CHAPTER IV

DEVELOPMENT OF ADAPTIVE ORDER SEARCH AND TANGENT WEIGHTED TRADE-OFF FOR MOTION ESTIMATION IN H.264/AVC

4.1 INTRODUCTION

The advancement in the mobile communication technologies is of far range. The extended advancement made ease of assessing to the portable devices and their applications. The emerging advanced technologies lead the portable devices to execute the complex multimedia applications involving the video processing, image processing etc. The reach of the mobile portable devices depends on the ability to adapt such complex multimedia applications. Video compression plays an important role in reducing the bandwidth required for the transmission and the storage space needed for the application oriented environments [1, 96]. Video compression is nothing but the combined sequence of image compression. The video engaged for the compression is converted into image frames and the corresponding image compression results in the video compression. The video combines a sequence of images or frames, with related temporal and spatial dimensions. The two consecutive frames in the video sequence consist of the similar motion movement blocks or dissimilar motion blocks. The prediction of the similar motion block in the repeated sequence and reducing the similarity lead to the objective attainment of the video compression. The dissimilarity in the frames is due to the change in the position of the object or the camera, or due to the noise. The intention of the video compression is to reduce the quantity of the data used for the representation of the video frames.

The goal of the video compression is achieved by reducing the redundancy present between the frames. The redundancy present in the frames is defined by some of their predictable properties. In order to compress the data to be transmitted, determination of the redundancy present in the data by the predictable property is desirable. The predictable property predicts the real data to be transmitted and the error between the predicted data. The data free of the prediction error is only transmitted. The negligence of the data with the prediction error doesn't compromise the quality of the video [8].

Video compression algorithms are available in accumulative mass. In the vast coding algorithms, H.264/AVC is a standard video coding algorithm [100, 101]. The performance of H.264/AVC coding algorithm is better than the existing algorithms [94, 95]. The better performance provided by the H.264/AVC coding algorithm is because of the presence of some additional advanced coding tools such as spatial-domain intra prediction, variable block-size motion estimation [102, 103], a motion vector with quarter pixel accuracy, multiple reference frames, de-blocking filter, rate-distortion optimization [98, 104] and a novel entropy coding compared to the previous coding algorithms [93, 94].

Motion estimation plays an important role in video compression. Motion estimation process is the complex process because it uses multiple reference frames and variable size blocks. In the motion estimation, each frame is divided or sliced into blocks and is predicted using the reference frame for the exact match. In the matching, the consecutive blocks of the frames are tested for the match, if the match is found, the motion block movement is detected and the detected one is considered crucial among other blocks and the matching continues until the end of the blocks in the frames. The blocks containing the motion are upheld with caution during the redundancy reduction. The idea behind the prediction is that the each block is subjected to the independent motion of the reference block. The performance of the matching process depends on some of the entities associated with the blocks such as the block size and the shape, in addition to block entity the performance depends on the type of the matching precision and the motion model used.

The research efforts are committed to representing the problem associated with the matching criteria. The search algorithm with the well-maintained block entity and the best precision and motion model are the measures obligatory. The coding standard such as H.264/AVC [95] utilizes the block size of 4×4 for the coding. In H.264/AVC, the image frames are divided into the 4×4 block using an advanced tree structured block partitioning method. The H.264/AVC standard uses the multiple number of reference frames i.e. the reference frame increment increases the number of the matching situations for the motion block detection, several motion estimation models are also required, the motion vector presence in the sub block of the block is also a problem.

The full search algorithm [33] is a type of search algorithm; in the block estimation each candidate vector is checked using the adapted search algorithm. The PSNR and SSIM measurements with the reduced computational time are the need of the search algorithm. So, the researchers are now trying to find a block motion estimation algorithm with high compression [77]. Some researchers used the motion assisted merging or leaf merging after quad tree based segmentation [97] for the motion estimation. The motion assisted merging reduces the number of segments or blocks, by prediction using the same motion vectors. The encoding time reduction is also a vital entity in the search algorithm. The motion estimation process with encoding time up to 70% (one ref. frame) to 90% (five ref. frames) of total encoding time of the H.264/AVC encoder is proved in [99, 100]. Being the best video coding standard with high SSIM and PSNR, the motion estimation and compensation process in the standard H.264/AVC can be improved to reduce the computational time, size of the video.

In this chapter, we developed an adaptive order search and tangent weighted trade-off criterion function for motion estimation and compensation in the H.264/AVC coding standard. The input video is read out and the frames are extracted. The proposed hybrid search algorithm called, AOSH algorithm is applied to search through the frame after dividing the frame into a set of blocks to find out the best matching block.

The AOSH search algorithm incorporates the square and the hexagon search pattern for the searching process. The square and the hexagon search employed are adaptive order based search. The order finalizes the size of the square and the hexagon search, thereby reducing the total number of search points needed. The reduction in the search points reduces the complexity accompanied with the motion estimation. The best matching block match detected using the search algorithm needs to be evaluated. Moreover, a tangent weighted trade off criterion is newly developed for the evaluation of the matching points and chooses the best matching block with better compression performance. The trade-off criteria considered here are the rate and the distortion parameters [109]. After the suitable search point detection and evaluation using the AOSH search algorithm and tangent weighted trade-off criterion function, motion vectors are identified and they undergo transformation and quantization process. Finally, the bit stream corresponding to the data input is generated using the CAVLC coding upon the quantized values. The main contributions of the chapter are as follows:

i. AOSH:

A new search algorithm called, adaptive order square hexagon search algorithm is developed integrating the square and the hexagon search. The order of the square and the hexagon search is changed adaptively for the ease of the match block detection with less searching time and searching point.

ii. Tangent weighted trade-off criteria:

Tangent weighted trade-off function is developed by considering the bi-objective parameters; rate and distortion. The trade-off function evaluates the matching block and provides the best one suitable for improved compression performance. In the trade off criterion, rate and distortion parameters are weighted using tangent function.

4.2 AOSH SEARCH ALGORITHM AND TANGENT WEIGHTED TRADE-OFF FOR H.264/AVC

The proposed AOSH search algorithm and the tangent weighted trade-off criterion for the H.264/AVC coding standard are discussed in this section. The overall block diagram of the proposed AOSH search algorithm and tangent weighted trade-off criterion for motion estimation in H.264/AVC standard is shown in Figure 4.1.

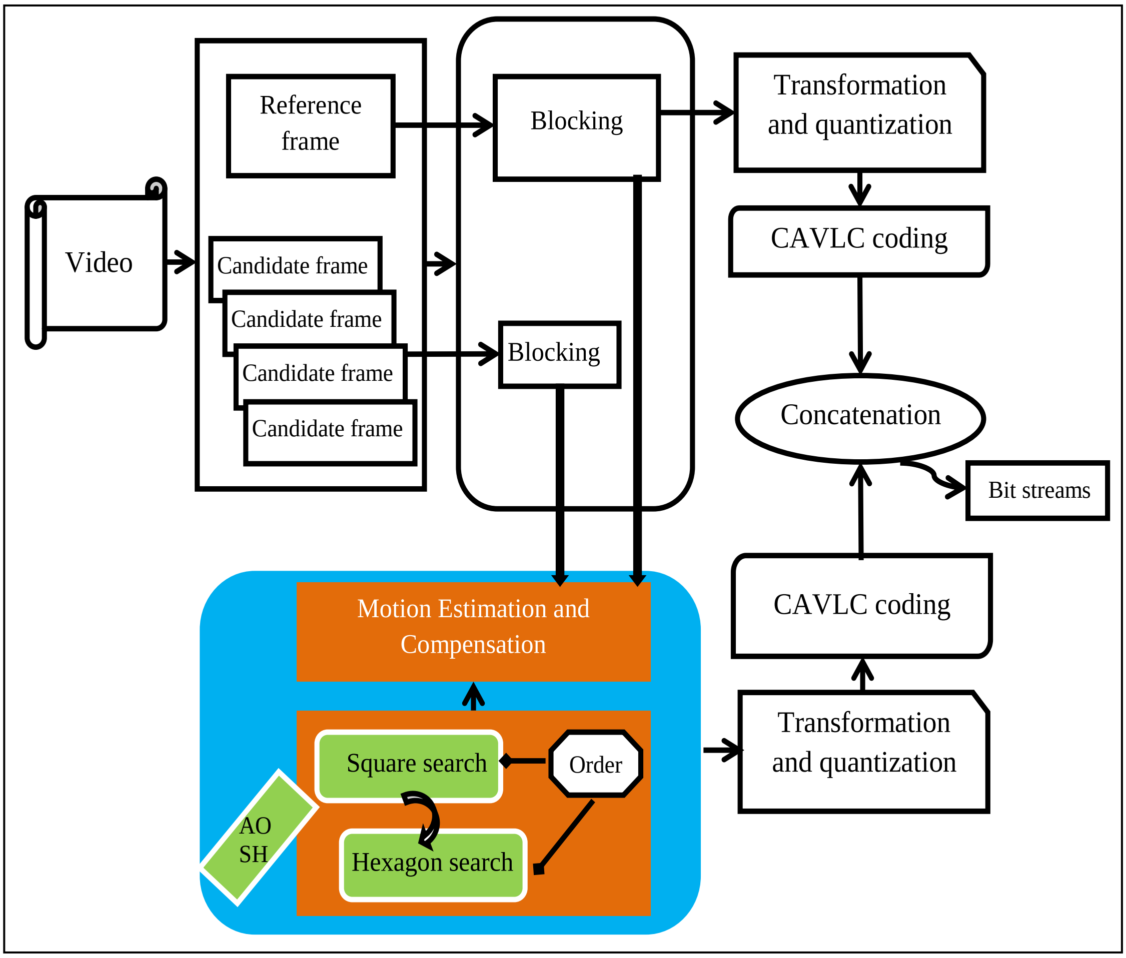


Figure 4.1: Proposed AOSH Search Algorithm for Motion Estimation

The video to be compressed is given as the input to the video compression standard. At first, the input video is read out and the image frames from the videos are extracted.

The motion movement detection in the consecutive frame is the objective of the motion estimation. The hybrid search is the procedure employed for the objective attainment. The image frames extracted from the videos are split into macro blocks, these are used to do the hybrid search.

AOSH algorithm is the integration of the square and the hexagon search algorithm. The order of the square and the hexagon search algorithm chooses the shape of the square or hexagon pattern [107]. The search algorithm is proposed with the higher order by the rate-distortion trade off. The searching process is either by hexagon or square and the order required is based on the characteristics of the proposed distortion trade-offs. By the incorporation of the square and the hexagon search, the optimal search points with the motion movement are detected in the consecutive frames by the comparison between the reference frame and the candidate frames. Upon the suitable search point selection, the motion vectors are generated for the respective motion block.

The integer transformation and the quantization are performed over the motion vectors. The encoding process is performed using the Context-Adaptive Variable-Length Coding (CAVLC) for the generation of the bit streams corresponding to the video input. The coded bit streams are transmitted with less bandwidth and the storage space aiding in many of the multimedia applications.

4.2.1 Adaptive order square search

Square search is one of the important search method used for the motion estimation procedure in the video compression standards. The searching scheme in the square search is the center biased one [106]. It has a half way top provision so that the immediate stopping of the search procedure after the optimal search point detection is possible. The square search algorithm fixes the pattern size without depending on the parameter value.

The square search algorithm proposed in this chapter i.e., the AOSH algorithm is the adaptive order square search algorithm. The normal square search algorithm is modified with the adaptive order square search process. The square search and the order of the search are adaptively chosen based on the trade-off criterion. In the square search for motion estimation, the systematic pattern initiates with five search points.

The search points are located as four search points in the corner and one at the center point, thereby forming a square shaped search pattern. The square search is not restricted to any of the direction issues. The initial search of the square search pattern includes all directions so that it can quickly detect any motion and also easier to detect the direction of the movement.

Usually, the motion movement in the video sequence is symmetrical either in the upward or downward direction. The direction restriction to the downward and the upward makes the square search more beneficial, robust and easy to implement in hardware. The symmetrical searching capability of the square search found easily detecting the best matching motion movement block. The square search is dependent on a condition that there should be no abnormal motion behaviour in the video considered for the compression. The abnormal motion behaviour is falsely detected by the square search algorithm. In adaptive order square search algorithm, the movement of the search point is searched through five points but the order of the search can be either 0 or 1. The parameter order is introduced to handle the size of the square to be searched through it. The depth of the square can be easily detected using the following Equation (4.1) which is formulated using the parameters order and a constant,

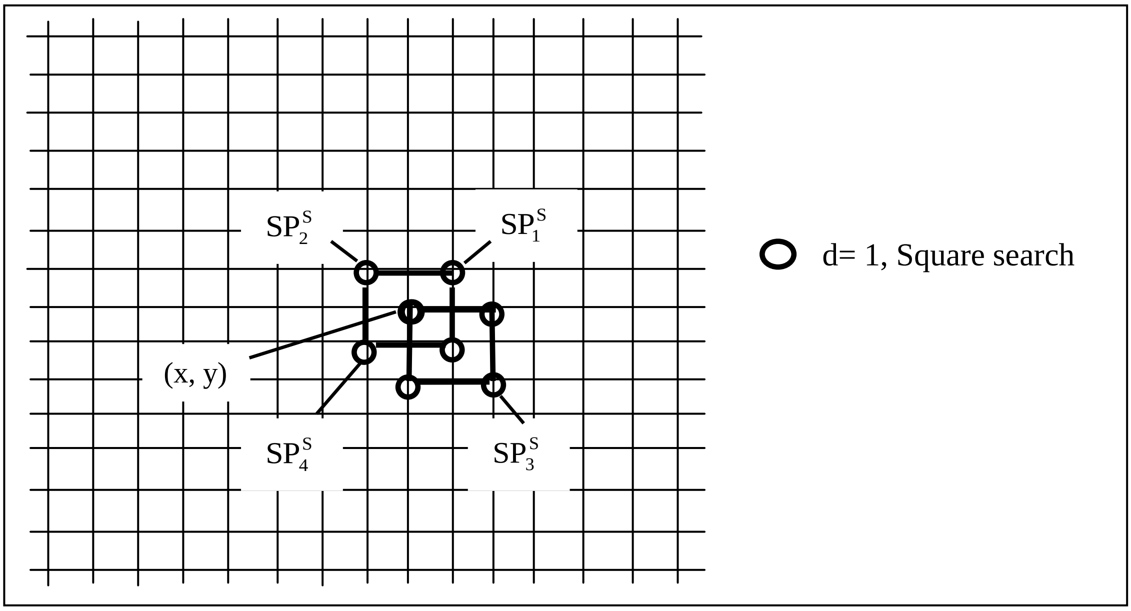
$$d = O + C \tag{4.1}$$

Here, d is the depth, O is the order and C is the constant. The size of the square contributes the small square search covering the small region or large square search covering a larger region. The size of the square depends on the order of the search and a parameter C which is the constant to be fixed manually.

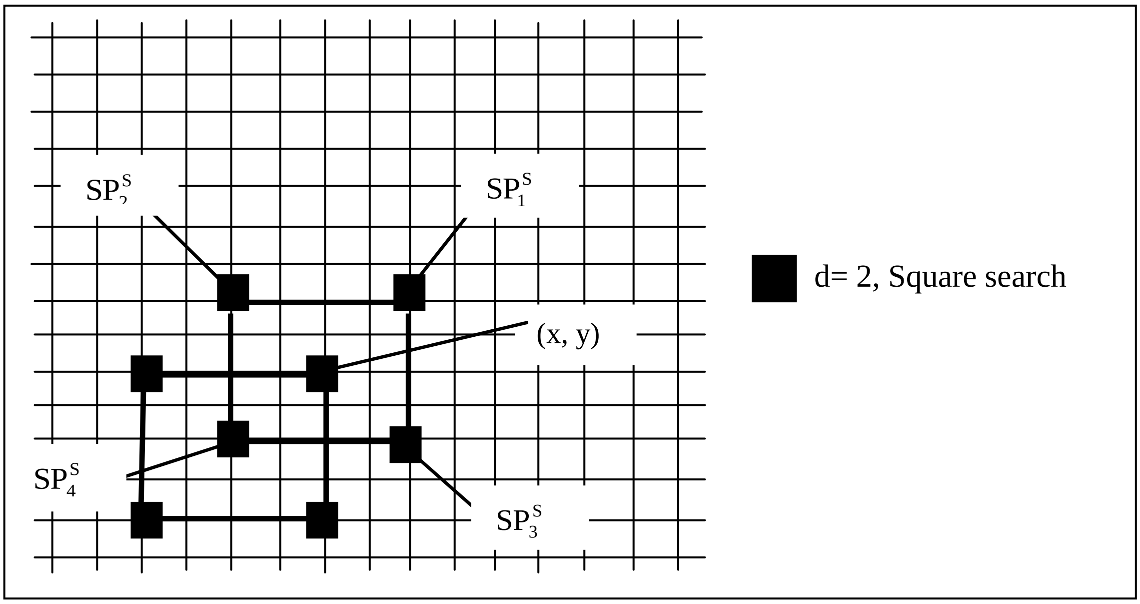
The nature of the order value which is either 0 or 1, two sets of square points i.e., large and small are introduced here for searching to find the best matching block. The size of the search pattern doesn't change the total number of search points. For the order value 0, the depth of the square search is 1 and for the order value 1, the depth of the square search is 2.

The adaptive order square search is shown in Figure 4.2 given below. Figure 4.2(a) shows the small square search with the depth value d=1. Figure 4.2(b) shows the large square search with the depth value d=2.

Let us assume that (x, y) is the center location of the search point. The searching of the best matching block through square search is possible done along the corner points with the provision of the center points. The search points defined for the square region is given by Equation (4.2).



(a)



(b)

Figure 4.2: Adaptive Order:

(a) Square search with order 0 (b) Square search with order 1

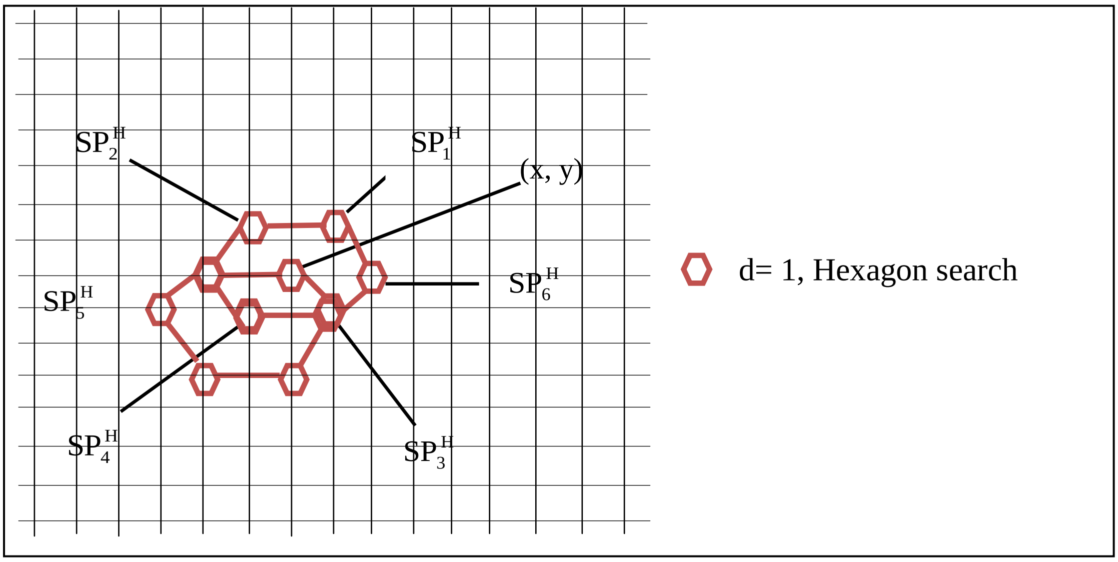
$$
\left.\begin{aligned}
SP_1^S &= (x+p, y+q)\ ; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\
SP_2^S &= (x-p, y+q)\ ; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\
SP_3^S &= (x+p, y-q)\ ; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\
SP_4^S &= (x-p, y-q)\ ; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d
\end{aligned}\right\} \tag{4.2}
$$

Where, $SP_1^S, SP_2^S, SP_3^S, SP_4^S$ are the search points defined in the square region of dimension $d \times d$. The square search points are differentiated from one another by the direction associated with it. The square search points are obtained by traversing the block to upper-right, upper-left, down-left, and down-right direction. The center point (x, y) is considered from the reference frame. Upon searching, the search point with the best match to the reference block is chosen as the suitable search points for the motion vector generation. The video with the abnormal motion behaviour is challenging to estimate the motion movement using the adaptive square search.

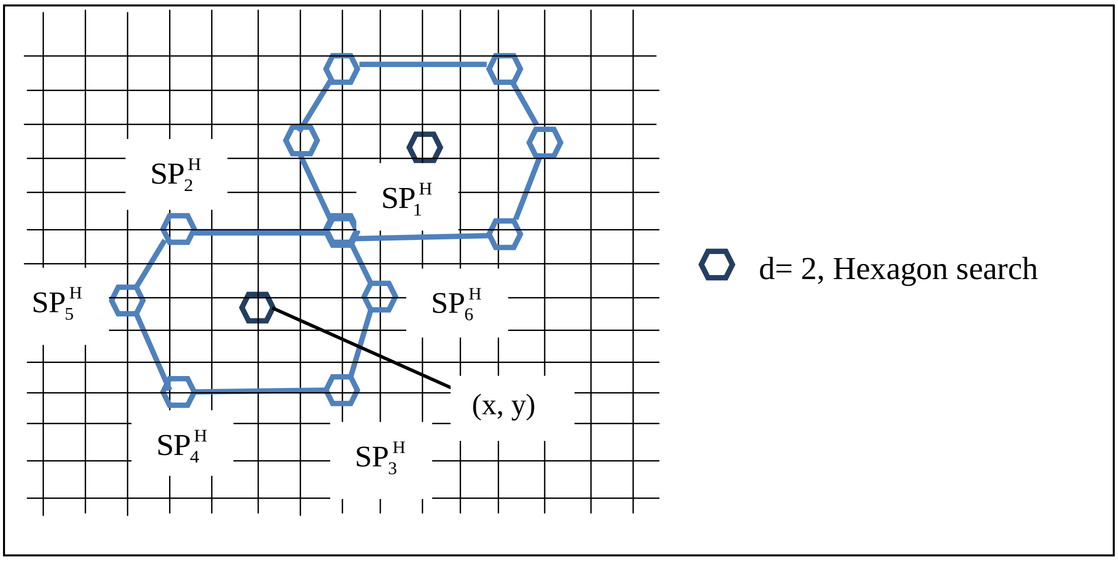
4.2.2 Adaptive order hexagon search

The hexagon search pattern [108] is another apt method for the block matching algorithm in the video compression standard. In recent years, the hexagon search algorithm is considered to be the crucial one because of the many advantages provided by it. The hexagonal search has received the considerable development over the other faster algorithms as it adopts a hexagonal search pattern to attain the best matching block with the faster processing time. The processing time of the hexagon search is lower compared to other block matching algorithms since the search point's evaluation requirement is fewer. In addition, the video sequence with the abnormal motion behaviour which is a restriction criterion in the square search is easily detected using the hexagon search algorithm. The abnormal motion movement includes the motion due to scenarios like the translation, zooming, pan and tilt. The abnormal motion movements can be easily detected using the search points defined through the hexagonal corner. This can be easily reachable if the search algorithm utilizes the hexagon search of the center pixels to find the best matching blocks. The initiation of the hexagon search algorithm starts up with seven corner searching points and a central search point. The best matching block is detected based on the rate-distortion trade off condition.

In the AOSH search algorithm, a conventional hexagonal search algorithm is modified with the adaptive order concept and the trade-off criterion. In adaptive order hexagon search algorithm, the center pixels are moved to the search points define through the hexagonal shapes. The depth of the hexagonal shape is described by the parameter; order. The order is either of values 0 or 1. The size of the hexagon to be searched through is chosen by the order value.



(a)



(b)

Figure 4.3: Adaptive Order:

(a) Hexagon Search with Order 0 (b) Hexagon Search with Order 1

The depth of the hexagon search varied according to the size of the hexagon search pattern. The depth of the hexagon is the summation of the order of the search and a constant C to be fixed manually. So, two sets of hexagonal points are introduced in the proposed adaptive order hexagonal pattern for searching to find the best matching block. They are small hexagon search pattern and large hexagon search pattern differing from one another with the area coverage.

The center points in the hexagon search are moved through the six corner points of the hexagon to discover the best matching block of the reference frame. Figure 4.3 shows the adaptive order hexagon search. The hexagon search with the depth 1 and the order 0 is represented in Figure 4.3a. Hexagon search with depth 2 and the order 1 is represented in Figure 4.3b. The 'depth 1' hexagon indicates the small hexagon shape and 'depth 2' hexagon indicates the large hexagon shape.

Let us assume that (x, y) is the center location of the search point in the hexagonal shape. The searching of the best matching block is done through the six corner points of the hexagon based on the following Equation (4.3) where the order is also an important parameter.

$$\left.\begin{array}{l} SP_1^H = (x+p, y+q) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\ SP_2^H = (x-p, y+q) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\ SP_3^H = (x+p, y-q) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\ SP_4^H = (x-p, y-q) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\ SP_5^H = (x-2*p, y) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \\ SP_6^H = (x+2*p, y) \;; \quad p = 0,\ldots,d \quad ; q = 0,\ldots,d \end{array}\right\} \qquad (4.3)$$

Where, SP_1^H , SP_2^H , SP_3^H , SP_3^H , SP_4^H , SP_5^H , and SP_6^H are the six search points defined in the hexagon corner associated with the directions. The search points are obtained by traversing the block to the upper-right, upper-left, down-left, down-right, middle-left, and middle right directions.

4.2.3 Tangent weighted trade-off criterion

The best matching block estimation in the AOSH search algorithm is performed using the trade-off function. Designing of the trade-off criterion is an important step in the motion movement block detection. The optimal center pixel selection after the incorporation of the different shapes i.e. square and the hexagon search shape is necessary. The block was chosen or the best-matched block selection should have good correlation with the reference block and it should be better in the compression performance. The block with the better compression performance comprises the fact that the number of bits generated after the coding of the respective block using the encoding scheme must be minimum. The blocks with minimum motion are stored with minimum bits and the blocks with abnormal motion behaviour utilize larger bits for the storage. The formulation of the best block using the trade-off criteria is selected considering two parameters called, rate and distortion. The parameter rate is considered because it brings the best performance through the minimization of the storage bits. The parameter distortion is considered because it brings the best performance through the minimum motion block. Using the rate and the distortion parameter, the trade-off is formulated as maximization function Equation (4.4) as follows:

$$T(b) = \eta * T_d(b) + \tau * T_r(b) \tag{4.4}$$

Here, η and τ are the weighted constant parameters of the distortion parameter, T_d and rate parameter, T_r respectively. The first parameter in the formulated Equation belonging to the distortion $T_d(b)$ is dependent on the motion-dependent parameter $M(b)$ and the weighted function $W_d(b)$. The value of the distortion parameter should be maximal for the better compression performance. The distortion parameter is given by the Equation (4.5);

$$T_d(b) = M(b) * W_d(b) \tag{4.5}$$

Here, the motion dependent parameter is given by the Equation (4.6),

$$M(b) = \frac{S(b) - A(b)}{S(b)} \tag{4.6}$$

Where, $S(b)$ is the number of pixels in the block and $A(b)$ is the number of pixels which have the dynamic values as previous frames. The motion dependent parameter $M(b)$ is computed by taking the difference in the ratio of the size of the block with respect to the pixels having the static values. The maximum value attainment in the motion dependent parameter is possible if the number of the equivalent block is similar to that of the reference block. Otherwise, the motion dependent parameter values will be towards zero. The weightage for the motion dependent parameter $M(b)$ is given by the Equation (4.7),

$$W_d(b)=\left[\frac{1}{1+exp[-S(b)]}\right] \tag{4.7}$$

Where, $S(b)$ is the number of pixels in the block.

The weightage function of $M(b)$ depends on the size of the block and the tangent function which always lies between the value 0 and 1. When the size of the block is maximal, the value of the weightage function is one. The second parameter in the trade-off maximization function is the rate parameter, $T_r(b)$ which is dependent on the rate-dependent parameter $R(b)$ and the weightage function associated with it $W_r(b)$. The rate dependent parameter controls the bit generated for the storage of the video sequence taken for the video compression. It presumes the bit required to store the block before and after CAVLC coding. The rate parameter should have maximum value for the better compression performance. The rate-parameter $T_r(b)$ is given by Equation (4.8),

$$T_r(b)=R(b)*W_r(b) \tag{4.8}$$

Here, $\mathrm{R(b)}$ is the rate dependent parameter and $\mathrm{W_r(b)}$ is the weightage function of the rate-dependent parameter. The rate-dependent parameter $\mathrm{R(b)}$ is given by Equation (4.9),

$$R(b)=\frac{R_B(b)-R_A(b)}{R_A(b)} \tag{4.9}$$

Where, $R_A(b)$ is the number of bits required for storing the block after compression and $R_B(b)$ is the number of bits required for storing the block before compression.

The rate-dependent parameter is calculated by finding the difference ratio of the bits required to store the block before and after compression. The maximum value of $R(b)$ tends to better compression performance. Otherwise, the value tends to zero. The weightage function associated with the rate dependent parameter is given by Equation (4.10),

$$W_r(b)=\left[\frac{1}{1+exp[-R_A(b)]}\right] \tag{4.10}$$

Where, $R_A(b)$ is the number of the bits required for storing the block after compression. Based on $R_A(b)$ and the tangent function value between 0 and 1, the value of the weighted function is computed for the rate-dependent parameter.

4.2.4 AOSH search algorithm

The proposed AOSH algorithm is discussed in this section. The adaptive order square search pattern and adaptive order hexagon search pattern are shown in Figure 4.4a and 4.4b respectively. From the Figure 4.4a, it is clear that depending on the order value the size of the square search varies for the reversal process. When the order value is zero, small square set search pattern is utilized for the search and when the order value is 1, the large square search pattern is utilized for the search. For the square search irrespective of the order value, the number of the search points in the square search is 5. So, the adaptive order search will do two square search processes for every searching process. Similarly, from the Figure 4.4b, it is clear that depending upon the constant value and the order value, the size of the hexagon search pattern varies with the same number of the search points. Likewise in the adaptive order hexagon search, a set of hexagon search is used for every searching process.

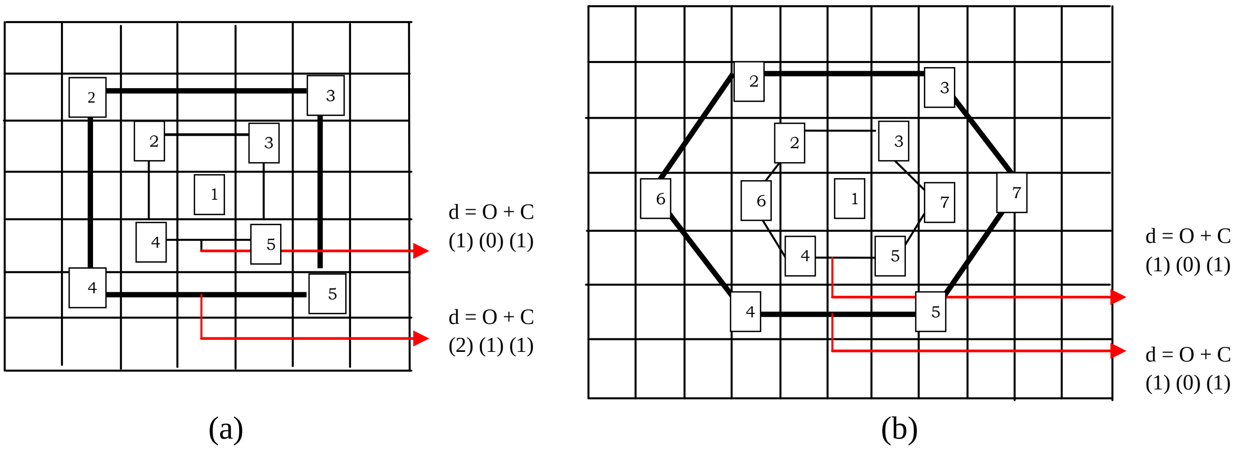


Figure 4.4: Search Pattern a) Adaptive Order Square Search b) Adaptive Order Hexagon Search

The AOSH algorithm is described using the following four steps.

AOSH algorithm:

Input: Initial Point, (x, y), constant, C; Output: Best matching point (x_b, y_b)

Procedure:

Start

Read initial point, (x, y)

Draw block for the initial point

Find the trade-off criterion, $T(b)$

For order 0 and 1

Find depth d

Get search points SP_1^S , SP_2^S , SP_3^S and SP_4^S using square

Get search points SP_1^H , SP_2^H , SP_3^H , SP_4^H , SP_5^H and SP_6^H using hexagon

Find the trade-off criterion, $T(b)$ for all the search points

End for

Select the best point (x_b, y_b)

Return the best point (x_b, y_b)

End

Step 1: Start

The center point (x, y) of the frame is located in the candid frame at the same location of the reference frame. By fixing the center point located as the center pixel, a block of size BxB is formed.

Step 2: Adaptive order square search

For the given S value, the adaptive square search is performed. The square search is done with respect to the depth and the order by traversing the four corner points of the square search pattern towards left, right, top and bottom direction. For the order 0 and 1, the search points visited are utilized to form a block by fixing the current point as the center pixel. The best matching block will be searched using the adaptive order square search process.

Step 3: Adaptive order hexagon search

After the completion of the adaptive order square search, the hexagon search is performed to find the best matching block. The traversing of the corner points in the hexagon search is performed in 6 directions. By using the search points of the hexagon with order 0 and 1, the best matching block with the motion movement similar to the reference frame is detected.

Step 4: Tangent weighted trade-off criterion

For all the pixels visited through the adaptive order square and hexagon search, trade-off function is computed for every block. The block with the maximum trade-off value is chosen as the best matching block.

4.2.5 Encoding with AOSH search algorithm

The encoding issues [111] related to the AOSH search algorithm is discussed in this section. The bit generation after the motion vector compeers using the encoding scheme in the video compression standard using the AOSH search algorithm for the motion estimation is similar to that of the H.264/AVC standard encoding scheme. For a video, the imperative attributes of the image frames are initially encoded with 8 or 16 bits. The important fields of the image frames are the height of the frame, the width of the frame, number of frames, frame rate and scalable quantization parameters. I frame in the video sequence are compressed using the coding techniques such as CAVLC [112]. I frames are the least compressible code since the decoding of I frame doesn't add up any of the image frames. Before coding, the frames are subjected to the process like blocking, integer transformation, and quantization. Every frame is encoded based on the motion vectors and the mode. The mode is used to recognize the location of the search point encoded in the bits generated. The mode is encoded within three bits of the generated bit stream. The order of the search is denoted by the first bit, the direction of the movement i.e., either upward or downward direction is denoted by the second bit, the direction of the movement either in the left or right direction is denoted by the third bit. After the mode insertion into the bits, bit coding given by the CAVLC is the slot in. This process is continued for the entire image frame extracted from the video sequence taken for the compression.

4.3 SUMMARY

In this chapter, an adaptive order search algorithm and tangent weighted trade-off function for motion estimation in the H.264/AVC video compression standard was presented. The adaptive order search algorithm which is AOSH is developed by integrating the square and hexagon search algorithm for an adaptive order of the depth. The size of the search pattern i.e., both the square and hexagon searches are chosen by the order value of the search. This AOSH search algorithm improves the searching capability of the motion estimation process with the less computation complexity. In addition to the search algorithm, a tangent weighted trade off function was designed to evaluate the search points resulted from the search algorithm. The parameters considered for the trade-off function were rate and distortion. These two enhancements are applied to the motion estimation process to improve the visual quality as well as compressive performance and also computational time. The simulation results of AOSH search and tangent weighted trade-off based video compression method are discussed in Chapter 6.