



# REDUCTION OF MEMORY REQUIREMENTS USING SUM OF ABSOLUTE DIFFERENCES(SAD)

G.M.G Madhuri

Associate Professor, Department of Electronics and Communication Engineering,  
Pscmrct, Vijayawada, India

S. Padmavathi, P. Jhansi, N. Sai Kumar

Student, Department of Electronics and Communication Engineering,  
Pscmrct, Vijayawada, India

**ABSTRACT:** In the video compression, motion estimation is the most computationally intensive task which occupies almost 80% of the computational load. On the other hand, the demand for ultra- high-definition (UHD) introduces another four times increase on the load which ultimately increases the latency and memory requirements. The Sum of Absolute Differences (SAD) reduces memory requirements to a large extent by utilizing the absolute differences reuse strategy. SAD reuse is a critical technique to achieve real time processing in hardware, where computed SAD values are stored and hierarchically used for large blocks. This method employs three basic units namely SAD computation, SAD comparator, SAD summation which computes the SAD values and also compares them. The “Reduction of memory requirements using Sum of Absolute Differences (SAD)” is achieved using MATLAB.

**Keywords:** Video compression, Motion Estimation (ME), ultra- high-definition (UHD), Sum of Absolute Differences (SAD)

## I. INTRODUCTION

Digital video contents have become very prevalent in current periods especially after the growth of internet and smart devices. Video streaming is one of the significant requests that is constrained with by the internet connection speed. Advanced Video Coding (AVC) Hsia et al. (2010), Chen et al.(2006) [1-2] compresses the video content efficiently with high quality. The latest video compression technique used is High Efficiency Video Coding (HEVC) Sullivan et al.(2012), Bross et al.(2013) [3-4] which employs motion estimation (ME) Z.Chen et al.(2006) [5]. The motion estimation is the intensive

task which occupies 80% of the computational load. The computational complexity is more when compared to the earlier video encoding methods. The requirement for ultra-high definition (UHD) has increased which introduces a greater number of computations in ME.

Several algorithms exist for HEVC to ease the number of search points like triangle search, hexagon search, diamond search J.Y. Than et al. (1998) [6], three step searches R.Li et al.(1994) [7], predictive zonal search A.M.Tourapis et al.(2002) [8]. These search methods will drive for constant block size. For uneven block size we can employ the search methods like horizontal multigrid search and block based gradient decent search. But all these search algorithms consume more amount of time and also computational requirements for these search methods are very high.

To meet the computational requirements this paper proposes an architecture employing full search motion estimation. The proposed architecture utilizes sum of absolute differences (SAD) [9-13] strategy for finding the motion vectors (MV). SAD method is a vital technique to achieve real time processing in hardware where the SAD values are stored and are used in larger blocks.

## II. VIDEO COMPRESSION TECHNIQUES

The earlier methodology used for video compression are MPEG-1, MPEG-2, MPEG-4, AVC/H.264, H.265/HEVC. Compression is a technique which is used to encode the data into fewer bits. In simple it is the process of reducing the image data. The devices that are used for encoding and decoding process are called as Encoder and Decoder respectively. CODEC is the device which consists of both encoder and decoder.



There are two types of compression namely lossy and lossless. The lossy transmission includes 100% recovery of original data from the compressed data. In lossless transmission we cannot recover the 100% data from the compressed one.

#### A.MPEG-1:

This is the first standard of MPEG (Moving Picture Coding Experts Group) committee that was released publicly. This technique emphasises more on the compression ratio rather than on the video quality. It achieves low computation for encoding and decoding which is its main feature. The frames are encoded here based on three ways namely Intra coded frame(I-frame), Predictive coded frame(P-frame), Bi-directional coded frame(B-frame).

#### B.MPEG-2:

MPEG-2 was an extended technique of MPEG-1 which can achieve high quality of images and videos. MPEG-2 is completely compatible with MPEG-1. The main advantage in MPEG-2 is that the user can play the same video at different frame rates and resolutions. The key purpose of MPEG-2 was to provide interlaced video from conventional television, but today this is not used because the conventional television is not used. MPEG-2 is not decided for High definition television (HDTV) because these television use internets and it cannot support the high bandwidth requirements.

#### C.MPEG-4:

The MPEG-4 was an inclusive superset for MPEG-1 and MPEG-2. This technique provides low bit rates but the video quality was good. It was designed for interactive environments such as multimedia application, video communication, consumer applications on a mobile device. MPEG-4 provides better reusability of content and a greater protection of copyrights for multimedia producers. MPEG-4 uses the mechanism of motion estimation. Motion estimation (ME) observes the movement of objects in an image or a video to attain vectors symbolising the estimated motion. In simple ME is the process of determining the transformation between the motion of images or videos.

#### D.AVC/H.264:

Advanced Video Coding (AVC) is a type of standard for the compression of digital video. The main objective for the development of H.264 is to double coding efficiency compared with earlier compression techniques. AVC is also known as H.264. The methodology involved in AVC is to

encode and decode a video. AVC also uses the mechanism of ME. AVC does video compression using three phases namely predict, transform and encode. We can also encode variable block sizes of size 16x16, 8x8, 16x8, 8x16, 8x4, 4x8 or 4x4. First using the intra prediction the different variable block sizes are predicted from the boundary samples of adjacent samples then these blocks are transformed into frames. These frames are encoded using an encoder and later are decoded or a codec (A codec consists of both encoder and decoder) can be used. AVC uses inter frame compression which 2 at numerous frames and records which parts of the frames are altering. Thus, it captures only the motion inside a frame and removes the rest. This helps in the reduction of memory.

#### E. HEVC/H.265:

High Efficiency Video Coding (HEVC) is the progressive version of H.264. It achieves better coding efficiency than its antecedent AVC by incorporating a larger block size and a variety of partitioning modes. ME is the most intensive task which occupies 80% of computational load. In HEVC the basic encoding block was increased from 16x16 to 64x64 by using hierarchical quad tree structure. The mechanism involved in HEVC is that first it converts a video into pictures and then these pictures are splitted into block shaped regions by using the intra picture prediction. After the prediction methods are completed the picture goes through different loop filters and the final picture representation is stored in decoded picture buffer. Pictures stored in the buffer are used for the prediction of the rest of the images in a video. These frames are then transformed into bit stream. This bitstream is transmitted to a decoder which decompresses the bit stream to create a sequence of decoded frames. These frames are now compressed to a video. The main advantage in HEVC is it has greater flexibility in prediction modes, more sophisticated interpolation and deblocking filters compared to AVC.

### III. MOTION ESTIMATION IN H.264

Motion estimation is the core appliance of H.264/AVC. ME helps in converting a 2D image to other dimensions. To increase the data reuse and reduce the on-chip memory access ME follows inter macro block N-parallel data reuse scheme. ME basically removes redundancy between successive video frames by finding the best match of a macro-block in the current frame with the candidates in the reference frame.

An image is a binary representation of visual information, such as drawings, pictures,



graphs, or individual video frames. An image may be color or black and white. A black and white image is representation of 1's and 0's, whereas a color image is acquired in three primary colors, namely, red, green, and blue. Similarly, a color image is displayed in the three prime colors. The three components, namely, the red, green, and blue have high correlations. Because of the high correlation among the primary color components, it is not possible to process each component individually without introducing false coloring. Instead, we must first express the RGB components in three other orthogonal components and then process each component in the transformed space individually. One such component space is the Y Cb Cr space, where Y is the luma and Cb and Cr are the color differences or chroma. This is the color space adopted in the JPEG standard.

JPEG is an acronym for Joint Photographic Experts Group and defines a set of still picture grayscale and color image compression algorithms. There are two compression algorithms in the standard, one is based on 2D DCT and the other is based on spatial prediction methods. The DCT-based algorithm is intended for compression quality ranging from very good to visually indistinguishable.

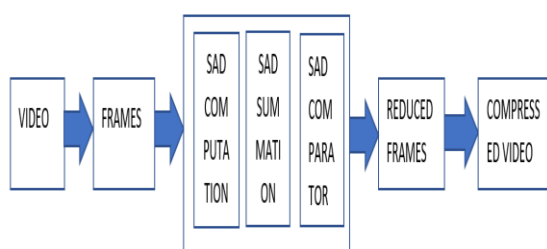
The Sum of Absolute Differences (SAD) is one error measure criterion and is widely used because of its simplicity. The SAD of a block can be calculated by using the difference between the current frame and reference frame.

#### IV. FULL SEARCH IN HEVC

Searching the best match in a complete area of frames can be achieved by comparing it with each of relative frames, which is time consuming as it searches all the frames and does  $n$  number of comparisons. To avoid these comparisons of  $n$  times, it can find the relative neighbor of that block position in source frame and if there exists a similarity then it can be retained. Thus, to do this kind of operations, we require a 'search window'. In this paper we proposed a methodology called as full search algorithm. The full search method is one of the costliest but best available algorithm. The full search algorithm is mainly used for video compression. The full search algorithm is one of the techniques that employs motion estimation where first a video is converted into frames then by using the search mechanism it eliminates the same frames and produces the compressed video.

#### V. ARCHITECTURE OF PROPOSED METHADODOLOGY

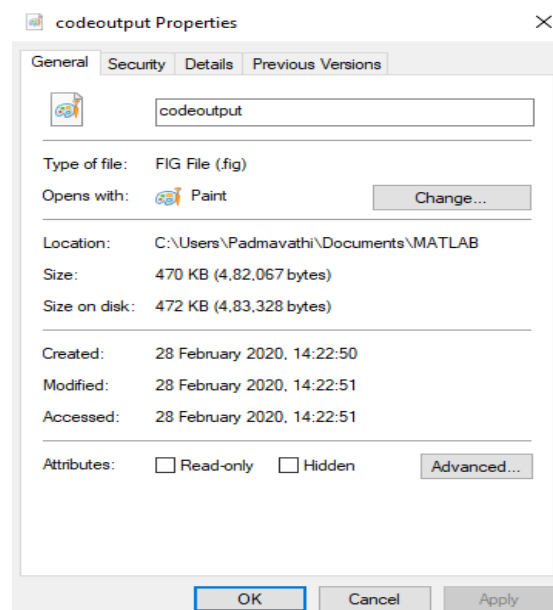
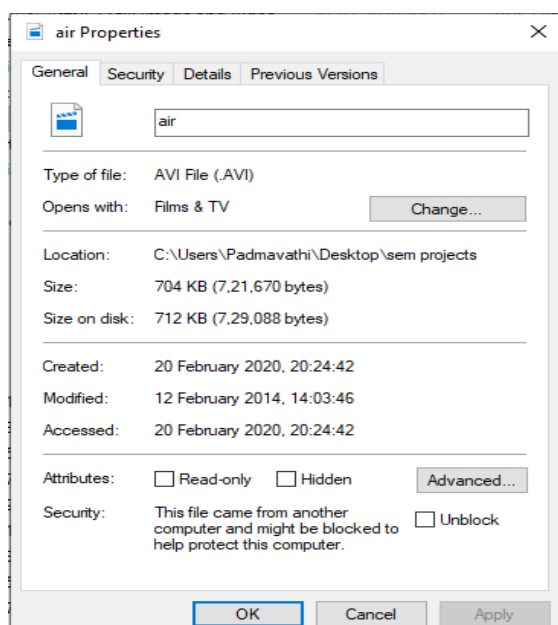
In this paper we proposed a methodology for video compression which can reduce the time when compared to HEVC. The proposed architecture for HEVC/H.265 variable block size ME uses a full search method. The full search method uses the Sum of Absolute Differences (SAD) in order to compress the video. The SAD calculates the difference between two frames, in simple it removes the temporal redundancies between two frames. The block diagram for the proposed architecture is shown below in fig.1. In the proposed architecture a video is given as input and this video is first converted into frames. These frames are further given to the SAD blocks which consists of SAD computation, SAD summation, SAD comparator. The SAD computation block calculates the SAD values of 4x4 blocks. The SAD summation block calculates the SAD values of 4x8, 8x4, 8x8, 16x4, 16x8, 16x16. The SAD comparator block calculates the rest of the SAD values and compares all the frames and produces the reduced number of frames. The implementation is done in MATLAB by giving the video as an input using a keyword called Video Reader with the path specified. In the next step to convert the video into frames, a for loop can be utilized where the frames are first read from the video and later are downloaded as the images using the "imwrite" keyword in MATLAB. Based on the block size  $N$  the imwrite function helps us convert the binary format of the frames into images (in the form of .jpg). Now to compress the number of frames we are using a full search method which is nothing but utilizing the Sum of Absolute Differences (SAD). Consider a reference frame where every frame is compared with the current frame. When both reference frame and current frame are same then the current frame is removed and thus, we are reducing the memory requirements of the input video. The search location  $k$  and search range  $R$  can help us to change the current frame. Initially the motion vectors  $mv_x$ ,  $mv_y$  are set to 0. The SAD blocks namely SAD computation, SAD summation, SAD comparator can be implemented in MATLAB using a keyword called "abs" in a for loop. A if condition helps us to differentiate whether the current frame and reference frame are having same information. Finally, these reduced number of frames are selected out of the total number of frames using the functions `iblk` and `jblk` which selects the required block of data. These reduced frames are observed and are turned to a video which is the compressed video.



The main advantage of using SAD is one of the costliest but most available methods for video compression because it reduces the temporal redundancies between the frames and thus helps in compressing the video. It achieves better coding efficiency than its antecedent AVC by incorporating a larger block size and a variety of portioning modes. The full search mechanism helps us to capture only the motion between two similar frames and thus the number of frames are reduced and so we get a compressed video with reduced memory requirements.

## VI. CONCLUSION AND RESULTS

To validate the results, a full search motion estimation using SAD is described in MATLAB. The code executed using MATLAB, takes input as the video from the specified path in your computer. The video is later converted to frames, a reference block and a current block are taken and the temporal redundancies between these frames are removed. Thus, we will be able to get reduced number of frames than the original frames that are given as input. The SAD mechanism being a loop oriented one helps us to reduce the amount of time for execution.



## VII. ACKNOWLEDGMENT

This research was supported by our institution, PSCMRCE. Padmavathi, Jhansi and Sai Kumar are thankful to our professor Mrs.G.M.G. Madhuri and to our college management who have been continuously supporting us.

## VIII. REFERNECS

- [1] S.C.Hsia, P.YHong.(2010). ‘Very large-scale integration (VLSI) implementation of low complexity variable block size motion estimation for H.264/AVC coding’ in IET Circuits Devices Syst., (pp. 414–424)
- [2] C.Y.Chen, S.Y.Chein, Y.W.Huang, T.C. Chen, T.C.Wang, L.G.Chen. (2006). ‘Analysis and architecture design of variable block size motion estimation for H.264/AVC’ in IEEE Trans. On circuits and systems, vol.53, no.3.
- [3] GJ.Sullivan, J.R.Ohm, Woo-Jin Han, T.Wiegand.(2012). ‘Overview of the High Efficiency Video Coding (HEVC) standard’ in IEEE Trans. Circuits Syst. Video Technol.,( pp. 1649–1668 )
- [4] B.Bross, W.-J.Han, J.-R.Ohm, G.J.Sullivan,Y.-K.Wang, T.Wiegand. (2013). ‘High Efficiency Video Coding (HEVC) text specification draft 10’,Joint Collaborative Team on Video Coding (JCT-VC) document JCTVC-L1003\_v34.



[5] Z.Chen, J. Xu, Y. He, (2006). 'Fast integer-pel and fractional-pel motion estimation for H.264/AVC', *J. Vis. Commun. Image Represent.*, 17, (2), (pp. 264–290).

[6] J.Y.Tham, S.Ranganath. M.Ranganath, (1998) 'A novel unrestricted center biased diamond search algorithm for block motion estimation', in *IEEE Trans. Circuits Syst. Video Technol.*, 8, (4), (pp. 369–377)

[7] R.Li, B.Zeng, M.L.Liou .(1994): 'A new three-step search algorithm for block motion estimation', in *IEEE Trans. Circuits Syst. Video Technol.*, 4, (4), (pp. 438–442)

[8] A.M.Tourapis (2002) : 'Enhanced predictive zonal search for single and multiple frame motion estimation', *SPIE Vis. Commun. Image Process.*, 4671, (pp. 1069–1079)

[9] N.C.Vayalil, A.Safari, Y. Kong, (2015): 'ASIC design in residue number system for calculating minimum sum of absolute differences' in *Tenth Int. Conf. on Computer Engineering Systems (ICCES)*, ( pp. 129–132)

[10] S. Rehman, R. Young, C. Chatwin, P. Birch, (2009) 'An FPGA Based Generic Framework for High Speed Sum of Absolute Difference Implementation', *Europ. Jour. Scient. Res.*, vol. 33, no. 1.

[11] P. Nalluri, L.N. Alves, A. Avarro,(2013) 'A novel SAD architecture for variable block size motion estimation in HEVC video coding', *International Symposium on System on Chip (SoC)*, (pp. 1-4).

[12] Purnachand Nalluri, Luis Nero Alves, Antonio Avarro.(2014). 'High speed SAD architectures for variable block size motion estimation in HEVC video coding', in *Image Processing, IEEE International Conference*, 1233 – 1237.

[13] Ahmed Medhat, Ahmed Shalaby, Mohammed S. Sayed, Maha Elsabrouty and Farhad Mehdipour. (2014). 'A Highly Parallel SAD Architecture for Motion Estimation in HEVC Encoder', in *Circuits and Systems (APCCAS), IEEE Asia Pacific Conference*, 280 - 283.