INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

High Efficiency Adaptive Variable Block size Motion Compensation Algorithm for Video Compression Standards

Ruksana Begum Shaik[1], T.Venkata Krishnamoorthy[2]

[1]HolyMary Institute of Technology and Science,
Bogaram,Hyderabad,Andhra Pradesh,India.
murthy.tvk@gmail.com

[2]HolyMary Institute of Technology and Science,
Bogaram,Hyderabad,Andhra Pradesh,India.
rukkuhyd@gmail.com

Abstract: *A new Algorithm investigates for higher video compression and Low bit rate for different video standards. In this proposed algorithm implement Absolute Block Truncation code for High-level quantization and highly correlated Operations for getting equal perceptual video Quality. Previous work focused on implementing pixel truncation using fixed block size motion estimation. However pixel truncation fails to give satisfactory result for smaller block partition. Avoiding this problem using in this proposed algorithm. H.263, MPEG-4 Part 2, H.264/MPEG-4 AVC, and VC-1 give the encoder the ability to dynamically choose what block size will be used to represent the motion, and this proposed algorithm reduces computations, Elapsed time and increased Computations on processing, PSNR without video quality loss.*

Keywords: *MPEG-4, AVC, Absolute Block Truncation code, PSNR, Elapsed time*

1. INTRODUCTION

Video images are created from still frames shown one after another, generally at a rate of at least 15 frames per second. There are two major standards for video: PAL (Phase Alternation Line) video systems: It is a colour-encoding system used in broadcast television systems in large parts of the world.These video system are used in Europe and India. NTSC (National Television System Committee) video systems:

It is the analog television system used in most of the Americas, Japan, South Korea, Taiwan, the Philippines, Burma, some Pacific island nations and territories. NTSC is also the name of the U.S. standardization body that adopted the NTSC broadcast standard. The first black-and-white NTSC standard for broadcast was developed in 1941 and had no provision for color transmissions. In 1953 a second standard was issued, which allowed color broadcasting to be compatible with the existing stock of black-and-white receivers, while maintaining the broadcast channel bandwidth already in use. NTSC was the widely adopted broadcast color system. F is based on the NTSC video system. NTSC receiver has a tint control to perform colour correction manually. If this is not adjusted correctly, the colours may be faulty. The PAL standard automatically removes errors by utilizing phase cancellation of the color signal So, a tint control is necessary. Errors cancelled out in PAL less than that compare to NTSC standard.

1.1. NEED FOR COMPRESSION

Since the early days of the motion-picture industry, what viewers have perceived to be moving pictures have actually been sequences of still photographs? The trick, of course, is that the pictures change sufficiently often to convince the eye that what we are watching is real movement. In television, we don’t even get to see one complete picture at a time. Designers rely on the persistence of vision of the eye to create the illusion of a complete moving picture, when all that is moving are the three electron beams in the cathode ray tube. The results

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

are nevertheless quite acceptable.

With the advent of digital television, engineers have been tempted to stretch their powers of deception even further. The digital video signal is free of noise and is very robust, but alas, it occupies too much bandwidth for economical tape recording, let alone disk recording or transmission over the air. Fortunately, in most television pictures there is a lot of repetitive detail in plain backgrounds, blue sky, and common successive frames, which we can simply discard without the eye noticing that it has again been cheated. To speedup the image transmission and reduce image storage space, we need compression.

2. GLOSSARY IN MPEG COMPRESSION

There are three types of frames used in MPEG compression [2] – I frames (intra frames), P frames (predicative frames), and B frames (bi-directional frames).

- I frames are compressed without reference to any other frames. That is, they are compressed using just the information in the frame itself, in the same way still images are compressed (using the DCT, quantization, run-length encoding, etc.). This is called intra coding.
- There are generally two or more I frames each second (more often three to six), and particularly complex frames are encoded as I frames.
- P and B frames are encoded with reference to the previous frame, i.e., they are inter coded. P frames are encoded with reference to a previous frame, called forward prediction [9].
- B frames are encoded with reference to both the previous frame and the next frame. This is called forward and backward prediction.

Use of forward and backward prediction makes a high compression rate possible, because it is necessary to record only the changes from one frame to the next. A reference (I) frame plus the following B and P frames before the next I frame together define a Group of Pictures (GOP). The size of the GOP can be set to 8, 12, or 16 to optimize encoding to suit different movies and display formats.

Simply described, MPEG's basic principle is to compare two compressed images to be transmitted over the network, and using the first compressed image as a reference frame (called an I-frame), only sending the parts of following images (B- and P-frames) that differ from the reference image. The network viewing station will then reconstruct all images based on the reference image and the "difference data" contained in the B- and P-frames.

A typical sequence of I-, B-, and P-frames may look as below. Note that a P-frame may only reference a foregoing I- or P-frame, while a B-frame may reference both foregoing and coming I- and P-frames:

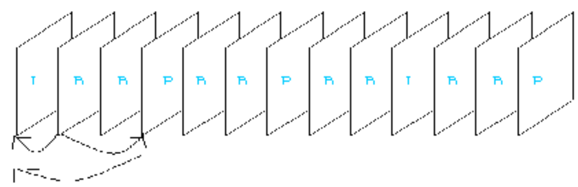


Figure 2.1(a): MPEG Frame sequence order

Macro blocks are blocks [3] of 16 pixels by 16 pixels within a frame. The encoder uses motion compensated prediction for P frames and B frames. That is, it detects macro blocks that don't change from one frame to the next, or that change only by moving. For each macro block, a search is made for the closest match in the search area of the previous picture. When a match is found, a motion vector is computed. The motion vector records how far the macro block has moved, and in what direction.

In video compression, Motion compensation describes a picture in terms of where each section of that picture came from, in a previous picture. This is often employed in video compression. A video sequence consists of a number of pictures - usually called frames. Subsequent frames are very similar, thus containing a lot of redundancy. Removing this redundancy helps achieve the goal of better compression ratios. Frames can also be predicted from future frames. The future frames then need to be encoded before the predicted frames and thus, the encoding order does not necessarily match the real frame order. Such frames are usually predicted from two directions, i.e. from

ISSN 2320-6802
INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

the I- or P-frames that immediately precede or follow the predicted frame. These bi-directionally predicted frames are called B-frames.

2.1 ADVANTAGES OF MPEG:

Due to its simplicity, M-JPEG is a good choice for use in many applications. JPEG is a widely available standard in many systems often by default. It's a simple compression/decompression technique, which means the cost, in both system time and money, for encoding and decoding is kept low. Since M-JPEG doesn't make use of a video compression technique, it generates a relatively large amount of image data that is sent across the network. The benefit of MPEG: the ability to give a relatively high image quality at a lower bit-rate (bandwidth usage). This can be especially important if the available network bandwidth is limited, or if video is to be stored (recorded) at a high frame rate and there are storage space restraints. The lower bandwidth demands come at the cost of higher complexity in encoding and decoding, which in turn contributes to a higher latency when compared to M-JPEG.4

3. ALGORITHM FOR MPEG ZOMPRESSION – VIDEO COMPONENT

1. The High Efficient Adaptive compression is done in a two-pass process. The first pass analyzes the video file to determine which frames can be compressed as I frames, which as P frames, and which as B frames. The size of the GOP and the minimum and maximum bit rates are set before the first pass.
2. The frames are divided into blocks of 16 pixels X 16 pixels called macro-blocks.
3. The RGB video signal is transformed to YUV. YUV represents a frame as a luminance component (Y) and two chrominance components (U and V). It is a better representation for compression purposes because some of the chrominance information can discard without loss of clarity to the human eye.
4. Discarding the chrominance information is called sub sampling or down sampling. Originally, each 16 X 16 block of pixels has three pieces of information associated with it – the R, G, and B components. When RGB is translated into YUV, a Y component is generated for each of the 16 X 16 pixels, but U and V components are generated only for each group of 4 pixels. (Only ¼ of the pixels are kept, where each pixel is some "average" of four neighboring ones.) Thus, for a 16 X 16 macro-block, there are four 8 X 8 blocks of luminance data (Y), but only one 8 X 8 block each for the two chrominance components, U and V. then Adaptively select block sizes , here not maintain square blocks, the blocks sizes are adaptively changes due with picture. When the frame is decoded, the missing chrominance data can be regenerated by interpolation, often simply by duplicating the averaged pixel four times.

5.For a P frame of a B frame, the encoder determines how the macro-blocks have moved from one frame to the next and then records a corresponding motion vector (how much and in what direction a block has moved) and prediction error compensation (how much the block might have "tilted" during the move) for each macro-block.

6.For I, P, and B frames, each macro-block is compressed using the discrete cosine transform.

7.The high frequency coefficients are discarded.

8.The coefficients are quantized.

9.The macro-block is diagonalized and run-length encoding is performed on it.

10.Huffman encoding is done on the remaining values.

4. MOTION ESTIMATION FOR MOVING OBJECTS

As such, each forward and backward predicted macro-block may contain 2 motion vectors, so true bi directionally predicted macro blocks will utilize 4 motion vectors. Below figure shows how a potential predicted Frame 2 can be generated from Frame 1 by using motion estimation [7].

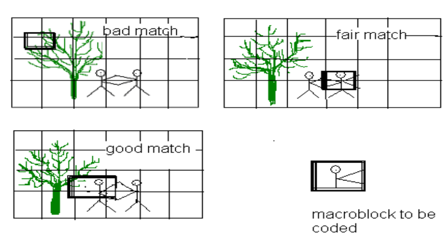


Figure 4.1: Motion Estimation Macro block

In this figure, the predicted frame is subtracted from the desired frame, leaving a (hopefully) less complicated residual error frame that can then be encoded much more efficiently than before motion estimation. It can be seen that the more accurate the motion is estimated and matched, the more likely it will be that the residual error will approach zero, and the coding efficiency will be highest. Further coding efficiency is accomplished by taking advantage of the fact that motion vectors tend to be highly correlated between macro blocks. Because of this, the horizontal component is compared to the previously valid horizontal motion vector and only the difference is coded. This same difference is calculated for the vertical component before coding. These difference codes are then described with a variable length code for maximum compression efficiency.

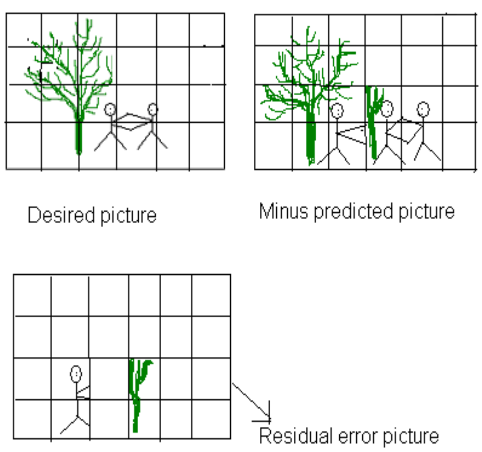


Figure 4.2: Final Motion Estimation Prediction

Of course not every macro block search will result in an acceptable match. If the encoder decides that no acceptable match exists (again, the "acceptable" criterion is not MPEG defined, and is up to the system designer) then it has the option of coding that particular macro block as an intra macro block, even though it may be in a P or B frame. In this manner, high quality video is maintained at a slight cost to coding efficiency.

The temporal prediction technique used in MPEG video is based on motion estimation. The basic premise of motion estimation is that in most cases, consecutive video frames will be similar except for changes induced by objects moving within the frames. In the trivial case of zero motion between frames (and no other differences caused by noise, etc.), it is easy for the encoder to efficiently predict the current frame as a duplicate of the prediction frame. When this is done, the only information necessary to transmit to the decoder becomes the syntactic overhead necessary to reconstruct the picture from the original reference frame. When there is motion in the images, the situation is not as simple. Motion Estimation (ME) is an important part of any video compression system, since it can achieve significant compression by exploiting the temporal redundancy existing in a video sequence. Unfortunately it is also the most computationally intensive function of the entire encoding process.

In motion estimation [5] the current image is divided into macro Blocks (MB) .Most of algorithms have been proposed for motion estimation use from BMA based (Block Matching Algorithms) methods. In this methods, motion estimation is performed for a N×M blocks of current frame, It is done with checking entire N×M blocks from search area situated in the reference frame(s) and calculating the difference between the current block and other reference blocks and finally choosing the block that has the most similarity (minimum SSD) to the early block in current frame. Then, difference of two blocks as residual (motion compensated residual) and the distance of them as motion vector, is coded and transmitted.

4.1 WEIGHTED PREDITION IN FRAME SLICES

In previous standards, bi-prediction has typically been performed with a simple (1/2, 1/2) averaging of the two prediction signals, and the prediction in the so-called P macro block types hasn't used weighting. However, in H.264/AVC [5], an encoder can specify scaling weights and offsets to be used for each prediction signal in the P and B macro blocks of a slice. The weighting and offset values can be inferred from temporally related relationships or can be specified explicitly. It is even allowed for different weights and offsets to be specified within the same slice for performing MCP using the same particular reference picture.

5) Transform, Scaling, and Quantization: Similar to previous video coding standards, H.264/AVC uses spatial transform coding of the prediction residual. However, in H.264/AVC, the transformation is applied to 4X4 blocks (instead of the larger 8 8 blocks used in previous standards), and instead of providing a theoretical inverse DCT formula to be approximated by each implementer within specified tolerances, a separable integer transform with similar properties to a 4 4 DCT is used. Its basic matrix is

ISSN 2320-6802

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS…..

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}.$$

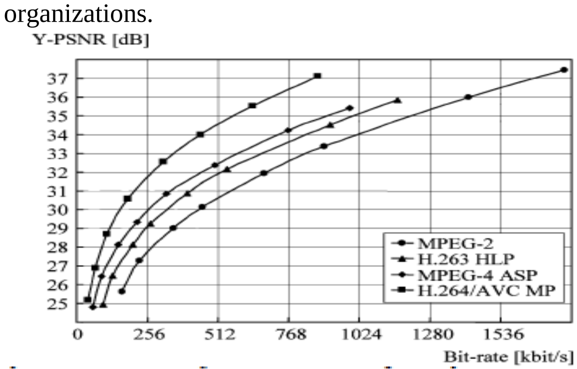
The transform coding process is similar to that in previous standards, but since the inverse transform is defined by very simple exact integer operations, inverse-transform mismatches are avoided and decoding complexity is minimized. There are several reasons for using a smaller transform size (4X4) than was used in prior standards (8X8). One of the main improvements of the present standard is the improved prediction process both for Inter and Intra. Consequently, the residual signal has less spatial correlation. This generally means that the transform has less to offer concerning de-correlation, so a 4x4 transform is essentially as efficient.

• With similar objective compression capability, the smaller 4X4 transform has visual benefits resulting in less noise around edges (referred to as "mosquito noise" or "ringing" artifacts).

• The smaller transform requires less computation and a smaller processing word length. For the luma component in the Intra 16X16 mode and for the chroma components in all Intra macro blocks, the DC coefficients of the 4X4 transform blocks undergo a second transform, with the result that the lowest-frequency transform basis functions cover the entire macro block. This additional transform is 4X4 for the processing of the luma component in Intra 16X16 mode and m is 2X2 for the processing of each chroma component in all Intra modes. Extending the length of the lowest-frequency basis functions by applying such a secondary transform tends to improve compression performance for very smooth regions. A quantization parameter (QP) is used for determining the quantization of transform coefficients in H.264/AVC. It can take on 52 values. The quantization step size is controlled logarithmically by QP rather than linearly as in previous standards, in a manner designed to reduce decoding complexity and enhance bit rate control capability. Each increase of six in QP causes a doubling of the quantization step size, so each increase of one in QP increases the step size by approximately 12%. (Often a change of step size by approximately 12% also means roughly a reduction of bit rate by approximately 12%.) The quantized transform coefficients of a block generally are scanned in a zigzag fashion and transmitted using entropy coding. The 2X2 DC coefficients of the chroma component are scanned in raster-scan order. In H.264/AVC, the same set of levels is used with all profiles, and individual implementations may support a different level for each supported profile. Fifteen levels are defined, specifying upper limits for picture size (from 99 to 36 864 macro-blocks per picture), decoder-processing rates (from 1485 to 983 040 macro-blocks per second), CPB size, DPB size, bit rate (from 64 kb/s to 240 Mb/s), etc.

5. PERFORMANCE COMPARISIONS AND RESULTS

To illustrate the performance gains that can be achieved when using H.264/AVC, we report the results of an experiment targeting video streaming applications (one of several application experiments reported in [51]). The measure of fidelity is luma peak signal-to-noise ratio (PSNR), which is the most widely, used such objective video quality measure where MSE is the mean squared error between the original and the corresponding decoding sample values. H.264/AVC has been developed and standardized collaboratively by both the ITU-T VCEG and ISO/IEC MPEG organizations.



$$\mathrm{PSNR} = 10 \log 10(255^2/\mathrm{MSE})$$

H.264/AVC represents a number of advances in standard video coding technology, in terms of coding efficiency improvement, error/loss robustness enhancement, and flexibility for effective use over a broad variety of network types and application domains. Its VCL design is based on conventional block-based motion-compensated [3] hybrid video coding concepts, but with some

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

important differences relative to prior standards, which include:

• Enhanced motion prediction capability;
• Use of a small block-size exact-match transforms;
• Adaptive in-loop de-blocking filter;
• enhanced entropy coding methods.

5.1 FULL SEARCH AND DIAMOND SEARCH ALGORITHM RESULTS

MB	CP P4	CP B2	TC	Pf	ET(Sec)	SSIM
8	51076	51076	408608	82.54	40.9220	0.671
16	11236	11236	89888	83.209	21.17	0.682
32	2116	2116	16298	83.16	13.484	0.711
64	256	256	2048	83.54	10.03100	0.790

MB: MacroBlock CP: Computations on processing
ET: Elapsed time Pf: Pefcal function

Table: 5.1 Full Search Algorithm Results

MB	CP P4	CP B2	TC	Pf	ET(Sec)	SSIM
8	6270	6203	43259	82.58	11.359	0.729
16	1461	1525	10074	83.214	9.70400	0.872
32	340	300	2056	83.17	8.984	0.882
64	47	24	280	83.568	8.70300	0.909

Table: 5.2 Diamond Search Algorithm Results for Adaptive Variable block size motion algorithm

6. CONCLUSION

Its video coding based on Conventional blocked based motion compensated [3] hybrid coding concept ,we will encode only specific part in that video in spatial and temporal coefficients and simultaneously capture local correlation between the wavelet coefficients of natural video sequences across both space and time. The proposed algorithm will adaptively adjust tilter coefficients by using the previously reconstructed adjacent blocks and their prediction blocks. Diamond search algorithm improves time and difference of peak signal to noise ratio by 0.0476 dB as compared the full search algorithm. Content adaptive search technique algorithm improves time by 31.328 sec and difference of peak signal to noise ratio by 0.0544 dB as compared the full search algorithm. Content adaptive search technique algorithm improves time by 1.765 sec and difference of peak signal to noise ratio by 0.0680 dB as compared the full search algorithm.

REFERENCES

[1]. Jiancong Luo, Ishfaq Ahmad ,Yongfang Liang, "A Fast Adaptive Motion Estimation Algorithm ",IEEE Transactions on Circuits and Systems for Video Technology VOL.16, NO.3 March 2006, 420-438.

[2]. Jiancong Luo, Ishfaq Ahmad, Viswanathan Swami Nathan, "A multistage fast motion estimation scheme for video compression", IEEE Transaction on image processing March, 2004, pp. 1441-1444.

[3]. Jiancong Luo, Ishfaq Ahmad, Yongfang Liang," Motion estimation for the content adaptive video compression" IEEE Transactions on Circuits and Systems for Video Technology, vol.18, No.7, July 2008, pp.900-909.

[4]. J. Feng, K. T. Lo, H. Mehrpour, and A. E. Karbowiak, "Adaptive block matching motion estimation algorithm for video coding," IEE Electron. Lett. vol.31, no.18, pp. 1542–1543, 1995.

[5].M.M.Azadfar, A.Ahmadi "Implementation of fast motion algorithms and comparison with the full search method in H.264", IJCSNS International Journal of Computer Science and Network Security VOL.8, No.3, March 2008, pp.139-143.

[6]. M. Ezhilarasan and P. Thambidurai"Simplified Block Matching Algorithm for Fast Motion Estimation in Video Compression", Journal of Computer Science 282-289, 2008. ISSN 1549-3636 © Science Publications

[7]. Shan Zhu and Kai-Kuang Ma " A New Diamond Search Algorithm for Fast Block Matching Mot ion Estimation", International Conference on Information, Communications and Signal Processing ICICS '97 Singapore, 9-12 September 1997 ,pp. 292-296.

[8]. Yoshihiro Noguchit, Jun Furukawatt and Hitoshi Kiyatt" A Fast Full Search Block Matching Algorithm for MPEG-4 Video", IEEE Transactions on Circuits and Systems for Video Technology, 1996, pp 61-65.

[9] S. Sharma, P. Mishra, S. Sawant, C.P. Mammen, and V.M. Gadre, "Predecision strategy for coded/non-coded mbs in mpeg4", 2004 International Conference on Signal

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

Processing and Communications (SPCOM), pp. 501 – 5, 2004.

[10] Hangu Yeo and Yu Hen Hu, "A novel architecture and processor-level design based on a new matching criterion for video compression", VLSI Signal Processing, IX, pp. 448 – 57, 1996.

[11] Seongsoo Lee, Jeong-Min Kim, and Suo-Ik Chae, "New motion estimation algorithm using adaptively quantized low bit-resolution image and its vlsi architecture for mpeg2 video encoding", IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, no. 6, pp. 734 – 44, Oct 1998.

Authors Biography

T.Venkata KrishnaMoorthy is an Assistant Professor in Department of ECE, Holymary Institute of Technology and Science. He has a total teaching Experience of 5 years. His interest includes Teaching, signal Processing, Image Processing. His research Areas are Adaptive Processing for Images and Video signals.

Ruksana Begum Shaik is an Assistant Professor in Department of ECE, Holymary Institute of Technology and Science. She has a total teaching Experience of 2 years. Her interest includes Teaching, Image Processing. Her research Areas are image processing and Digital Electronics.