent node and all its descendants are insignificant with respect to the threshold. A coefficient of a zerotree for a threshold T is coded as zerotree root (ZTR) if it is not the descendant of a previously found ZTR for that threshold T. However, if the coefficient is insignificant with respect to threshold T but has some significant descendants, it is coded as isolated zero (IZ). For the coefficient that is found to be significant with respect to T, it is coded as positive significant (POS) or negative significant (NEG) depending on the sign of the coefficient.

Two lists need to be maintained during the EZW coding: a dominant list which contains the coordinate of the coefficients that have not yet been found to be significant and a subordinate list which contains the magnitude of those coefficients that have already been found to be significant [6]. For each threshold, all coefficients in the dominant list are scanned and coded. Those coefficients that are coded as significant are then transferred to the subordinate list. In the subordinate pass, each of the coefficients in the list is refined to an additional bit of precision. This encoding process is repeated until all wavelet coefficients are coded or the target rate has been met. The encoded symbols stream that contains a mixture of four symbols (POS, NEG, ZTR and IZ) and the refinement bit '1' or '0' is then arithmetically coded for transmission.

The inherent ability of embedded coding to generate the encoded bit stream in the order of importance allows the encoder to stop encoding at any point when a target rate or target distortion metric is met. Similarly, at any given bit stream, the decoder can cease decoding at any point and yet provides a full image reconstruction. This allows signal-to-noise ratio (SNR) scalability as the reconstructed image quality is dependent on the number of bits that have been decoded. It also allows progressive transmission as the image is decompressed with increasing accuracy [6].

2.1.3. Set-partitioning in hierarchical trees image coding. SPIHT [7] coding adopts a set-partitioning approach where the significance test result is binary. A coefficient is encoded as significant i.e. Sn(i, j) = 1 if its value is larger than or equal to the threshold T, or as insignificant i.e. Sn(i, j) = 0 if its value is smaller than T. Also, there are two types of descendant trees described in the SPIHT algorithm: type A or D(i, j) set which contains all the descendants of node (i, j) and type B or L(i, j) set which contains all the grand descendants of node (i, j).

There are two coding passes in SPIHT algorithm, the sorting pass and the refinement pass. During the sorting pass, a significance test is performed on the coefficients based on the order in which they are stored in the List of Insignificant Pixels (LIP). Elements in LIP that are found to be significant with respect to the threshold are moved to the List of Significant Pixels (LSP) list. A significance test is then performed for the sets in the List of Insignificant Sets (LIS). Here, if a set in LIS is found to be significant, the set is removed from the list and is partitioned into four single elements and a new subset. This new subset is added back to LIS and the four elements are then tested and moved to LSP or LIP depending on whether they are significant or insignificant with respect to the threshold.

Refinement is then carried out on every coefficient that is added to the LSP except for those that are just added during the sorting pass. Each of the coefficients in the list is refined to an additional bit of precision. Finally, the threshold is halved and SPIHT coding is repeated until all the wavelet coefficients are coded or until the target rate is met.

The set-partitioning approach that SPIHT adopts is an efficient and compact technique that improves the performance of EZW even without arithmetic coding [7].

2.1.4. Embedded block coding with optimized truncation. The popular image compression standard JPEG2000 adopts the EBCOT image compression technique [8]. As its name implies, the embedded coding is carried out on a block-based basis. Figure 3 shows a block diagram of JPEG2000. In EBCOT, the information coding and information ordering are separated and is referred to as two tiers coding [9].

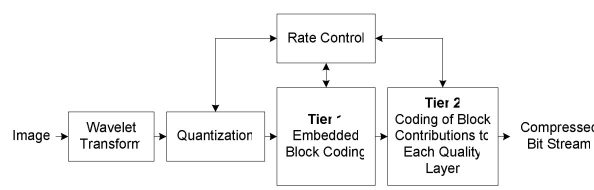

Figure 3. Block diagram of JPEG2000.

In Tier 1, the wavelet-transformed image is partitioned into blocks of samples known as code-blocks as shown in Figure 4 [8]. Each of these code-blocks is encoded independently with a context-based bit-plane adaptive arithmetic coding [9]. An embedded bit-stream is then generated for every code-block. However, since the subbands are partitioned and are encoded independently, parent-children dependencies across the subbands could not be exploited [8].

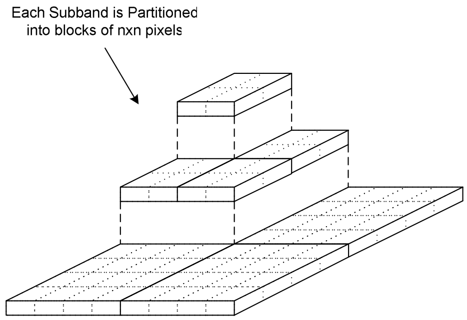

Figure 4. Partitioning of wavelet subbands into small code-blocks.

In Tier 2, the independently generated bit-streams from Tier 1 are subdivided into small ‘chunks’. These chunks are then interleaved and packed into different quality layers depending on their contributions to the layers. This process is referred to as ‘packetization’ [8] [9] and is shown in Figure 5. At the end of the Tier 2 process, a compressed bit-stream is obtained and ready for transmission.

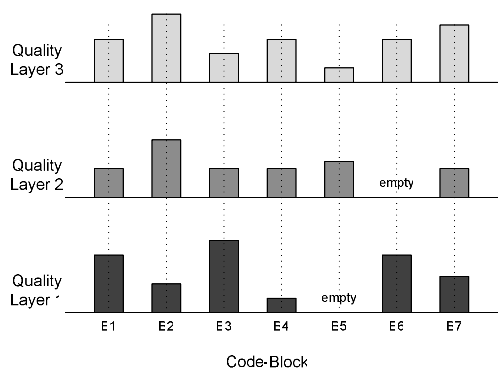

Figure 5. Embedded code-block bit-streams at different quality layers.

The length of each encoded bit-stream from each code-block can be varied and truncated depending on the rate-distortion ratio. In addition, the quantization step size can also be adjusted [9]. All these adjustments can be made using the rate-control mechanism. Besides producing bit-streams that have SNR scalability, EBCOT also allows resolution scalability as well as random access features.

2.2. Second generation image coding

2.2.1. Pyramidal / multiresolution coding. Although pyramidal coding had already been introduced during the very early stage in the development of image coding, it is considered here as a second generation image coding since the hierarchical coding methodology is similar to the nervous system in the Human Visual System (HVS) [10]. Figure 6 shows the process of pyramidal coding. The Laplacian pyramid algorithm breaks up an image into components based on spatial frequency [12]. The value at each node in the pyramid represents the difference between two Gaussian-like functions convolved with the original image.

Pyramidal coding provides a multiresolution representation of an image. A low resolution image that contains low frequency components is first transmitted. This low resolution image contains most of the energy of the image and it can be encoded with relatively fewer bits [5]. High frequency components that are encoded at a higher resolution are then progressively transmitted. However, on certain images, these high frequency features such as points, edges and lines might require more bits to encode than the actual full resolution image [5]. In this situation, pyramid coding will become less efficient as an encoding algorithm.

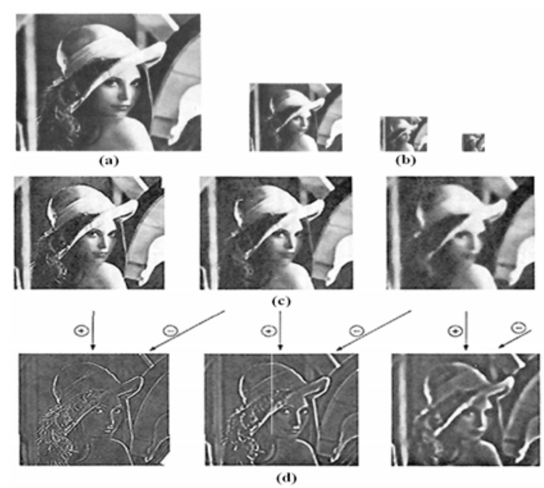

Figure 6. Process of pyramidal coding [12]. (a) Original image “Lenna”. (b) Gaussian pyramid image. (c) Gaussian interpolation image. (d) Laplacian pyramid image.

2.2.2. Directional decomposition based coding. From the study and analysis on the nature of HVS, edge information is found to be vitally important in the perception of image [13]. However, this information is often distorted when the image is encoded with conventional coding schemes such as transform coding, subband coding and wavelet coding [14].

Directional filtering coding [15] emphasize more on edge detection to achieve high compression ratios. It

978-1-4244-2328-6/08/$25.00 © 2008 IEEE

works based on the fact that the human eye is composed of directional-sensitive neurons [14] [15]. A directional filter is used to exploit the relationship between the edges and its contribution to the image spectrum [10] [11] [13]. This filter is defined in [10] as "a filter that performs high pass filtering along the principal direction and low pass filtering along the orthogonal direction".

During directional filtering, the original image is decomposed into a low pass image and a number of high pass images. The edge information in the image is well preserved as each of the high pass images contains the edge information in one principal direction [13]. Since the low pass image contains no edges, transform coding can be applied whereas the high pass images are used for edge detection and coding.

2.2.3. Segmentation based coding. By utilizing the fact that the human eye is good at identifying regions that are similar and grouping them accordingly [11], this coding works by partitioning the image into sub-regions based on their texture structure. These regions are surrounded by contours and both the contour and texture regions are coded separately.

Figure 7 shows the process of segmentation based coding. Preprocessing is first carried out on the image to remove noise as well as the non-useful regions based on the characteristics and analysis of HVS [11]. A region growing based coding known as the contour texture modeling was proposed by [16]. During the segmentation process, each pixel and its neighboring pixels are examined to determine if they share the same properties based on their grey scale levels. Pixels with the same properties are assigned to the same region. This region growing process is repeated until all the pixels in the image are assigned to some region. Weakly contrasted adjacent regions and small regions are then merged to reduce the number of regions obtained [10]. Finally, contour coding and texture coding are then applied on the contour and texture regions respectively.

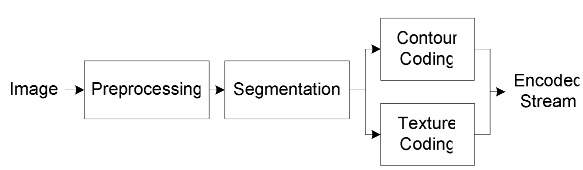

Figure 7. Process of segmentation based coding.

One of the major difficulties faced in the implementation of this segmentation based coding is the setting of thresholds which is used to determine if two pixels are having the same properties. The value of this threshold cannot be fixed even for the coding of an image [16].

2.2.4. Vector quantization. Here, highly correlated image pixels are first grouped into blocks of sample set. Each of these blocks will have a best approximation vector that can represent every pixel in the given partitioned area [28]. Quantization is carried out on these blocks of pixels and each block is then independently encoded. Because of this, vector quantization is also known as block quantization or pattern matching quantization [27].

The complexity of a vector quantization coder largely depends on the encoding/decoding algorithms used. Some examples of vector quantization algorithms include the tree-structures vector quantization, product vector quantization, address vector quantization, multistage vector quantization and pyramid vector quantization [25] [26]. All these algorithms function based on a similar concept which is described below.

Figure 8 shows the block diagram of vector quantization coding. After preprocessing, an image is partitioned into blocks of pixels where each of them has a representation vector, K. This vector is compared against a predefined set of vectors (codewords) in a lookup table (codebook) and the codeword with the best match is chosen. However, the index of the codeword is transmitted instead of the codeword itself in order to achieve a higher compression ratio since the index can be represented with a fewer number of bits [26]. At the decoding side, the codeword is picked from the codebook based on the index value received and the image is then reconstructed. Since a large number of vectors are encoded into a finite set of codewords available in the codebook, vector quantization provides a lossy compression [25] [26].

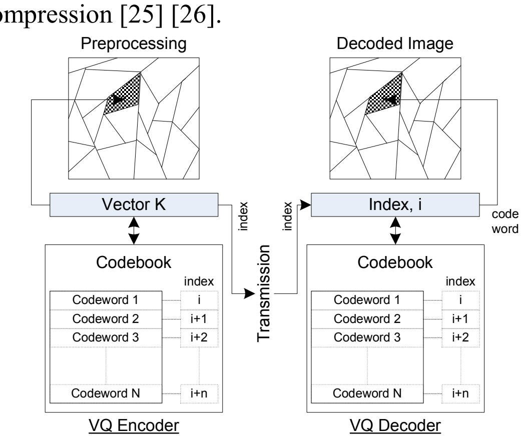

Figure 8. Process of vector quantization (VQ).

978-1-4244-2328-6/08/$25.00 © 2008 IEEE

3. Observation and analysis

Table I summarizes the properties and characteristics of the eight image compression algorithms presented in this paper. The purpose of this analysis is to find the most suitable image compression algorithm for implementation in a wireless sensor network.

The criteria of selection is that the chosen algorithm should possess most of the preferred characteristics of a WSN operating in a hardware constrained environment, including a fast and efficient image processing capability, low memory requirement, high compression quality, less complex system and low computational load.

Based on the analysis of the information summarized in Table I and extensive literature studies conducted on the eight image compression algorithms, the following observations are noted:

In contrast to the first generation image compression algorithms that remove the redundancy in an image by exploiting the similarity between image pixels, the second generation image compression algorithms that incorporate the properties of HVS identify the features within the image and process these features to achieve compression [11]. This property of second generation image coding that emphasizes on exploring the "content" of an image and makes use of this information to achieve compression requires more complex and extensive image processing compared to the first generation image coding which performs wavelet transformation.

Also, most of the second generation image compression algorithms provide lossy compression and they rely heavily on initial segmentation [10] [11]. During the segmentation process, image pixels are first classified into contour and texture regions followed by a region growing process. Here, the whole image is expected to be stored and available in memory during preprocessing and this is difficult to achieve especially when available memory is limited. Furthermore, the segmentation process requires extensive computation and this not only increases the complexity of the coder but also decreases the processing speed which makes its implementation in a real-time environment not feasible.

In comparison, first generation image coding performs DCT or DWT to decompose an image into a domain that is more suitable for compression. This is done without incurring any cost in terms of excess redundancy since the transformed image size remains the same as its original size. Since DCT causes blocking artifacts which reduces the quality of the reconstructed image, the wavelet-transformed coding is preferred over the DCT-based image coding. In addition, wavelet transformation can be carried out without the need for a full image transformation [17]. This enables a very low memory implementation of the image coder.

SPIHT image coding generates bit streams according to the order of importance based on a set-partitioning algorithm. Besides having all the embedded properties that EZW coding possesses, it also gives a better performance than EZW [7]. Although EBCOT coding provides a higher compression efficiency as compared to SPIHT, its multi-layered coding procedures are very complex and computationally intensive. Also, the need for multiple coding tables for arithmetic coding requires extra memory allocation which makes the hardware implementation of the coder more complex and expensive [18]-[20].

Various modifications have been proposed by many researchers on the traditional SPIHT image compression algorithm for implementation in a hardware constrained environment. Listless coders [21] [22] have been introduced to solve the problem of extensive memory requirement needed for the maintenance of lists required in SPIHT coding. The strip-based SPIHT image compression [23] and packetized SPIHT image compression [24] are two such examples of SPIHT image coders with very low memory requirement.

From the above comprehensive analysis, SPIHT clearly has many advantages over the other coding techniques for implementation in a WSN.

978-1-4244-2328-6/08/$25.00 © 2008 IEEE

Table I. Summary of analysis on first generation and second generation image compression algorithms.

Characteristic	First Generation Image Compression Algorithm				Second Generation Image Compression Algorithm			
	JPEG	EZW	SPIHT	EBCOT	Pyramidal	Directional Decomposition	Segmentation	Vector Quantization
Preprocessing (Transformation)	DCT	DWT	DWT	DWT	Laplacian Pyramid	Directional filtering	Region growing	Region clustering
Coding Table / Codebook	no	no	no	yes	Depends on entropy coding used		yes	yes
Post processing (Entropy Coding)	Arithmetic Coding	Arithmetic Coding	Not needed	Arithmetic Coding	Entropy Coding	Entropy Coding	Entropy Coding	Entropy Coding
Memory Requirement	Low	Average	Average	High	Average	Average	High	High
Computation Load	Low	Low	Low	High	Average	Average	Extensive	Extensive
System Complexity	Low	Average	Average	High	Low	Average	High	High
Coding Speed	High	High	High	Average	High	Average	Low	Low
Compression Quality	Low	Average	High	High	Low	Average	High	High
Most suitable for WSN			✓					

978-1-4244-2328-6/08/$25.00 © 2008 IEEE

4. Conclusion

Based on the results of the evaluation conducted on these eight popular image compression algorithms from both the first generation and second generation image coding presented in this paper, it is found that SPIHT wavelet-based image compression is the most suitable image compression algorithm for implementation in a wireless sensor network in a hardware constrained environment.

5. References

[1] I.F. Akyildiz, T. Melodia, K. Chowdhury, "Wireless multimedia sensor networks: A Survey", *Elsevier - Computer Networks*, Mar. 2007.

[2] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, J. Anderson, "Wireless sensor networks for habitat monitoring", *WSNA'02*, Atlanta, Georgia, USA, Sept. 2002.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramanian, E. Cayirci, "A survey on sensor networks", *IEEE Communications Magazine*, Aug. 2002.

[4] E. Magli, M. Mancin, L. Merello, "Low-complexity video compression for wireless sensor networks", *Proceedings in International Conference on Multimedia and Expo.*, vol. 3, no. 9, pp. 585-588, Jul. 2003.

[5] S. J. Golin, "Method and apparatus for encoding a digital motion video signal using pyramid specific filtering," *US Patent, No. 813884.* Jul. 1994.

[6] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3445-3462, Dec. 1993.

[7] A. Said and W. A. Pearlman, "A new fast/efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 12, pp. 243-250, Jun. 1996.

[8] D. Taubman, "High performance scalable image compression with EBCOT", *IEEE Trans. Image Process.*, vol. 9, no.7, pp. 1158-1170, Jul. 2000.

[9] H. Zhang and J. Fritts, "EBCOT coprocessing architecture for JPEG2000", *Proceedings of the SPIE*, vol. 5308, pp. 1333-1340, Jan. 2004.

[10] M. Kunt, A. Ikonomopoulos, M. Kocher, "Second-generation image-coding techniques", *Proceedings of the IEEE*, vol. 73, no. 4, pp. 549- 574, Apr. 1985.

[11] M. M. Reid, R. J. Millar, N. D. Black, "Second-generation image coding: An overview", *ACM Computing Surveys*, vol. 29, no. 1, Mar. 1997.

[12] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code", *IEEE Transactions on Communications*, vol. 31, no. 4, Apr. 1983.

[13] C. N. Zhang and X. Wu, "A hybrid approach of wavelet packet and directional decomposition for image compression" *IEEE Canadian Conference on Electrical and Computer Engineering*, Edmonton, Alta., Canada, vol. 2, pp. 755-760, Dec. 1999.

[14] Z. Zhou and A. N. Venetsanopoulos, "Morphological methods in image coding", *IEEE International Conference on Acoustics, Speech, and Signal Process.* (ICASSP-92), San Francisco, CA, USA, vol. 3, pp. 481-484, Mar. 1992.

[15] A. Ikonomopoulos and M. Kunt, 'High compression image coding via directional filtering", *Signal Process*. no. 8, pp. 179-203, 1985.

[16] M. Kocher and M. Kunt, "Image data compression by contour texture modeling", *Proc. Soc. Photo-Optical Instrumentation Eng. (SPIE) 397*, pp. 132-139, 1983.

[17] C. Chrysafis and A. Ortega, "Line-based, reduced memory, wavelet image compression," *IEEE Trans. Image Process.,* vol. 9, no. 3, pp. 378-389, Mar. 2000.

[18] W. B. Huang, W. Y. Su, Y. H. Kuo, "VLSI implementation of a modified efficient SPIHT encoder ", *IEEE Trans. Fundamentals.*, vol.E89-A, no.12, pp. 3613-3622, Dec. 2006.

[19] J. Jyotheswar and S. Mahapatra, "Efficient FPGA implementation of DWT and modified SPIHT for lossless image compression", *Elsevier – Journal of Systems Architecture 53*, pp. 369-378, 2007.

[20] Z. Y. Cao, Z. Z. Ji and M. Z. Hu, "An image sensor node for wireless sensor networks", International Conference on Information Technology: Coding and Computing (ITCC 2005), vol. 2, no. 4-6, pp. 740-745, Apr. 2005.

[21] F. W. Wheeler and W. A. Pearlman, "SPIHT image compression without lists", *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing,* vol. 4, pp. 2047-2050, Jun. 2000.

[22] W.-K. Lin and N. Burgess, "Listless zerotree coding for color image", in *32nd Asilomar Conference on Signals, Systems and Computers,* Monterey, CA, Nov. 1998.

[23] R. K. Bhattar, K. R. Ramakrishnan, and K. S. Dasgupta, "Strip based coding for large images using wavelets," *Elsevier - Signal Processing: Image Communication 17*, pp. 441-456, 2002.

[24] F. W. Wheeler and W. A. Pearlman, "Low-memory packetized SPIHT image compression", *33rd Asilomar*

Conference on Signals, System and Computers, Monterey, CA, pp. 1193-1197,1999.

[25] A. Vasuki and P. T. Vanathi, "A review of vector quantization techniques", *IEEE Potentials*, vol. 25, no. 4, pp. 38 47, July-Aug. 2006.

[26] N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: a review", *IEEE Transactions on Communications*, vol. 36, no. 8, pp. 957-971, Aug. 1988.

[27] N. M. Nasrabadi and Y. Feng, "A multilayer address vector quantization technique", *IEEE Transactions on Circuits and Systems*, vol. 37, no. 7, pp. 912-921, Jul. 1990.

[28] G. F. McLean, "Vector quantization for texture classification", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 637-649, May/Jun. 1993.

[29] D. Cruz, T. Ebrahimi, J. Askelof, M. Larsson, C. Christopoulos, "Coding of Still Picture", *45th Proc. of SPIE Applications of Digital Image Processing*, vol. 4115, Jul. 2000.

978-1-4244-2328-6/08/$25.00 © 2008 IEEE