A Novel P-Skip Determination Algorithm without the Reference Frame

by

Neha Chitkara (MT13129)

Submitted in partial fulfillment of the requirements for the degree of

Master of Technology (VLSI and Embedded Systems)

to

Indraprastha Institute of Information Technology,Delhi

2015, July

1. Reviewer:

2. Reviewer:

Day of the defence:

Signature of guide:

Certificate

This is to certify that the thesis titled "A Novel P-Skip Determination Algorithm without the Reference Frame" being submitted by Neha Chitkara to the Indraprastha Institute of Information Technology Delhi, for the award of the Master of Technology (VLSI and Embedded Systems), is an original research work carried out by her under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

July, 2015

Dr. Alexander Fell
Department of Electronics and Communication
Indraprastha Institute of Information Technology, Delhi

Mr. Ajit Singh Motra
Sr. Staff Engineer, ST Microelectronics, Greater Noida

Mr. Mahesh Narain Shukla
Project Manager, ST Microelectronics, Greater Noida

Acknowledgements

I owe a debt of gratitude to Dr. Alexander Fell and Mr. Ajit Singh Motra for the vision and foresight which inspired me to conceive this project. It is also my duty to record my thankfulness to Mr. Mahesh Narain Shukla, Mr Vidit Gupta, Mr. Pankaj Kumar Bansal and Ms. Dipti Taur for selflessly helping me in bringing this project to a successful end.

I also express my regards to one and all who have, directly or indirectly, helped me in this venture.

Lastly, I would like to thank my family and friends for their fathomless encouragement and support.

Abstract

In this thesis a novel, low memory P-Skip mode decision algorithm for encoding video sequences is proposed. The existing algorithms for P-Skip estimation require reading the previous frames from the memory. The proposed algorithm saves significant memory by eliminating the need for storing the reference frame by maintaining an optimal 'Similarity Metric (SM)' for each macroblock. P-Skip decision is taken on the basis of the correlation between the similarity metrics of co-located macroblocks in consecutive frames. A Weighted Matrix (WM) based SM is chosen in this thesis. Also, the significance of the objective Video Quality Assessment (VQA) metrics, like Peak-Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) in finding the artifacts at the macroblock level, is gauged in this thesis. Experimental results show that a significant memory reduction of 97.46% is achieved with a 0.193 dB increase in the PSNR (quality) at the cost of 0.3368% increase in bitrate. Application areas include Wireless Gigabit (WiGig) use cases, low cost video calls and sharing of synthetic content over the Internet.

Contents

List of Figures

List of Tables

LIST OF TABLES

List of Acronyms

AVC Advanced Video Coding

BD Bjontegaard Delta

B-frame Bi-directional predicted frame

Cb blue-difference

CBP Coded Block Pattern

Cr red-difference

dB decibels

DC Direct Cosine

DCT Direct Cosine Transform

HEVC High Efficiency Video Coding

Intra16×16 intra prediction mode having a partition size of 16×16

Intra4×4 intra prediction mode having a partition size of 4×4

IE Intra Estimation

I-frame Intra coded frame

intra-only profile a profile of the Joint Model (JM) reference software for the H.264 encoder and decoder provided by the Joint Video Team (JVT), which permits only intra prediction for encoding and decoding

JM Joint Model

JM reference software the software for the H.264 encoder and decoder provided by the Joint Video Team (JVT)

JVT Joint Video Team

LAN Local Area Network

macroblock 16×16 pixel area

LIST OF ACRONYMS

MMS Multimedia Messaging Service

MSE Mean Square Error

MV Motion Vector

MVD Motion Vector Difference

P16×16 inter prediction mode having apartition size of 16×16

P16×8 inter prediction mode having a partition size of 16×8

P-frame Predicted frame

PMV Predicted Motion Vector

P-Skip Skip prediction mode in a P-frame

PSNR Peak-Signal-to-Noise Ratio

PXP Pixel Pipeline

QP Quantization Parameter (depicting the level of quantization)

RD Rate Distortion

SM Similarity Metric

SSIM Structural Similarity Index

SAD Sum of Absolute Differences

SATD Sum of Absolute Transformed Differences

VLC Variable Length Coding

VQA Video Quality Assessment

WM Weighted Matrix

WiGig Wireless Gigabit

Chapter 1

Introduction and Organization

1.1 Introduction

Video compression techniques are an inevitable part of transmitting videos over any network. Various compression standards are available nowadays which exploit redundancies and produce a compressed bitstream. H.264/Advanced Video Coding (AVC)(1) is currently one of the most widely used compression formats and offers a number of advantages over the preceding standards. Features like small block-size transform, flexible 16×16 pixel area (macroblock) ordering, better skipped and direct motion inference improve the efficiency of this standard. It is presently deployed in areas like broadcast over cable and satellite, conversational services over Ethernet, Local Area Network (LAN), wireless and mobile networks.

H.264 supports three frame types: Intra coded frame (I-frame), Predicted frame (P-frame) and Bi-directional predicted frame (B-frame) and two types of predictions: intra prediction and inter prediction(1). Intra prediction uses neighbouring blocks of the same frame whereas inter prediction makes use of one or more frames of the compressed video which help in defining the future frames (reference frames). I-frames are coded using intra prediction only, P-frames may be coded by intra prediction or by forward prediction using previous I or P reference frames. In a P-frame, there is a provision for skipping non-moving parts of a frame and the prediction mode is known as P-Skip.

In this thesis a novel idea for P-Skip mode decision, i.e. P-Skip estimation in video encoder without the use of any reference frame, is proposed. The existing algorithms for P-Skip estimation make use of methods, such as Mean Square Error (MSE) calculation, which consume significant memory by saving the previous frames (reference frames) in the memory. In the proposed algorithm, a metric known as the SM is stored for each macroblock and the P-Skip decision is taken based on the correlation between the SMs of co-located macroblocks in consecutive frames. This metric is calculated using the pixel values of the macroblocks, and is responsible for distinguishing between macroblocks. For comparing any two macroblocks, the difference between their respective SMs is considered. If this difference lies within a set threshold, the macroblocks are considered to have similar content and can be skipped. This approach finds its major applications in WiGig scenarios and in case of battery operated systems. WiGig specification(2), which allows devices to communicate wirelessly at multi gigabit speeds, has included H.264 as primary compression algorithm to transmit display buffers from screen to screen. It specifies the use of a profile of the Joint Model (JM) reference software for the H.264 encoder and decoder provided by the Joint Video Team (JVT), which permits only intra prediction for encoding and decoding (intra-only profile) along with the support for P-Skip. The proposed algorithm is, thus, implemented on H.264 video codec. Few other applications of the work are remote operated surveillance cameras, low cost video calls and sharing of synthetic content over the Internet.

1.2 Organization

This thesis is organized in 7 chapters. Chapter II provides an introduction into H.264 video encoder and P-Skip and chapter III presents the existing literature related to the thesis. Chapter IV defines the problem and an appropriate solution is presented in chapter V. Results are discussed in chapter VI and chapter VII concludes the thesis.

Chapter 2

Generic Encoder and P-Skip

2.1 Generic Encoder

Videos are subjected to compression before transmission over any network. A com-

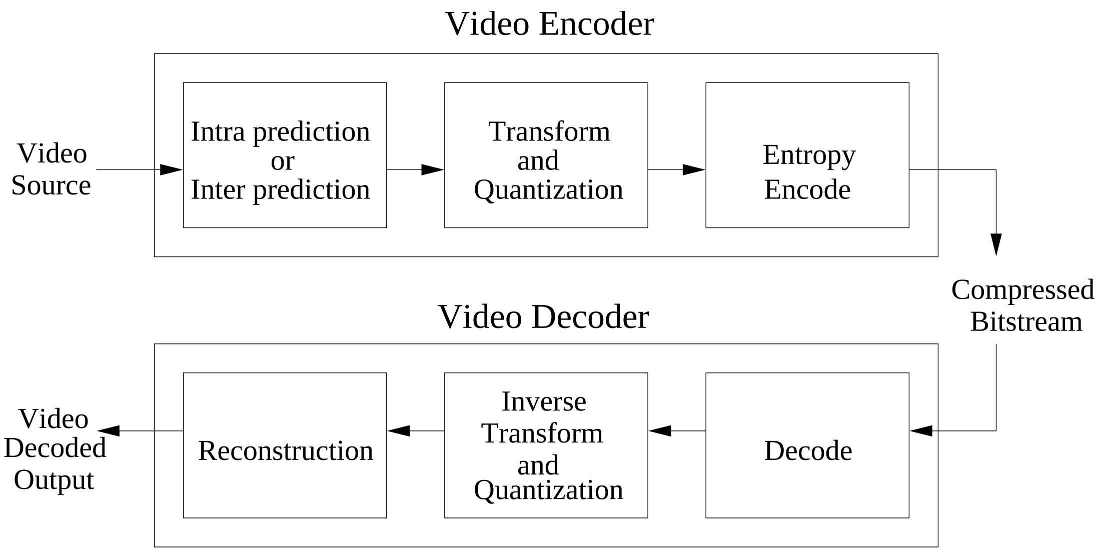


Figure 2.1: Block diagram of video coding system

pressed bitstream (sequence of bits) is generated after the input video passes through the stages of prediction, transformation, quantization and entropy encoding (refer to figure 2.1). At the decoder end, the compressed bitstream is subjected to inverse quantization, inverse transform and reconstruction.

As seen from the figure 2.2, the current frame of the input video is fed to the prediction stage and each macroblock of the frame is encoded either by using intra prediction

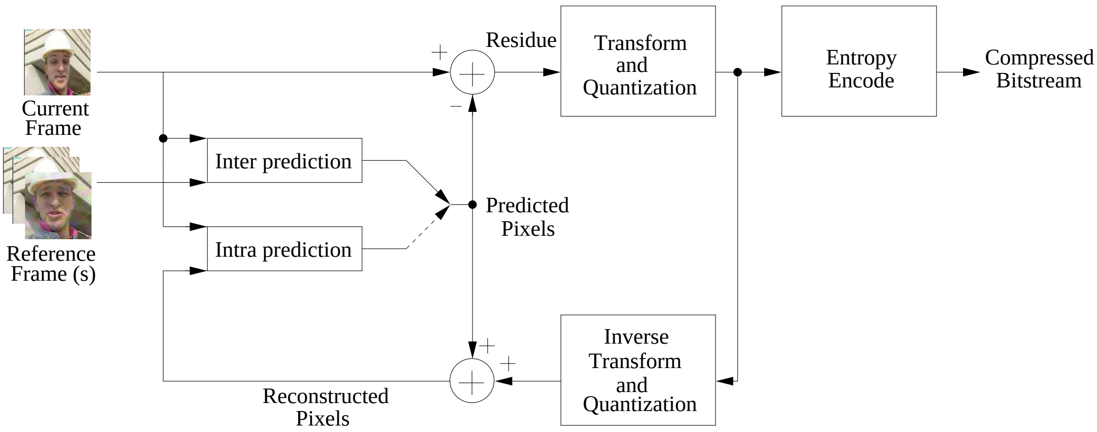


Figure 2.2: Block diagram of H.264 video encoder

or inter prediction. In the prediction stage (comprising of inter prediction and intra prediction blocks), a prediction macroblock is calculated using the reconstructed pixels. In the intra prediction, each macroblock is predicted using the reconstructed pixels of the neighbouring macroblocks. In inter prediction, each macroblock is predicted using the reconstructed macroblocks of one or more reference frames. The unit deciding the best mode receives the prediction pixels and the current macroblock. The distortion is then calculated using the Sum of Absolute Differences (SAD) metric by accumulating the absolute differences of the corresponding pixels of the two blocks in comparison.

The best mode is chosen from the available modes (intra prediction and inter prediction modes) by the H.264 encoder by calculating the Rate Distortion (RD) cost. The mode having the minimum cost is determined by the equation 2.1(3, 4).

$$J_{mode}(\theta) = D(\theta) + \lambda_{mode} \times R(\theta) \tag{2.1}$$

where $D(\theta)$ represents the distortion between the original and the reconstructed unit, $R(\theta)$ represents the number of bits needed to encode the headers, the residual transform coefficients and the Motion Vector (MV)s and λ_{mode} represents the Lagrangian multiplier (weight factor)(5).

The transform unit applies the Direct Cosine Transform (DCT) and the Hadamard transform on the residual and converts it into the frequency domain. This transformed residual is made free from insignificant information by quantization. The quantized information is sent to the compression stage which includes run length coding, entropy coding and variable length coding and thereby, a compressed bitstream is generated. The quantized information is also sent to the reconstruction unit after performing inverse quantization and inverse transform. The reconstruction unit generates the reconstructed pixels by adding the residual and the prediction pixels of the best mode for each macroblock. These reconstructed pixels are then used to encode further macroblocks.

2.2 H.264 Prediction Modes

In H.264, each macroblock in a frame can be encoded in one of the available prediction modes and the major prediction types are intra prediction and inter prediction (refer to figure 2.3). The intra prediction exploits the spatial redundancy of pixels and utilizes the current frame for prediction whereas the inter prediction makes use of the temporal redundancy between pixels of different frames and requires to save the reference frames also. Estimation is the process of finding the best match of pixel blocks. It is known as intra estimation and motion estimation for intra frame and inter frame coding respectively.

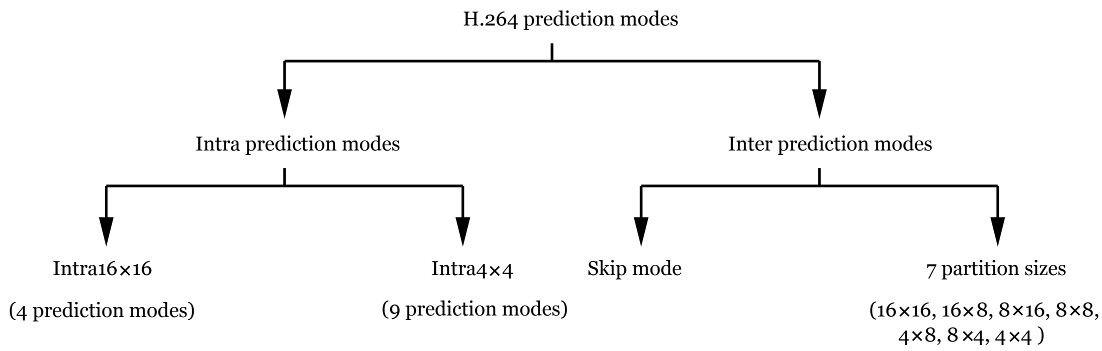


Figure 2.3: H.264 prediction modes

Intra prediction consists of mainly 2 types of modes: intra prediction mode having a partition size of 16×16 (Intra16×16) and intra prediction mode having a partition size of 4×4 (Intra4×4). The Intra16×16 mode utilizes 4 prediction modes and is well suited for plain areas whereas the Intra4×4 mode utilizes 9 prediction modes and works well for the areas with high texture(3).

Inter prediction chooses from 7 partition sizes and uses block matching algorithms to match blocks in the current frame to those of the reference frame. The difference in their positions is characterized by a vector known as the Motion Vector (MV). A Predicted Motion Vector (PMV) is calculated using the previously calculated MVs. The Motion Vector Difference (MVD) is calculated by subtracting the PMV from the current MV. The residual is calculated by subtracting the chosen best match from the current block to compensate for the motion. This process of finding the residual is known as motion compensation. The MVD and the residual are encoded and the encoded bitstream is finally transmitted. Inter prediction has a separate mode that enables skipping of macroblocks and is known as skip mode(3).

In summary, an encoder can choose from 7 inter partition block sizes, 13 intra prediction modes and a skip mode(4, 6).

2.3 P-Skip Estimation

In P-frames, a special inter prediction mode, known as the P-Skip mode, is made available for cases in which a macroblock is similar to its co-located macroblock in the reference frame. Hence, the residual and the MVD are not required to be encoded resulting in high compression ratios. For a macroblock in a P-frame to be skipped, the software for the H.264 encoder and decoder provided by the JVT (JM reference software)(7, 8) specifies four conditions:

1. The best motion compensation block size is 16×16
2. Previous frame is the reference frame
3. PMV is the best MV
4. The transform coefficients of the 16×16 block size are all quantized to zero(9)

Early encoding of skip mode can save further computations resulting in high encoding speed(10). It is to note that for a macroblock in the current frame to be encoded as P-Skip, it need not be completely the same but must be similar to the co-located macroblock in the reference frame. This is because the skip decision is taken using quantized coefficients.

Chapter 3

Related Work

The literature review is classified into 3 major sections discussing efficient P-Skip algorithms, reference frame compression techniques and the use of similarity measures in video processing respectively.

3.1 Literature Review on P-Skip

Schemes for an efficient P-Skip algorithm are explored in this section. These include early skip detection or other metrics like DCT coefficients for deciding skip. Without the RD optimization(11), skip decision is taken after expensive intra prediction and inter prediction computations(7). Early skip decision can be taken by checking the skip mode first and then proceeding to the expensive intra computations, if conditions for P-Skip are not satisfied(12, 13). There is also an option of early skip detection based on multi-step skip decision having thresholding on RD cost(14) which uses the principle that for slow moving sequences, 70% - 80% macroblocks are encoded using skip mode.

(15) decides the mode of the macroblock under consideration (candidate mode) using the mode of the co-located macroblock (co-located mode). An efficient estimation is carried out by using the fact that the chances of the current macroblock to be encoded using P-Skip mode, are greater than 80%, if the mode of the co-located macroblock is P-Skip. If the co-located mode is P-Skip, the candidate mode is chosen as P-Skip. If the co-located mode is inter prediction mode having apartition size of 16×16 (P16×16), candidate modes are P-Skip and P16×16. If the co-located mode is inter prediction mode having a partition size of 16×8 (P16×8), candidate modes are P-Skip, P16×16 and P16×8 and so on. One mode is chosen from the available candidate modes based

on the equation 2.1.

(9) improves the efficiency of the skip decision by utilizing P-Skip conditions along with the mode information of neighbouring macroblocks. For declaring a macroblock as P-Skip, two conditions must be fulfilled. Firstly, the modes of the top and the left macroblocks of the current macroblock in the current frame and the modes of the right and the left macroblocks of the co-located macroblock in the reference frame must be P-Skip. Secondly, the SAD of the current macroblock with respect to the co-located macroblock should be less than the average SAD for the skipped macroblock with respect to the co-located macroblock.

In (16), the skip decision is refined by using the cost calculated by the metric SAD on the sub-macroblock level. This metric is of lower complexity than SAD of the complete macroblock. The macroblock is skipped if this cost is less than the mean encoding cost of the previous frame.

(17) relaxes the P-Skip conditions for the boundary parts of the frame (15% of the top, bottom, left and right areas) based on the observer's sensitivity. This utilizes the fact that while watching any video, the prime focus of any viewer is on the central part of the frame. Therefore, the conditions for the boundary parts of the frame are made less stringent to allow more P-Skips at the boundary.

(18) decides skip based on values of DCT coefficients by using thresholding in the transform domain. A coefficient cost is assigned based on the level and the run length of DCT coefficients where the level refers to the absolute value of the quantized coefficient and the run length refers to the number of consecutive zero levels before reaching a non-zero level. The cumulative luma and chroma coefficient costs are calculated by summing the individual luma and chroma coefficient costs respectively and if the cumulative coefficient costs are less than a set threshold, skip is declared.

All the above techniques need to retrieve and save the reference frame for deciding P-Skip and consume high memory space in the process. The proposed algorithm saves information of the co-located macroblock in the form of an SM and does not require to save the full reference frame.

3.2 Literature Review on Reference Frame Compression

The present literature on the reference frame compression discusses memory bandwidth reduction by compressing the reference frame, storing it, decompressing it and using the decompressed reference frame to enable motion estimation. The techniques for reference frame compression can be lossless(19, 20, 21, 22) or lossy(23, 24, 25). The maximum available compression in the literature using the lossless and lossy techniques is 60%(20). The proposed algorithm provides a compression of 97.46% and does not require to save the reference frame. The information from the reference frame is saved as an SM which consumes significantly less memory than saving the compressed reference frame and then decompressing it for motion estimation.

3.3 Literature Review on Similarity Measure

(26) elaborates the use of the similarity measures in temporal video segmentation. For temporal segmentation of an uncompressed video, the techniques are broadly classified as pixel based, block based and histogram based. Pixel based techniques, such as the one in (27), require pixel to pixel comparison (intensity or colour values) in successive frames. A normalized sum of the pixel differences is calculated and compared to a threshold. However, such techniques are sensitive to camera movements.

Block based techniques, as introduced in (28, 29, 30), divide a frame into a number of blocks and each block is compared to the co-located block in the reference frame. The block based technique of (31) computes a likelihood ratio for blocks and compares it to a set threshold for deciding whether there is a sharp or a gradual transition. However, full reference frame is utilized in these techniques.

Histogram based comparison techniques(32, 33, 34) compare histograms of consecutive frames and decide on scene changes based on the difference in the histograms. These techniques are not robust as two frames with totally different content may have similar histograms.

It is deduced from the existing literature that the similarity measure based algorithms are used extensively in image and video processing. However, the reference frame elimination based on the similarity measure is an unexplored area so far and is proposed in this thesis.

Chapter 4

Problem Definition

Compression is achieved by using intra or inter prediction. As already stated, the inter prediction including P-Skip requires to save the reference frames in the memory. To read and store a frame of resolution 4K×2K with a pixel bit depth of 8, approximately 8 Mbytes memory space is required. This is significant if many frames need to be stored in the external memory.

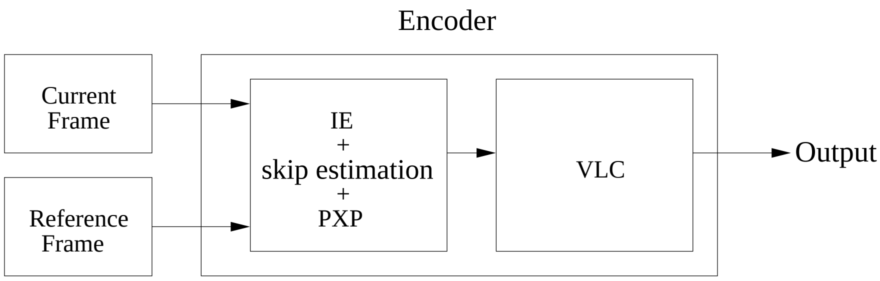


Figure 4.1: Existing implementation

For cases in which memory consumption is of high concern, inter prediction can be omitted and the intra-only profile of the Joint Model (JM) reference encoder can be used. This profile supports only intra prediction modes and therefore, does not require the reference frame for prediction. However, it results in low compression. Thus, a method which provides higher compression in the intra-only encoder without compromising on the memory requirements is addressed in this thesis. A high compression algorithm can be achieved by fusing the advantage of intra prediction (low memory requirement) and inter prediction (high compression) into a P-Skip determination algorithm which does not require to save the reference frame.

The earlier implementations for P-Skip, as shown in the figure 4.1, read the current and the reference frame from the memory. The information is fed into the module for

Intra Estimation (IE), skip estimation and Pixel Pipeline (PXP). PXP includes mode decision, transformation and quantization. The motion estimation is performed even if only P-Skip is needed (like in WiGig use cases). Finally, the decision is taken between P-Skip and the intra prediction modes by the mode decision module of the PXP.

Chapter 5

Proposed Implementation

In the design for the proposed algorithm, only the I-frames and P-frames are considered for encoding a sequence. The B-frames require the past and the future reference frames and are not considered for this algorithm. All modes for inter prediction except P-Skip are disabled and the 13 intra prediction modes and the P-Skip mode are utilized in a P-frame. The intra estimation is carried out as in the JM reference software. However, for skipping the macroblocks in a P-frame, a different approach is followed in the thesis.

The proposed algorithm, as shown in figure 5.1, eliminates the need for reading the reference frame by maintaining an SM in the form of a matrix for each frame. Each value in the matrix corresponds to one macroblock. The difference of the SM values of the co-located macroblocks in the consecutive frames is compared to a threshold value and the skip decision is taken. The standard intra prediction process is followed for the macroblocks which are not skipped.

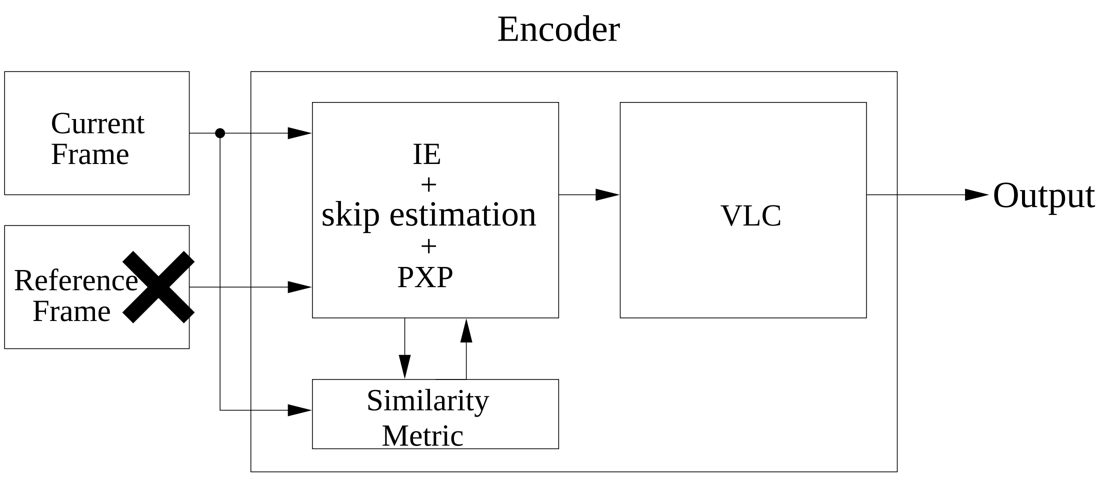


Figure 5.1: Proposed implementation

5.1 Algorithm

The adopted procedure is depicted by the algorithm flowchart in figure 5.2. For each sequential frame N with $N \geq 1$, an SM is saved for each macroblock. From the second frame onwards, the SM of the current macroblock in the current frame N is compared with the SM of the co-located macroblock of the previous frame $N-1$. If the difference of the SM values of the two macroblocks is less than the set threshold T, the mode of the macroblock is set to P-Skip. If the SM difference is greater than the threshold, the mode of the macroblock is set to either Intra16×16 or Intra4×4. This decision is taken according to the standard procedure followed by the JM reference encoder for determining the mode of a macroblock. The same procedure is followed for all macroblocks of the frame and for all frames of the video sequence.

Note that in the earlier implementations of P-Skip, there may be cases at low bitrates when the number of non-zero coefficients after quantization are truncated to zero and the MV points to a perceptibly different area. Anomalies, known as artifacts, can be seen in such cases. Since the proposed method is based on the concept of SM, an uncorrelated area will never be declared as skip and such artifacts do not emerge at low bitrates.

5.2 Similarity Metric

The SM is used to quantify the resemblance between two co-located macroblocks. It is dependent on the sub-block size, the pixel values of that sub-block and the Quantization Parameter (depicting the level of quantization) (QP) at which the encoder is presently working. A myriad number of metrics such as average, weighted average, mean absolute difference, mode, median, likelihood ratio can be utilized as an SM. However, the SM must be chosen carefully as some metrics may cause artifacts due to faulty decisions.

Table 5.1 lists examples of some experimented SMs which produce high bitrate gains with artifacts. The results (Bjontegaard Delta (BD)-Bitrate and BD-PSNR(35)) mentioned in table 5.1 are obtained by comparing the P-Skip disabled JM reference encoder configuration and the P-Skip enabled (proposed algorithm) configuration with motion estimation disabled in both the cases. The table shows that the proposed algorithm gives a considerable bitrate gain (25% - 49.25%) over the P-Skip disabled JM reference encoder configuration. However, these SMs produce artifacts. Thus, an SM must be devised which provides artifact-free compression.

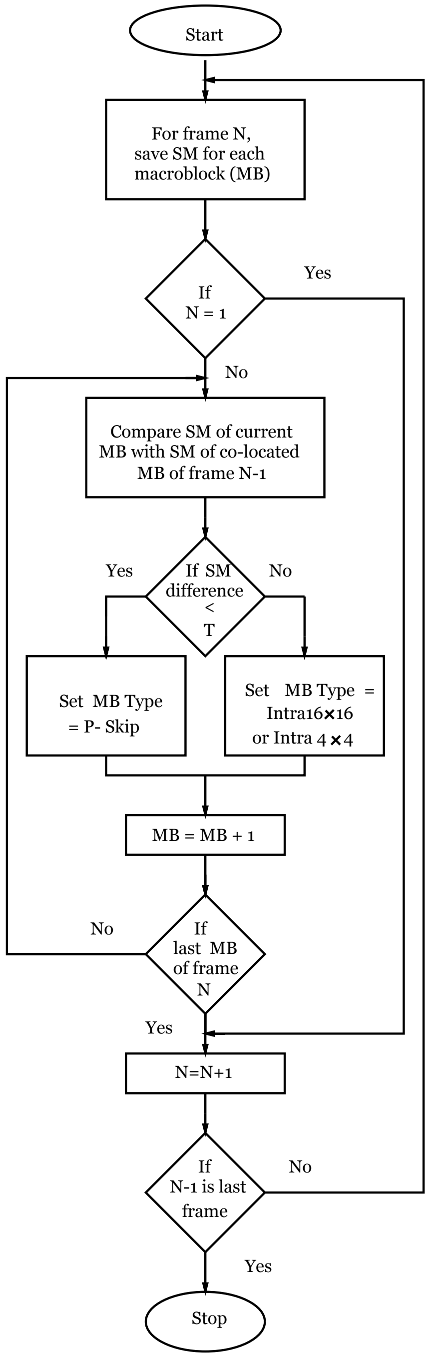


Figure 5.2: Algorithm flowchart

Table 5.1: Examples of SMs which produce artifacts

Methods	BD-Rate (%)	Artifacts
Luma average	-49.25	Yes
Luma average with 8×8 partitions	-43.3	Yes
WM with linear finite shift register	-34	Yes
Different partition sizes (16x4 and 4x16)	-25	Yes
Simple luma Mean Absolute Deviation	-30	Yes

5.3 Sample Similarity Metric

The following section presents a sample SM which detects P-Skips correctly. The procedure adopted to obtain the SM in case of a frame sequence in the YUV 4:2:0 format(36) is as follows:

For the luma pixel data (16×16 pixel areas), each macroblock is partitioned into 4 blocks, each of 8×8 dimensions. A total of six partitions are considered: four for the luma component and one each for the chroma components blue-difference (Cb) and red-difference (Cr) chroma component (8×8 pixel areas). This is shown in the figure 5.3. Weighted average is calculated on the luma and chroma pixel data using luma WM and chroma WM respectively as shown in the equations 5.1, 5.2 and 5.3.

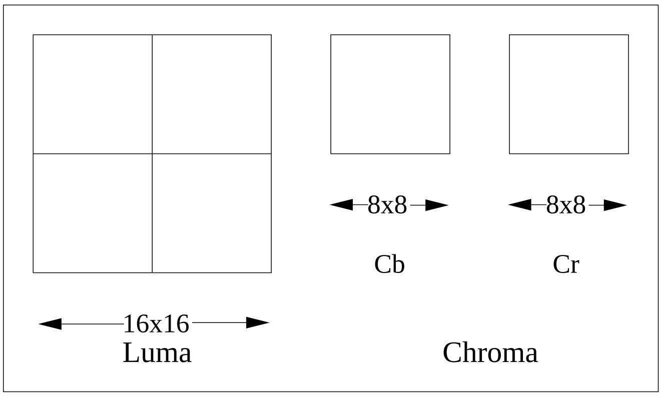


Figure 5.3: Chosen sub-block sizes for the sample SM

In the proposed implementation,

$$SM_{Luma} = (\sum_{j=1}^{j=8}\sum_{i=1}^{i=8} \alpha\_luma_{ij} \times pix\_luma_{ij})/div\_fac \tag{5.1}$$

$$SM_{Cb} = (\sum_{j=1}^{j=8}\sum_{i=1}^{i=8} \alpha\_chroma_{ij} \times pix\_cb_{ij})/div\_fac \quad (5.2)$$

$$SM_{Cr} = (\sum_{j=1}^{j=8}\sum_{i=1}^{i=8} \alpha\_chroma_{ij} \times pix\_cr_{ij})/div\_fac \quad (5.3)$$

where the sub-block size is of 8×8 pixels, α_luma is the luma WM weighting factor, α_chroma is the chroma WM weighting factor, pix_luma, pix_cb and pix_cr are the original pixel values for luma, Cb and Cr respectively. The division factor is represented by div_fac and it is equal to 128 for both the luma and chroma. SM_{Luma} is calculated for all the four partitions of the luma macroblock data.

5.4 Weighted Matrix

Averages are one of the most powerful yet simplest tools in statistics. However, simple average fails in the cases where the net variation becomes zero. Consider a scenario in the figure 5.4, in which an object moves within an 8×8 sub-block having the same background with respect to the co-located frame. The black blocks depict the current position of the object under consideration and the shaded block depicts its previous position. In this case, the average value for the two blocks is the same. This problem is solved by attaching weights to the pixel positions. The weighted averages for these two blocks come out to be different and the algorithm does not misjudge these two blocks as similar. This concept of WM is used to capture the relative positions of the pixels within a macroblock.

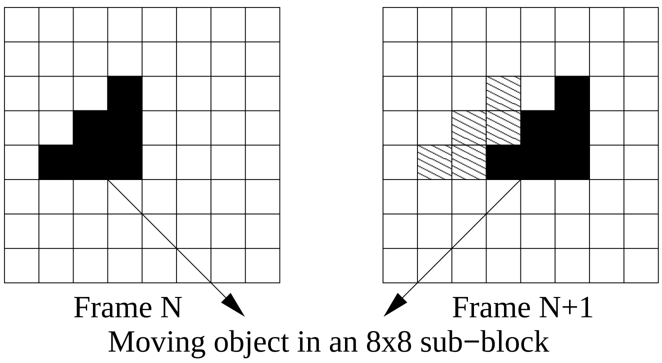
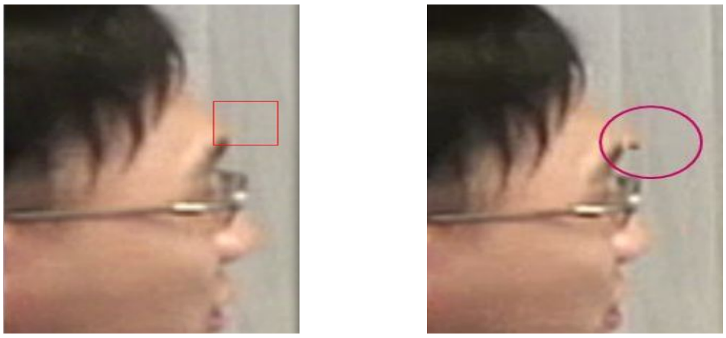


Figure 5.4: Example showing object moving in an 8×8 sub-block when background is the same

In this application, a WM must satisfy the following three conditions:



(a) Object exiting macroblock in frame N-1

(b) Frame N with artifact due to wrong skip detection

Figure 5.5: Example depicting effect of lower weights at corners

1. Each pixel must use a significantly different weight irrespective of its direction of movement
2. There must be higher weights at the corners
3. Neighbouring weights must be different

The condition 1 ensures that movement in any direction is captured effectively by the WM and condition 3 increases the probability of detecting the motion correctly. The condition 2 utilizes the fact that any motion between the consecutive frames is subtle and the pixel change at the corners is more significant than at the centre. This is due to the fact that the centre pixels remain within the macroblock boundaries even after motion whereas the corner pixels become a part of the neighbouring macroblocks. Thus, the corner pixels have more significance than the centre pixels. This is proved using figure 5.5(a) which depicts a similar instance in which the object (part of an eyebrow of the person moving to the left side) is exiting the macroblock. In the next frame (refer to figure 5.5(b)), an artifact is observed due to wrong P-Skip decision. Higher weights at the corners amplify the contribution of the corner pixels and such artifacts are eliminated.

The luma WM is a 16×16 matrix corresponding to each macroblock. An example of the luma WM, which satisfies these conditions is shown in figure 5.6. A pattern is followed while constructing the 8×8 sub-blocks of the WM with consecutive odd numbers being placed firstly after a horizontal offset ($offset\_x$) of 4 and then a vertical offset ($offset\_y$) of 4. The placement is folded, i.e. after the completion of any round of placement (complete traversal of the matrix), the next value is placed at the next free location from the beginning by following the same pattern. This pattern is followed to allow high differences in weights of neighbouring positions (condition 3). This 8×8 sub-block is mirrored in all the remaining directions, i.e. it is flipped vertically, horizontally and diagonally to form a 16×16 luma WM. This mirroring ensures higher weights at the corners and lower weights at the centre of the 16×16 luma WM (condition 2).

127	119	111	103	125	117	109	101	101	109	117	125	103	111	119	127
95	87	79	71	93	85	77	69	69	77	85	93	71	79	87	95
63	55	47	39	61	53	45	37	37	45	53	61	39	47	55	63
31	23	15	07	29	21	13	05	05	13	21	29	07	15	23	31
123	115	107	99	121	113	105	97	97	105	113	121	99	107	115	123
91	83	75	67	89	81	73	65	65	73	81	89	67	75	83	91
59	51	43	35	57	49	41	33	33	41	49	57	35	43	51	59
27	19	11	03	25	17	09	01	01	09	17	25	03	11	19	27
27	19	11	03	25	17	09	01	01	09	17	25	03	11	19	27
59	51	43	35	57	49	41	33	33	41	49	57	35	43	51	59
91	83	75	67	89	81	73	65	65	73	81	89	67	75	83	91
123	115	107	99	121	113	105	97	97	105	113	121	99	107	115	123
31	23	15	07	29	21	13	05	05	13	21	29	07	15	23	31
63	55	47	39	61	53	45	37	37	45	53	61	39	47	55	63
95	87	79	71	93	85	77	69	69	77	85	93	71	79	87	95
127	119	111	103	125	117	109	101	101	109	117	125	103	111	119	127

Figure 5.6: Sample luma WM

The chroma WM is an 8×8 matrix corresponding to each chroma component. An 8×8 sub-block of the luma WM cannot be used as the chroma WM because of low weights at the corners. The chroma WM, as shown in the figure 5.7, is constructed by swapping two rows and then two columns (shaded rows and columns in the figure) of the bottom-right 8×8 sub-block of the luma WM. The re-ordering ensures that the corner pixels of the chroma WM have higher values than the centre pixels (condition 2).

Note that other WMs or an altogether different kind of SM can also be used. The presented methodology is just one example of the procedure followed in this thesis.

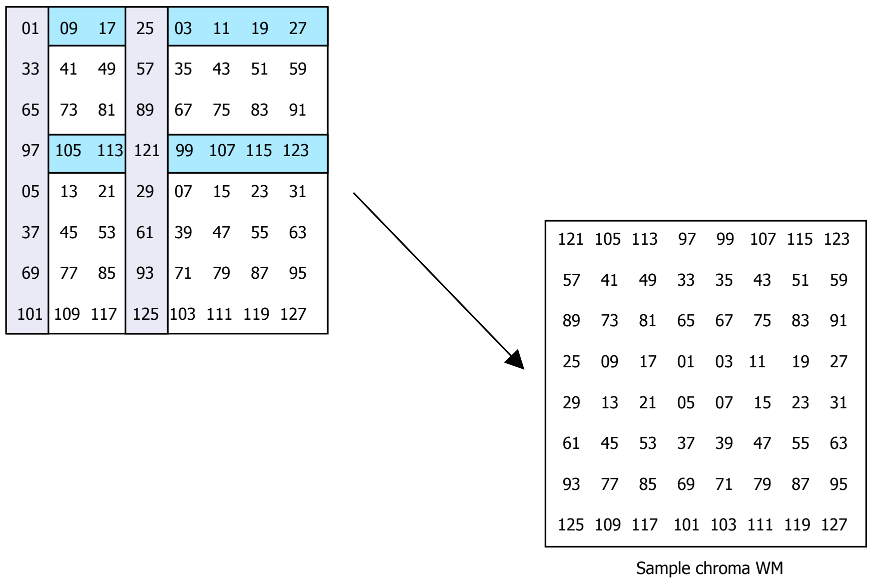


Figure 5.7: Sample chroma WM

5.5 Variations of sample Similarity Metric

Table 5.2 shows results for a few variations of the working SM. The results mentioned in table 5.2 are obtained by comparing the P-Skip disabled JM reference encoder configuration and the P-Skip enabled (proposed algorithm) configuration with motion estimation disabled in both the cases. WM_coeff refers to the range of coefficients used in the WM. The 1-64 WM ($1 \leq WM\_coeff \leq 64$) and the 1-255 WM ($1 \leq WM\_coeff \leq 255$) are constructed in a similar manner as the 1-127 WM ($1 \leq WM\_coeff \leq 127$) is constructed. No_part refers to the number of partitions of a macroblock and this decides the sub-block size.

The memory consumption by a partition in bits can be calculated by using the equation 5.4 where the number of bits per pixel is represented by *bpp*. The maximum value of the pixel is calculated by using the formula $2^{bpp} - 1$ and is equal to 255 for an 8 bit pixel.

$$Memory\ per\ partition = \log_2\left(\frac{\sum(WM\_coeff)\ \times\ (2^{bpp}-1)}{div\_fac}\right)\ bits \tag{5.4}$$

A variation of the sample SM, having $1 \leq WM\_coeff \leq 64$, with the luma macroblock divided into 4 partitions of 8×8 pixels each and with 4 partitions for Cb and Cr each, provides a bitrate gain of 11.9% with 0.74 dB increase in PSNR. Such a configuration has a total of 12 partitions (4 for luma, 4 for Cb and 4 for Cr) and uses 12 bits per partition (refer to equation 5.4). However, a variation of the sample SM using 13 bits per partition, having $1 \leq WM\_coeff \leq 64$, with the luma macroblock divided into 4 partitions of 8×8 pixels each and with one partition for Cb and Cr each provides a bitrate gain of 6.9% with 0.43 dB increase in PSNR.

Table 5.2: Other variations of the chosen SM which do not produce artifacts

WM_coeff	**Luma**		**Chroma (Cb or Cr)**		**Results**		**Memory per macroblock (bits)**		
	No_part	Div_fac	No_part	Div_fac	**BD-Rate (%)**	**BD-PSNR (dB)**	**Memory per partition**	**Number of partitions**	**Total memory**
1-64	4	256	4	64	-11.9	0.74	12	12	144
1-127	4	256	4	64	-14.57	0.91	12	12	144
1-64	4	128	1	128	-6.9	0.43	13	6	78
1-127	4	256	1	256	-12.48	0.78	13	6	78
1-255	4	128	1	128	-13.57	0.84	14	6	84

5.6 Threshold Calculation

Video sequences are analysed subjectively and the threshold is fixed to a value where no artifacts or abnormalities are observed. The thresholding process followed in this thesis, is not stringent and can be adjusted for the required quality and bitrate gain. This makes the proposed algorithm applicable to areas like object recognition, where minor artifacts do not alter the overall functionality of the system and where a higher bitrate gain is desirable.

5.7 Video Quality Assessment

Assessing the quality of a video, after being passed through a video processing or transmission system, is very crucial. The VQA techniques can either be objective or subjective. The objective VQA refers to the assessment of the quality of a video using mathematical models whereas the subjective VQA refers to the analysis of the video based on the user's perception(37).

The objective quality metrics like MSE, PSNR and SSIM(38, 39) provide an abstract insight into the quality of a video. PSNR is the logarithmic ratio of the maximum signal power to the noise power and is measured in decibels (dB) (refer to equation 5.5).

$$PSNR(r,t) = 10\log_{10}\left(\frac{255^2}{MSE(r,t)}\right)\ dB \tag{5.5}$$

where r and t refer to the reference and the test image of $M \times N$ dimensions respectively. MSE is mathematically defined by equation 5.6.

$$MSE(r,t) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (r_{ij} - t_{ij})^2 \quad (5.6)$$

SSIM is a ratio index, ranging from [0,1], which describes the extent of the structural similarity between two images. The SSIM value of one depicts the maximum similarity(39) (refer to equation 5.7).

$$SSIM(r,t) = \left(\frac{2\mu_r \mu_t + C_1}{\mu_r^2 + \mu_t^2 + C_1} \right) \left(\frac{2\sigma_r \sigma_t + C_2}{\sigma_r^2 + \sigma_t^2 + C_2} \right) \left(\frac{\sigma_{rt} + C_3}{\sigma_r \sigma_t + C_3} \right) \quad (5.7)$$

where μ_r and μ_t refer to the mean luminance of the reference frame and the test frame respectively, σ_r and σ_t refer to the standard deviation of the reference and the test frame respectively and σ_{rt} refers to the covariance between the reference frame and the test frame. C_1, C_2 and C_3 are constants.

The frame level PSNR and SSIM values are used to quantify the quality of a frame in a video. Artifacts, if present, are observed at the macroblock level. The usefulness of these metrics in providing information about the presence or absence of artifacts at the macroblock level is also gauged in this section.

Table 5.3: PSNR and SSIM at macroblock level

Figure	**Arti-facts**	**Macroblock PSNR (dB)**	**SSIM per 8×8 block**			
			Block 1	**Block 2**	**Block 3**	**Block 4**
5.8 (a)	Yes	31.006	0.9716	0.9990	0.9776	0.9362
5.8 (b)	Yes	43	0.9995	0.9995	0.9999	0.9972
5.8 (c)	Yes	33.55	0.9477	0.75001	0.8847	0.6475
5.8 (d)	No	48	0.9822	0.3857	0.9848	0.9320

Table 5.3 shows values of these objective quality metrics for selected macroblocks. According to the JM reference software, the PSNR is calculated at the macroblock level and the SSIM is calculated per 8×8 sub-blocks of a macroblock.

It is generally accepted that the frames with a PSNR value of less than 35 dB and an SSIM value of less than 0.9 are considered to be of inferior quality(40). The macroblock presented in the figure 5.8 (a) has an artifact. For this macroblock, the PSNR value is less than 35 dB and the SSIM value is greater than 0.9. The PSNR considers it to be of inferior quality while the SSIM considers it to be of superior quality. Thus for this

(a) With artifact

(b) With artifact

(c) With artifact

(d) Without artifact

Figure 5.8: Screenshots of a frame of the video sequence vidyo1_1280×720_60 showing macroblocks under consideration (red block)

case, the judgment of SSIM is erroneous.

It can be seen that the PSNR correctly judges the presence/absence of artifacts in the figures 5.8 (a), 5.8 (c) and 5.8 (d) and the SSIM in figure 5.8 (c). However, the judgment of PSNR for the figure 5.8 (b) and the judgment of SSIM for the figures 5.8 (a), 5.8 (b) and 5.8 (d) is erroneous. This proves that these metrics are not fully efficient in locating artifacts at the macroblock level. Thus for this thesis, the subjective methods are found to be more trustworthy for threshold calculation and for VQA.

Chapter 6

Results

Experiments were carried out on the High Efficiency Video Coding (HEVC)(41) sequence set provided by the standard body, containing 23 sequences of varied nature provided in the YUV 4:2:0, 8 bit format. The results for the widely used bitrate range of upto 20 Mbps are presented in this section. Figure 6.1 shows the snapshot of a frame of the video sequence vidyo1_1280×720_60 encoded using the proposed algorithm. The yellowish-green boxes refer to the macroblocks encoded using the Intra4×4 prediction mode, the green boxes refer to the macroblocks encoded using the Intra16×16 mode and the remaining macroblocks are encoded using the P-Skip mode. It is seen that the static part (background) is correctly encoded using the P-Skip mode and the moving objects such as the people are encoded using the intra prediction modes.

6.1 BD-Bitrate vs BD-PSNR results

Table 6.1 shows BD-Bitrate vs BD-PSNR results for the HEVC sequences. The results are calculated using the Bjontegaard model for calculating the average differences between two RD plots(35). The model finds the average difference by fitting both curves using a minimum of four data points each, by calculating expressions for the integrals of the curves and then by dividing the difference between the integrals by the integration interval. The performance of the reference algorithm and the proposed algorithm is thereby compared.

Overall, intra prediction along with the P-Skip mode in the JM reference encoder provides a bitrate gain of 13.2442% and a PSNR gain of 0.9507 dB as compared to the intra-only profile of the JM reference encoder whereas the proposed algorithm provides a bitrate gain of 12.9074% and a PSNR gain of 1.1437 dB (refer to table 6.1) as

Figure 6.1: The snapshot of a frame of video sequence vidyo1_1280×720_60 encoded using the proposed algorithm

Table 6.1: Comparison of the JM reference encoder and the proposed algorithm

	JM reference (intra-only) vs JM reference (intra + P-Skip)		**JM reference (intra-only) vs proposed algorithm**	
	BD-Rate (%)	**BD-PSNR (dB)**	**BD-Rate (%)**	**BD-PSNR (dB)**
Luma	-10.1159	0.7945	-9.9071	0.6022
Cb	-15.2199	1.1842	-15.0811	2.3475
Cr	-16.7773	1.3274	-16.2492	2.4109
Overall	-13.2442	0.9507	-12.9074	1.1437

compared to the intra-only profile of the JM reference encoder with significantly less memory. Thus, the proposed algorithm improves the PSNR by 0.193 dB (1.1437 dB - 0.9507 dB) at the cost of 0.3368% increase in the bitrate. Note that the overall result is not the average of luma, Cb and Cr results. These are calculated separately by the Bjontegaard model and cannot be simply averaged. The luma, Cb and Cr results are calculated by taking only luma, Cb and Cr pixels respectively into consideration. The overall results are calculated by taking all luma, Cb and Cr pixels into consideration.

6.2 Memory Requirements

Table 6.2 shows a comparison of the memory required in the existing implementation and the proposed implementation of P-Skip for a video sequence of a resolution 1920×1088 pixels. The existing algorithm uses 3072 bits for saving the macroblock data (8 bits per pixel × (256 luma pixels + 64 Cb pixels + 64 Cr pixels per macroblock)). The proposed algorithm using the sample WM, utilizes only 13 bits per partition (refer to equation 5.4) with a total of 6 partitions: 4 for luma and 1 for Cb and Cr each. In totality, 78 bits are required to save the macroblock data in the memory. Thus, the proposed algorithm reduces the memory consumption by 97.46% $\left(\frac{(3133.44-79.56)\ Kbytes}{3133.44\ Kbytes} \times\ 100\%\right)$. The memory required per frame for the proposed algorithm is 79.56 Kbytes (memory in bytes per macroblock × number of macroblocks) as compared to 3133.44 Kbytes for the existing implementation. Also, for a frame rate of 120 bytes/sec, the memory bandwidth gets reduced from 375.96 Mbytes/sec to 9.54 Mbytes/sec.

Table 6.2: Memory requirements

	Existing implementation	**Proposed implementation**
Memory/MB	3072 bits	78 bits
Memory/Frame	3133.44 Kbytes	79.56 Kbytes
Memory Bandwidth	375.96 Mbytes/sec	9.54 Mbytes/sec

The results for the individual sequences are listed in table 6.3. For the artificial sequences of the HEVC sequence set like the SlideShow and the SlideEditing (videos containing slideshows) and for some other sequences like the Jamesboat, the proposed algorithm provides a higher bitrate gain with significantly less memory consumption than the existing JM encoder with P-Skip. For the sequence SlideShow, a bitrate gain of 59.6108% is achieved by the proposed algorithm as compared to a bitrate gain of 36.3132% by the existing implementation with reduced memory consumption. Thus, for the applications such as surveillance cameras, transmission of synthetic content over the Internet and WiGig applications, the proposed algorithm can be deployed efficiently.

For the sequences of the set like Four_people, KristenAndSara and Kimono the proposed algorithm provides lesser bitrate gain than the existing JM reference encoder with P-Skip but uses significantly less memory in the process.

The figures 6.2 and 6.3 show the Rate Distortion (RD) curves for various QP configurations for two of the sequences of the table 6.3 namely SlideShow and ChinaSpeed respectively. The graphs depict that for the same bitrate, the proposed algorithm

Table 6.3: BD-Bitrate vs BD-PSNR for all sequences

Sequence name	**JM reference (intra-only) vs JM reference (intra + P-Skip)**		**JM reference (intra-only) vs proposed algorithm**	
	BD-Rate (%)	**BD-PSNR (dB)**	**BD-Rate (%)**	**BD-PSNR (dB)**
BQ_terrace	-26.0531	1.4589	-24.631	1.4163
Basketball_drill	1.5688	-0.1122	2.4421	-0.1451
Basketball_Drive	-37.0581	1.7337	-35.5049	1.7102
Blowing_Bubbles	5.6397	-0.3602	5.7759	-0.3601
Cactus	-2.9180	0.0497	6.9033	-0.2782
ChinaSpeed	-15.8074	1.2505	-22.1794	2.0703
Four_people	-24.4918	0.9130	-14.4581	1.0130
Keiba	10.6284	-0.5392	10.5356	-0.5388
Kimono	-43.7008	2.5163	-41.5841	2.4786
KristenAndSara	-24.5057	0.8627	-13.5325	-8.6561
NFL_1080i60_TS_cut2	1.7435	-0.1084	0.8666	-0.0541
Peopleonstreet	6.1158	-0.3704	9.9972	-0.6141
RaceHorses	8.3077	-0.4694	8.2827	-0.4682
SlideEditing	-65.4204	8.8502	-90.7932	8.4365
SlideShow	-36.3132	3.5172	-59.6108	8.5346
Tennis	-50.974	2.8515	-50.2496	2.8463
TropicalFish	6.7275	-0.4910	6.7777	-0.4936
Jamesboat	7.0916	-0.3841	3.0703	-0.1821
Terminator	11.1832	-0.6258	12.9591	-0.7444
Tr_flying	9.3239	-0.5434	10.1081	0.5958
Traffic	-0.1562	-0.0381	6.7432	0.3771
Vidyo1	-22.664	1.0126	-13.8281	0.8684
Vidyo4	-22.9737	0.9452	-15.3855	0.9428

provides a better quality output.

In summary over all the sequences, the proposed algorithm provides 97.46% savings in memory consumption at a better quality (0.193 dB increase in PSNR) at the cost of 0.3368% increase in bitrate.

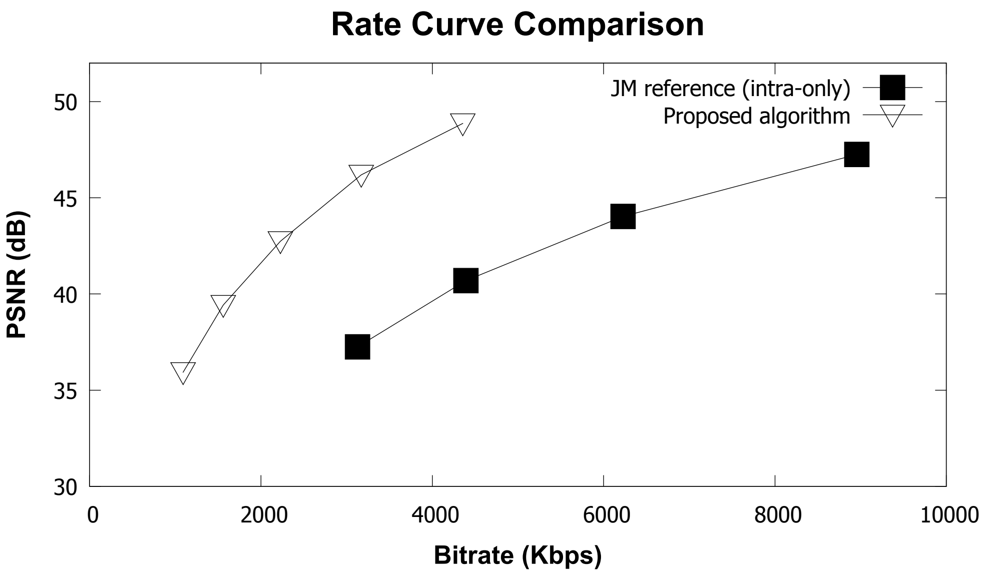


Figure 6.2: RD curves for the sequence SlideShow

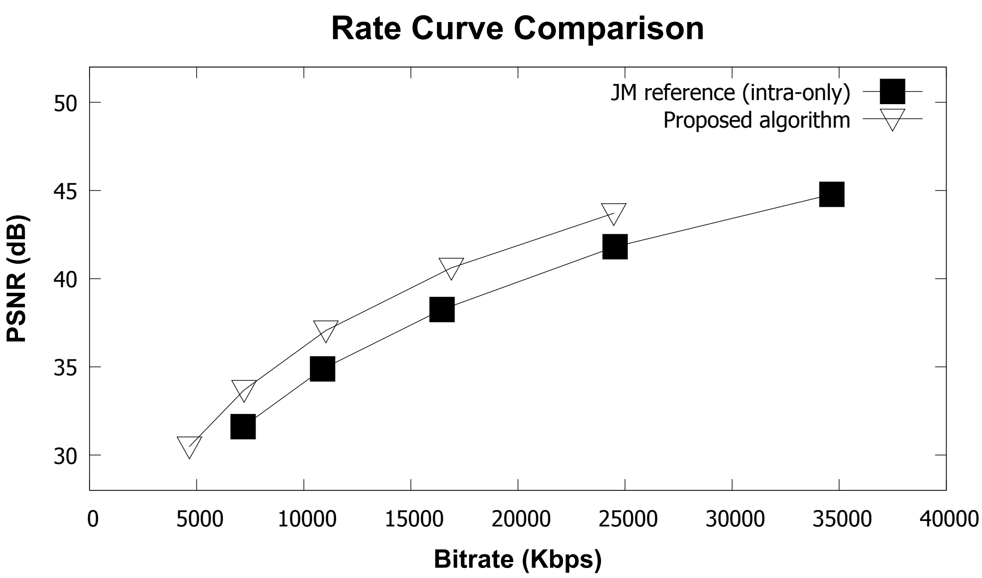


Figure 6.3: RD curves for the sequence ChinaSpeed

Chapter 7

Conclusion

The thesis proposes a novel, low memory P-Skip determination algorithm for video encoders. P-Skip estimation is performed by saving an SM per macroblock. The P-Skip mode decision is taken based on the correlation between the SMs of the co-located macroblocks in the consecutive frames. The standard intra prediction process is followed for the non-skipped macroblocks. A sample WM based SM is presented in the thesis. An altogether different kind of SM can also be chosen. The proposed algorithm (using the sample SM) provides a significant memory reduction of 97.46% at a better quality (0.193 dB increase in PSNR) at the cost of 0.3368% increase in bitrate over the tested sequences. These advantages make this work competent to be used in applications with low memory requirements like surveillance cameras, low cost video calls and WiGig applications. Also, the significance of the objective VQA metrics in locating the artifacts at the macroblock level is gauged in this thesis. It is concluded that the objective VQA metrics like the PSNR and the SSIM fail to effectively locate the macroblocks with artifacts. Thus for the thesis, the subjective VQA metrics are preferred over the objective VQA metrics for the assessment of the quality of a video sequence.

References

[1] I.-T. R. H.264, *Advanced video coding for generic audio-visual services*, Std., November 2007, [Online] Available https://www.itu.int/rec/T-REC-H.264.

[2] *WiGig White Paper: Defining the Future of Multi-Gigabit Wireless Communications*, July 2009, [Online] Available: http://wirelessgigabitalliance.org/specifications/.

[3] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 560–576, July 2003.

[4] I. Richardson, *White Paper: An Overview of H.264 Advanced Video Coding*, VCodex, 2007, [Online] Available www.vcodex.com/h264.html.

[5] J. L. Gonzalez-de Suso, A. Jimenez-Moreno, E. Martinez-Enriquez, D. de Maria, and Fernando, "Improved Method to select the Lagrange Multiplier for Rate-Distortion Based Motion Estimation in Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 452–464, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1109/TCSVT.2013.2276857

[6] J.-W. Chen, C.-Y. Kao, and Y.-L. Lin, "Introduction to H.264 Advanced Video Coding," in *Design Automation, 2006. Asia and South Pacific Conference on*, January 2006, pp. 736–741.

[7] *JM reference software manual (Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6))*, [Online] Available http://www.scribd.com/doc/48732371/JM-Reference-Software-Manual-JVT-AE010.

[8] S. Kim, H. Kim, T. Chung, and J.-G. Kim, "Design of H.264 video encoder with C to RTL design tool," in *SoC Design Conference (ISOCC), 2012 International*, Nov 2012, pp. 171–174.

[9] C. Grecos and M. Y. Yang, "Fast inter mode prediction for P slices in the H.264 video coding standard," *Broadcasting, IEEE Transactions on*, vol. 51, no. 2, pp. 256–263, June 2005.

[10] Q. Liu and T. Ikenaga, "Rate-distortion optimization based skip mode early detection in H.264," in *Intelligent Signal Processing and Communication Systems (ISPACS), 2010 International Symposium on*, Dec 2010, pp. 1–4.

[11] T. Wiegand and G. Sullivan, "Rate distortion Optimization for video compression," *IEEE Signal Processing Magazine*, vol. 15, pp. 74–90, November 1998.

[12] J. L. Inchoon Choi and B. Jeon, "Efficient coding mode decision in MPEG-4 part-10 AVC/H.264 main profile," in *Image Processing, 2004. ICIP '04. 2004 International Conference on*, vol. 2, Oct 2004, pp. 1141–1144 Vol.2.

[13] B.-Y. C. Shih-Hsuan Yang and K.-H. Wang, "H.264 Fast Inter-Mode Selection Based on Coded Block Patterns," in *Institute of Electronics, Information and Communication Engineers*, 2008.

[14] Y. Ding, Y. Si, and C. Yao, "A Novel Inter Mode Decision Algorithm for H.264/AVC," in *Image and Signal Processing, 2008. CISP '08. Congress on*, vol. 1, May 2008, pp. 334–338.

[15] B.-G. Kim and C.-S. Cho, "Fast Inter-Mode Decision Algorithm for P Slices in H.264/AVC Video Standard," in *Consumer Electronics, 2007. ISCE 2007. IEEE International Symposium on*, June 2007, pp. 1–6.

[16] Y. Ivanov and C. Bleakley, "Skip Prediction and Early Termination for Fast Mode Decision in H.264/AVC," in *Digital Telecommunications, , 2006. ICDT '06. International Conference on*, Aug 2006.

[17] Y. L. Zhijiao Guo, Nan Sang and X. Wen, "An improved P-Skip mode decision algorithm for H.264/AVC based on observers' low sensitivity to boundary parts of video images," in *Broadband Network and Multimedia Technology (IC-BNMT), 2011 4th IEEE International Conference on*, Oct 2011, pp. 227–231.

[18] Z. Wei and K. N. Ngan, "A Fast Macroblock Mode Decision Algorithm for H.264," in *Circuits and Systems, 2006. APCCAS 2006. IEEE Asia Pacific Conference on*, Dec 2006, pp. 772–775.

[19] G. Povala, D. Silveira, L. Amaral, B. Zatt, M. Porto, and L. Agostini, "An efficient reference frame compression approach for video coding systems," in *Circuits and Systems (LASCAS), 2014 IEEE 5th Latin American Symposium on*, Feb 2014, pp. 1–4.

[20] X. Bao, D. Zhou, and S. Goto, "A lossless frame recompression scheme for reducing DRAM power in video encoding," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, May 2010, pp. 677–680.

[21] D. Silveira, G. Sanchez, M. Grellert, V. Possani, and L. Agostini, "Memory bandwidth reduction in video coding systems through context adaptive lossless reference frame compression," in *Programmable Logic (SPL), 2012 VIII Southern Conference on*, March 2012, pp. 1–6.

[22] D. Silveira, M. Porto, and L. Agostini, "A lossless approach for external memory bandwidth reduction in video coding systems and its VLSI architecture," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, July 2013, pp. 1–6.

[23] M. Budagavi and M. Zhou, "Video coding using compressed reference frames," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, March 2008, pp. 1165–1168.

REFERENCES

[24] Y. Ivanov and D. Moloney, "Reference Frame Compression Using Embedded Reconstruction Patterns for H.264/AVC Decoder," in *Digital Telecommunications, 2008. ICDT '08. The Third International Conference on*, June 2008, pp. 168–173.

[25] A. Gupte, B. Amrutur, M. Mehendale, A. Rao, and M. Budagavi, "Memory Bandwidth and Power Reduction Using Lossy Reference Frame Compression in Video Encoding," *Circuits and Systems for Video Technology*, vol. 21, no. 2, pp. 225–230, Feb 2011.

[26] S. C. Irena Koprinska, "Temporal video segmentation: A survey," in *Signal Processing: Image Communication 16*, 2001, pp. 477–500.

[27] S. K. T. Kikukawa, "Development of an automatic summary editing system for the audio visual resources," in *Institute of Electronics, Information and Communication Engineers*, 1992, pp. 204–212.

[28] B. Shahraray, "Scene change detection and content-based sampling of video sequences," in *Proceedings of IS&T/SPIE*, vol. 2419, 1995, pp. 2–13.

[29] J. L. W. Xiong and M. Ip, "Net comparison: a fast and e!ective method for classifying image sequences," in *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases III,San Jose, CA,*, vol. 2420, 1995, pp. 318–328.

[30] W. Xiong and J. Lee, "Efficient scene change detection and camera motion annotation for video classification," in *Comput. Vision Image Understanding*, 1998, pp. 166–181.

[31] R. Kasturi and R. Jain, *Computer Vision: Principles*. IEEE Computer Society Press, Washington DC, 1991, ch. Dynamic vision, pp. 469–480.

[32] N. V. Patel and I. K. Sethi, "Video Shot Detection and Characterization for Video Databases," *Pattern Recognition (Journal)*, vol. 30, pp. 583–592, 1997.

[33] Y. Tonomura, "Video Handling Based on Structured Information for Hypermedia Systems," in *International Conference on Multimedia Information Systems '91*. New York, NY, USA: McGraw-Hill, Inc., 1991, pp. 333–344. [Online]. Available: http://dl.acm.org/citation.cfm?id=128067.128096

[34] K. A. S. Zhang, HongJiang and S. W., "Automatic Partitioning of Full-motion Video," *Multimedia Systems (Journal)*, vol. 1, no. 1, pp. 10–28, Jan. 1993. [Online]. Available: http://dx.doi.org/10.1007/BF01210504

[35] G. Bjontegaard, *Calculation of average PSNR differences between RD-curves*, ITU - Telecommunications Standardization Sector, [Online.

[36] I. E. Richardson, *The H.264 Advanced Video Compression Standard*, 2nd ed. Wiley, 2010.

[37] K. Seshadrinathan, R. Soundararajan, A. Bovik, and L. Cormack, "Study of Subjective and Objective Quality Assessment of Video," *Image Processing, IEEE Transactions on*, vol. 19, no. 6, pp. 1427–1441, June 2010.

[38] K. Kipli, S. Krishnan, N. Zamhari, M. Muhammad, S. Masra, K. L. Chin, and K. Lias, "Full reference image quality metrics and their performance," in *Signal Processing and its Applications (CSPA), 2011 IEEE 7th International Colloquium on*, March 2011, pp. 33–38.

[39] A. Hore and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, Aug 2010, pp. 2366–2369.

[40] F. Shanableh, T.and Ishitaq, "Macroblock level quality assessment using video-independent classification," in *9th International Symposium on Mechatronics and its Applications (ISMA)*, April 2013, pp. 1–5.

[41] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.