A Novel Implementation of Wavelet Transform and Lzma for Compression and Decompression of Document Images

M.SUDHA
Assistant Professor, Sudharsan Engineering College, Sathiyamangalam, Pudukkottai

Dr.S.PALANI
Director, Sudharsan Engineering College, Sathiyamangalam, Pudukkottai

Abstract - We compress the document image by the use of LZMA algorithm. It is stands for Lempel-Ziv-Markov chain algorithm. It is used for lossless compression. In pre-processing we remove the noise from the image. Then we apply the wavelet transform to the image. It will decompose the image. Here we quantize the image by the use of wavelet. It is more robust under transmission. For encoding process it will use the adaptive binary range coder. Finally we get the encoded value from the algorithm. After compression we apply the Decompression algorithm. The output of decompression will give to the Inverse wavelet transform. Finally it will produce the reconstructed image. We proposed the LZMA algorithm as a best performing one. Here we achieved compression ratios were improved between 5% to 15% in comparing with the plain best performing lossless compression algorithm (LZMA).

Keywords: Document image, LZMA, Compression ratio, wavelet transform, Pre- processing.

I. INTRODUCTION

Here we compress the document image by the use of LZMA algorithm. It is stands for Lempel-Ziv-Markov chain algorithm. It is used for lossless compression. In pre-processing we remove the noise from the image. Then we apply the wavelet transform to the image. It will decompose the image. Here we quantize the image by the use of wavelet. It is more robust under transmission. Then we apply the LZMA algorithm. The total image pixels are considered as a stream bits. Then the bits are separated as a number of packets. Then the packets bits are encoded. For encoding process it will use the adaptive binary range coder. Finally we get the encoded value from the algorithm. After compression we apply the Decompression algorithm. It is the reverse process of compression. The output of decompression will give to the Inverse wavelet transform. Finally it will produce the reconstructed image. we proposed the LZMA algorithm as a best performing one. Here we achieved compression ratios were improved with predictive modeling in place - typically we observed CR improvement between 5% to 15% in comparing with the plain best performing lossless compression algorithm (LZMA).

II. PRE-PROCESSING

In Filtering we are applying Gaussian filtering to our input image. Gaussian filtering is often used to remove the noise from the image. Here we used imfilter function to our input image. Gaussian filter is windowed filter of linear class, by its nature is weighted mean. Named after famous scientist Carl Gauss because weights in the filter calculated according to Gaussian distribution.

1D Gaussian filter, or Gaussian blur algorithm:

Given window size $2N+1$ calculate support points $x_n=3n/N$, $n=-N, -N+1, \ldots, N$;

1. Calculate values G''_n;
2. Calculate scale factor $k'=\sum G''_n$;
3. Calculate window weights $G'_n=G''_n/k'$;
4. For every signal element:
    1. Place window over it;
    2. Pick up elements;
    3. Multiply elements by corresponding window weights;
    4. Sum up products — this sum is new filtered value.

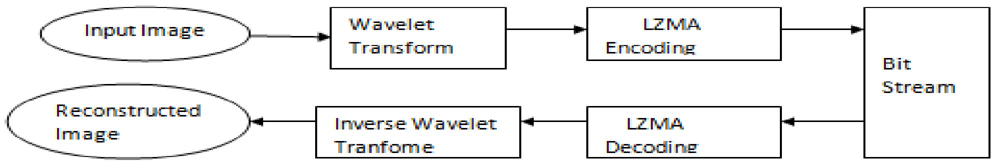

Figure 1: System Architecture

III. WAVELET TRANSFORM

Wavelet Transform has become an important method for image compression.Wavelet based coding provides substantial improvement in picture quality at high compression ratios mainly due to better energy compaction property of wavelet transforms.

Wavelet transform partitions a signal into a set of functions called wavelets.Wavelets are obtained from a single prototype wavelet called mother wavelet by dilations and shifting.The wavelet transform is computed separately for different segments of the time-domain signal at different frequencies.

A. Subband coding:

A signal is passed through a series of filters to calculate DWT.Procedure starts by passing this signal sequence through a half band digital low pass filter with impulse response h(n).Filtering of a signal is numerically equal to convolution of the tile signal with impulse response of the filter.

$$x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k].h[n-k] \quad \textbf{(1)}$$

A half band low pass filter removes all frequencies that are above half of the highest frequency in the tile signal.Then the signal is passed through high pass filter.The two filters are related to each other as

$$h[L-1-n] = (-1)^n g(n) \quad \textbf{(2)}$$

Filters satisfying this condition are known as quadrature mirror filters.After filtering half of the samples can be eliminated since the signal now has the highest frequency as half of the original frequency.The signal can therefore be subsampled by 2,simply by discarding every other sample.This consitutes 1 level of decomposition and can mathmatically be expressed as

$$Y_1[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n-k]$$

$$\textbf{(3)}$$

$$Y_2[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n+1-k]$$

where $Y_1[n]$ and $Y_2[n]$ are the outputs of low pass and high pass filters,respectively after subsampling by 2.

This decomposition halves the time resolution since only half the number of sample now characterizes the whole signal.Frequency resolution has doubled because each output has half the frequency band of the input.This process is called as sub band coding.It can be repeated further to increase the frequency resolution as shown by the filter bank

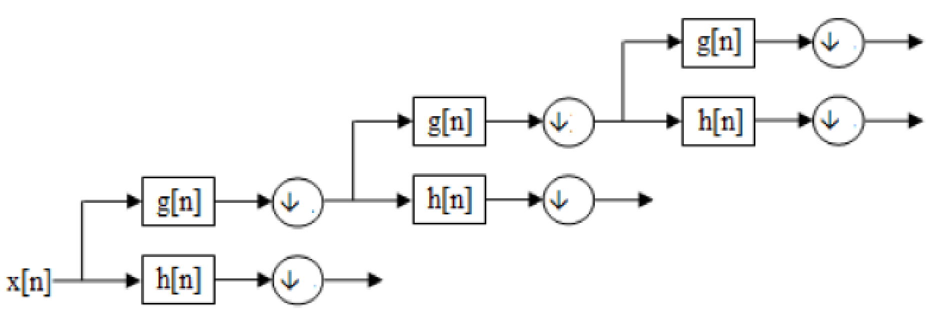

Figure 2: Filter Bank

B. Compression steps:

1.Digitize the source image into a signal s, which is a string of numbers.
2.Decompose the signal into a sequence of wavelet coefficients w.
3.Use threshold to modify the wavelet coefficients from w to w'.
4.Use quantization to convert w' to a sequence q.
5.Entropy encoding is applied to convert q into a sequence e.

C. Digitation:

The image is digitized first.The digitized image can be characterized by its intensity levels,or scales of gray which range from 0(black) to 255(white), and its resolution, or how many pixels per square inch.

D. *Thresholding*:

In certain signals, many of the wavelet coefficients are close or equal to zero.Through threshold these coefficients are modified so that the sequence of wavelet coefficients contains long strings of zeros.

In hard threshold ,a threshold is selected. Any wavelet whose absolute value falls below the tolerance is set to zero with the goal to introduce many zeros without losing a great amount of detail.

E. *Quantization:*

Quantization converts a sequence of floating numbers w' to a sequence of integers q.The simplest form is to round to the nearest integer.Another method is to multiply each number in w' by a constant k, and then round to the nearest integer.Quantization is called lossy because it introduces error into the process,since the conversion of w' to q is not one to one function.

F. *Entropy encoding:*

With this method, a integer sequence q is changed into a shorter sequence e,with the numbers in e being 8 bit integersThe conversion is made by an entropy encoding table.Strings of zeros are coded by numbers 1 through 100,105 and 106,while the non-zero integer

IV. LZMA ENCODING

LZMA uses a dictionary compression algorithm (a variant of LZ77 with huge dictionary sizes and special support for repeatedly used match distances), whose output is then encoded with a range encoder, using a complex model to make a probability prediction of each bit. The dictionary compressor finds matches using sophisticated dictionary data structures, and produces a stream of literal symbols and phrase references, which is encoded one bit at a time by the range encoder: many encodings are possible, and a dynamic programming algorithm is used to select an optimal one under certain approximations.

Prior to LZMA, most encoder models were purely byte-based (i.e. they coded each bit using only a cascade of contexts to represent the dependencies on previous bits from the same byte). The main innovation of LZMA is that instead of a generic byte-based model, LZMA's model uses contexts specific to the bitfields in each representation of a literal or phrase: this is nearly as simple as a generic byte-based model, but gives much better compression because it avoids mixing unrelated bits together in the same context. Furthermore, compared to classic dictionary compression (such as the one used in zip and gzip formats), the dictionary sizes can be and usually are much larger, taking advantage of the large amount of memory available on modern systems.In LZMA compression, the compressed stream is a stream of bits, encoded using an adaptive binary range coder. The stream is divided into packets, each packet describing either a single byte, or an LZ77 sequence with its length and distance implicitly or explicitly encoded. Each part of each packet is modelled with independent contexts, so the probability predictions for each bit are correlated with the values of that bit (and related bits from the same field) in previous packets of the same type.

Distances are logically 32-bit and distance 0 points to the most recently added byte in the dictionary.The distance encoding starts with a 6-bit "distance slot", which determines whether how many further bits are needed. Distances are decoded as a binary concatenation of, from most to least significant, two bits depending on the distance slot, some bits encoded with fixed 0.5 probability, and some context encoded bits, according to the following table (distance slots 0-3 directly encode distances 0-3).

V. LZMA DECODING:

LZMA data is at the lowest level decoded one-bit at a time by the range decoder, at the direction of the LZMA decoder.Context-based range decoding is invoked by the LZMA algorithm passing it a reference to the "context", which consists of the unsigned 11-bit variable *prob* (typically implemented using a 16-bit data type) representing the predicted probability of the bit being 1, which is read and updated by the range decoder (and should be initialized to 2^10, representing 0.5 probability). Fixed probability range decoding instead assumes a 0.5 probability, but operates slightly differently than context-based range decoding. The range decoder state consists of two unsigned 32-bit variables, *range* (representing the range size), and *code* (representing the encoded point within the range). Initialization of the range decoder consists of setting *range* to 2^32 - 1, and *code* to the 32-bit value starting at the second byte in the stream interpreted as big-endian; the first byte in the stream is completely ignored.

Normalization proceeds in this way:

1. Shift both *range* and *code* left by 8 bits
2. Read a byte from the compressed stream
3. Set the least significant 8 bits of *code* to the byte value read

Context-based range decoding of a bit using the *prob* probability variable proceeds in this way:

1. If *range* is less than 2^24, perform normalization
2. Set *bound* to floor(*range* / 2^11) * *prob*
3. If *code* is less than *bound*:
    1. Set *range* to *bound*
    2. Set *prob* to *prob* + floor((2^11 - *prob*) / 2^5)
    3. Return bit 0
4. Otherwise (if *code* is greater or equal than *bound*):
    1. Set *range* to *range* - *bound*
    2. Set *code* to *code* - *bound*
    3. Set *prob* to *prob* - floor(*prob* / 2^5)
    4. Return bit 1

Fixed-probability range decoding of a bit proceeds in this way:

1. If *range* is less than 2^24, perform normalization
2. Set *range* to floor(*range* / 2)
3. If *code* is less than *range*:
    1. Return bit 0
4. Otherwise (if *code* is greater or equal than *range*):
    1. Set *code* to *code* - *range*
    2. Return bit 1

The Linux kernel implementation of fixed-probability decoding in rc_direct, for performance reasons, doesn't include a conditional branch, but instead subtracts *range* from *code* unconditionally, and uses the resulting sign bit to both decide the bit to return, and to generate a mask that is combined with *code* and added to *range*.

VI. RESULTS:

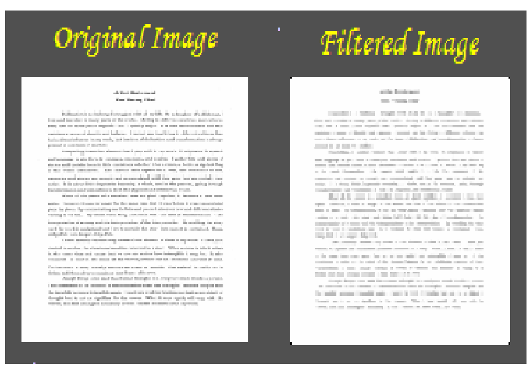

Figure 3: Original document image and its pre-processed image

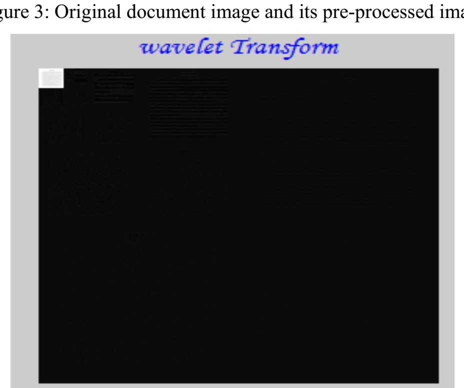

Figure 4: Wavelet transformed image

Figure 5: LZMA Decompression

Figure 6: Reconstructed original document image

Compression ratio and bits per pixel achieved through this method is

cratio =   6.5087

bitspp =   1.2291

VII. CONCLUSION

In this paper we propose a novel method in the compression of LZMA algorithm. The algorithm is used for the lossless compression. In preprocessing we remove the noise from the image. Then we quantize the image by the use wavelet transform. The wavelet transform will decompose the images. After decomposing we apply the LZMA algorithm. We get the Bit Stream of the image. We decompress the image by LZMA decoding. Finally we apply the inverse wavelet transform. Our method provides the better result than the other method.

REFERENCES

[1] "IEEE Recommended Practice for the Transfer of Power Quality Data", IEEE STD 1159.3-2003 , pp. 1-119, 2004, doi:1O.1109/IEEESTD.2004.94416
[2] S. Santoso and E.J. Powers and W.M. Grady, "Power quality disturbance data compression using wavelet transform methods", IEEE Trans. Power Deliv. 12, pp. 12501256, 1997
[3] D. Solomon, "Data compression: the complete reference", Springer, ISBN 1-84628-602-6, p. 241, 2007.
[4] G. Nigel and N. Martin, "Range encoding: An algorithm for removing redundancy from a digitized message", Video & Data Recording Conference, Southampton, UK, July 2427, 1979.
[5] H. Musmann and D. Preuss, "Comparison of Redundancy Reducing Codes 712 for Facsimile Transmission of Documents", IEEE Trans. Communications, vo1.25, no.11, pp. 1425- 1433, Nov 1977