VIDEO COMPRESSION AND DENOISING USING ADAPTIVE VARIABLE BLOCK SIZE MOTION COMPENSATION ALGORITHM

T.Venkata Krishna Moorthy, Asst.Professor , G.Jagadeswara Reddy

Abstract: Variable block-size motion compensation (VBSMC) is the one of the technique for video compress and denoising to select the size of the blocks adaptively. In In previous algorithms ,instead fixed blocks implements, blocks in each frame of video do not need to be fixed it changes according with image size in MPEG-4 Part and H.264/MPEG-4 AVC and using 3D DCT-based filtering technique proposed for highly correlated spatial layers taken from consecutive frames of video. We show the utility of reliable motion estimation to establish temporal correspondence across frames in order to achieve high-quality video denoising. It provides a competitive performance with state-of-the-art video denoising and compressing methods both in terms of PSNR, SSIM, increased computational ratio and image quality.

Keywords:Motion Comensation, AVC, BMC, H.264, SSIM,PSNR, MPEG-4

1.INTRODUCTION ABOUT VIDEO COMPRESS

Compression is useful because it helps reduce resources usage, such as data storage space or transmission capacity. Because compressed data must be decompressed to use, this extra processing imposes computational or other costs through decompression.

Now digital video recording and network video have entered the mainstream, attention has turned to methodologies for transmitting and storing digital data in the most effective manner to achieve lower costs and desired performance. Image compression is applied on an individual image by making use of similarities between neighbouring pixels in the image and the limitations of the human visual system. JPEG is an example of such an image compression technique.

The effectiveness of an image compression technique is given by the compression ratio, calculated as the original (uncompressed) image file size divided by the resulting (compressed) image file size. At a higher compression ratio, less bandwidth is consumed at a given frame rate. Or, if the bandwidth is kept the same, the frame rate is increased. At the same time, a higher compression ratio results in lower image quality for each individual image.

Standard of Image Compression – JPEG

JPEG is a well-known image compression method, was originally standardized in the mid-1980s in a process started by the Joint Photographic Experts Group. JPEG compression can be done at different user defined compression levels, which determine how much an image is to be compressed. The compression level selected has a direct relation to the image quality requested. Besides the compression level, the image scene itself also has an impact on the resulting compression.

When compressed video , the Video quality is a characteristic of a video passed through a video transmission/processing system, a formal or informal measure of perceived video degradation (typically, compared to the original video). Video processing systems may introduce some amounts of distortion or artifacts in the video signal, so video quality evaluation is an important problem.

With today's advances in sensor design, the image/video is relatively clean for high-end digital cameras at low sensitivities,but it remains noisy for low cost cameras at high sensitivities, e.g., low light condition, high ISO setting and high speed rate. The problem of removing image noise is still of acute and in fact growing importance with the prevalence of webcams and mobile phone cameras. In general, video data tend to be more noisy than single image due to high speed capturing rate of video camera. Video denoising aims at efficiently removing noise from all frames of a video by utilizing information in both spatial and temporal domains. Such an integrated approach is more optimal than independently applying a single-image denoising method on each frame of the video, as there exist high temporal redundancies in a video compared to a single image.

2. VEDIO COMPRESS IN MPEG -4 STANDARD

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

1.The 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 macroblocks.

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 discarded without loss of clarity to the human eye.

4.Discarding the chrominance information is called subsampling or downsampling. 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 macroblock, 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.

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 macroblocks 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 macroblock.

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

7. The high frequency coeffients are discarded.

8. The coefficients are quantized.

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

10. Huffman encoding is done on the remaining values.

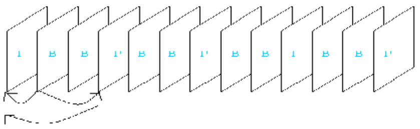

Fig: 2.1(a)

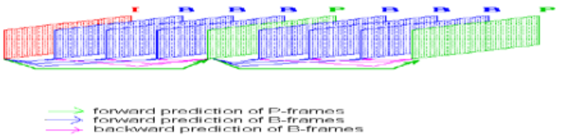

Fig: 2.1(b)

Fig2.1 (a) & (b): MPEG Frame sequence order

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.

3.1 Effect of block size on PSNR

This is yet another area which needs study and analysis. It is intuitive to think that smaller block size will give fine quality. But smaller block sizes will have more number of blocks, and hence more number of motion vectors, thereby increasing the size of encoded bit stream. Some areas in an image can be very detailed others can be uniform. To co-op up with this situation, variable block size support is required. The size of the block will be smaller for detailed areas and larger for uniform areas of the image. As it is also shown in Section 2.4 earlier. The following figure 3.7 shows how the PSNR varies with the block size.

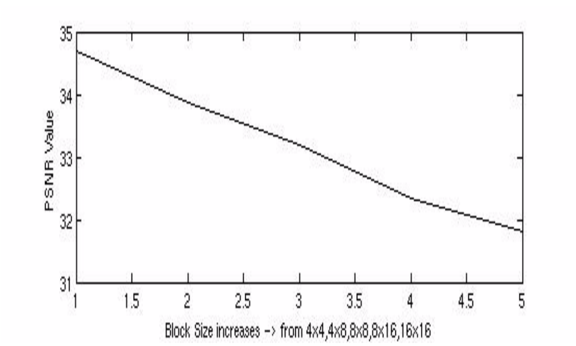

Fig.3.1 PSNR (or quality of image) decreases as the block size increases

The figure 2.4 above shows clearly that as we increase the block size, the image quality metric i.e PSNR falls. Final conclusions from the Analysis and inputs to hardware design

1). Algorithm to be used 3 iterations Modified Diamond Search Algorithm described in Section2.5.

2). If possible Variable block size will be used.

3). Frame Depth: The number of frames to be searched would be kept to 3. In case the PSNR drops below 30 for this value, the frame Depth will be reduced to 2.

4. VIDEO ENCODER AND DECODER

A lossless data compression technique removes (DCT)the redundancies without any loss of information.

Two types of encoding methods Huffman Encoding and Run Lengh Encoding.

Discrete cosine transform is a loss compression algorithm analyzes the frequency components present in the sample, and discards those frequencies which do not affect the image as the human eye perceives it. The transform tends to concentrate the energy into the low-frequency coefficients and many of the other coefficients are near-zero. The frequency domain is a better representation for the data because it makes it possible for you to separate out – and throw away – information that isn't very important to human perception. The human eye is not very sensitive to high frequency changes – especially in photographic images, so the high frequency data can, to some extent, be discarded.

A discrete cosine transform (DCT) is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers.

The DCT is often used in signal and image processing, especially for loss data compression, because it has a strong "energy compaction" property: most of the signal information tends to be concentrated in a few low-frequency components of the DCT. Discrete cosine transform is a loss compression algorithm that samples an image at regular intervals, analyzes the frequency components present in the sample, and discards those frequencies which do not affect the image as the human eye perceives it. The DCT is used in JPEG image compression, MJPEG, MPEG, and DV video compression.

There, the two-dimensional DCT of blocks are computed and the results are quantized and entropy coded. In this case, N is typically 8 and the DCT formula is applied to each row and column of the block. The result is an 8 × 8 transform coefficient array in which the (0,0) element (top-left) is the DC (zero-frequency) component and entries with increasing vertical and horizontal index values represent higher vertical and horizontal spatial frequencies.

Discrete cosine transforms (DCT) express a function or a signal in terms of a sum of sinusoids with different frequencies and amplitudes. Like the discrete Fourier transforms (DFT), a DCT operates on a function at a finite number of discrete data points. The obvious distinction between a DCT and a DFT is that the former uses only cosine functions, while the latter uses both cosines and sine's (in the form of complex exponentials).

After DCT large value is at the top-left corner. This is the DC coefficient. The remaining 63 coefficients are called the AC coefficients. The advantage of the DCT is its tendency to aggregate most of the signal in one corner of the result, as may be seen above.

Each block of the frame is processed independently with an 8x8 discrete cosine transform (DCT). This transform generates a representation of each 8x8 block in the frequency domain instead of the spatial domain. Since, as noted above, the pixels in each block of a natural video are likely to be correlated, the resulting DCT coefficients typically consist of a few large values and many small values. The relative sizes of these coefficients represent how important each one is for reconstructing the block accurately.

The coefficients are then quantized. This quantization process is loss because the true floating point values of the DCT coefficients are not preserved. Instead, the quantized coefficients are just an approximation of the true coefficients and the quality of the approximation determines the quality of the frame reconstructed from this bit stream.

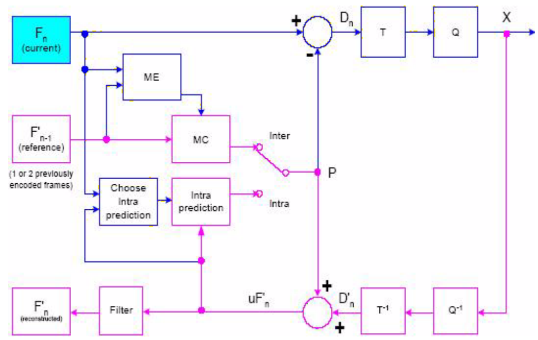

Fig 4.1 Video encoder block diagram

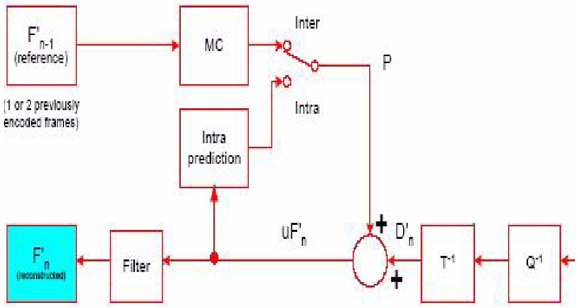

Fig.4.2 Video decoder block diagram

4.1 ADAPTIVE VARIABLE BLOCK SIZE MOTION COMPENSATION ALGORITHM FOR VIDEO DENOISING

4.1.1 Block Matching Methods

Block-matching motion estimation (BMME) is the most widely used motion estimation method for video coding. Interest in this method was initiated by Jain and Jain and he praposed a block-matching algorithm (BMA)in 1981. The current frame, ft , is first divided into blocks of M × N pels. The algorithm then assumes that all pels within the block undergo the same translational movement. Thus, the same motion vector, d=[dx,dy]T , is assigned to all pels within the block. This motion vector is estimated by searching for the best match block in a larger search window of (M +2dmx)×(N +2dmy) pels centered at the same location in a reference frame, ft−Δt , where dmx and dmy are the maximum allowed motion displacements in the horizontal and vertical directions, respectively.

In order to construct the constraint sets on a reference frame j, we first compute the motion-compensated kernel function between the reference frame j and another neighboring frame k. This frame requeirs accurate motion estimates, since incorrect motion estimation may lead to constraint sets that are not consistent with the original block-artifact-free frame.

4.1.3 MSE Criterion

Considering (k−l) as the past references frame l > 0 for backward motion estimation, the mean square error of a block of pixels computed at a displacement (i, j) in the reference frame is given by

$$MSE(i,j) = \frac{1}{N^2}\sum_{n_1=0}^{N-1}\sum_{n_2=0}^{N-1}[s(n_1,n_2,k) - s(n_1+i, n_2+j, k-i$$

4.1.4 SAD Criterion

The SAD measure at displacement (i, j) is defined as

$$SAD(i,j) = \frac{1}{N^2}\sum_{n_1=0}^{N-1}\sum_{n_2=0}^{N-1}[s(n_1,n_2,k) - s(n_1+i, n_2+j, k-l)]$$

The SAD criterion shown in equation 2.3 requires N2 computations of subtractionswith absolute values and additions N2 for each candidate block at each search position. The absence of multiplications makes this criterion computationally more attractive and facilitates easier hardware implementation.

The **structural similarity** (SSIM) index is a method for measuring the similarity between two images. The SSIM index is a full reference metric, in other words, the measuring of image quality based on an initial uncompressed or distortion-free image as reference. SSIM is designed to improve on traditional methods like peak signal-to-noise ratio (PSNR) andmean squared error (MSE), which have proved to be inconsistent with human eye perception.

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

4.1.5 Block Size

Motion compensation for each 16 × 16 macro block can be performed using a number of different block sizes and shapes. The Orignal luminance componet of each macro block) may be spilt into 4 kinds of size: 16 × 16, 16 × 8, 8 × 16, 8 × 8, as shown in Figure 2.5 (a). Each of the sub-divided regions is a macro block partition. If the 8 × 8 mode is chosen, each of the four 8×8 may be spilt into 4 kinds of size:8×8, 8×4, 4×8,4 ×4 . These partitions and sub-partitions compose to a large

In general, a large partition size is appropriate for homogeneous areas of the frame and a small partition size may be beneficial for detailed ares.

5. SIMULATION RESULTS OF MOTION ESTIMATION ALGORITHMS

Video is compressed by using different motion estimation algorithms. This project analyzes and compare the three algorithms such as full search, diamond search& Content adaptive search technique based on total check points i.e. search points, time for computing the program and visual quality of an video interims of peak signal to noise ratio(PSNR) and the picture quality measure with SSIM.

5.1 ALGORITHM

- Read the video
- Compression ratio is limited to 256M bytes that are why motion vectors are displayed in 2x256 matrixes because the video system is reader as NTSC video system.
- The video is divided into group of pictures(GOP).Each picture read as an frame
- Each frame or image is resized to 128x128.this image is further divided into macro blocks after that compression process is performed based on MPEG frame sequencing. Grand total is 76146730 elements using 80190488 bytes.
- After processing the main program in command file number of computations, total check points, processing time are displayed.
- Make sure that the reconstructe image has pixel intensities between 0 to 25. if there are an of them out this range, project them to the closet bound(0 or 255)
- Stop , if the stopping criterion is reached; else, choose an other frame k, and repeat these steps.
- Visual quality of a video is calculated by using pefcal function. Pefcal function is used for performance calculations. This function calculates the signal to noise ratio, peak signal to noise ratio and normalized root mean square error
  Below table shows corresponding results for various frames.

5.2 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

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

6 CONCLUSION

Video Compression makes it possible to use, transmit, or manipulate videos easier and faster. Many applications benefits from video compression. Compression is the process of converting data to a format that requires fewer bits. If exists a loss of information along this process, the compression is loss. Lossless compression is a manner in which no data is lost. The compression ratio obtained is usually not sufficient and loss compression is required.

MPEG reduces the amount of storage needed, increases the amount of time video can be stored, reduces the network bandwidth used by the surveillance system. So, MPEG standard is best suited for video compression than compared to all other standards.

MPEG-4 Part and H.264/MPEG-4 AVC, gives 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, increased SSIM values and reduce visible block boundaries.

.

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.

8 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.

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 4 years. His interest include Teaching, signal Processing, Image Processing. His research Areas are Adaptive Processing for Images and Video signals.

Dr.G.Jagadeswara Reddy obtained his B.Tech in ECE from KSRMCE, Kadapa and M.Tech from SJCE,Mysore. He got Ph.D Degree from JNTUH, Hyderabad. He is the principal of NEC, Nellore,A.P., He is having more than 20 years experience and has 12 technical publications in National/international journals and conferences. His research areas and publications include signal processing, compression and denoising alogorithtms. He has CoAuthored books on Digital signal Processing and Electromagnetic Waves and Transmission lines. Hi is a life member of ISTE(India).