ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

Novel Technique for Lossless Compression of Video Using RCRC

Jayshree R. Pansare [1], Ketki R. Jadhav [2]

Assistant Professor, Department of Computer Engineering, MES College of Engineering, Pune, India[1]

Department of Computer Engineering, MES College of Engineering, Pune, India[2]

ABSTRACT: We present the innovative lossless video compression scheme using Row-Column (RCRC) coding algorithm. Proposed method consists of two main components: Codebook model and RCRC algorithm. Codebook model is combination of two elements, 8×8 block and their representative huffman codes and consist of probability occurrences of those blocks. RCRC is a skip line encoding based technique applied on rows and columns of 8×8 block is an alternative to codebook model compresses those block which are missed in codebook model. Experimentation results shows that proposed reduces required memory upto 95%. And gives better CR about 0.18 bpp.

KEYWORDS*:* Lossless compression, Codebook model, RCRC, Huffman coding, K-means clustering etc.

I. INTRODUCTION

Because of rapid growth in audio, video applications, video compression techniques become an important area for the researchers. Video Compression techniques are either software or hardware implementations. While offering the purpose in several applications such as Video tax, Videoconferencing, videophone, Cable TV distribution, Digital storage media, Network video information provider, Database services video library, Video-on-demand, Video surveillance, Broadcasting, video compression take advantage of temporal, statistical and spatial redundancies to achieve maximize the CR of videos. Even incredible research happens in this area, it is still gains attention of users because of computational requirement and quality at low bit rates.

Proposed method gives details about the innovative LSVC (Lossless System for Video Compression) using codebook and row-Column reduction coding. Lossless compression techniques are useful in giving good quality. Even though lossy compression techniques gives better CR, in accuracy-demanding applications such as IoT (Internet of Things), Hyperspectral imagining etc. loss of information is an undesirable. Codebook model consist of 8*8 blocks and their correlated huffman codes for compression of 8×8 blocks. RCRC is a newly proposed iterative algorithm based on reducing similar rows and columns respectively is an alternative to codebook model.

Entropy encoder, huffman coding developed by David Huffman in 1952 is a main concept has been used for development of codebook. It can uses another entropy encoder, arithmetic coding within codebook implementation instead of huffman coding. For generation of codebook, frequency analysis on no of randomly chosen binary images has been done and they possibly diverse as possible as can. Reduces the noise affected binary images before check against the proposed method. So it can be helpful in further processing. K-means is clustering algorithm used for color separation of images [2].

The rest of paper is organized as follows: Section II includes literature survey includes concepts used by our method and reviews methods which are lossless uses different techniques. Section III presents system architecture shows flow of system. Under the subsections it explains the purpose each step took to compress the video, and how it is useful? Section IV explains no of algorithms used by our system consist of K-means algorithm, Huffman algorithm and RCRC algorithm. Section V gives performance analysis obtained when our method applied on no of videos and shows how our method useful

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

in reducing required memory. And finally, section VI concludes the proposed system and contribution with system and future enhancement related with topic.

II. LITERATURE SURVEY

A. *Block Coding*

Proposed method based on concept of dividing binary image or bi-level image into 8×8 blocks having non-overlapping structure. Concept of block coding has been described and abstracted in [1]. Blocks either have value 0 or 1-bit. Encoding of 0-valued pixels consist of one single bit 0 followed by rows of blocks. 1-valued pixels are encoded similar to 0-valued pixels but uses 1 instead 0 as one single bit.

B. *Lossless Video Compression*

Combination of dictionary-based algorithm and huffman coding [10] gives high compression ratio. Proposed scheme uses an adaptive quotient bit truncation within dictionary-based algorithm to create codewords for all video pixels and allocate bit values to codewords by using huffman coding. After that it uses an algorithm that again reduces size of codeword bits. This method gives better compression ratio than traditional systems.

Reduction of temporal redundancy and spatial redundancy using movement estimation and then GAP respectively with adaptive method is an ideal way to reduce redundancy in video formats. It is lossless method gives better compression than JPEG-LS and CALIC method [11]. Backward adaptive temporal prediction and an integer wavelet transform gives better compression performance by reducing temporal and spatial redundancies when applied on separate video frames [12]. Lossless video compression using new context tree on existing MPEG gives compression which is higher than JPEG-LS by 24% and CALIC by 22% achieves expected processing speed, low computational complexity and high compression performance [13]. Prefix sequence matching, traversing the context tree and context thresholding to find the insights show at the inward hub closest to the ceasing point, settled request of setting structure with a most extreme number of four movement repaid mistakes and to quantize the higher end of blunder esteems into a solitary measurements cluster respectively.

III. SYSTEM ARCHITECTURE

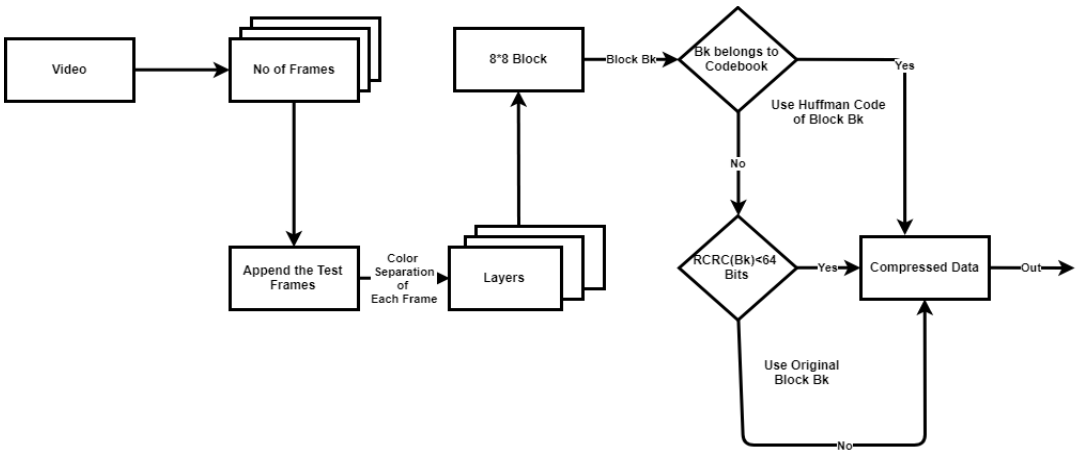


Fig.1: Architecture of Proposed System

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

As video is sequence of frames proposed method extracts no of frames from video as shown in Fig. 1 and applies proposed method on those frames. Proposed technique has gone through no of stages as follows:

A. Preprocessing stage

Preprocessing performs trimming of margins and then frame expansions for each extracted frame so they can be divisible by 8. Preserving such casings would increment the relative likelihood of 8×8 blocks comprising of zeros or ones if the foundation is white or dark, individually. If we consider v and g as a height and width of the frame. Conversion of these dimensions to new one v* and g* has been done such that:

- If v mod8_=, then v+8-vmod8:
- If g mod8_=, then g+8-gmod8.

Thus, the new dimensions with new height and width for frame v*×g*. Color layer separation has been done using K-means algorithm on each appended frame. Separated layers are a bi-level image consist of only two colors uses bit '1' to represent color and bit '0' for background representation. Color separation inspires for development of 8×8 matrix. So it can be check against codebook and RCRC module.

B. Huffman Coding

Codebook model is a combination of 8×8 block and their correlated huffman codes, so that's why 1st we have to understand that what is a huffman codes, use of it, and characteristics of it. Huffman coding is a concept of entropy encoding used for probability reduction for frequently occurring symbols. On the off chance that the entropy of a specific message is N, at that point the most noteworthy compression ratio that can be accomplished for that message is (M- N)/N, where N is the size of the message. Less entropy results in better CR for input data. Huffman Coding preserves the concept of variable-length code table. Variable–length code table have two entries, 1st is codewords and another one is frequency occurrences of each symbol. It compresses the symbols by small codewords, who have large frequency of occurrence and large codewords for less frequently occurring symbols.

C. Codebook Generation Process

Like we said codebook model is a combination of huffman codes with their frequency of occurrence and related blocks is fixed-to-variable sized codebook. Fixed part consists of 8×8 block and variable part consists of correlated huffman codes. 120 randomly chosen binary images are used for development of codebook model by performing frequency analysis on those image blocks. Size of huffman codes varies from 1 bit to 17 bit. Reason behind to choosing 8×8 blocks is factual balancing between huffman codes, entropy and block dimensions. Choosing block dimensions which has dimension size less than 8×8 blocks like 2×2 and 4×4 resulted in less entropy reduction and doesn't give reliable factual probability distribution experimentally, while even blocks with larger dimensions has been chosen alphabet size imposes the factual limit in selecting block, even if it gives better entropy. Also choosing block size larger than 8×8 blocks affect the performance row-column reduction coding algorithm because of increase in vector size.

D. Row-Column Reduction Coding

Row-Column Reduction Coding (RCRC) algorithm is newly proposed iterative algorithm reduces similar rows and columns from 8×8 block using concept of Row-Reference Vector (RRV) and Column-Reference Vector (CRV). It assigns each row and column of 8×8 block the RRV and CRV starts with RRV0 to RRV7 and CRV0 to CRV7 respectively. It is an alternative to codebook model, when for the input block there is no huffman code in codebook model. Advantage of RCRC, it is easy to understand and simple. Process of RCRC has been showed in following figures.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

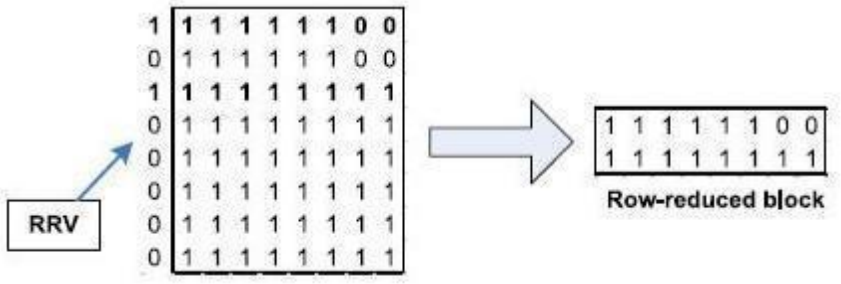
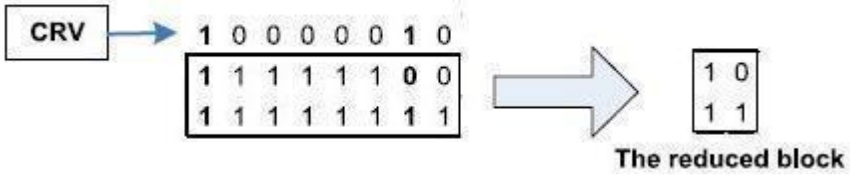


Fig.2: The row-reduction of block

As seen in Fig 2. If we see that RCRC reduces the rows with stream '11111100' and '11111111' in single stream. For doing so, it assigns '0' to child row and '1' to parent row.



After that it applies column reference reduction on this row reduced block, gives results as shown Fig 3. Process of column reduction is similar to RRV reduction by assigning'0' to child column and '1' to parent column.

Fig.3: The column-reduction of block after row-reduction of block.

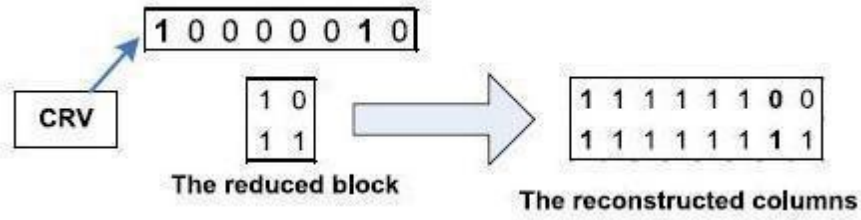


Fig.4: Reconstruction of column using column-reference vector (CRV).

Reconstruction of 8×8 block is opposite to the entire process of RCRC compression of 8×8 block shown in Fig 4 and Fig 5.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

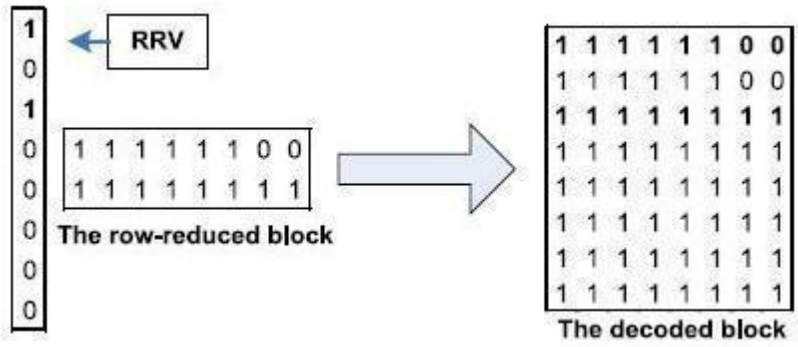


Fig.5: Reconstruction of row using row-reference vector (RRV)

E. Coding Conditions

Coding of 8×8 block using codebook model or RCRC algorithm come up with certain cases as explained below. Process is easy to understand and explain.

Case 1: If the input 8×8 block is found in codebook, compression of block is done using 2 cases as explained below.

Case 1a: If corresponding huffman code of 8×8 block is a small sized codeword , in our case 1 bit in codebook, method compresses that block using '11' in combination with that 8×8 block.

Case 1b: If length of huffman code is greater than 1 bit, method compresses that block using '00' with additional 5 binary bits equals to length of correlated huffman code. This 5 additional bits tells the decoder to read the next correlated bits of huffman code.

Case 2: If input 8×8 block not found in codebook, method compresses it by using RCRC algorithm. It uses '01' in combination with output of RCRC algorithm.

Case 3: If input 8×8 block is never found in codebook or get compressed using RCRC, process transfers that block as it is. This might affects the compression performance of our method. Process uses the two bits '01' with that block.

IV. ALGORITHMIC APPROACH

Proposed method uses 3 algorithms K-means clustering, huffman coding and Row-Column Reduction Coding. Working each algorithm is explained as below.

A. K-means Algorithm

The consequences of the K-means clustering calculation are:

- The centroids of the K groups, which can be utilized to mark new information
- Labels for the preparation information (every datum point is appointed to a solitary group)

As opposed to characterizing bunches before taking a gander at the information, grouping enables you to discover and dissect the gatherings that have framed naturally. The "Picking K" area underneath portrays how the quantity of gatherings can be resolved. Every centroid of a bunch is a gathering of highlight esteems which characterize the subsequent

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

gatherings. Inspecting the centroid include weights can be utilized to subjectively translate what sort of gathering each group speaks to.

Consider x1 are input image to K-means algorithm and we want to cluster it n no of subimages according to color of pixels, if q(a, b) are the input pixels and C= (c1, c2, c3...) are centers of cluster. K-means process those pixels as follows:

- Initiates the clusters and centers.
- For each pixel find out nearest centered pixel in image.

$$d = ||q(a; b)|| Ck$$

- Assign each pixel to nearest centroid.
- After assigning pixels, recalculate new position for centers using following formula.

$$c = 1/v\sum b \epsilon C\sum a\epsilon C\ q(a; b)$$

- Repeat the process until all similar pixels are obtained.
- According to pixels reshape no of subimages.

Fig. 6 shows flowchart of K-means for image segmentation in different clusters.

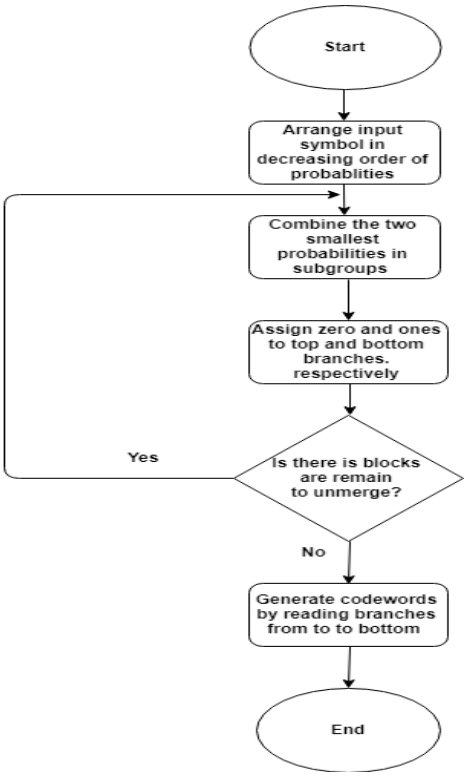


Fig.6: Flow Chart of K-means algorithm

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

B. Huffman Coding

As we know that huffman coding is process of frequency reduction, procedure of redundancy reduction in huffman algorithm goes through following stages:

- Start.
- Read the symbols and arranges them according to their frequency of occurrences in descending order.
- Merges probabilities of two smallest symbols gives highest probability symbols.
- Repeats the process until all symbols get reduced.
- End.

Fig. 7 shows huffman coding flow chart, processing used for obtaining codewords for 8×8 block.

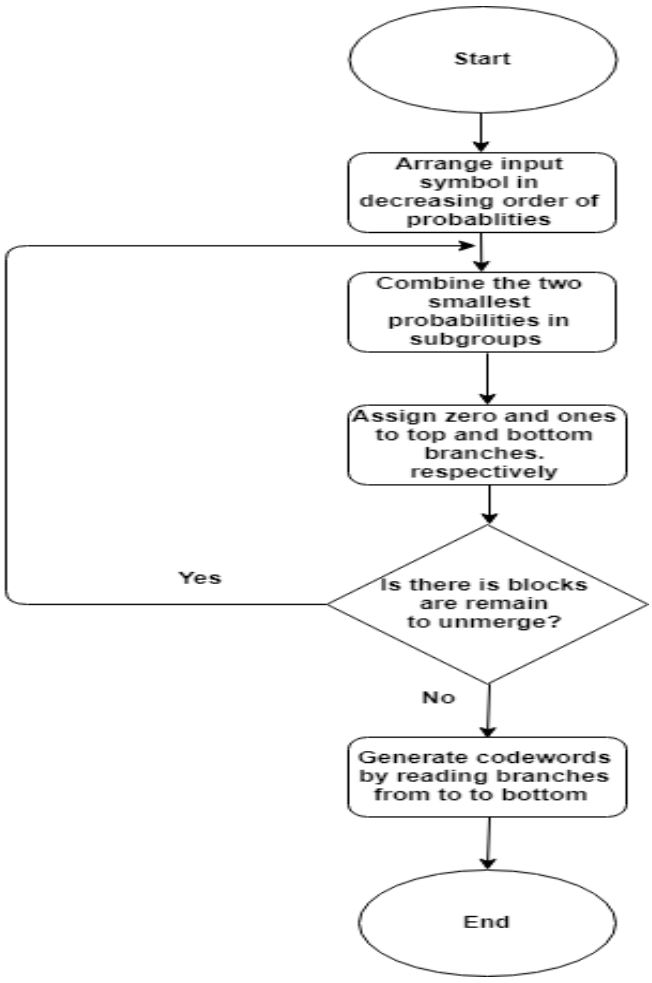


Fig.7: Flow chart of Huffman algorithm

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

C. Row-Column Reduction Coding

RCRC reduces those 8×8 block whose corresponding huffman coding not found in codebook. Consider RRVi and CRVi assigned vectors to rows and columns of 8×8 block. Where i=0, 1…7 for both rows and columns. Then process of RCRC goes through following stages.

- Takes the input as an 8×8 block.
- First performs the row reduction.
- Reads the rows of 8×8 block.
- If RRVi=RRVj, it reduces the row RRVj by keeping RRVi. Where j=1…..7 rows in block.
- Repeats the process until all the similar rows are get reduced.
- Then it reduces the columns by similar process used for Row reduction.
- Output is a reduced block for input 8×8 block.
- End

V. PERFORMANCE ANALYSIS

Experimentation has been performed on four videos, shows that size of original video and compressed video. It compares original size of video with compressed size. Experimental results shows that how easily and by using simple techniques like K-means, codebook model and RCRC algorithm proposed method compresses the video at expectable level. System performs no of operations on no of frames of videos. As video is no frames, system uses base of image compression method to compress the video frames. Table I represents results of our method on 4 different videos of different size and types.

Table I: Compression Results

Sr.	Name	Size of original video	Compressed Video	Compression ratio in bpp
1	Wildlife.wmv	25 MB	1 MB	0.04
2	VID-20170704-WA0003.mp4	4.12 MB	2.56 MB	0.65
3	seagulls_on_the_beach.mp4	23.2 MB	416 KB	0.017
4	Raindrops_Videvo.mp4	56.4 MB	904 KB	0.016
Avg.		106.17MB	4.76MB	0.18

We can see that the large difference between average size of original videos and average size of reduced videos. On an overage proposed method reduces size of videos upto 95% gives better storage space saving. Proposed method gives better CR than other methods.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

VI. CONCLUSION

Proposed method LSVC is lossless video compression technique. To achieve better entropy reduction and quality of compressed video, proposed method is lossless compression scheme uses codebook model or RCRC algorithm to compress frames of videos. This method is simple, efficient and cost effective method. It gives better compression ratio than existing video compression techniques in bpp. Reduces required memory on an average upto 95%. Method uses K-means algorithm and huffman coding which are most popular and efficient techniques preferred by most of researchers. We can improve performance of system by performing frequency analysis on more binary images for implementation of codebook than used in our proposed method.

REFERENCES

1. M. Kunt and O. Johnsen, "Block coding of graphics: A tutorial review," Proc. IEEE, vol. 68, no. 7, pp. 770-786, Jul. 1980.
2. A. M. Rufai, G. Anbarjafari, H. Demirel, "Lossy medical image compression using Huffman coding and singular value decomposition," IEEE Conference on Signal Process. and comm. app., Sep. 2013.
3. H. c. Kuo, Y. L. Lin, "A simple and effective lossless compression algorithm for video display frames." IEEE Conference on Mul. And Expo (ICME), Sep. 2011.
4. Z. Ming-Feng, Hu Jia, Z. Li-Ming, "Lossless video compression using combination of temporal and spatial prediction," IEEE Conference on Neural Net. and Sig. Pro., April 2004.
5. Sanjay Kumar Gupta, "An Algorithm For Image Compression Using Huffman Coding Technique," (IJARSE) International Journal of Advance Research in Science and Engineering, Vol. 5, No. 7, pp. 69-75, July 2016.
6. S. Zahir and A. Borici, "A fast lossless compression scheme for digital map images using color separation," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process. , pp. 1318-1321, Mar. 2010.
7. Z. Ming-Feng, Hu Jia, Z. Li-Ming, "Lossless video compression using combination of temporal and spatial prediction," IEEE Conference on Neural Net. And Sig. Pro., April 2004.
8. A. Bookstein and S. T. Klein, "Is Huffman coding dead?" Computing, vol. 50, no. 4, pp. 279–296, 1993.
9. Paul G. Howard and Jeffrey Scott Vitter, "Analysis Of Arithmetic Coding For Data Compression," Information Processing & Management, Vol. 28, No. 6. pp. 749-763, 1992.
10. H. c. Kuo, Y. L. Lin, "A simple and effective lossless compression algorithm for video display frames," IEEE Conference on Mul. And Expo(ICME), Sep. 2011.
11. Z. Ming-Feng, Hu Jia, Z. Li-Ming, "Lossless video compression using combination of temporal and spatial prediction," IEEE Conference on Neural Net. and Sig. Pro., April 2004.
12. S. G. Park, E. J. Delp, H. Yu, "Adaptive lossless video compression using an integer wavelet transform," IEEE Conference on Image Processing, April 2005.
13. J. Jiang, G. Xiao, "Adding lossless video compression to MPEGs," vol. 8, no. 3, pp. 622-625, June 2006.