Thesis for the *Doctor of Philosophy*

Motion Estimation Skipping Strategies for Low-complexity HEVC Encoding

Sang-hyo Park

Graduate School of Hanyang University

August 2017

Thesis for the *Doctor of Philosophy*

Motion Estimation Skipping Strategies for Low-complexity HEVC Encoding

Thesis Supervisor: Euee S. Jang

A Thesis submitted to the graduate school of Hanyang University in partial fulfillment of the requirements for the degree of *Doctor of Philosophy*

Sang-hyo Park

August 2017

Department of Computer and Software
Graduate School of Hanyang University

This thesis, written by Sang-hyo Park,
has been approved as a thesis for the *Doctor of Philosophy.*

August 2017

Committee Chairman:	Prof. Jong-Il Park	(Signature)
Committee member:	Prof. Taesoo Kwon	(Signature)
Committee member:	Prof. Ronggang Wang	(Signature) Ronggang Wang
Committee member:	Dr. Kiho Choi	(Signature)
Committee member:	Prof. Euee S. Jang	(Signature)

Graduate School of Hanyang University

CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

AVC	Advanced Video Coding
AMP	Asymmetric Motion Partition
BD-rate	Bjøntegaard-Delta bitrate
BHME	Bidirectional Half-pixel Motion Estimation
BIME	Bidirectional Integer-pixel Motion Estimation
BME	Bidirectional Motion Estimation
BMC	Bidirectional Motion Compensation
BSME	Bidirectional Sub-pixel Motion Estimation
BQME	Bidirectional Quarter-pixel Motion Estimation
CTC	Common Test Condition
CU	Coding Unit
DCT	Discrete Cosine Transform
FPS	Frame per second
FVC	Future Video Coding
GBP	Generalized B Picture
HbMV	Half-pixel best Motion Vector
HD	High Definition
HEVC	High Efficiency Video Coding
HM	HEVC Test Model
HME	Half-pixel Motion Estimation
HMV	Half-pixel Motion Vector
IbMV	Integer-pixel best Motion Vector
IME	Integer-pixel Motion Estimation
IMV	Integer-pixel Motion Vector
JCT-VC	Joint Collaborative Team on Video Coding
JVET	Joint Video Exploration Team

LD	Low-delay
LDSP	Large Diamond Search Pattern
MC	Motion Compensation
ME	Motion Estimation
MPEG	Moving Picture Experts Group
MV	Motion Vector
MVP	Motion Vector Prediction
PSNR	Peak Signal-to-Noise Ratio
PU	Prediction Unit
QbMV	Quarter-pixel best Motion Vector
QME	Quarter-pixel Motion Estimation
QMV	Quarter-pixel Motion Vector
QP	Quantization Parameter
RA	Random Access
RAM	Random Access Memory
RD	Rate-Distortion
SAD	Sum of Absolute Difference
SATD	Sum of Absolute Transformed Difference
SDSP	Small Diamond Search Pattern
SME	Sub-pixel Motion Estimation
TU	Transform Unit
UIME	Unidirectional Integer-pixel Motion Estimation
UME	Unidirectional Motion Estimation
USME	Unidirectional Sub-pixel Motion Estimation
VSM	Very Simple Motion

ABSTRACT

Motion Estimation Skipping Strategies for Low-Complexity HEVC Encoding

Sang-hyo Park

Dept. of Computer and Software

The Graduate School

Hanyang University

In video coding, motion estimation (ME) that predicts a block among temporally correlated frames has had a crucial impact on not only the compression efficiency, but also the computational complexity. Particularly, fast ME algorithms has been a pivot in much research that attempts to reduce the complexity of video encoder while preserving the compression efficiency as much as possible. However, the overall complexity of ME including time and memory complexity was increased much further in the high efficiency video coding (HEVC) codec to meet twice better coding efficiency than AVC/H.264 codec due to the diversified changes: more various motion partitions, more various motion directions, and more elaborated motion accuracy. Since encoder has to encode huge pixels due to the increased video resolution such as 4K ultra-high-definition (UHD) and 8K UHD, the complexity of ME should be overcome for low-complexity HEVC encoding.

To relieve the complexity of HEVC, two ME skipping strategies are proposed in this dissertation by considering those various changes of HEVC. One is a bidirectional motion

estimation (BME) skipping strategy that avoids most compression-inefficient search points for bidirectional motion that has high computational complexity due to additional interpolation process in contrast with unidirectional motion estimation (UME). The proposed method exploits the stochastic correlation of the context of prediction units (PUs)—one of the special characteristic of HEVC—to determine whether BME is critical or not. The other is a skipping strategy of sub-pixel motion estimation (SME) that avoids inefficient search points of half-pixel and of quarter-pixel motions and skips associated interpolation process. Utilizing the correlation of the context of PUs, the second proposed method determines whether SME is critical or not during the encoding of UME and BME. The proposed method additionally checks SME for far reference frames from the current frames to decrease the complexity of SME further. When utilizing two proposed strategies, low-complexity HEVC encoding could be achieved, sustaining reasonable compression performance.

To demonstrate the performance of motion estimation skipping strategies, the following complexity measurement methods were carefully chosen: the number of related function calls, running time, and memory access count. Through experimental results, the proposed BME skipping strategy showed that the time complexity of bi-prediction can be reduced to 36% on average while losing a 0.5% increase of BD-rate, outperforming existing methods in view of encoding time, number of function calls, and memory access. In addition, the proposed SME skipping strategy showed that the time complexity of ME can be reduced to 51% (random access case) and 52% (low-delay case) on average, outperforming existing method. Therefore, the proposed two ME skipping strategies should be considered in low-complexity HEVC encoder for real-time broadcasting systems, surveillance applications, and mobile video encoding devices of battery-constraint.

TO THE GRACE OF GOD,

TO MY LOVELY WIFE,

TO MY PARENTS.

CHAPTER 1
INTRODUCTION

In real-time broadcasting systems and high-resolution video surveillances, encoder complexity should be overcome for the implementation of a fast encoder with reasonable video quality at low bitrates. In addition, many mobile devices face a similar problem during video encoding for video conferencing and user generated content (including live streaming), which is one of battery-constraint. Since the resolution of video on demand became much higher, it is necessary to optimize such encoder that suffers computational complexity or battery constraint. For example, at the time of a standardization project of AVC/H.264, a 352 x 288 resolution was widely used as one of common video test sequence such as *forman*. It seemed reasonable to set 16 x 16 block as a basic coding unit of AVC/H.264 as shown in a red-lined block of Figure 1.1, which is called macroblock. An overview paper of AVC/H.264 standard [1] published in 2003 also used the video sequence to describe the characteristics of AVC/H.264. On the contrary, HD 1080p sequence had been one of common test sequence and of main target for HEVC standardization project. It is clear that such HD sequence has much more pixels to be compressed than *forman* sequence as shown in Figure 1.2. Note that the red-lined 16 x 16 block seems very small to cover *Kimono* sequence in Figure 1.2 in comparison with the sequence in Figure 1.1.

Figure 1.1—A common video test sequence, *foreman* (352 x 288), and the size of basic unit (macroblock, 16 x 16) for the standardization project of AVC/H.264.

Figure 1.2—A common video test sequence, *Kimono* (1920 x 1080), for the standardization project of HEVC.

The complexity of encoder was increased even further in the latest video coding standard, high efficiency video coding (HEVC) since many complexity-intensive techniques were integrated into HEVC encoder to satisfy twice better compression performance than AVC/H.264 [2]. To be specific, HEVC adopted new but complex coding tools and increased the cases to be investigated in comparison to the predecessor as particularly shown in the following bullets (refer to [3] for the detailed description):

- The maximum coding unit (CU) size is increased with quadtree coding structure: a 64 x 64 block pixels can be encoded by one 64 x 64 CU or various small CUs (8 x 8 is the minimum CU size) whereas 16 x 16 was the maximum size of AVC/H.264.
- Each predicted block can be encoded by various transform unit (TU) size.
- The number of intra prediction directions to be tested is increased roughly four times at maximum: from 9 (AVC/H.264) to 35 (for the certain CU size).
- The number of partition for inter prediction is increased: asymmetric motion partitions are newly added.
- New techniques such as sample adaptive offset are added.

Thus, it is vital to reduce the encoder complexity caused by HEVC, while preserving the enhanced compression performance as much as possible.

The motion estimation (ME) process including the associated motion compensation process is critical, which comprises most of the encoding complexity [2], [4]. HEVC encoder could utilize parallel ME block by block in a frame if possible to effectively speedup encoding time. However, it is hard to utilize parallel ME in a frame without losing coding efficiency because ME process utilizes spatial neighbor blocks and temporally co-located block to save the bit cost of motion vector (MV). Accordingly, many researches have attempted to develop a fast ME algorithm that reduces search points on certain reference frames for unidirectional ME [5]-[10]. Among many fast ME methods, a popular method is developing a low-complexity search pattern. Reducing the points effectively, the diamond search method [8] is adopted as a basis in the ME process of HEVC. Recently, different search methods were studied, using the rate-distortion cost for finding an adequate direction [9] or using a confidence interval with a Coding Unit (CU) structure [10]. The aforementioned methods decreased the search points of unidirectional ME (UME) significantly, but further reduction of the ME process is necessary, especially for the tool that has recently received much attention in HEVC: 1) bidirectional ME (BME) and 2) sub-pixel ME (SME).

The complexity of those UME, BME, and SME can be reduced by skipping some mode decision processes if special conditions are satisfied. Rhee *et al*. summarized existing fast mode decision methods for AVC/H.264 and applied some of the available ones to HEVC [11]. In addition, researchers investigated early prediction unit (PU) decision [12]-[13] or CU decision methods [14]-[16] further decrease encoding time, as ME belongs to various Prediction Units (PUs), which belong to various CUs. However, when the conditions of the aforementioned methods were not satisfied, the encoder cannot avoid the burden of ME. Especially, to perform BME and SME, it is unavoidable for an encoder without complexity-intensive interpolation filtering, which should be relieved.

In this dissertation, an efficient bi-prediction decision strategy is firstly presented, which avoids most of the bi-prediction process. The proposed method exploits correlated prediction unit (PU) information (MV accuracy and PU partition) and determines whether or not the current block is static, assuming that bi-prediction will be used in a motion-intensive or complex texture area. The experimental results demonstrated that the proposed technology reduced encoding complexity significantly compared to HEVC, with negligible coding loss and outperformed existing methods. Of the proposed method, original work with preliminary results were published in SPIE [17].

Secondly, an efficient SME skipping strategy is presented in this dissertation, which contributes on lightening the entire complexity of ME. The proposed SME skipping strategy exploits correlated PU information similar with that of the proposed BME skipping strategy to determine whether or not the current block is static. Additionally, quarter-pixel ME (a part of SME) and associated interpolation filtering can be skipped according to the ME result in the first reference frame, assuming that sub-pixel MVs will be used in a motion-intensive or complex texture area. In similar with the results of the first proposed strategy, the experimental results demonstrated that the proposed technology reduced ME encoding complexity significantly compared to HEVC, with negligible coding loss and outperformed existing method.

The dissertation is organized as follows. Section 2 describes the overview of the entire ME process in HEVC and reviews related works. Section 3 analyzes the complexity of ME and identifies the redundant BME and SME processes. The stochastic approaches that are utilized to reduce the redundancy of both BME and SME are presented in Section 4, and the proposed efficient ME skipping strategies are described in Section 5. In the section, the coding condition and experimental results are shown as well. Finally, Section 6 concludes this dissertation.

CHAPTER 2
MOTION ESTIMATION IN HEVC

This section describes the motion estimation process of HEVC in general coding configuration and classifies related works. Firstly, unidirectional ME specified to HEVC is described. The UME includes integer-pixel ME (IME) as well as SME that contains interpolation filtering process. Secondly, BME is also described, which expended the basic concept from that of conventional video codecs. Finally, related works that attempted to reduce the complexity of the associated ME process were classified and reviewed.

2.1. Overview of Motion Estimation in HEVC

In the HEVC reference encoder, a CU can be compressed by various PU partitions as shown in Figure 2.1, and each PU partition can be predicted by several prediction methods: 1) uni-prediction, including SME; 2) bi-prediction, including SME; and 3) merge prediction (not for 2N x 2N PU partition). Firstly, ME process for uni-prediction can be summarized in two parts: finding the best starting point and searching cost-efficient point using a certain search pattern. Since the performance of search pattern can be varied by the starting point, HEVC constructs initial candidates of starting point by deriving MVs of spatially or temporally correlated PUs (i.e., motion vector prediction, MVP) and decides a cost-efficient candidate among them [18]. After the best integer-pixel MV is searched, SME will be performed from the best IMV position.

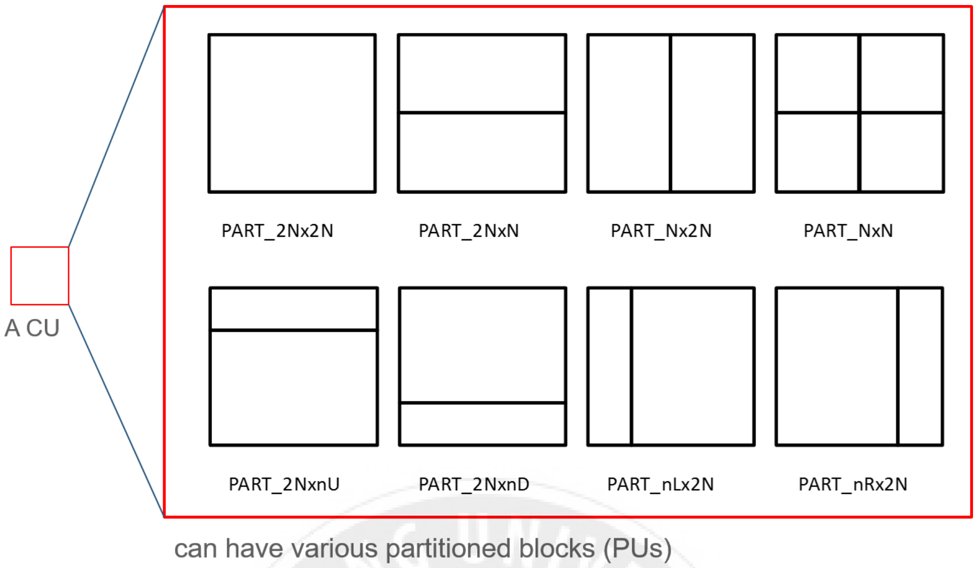


Figure 2.1—Various PU partitions for a CU in HEVC.

Secondly, exploiting those results of uni-prediction—the MVP of each direction and the best MV of each direction—bi-prediction that combines two blocks to predict the current block is performed. Since HEVC has a special quadtree-based coding structure, MVP can utilize following candidates: spatial neighbor PUs (left, below-left, above, above-right, and above-left PUs), temporally co-located PUs, and zero MV if available. Finally, merge prediction is a prediction technique that predicts a block by constructing candidates from spatially or temporally correlated PUs.

Particularly, the bi-prediction method predicts motion based on the results of two uni-prediction MVs: one is designated as L0 (reference picture list 0), and the other as L1 (reference picture list 1). Reference picture lists can contain different indices of reference frames. However, since the two lists have the same indices with the same order in the GBP case, the ME process for uni-prediction can be avoided for the prediction of L1, which is common in the recent HEVC reference encoder. In the GBP case, MV for L1 is simply obtained from the MVP value.

An overview of the HEVC encoding process for CU encoding in the GBP case is shown in Figure 2.2. Note that, as shown in Figure 2.2, bi-prediction is performed as much as the number of performing uni-prediction in every PU partition. Although reducing the number of complexity-intensive bi-prediction (complexity issues are dealt with in Section 3) has not been thoroughly investigated, the issue of uni-prediction has been heavily addressed in the literature.

On the other hand, MV for L1 is obtained from the general ME process (i.e., IME and SME) for CU encoding in random access scenario. An overview of the HEVC encoding process for CU encoding in random access case is shown in Figure 2.3. Due to the additional ME processes compared to the GBP case in Figure 2.2, the associated encoding complexity should be reduced as well.

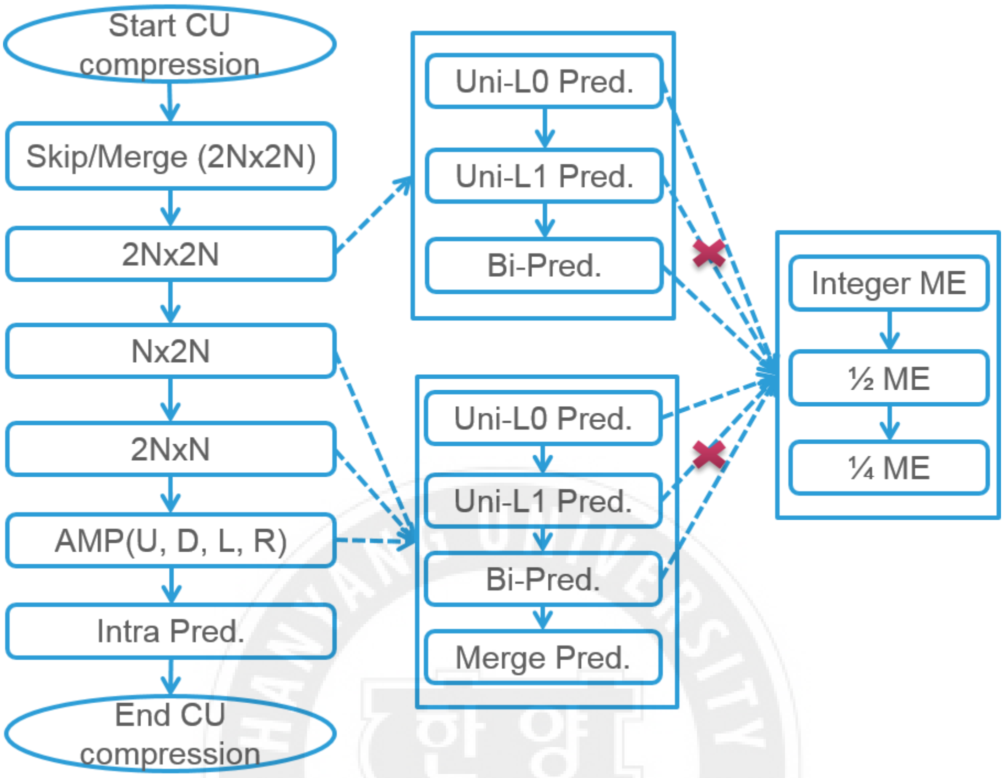


Figure 2.2—Overview of CU compression in the GBP case of HEVC. AMP is an asymmetric motion partition (up, down, left, and right).

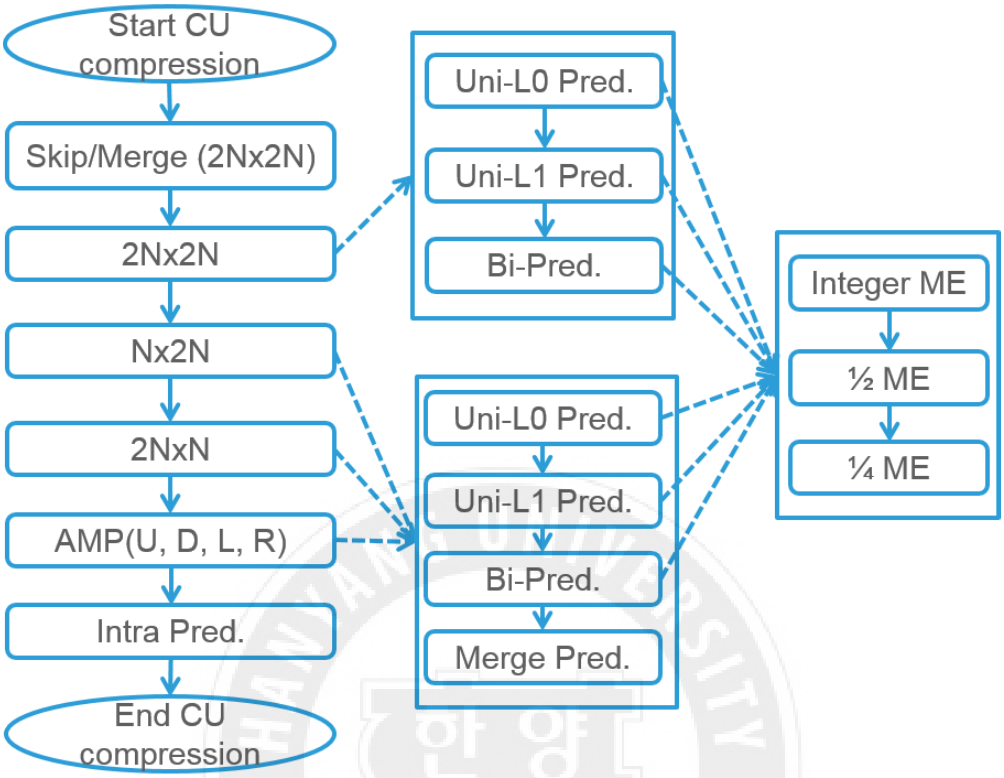


Figure 2.3—Overview of CU compression in the random access scenario of HEVC. AMP is an asymmetric motion partition (up, down, left, and right).

To decide which predicted block is the most efficient in terms of compression performance, HEVC generally takes several cost functions to measure picture distortion. For integer-pixel ME, sum of absolute difference (*SAD*) function is generally used as shown in (1):

$$SAD = \sum_{x,y} |A(x,y) - B(x,y)|, \tag{1}$$

where *A* denotes a two-dimensional matrix containing intensities of pixel elements for reference block, *B* denotes a two-dimensional matrix containing intensities of pixel elements for predicted block, and *x* and *y* denote the pixel positions in a block. For sub-pixel ME, either *SAD* in (1) or sum of absolute transformed difference (*SATD*) function can be used depending on Hadamard transform flag. The *SATD* function is shown in (2):

$$SATD = \{\sum_{x,y} H(|A(x,y) - B(x,y)|)\}/2, \tag{2}$$

where *H* represents a Hadamard transform matrix.

In addition, Lagrangian multiplier is used to minimize rate-distortion (RD) cost. Let the RD cost of predicted block be J, and the RD-cost J_{SAD} in the case that SAD is used for distortion cost is specified by the following equation:

$$J_{SAD} = SAD + \lambda \cdot R, \tag{3}$$

where R denotes a bitrate cost of a predicted block and λ is a Lagrangian multiplier weighted by QP offset and slice types. The RD-cost J_{SATD} in the case that $SATD$ is used for distortion cost is specified by the following equation:

$$J_{SATD} = SATD + \lambda \cdot R. \tag{4}$$

2.2. Bidirectional Motion Estimation in HEVC

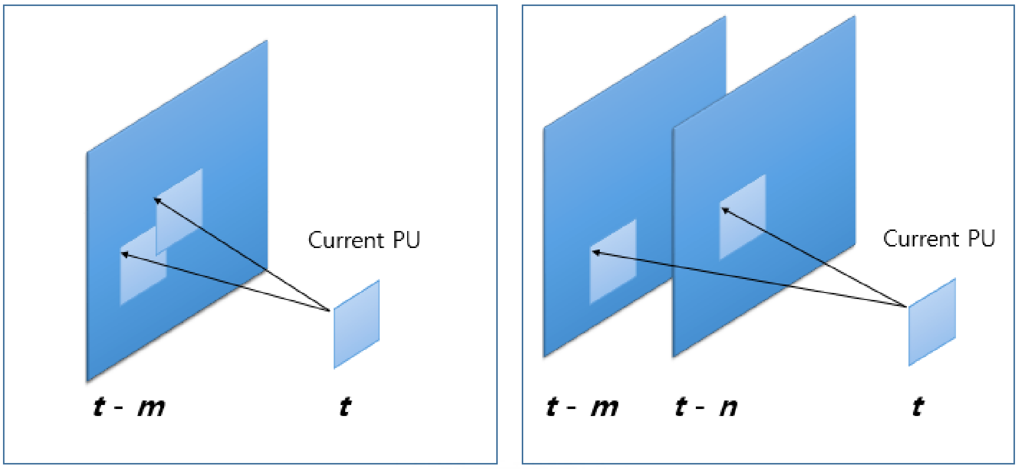
Although the concept of bi-prediction has existed for several decades, an additional definition of bi-prediction in the generalized B picture (GBP) is quite intriguing. Traditionally, the B picture predicts a block from two future and past frames in MPEG-2. On the other hand, GBP is more general in that it predicts a block not only from two temporally opposite frames, but also from temporally same directional frames (e.g., two previous frames) in AVC/H.264 [19]. In HEVC, GBP usually means referring only to previous frames so that this picture type can help low-delay decoding applications compared to a random-access scenario (i.e., IBBP structure).

There are two possible cases of bi-prediction in the GBP of HEVC. As shown in Figure 2.4, the current frame t can refer to (t-m)–th previous frame as one of reference frames or refer to both (t-m)–th and (t-n)–th previous frames. Investigating the combined block, the

HEVC encoder can compress the current block more efficiently due to the additional opportunity for ME. However, it is highly desirable to avoid substantial encoding complexity, such as operational complexity, as well as memory access time, which should be analyzed and reduced for a fast HEVC encoding application. The analysis of the associated encoding complexity will be described in Subsection 3.1 and Subsection 3.2.

2.3. Sub-pixel Motion Estimation in HEVC

SME that finds half-pixel and/or quarter-pixel MVs follows the IME process—finding the best MV in integer-pixel level (hereafter, IbMV). Accordingly, HEVC encoder commonly sets the starting position of SME to the previously obtained IbMV position and generates imaginary samples for half-pixels by DCT-based 8-tap filter and for quarter-pixels by DCT-based 7-tap filter. HEVC hierarchically searches the best sub-pixel MV of which rate-distortion cost is minimum as shown in Figure 2.5. Firstly, an encoder finds the best half-pixel MV (HbMV) by searching 8-points near IMV. Secondly, the encoder finds the best quarter-pixel MV (QbMV) by searching 8-points near the best HMV. Although HEVC encoder can avoid the significant encoding complexity compared to AVC/H.264 by searching hierarchically, it cannot avoid the substantial complexity of interpolation filtering for generating half-pixel and quarter-pixel samples. The complexity (operational complexity and memory access) will burden implementing fast HEVC encoder, which should be overcame. The analysis of the associated encoding complexity will be described in Subsection 3.1 and Subsection 3.3.



(a) (b)

Figure 2.4—Two possible scenarios for bi-prediction of GBP in HEVC. (a) is bi-prediction within one reference frame, and (b) is bi-prediction from two difference reference frames. In both cases, the current PU only refers previous frames (m and n are positive numbers).

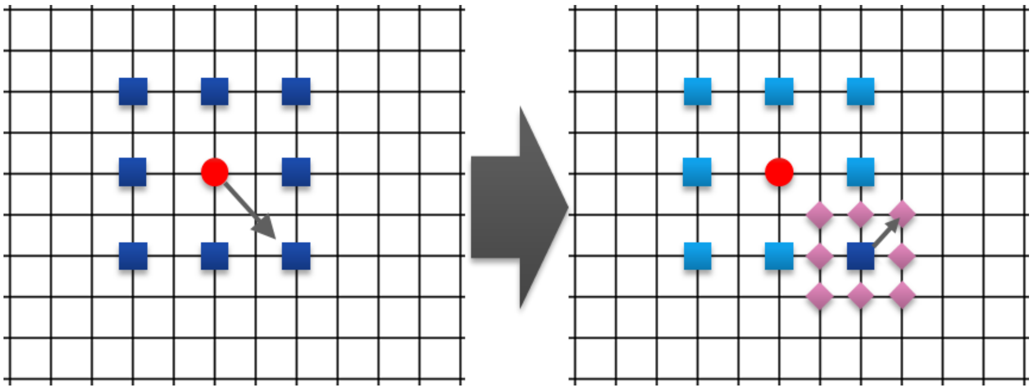


Figure 2.5— Hierarchical sub-pixel motion search. The stage of motion search goes through red circle (IMV position), blue square (half-pixel positions) and pink diamond (quarter-pixel positions). The basic unit of grid is quarter-pixel MV.

2.4. Related Work

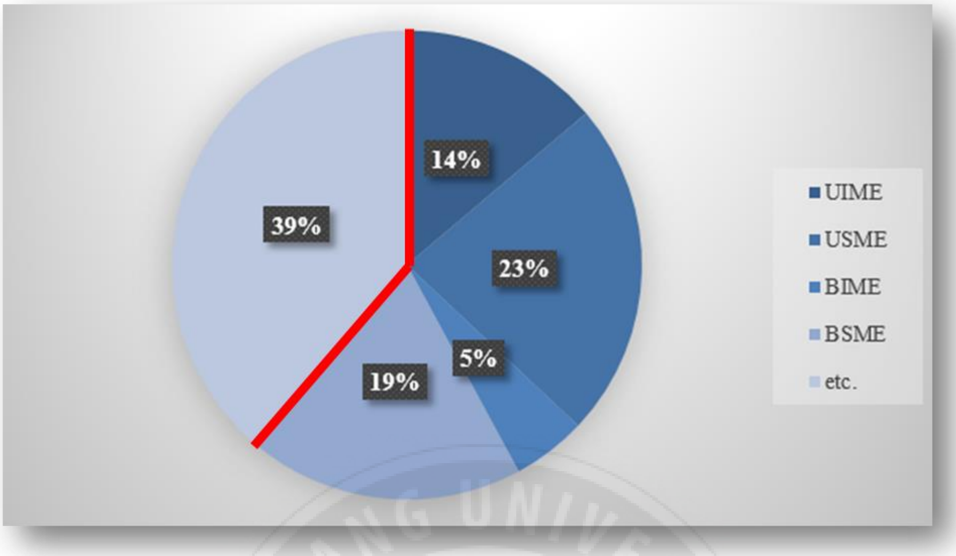
A few researches have attempted to solve the additional complexity occurring by bi-prediction [20]-[24]. To avoid some bi-prediction search, the distortion cost of other prediction modes (uni-prediction or merge mode) was used [20], [24]. Rhee and Lee [21] attempted to utilize the correlation between CUs to avoid the bi-prediction, applying their method to small CUs only. Another study exploited the motion vector (MV) and associated reference picture to avoid useless bi-prediction in the case that bi-prediction is the same as uni-prediction [22]. In the research of Du *et al*. [23], motion vector prediction (MVP) was also used to reduce the search range for integer-level bidirectional ME, which does not consider the computationally-intensive motion compensation processes as well as half-/quarter-pixel bidirectional ME. Whenever a B frame is used, these methods can reduce the complexity of bi-prediction adequately, albeit considerable room for feasible improvements exist, as they did not consider the special characteristics of bi-prediction, including prediction direction or motion complexity in a CU.

While much research has attempted to develop a fast integer-pel ME algorithms (e.g., diamond search [8], directional search [9], and statistical approach [10]), the topic of SME complexity in HEVC has not been heavily investigated despite its significant complexity problem. Particularly, the complexity problem of SME in HEVC was recently revealed by a few researchers. Blasi *et al.* [25] claimed that SME is the second most computationally expensive module in HEVC encoding by measuring execution time, showing that SME took roughly 34% of the total encoding time—which is significant when considering the fact that HEVC has lots of complex modules. Furthermore, *Lv et al*. [26] attempted to prove that the interpolation filters of HEVC has much more complexity that that of AVC/H.264 since HEVC equipped different filtering techniques and increased filter tap size. Assuming that the main complexity of SME was raised from the interpolating process of imaginary pixels for sub-pixel MVs, most of researchers targeted on reducing, reusing, or avoiding the interpolation for their fast SME methods [25], [27], [28], [29], [30]. The majority of fast SME algorithms attempted to determine the best sub-pixel MV by formulating an error-surface of integer-pixel MVs so that other searching time and the associated interpolation process for sub-pixel MVs can be hugely skipped [28], [29], [30]. Those methods used the RD cost (or just distortion cost) of integer-pel MVs for error-surface model, but resulted in noticeable coding loss (more than 3% of bitrate on equal PSNR on average). In addition, Blasi *et al.* [25] developed more compression-efficient but a little bit slow algorithm for fast SME using the RD costs and additional Sobel operator to detect edges in a frame. On the contrary, without losing coding efficiency, Choi *et al.* [27] reused interpolated pixels for sub-pixel MVs so that huge redundant operations can be optimized, sacrificing the memory that needs to store such interpolated pixels for repetitive reuse. As a result, there is still room for fast SME algorithm that does not harm coding efficiency as well as memory complexity, which should be resolved for low-complexity HEVC encoder.

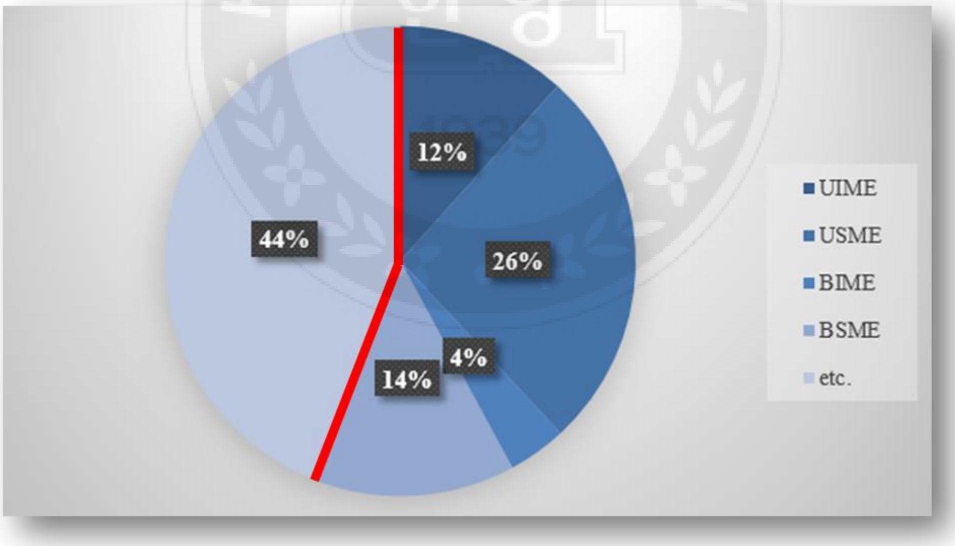
CHAPTER 3
COMPLEXITY ANALYSIS OF MOTION ESTIMATION IN HEVC

3.1. Time Complexity of Motion Estimation during HEVC encoding

To understand the time complexity impact of ME, I measured the time that each ME tool spend during entire HEVC encoding. In this measuring experiment, HM 15.0 was used with minor modifications that only attempted to measure time. Sequences chosen to be encoded in this experiment were four Class C (832 x 480) sequences and four Class D sequences, which were defined in common test condition (CTC) [31]. Both Low delay (LD) main and Random Access (RA) main configurations were used as recommended in CTC except QP: only QP 32 (mid-range in CTC) was employed to make the experiment simple. Time was measured in PC.



(a)



(b)

Figure 3.1—Time distribution of HEVC encoding: (a) is under LD case and (b) is under RA case

Figure 3.1 shows the time distribution of ME during HEVC encoding upon the aforementioned coding conditions. Note that, in the Figure 3.1, USME denotes not only time for ME, but also the associated interpolation filtering process for USME. Similarly, BIME and BSME include additional time that interpolates pixels. Accordingly, it is not surprising that total ME tools consume the most time of encoding as this feature can be similarly shown in other recent video codecs such as AVC/H.264. However, it is intriguing that, although many researchers focused on reducing the complexity of IME (UIME and BIME), SME (USME and BSME) is the highest part of ME complexity. In LD case, SME takes 42% in total encoding while IME takes 19%. In RA case, SME takes 40% while IME takes 16% only. In conclusion, SME that has twice more time complexity than IME should be highly focused on reducing its complexity for low-complexity HEVC encoder.

3.2. Complexity Analysis of Bidirectional Motion Estimation in HEVC

Similar to UIME, BIME also has to go through many searching points with different searching patterns. Diamond search [8] is a well-known method, which is adopted to HEVC as one of the searching pattern methods for UIME. On the other hand, BIME searches points in a squared area. Equation (5) briefly represents the computational complexity of bidirectional ME C_{BIME}, which shows how many points should be searched to compute distortion cost, as shown in:

$$C_{BIME} = (2 \times R)^2 \times N \ , \tag{5}$$

with given search range R (i.e., the maximum distance from the starting point for ME) and the number of PU N. In Big O notation, the complexity of (5) is $O(N \times R^2)$.

To understand the relative impact of UIME and BIME, the complexity of diamond-shaped UIME is also compared in this dissertation, since the diamond shape search range is a default setting for the first searching method in HEVC. For the diamond search method, the number of points to be searched depends on the distance from the starting point: 1) small diamond search pattern (SDSP) at one pixel distance; and 2) large diamond search pattern (LDSP) at longer distances. For SDSP, only four points (up, down, left, and right) are searched; whereas, for LDSP, eight points that compose a diamond shape (◇) by surrounding the starting point are searched [8]. Thus, the computational complexity of diamond UIME C_{DIME}, that represents how many points should be searched, can be regarded as shown in (6):

$$C_{DIME} = \{4 + \left(8 \times \log_2 \frac{R}{2}\right)\} \times N \ , \tag{6}$$

which can be represented as O(N × log R) in big-O notation with given N and R. Note that *R* was commonly bigger than 16, and 256 is often used in recent HMs and future video coding (FVC) project. In order to observe *R*'s problem by fixing *N* as a constant number, (5) and (6) can be simplified as $O(R^2)$ and $O(\log R)$, respectively. To reduce the squared complexity function, one straightforward solution is to scale down *N* for BIME by an early decision algorithm that finds and avoids the redundant searching area. Based on careful observation on the characteristics of HEVC described in Section 2, I describe a meaningful probability that helps to identify the redundant area for bidirectional prediction in Subsection 4.1.1 and Subsection 4.2.1.

Memory access constitutes another complexity issue in BME (herein BME includes BIME, BHME, and BQME). Compared to UIME, that searches blocks within previous frames stored in the reference frame buffer, BME requires interpolating two blocks by a complexity-intensive interpolation filter. In that case, two blocks can be made from two different frames, as shown in (b) of Figure 2.3, which substantially increases the memory access time for accessing reference frames.

To understand the impact of memory access, the HEVC encoder is analyzed in this dissertation by following a similar approach [32] that was adopted by the Joint Collaborative Team on Video Coding (JCT-VC) standardization group on measuring the complexity of the IF, which was also used similarly by Kim *et al.* [22]. Additionally, as the BME process includes quarter-pixel accuracy as well, the memory complexity of both BHME and BQME is measured. Table 3.1 shows the formula for the memory access of the following interpolation filtering (IF) processes related to MV: 1) the IF process for the second MV (i.e., *mvL1*, not *mvL0*); 2) the IF process for BHME; and 3) the IF process for BQME. In the worst case scenario, for example, a storage of roughly 50 gigabytes is required for the aforementioned interpolation filtering processes when the *BQTerrace* sequence is encoded for 10-second video (with 1920 x 1080 resolution at 60 frames per second (FPS)) by only 8 x 4 or 4 x 8 PUs (i.e., the minimum PU sizes for inter prediction).

Table 3.1—Formulas for memory access of interpolation filtering (IF) process in BME

Process	**Formula**
IF for the second MV (mvL1)	9 * w * h + 39 * w + 39 * h + 169
IF for BHME	w * h + 8 * w + 8 * h + 64
IF for BQME	w * h + 7 * w + 7 * h + 49

Note: *w* means width per each PU, and *h* means height per each PU.

3.3. Complexity Analysis of Sub-pixel Motion Estimation in HEVC

In general, the recent HEVC encoder has many rooms to early terminate UIME if the associated block seems not to necessarily search far from the starting position, which contributes on reducing encoding complexity. Accordingly, the recent HM reference software such as HM 15.0 can determine integer-pixel's best case with only 12 search points (small points 4, and big diamond 8 points), whereas sub-pixel's best case (=is the same as the worst case) is larger than that of UIME case, which is always 8 * 2 = 16 points for USME and the same points for BSME. Thus, roughly SME searches points three times larger than that of IME in the best case. Moreover, SME can only be carried after interpolating the associated pixels, which inevitably resulted in increasing lots of complexity burden on encoder.

Interpolation process for SME in HEVC requires far more memory space than the predecessor AVC/H.264 due to the increased filter tap sizes. To be specific, HEVC adopted DCT-based interpolation filter, and its tap size varies on the target sample: 1) 8 tap for half-pixel interpolation and 7 tap for quarter-pixel interpolation on luma sample and 2) on chroma sample, 4 tap interpolation. Detailed process to generate sub-pixels are presented in Subsection 8.5.3.3.3 of HEVC standard specification [33].

Table 3.2—Formulas for memory access of IF process in SME

Process	Formula
IF for HME	w * h + 8 * w + 8 * h + 64
IF for QME	w * h + 7 * w + 7 * h + 49

Note: *w* means width per each PU, and *h* means height per each PU.

Investigating the HEVC specification and HM reference software, the memory complexity of interpolation filtering process was formulated for luma component SME as shown in Table 3.2. Accordingly, the memory complexity of interpolation for SME is $O(w \cdot h)$ in Big O notation where w is width of PU, h is height of PU, and w and h are natural number. Since w is at least one-fourth of height h, the memory complexity can be re-written as $O(h^2)$. Thus, considering the high resolution, this high memory complexity could be a critical problem for memory-constrained encoder application. In the worst case scenario, a memory access of roughly 28 gigabytes may be spent for the interpolation filtering processes of SME (USME and BSME) when the *BQTerrace* sequence is encoded for 10-second video (with 1920 x 1080 resolution at 60 FPS) by 8 x 4 or 4 x 8 PUs (i.e., the minimum PU sizes for inter prediction).

CHAPTER 4
MOTION ESTIMATION SKIPPING STRATEGIES

This section describes two motion estimation skipping strategies for low-complexity HEVC encoding. As pointed in Section 2 and Section 3, two complex modules—BME and SME—are targeted, and each fast decision strategy per each module is proposed in this dissertation. Firstly, statistical approach was taken to find redundancy of ME. Secondly, based on the deducted idea from the probabilities, efficient BME and SME skipping strategies are presented in detail. Finally, experimental results demonstrates the efficiency of proposed strategies in terms of time complexity as well as of memory complexity

4.1. Bidirectional Motion Estimation Skipping Strategy for Low-complexity HEVC Encoding

4.1.1. Correlation of Bi-prediction with Previously Encoded ME Information

In this subsection, the redundancy of BME is analyzed with associated probability so that such redundant BME can be avoided to reduce encoding complexity. Assuming some area that is quite static and if integer-level ME is sufficient (i.e., fractional ME by interpolating real pixels to make imaginary pixels exhibit worse results), BME may not be necessary because the bidirectional prediction technique aims to capture motion between future and previous frames. To validate the aforementioned assumption, commonly used sequences [31] were selected for the analysis of the correlation between bi-prediction and MV accuracy. Let $n(I)$ be the number of the best integer MV during unidirectional ME, and let $n(B)$ be the number of the best bidirectional prediction during PU encoding that consists of IME, SME, and BME. The conditional probability $p(B|I)$ that can determine the correlation of BME on IME/SME results can be obtained through (7) as follows:

$$p(\mathrm{B}|\mathrm{I}) = \frac{p(\mathrm{B}\cap\mathrm{I})}{p(\mathrm{I})} = \frac{n(\mathrm{B}\cap\mathrm{I})}{n(\mathrm{I})} \ , \qquad (7)$$

where $p(I)$ represents the probability that integer MV is the best MV among IME and SME; and $p(B\cap I)$ represents the probability that IMV is the best MV, and simultaneously bi-prediction is the best prediction direction. To obtain the statistics through HEVC encoding, three test sequences (i.e., *BasketballDrive*, *Kimono*, and *BasketballDrill* sequences) were chosen, which can be regarded as representative sequences for motion-intensive or complex-textured video. Table 4.1 shows that $p(B|I)$ is small, especially 2% for 16 x 16 CU size and 1% for 8 x 8 CU size, on average. Note that for the 8 x 8 case, $p(B|Pt_{2Nx2N})$ is always zero percent because bi-prediction is not allowed for other partition in 8 x 8 CU. This correlation can greatly assist in determining whether or not BME is needed. For the exception case, i.e., 64 x 64 CU size, $p(B|I)$ is quite high. Since it is assumed that a larger block tends to have a more complex texture, a variety of prediction types can occur, and the correlation of each ME result can be low. Thus, results can be concluded, as bi-prediction can be redundant in most small CU sizes when integer MV is the best case among IME and SME. This will be utilized for the proposed method described in the next subsection.

Table 4.1—The probability of bi-prediction in the context of IME

CU size	BasketballDrive (1920x1080)		Kimono (1920x1080)		BasketballDrill (832x480)		Average p(B|I)
	n(I)	p(B|I)	n(I)	p(B|I)	n(I)	p(B|I)	
64x64	146,282	39%	85,946	39%	81,392	25%	34%
32x32	775,023	13%	373,644	10%	459,360	9%	11%
16x16	3,170,470	3%	1,540,616	2%	2,028,836	2%	2%
8x8	2,470,601	1%	1,288,071	1%	1,662,499	0%	1%

Table 4.2—The probability of bi-prediction in the context of 2Nx2N encoding

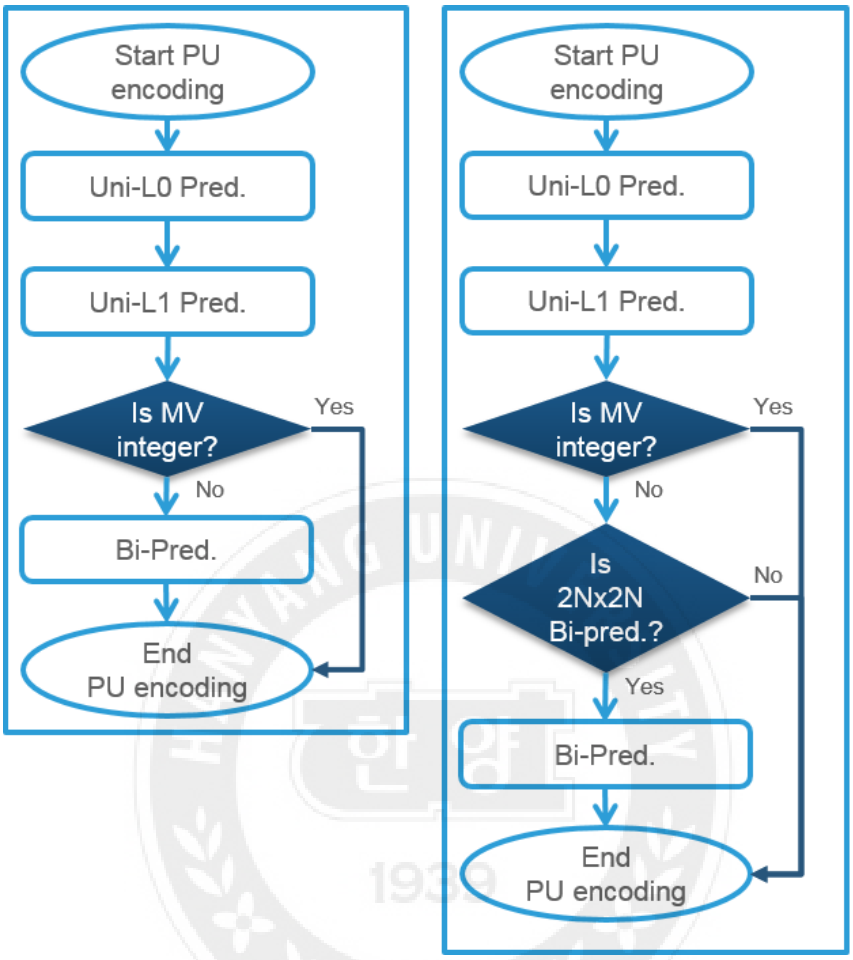
CU size	**BasketballDrive (1920x1080)**		**Kimono (1920x1080)**		**BasketballDrill (832x480)**		**Average** $p(B\vert Pt_{2Nx2N})$
	$\boldsymbol{n(I)}$	$p(B\vert Pt_{2Nx2N})$	$\boldsymbol{n(I)}$	$p(B\vert Pt_{2Nx2N})$	$\boldsymbol{n(I)}$	$p(B\vert Pt_{2Nx2N})$	
64x64	58,599	29%	24,281	26%	9,312	39%	31%
32x32	275,734	16%	126,045	12%	48,979	25%	18%
16x16	1,097,826	5%	496,462	3%	202,874	10%	6%
8x8	4,263,603	0%	1,931,831	0%	775,991	0%	0%

Furthermore, one additional indicator for redundancy can be found when recognizing the special characteristic of HEVC: various PU partitions in a CU may predict the current block with a similar prediction direction. For example, if a 2N x 2N PU was encoded by bidirectional prediction during encoding, other PUs, such as 2N x N and N x 2N, might try to be encoded by the same directional prediction, and vice versa. This PU redundancy was usually exploited for fast PU size decision algorithms (e.g., refer to an early PU decision method [12]). In this dissertation, this redundancy is also investigated to lessen bi-prediction encoding.

Let the conditional probability $p\ (B|Pt_{2Nx2N})$ be the probability of bi-prediction chosen as the best direction in a CU, given the prior probability that 2N x 2N PU did not select bi-prediction as the best direction. With the same aforementioned experimental environment and material, Table 4.2 shows that the trend of $p(B|Pt_{2Nx2N})$ according to CU size is quite similar to $p(B|I)$ in Table 4.1. The probability $p(B|Pt_{2Nx2N})$ is likely to be very small, except for the largest CU size, which means that small CU sizes can avoid unusual bi-prediction processes for fast encoding.

4.1.2. Proposed Method

In this dissertation, a bi-prediction skipping method that conditionally avoids the bi-prediction process is proposed based on two special cases dealt with in Subsection 4.1.1. The first condition is whether or not the best MV of uni-prediction is integer-level. As discussed in Subsection 4.1.1, if the best MV of uni-prediction is an integer unit, then bi-prediction is not likely to be the best direction, in most cases. The second condition is based on the correlation of the prediction direction of each PU partition in a CU, which checks whether or not the final direction of 2N x 2N PU is bi-prediction. Since 2N x 2N PU information is needed, 2N x 2N PU encoding must be preceded, and only remaining partitions can exploit this condition.



(a) (b)

Figure 4.1—Overview of ME process in a PU, including the proposed BME skipping decision method: (a) is for 2N x 2N PU encoding, and (b) is for other PU partitions.

An overview of the ME process that includes the proposed decision method over the HEVC reference encoder is presented in Figure 4.1. In both subfigures in Figure 4.1, the proposed condition states (conditional branches) are inserted with a darker color as diamond-shaped decision blocks. For 2N x 2N PU encoding, only the first condition factor can be observed. However, for other PU partitions {2N x N, N x 2N, 2N x nU, 2N x nD, nL x 2N, nR x 2N}, both conditions are checked to avoid bi-prediction. When those proposed conditions are met for a certain block (a PU in HEVC), the proposed encoder will not go through any processes related to bi-prediction: BIME, BHME, BQME, and the bidirectional motion compensation (BMC). In comparison with the reference HEVC encoder that searches a bi-predicted block within multiple reference frames and squared search range as shown in Figure 2.3, the proposed method can skip those complex searching significantly. According to the common test conditions that set the number of reference frames to four, HEVC searches four frames in most cases for a block (i.e., a PU), whereas the proposed method would not spend any time in searching frames when the proposed condition is met for the block. Note that no ME process is performed for uni-prediction L1 in the GBP case in default HEVC encoding configuration.

4.1.3. Experimental Results

Extensive experiments were conducted according to CTC [31] that specified test material and coding conditions, which is usual in the standardization process of HEVC, as well as in the literature. An overview of the test environment is briefly presented in Table 4.3. Note that LD scenario was chosen to evaluate the effectiveness of the proposed method since LD scenario can have the most B frames. In the test environment, the computational complexity is evaluated in three ways to effectively measure the efficiency of the proposed method: 1) the number of bi-prediction function calls; 2) the actual encoding time; and 3) the number of memory access. To evaluate the time complexity, the following development environment was employed: four-core CPUs with eight logical threads running at 4.00 GHz, at least 16 gigabytes random access memory (RAM), a solid-state drive and a RAM disk, and a 64-bit operating system (OS). In addition, the compression performance is shown in terms of Bjøntegaard-Delta bitrate (BD-rate) [34] measurement method that represents bitrate ratios in assumption of equal peak signal-to-noise ratio (PSNR), using bitstream encoded by four QP values as shown in Table 4.3.

As an anchor, the recent HEVC test model (HM) 15.0 is used, excluding the reduction of the identical bi-prediction motion compensation technique (RoIBMC) [22], which has been adopted to HEVC so that the performance of RoIBMC can also be compared with the proposed method. The upper CU-based early bi-prediction skipping (UCU-EBS) method [21] and the motion-based adaptive search range (MASR) method [23] were also tested to evaluate the comparative performance of the proposed method. The reason to choose two methods for comparison is as follows: UCU-EBS method is the most similar method with the proposed method and reported significant complexity reduction, and MASR method is a reputable method in the field of reducing BME complexity. Those existing methods and the proposed method were built on top of the anchor and evaluated. Since the performance of UCU-EBS can be different according to the range setting, 16 x 16 and 8 x 8 CUs were targeted for the best performance in this dissertation.

Table 4.3—Test materials and conditions

Test sequences, resolution, frame rate, and motion types	Class B (1920 x 1080): Kimono (24, motion-intensive), ParkScene (24, motion-intensive), Cactus (50, static), and BasketballDrive (50, motion-intensive)
	Class C (832 x 480): BasketballDrill (50, motion-intensive), BQMall (60, motion-intensive), PartyScene (50, motion-intensive), and RaceHorsesC (30, motion-intensive)
	Class D (416 x 240): BasketballPass (50, motion-intensive), BQSquare (60, motion-intensive), BlowingBubbles (50, motion-intensive), RaceHorses (30, motion-intensive)
	Class E (1280 x 720): FourPeople (60, static), Johnny (60, static), and KristenAndSara (60, static)
Total encoded frames	Frame rate multiple by 10 s
Software	HEVC Test Model (HM) 15.0
Configuration	Low-delay B main
Quantization parameter	22, 27, 32, and 37

4.1.3.1. Function Call Ratio

To understand the impact of how many times the proposed method skips redundant BME and BMC, the number of function calls for bi-prediction was measured. Note that this function includes not only integer-level BME, but also computationally expensive fractional-level BME that requires interpolating pixels. Since the BME can be performed as many as the available number of reference frames, each BME per reference frame is counted accumulatively. In addition, one function call should not be counted per PU equally as the PU size varies in the current HEVC coding structure. Accordingly, an equation is developed to count the function call with weight of CU of which area has width *W* by height *H* as the following:

$$\Phi_m = \sum_{d=0}^{max(CU\ Depth)} \frac{W_d \cdot H_d}{\sqrt{max(W_d) \cdot max(H_d)}} \cdot Count_d, \tag{4}$$

where Φ_m is the sum of weighted bi-prediction function calls of a method *m* according to the CU depth-dependent weighting factor, $Count_d$ is the number of bi-prediction function call counted in the depth *d*, W_d and H_d are the width and height of CU for the depth *d*, and $max(X)$ is the function that outputs maximum integer value among input list *X*. In general, the maximum number of CU depth is 3 as CU depth $\in\{0, 1, 2, 3\}$ and the maximum number of CU width is 64, which is the same as the maximum CU height.

Table 4.4—Ratio of bi-prediction function call compared to the anchor

Sequence		Φ_{anchor}	R_{MASR}	$R_{UCU\text{-}EBS}$	$R_{Proposed}$
Class B	Kimono	540,786,575	99.94%	91.46%	29.18%
	ParkScene	556,072,719	99.94%	89.67%	32.92%
	Cactus	1,137,054,611	99.91%	93.44%	18.73%
	BasketballDrive	1,159,052,327	99.95%	90.50%	31.25%
Class C	BasketballDrill	229,651,845	99.93%	90.59%	23.99%
	BQMall	277,646,125	99.92%	91.02%	29.46%
	PartyScene	257,687,014	99.93%	92.18%	44.29%
	RaceHorsesC	153,196,700	99.94%	84.09%	37.58%
Class E	FourPeople	574,292,643	99.98%	96.88%	9.66%
	Johnny	566,854,258	99.96%	95.78%	10.91%
	KristenAndSara	571,965,802	99.95%	95.47%	14.15%
Average		457,942,464	99.94%	91.85%	23.07%

Note: Average values are calculated by geometrical mean.

Table 4.4 shows the ratio R_m of Φ_m, where $m \in$ {MASR [23], UCU-EBS [21], the proposed method}. R_m is computed by Φ_m divided by Φ_{anchor} as shown in (5) as follows:

$$R_m = \frac{\Phi_m}{\Phi_{anchor}} \ . \tag{5}$$

As shown in the table, MASR does not contribute much on reducing bi-prediction processes, although it might be helpful on relieving some part of the bi-prediction processes. UCU-EBS certainly reduces some function calls, but the proposed method further reduced bi-prediction function calls in all test sequences. On average, the proposed method removed 77% of bi-prediction function calls that should be performed when the anchor was used. Compared to the existing method, the ratio of the function call number in the proposed method is always smaller than UCU-EBS, specifically 46-87% smaller than that of UCU-EBS.

4.1.3.2. Encoding Time and Compression Performance

Since the bi-prediction process is only a part of the encoder, it is important to check the overall complexity (in terms of execution time) and coding efficiency. Additionally, the execution time needed for bi-prediction process that consists of BIME, BHME, and BQME was measured. The total encoding time ratio T_{total} that is compared to the anchor can be obtained as the following:

$$T_{total} = \frac{\text{geomean}(T_m)}{\text{geomean}(T_{anchor})} \ , \tag{10}$$

where m represents one of the aforementioned methods and geomean represents geometric mean. The bi-prediction time ratio T_{bi} can also be obtained similarly as in (10). For the compression performance measurement, Y, U, and V are used, which represent BD-rate ratios of luminance and chrominance color components from those of the anchor.

Table 4.5, Table 4.6, and Table 4.7 show the compression performance (BD-rate results of *Y*, *U*, and *V*) and the time complexity (T_{total} and T_{bi}) of the existing methods (MASR and UCU-EBS) and the proposed method in comparison with the anchor. In general, all methods did not degrade compression performance noticeably. On the contrary, when comparing the time complexity of bi-prediction, it is clear that the proposed method outperforms the others: on average, the proposed method reduced the bi-prediction time to 36%, MASR reduced the bi-prediction time to 92%, and UCU-EBS to 83%. By evaluating each sequence, it is noted that two existing methods reached at most 84% (*FourPeople* case by MASR) and 62% (*BasketballPass* case by UCU-EBS), whereas significant reduction is achieved by the proposed method for *FourPeople* sequence with a substantially reduced complexity (i.e. 13%). In the evaluation of the total encoding time, existing methods resulted a decrease of averagely 2-4% in complexity; the proposed method a decrease of 16% (four times more than others). It is also noted that the proposed method degraded the BD-rate 0.5% on average. The BD-rate loss of the proposed method can be regarded as insignificant. The proposed method mostly sustained the coding efficiency, but for *BQSquare, PartyScene, and RaceHorses* sequences, in which the losses of BD-rate in Y (luminance), U (chrominance), and V (chrominance) components are quite noticeable, showing not much the complexity reduction. Since these sequences contain many moving objects with the camera zooming in frame-by-frame, bi-prediction can be efficient in this case. On the other hand, the proposed method significantly reduced the encoding time when encoding sequences were in Class E (more than 20% of the total encoding time), which mostly consists of static scenes, as these sequences are for video conferencing that usually has no camera motion, but a few moving objects.

Table 4.5—Comparison results of BD-rate, bi-prediction time, and encoding time for the MASR [21] method compared to the anchor

		BD-rate_{MASR}			T_{MASR} / T_{anchor}	
Sequence		**Y**	**U**	**V**	T_{bi}	T_{total}
Class B	Kimono	0.2%	0.6%	0.2%	98%	99%
	ParkScene	-0.1%	-0.3%	-0.2%	95%	99%
	Cactus	0.2%	0.2%	0.1%	89%	97%
	BasketballDrive	0.3%	0.1%	0.3%	97%	100%
Class C	BasketballDrill	0.3%	0.4%	0.7%	89%	98%
	BQMall	0.2%	0.2%	-0.1%	95%	99%
	PartyScene	0.2%	0.1%	0.1%	93%	99%
	RaceHorsesC	0.1%	0.4%	0.3%	97%	100%
Class D	BasketballPass	0.1%	-0.7%	0.1%	94%	99%
	BQSquare	0.2%	-0.6%	-0.2%	94%	99%
	BlowingBubbles	0.2%	-0.3%	0.1%	91%	98%
	RaceHorses	0.0%	0.6%	0.5%	97%	100%
Class E	FourPeople	0.2%	0.2%	0.4%	84%	95%
	Johnny	0.2%	0.1%	1.0%	86%	96%
	KristenAndSara	0.3%	0.0%	0.4%	88%	96%
Average		**0.2%**	**0.1%**	**0.2%**	**92%**	**98%**

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

Table 4.6—Comparison results of BD-rate, bi-prediction time, and encoding time for the UCU-EBS [23] method compared to the anchor

Sequence		BD-rate$_{UCU-EBS}$			$T_{UCU-EBS}$ / T_{anchor}	
		Y	**U**	**V**	T_{bi}	T_{total}
Class B	Kimono	0.1%	0.4%	0.2%	90%	98%
	ParkScene	0.1%	-0.1%	-0.2%	87%	97%
	Cactus	0.2%	0.4%	0.1%	92%	98%
	BasketballDrive	0.2%	0.3%	0.2%	89%	97%
Class C	BasketballDrill	0.2%	0.5%	0.4%	89%	97%
	BQMall	0.3%	0.0%	0.0%	89%	97%
	PartyScene	0.3%	0.1%	0.7%	91%	98%
	RaceHorsesC	0.5%	0.5%	0.5%	81%	96%
Class D	BasketballPass	0.5%	0.0%	0.3%	62%	92%
	BQSquare	1.6%	2.8%	0.1%	69%	92%
	BlowingBubbles	0.8%	0.5%	1.2%	64%	91%
	RaceHorses	0.5%	0.7%	0.9%	65%	93%
Class E	FourPeople	0.1%	-0.5%	-0.1%	96%	99%
	Johnny	0.1%	0.3%	0.7%	95%	99%
	KristenAndSara	0.1%	0.0%	-0.3%	95%	98%
Average		**0.4%**	**0.4%**	**0.3%**	**83%**	**96%**

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

Table 4.7—Comparison results of BD-rate, bi-prediction time, and encoding time for the proposed method compared to the anchor

Sequence		BD-rate$_{prop}$			T_{prop} / T_{anchor}	
		Y	**U**	**V**	T_{bi}	T_{total}
Class B	Kimono	0.4%	0.9%	0.6%	40%	86%
	ParkScene	0.3%	0.5%	0.6%	44%	86%
	Cactus	0.6%	0.9%	0.0%	25%	81%
	BasketballDrive	0.6%	1.0%	0.6%	42%	87%
Class C	BasketballDrill	0.5%	0.4%	0.8%	30%	83%
	BQMall	0.6%	0.6%	0.5%	39%	84%
	PartyScene	0.6%	0.8%	0.9%	52%	88%
	RaceHorsesC	0.5%	0.7%	0.8%	46%	89%
Class D	BasketballPass	0.3%	-0.9%	0.7%	50%	89%
	BQSquare	0.9%	0.8%	0.8%	62%	90%
	BlowingBubbles	0.5%	0.2%	0.4%	52%	88%
	RaceHorses	0.2%	0.0%	0.3%	55%	91%
Class E	FourPeople	0.8%	0.3%	1.0%	13%	73%
	Johnny	0.9%	1.0%	1.0%	15%	73%
	KristenAndSara	0.6%	0.7%	0.3%	20%	76%
Average		**0.5%**	**0.5%**	**0.6%**	**36%**	**84%**

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

According to CTC, four QP values should be used to get a BD-rate for coding efficiency. As QP goes higher, bitstream size goes smaller and the encoding time also goes slightly faster. Intriguingly, I observed that the proposed method reduced the bi-prediction time more effectively when QP was high. For example, the time differences between QP equal to 22 and QP 37 are 18% in *PartyScene* sequence at maximum and 7% in *FourPeople* sequence at minimum. In high QP setting, it is assumed that the encoder tends to select low-bit prediction modes (e.g., skip mode that skips delivering motion-related data to bitstream) than the other modes where the proposed method is more likely to be effective. In other words, 2N x 2N PU tend not to be encoded with bi-prediction mode, and each PU tend to be encoded with integer-level MVs, so the proposed conditions can be satisfied easily.

Figure 4.2 shows brief comparison results of coding efficiency and of BME time saving. It is clearly shown that the proposed method saved much further BME time compared to existing methods while the coding loss is similarly sustained. Whereas UCU-EBS method saved at most 17%, the proposed method saved 64% which is roughly four times than that of UCU-EBS method. In conclusion, the proposed BME skipping strategy can reduce significant BME time, sustaining coding efficiency as similar as the existing methods showed.

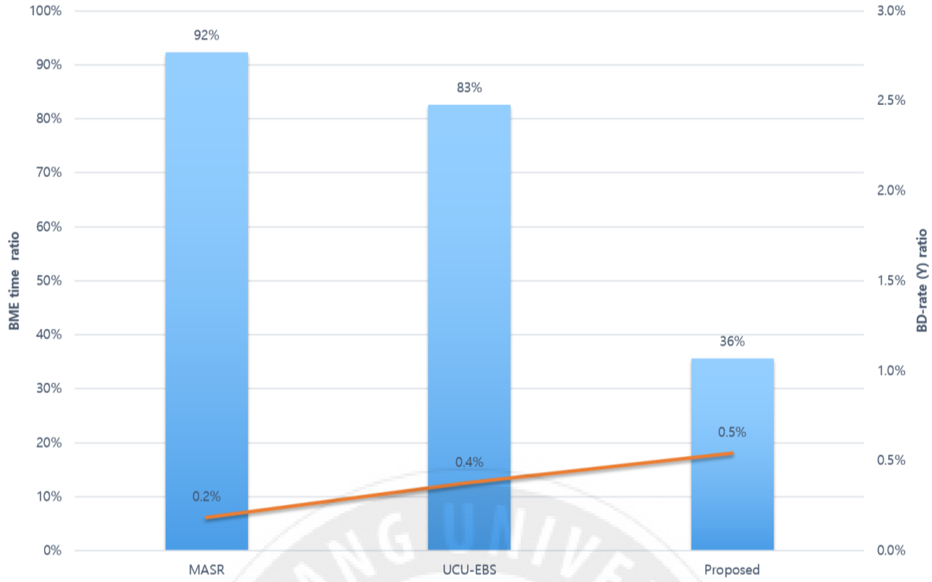


Figure 4.2—Average performance results of three methods compared to the anchor

Table 4.8—Ratio of memory access reduction compared to RoIBMC

Sequence	RoIBMC	Ratio of methods to RoIBMC		
		MASR	UCU-EBS	Proposed
Kimono	98,493,160,362	87%	86%	37%
ParkScene	99,386,754,484	88%	83%	41%
Cactus	208,613,949,684	87%	90%	22%
BasketballDrive	210,993,864,544	88%	85%	39%
Average	154,371,932,269	88%	86%	35%

4.1.3.3. Memory Access

The same formula defined in Table 3.1 was used to measure the number of memory access. In this subsection, the comparative percentage of the number of memory access is shown by comparing UCU-EBS and the proposed method. Additionally, RoIBMC was compared, which is the state-of-the-art algorithm that decreased MC complexity significantly in a straightforward manner. In this experiment, the HD 1080p sequences were tested with QP equal to 32, as the memory issue is critical to the video encoding application when the resolution is large. Table 4.8 presents the ratio of the number of memory access that is curtailed by MASR, UCU-EBS, and the proposed method in comparison with RoIBMC. The proposed method clearly demonstrated high efficiency in terms of memory access, as well in this subsection. Specifically, the memory access of interpolation filtering processes related to bi-prediction is decreased to 35%, on average, even though MASR can decrease the memory access to 88%, and UCU-EBS to 86%, on average.

4.2. Sub-pixel Motion Estimation Skipping Strategy for Low-complexity HEVC Encoding

4.2.1. Correlation of Sub-pixel MV with Previously Encoded ME Information

In this subsection, the redundancy of SME is analyzed with associated probability to avoid redundant SME. As similarly assumed in Subsection 4.1.1, some area that is quite static and if quarter-pixel MV is insufficient, SME (or at least QME) may not be necessary because the sub-pixel MVs aims to capture a certain motion more accurately, not to capture no motion. In addition to this assumption, various PU partitions in a CU may predict the current block with a similar motion accuracy. For example, if a 2N x 2N PU was encoded by either zero MV or IMV or SKIP mode (no MVD encoding is needed) during encoding, other PUs, such as 2N x N and N x 2N, might try to be encoded by IMV. Similar PU redundancy was exploited for fast IME algorithms (e.g., fast IMV search [35], [36]). In this dissertation, this PU redundancy to discover very static motion is investigated to lessen SME encoding.

Let $n(VS\_Pt_{2Nx2N})$ be the number of case that 2N x 2N PU selects very simple motion (VSM)—zero MV, IMV, or SKIP mode—as the best result, and let the conditional probability $p(VS\_Pt_{best}|VS\_Pt_{2Nx2N})$ be the probability of the VSM chosen as the best result during inter prediction encoding, given the prior probability that 2N x 2N PU selects the VSM as the best result. Table 4.9 is the statistical results obtained from 6 sequences in CTC [31] through HM 16.14 encoding fully with QP 32 under LD configuration. It is clearly shown that $p(VS\_Pt_{best}|VS\_Pt_{2Nx2N})$ is very high (99%) on average, and no sequences has less than 98%. Thus, SME can be useless in most PU blocks when 2N x 2N PU was encoded by VS motion. This will be utilized for the proposed method described in the next subsection.

Furthermore, additional possibility that SME is inefficient exist during multiple reference frame selection. HEVC encoder commonly searches motion among four reference frames, but except the very previous one, the distance between the current frame and other reference frames can be quite far due to the hierarchically layered frames with different QP assignment strategy (the associated description can be found in HM encoder description [18]). With this special characteristics of HEVC, I assumed that the motion accuracy result of the first reference frame search may be a good hint to understand whether SME of other reference frame search is useless or not.

Table 4.9—The probability of inefficient SME case in the context of very static motion under LD configuration

Sequence		$n(VS\_Pt_{2Nx2N})$	$p(VS\_Pt_{best} \mid VS\_Pt_{2Nx2N})$
Class B	Kimono	8,883,733	99.45%
	BasketballDrive	19,419,439	99.29%
Class C	BasketballDrill	3,554,010	99.15%
	BQMall	4,390,051	98.81%
Class E	FourPeople	11,145,557	99.65%
	Johnny	11,306,115	99.81%
Average		9,783,151	99.36%

Table 4.10—The probability of inefficient QME case in the context of reference frame search under LD configuration

Sequence		$n(\sim QMV_{ref\_0})$	$p(\sim QMV_{ref\_oth} \mid \sim QMV_{ref\_0})$
Class B	Kimono	121,150,402	51.91%
	BasketballDrive	397,770,283	83.51%
Class C	BasketballDrill	69,063,886	76.53%
	BQMall	63,287,555	61.67%
Class E	FourPeople	260,612,442	93.47%
	Johnny	250,306,529	90.40%
Average		193,698,516	76.25%

To validate the assumption, the followings are investigated: the relevance between motion accuracy—particularly, quarter-pixel MV—of the first reference frame search and that of other frames. Let $n(\sim QMV_{ref\_idx\_0})$ be the number of the case that the best MV of reference frame index 0 (hereafter *ref_idx_0*) is not quarter-pixel MV, and let $n(\sim QMV_{ref\_idx\_oth})$ be the number of the case that the best MV of other reference frame indices except 0 (hereafter *ref_idx_oth*) is not quarter-pixel MV. By counting $n(\sim QMV_{ref\_idx\_0})$ and $n(\sim QMV_{ref\_idx\_oth})$, it is possible to derive the conditional probability $p(\sim QMV_{ref\_idx\_oth}|\sim QMV_{ref\_idx\_0})$ which can reveal the effectiveness of QME in most reference frames when the best MV in *ref_idx_0* is given. The statistics of the conditional probability $p(\sim QMV_{ref\_idx\_oth}|\sim QMV_{ref\_idx\_0})$ is shown in the Table 4.10 obtained from 6 sequences in CTC [31] through HM 16.14 encoding fully with QP 32 under LD configuration. On all sequences, $p(\sim QMV_{ref\_idx\_oth}|\sim QMV_{ref\_idx\_0})$ is more than 50%, and on average, the conditional probability reached 76%, which can be interpreted that there exist correlation of QMV with reference frame search when knowing motion accuracy of *ref_idx_0* during ME. This will also be utilized for the proposed method described in the next subsection.

4.2.2. Proposed Method

In this subsection, a SME skipping method that conditionally avoids QME and/or HME processes is presented based on two special cases dealt with in Subsection 4.2.1. The first condition is whether or not 2N x 2N PU prefers VSM (zero MV, IMV, or SKIP mode) during encoding. As discussed in Subsection 4.2.1, if the best result of 2N x 2N PU were VSM during encoding, then SME was likely to be useless in most cases. The second condition is based on the correlation of QMV with reference frame index, which checks whether or not the final result of ME in the index *ref_idx_0* is not QMV (i.e., motion accuracy is either IMV or HMV). Since 2N x 2N PU information is needed, 2N x 2N PU encoding must be preceded, and only remaining partitions can exploit this condition.

Two overview flowcharts of ME process that includes the proposed decision method over the HEVC reference encoder are presented in Figure 4.3 and Figure 4.4. In both figures, the proposed condition states (conditional branches) are inserted with a darker color as diamond-shaped decision blocks. For 2N x 2N PU encoding, only the second condition factor (correlation between QME and reference frame index) can be observed to avoid QME. However, for other PU partitions {2N x N, N x 2N, 2N x nU, 2N x nD, nL x 2N, nR x 2N}, both first and second conditions are checked to avoid SME. When those proposed conditions are met for a certain block (a PU in HEVC) during each reference frame search, the proposed encoder will not go through any processes related to SME (or QME only): motion estimation process as well as IF process.

In comparison with the reference HEVC encoder that searches the best sub-pixel MV within multiple reference frames after interpolating pixels, the proposed method can skip those complex searching and the associated IF processes significantly. According to the CTC that sets the number of reference frames to four, HEVC searches four frames in most cases for a block (i.e., a PU), whereas the proposed method would not spend any time in searching frames when the proposed SME skipping condition in Subsection 4.2.1 is met for the block. Moreover, the proposed method would not spend IF process for QMV when the firstly proposed condition in Subsection 4.2.1 is met for the block.

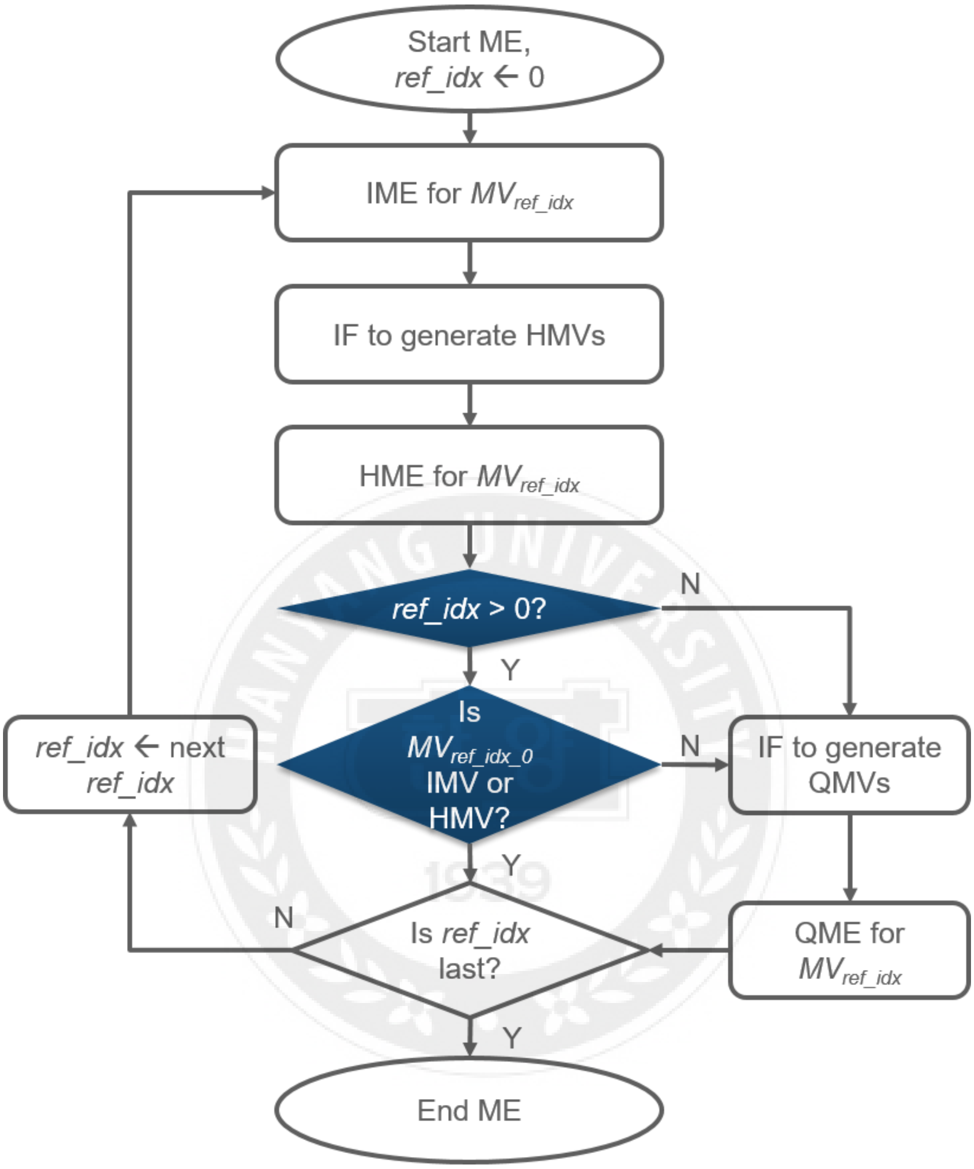


Figure 4.3—Overview of ME process in a PU, including the proposed SME skipping method for 2N x 2N PU encoding.

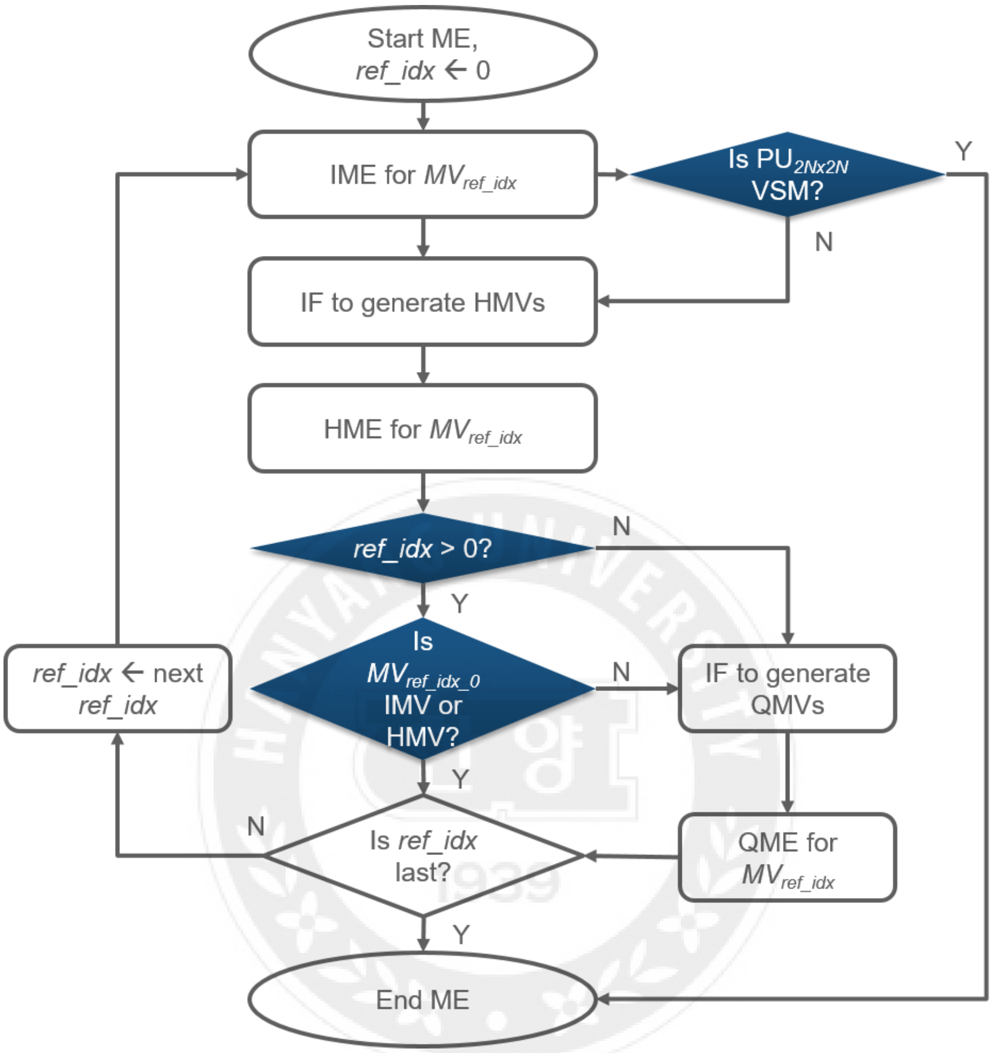


Figure 4.4—Overview of ME process in a PU, including the proposed SME skipping method for other PU partitions encoding except 2N x 2N PU.

4.2.3. Experimental Results

Extensive experiments were conducted according to the common test condition (CTC) [31]. An overview of the test environment is briefly presented in Table 4.11. In the test environment, the computational complexity is evaluated in two ways to effectively measure the efficiency of the proposed method: 1) the actual encoding time and 2) the number of memory access. To evaluate the time complexity, the following development environment was employed: four-core CPUs with eight logical threads running at 4.00 GHz, at least 16 gigabytes RAM, a solid-state drive, and a 64-bit OS. In addition, the compression performance is shown in terms of BD-rate measurement method, using bitstream encoded by four QP values as shown in Table 4.11.

As an anchor, the recent HM 16.14 was used. In addition to evaluate the comparative performance of the proposed method, Error Surface-based Fast SME (ES-FSME) algorithm [28] regarded as a representative research on skipping SME by error surface modeling was also tested. This existing method and the proposed method were built on top of the anchor and evaluated. Note that the performance of ES-FSME can be different according to three sub-strategies in [28]. Among them, the fastest strategy of [28] was chosen in this dissertation.

Table 4.11—Test materials and conditions

Test sequences, resolution, frame rate	Class A (2560 x 1600): Traffic (30), PeopleOnStreet (30), Nebuta (60, bit depth 10), SteamLocomotiveTrain (60, bit depth 10)
	Class B (1920 x 1080): Kimono (24), ParkScene (24), Cactus (50), BasketballDrive (50), BQTerrace (60)
	Class C (832 x 480): BasketballDrill (50), BQMall (60), PartyScene (50), and RaceHorsesC (30)
	Class D (416 x 240): BasketballPass (50), BQSquare (60), BlowingBubbles (50), and RaceHorses (30)
	Class E (1280 x 720): FourPeople (60), Johnny (60), and KristenAndSara (60)
Total encoded frames	Frame rate multiple by 10 seconds except Class A sequences (frame rate multiple by 5 seconds)
Software	HEVC Test Model (HM) 16.14
Configuration	Random Access (RA) main and Low-delay B (LD) main
QP	22, 27, 32, and 37

4.2.3.1. Encoding Time and Compression Performance

Since the SME is only a part of the encoder, it is important to check the overall computational complexity (in terms of execution time) and coding efficiency. Additionally, the execution time needed for SME that consists of IF process to generate sub-pixels was measured. The total encoding time ratio, T_{total}, and the time ratio of SME, T_{SME}, can also be obtained similarly as in (10). For the compression performance measurement, *Y*, *U*, and *V* are used.

The compression performance and the time complexity (T_{total} and T_{SME}) of the proposed method is presented in Table 4.12 (RA configuration) and Table 4.13 (LD configuration) in comparison with the anchor. In both configurations, the proposed method mostly maintained the coding efficiency of the anchor, showing less than a 0.8% loss of BD-rate on average. Nevertheless, the proposed method significantly reduced T_{SME}, which reached about a half of the original complexity of the anchor. Thanks to the successful reduction of the most time-consuming module at encoder, the total encoding time could be substantially reduced—especially, the time was reduced by 77% in LD configuration. In RA case, the best performance in SME time saving appeared at *Cactus* sequence that contains just a few moving objects, reducing the complexity of the anchor by 60%. In LD case, the significant time reduction of SME appeared at *class E* sequences, reducing the time to 30-34%.

For the comparison with the ES-FSME method, the aforementioned performance was measured, and presented in Table 4.14 (RA case) and Table 4.15 (LD case). On average, ES-FSME method decreased substantially the overall encoding time and the SME time regardless of coding configuration. However, ES-FSME method faced noticeable coding loss (i.e., a 3.2% loss of BD-rate in RA case and a 2.8% loss in LD case, on average). This result may mean that the accuracy of error surface model used in ES-FSME method might not so accurate to predict the best sub-pixel MV although the coding time was quite decreased substantially.

When comparing the performance results between the proposed method and the existing method, it is clear that the proposed method outperforms the existing method in terms of not only encoding time, but also coding efficiency. To be specific, the proposed method showed a 2.8% gain of BD-rate and an 8% gain of SME time saving on average under RA case in comparison with ES-FSME method. Additionally, the proposed method showed a 2.1% gain of BD-rate and a 14% gain of SME time saving on average under LD case in comparison with ES-FSME method. The average comparison result is shown in Figure 4.5 which clearly shows the excellence of the proposed method, particularly sustaining coding efficiency (less than 1%) whereas the ES-FSME method dropped roughly a 3% loss of BD-rate.

Table 4.12—Comparison results of BD-rate, SME time, and encoding time for the proposed method compared to the anchor under RA case

Sequence		**Y**	**U**	**V**	T_{SME}	T_{total}
Class A	Traffic	0.3%	0.0%	0.0%	41%	74%
	PeopleOnStreet	0.4%	0.5%	0.3%	59%	90%
	Nebuta	0.1%	0.2%	0.4%	67%	89%
	SteamLocomotive	0.2%	0.4%	0.4%	43%	86%
Class B	Kimono	0.1%	0.2%	-0.1%	48%	85%
	ParkScene	0.3%	0.1%	0.0%	45%	79%
	Cactus	0.3%	0.2%	0.2%	40%	82%
	BasketballDrive	0.2%	0.1%	-0.1%	49%	88%
	BQTerrace	0.6%	-0.1%	-0.1%	43%	76%
Class C	BasketballDrill	0.3%	0.1%	0.1%	47%	85%
	BQMall	0.4%	0.0%	0.0%	47%	82%
	PartyScene	0.6%	0.3%	0.3%	54%	84%
	RaceHorsesC	0.4%	0.3%	0.3%	65%	92%
Class D	BasketballPass	0.3%	0.2%	0.2%	57%	88%
	BQSquare	1.1%	0.1%	0.1%	47%	77%
	BlowingBubbles	0.5%	0.2%	0.1%	50%	80%
	RaceHorses	0.5%	0.2%	-0.1%	67%	91%
Average		0.4%	0.2%	0.1%	51%	84%

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

Table 4.13—Comparison results of BD-rate, SME time, and encoding time for the proposed method compared to the anchor under LD case

Sequence		**Y**	**U**	**V**	T_{SME}	T_{total}
Class B	Kimono	0.1%	0.3%	0.0%	53%	80%
	ParkScene	0.6%	0.7%	0.5%	51%	77%
	Cactus	0.3%	0.3%	0.1%	55%	81%
	BasketballDrive	0.5%	0.1%	0.3%	44%	74%
	BQTerrace	0.9%	1.5%	0.8%	48%	75%
Class C	BasketballDrill	0.4%	0.0%	0.5%	50%	78%
	BQMall	0.6%	0.8%	1.1%	51%	78%
	PartyScene	1.0%	1.1%	0.9%	63%	84%
	RaceHorsesC	0.4%	0.5%	0.5%	68%	88%
Class D	BasketballPass	0.3%	0.3%	0.4%	60%	84%
	BQSquare	2.1%	1.4%	2.8%	59%	81%
	BlowingBubbles	1.1%	1.2%	1.5%	58%	81%
	RaceHorses	0.5%	0.5%	0.2%	70%	89%
Class E	FourPeople	0.5%	0.1%	0.2%	31%	62%
	Johnny	0.9%	2.1%	2.0%	30%	60%
	KristenAndSara	0.6%	0.1%	0.9%	34%	64%
Average		0.7%	0.7%	0.8%	52%	77%

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

Table 4.14—Comparison results of BD-rate, SME time, and encoding time for the ES-FSME method compared to the anchor under RA case

Sequence		Y	U	V	T_{SME}	T_{total}
Class A	Traffic	2.3%	0.5%	0.7%	58%	84%
	PeopleOnStreet	3.5%	3.2%	2.9%	61%	93%
	Nebuta	0.3%	0.8%	0.5%	63%	92%
	SteamLocomotive	0.8%	1.0%	0.5%	0%	0%
Class B	Kimono	1.4%	0.8%	1.0%	60%	89%
	ParkScene	2.6%	1.2%	1.2%	60%	86%
	Cactus	2.3%	1.5%	1.7%	54%	87%
	BasketballDrive	2.6%	2.0%	1.8%	60%	91%
	BQTerrace	2.3%	0.8%	0.9%	58%	84%
Class C	BasketballDrill	2.2%	1.4%	1.4%	56%	88%
	BQMall	4.1%	2.4%	2.5%	58%	87%
	PartyScene	4.6%	2.7%	2.9%	59%	87%
	RaceHorsesC	4.7%	2.8%	3.0%	61%	92%
Class D	BasketballPass	3.8%	2.9%	2.6%	59%	89%
	BQSquare	5.4%	3.1%	3.6%	59%	85%
	BlowingBubbles	5.1%	3.1%	3.2%	57%	85%
	RaceHorses	6.0%	3.7%	3.6%	61%	90%
Average		3.2%	2.0%	2.0%	59%	88%

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

Table 4.15—Comparison results of BD-rate, SME time, and encoding time for the ES-FSME method compared to the anchor under LD case

Sequence		**Y**	**U**	**V**	T_{SME}	T_{total}
Class B	Kimono	1.1%	0.5%	0.5%	68%	87%
	ParkScene	2.4%	1.1%	1.0%	68%	86%
	Cactus	1.8%	0.9%	1.2%	68%	88%
	BasketballDrive	2.1%	0.9%	1.0%	64%	85%
	BQTerrace	1.2%	0.5%	-0.4%	66%	86%
Class C	BasketballDrill	2.0%	2.0%	1.5%	66%	86%
	BQMall	2.7%	1.8%	2.5%	67%	86%
	PartyScene	3.1%	2.6%	2.7%	68%	88%
	RaceHorsesC	4.0%	2.7%	2.8%	69%	89%
Class D	BasketballPass	3.1%	3.0%	2.9%	67%	88%
	BQSquare	4.1%	5.4%	3.1%	67%	87%
	BlowingBubbles	3.8%	3.2%	3.7%	67%	87%
	RaceHorses	5.6%	3.7%	3.7%	69%	89%
Class E	FourPeople	2.2%	1.3%	0.9%	61%	80%
	Johnny	3.1%	2.1%	2.1%	62%	80%
	KristenAndSara	2.6%	1.4%	2.0%	63%	81%
Average		2.8%	2.1%	2.0%	66%	86%

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

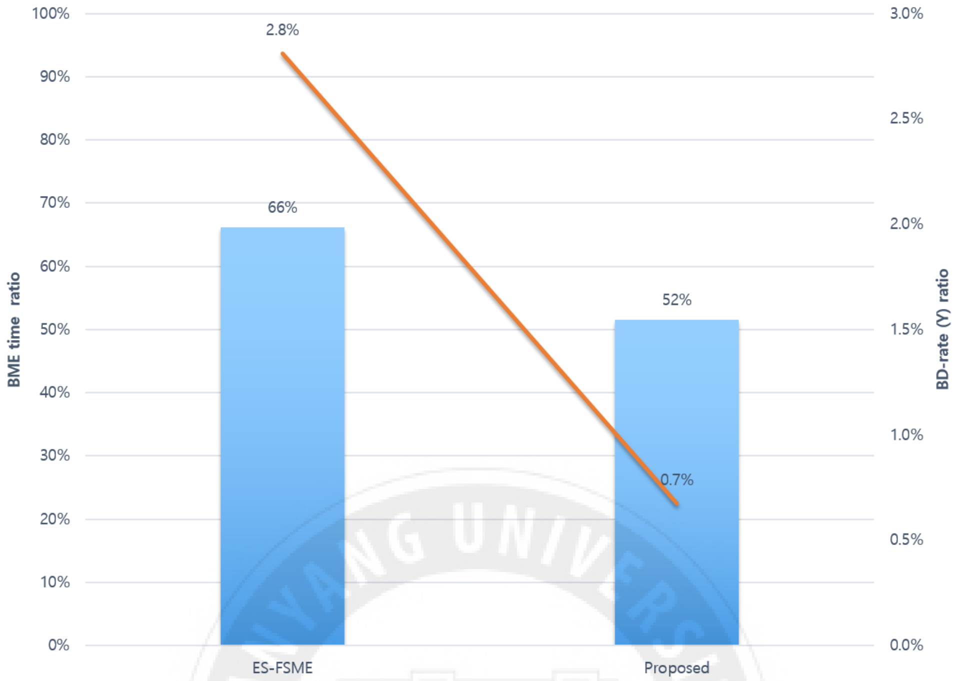


Figure 4.5—Average performance results of two methods compared to the anchor

4.2.3.1. Memory Access

The same formula defined in Table 3.2 was used to measure the number of memory access. In this subsection, the comparative percentage of the number of memory access is shown by comparing ES-FSME method and the proposed method. In this experiment, four HD 1080p sequences recommended in CTC were tested with QP equal to 32, as the memory issue is critical to the video encoding application when the resolution is large. Table 4.16 presents the ratio of the number of memory access that is curtailed by ES-FSME method and the proposed method in comparison with the anchor. The proposed method clearly demonstrated high efficiency in terms of memory access, as well in this subsection. Specifically, the memory access of IF processes related to SME is decreased to 41%, on average, even though ES-FSME can decrease the memory access to about 79% on average.

4.3. Results of Combined Motion Estimation Skipping Strategies

In this dissertation, two ME skipping strategies are presented to reduce the encoding complexity of HEVC. To understand the effect of combining two strategies, additional experiments are conducted in this subsection by integrating two strategies in one HM platform. In other words, the proposed BME skipping strategy was integrated in the BME process of HM and the proposed SME skipping strategy was integrated in the SME process of the same HM. Particularly, HM 16.14 was chosen for the platform. Similar coding conditions and test environment were carried for this experiment. The differences of the test condition from Subsection 4.1 and Subsection 4.2 are as follows: six sequences are chosen, and QP equal to 32 is only used to measure running time.

Table 4.16—Ratio of memory access reduction compared to Anchor

Sequence	Number of memory access (anchor)	Ratio of methods to Anchor	
		ES-FSME	Proposed
Traffic	184,351,636,740	77.0%	34.0%
PeopleOnStreet	207,496,611,324	80.4%	52.3%
Nebuta	432,922,079,168	84.6%	60.1%
SteamLocomotive	376,764,899,636	78.8%	34.9%
Kimono	148,696,019,376	82.2%	39.4%
ParkScene	148,448,001,888	82.2%	36.8%
Cactus	318,644,670,708	74.2%	34.1%
BasketballDrive	324,831,935,148	82.0%	41.7%
BQTerrace	378,595,765,812	79.1%	33.2%
BasketballDrill	63,997,628,568	75.5%	40.5%
BQMall	76,474,547,735	79.0%	39.8%
PartyScene	64,938,861,600	81.1%	45.6%
RaceHorses	40,566,948,564	83.4%	55.7%
BasketballPass	16,119,769,992	80.2%	49.8%
BQSquare	18,246,323,189	81.5%	37.7%
BlowingBubbles	15,395,813,544	79.2%	42.4%
RaceHorses	10,065,074,376	84.0%	58.8%
FourPeople	140,265,184,752	69.9%	28.2%
Johnny	166,655,462,068	70.6%	27.2%
KristenAndSara	139,696,778,652	72.7%	28.5%
Average	**163,658,700,642**	**78.88%**	**41.04%**

As shown in Table 4.17, the combined method that integrated two ME skipping strategies reduced ME-related time significantly. To be specific, the combined method reduced overall ME time to 56.6% and total encoding time to 69.4% on average, while BD-rate of luma component was dropped only a 1% loss on average. The best case can be discovered at *BasketballDrill* sequence, showing a 49% decrease in ME time. As a result, it is demonstrated that combining two ME skipping strategies can reduce HEVC encoding time significantly while not losing substantial coding loss in all color components. Thus, it would be useful to integrate the proposed two strategies for low-complexity HEVC encoder to satisfy video encoding constraints such as real-time and low-power encoding.

Table 4.17—Comparison results of BD-rate, ME time, and total encoding time for the combined proposed methods under LD case

Sequence	Y	U	V	T_{ME}	T_{total}
Kimono	0.4%	1.0%	0.7%	55%	68%
BasketballDrive	0.9%	1.3%	0.8%	58%	69%
BasketballDrill	1.0%	0.9%	0.9%	49%	64%
BQMall	1.3%	1.2%	1.7%	52%	65%
PartyScene	1.7%	1.9%	1.5%	60%	73%
RaceHorses	0.8%	1.1%	1.3%	65%	76%
Average	**1.0%**	**1.2%**	**1.2%**	**56.6%**	**69.4%**

Note: For the average value of time results, geometric mean is used, and arithmetic mean is used for others.

CHAPTER 5
CONCLUSION

In this dissertation, two efficient ME skipping strategies are presented to help developing low-complexity HEVC encoder. First of all, the problem of the BME process in HEVC encoding was analyzed in view of computational complexity. Accordingly, an efficient BME decision method was proposed, which conditionally performed BME based on motion information (MV accuracy and PU partition). Evaluated in terms of encoding time, the number of function calls, and the number of memory access, the proposed BME skipping method has demonstrated high efficiency in reducing encoding complexity, and significantly outperformed existing methods in reducing the BME (BIME, BHME, and BQME) and the associated BMC complexity. Second of all, the complexity problem of the SME process in HEVC encoding was analyzed, and an efficient SME decision method that enabled skipping redundant SME process including the associated IF processes was proposed. Experimental results demonstrated the efficiency of the proposed SME skipping method by reducing significant encoding complexity while sustaining coding efficiency. Moreover, the performance of the proposed method outperformed existing methods in reducing the SME clearly.

The contribution of the dissertation includes in brief three parts as follows: 1) of BME, the time complexity and the memory complexity are analyzed mathematically, 2) the memory complexity was significantly reduced, and 3) the strategies can simply applied to HEVC encoder applications. Firstly, the importance of the mathematical analysis of computational complexity should be highlighted. By formulating the complexity of BME using Big O notation, it becomes easier to predict the complexity of BME so that the associated parameters such as search range could be properly decided for a certain target complexity of HEVC. Secondly, the proposed two ME skipping strategies reduced the memory complexity of BME and of SME by more than half of the anchor. The memory complexity is critical in hardware platform as well as in mobile applications, which cannot avoid memory restriction and battery constraint. As high video resolution is required nowadays and will be in near future, reducing memory access in HEVC encoding will be one of main research topics.

Finally, the advantage of the proposed strategies is a portability to similar video encoders since the proposed methods only have simple condition statements. Thus, the strategies could simply ported into such HEVC-based encoders that needs accelerating encoding. For example, one of representative open source HEVC codec, x265 can be modified by simply adding the conditions proposed in this dissertation. Furthermore, the proposed strategies can be simply applied to FVC since FVC also takes quadtree coding structure—note that there exist additional tree (binary tree) in FVC—and almost same BME and SME processes during encoding. As the encoding complexity of FVC has been highlighted due to the roughly 20-40 times more complex than that of HEVC, the proposed strategies can contribute on relieving the complexity of FVC as well.

In conclusion, the proposed two ME skipping strategies should be considered in low-complexity video encoders for real-time broadcasting, surveillance, and many mobile devices that need video conferencing and/or user-generated content. In addition, the proposed strategies could help future research related to decreasing ME complexity of HEVC as well as of FVC and other video encoders that equips similar bi-prediction technique or sub-pixel ME technique.

BIBLIOGRAPHY

[1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuits Syst. Video Technol.*, 13(7), 560-576 (2003).

[2] J. Vanne, M. Viitanen, T. D. Hämäläinen, and A. Hallapuro, "Comparative Rate-Distortion-Complexity Analysis of HEVC and AVC Video Codecs," *IEEE Trans. Circuits Syst. Video Technol.*, 22(12), 1885-1898 (2012).

[3] G. J. Sullivan, J. –R. Ohm, W. –J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.*, 22(12), 1649-1668 (2012).

[4] F. Bossen, B. Bross, K. Sühring, and D. Flynn, "HEVC Complexity and Implementation Analysis," *IEEE Trans. Circuits Syst. Video Technol.*, 22(12), 1685-1696 (2012).

[5] S. Hong, D. Yang, B. Park, H. Kim, and S. Yu, "PU-type-dependent reference frame selection method for HEVC," *Signal Image Video Process.*, 11(1), 25-32, (2017).

[6] P. Nalluri, L. N. Alves, and A. Navarro, "Complexity reduction methods for fast motion estimation in HEVC," *Signal Process.-Image Commun.*, 39, 280-292, (2015).

[7] S. Park and E. S. Jang, "Comments on 'Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder'," *IEEE Trans. Broadcast.*, accepted.

[8] S. Zhu and K. –K, Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Process.*, 9(2), 287-290 (2000).

[9] S. –H. Yang, J. –Z. Jiang, and H. –J. Yang, "Fast motion estimation for HEVC with directional search," *Electron. Lett.*, 50(9), 673-675 (2014).

[10] N. Hu and E. -H. Yang, "Fast motion estimation based on confidence interval," *IEEE Trans. Circuits Syst. Video Technol.*, 24(9), 1310–1322 (2014).

[11] C. E. Rhee, K. Lee, T. S. Kim, and H. –J. Lee, “A survey of fast mode decision algorithms for inter-prediction and their applications to high efficiency video coding,” *IEEE Trans. Consumer Electron.*, 58(4), 1375-1383, (2012).

[12] J. Vanne, M. Viitanen, T. and D. Hamalainen, “Efficient mode decision schemes for HEVC inter-prediction,” *IEEE Trans. Circuits Syst. Video Technol.*, 24(9) (2014)

[13] A. Lee, D. Jun, J. Kim, J. S. Choi, and J. Kim, “Efficient inter-prediction mode decision method for fast motion estimation in high efficiency video coding,” *ETRI J.*, 36(4) (2014).

[14] G. Correa, P. Assuncao, L. Agostini, and L. A. Silva Cruz, “Complexity control of high efficiency video encoders for power-constrained devices,” *IEEE Trans. Consumer Electron.*, 57(4), 1866-1874 (2011).

[15] K. Choi, E. S. Jang, “Fast coding unit decision method based on coding tree pruning for high efficiency video coding,” *Opt. Eng.*, 51(3), 030502 (2012).

[16] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, “An effective CU size decision method for HEVC encoders,” *IEEE Trans. Multimedia*, 15(2), 465-470 (2013).

[17] S. Park, S. Lee, E. S. Jang, D. Jun, and J. –W. Kang, “Efficient biprediction decision scheme for fast high efficiency video coding encoding,” *J. Electron. Imaging.*, 25(6), 063007 (2016).

[18] C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. Sullivan, “High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Improved Encoder Description Update 7,” *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, Document JCTVC-Y1002, Chengdu (2016).

[19] M. Flierl and B. Girod, “Generalized B pictures and the draft H.264/AVC video-compression standard,” *IEEE Trans. Circuits Syst. Video Technol.*, 13(7), 587-597 (2003).

[20] C. E. Rhee, “Skipping prediction directions based on the cost relationship between multi-directional predictions for an HEVC encoder,” *IEICE Trans. Inf. Syst.*, E97-D(9), 2541-2544 (2014).

[21] C. E. Rhee and H. –J. Lee, “Early decision of prediction direction with hierarchical correlation for HEVC compression,” *IEICE Trans. Inf. Syst.*, E96-D(4), 972-975 (2013).

[22] K. –Y. Kim, H. –Y. Kim, J. –S. Choi, and G. –H. Park, "MC complexity reduction for generalized P and B pictures in HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, 24(10), 1723-1728 (2014).

[23] L. Du, Z. Liu, T. Ilkenaga, and D. Wang, "Linear adaptive search range model for uni-prediction and motion analysis for bi-prediction in HEVC," *Proc. 2014 IEEE international conference on image processing (ICIP),* (2014).

[24] J. Kim, D. Jun, S. Jeong, S. Cho, J. S. Choi, J. Kim, and C. Ahn, "An SAD-Based selective bi-prediction method for fast motion estimation in high efficiency video coding," *ETRI J.*, 34(5), 753-758 (2012).

[25] S. G. Blasi, I. Zupancic, and E. Izquierdo, "Adaptive Precision Motion Estimation for HEVC Coding," in *Proc. 31th Picture Coding Symposium (PCS),* Cairns (2015).

[26] H. Lv, R. Wang, X. Xie, H. Jia, and W. Gao, "A comparison of fractional-pel interpolation filters in HEVC and H.264/AVC," in *Proc. Visual Communications and Image Processing (VCIP),* San Diego (2012).

[27] K. Choi and E. S. Jang, "Data reuse-based fast subpixel motion estimation for high efficiency video coding," *Opt. Eng.*, 53(6) (2014).

[28] Y. Li, Z. Liu, X. Ji, and D. Wang, "HEVC Fast FME Algorithm using IME RD-Costs based Error Surface Fitting Scheme," in *Proc. Visual Communications and Image Processing (VCIP),* Chengdu (2016).

[29] R. Fan, Y. Zhang, B. Li, and G. Wang, "Multidirectional parabolic prediction-based interpolation-free sub-pixel motion estimation," *Signal Process.-Image Commun.*, 53, 123-134 (2017).

[30] W. Dai, O. C. Au, W. Zhu, W. Hu, P. Wan, and J. Li, "A robust interpolation-free approach for sub-pixel accuracy motion estimation," in *Proc. IEEE International Conference on Image Processing (ICIP),* Melbourne (2013).

[31] F. Bossen, "Common test conditions and software reference configurations," *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,* Document JCTVC-L1100, Geneva (2013).

[32] A. Alshin and E. Alshina, "CE3: DCT derived interpolation filter test by Samsung," *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, Document JCTVC-F247, Torino, (2011).

[33] ITU-T, *Rec. H.265 (04/15), Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of Moving Video, High Efficiency Video Coding*, [Online] https://www.itu.int/rec/T-REC-H.265-201504-S/en.

[34] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *ITU-T SG16/Q6 VCEG*, Document VCEG-M33, Texas (2001).

[35] P. Nalluri, L. N. Alves, and A. Navarro, "Complexity reduction methods for fast motion estimation in HEVC," *Signal Process.-Image Commun.*, 39, 280-292 (2015).

[36] Z. Pan, J. Lei, Y. Zhang, X. Sun, and S. Kwong, "Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder," *IEEE Trans. Broadcast.*, 63, 675-684 (2016).

ACKNOWLEDGEMENT

The end point which seemed very near, but not as easy as I thought, can be approached finally. Now I am facing joy and my heart throbbing to write a next sentence of my life, but I also remember there exist lots of difficulties and lessons. The achievement of mine by far, acknowledging that it is impossible by myself with my strength and my wisdom, I would like to express my deep appreciation and indebtedness through these pages.

"Here is a trustworthy saying that deserves full acceptance:
Christ Jesus came into the world to save sinners—of whom I am the worst.
But for that very reason I was shown mercy so that in me, the worst of sinners,
Christ Jesus might display his immense patience as an example
for those who would believe in him and receive eternal life."
1 Timothy 1:15-16 (NIV)

First of all, I thank to God who shone light and hope, has led my life, and will lead. Before knowing God, I was negative on my life and pessimistic on the world. However, with God, I can enjoy and have endless energy to live. In addition, God let me know many things that corrects the center of my mind through Bible and through people and environment around me. He made me study one important thing on the way of PhD, modesty, so that I could have a mind to study continuously. When I felt difficulty, he made me listen and see awesome and beautiful words.

Second of all, I thank to my advisor Prof. Euee S. Jang. For six and half years of my graduate school life, he advised me of not only academic lessons, but also priceless values in life. Moreover, he accepted the deficiency of mine, was patient and encouraged me. Thanks to his kind consideration, I could sincerely and energetically walk on the way of PhD. With coauthoring more than 10 international journal/conference papers and more than 50 standardization contribution documents, I could hear lots of practical advices. Learning logical mind and understanding the view of readers will be substantially helpful in various situations of writing and presenting.

Third of all, I thank to dissertation committee members—Prof. Jong-Il Park, Prof. Taesoo Kwon, Prof. Ronggang Wang, and Dr. Kiho Choi—who shared detailed advices, deep insights, and warm encouragements. It would be a just simple paper without their helps, but I can increase the depth of the dissertation.

I thank to people of Digital Media Lab in Hanyang University. There are lots of memory by sharing time in the lab with them, enjoying with them and learning from them, which should be carved in my heart to remember. Before becoming member of the lab, I had lots of deficiency and flaw, but they have understood, advised and helped me. I would like to strength this relationship and help them as much as possible.

I thank to my family. Particularly, I'd like to thank to my lovely wife, Eunji Kim. She truthfully loved, waited, and encouraged me whenever I am exhausted or frustrated. Thanks to her, I could be patient and enjoy the life of graduate school which seems quite long. Also, I could finalize my PhD way sincerely and vigorously. It should be noted that the food my wife cooked is always very delicious, which is mysterious and amazing (because the term "always" includes even her first try of new food).

I also thank to my mother and father. For long times, they loved me and supported me. I think that the grace of parents is very deep and wide. To my only one brother, I also want to say my appreciation. Acknowledging my busy life, he came to my university many times to encourage me and pray for me.

Finally, I thank to my friends, colleagues, and church people. I appreciate their encouragement and prayers. Even a simple word, it supported me. Since the name of people are many, I will contact each of them privately in near future.

LIST OF PUBLICATIONS

International Journals

[1] **S. Park,** S. Lee, E. S. Jang, D. Jun, and J. –W. Kang, “Efficient biprediction decision scheme for fast high efficiency video coding encoding,” *Journal of Electronic Imaging*, vol. 25, no. 6, Nov. 2016.

[2] R. Wang, T. Huang, **S. Park**, J. –G. Kim, E. S. Jang, C. Reader, and W. Gao, “The MPEG Internet Video-Coding Standard,” *IEEE Signal Processing Magazine*, vol. 33, no. 5, Sep. 2016.

[3] **S. Park**, K. Choi, and E. S. Jang, “Zero coefficient-aware fast butterfly-based inverse discrete cosine transform algorithm,” *IET Image Processing*, vol. 10, no. 2, Feb. 2016.

[4] **S. Park** and E. S. Jang, “Comments on ‘Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder’,” *IEEE Transactions on Broadcasting*, accepted.

International Conferences

[1] **S. Park** and E. S. Jang, “An efficient motion estimation method for QTBT structure in JVET future video coding,” *Proc. Data Compression Conference (DCC)*, 2017. [Poster]

[2] **S. Park**, H. Xu, and E. S. Jang, “Efficient Frame-By-Frame QP Assignment Method for Internet Video Coding,” *Proc. IEEE 5th Global Conference on Consumer Electronics (GCCE)*, Oct. 2016. [Oral]

[3] **S. Park** and E. S. Jang, “Efficient GOP-Level QP Assignment for HEVC,” *Proc. International Workshop on Advanced Image Technology (IWAIT)* 2016, 2016. [Oral]

[4] **S. Park,** H. Xu, and E. S. Jang, “Complexity Evaluation of Intra Prediction Mode in HEVC Decoder,” *Proc. International Workshop on Advanced Image Technology (IWAIT) 2016*, 2016. [Oral]

[5] R. G. Wang, G. Li, **S. Park**, J. Kim, T. Huang, E. S. Jang, and W. Gao, “Overview of MPEG internet video coding,” *Proc. SPIE,* Sep. 2015. [Oral]

[6] **S. Park**, S. Lee, H. Xu, and E. S. Jang, "Temporal correlation-based fast encoding algorithm in HEVC intra frame coding," *Proc. IEEE 5th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, Sep. 2015. [Oral]

[7] **S. Park** and E. S. Jang, "Objective and subjective evaluation of MPEG internet video coding," *Proc. 2015 IEEE International Conference on Consumer Electronics (ICCE)*, Jan. 2015. [Oral]

[8] J. Y. Choi, **S. Park**, and E. S. Jang, "A case study of bitstream syntax reconfiguration for MPEG Internet video coding," *Proc. 2014 4th IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, Sep. 2014. [Oral]

[9] **S. Park**, K. Choi, G. Noh, and E. S. Jang, "Frame-based adaptive selection of ALF for fast HEVC decoding," *Proc. 2012 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, Jun. 2012. [Oral]

[10] **S. Park**, K. Choi, and E. S. Jang, "CU depth-based ALF decision for fast HEVC encoding," *Proc. 2012 IEEE 16th International Symposium on Consumer Electronics (ISCE)*, Jun. 2012. [Oral]

Domestic Journals

[1] **박상효**, 장의선, "저지연 HEVC 부호화기를 위한 효율적인 프레임별 양자화 파라미터 할당 방법", *방송공학회논문지*, 제 21 권 제 3 호, 2016 년 5 월.

[2] 이승호, **박상효**, 장의선, "Rough Mode Decision 과 Most Probable Mode 에 기반을 둔 HEVC 고속 인트라 예측 모드 결정 방법", *방송공학회논문지*, 제 19 권 제 2 호, 2014 년 3 월.

[3] 박수빈, 최기호, **박상효**, 장의선, "고속 HEVC 부호화를 위한 효율적인 PU 레벨 움직임예측 병렬화 구현", *방송공학회논문지*, 제 18 권 제 2 호, 2013 년 3 월.

Domestic Conferences

[1] 이승호, **박상효**, 장의선, "색차 신호 DC 계수 조정을 통한 영상 부호화 성능 향상 기법", *2016 년 한국통신학회 동계학술대회*, 2016.

[2] 김명균, 이승호, **박상효**, 장의선, "향상된 MPEG IVC 모션 벡터 비트 비용 계산 방법", *2015 년 제 28 회 신호처리합동학술대회*, 2015.

[3] **박상효**, 이승호, 김명균, 장의선, “Type-1 동영상 표준을 위한 MPEG IVC 인코더의 압축 성능 평가”, *2015 년 한국방송공학회 하계학술대회*, 2015.

[4] 이승호, **박상효**, 장의선, “MPEG IVC 를 위한 효율적인 매크로블록 타입 문맥 구조”, *한국통신학회 2014 년도 추계종합학술발표회*, 2013.

[5] 이승호, **박상효**, 장의선, “HEVC 에서 Rough mode decision 과 Most probable mode 에 기반을 둔 고속 인트라 예측 모드 결정 방법”, *2013 년 한국방송공학회 추계학술대회*, 2013..

[6] 홍승보, 최기호, **박상효**, 장의선, “HEVC 고속 복호화를 위한 SIMD 기반의 IDCT 병렬 프로그래밍 기법”, *2013 년 한국방송공학회 하계학술대회*, 2013.

[7] 김규동, 최기호, **박상효**, 장의선, “동영상 압축 표준의 특허 DB 분석 및 동향에 관한 연구”, *한국통신학회 2013 년도 하계종합학술발표회*, 2013

[8] 박수빈, 최기호, **박상효**, 장의선, “고속 HEVC 부호화를 위한 효율적인 PU 레벨 움직임예측 병렬화 구현 기법”, *한국방송공학회 2012 년도 추계학술대회*, 2012.

List of Patents

Patents Granted

Korean

[1] 부호화/복호화 장치 및 그 방법과 이를 구현하기 위한 프로그램이 기록된 기록매체(Patent No. 10-1619306, 2016 년 5 월)

[2] 부호화 유닛 정보를 이용한 적응적 루프 필터의 절차 간소화(Patent No. 10-1337990, 2013 년 12 월)

[3] 적응적 루프 필터의 프레임 기반 적응적 선택 방법(Patent No. 10-1307431, 2013 년 9 월)

Patents Applied

Korean

[1] 고속 영상 부호화 방법 및 장치(App. No. 2017-0058195)

[2] 시간 상관도에 기반한 고속 영상 부호화 방법(App. No. 2015-0123524)

[3] 움직임 벡터 비트 비용의 사용 방법 및 부호화 장치(App. No. 2015-0144071)

LIST OF CONTRIBUTIONS

Joint Video Exploration Team (JVET) of ISO/IEC and ITU-T

[1] **S. Park**, H. Xu, and E. S. Jang, “Evaluation report of SJTU test sequences for future video coding standardization”, JVET-B0061, Feb. 2016.

Joint Collaborative Team on Video Coding (JCT-VC) of ISO/IEC and ITU-T

[1] K. Choi, **S. –H. Park**, and E. S. Jang, “Coding tree pruning based CU early termination”, JCTVC-F092, Jul. 2011

[2] K. Choi, G. Noh, **S. Park**, and E. S. Jang, “Cross-check of Early skip detection for HEVC (JCTVC-G543)”, JCTVC-G794, Nov. 2011

ISO/IEC JTC1/SC29/WG11 (MPEG)

[1] M. Park, K. Choi, J. Y. Choi, **S. Park**, and E. S. Jang, “IVC-EE: Reference frame improvement”, M30406, Jul. 2013

[2] M. Park, K. Choi, **S. Park**, and E. S. Jang, “Cross-check of M30200”, M30432, Jul. 2013

[3] **S. Park**, K. Choi, M. Park, and E. S. Jang, “Information of IVC anchors for subjective assessment test”, M31626, Oct. 2013

[4] M. Park, K. Choi, J. Y. Choi, **S. Park**, and E. S. Jang, “IVC-EE: Cross check of m31516 (Improvement of intra prediction with multiple modes for Internet Video Coding)”, M31559, Oct. 2013

[5] K. Choi, M. Park, **S. Park**, and E. S. Jang, “IVC-EE: Cross check of m31515 (Smoothed Planar Intra prediction for Internet Video Coding)”, M31558, Oct. 2013

[6] M. Park, K. Choi, J. Y. Choi, **S. Park**, and E. S. Jang, “IVC-EE: Reference frame improvement”, M31557, Oct. 2013

[7] **S. Park**, K. Choi, J. Y. Choi, M. Park, and E. S. Jang, “Comments on VCB WD”, M32443, Jan. 2014

[8] M. Park, K. Choi, J. Y. Choi, **S. Park**, and E. S. Jang, “Deblocking Method for Internet Video Coding”, M32445, Jan. 2014

[9] K. Choi, M. Park, **S. Park**, and E. S. Jang, “Cross-check of m32144”, M32455, Jan. 2014

[10] S. Lee, **S. Park**, M. Park, and E. S. Jang, “Report on the possible optimization of VCB syntax structure”, M32963, Mar. 2014

[11] **S. Park**, J. Y. Choi, and E. S. Jang, “Comments on WD3 of VCB”, M32964, Mar. 2014

[12] **S. Park**, J. Y. Choi, S. Lee, M. K. Kim, M. Park, and E. S. Jang, “Updated Internet Video Coding Test Model (ITM) v 8.0”, M33144, Mar. 2014

[13] M. K. Kim, M. Park, **S. Park**, J. Y. Choi, and E. S. Jang, “Extended Deblocking Method for Internet Video Coding”, M33146, Mar. 2014

[14] M. Park, M. K. Kim, **S. Park**, and E. S. Jang, “Cross-check of Improvement on Non-Reference P frame coding for IVC (M32727)”, M33148, Mar. 2014

[15] J. Y. Choi, **S. Park**, and E. S. Jang, “Proposal of 16x16 intra prediction for P and B frames in IVC”, M34369, Jul. 2014

[16] S. Lee, **S. Park**, and E. S. Jang, “Efficient macroblock syntax structure for IVC”, M34375, Jul. 2014

[17] **S. Park**, S. Lee, J. Y. Choi, and E. S. Jang, “Comments on WD2 of IVC”, M34457, Jul. 2014

[18] M. K. Kim, **S. Park**, and E. S. Jang, “Cross-check for Non-Reference P Frame Coding in Multiple Reference Frame of Internet Video Coding (M34108)”, M34502, Jul. 2014

[19] M. K. Kim, **S. Park**, and E. S. Jang, “Cross-check for Lagrange Multiplier Selection for Non-Reference P Frames in Internet Video Coding (M34109)”, M34503, Jul. 2014

[20] S. Lee, **S. Park**, and E. S. Jang, “Cross-check for De-blocking method for internet video coding (M34230)”, M34504, Jul. 2014

[21] **S. Park**, R. Wang, and E. S. Jang, “Objective evaluation of IVC compared to WebVC”, Jul. 2014

[22] **S. Park**, S. Lee, M. K. Kim, and E. S. Jang, “Performance evaluation of ITM 10.0 over WebVC according to the VCC condition”, M35003, Oct. 2014

[23] J. Y. Choi, **S. Park**, and E. S. Jang, “Comments on WD3 of Internet Video Coding (IVC)”, M35004, Oct. 2014

[24] J. Y. Choi, **S. Park**, and E. S. Jang, “Cross-check of m34973 (Improvement on Lagrange Multiplier Selection for Internet Video Coding)”, M35352, Oct. 2014

[25] J. Y. Choi, **S. Park**, and E. S. Jang, “Cross-check of m35054 (De-blocking improvement for internet video coding)”, M35353, Oct. 2014

[26] S. Lee, **S. Park**, and E. S. Jang, “Bug report on ME initialization in internet video coding (IVC) encoder”, M35746, Feb. 2015

[27] S. Lee, **S. Park**, and E. S. Jang, “Enhanced encoding technique on the quantized coefficients of chroma for internet video coding (IVC)”, M35747, Feb. 2015

[28] S. Lee, **S. Park**, and E. S. Jang, “Improved set of reference frames for internet video coding (IVC)”, M35748, Feb. 2015

[29] **S. Park**, S. Lee, M. K. Kim, and E. S. Jang, “Performance evaluation of IVC encoder (ITM 11) according to the VCC condition”, M35749, Feb. 2015

[30] R. Wang, T. Huang, **S. Park**, E. S. Jang, and J. –G. Kim, “Prior art analysis on the adopted IVC technologies”, M35990, Feb. 2015

[31] S. Lee, **S. Park**, and E. S. Jang, “Cross-check of M35786 (Subjective quality improvement for internet video coding)”, M36044, Feb. 2015

[32] S. Lee, **S. Park**, and E. S. Jang, “Chroma enhancement technique on the intra predicted block for IVC encoding”, M36710, Jun. 2015

[33] **S. Park** and E. S. Jang, “Cross-check of m36681 (Performance evaluation of internet video coding)”, M36781, Jun. 2015

[34] **S. Park**, R. Wang, J. –G. Kim, ““Internet Video Coding Test Model (ITM) v 13.0”, N15429, Jun. 2015

[35] H. Kim, **S. Park**, and E. S. Jang, “Draft Text of ISO/IEC 23002-5:2013/PDAM3 Reference Software for Parser Instantiation from BSD”, M37103, Oct. 2015

[36] M. K. Kim, S. Lee, **S. Park**, and E. S. Jang, “MV bit cost calculation method for IVC encoder”, M37355, Oct. 2015

[37] M. K. Kim, S. Lee, **S. Park**, and E. S. Jang, "Reference frame distance-weighted ME for IVC encoder", M37356, Oct. 2015

[38] **S. Park**, and E. S. Jang, "Performance evaluation of 4K test sequences for future video coding", m37357, Oct. 2015

[39] H. Kim, **S. Park**, and E. S. Jang, "Proposal for New Amendment of ISO/IEC 23002-4:2014 for the Parser Instantiation from BSD", M37428, Oct. 2015

[40] M. K. Kim, S. Lee, **S. Park**, H. Xu, and E. S. Jang, "Combined method of m37355 and m37356 in IVC encoder", M37470, Oct. 2015

[41] **S. Park**, E. S. Hang, R. Wang, and J. –G. Kim, "Collection of information related to adopted IVC technologies", N15762, Oct. 2015

[42] **S. Park**, R. Wang, and J. –G. Kim, "Internet Video Coding Test Model (ITM) v 14.0", N15760, Oct. 2015

[43] **S. Park**, S. Kuk, H. Xu, Y. Park, and E. S. Jang, "Report on the decoding complexity of IVC", M37783, Feb. 2016

[44] **S. Park**, Y. Park, and E. S. Jang, "Information on how to improve Text of DIS 14496-33 IVC", Feb. 2016

[45] **S. Park**, W. C. Oh, and E. S. Jang, "Cross-check of m37799 (Extension of Prediction Modes in Chroma Intra Coding for Internet Video Coding)", Feb. 2016

[46] **S. Park**, R. Wang. and J. –G. Kim, "Internet Video Coding Test Model (ITM) v 14.1", N16038, Feb. 2016

[47] R. Wang, **S. Park**, T. Huang, E. S. Jang, and J. –G. Kim, "Study Text of ISO/IEC DIS 14496-33 Internet Video Coding", N16034, Feb. 2016.

국문요지

비디오 압축에서의 움직임 예측(Motion estimation) 기술은 시간적으로 상관이 있는 프레임 간에서 블록을 예측하는 기술로서, 압축률에서와 계산 복잡도 측면에서 결정적 영향을 끼친다. 이러한 비디오 부호화기의 복잡도를 줄이기 위한 연구의 중심이 되어 온 것은 고속 움직임 예측 알고리즘들이며, 이는 기존 압축률을 최대한 보존하는 것을 함께 고려해왔다. 그러나 이러한 움직임 예측의 복잡도는 HEVC 코덱에서 훨씬 많이 증가했는데, 이는 AVC/H.264 코덱보다 두 배의 압축률을 만족시키기 위해 다양한 변화들을 수용했기 때문이다. 일례로, HEVC 코덱은 AVC/H.264 에 비해 더 많은 움직임 파티션(Motion partition), 더 많은 움직임 방향(Motion direction), 그리고 더 정교한 움직임 정확도(Motion Accuracy)를 채용하였다. 게다가, 최근에는 비디오 해상도가 급격히 증가한 것을 고려할 때, 움직임 예측의 복잡도는 분명 해결해야 할 문제라 할 수 있겠다.

HEVC 의 복잡도를 경감시키기 위해서, 본 학위논문에서는 상기 서술한 HEVC 의 다양한 변화들을 고려하는 두 가지 움직임 예측 생략 전략을 제안한다. 하나는 양방향 움직임 예측(Bidirectional motion estimation)을 생략하는 전략으로써, 대부분의 압축률에 비효율적인 검색 지점을 피하고자 한다. 특히, 단방향 움직임 예측(Unidirectional motion estimation)에는 없는 추가적인 보간 절차(interpolation process)로 인한 높은 계산 복잡도를 가지고 있는 양방향 움직임을 위한 검색지점을 피하는 것을 목표로 한다. 본 제안 방법은 예측 유닛(Prediction unit, HEVC 의 특별한 특징 중 하나)의 문맥에 담긴 통계적 상관관계를 활용하여, 어떤 양방향 움직임 예측이 치명적인지 아닌지를 결정한다. 두 번째 제안 방법은 부화소 움직임 예측(Sub-pixel motion estimation)의 생략 전략이다. 이는 1/2 픽셀과 1/4

픽셀 움직임들의 비효율적인 검색 지점을 피하고 이와 관련된 보간 절차를 생략하는 것을 목표로 한다. 예측 유닛의 상관관계를 활용하면, 이 두 번째 제안 전략은 어떤 부화소 움직임 예측이 치명먹인지 아닌지를 단방향 움직임 예측과 양방향 움직임 예측의 부호화 과정 중에서 판별할 수 있다. 이러한 두 번째 제안 전략에서는 부화소 움직임 예측 중에 추가적으로 확인하는 과정—현 프레임으로부터 먼 거리의 참조 프레임(Reference frame)을 추가로 확인—의 결과에 따라 부화소 움직임 예측의 복잡도를 더 줄인다. 이러한 두 가지의 제안된 전략을 활용하면, HEVC 의 저복잡도 부호화가 가능하고, 압축 성능도 합리적으로 유지할 수 있다고 본다.

본 움직임 예측 생략 전략의 성능을 증명하기 위하여, 다음의 복잡도 측정방식을 주의깊게 선택하였다. 이는 관련 함수 호출 횟수, 실행 시간, 메모리 접근 횟수이다. 실험결과를 통해서, 제안된 양방향 움직임 예측 생략 전략이 양방향 예측 시간 복잡도를 평균적으로 30%로 줄일 수 있음을 보였으며, 또한 다른 기존 방법들과 비교할 때, 부호화 시간이나 함수 호출 횟수나 메모리 접근 측면에서도 모두 제안 기술이 더 우수함을 보였다. 게다가, 두 번째로 제안된 부화소 움직임 예측 생략 전략 또한 움직임 예측 시간 복잡도를 평균적으로 임의접근 방식(Random access case)에서 51%로, 저지연 방식(Low-delay case)에서 52%로 줄였으며, 기존 기술보다 우수함을 보였다. 그러므로, 본 학위논문에서 제안한 두 움직임 예측 전략은 저복잡도 HEVC 부호화기에 고려되어야 한다고 본다. 특히 실시간 방송 시스템이나 감시 카메라 어플리케이션 및 (전력 공급이 제한되는)모바일 비디오 부호화 장치를 위한 저복잡도 HEVC 부호화기를 구현함에 있어서 본 제안 기술들은 심사숙고 될 필요가 있다고 결론 짓는다.

감사의 글

가까울 듯 했지만 막상 쉽지 않았던 박사의 길에 마침표를 찍게 되었습니다. 제 인생의 다음 문장을 쓰고자 하는 기쁨과 설렘이 있지만, 그간 많은 어려움과 교훈이 있음을 기억합니다. 지금까지의 성취는 저 혼자만의 힘과 지혜로 불가능함을 알기에, 부족하나마 지면을 빌어 여러 분들께 감사의 말씀을 올리고자 합니다.

"미쁘다 모든 사람이 받을 만한 이 말이여 그리스도 예수께서 죄인을 구원하시려고 세상에 임하셨다 하였도다 죄인 중에 내가 괴수니라 그러나 내가 긍휼을 입은 까닭은 예수 그리스도께서 내게 먼저 일체 오래 참으심을 보이사 후에 주를 믿어 영생 얻는 자들에게 본이 되게 하려 하심이라",

디모데전서 1 장 15-16 절(개역개정).

먼저, 지금까지의 삶을 이끌어 주시고 앞으로의 삶도 함께해 주실, 나에게 빛과 소망을 주신 하나님께 감사드립니다. 하나님을 알기 전에는 삶에 대해 부정적이고 세계에 대해 비관적이었지만 알고 난 이후, 삶의 기쁨을 누릴 수 있게 되었고 끊임없는 활력이 생기게 되었습니다. 또한 나의 마음의 중심이 올곧게 설 수 있도록 성경을 통해서 많은 것을 알게 하셨고, 주위의 많은 사람과 환경을 통해서 배우게 하셨습니다. 박사과정의 길에서 중요한 것 중 하나인 겸손에 대해 계속 배우고 느끼게 하셨기에, 지속적으로 공부하려는 태도를 유지할 수 있었던 것 같습니다. 그리고 힘들 때마다 멋지고 아름다운 언어를 보고 듣게 하셨습니다.

다음으로, 저의 지도교수이신 장의선 교수님께 감사인사를 드립니다. 저의 대학원 6 년 반의 시간동안 학술적 가르침 뿐 아니라 인생의 여러 소중한 가치에 대한 주옥같은 말씀을 남겨주신 것에 감사드립니다. 또한, 부족한 저의 모습을 받아주심은 물론, 때로는 기다려주시고 때로는 격려해주신 덕분에 더욱 힘차게 그리고 열심으로 학업의 길을 걸어갈 수 있었습니다. 교수님과 10 편 이상의 국제 저널/컨퍼런스 논문을 함께 저술하며, 그리고 50 편 이상의 국제 표준화 기고서를 저술하며 많은 실질적 가르침을 배울 수 있었습니다. 논리적 사고를 배우고 독자의 입장을 배우게 된 것은 앞으로의 학술활동에서뿐 아니라 글을 작성하고 발표하는 다양한 상황에서 큰 도움이 될 것입니다.

셋째로, 바쁘신 일정 중에서도 본 논문이 완성될 수 있도록 세밀한 조언과 깊은 통찰력, 따뜻한 격려를 나누어주시며 논문을 심사해주신 박종일 교수님, 권태수 교수님, Ronggang Wang 교수님, 최기호 박사님께도 감사를 드립니다. 자칫 한 편의 단순한 논문으로만 끝날 뻔 했던 본 학위논문의 깊이를 더할 수 있게 되었습니다.

디지털미디어연구실의 선후배님들에게도 감사함을 전합니다. 연구실에서의 많은 시간을 함께 보내며 느꼈던 여러가지 기쁨과 교훈을 제 마음에 깊이 간직하도록 하겠습니다. 연구실에 들어가기 전까지, 들어가 있는 중에도 참 많이 어리석었고 흠이 많았지만, 이를 이해해 주시고 도움을 주셔서 감사드립니다. 서로에게 큰 힘이 되는 관계로 계속해서 인연의 끈을 놓치지 않도록 노력하겠습니다.

다음으로 제 가족들에게 감사인사를 드리고 싶습니다. 특별히 사랑스러운 아내, 김은지에게 감사를 드립니다. 지치거나 걱정이 가득할 때 진실하게 사랑해주고 기다려주며 격려해준 아내 덕분에 길어보이는 대학원 생활을 인내하고 즐길 수 있었습니다. 아내가 있었기에, 박사의 길을 성실하고 힘차게

마칠 수 있었던 것 같습니다. 여러분이 알아야 할 것 중 하나는, 우리 아내가 만들어주는 음식은 항상 맛있다는 점입니다. (이게 신기하고 놀라운 점은) “항상”이라는 것은 설사 그것이 첫 시도로 요리해보는 음식에도 해당함을 의미합니다.

어머니, 아버지께도 감사를 드립니다. 저를 오랜 기간 사랑해주시고 지원해주신 것에 감사를 드립니다. 부모님의 은혜는 깊고 넓은 것 같습니다. 오랜 기간, 끊임없는 관심과 격려에 감사드립니다. 또한 나의 하나뿐인 동생에게도 고맙다는 인사를 하고 싶습니다. 나의 바쁨을 알고, 학교까지 여러번 찾아와서 응원해주고 기도해주었습니다.

마지막으로 나의 친구, 동료, 그리고 교회 지인들에게도 감사를 드립니다. 박사학위를 위한 기도와 격려에 감사드립니다. 작은 한마디도 제게는 큰 힘이 되었습니다. 너무 많으니 개별적으로 연락드리겠습니다.

연구 윤리 서약서

본인은 한양대학교 대학원생으로서 이 학위논문 작성 과정에서 다음과 같이 연구 윤리의 기본 원칙을 준수하였음을 서약합니다.

첫째, 지도교수의 지도를 받아 정직하고 엄정한 연구를 수행하여 학위논문을 작성한다.

둘째, 논문 작성시 위조, 변조, 표절 등 학문적 진실성을 훼손하는 어떤 연구 부정행위도 하지 않는다.

셋째, 논문 작성시 논문유사도 검증시스템 "카피킬러"등을 거쳐야 한다.

2017년06월22일

학위명 : 박사

학과 : 컴퓨터·소프트웨어학과

지도교수 : 장의선

성명 : 박상효 (서명)

한 양 대 학 교 대 학 원 장 귀 하

Declaration of Ethical Conduct in Research

I, as a graduate student of Hanyang University, hereby declare that I have abided by the following Code of Research Ethics while writing this dissertation thesis, during my degree program.

"First, I have strived to be honest in my conduct, to produce valid and reliable research conforming with the guidance of my thesis supervisor, and I affirm that my thesis contains honest, fair and reasonable conclusions based on my own careful research under the guidance of my thesis supervisor.

Second, I have not committed any acts that may discredit or damage the credibility of my research. These include, but are not limited to : falsification, distortion of research findings or plagiarism.

Third, I need to go through with Copykiller Program(Internet-based Plagiarism-prevention service) before submitting a thesis."

JUNE 22, 2017

Degree : Doctor

Department : DEPARTMENT OF COMPUTER SCIENCE

Thesis Supervisor : Euee S. Jang

Name : Park Sang-hyo (Signature)