MOTION ESTIMATION TECHNIQUES BY EXPLOITING MOTION HISTORY AND DEPTH MAPS IN VIDEO CODING

LEE TSZ KWAN

Ph.D

The Hong Kong Polytechnic University

2017

The Hong Kong Polytechnic University

Department of Electronic and Information Engineering

Motion Estimation Techniques by Exploiting Motion History and Depth Maps in Video Coding

Lee Tsz Kwan

A thesis submitted in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

July 2016

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

_____(Signed)

LEE Tsz Kwan (Name of student)

"You cannot do anything

about the length of your life,

but you can do something

about its **width** and **depth**."

- Henry Louis Mencken -

Abstract

Video coding with low-delay hierarchical prediction structure is essentially introduced for real time video applications. This structure is currently adopted in various emerging video coding standards including MPEG4-Part 10 (H.264), high efficiency video coding (HEVC) and multi-view video coding (MVC). The only disadvantage of this structure is the requirement of motion estimation in distant reference frames. For maintaining high coding efficiency, a large search range in motion estimation can be employed in distant reference pictures. However, computational complexity will thus be increased dramatically.

In addition to the hierarchical prediction structure, the vision of the latest HEVC video coding standard provides a more flexible framework by confronting the tradeoff between coding efficiency and computational complexity. It is able to gain coding efficiency up to 50% bitrate reduction comparatively to H.264. By this advantage, HEVC is the emerging standard in the industry for providing video streaming applications and online TV advancements. The achievement of HEVC is obtained by introducing the new coding quad-tree structure on block partitions in motion estimation. However, this flexibility of recursive block partitioning for coded video quality induces heavy computations in an HEVC encoder. Therefore, this work investigates computational complexity reduction algorithms in emerging video coding standards.

The work on this thesis then contrives a number of fast algorithms for motion estimation. The adoption of motion vector composition (MV composition) for a fast motion estimation scheme in a low-delay hierarchical P-frame structure is firstly

proposed. It expedites the motion estimation process for distant reference frames in the hierarchical P structure. In addition, a vector selection algorithm is tailor-made with the proposed hierarchical P coding scheme to further improve the coding efficiency. Simulation results show that the proposed scheme can deliver a remarkable complexity savings and coding efficiency improvement on coding a frame in low temporal layers of the hierarchical P structure.

The rest of this work proposes to perform motion locus prediction before motion estimation. By this motion locus prediction, a suitable search range can be adjusted adaptively for motion estimation. Thanks to the rapid development of MVC and 3D videos, the state-of-the-art 3D coding framework provides multi-view plus depth video (MVD) in which the depth map is additional information to be encoded in the coded bitstreams. Depth maps record the distances of various objects in the scene from a viewpoint. With the depth maps from MVD sequences, we reveal the depth variation and the spatial correlation between blocks as well as the temporal correlation between the depth maps and the motion in texture, motion locus perdition can be achieved for speeding up the texture coding in an HEVC encoder. The depth information brings new room for designing an efficient adaptive search range (ASR) algorithm in HEVC. Simulation results show that the proposed ASR algorithms can offer a significant complexity reduction with negligible loss of coded video quality.

Publications arising from the thesis

Journal Papers

1. **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Adaptive Search Range for HEVC Motion Estimation based on Depth Information," *IEEE Transactions on Circuits and Systems for Video Technology*. (Accepted on 11 June, 2016)

2. **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Adaptive Search Range by Neighbouring Depth Intensity Weighted Sum for HEVC Texture Coding," *Electronics Letters*, vol. 52, no. 12, pp. 1018-1020, June 2016.

3. **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Motion Estimation in Low-delay Hierarchical P-frame Coding Using Motion Vector Composition," *Journal of Visual Communication and Image Representation*, 24 (8) (2013) 1243-1251.

Conference Papers

4. **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Depth-based Adaptive Search Range Algorithm for Motion Estimation in HEVC," in *Proceedings of International Conference on Digital Signal Processing (DSP 2014)*, Hong Kong, Aug. 2014, pp.919-923.

5. **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Motion Vector Composition in Low-delay Hierarchical P-Frame Coding," in *Proceedings of IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP 2013)*, Beijing, China, July 2013, pp. 551-555.

Acknowledgements

I would like to express my profound gratitude to my chief supervisor, Dr. Yui-Lam Chan and my co-supervisor, Prof. Wan-Chi Siu for their continual support, supervision and valuable suggestions throughout this research work. Their moral support and continuous guidance enabled me to complete my work successfully.

This work would not have been possible without the support and assistance of many individuals. I would like to take this opportunity to thank all members of the Centre for Signal Processing in The Hong Kong Polytechnic University. My interactive discussions with the group members have helped to formulate and develop this research work. It has been a wonderful time to me working with them.

Meanwhile, I am greatly appreciative of the Department of Electronic and Information Engineering (EIE) and The Hong Kong Polytechnic University (PolyU) for providing me a comfortable working environment and for their financial support to my research work. The work described in this research studies is partially supported by the Centre for Signal Processing, EIE, PolyU and a grant from the Research Grants Council of the HKSAR, China. Besides, I would like to express my sincere gratitude to Li Po Chun Charitable Trust Fund Scholarship. I am honored to be a recipient of this award during my studies. I acknowledge the research studentships and scholarship provided by the parties stated above.

I am indebted to my family, especially my husband Ben, for their love, encouragement and support throughout my life. Without their understanding and patience, it is impossible for me to complete this research study.

Contents

List of Figures

List of Tables

List of Abbreviations

2D	Two-dimensional
3D	Three-dimensional
AMCVS	Adaptive Multiple-Candidate Vector Selection
AMVP	Advanced Motion Vector Predictor
ASR	Adaptive Search Range
B-frame	Bi-directional Predicted Frame
BD	Bjontegaard
CB	Coding Block
CTU	Coding Tree Unit
CU	Coding Unit
DCT	Discrete Cosine Transform
DIBR	Depth-Image-Based Rendering
DMRMap	Depth/Motion Relationship Map
DVD	Digital Versatile Disc
E-FDVS	Enhanced Forward Dominant Vector Selection
FDVS	Forward Dominant Vector Selection
FS	Full-search

GOP	Group of Pictures
H.264	H.264/Advanced Video Coding (MPEG-4 Part 10)
HB	Hierarchical B
HD	High Definition
HDTV	High Definition Television
HEVC	High Efficiency Video Coding
HP	Hierarchical P
I-frame	Intra-coded Frame
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
ITU	International Telecommunication Union
ITU-T	Telecommunication Standardization Sector of ITU
JSVM	H.264 Joint Scalable Video Model
LAMASR	Linear Adaptive Model for- Adaptive Search Range Algorithm
LMF	Large Motion Frame
LSMF	Large and Small Motion Frame Algorithm
MC	Motion Compensation
MCVS	Multiple-candidate Vector Selection
ME	Motion Estimation
MLELD	Maximum Likelihood Estimation- Laplace Distribution Algorithm
MP3	MPEG-1 or MPEG-2 Audio Layer III

MPEG	Moving Picture Experts Group
MV	Motion Vector
MV Composition	Motion Vector Composition
MVC	Multi-View Coding
MVD	Multi-View Video plus Depth
MVP	Motion Vector Predictor
NDIWS	Neighboring Depth Intensity Weighted Sum
OTT	Over-the-top
P-frame	Predicted Frame
PB	Prediction Block
PSNR	Peak Signal-to-Noise Ratio
PU	Prediction Unit
QP	Quantization Parameter
RD	Rate-distortion
RDO	Rate-distortion Optimization
SAD	Sum of Absolute Difference
SCU	Smallest CU
SMF	Small Motion Frame
SR	Search Range
TMV	True Motion Vector
TU	Transform Unit
TV	Television
TZS	Test Zone Search

UHD	Ultra High Definition
VCD	Compact Disc digital video
VCEG	ITU-T Video Coding Experts Group
VLC	Variable Length Coding
VOD	Video-on-demand

Chapter 1

Introduction

1.1 Overview

Digital video becomes one of the essential principal media for daily content establishment and distribution. The proliferation of digital video is due to the demands on video streaming services, and popular usage of computers and mobile devices [1]. As an enormous amount of video data is being generated, transmitted, and stored all over the world, video compression is necessary for reducing the data rate. Digital video coding adopting motion estimation (ME) is initially designed as a crucial technique for the data compression [2]. It has revolutionized the broadcast and data storage industries over several decades. From broadcasting video contents coded by MPEG-2 [3], the coding techniques has been evolved to MPEG-4 [4, 5], H.264 [6, 7], and the most advanced High Efficiency Video Coding (HEVC) [8, 9]. All of them facilitate the convenience of exchanging and retrieving digital video. They are commonly adopted in video delivery applications such as high-definition

television (HDTV), video-on-demand, Internet video, and video sharing [10, 11]. Among the standards, H.264 has been the most popular formats and was jointly developed by the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) [12]. Applications such as Blu-ray players, online video streaming, web-based software applications, satellite television services, cable television services, and real-time video conferencing are commonly manipulated by H.264 over one decade [13]. Nowadays, the HEVC standard is the most recent joint work from the VCEG and MPEG [14, 15]. HEVC has been developing in the industry to have a bitrate reduction of 50% at similar perceptual quality compared to H.264 [37]. It facilitates video streaming applications and online TV advancements including over-the-top (OTT) delivery [16]. All these developments on emerging standards aim for supporting the near future video viewing experience such as ultra high definition (UHD) and 3D video viewing [17].

To support manifold video applications, the flexibility of the hierarchical coding structure in H.264 is firstly evoked and extended to HEVC later on [18, 19]. However, it becomes a challenge in video coding systems, of which high temporal dependency between frames may be destroyed. As a result, additional computations are required for motion locus prediction in or before ME in order to maintain the coding accuracy, which is not favor in real time applications [20–22]. Furthermore, HEVC which is the successor of H.264 achieves almost 50% of bitrate reduction with the similar level of video quality by its flexibility of block partitioning [23]. However, this flexibility of block partitioning in HEVC causes extra computational complexity in ME which aims for the prediction accuracy. It causes burden to an HEVC encoder. The research work in this thesis aims to reduce the computational complexity

in video coding by some enhanced motion locus prediction methods for the emerging coding standards in the delay-sensitive scenario. The development of emerging coding standards and new coding features are introduced in Section 1.2. As a wide variety on applications with video contents is desired, Section 1.3 discusses the proliferation of flexibility on prediction structure and partitioning in digital video coding standards. The challenges by implementing flexibility in the emerging video coding standards, H.264 [6, 7] and HEVC [8, 9, 14], are then briefed in Section 1.4. Furthermore, the motivations and objectives of this research studies for faster motion locus prediction are mentioned in Section 1.5. Finally, the organization of this thesis is presented in Section 1.6.

1.2 Development of Emerging Coding Standards and their New Coding Features

The flexible choice of arbitrary coding structure, which was impossible in previous video coding standards including H.261 [24], H.263 [25], MPEG-2 [26], and MPEG-4 visual [27], is allowed since H.264 [18, 19]. The increased flexibility in H.264 permits any frame to be acted as a reference frame for the arbitrary coding structure. It is formally named as hierarchical prediction structure in H.264 [28–30]. This hierarchical structure provides flexibility and improves coding efficiency but causing much burden in encoders due to extra computations and memory requirements [20, 21], as illustrated in the block diagram of Figure 1.1.

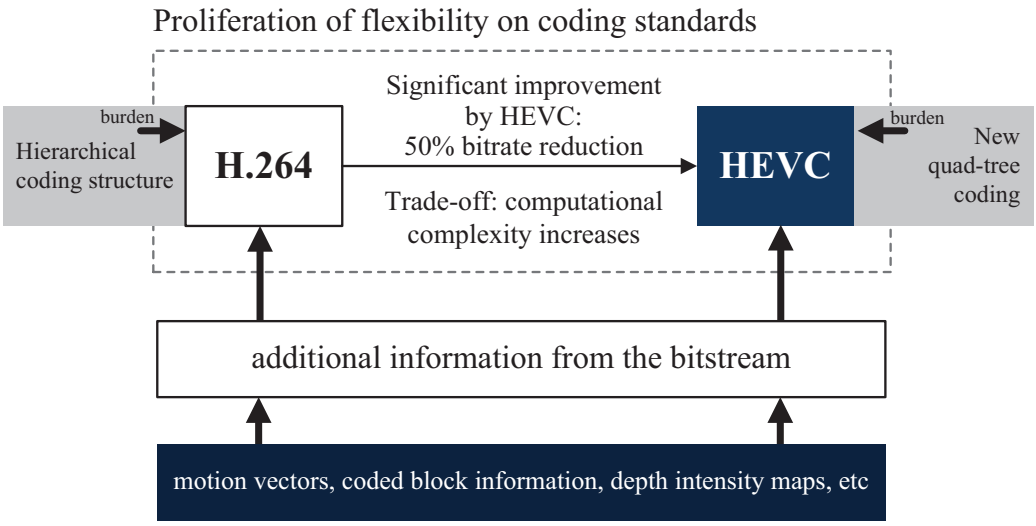


Figure 1.1: Illustration of relationship between emerging coding standards and new coding features.

The hierarchical prediction structure is also widely adopted in the emerging standards, HEVC [31] and mult-view coding (MVC) [32, 33]. Besides, HEVC also adopts quad-tree coding unit (CU) and prediction unit (PU) partitioning which are recursive representations [23]. With the increased flexibility on block partitioning, HEVC offers the best coding efficiency up to 50% bitrate reduction compared to H.264 [34] while the subjective quality of video encoded by HEVC has been proved to be maintained [35]. However, the new coding tools of HEVC lead to an increase in computational complexity [34], as illustrated in the block diagram of Figure 1.1. HEVC induces heavy coding computations caused by its new coding tools during ME for the above achievements [31, 36]. The Main profile of HEVC requires an encoder complexity ratio of 1.5$\times$ in average in hierarchical coding structure compared

to that in the High profile of H.264 [37]. The main reasons for the computation burden are from the flexible coding structures in frames and partitioning arrangement in a CU. It motivates us to reduce the complexity of the emerging coding standards.

Furthermore, with the proliferation of MVC and 3D video coding applications [38], texture plus depth or multi-view plus depth (MVD) is one of the efficient data representation formats in MVC and 3D video systems where virtual views generation is employed by a depth-image-based rendering (DIBR) technique [39, 40]. There are additional data to be encoded in the bitstreams for MVC and 3D displays which further increase the computational complexity and are not in favor of real-time applications [41, 42]. But, at the same time, we believe that they may provide some extra useful non-coded and coded information, such as coded motion vectors (MVs) and the depth intensity map for 3D videos as depicted in Figure 1.1, to expedite the video coding process in H.264 and HEVC. For instance, depth maps in 3D video provide extra characteristics on the distance of objects and movements in a scene from a viewpoint, which can be revealed to assist the coding process. Inheritance of coded parameters from texture to depth coding is commonly adopted [43, 44]. In our studies, reverse philosophy is applied. Depth information is utilized in speeding up the texture coding process.

1.3 Flexibility on Video Coding Structures

H.264 and HEVC are designed in response to the emerging needs for various video applications with high compression requirement. They are contrived to facilitate the

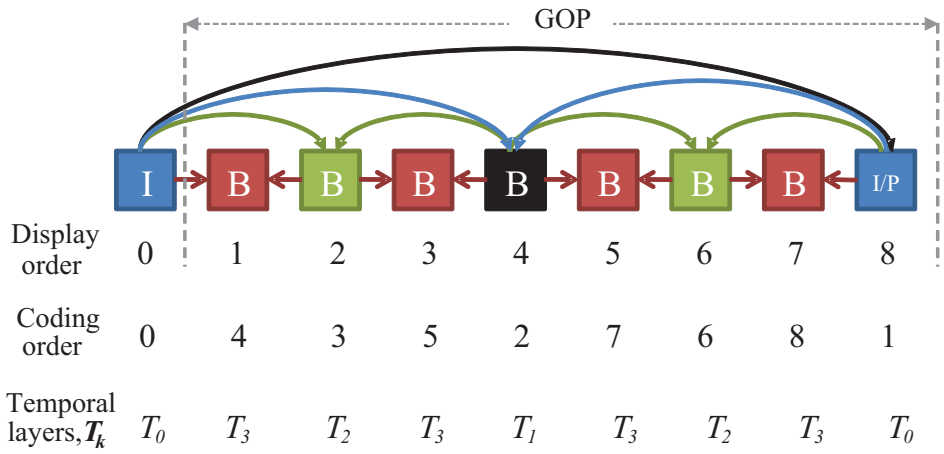


Figure 1.2: Illustration of typical HB coding structure with the GOP size of 8.

adoption of the encoded video representation in a flexible way for different scenarios. The new flexible encoding structures in H.264 and HEVC include hierarchical coding structure in which hierarchical B (HB) and hierarchical P (HP) structures are commonly used, quad-tree prediction structure and multi-view plus depth videos.

1.3.1 Hierarchical Coding Structure

In video coding standards, HB and HP structures are two common types of hierarchical prediction structures. A typical HB prediction structure [28, 29] is illustrated in Figure 1.2. In this figure, I/P-frames are key frames in the temporal base layer, denoted by T_0 which are firstly coded in a group of pictures (GOP). The non-key frames within the GOP are coded as B pictures. The B-frame in layer T_1 is coded with two reference frames in the lower temporal layer, T_0 [22]. After that, it becomes one of the reference frames to the frames in the next temporal layer, T_2 and so on. The HB structure applies bi-directional prediction which references from a future

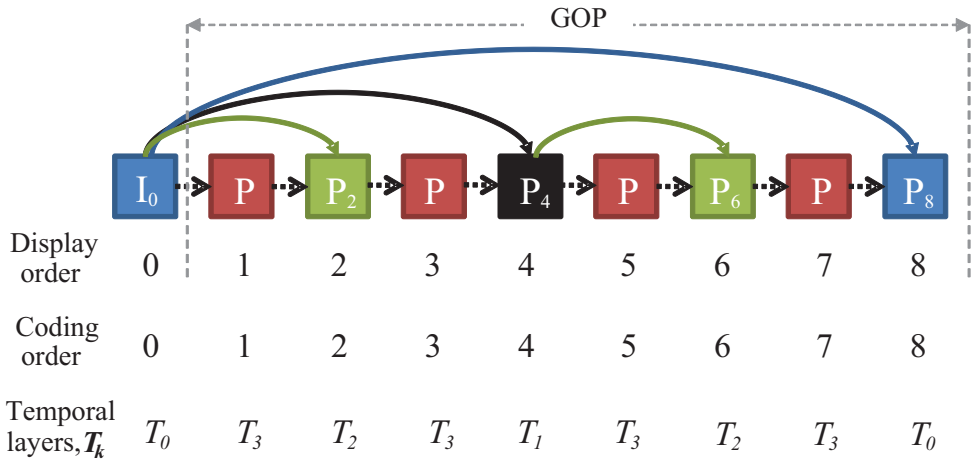


Figure 1.3: Illustration of typical HP coding structure with the GOP size of 8.

frame. As a result, the coding order for this HB structure has to be re-arranged such that future frames are coded before reference frames. This increases the associated coding delay and memory requirement [20, 21].

1.3.2 Temporal Hierarchical Coding Structure for Delay-Sensitive Applications

Constraining delay is of great importance for real-time applications such as video conferencing, live event broadcasting, and video surveillance [30, 45] in which the long-delay HB prediction structure in Figure 1.2 is not desirable. As a consequence, B-frame is not supported in the Baseline Profile of H.264, which is targeted to ultra-low delay video coding applications. Besides, a list of conditions requiring low algorithmic delay is specified in the newest HEVC standard [46–48]. An important concern of video coding for real-time applications is to achieve low latency without reordering of pictures during displays. Therefore, the HB structure is not favorable

for these real-time applications. In order to satisfy the low delay constraint [45, 49], a HP structure, as shown in Figure 1.3, employing only P-frames has been designed [30, 45]. This HP structure provides the same degree of temporal scalability as the HB structure but it does not employ motion-compensated prediction from future frames [28,50]. A typical HP structure with 4 hierarchy/temporal layers is illustrated in Figure 1.3. The leading I-frame and the last P-frame are the key frames in the temporal base layer, T_0. In this figure, the longest arrow represents the prediction at T_0 from the first I-frame, I_0, to the eighth P-frame, P_8 . The fourth P-frame, P_4, is in T_1 while the second and sixth P-frames, P_2 and P_6 , respectively are in the temporal layer, T_2. The predictions for temporal layers, T_1 and T_2, are shown accordingly in solid lines with different prediction distances. The predictions in T_3 are showed by dotted arrows in Figure 1.3. The non-key frames within a GOP are coded as P frames, which are different from the HB structure [19]. Since no B-frames are inserted, the frames can be coded according to the display order, as stated in Figure 1.3. It implies that the HP structure is a low-delay coding structure and is very suitable for delay-sensitive applications. However, HP structure requires ME between the current frame and a large distant reference frame. This is one of the limitations in the emerging video coding standard which motivates our research work on efficient ME techniques.

1.3.3 Quad-tree Prediction Structure in HEVC

HEVC achieves the coding gain mainly from its adoption of more flexible block partitions. However, high computational complexity is induced [51–53] since a quad-

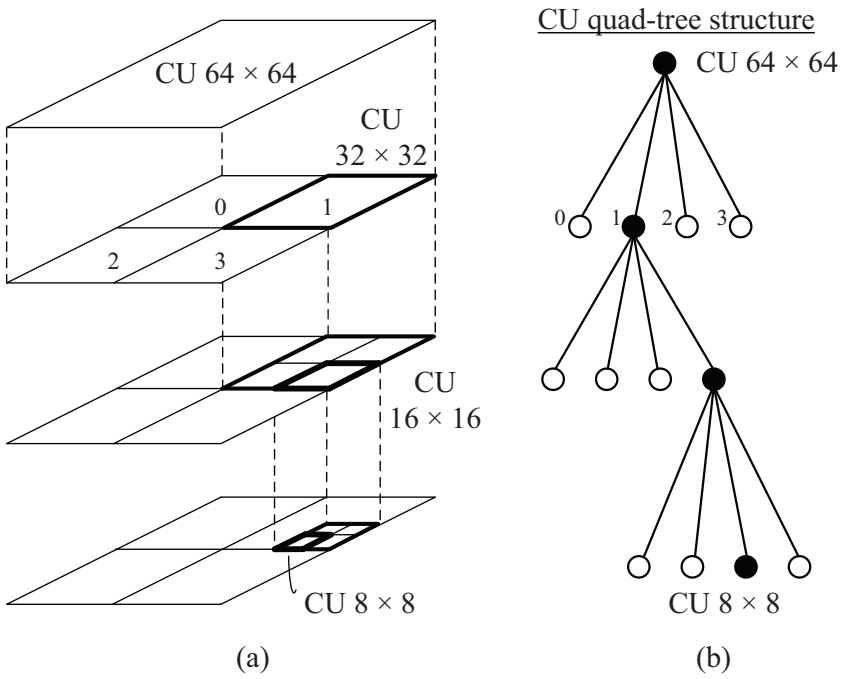


Figure 1.4: Flexible CU block partitioning in HEVC. (a) CU partition and its split. (b) CU quad-tree structure.

tree structure is applied for every prediction block where a recursive split of a CU is conducted for ME [54]. In the encoding process for a CU as shown in Figure 1.4, the largest size of a CU in 64×64 is the root of the split [55–57]. The CU will be split into four partitions, each with a size of 32×32 during ME. The optimal 32×32 CU (with partition index of 1) will be selected if it obtains the minimum rate-distortion (RD) cost among the 4 partitions and followed by another CU splitting process as depicted in Figure 1.4(a). The CU splitting process will be conducted until reaching the smallest size of the CU in 8×8. Finally, an optimal quad-tree structure will be formed as shown in Figure 1.4(b). The white dots are nodes as the non-split blocks while the black dots are the selected nodes to undergo the splitting process

(a) (b)

Figure 1.5: Color texture and its associated depth map for a frame. (a) Color texture in "Balloons". (b) Depth map in "Balloons".

in each CU layer. With this flexible block partitioning mechanism, inter prediction consumes about 60-70% of the whole encoding time [23, 34].

1.3.4 Multi-view plus Depth Videos

In recent years, we have witnessed the rapid development of 3D video technology. Among various 3D video representations, the multiview video plus depth (MVD) [39] is emerging as the most flexible format. The MVD includes both the color texture and the depth map of the captured scene. A texture frame of "Balloons" and its associated depth map are shown in Figure 1.5. In Figure 1.5(a), the objects in front are the bundle of balloons, and then the man holding a large balloon. The color texture stream captures both luminance and chrominance information of every pixel in the scenes while the depth map in Figure 1.5(b) records the distances of the objects associated with every pixel of the color textures. The lighter of the grayscale in the depth map, the closer of the object to the viewer. Therefore, the depth map reflects

the bundle of balloons in lighter grayscale than that of the man. At the decoder side, depth maps provide the flexibility that can be used to synthesize arbitrary numbers of extra views through Depth-Image-Based Rendering (DIBR) techniques [58]. It implies that the depth map for each frame is a piece of additional information to be encoded in the state-of-the-art 3D video encoder.

1.4 Problem Formulation from Flexibility Structure on Video Coding

Various flexible prediction structures in H.264 and HEVC result in increased computational complexity. This is absolutely not a favor for video coding in real time applications. Therefore, this thesis work starts investigations on the limitations due to increased flexibility in the prediction structures of both emerging H.264 and HEVC coding standards.

1.4.1 Limitations of Low-delay Hierarchical P Coding in H.264

In the HP structure mentioned in Section 1.3.2, encoding complexity is much higher than that of the classical IPPP structure. It is because ME over remote reference frames does occur in the base layer, T_0 in Figure 1.3. It causes rate-distortion (RD) performance deterioration. Such problem can be solved by using a larger search range (SR) in ME but the computational complexity will be increased significantly. Therefore, we propose to utilize the existing information provided in the bitstream to solve the complexity issue in video coding. In hopes of reducing the computa-

tional complexity in ME, MVs from previous frames are one kind of the existing information to exhibit the motion locus of the current block before ME for texture coding.

1.4.2 Limitations of Quad-tree Prediction in HEVC

With the implementation of the hierarchical coding structure [59] and the quad-tree block partitioning structure, HEVC requires higher computational complexity than previous coding standards [60]. With the recent exploration success of MVC and 3D video streams coded by HEVC, it is easily imagine that a geometric growth of the computations in ME will be induced because of the quad-tree block partitioning and more dependent bitstreams. A literature review in the aspect will be provided in Chapter 2. In order to reduce the computational complexity in ME of HEVC, estimating the motion locus as early as possible during the encoding process is an efficient research direction. Furthermore, in emerging standards such as HEVC and MVC, depth map information becomes an extra pieces of cue for the locus prediction for texture coding.

1.5 Motivation and Objectives

For computational complexity reduction in video coding, utilizing existing coded and non-coded information for reference in motion locus prediction during coding a bitstream is preferred. In this thesis, we are going to explore the available information in H.264 and HEVC for the enhanced motion locus prediction as depicted in Figure 1.6. Different existing information in H.264 and HEVC will be studied that

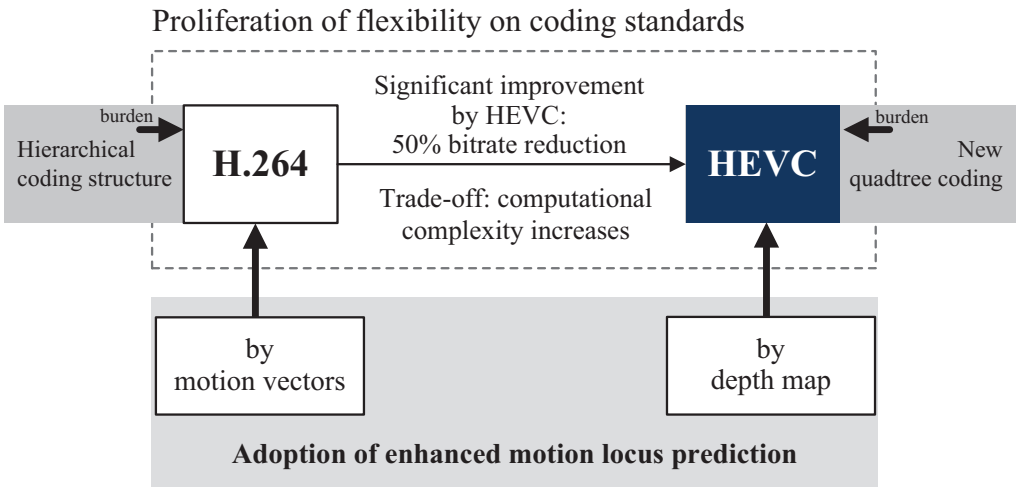


Figure 1.6: Illustration of the proposed enhanced motion locus prediction.

can be use to speed up the coding process. Therefore, the objective of this research work is to make use of the available information, motion history formed by MVs and depth maps respectively, in H.264 and HEVC to reduce computational complexity in ME for real-time applications. Firstly, we propose a new vector selection algorithm between reference frames in MV composition for HP coding structure. It results in accurate predicted MVs in large distant reference frames without much increased complexity. Secondly, depth intensity variance in depth maps are revealed for motions along z-direction and formulating the probable motion ranges for x- and y-direction. As a result, adaptive search range can be figured out such that complexity reduction can be achieved.

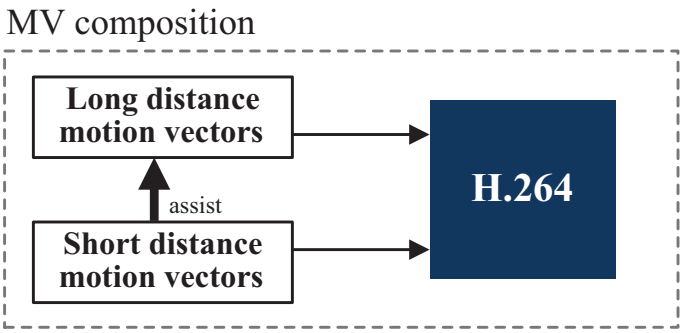


Figure 1.7: Illustration of the proposed MV composition for enhanced motion locus prediction in H.264.

1.5.1 MV Composition between Long Distance Frames

In the past, little research effort has been given to the HP structure, which requires ME for the current frame to a long distance reference frame. The simplest suggestion was enlarging the search range for ME but it will cause much more computational complexity. Our work proposes to adopt the technique of motion vector composition (MV composition) in coding videos with the HP structure. The MV components used in the proposed MV composition scheme are provided in the bitstream. In Figure 1.7, it shows that the existing information used for achieving the objective is the short distance MVs which are the MVs between consecutive frames. In addition to using our proposed vector selection algorithm for choosing the optimal short distance MVs, MV composition is conducted to form a long distance MV for HP coding. Without enlarging the search range for ME, our proposed MV composition successfully provides an accurate long distance MV and better RD performance. As a result, reasonable computational complexity is achievable in HP coding.

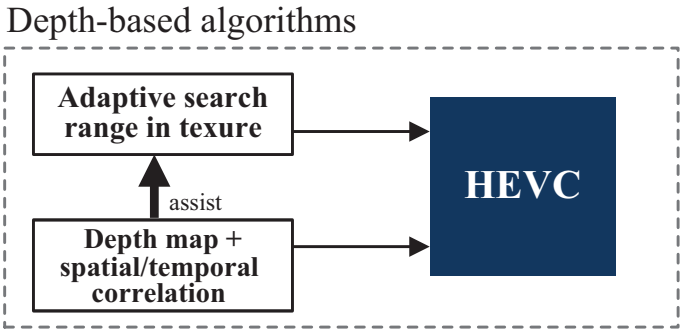


Figure 1.8: Illustration of the proposed adaptive depth-based weights for enhanced motion locus prediction in HEVC.

1.5.2 Spatial and Temporal Correlation and Depth Variations on Motion Locus

The state-of-the-art 3D coding framework contains texture plus depth map coding in which the depth map is an additional information to be included in the coded bitstreams for the viewing experience. Furthermore, depth maps contain intensity variations which record the distances of objects and movements in the scene from a viewpoint. Besides, the depth intensity is a piece of indicative information to decide how probable some blocks belong to the same object provided that blocks obtains same or very similar depth intensity values within a frame. Furthermore, it reveals characteristics of the movements in the frame and between frames to some extent. They are able to spot out the high activities regions across frames. In our work, depth maps with the high spatial and temporal correlation are explored to speed up HEVC texture coding by enhanced motion locus prediction as depicted in Figure 1.8. We observe that the depth intensity variance in depth maps between blocks and frames

reflecting the motion along z-direction reveals the probable motion locus ranges in x- and y-directions. It figures out object movements in texture streams. Therefore, this thesis aims to reduce the computational complexity of texture coding in an HEVC encoder by assigning a suitable search range in ME adaptively by exploiting motion history and depth maps. The depth intensity variance in depth maps are used to link up with the motion vectors from history in order to predict the motion ranges of the locus.

1.6 Organization of this Thesis

This thesis comprises six chapters. Prior to our description of the objectives and the main contributions in this thesis, Chapter 2 commences with a broad literature review of video coding techniques that are related to this work. The problems of flexible coding structure adoption in coded video due to the use of motion-compensated prediction in H.264 and HEVC are then addressed. The difficulties of extra computational complexity for the conventional ME in the emerging coding standards are issued in delay-sensitive applications. Previous schemes for tackling the intensive computational complexity in ME are introduced. All these motivate our research work on enhanced motion locus prediction in or before ME for alleviating the coding complexity.

In Chapter 3, a review of an MV composition technique in HP is presented. The usage of MV composition for complexity reduction in coding the HP structure is described. We discuss the issues in long distance motion locus prediction by MV composition in details. We argue that the conventional MV composition algorithms

suffer from inaccuracy of new composed MVs when the prediction between far away frames is required. By exploring the existing relevant motion information in the H.264 video, a new MV composition algorithm with the proposed vector selection method is suggested that could provide better performance for long distance motion locus prediction in the HP coding structure.

Chapter 4 initiates with an idea of an adaptive search range (ASR) adjustment for ME. It investigates the motion locus by the spatial correlation between neighboring blocks and their depth intensity variations. With this consideration on the depth intensity variation, a weighted motion locus will be established for reducing the search range in ME. As a result, computational complexity of HEVC can be reduced.

Chapter 5 extends the proposed ASR algorithm by considering the temporal correlation between the depth maps and motion in texture. It introduces how temporal motions in texture and depth intensity variation in depth map formulate the motion locus in details. A relationship map is then constructed for defining the proposed ASR. This chapter also reveals the influence of 3D-to-2D projection on motion activity on a 2D image plane. This projection factor is further proposed to be used for the ASR adjustment to achieve a more accurate motion locus. Comparisons and evaluations will be further conducted between works in Chapter 4 and Chapter 5.

Chapter 6 is devoted to conclusions of the work herein. We also summarize the contributions of this thesis in this chapter. Suggestions are also included for further research in this area.

Chapter 2

Literature Review

2.1 Background Research

The latest video coding standards such as H.264 [6] and High Efficiency Video Coding (HEVC) [14, 15] are basically developed for the convenience of storage and transmission. These standards aim at reducing the data size of video by adopting motion-compensated predictive coding in which only the residues between adjacent frames are stored instead of the frames themselves. Therefore, the data redundancy can be eliminated. However, spatial dependency among blocks and temporal dependency among frames will be established by motion-compensated predictive coding. Nowadays, the proliferation of delay sensitive video applications in high resolution videos, and multi-view and 3D videos leads to the demand on handling much more data volume in video coding [61]. Consequently, more computational effort is required in data redundancy elimination for compression provided that the spatial and temporal dependency will not be destroyed for the standards. It implies that algo-

rithms for reducing the computational complexity in video coding is desired. In recent years, various research efforts have been conducted for the purpose of complexity reduction.

To understand the rationale, objectives, and results of this thesis, background of this research is provided in this chapter. We are going to review the motion-compensated predictive techniques used in the emerging video coding standards in Section 2.2. Afterward, the limitations of using motion compensated predictive coding are addressed due to the spatial and temporal dependencies. In Section 2.3, a literature review for improving low-delay hierarchical P (HP) coding efficiency is presented. MV composition is one of the reliable solutions. Computational complexity increment of HEVC compared to H.264 is revealed in Section 2.4. As a consequence, a literature review for complexity reduction in HEVC will be provided. Finally, the chapter summary follows.

2.2 Digital Video Compression Fundamentals

Digital video is defined by video contents being stored and sent in digital form. However, it requires huge amount of data. Directly storing and transmitting digital video are not recommended because the bitrate is extremely high for most networks and storage devices to handle. The problem can be alleviated by compressing the video at the cost of degraded visual quality so that the format is more suitable for transmission and storage. Video compression techniques have been continually improving over decades in the video coding industry [62].

Different video coding standards are being developed to satisfy the requirements

of various applications with different requirements. It includes providing better picture quality, higher coding efficiency and higher error robustness. The Moving Picture Experts Group (MPEG) and Video Coding Experts Group (VCEG) are two major teams collaborating to develop digital video coding standards. The MPEG is a working group of the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). It aims at developing standards for compression, processing and representation of moving pictures and audio. MPEG-1 (ISO/IEC 11172) [63] and MPEG-2 (ISO/IEC 13818) standards [26] allow wide adoption of commercial products and services including DVD format, digital TV, and MP3 players. The VCEG is a working group of the International Telecommunication Union Telecommunication Standardization Sector (ITU-T). It develops a series of essential standards for video communications over telecommunication networks and computer networks. H.261 videoconferencing standard [64] was ratified and became a model for digital video coding. The following H.263 standard [65], informally known as H.263+ and H.263++, was created to improve the coding efficiency . One of the most commonly used video compression standards in the industry nowadays is H.264 which is also known as MPEG-4 Part 10 [5, 6]. It was jointly proposed by the ITU-T VCEG and the ISO/IEC MPEG in 2003. It aims at having an improved coding efficiency and provision of a video representation being friendly to network which addresses storage, broadcast and streaming applications. HEVC, the high efficiency video coding, is the most recent joint video standard development of the ITU-T VCEG and the ISO/IEC MPEG [8, 9]. It inherits coding features from H.264, and then introduces a larger block structure with the flexible sub-partitioning mechanism for higher coding efficiency. It has been reported that a bitrate reduction

of 50% can be achieved by HEVC as compared to H.264 with the similar subjective quality [37].

2.2.1 Hybrid Motion-compensated Predictive Coding

Compression of video data without noticeable degradation of the visual quality is possible since video consists of a high degree of redundancy. Digital video is regarded as a sequence of still pictures or frames. Unlike images, a video sequence contains temporal redundancy in addition to spatial redundancy. Spatial redundancy always exists within each frame due to the correlation between neighboring pixels. On the other hand, an object moving in front of a static background causes covered and uncovered regions in relatively small areas. Thus, temporally adjacent frames are often in high correlation in a video sequence, and this highly correlated frames of a video sequence results in temporal redundancy.

By removing the redundancy in a video sequence, it is possible to represent or compress the video data in a more compact form [66]. The higher the redundancy, the higher the achievable compression. Therefore, coding standards utilize the redundancy inherent in digital video data so as to achieve a impressive bitrate reduction.

2.2.1.1 Intra-frame Coding for Spatial Redundancy Elimination

In most still images and video frames, it is observed that the values of neighboring pixels are highly correlated. Therefore, the major techniques adopted to reduce the spatial redundancy are similar to that in image coding. Discrete cosine transform

8	16	19	22	26	27	29	34
16	16	23	24	27	29	34	37
20	23	25	27	29	34	34	38
22	22	26	27	30	34	37	40
21	25	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	27	29	34	38	46	56	69
27	29	35	29	46	56	69	82

Figure 2.1: A typical example of a quantization matrix.

(DCT) [67–71], quantization, and zig-zag scanning for entropy coding are used in H.264 and HEVC [72, 73]. DCT allows every block of pixels transformed into frequency domain to form a bundle of transform coefficients. The transformation packs the signal energy into a small number of coefficients, which can achieve efficient compression. HEVC supports various lengths of DCT in the sizes of 4, 8, 16, and 32 [74]. In natural images, if pixels in the block vary smoothly without any edges, several DCT coefficients could be enough to express the information of an entire block.

The DCT coefficients are set in a matrix according to their frequency. The most upper left corner coefficient in a block is the DC coefficient showing the average intensity value of the block. The rest are AC coefficients. They are arranged from low frequency to high frequency across the lower right corner. The DCT coefficients in the upper left corner, which stand for low frequency components, are more important than the high frequency coefficients in the lower right corner in a block. The reason is that the frequency response of human eyes tends to drop off as increasing

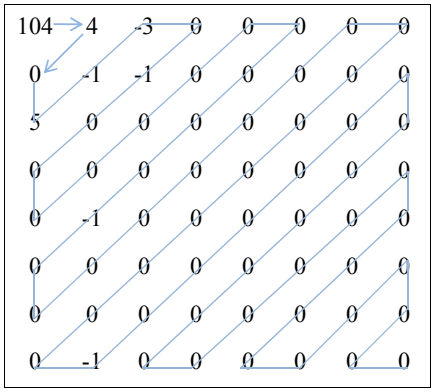


Figure 2.2: A DCT block after quantization and zig-zag scanning.

spatial frequency. Human eyes are more sensitive to tiny variation of intensity in slowly varying regions than in complex regions.

The output coefficients of DCT are entered to the quantization process [55, 75–77], which aims at expressing the sampled data at a finite number of levels. Each DCT coefficient is divided by the corresponding quantization factor at the corresponding position from a pre-defined 2D quantization matrix and is then rounded to the nearest integer to obtain the quantized DCT value. Figure 2.1 shows one example of a typical quantization matrix applied to a DCT block. The numerical values of the quantization matrix correspond to the relative importance of the DCT coefficients in terms of visual picture quality. This arrangement is due to the frequency response of human eyes. The higher the number, the less important the corresponding coefficients, as shown in Figure 2.1. The objective of this process is reducing the dimension of the DCT coefficients matrix and discarding those unimportant coefficients so that less bandwidth is required for transmission. Figure 2.2 shows a DCT block after quantization. Many quantized DCT block in the transform domain

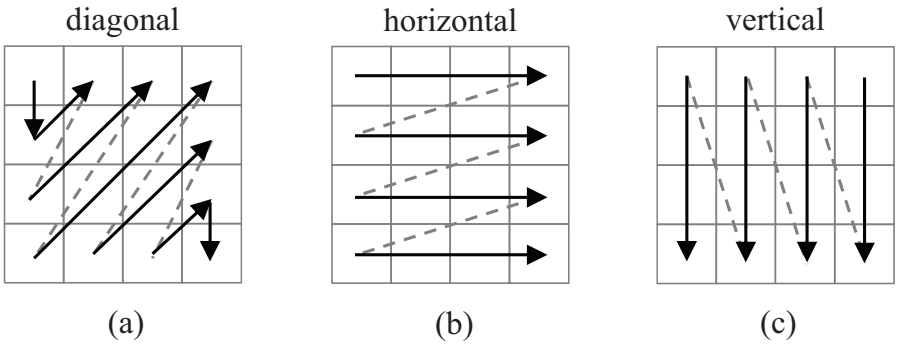


Figure 2.3: Scanning patterns for 4 $\times$4 a transform block in (a) diagonal, (b) vertical, and (c) horizontal order.

contain a large number of zeros, particularly in higher frequency regions [78]. The quantizer design for HEVC is similar to that of H.264 where the range of the quantization parameter (QP) is in the range from 0 to 51 [74].

After the quantization, zig-zag scanning [79, 80] is used to arrange the quantized coefficients into a 1-D array for entropy encoding for a transform block. The order of zig-zag scanning is depicted in Figure 2.2. This scanning order puts the low frequency coefficients prior to the high frequency coefficients [81]. Reordering in this zig-zag pattern tends to create long of runs of zero-value coefficients and this is beneficial to variable length coding (VLC) [80, 81]. Hence, higher coding efficiency is obtained since shorter code words is needed because large number of zero-value coefficients can be removed.

In the early development stage of HEVC, a diagonal scanning has been introduced to replace the conventional zig-zag scanning [82]. It is able to remove the context selection dependency on recently processed coefficients for all positions in the transform block without impact on coding efficiency. Later on, HEVC divides

a transform block into 4 $\times$ 4 sub-blocks and into categorized regions for parallelism [72]. In the final development stage, HEVC provides more scanning patterns for a 4 $\times$ 4 transform block according to different intra prediction modes used in a prediction block [83]. The scanning patterns include diagonal, horizontal and vertical orders for the adoption of the mode dependent coefficient scanning in HEVC. They are illustrated in a 4 $\times$ 4 transform block, respectively, in Figure 2.3(a), Figure 2.3(b), and Figure 2.3(c).

2.2.1.2 Inter-frame Coding for Temporal Redundancy Elimination

The previous section has briefly described the basic principles of intra-frame coding. In addition to intra-frame coding, inter-frame coding employs motion-compensated prediction to remove temporal redundancy between adjacent frames. Instead of directly transmitting pixels in a frame, pixels are predicted from a previously coded reference frame. This technique is known as motion estimation (ME) and motion compensation (MC), and is essentially the core of most hybrid video coding standards [84] like H.264 and HEVC.

In general, ME can improve the prediction accuracy between temporally adjacent frames by estimating any motion that has been taken place between the frame being encoded and its reference frame. ME can be performed with different granularities such as a pixel, a block, or an irregular region. Among them, the block-based approach is considered the most mature and practically useful one [85]. It makes an assumption that objects are in translational motion and the object displacement is constant within a small 2D block of pixels. The block-based approach does not aim at tackling problems related to rotational motion. For block-based ME, each video

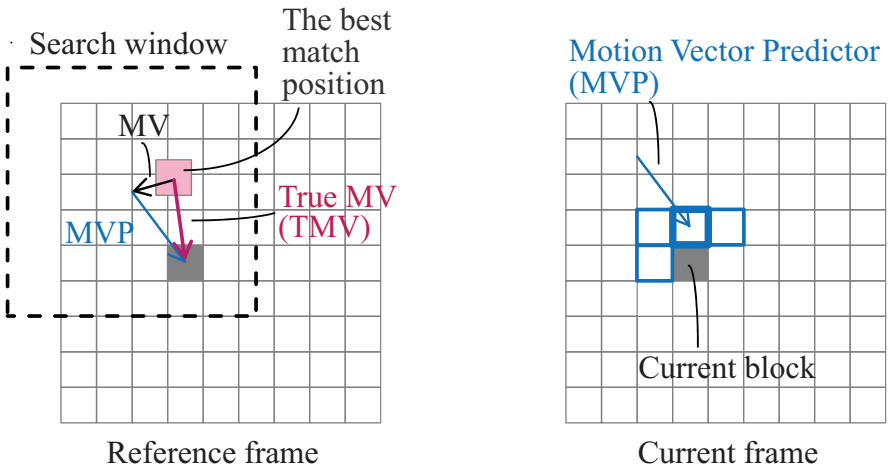


Figure 2.4: Block matching motion estimation and compensation.

frame is divided into non-overlapping prediction blocks for ME as illustrated in Figure 2.4. In H.264 standard, they are called macroblocks (MBs) and each MB is 16 $\times$ 16 pixels, which is the basic coding unit for ME. In HEVC, the blocks are called coding units (CUs) and the largest CU size is 64 $\times$ 64 pixels. Other CU sizes are 32 $\times$ 32 pixels, 16 $\times$ 16 pixels, and the smallest size is in 8 $\times$ 8 pixels. For the sake of simplicity, the basic coding unit is denoted as a block in this thesis for both H.264 and HEVC standards.

In Figure 2.4, the target block to be encoded in the current frame will undergo inter-frame prediction with reference to the previous coded frame. Within a predefined search range of which the center is aligned by a MV predictor (MVP) [86] as shown in Figure 2.4, all pixels of a block in the current frame are compared on a pixel by pixel basis on the corresponding block in the preceding reference frame. This is known as the matching criterion. There are many choices for the matching criteria. Among these criteria, the sum of absolute difference (SAD) is the most

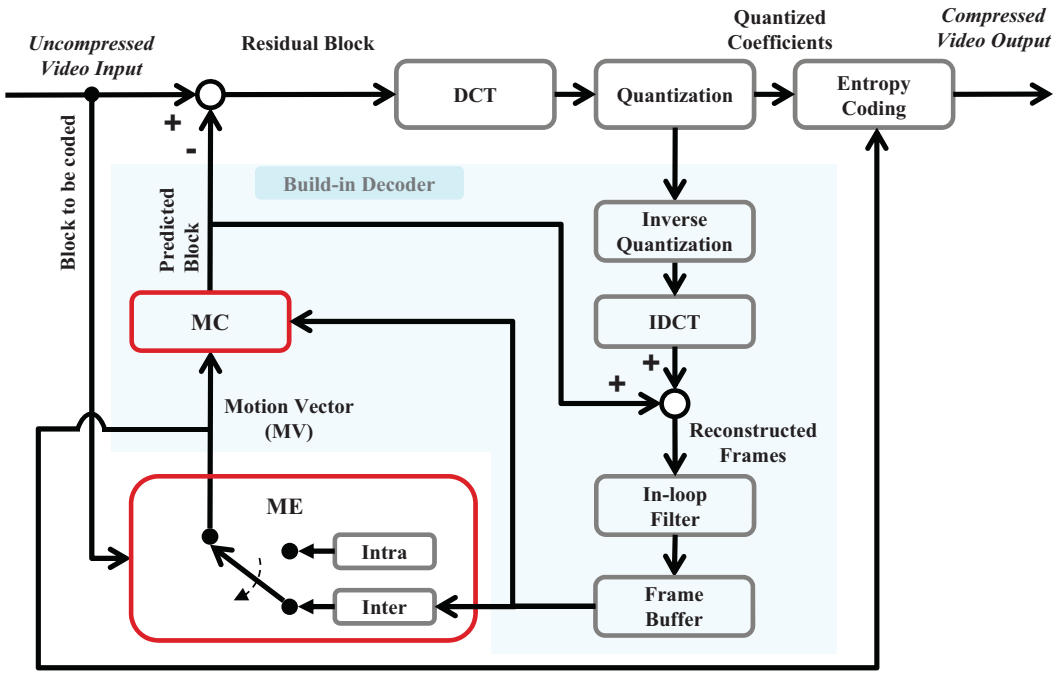


Figure 2.5: Block diagram of hybrid motion-compensated predictive encoder.

popular one. When the closest match is located, the relative displacement between the current block and the best matched block in the reference frame is encoded as the motion vector (MV) shown in Figure 2.4.

For ME, the amount of computation is proportional to the number of candidate blocks in the search range. The full-search (FS) algorithm evaluates the SAD at all possible locations of the search window to find the optimal MV. Hence, it is able to find the best-matched block which guarantees to obtain the minimal SAD. Nevertheless, ME is a computationally intensive task [85].

After ME, the predicted block is obtained from the reference frame based on

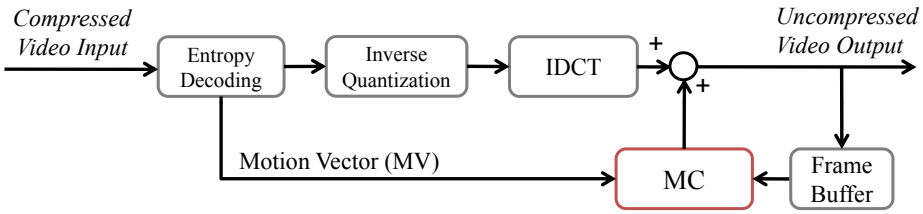


Figure 2.6: Block diagram of a decoder.

the MV using MC as shown in Figure 2.5. Then, the predicted block becomes the predictor for the current block and is subtracted from the current block to generate a residual block [66]. The more accurate the ME process is, the less entropy energy is obtained in the residual block after MC. Discrete cosine transform (DCT) [67–71], quantization, and entropy coding are further used in order to reduce the spatial redundancy [87] before it is stored or transmitted together with MVs. This procedure is quite similar in principle to that described in encoding of intra-frames. There is a build-in decoder in the encoder. Its job is to reconstruct the reference frame for ME and MC as depicted in Figure 2.5.

The decoder in Figure 2.6 uses the received MV to re-generate the predicted block since the reference frame is already in the decoder's buffer. The decoder then decodes the residual block and adds it to the predicted block to reconstruct the current frame. Compared to Figure 2.5, the structure of the decoder is essentially the same as the encoder, except the decoder does not need to perform ME. Since the decoder structure is part of the encoder, the predictions in the encoder and decoder are then synchronized.

2.2.2 Frame Types with Dependency

As we can see from the above, there are two basic types of frames adopted in video coding standards: those are encoded independently, not referencing from other frames (using intra-frame coding) and those are predicted from other frames (using inter-frame coding). The first are named as intra-coded frames (I-frames), which are coded independently without any temporal prediction to other frames. In other words, I-frames do not exploit any temporal redundancy. I-frames should be used at regular intervals in order to act as an access point for normal video playback and allow a random access operation as it is encoded without prediction from other frames. Although I-frames can provide these important features, the compression ratio is relatively small as only spatial redundancy reduction is carried out.

On the other hand, there are two types of inter-coded frames: predictive frames (P-frames) and bi-directional predictive frames (B-frames). P-frames undergo inter-frame coding using motion-compensated prediction from the preceding I- or P-frame to reduce temporal redundancy. By doing so, a significantly higher compression ratio is obtained. In practice, however, the number of P-frames between each successive pair of I-frames is limited. The reason behind is that any errors in the P-frame will be propagated to the next frame. In B-frames, the ME and MC further employ both past and future frames for prediction, which provides better ME when an object moves in front of or behind another object.

Figure 2.7 shows a sequence including all three frame types. The frames in-between successive I-frames is entitled a group-of-pictures (GOP). In this example, the GOP size is 12. In Figure 2.7, the first frame is an I-frame followed by two

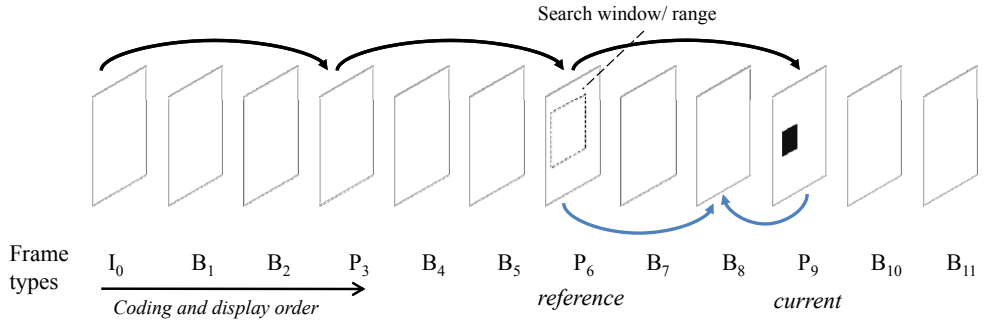


Figure 2.7: Temporal dependency between frames.

B-frames and one P-frame alternatively. The structure or size of each GOP can be changed in any standard in order to suit various applications. In Figure 2.7, the use of P-frames and B-frames exhibits temporal dependency in coded video data. In this example, P_9 depends on a prediction from a previously coded P_6, P_3, and then I-frame, I_0. The relationships among frames is denoted by black arrows in Figure 2.7. In addition, B-frames conduct predictions based on the preceding and following I- or P-frames. For instance, B_8 is referenced by the preceding P-frame, P_6 which is also referenced by P_3 and I_0. Therefore, B-frames should be encoded after the relevant P-frames. For this reason, the coding order is different from the display order. From the above discussion, temporal dependency is generally unavoidable in video coding. Such dependency favors the compression capability.

2.3 Complexity Reduction in HP Coding

In [28,30,45], the HP structure without backward motion prediction has successfully provided a low-delay encoded bitstream with the same degree of temporal scalability

as the HB structure. However, the increased temporal distance in the hierarchical prediction structure reduces the motion compensation accuracy [21]. Especially, a very long prediction distance in the lowest temporal layer, T_0, always incurs higher prediction error. Since all frames in T_0 are further acted as references for frames in succeeding temporal layers, it is desirable to improve the coding efficiency of the lowest layer whose predictions are based on long distance reference frames. Simply apply larger search range may tackle the problems but requires more computations in ME.

2.3.1 Search Range in HP Coding

To improve the HP coding efficiency, using more than one reference picture was suggested to be used in the reference picture list for a P-frame. In [28], the reference picture list includes preceding frames with the same temporal layer as the current P-frame. This idea successfully improves the concerned efficiency in HP coding when only two reference frames are utilized. However, the size of the storage buffer in the decoder is greatly increased since every preceding frame in the same layer should be saved, which may be from the previous GOP. In [88], a dual HP structure was proposed to tackle this shortcoming by allowing only the first frame of the corresponding GOP, which originally is kept in the single reference HP coding structure, to be the additional reference frame. This arrangement does not require extra storage buffer in the decoder anymore. However, it still involves predictions of a frame with a long distance reference. A straightforward way to improve the coding efficiency for the HP structure is to employ larger search range. However, the

computational complexity will also be increased significantly. This motivates us to propose a ME technique over a long distance reference frame without increasing the computational burden in the encoder. Therefore, a novel MV composition technique in ME is proposed to be adopted in HP coding in this thesis. Furthermore, a new vector selection approach has also been proposed for accurate ME results. Only ME on the current block to its short distant reference frame is required such that relatively smaller search range is enough for the predictions. Consequently, the motion locus for the current block to the long distance reference could be formulated by the short distance MV composition. This MV composition technique can therefore reduce the computational complexity while maintaining the efficiency. The conventional MV composition algorithms in video transcoding will be discussed in the next section.

In addition to using our proposed vector selection algorithm for choosing the optimal short distance MVs, MV composition is conducted to form a long distance MV for HP coding. Without enlarging the search range for ME, our proposed MV composition successfully provides an accurate long distance MV and better RD performance. As a result, reasonable computational complexity is achievable in HP coding.

Therefore, a novel MV composition technique is proposed to be adopted in HP coding in this thesis. Only ME on the current block to its short distant reference frame is required such that relatively smaller search range is enough for accurate ME results. Consequently, the motion locus for the current block to the long distance reference could be formulated by the short distance MV composition. This MV composition technique can therefore reduce the computational complexity while maintaining the efficiency. The conventional MV composition algorithms in video

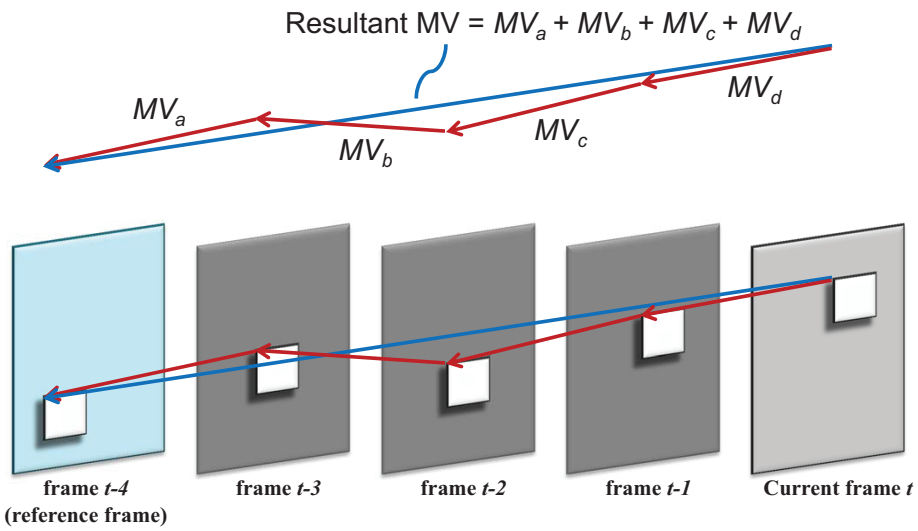


Figure 2.8: MV composition between current frame and reference frame.

transcoding will be discussed in the next section.

2.3.2 Existing Vector Selection Algorithms in MV Composition

MV composition, which is initially suggested for video transcoding [89, 90], is one of the solutions for reducing computational complexity in HP coding. The resultant MV is formed by the summation of selected MVs between a pair of consecutive frames as depicted in Figure 2.8. Therefore, the resultant MV is composed by $MV_a + MV_b + MV_c + MV_d$. The following sections will introduce some existing vector selection algorithms for choosing MV_a, MV_b, MV_c, and MV_d.

2.3.2.1 Median Vector Selection

In the median algorithm [91] as depicted in Figure 2.9, B_t^1 is the block being encoded in the current frame t and $MV_{t-1\rightarrow t}^1$ is the MV of B_t^1 referencing to the previous

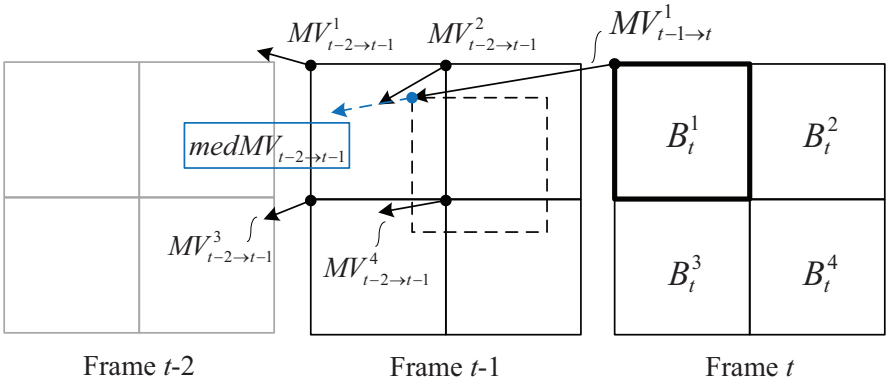


Figure 2.9: Median vector selection (Median).

frame $t-1$. Based on $MV^1_{t-1\to t}$, it points and covers some area in its reference frame $t-1$. The desired vector selection is to compute the median MV among motion vectors of the four neighboring block ($MV^1_{t-2\to t-1}$, $MV^2_{t-2\to t-1}$, $MV^3_{t-2\to t-1}$, $MV^4_{t-2\to t-1}$). The resultant MV between the current frame t and its final reference frame $t-2$ of B^1_t, $MV^1_{t-2\to t}$, is computed as

$$MV^1_{t-2\to t} = medMV_{t-2\to t-1} + MV^1_{t-1\to t} \tag{2.1}$$

where

$$medMV_{t-2\to t-1} = median\{MV^1_{t-2\to t-1}, MV^2_{t-2\to t-1}, MV^3_{t-2\to t-1}, MV^4_{t-2\to t-1}\}. \tag{2.2}$$

Using only the median vector to represent the in-between motions may lead to inaccurate results since irrelevant motion information in various block contents may be used if the objects undergo vigorous motions along frames.

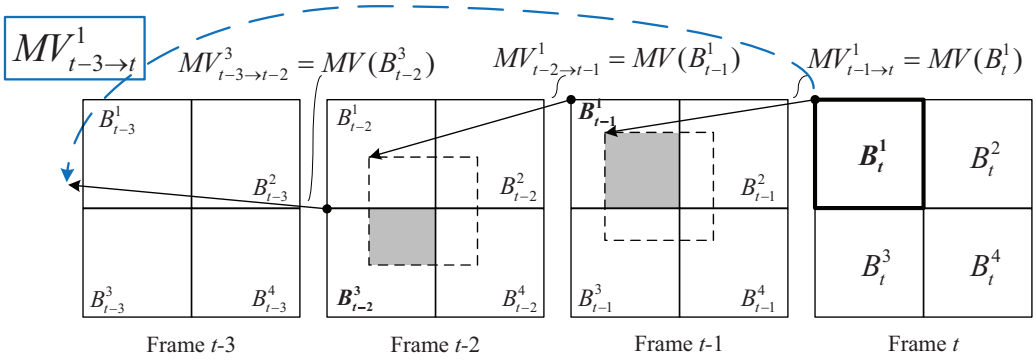


Figure 2.10: Forward dominant vector selection (FDVS).

2.3.2.2 Forward Dominant Vector Selection (FDVS)

Figure 2.10 illustrates an example of MV composition using FDVS [92, 93]. For each block, FDVS selects one dominant MV. A dominant MV is defined as the MV carried by the dominant block. The dominant block is the block that has the largest overlapping segment with the motion-compensated block of B^1_t in frame $t-1$. In Figure 2.10, the motion-compensated block of B^1_t (in dotted square in Frame $t-1$) overlaps with four blocks, B^1_{t-1}, B^2_{t-1}, B^3_{t-1} and B^4_{t-1} in frame $t-1$. In the illustration from Figure 2.10, FDVS selects B^1_{t-1} as the dominant block since it has the largest overlapping segment with the motion-compensated block of B^1_t, while its MV, $MV^1_{t-2\to t-1}$, becomes the dominant MV in the first step.

This dominant vector selection process is repeated until the target reference is reached, i.e. frame $t-3$ in this example. Therefore, in the second step of FDVS, the selected dominant MV in step 1, $MV^1_{t-2\to t-1}$, is used to point out the location of the motion-compensated block of B^1_{t-1} in frame $t-2$. Within the compensated area, B^3_{t-2} has the largest overlapping segment in frame $t-2$ as shown in Figure 2.10.

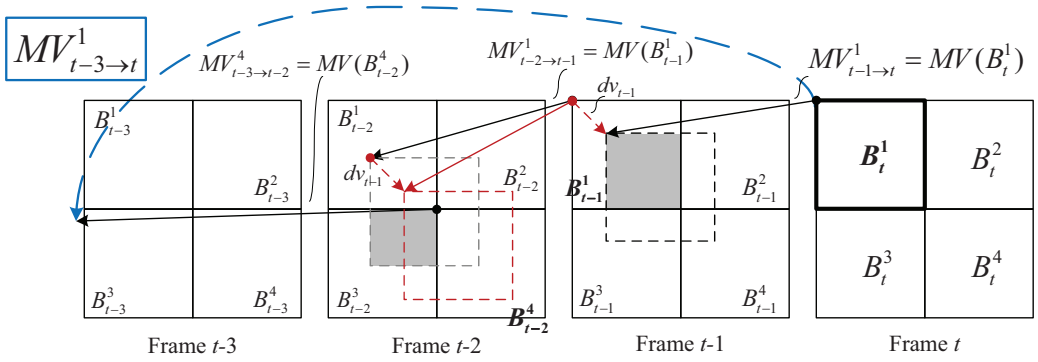


Figure 2.11: Enhanced forward dominant vector selection (E–FDVS).

FDVS defines it as the dominant block and its MV, $MV^3_{t-3\to t-2}$, is chosen as the dominant MV in this stage.

Therefore, the composed $MV^1_{t-3\to t}$ is composed by summing up the selected dominant MVs and can be written as

$$MV^1_{t-3\to t} = MV^3_{t-3\to t-2} + MV^1_{t-2\to t-1} + MV^1_{t-1\to t}. \tag{2.3}$$

The idea of FDVS is to find the most correlated blocks in-between frames for the current block and then use the MVs of these most correlated blocks to build the linkage between the current frame and the target reference frame. It could provide promising results for MV composition in frame-skipping transcoding [91, 93] and becomes the most popular vector selection algorithm in comparison with other existing algorithms.

2.3.2.3 Enhanced FDVS (E–FDVS)

E–FDVS [94] is the enhanced version of FDVS [92, 93] by further considering the difference between the selected dominant block and the selected dominant MV from

the incoming MV candidates. In Figure 2.11, the first step of the dominant MV selection is the same as that in FDVS. By $MV^1_{t-1\to t}$, the selected dominant block is B^1_{t-1} and the dominant MV is $MV^1_{t-2\to t-1}$ since B^1_{t-1} has the largest overlapping segment. At this moment, E–FDVS aims to avoid the mismatch between the dominant block and the dominant MV in frame $t-1$ propagating to frame $t-2$. It figures out the delta vector, dv_{t-1} between the dominant block and MV as shown in frame $t-1$ of Figure 2.11. This dv_{t-1} is appended to the dominant MV to locate the compensated area in frame $t-2$. The overlapping compensated area is then refined by this dv_{t-1} and the largest overlapping segment in frame $t-2$ by E-FDVS is different from that obtained by FDVS. E–FDVS selects B^4_{t-2} as the dominant block for frame $t-2$ and its MV, $MV^4_{t-3\to t-2}$ becomes the dominant MV.

From Figure 2.11, the delta vector between the dominant block and MV, denoted as dv_{t-1}, is added into the selected dominant MV and the refined resultant MV can be written as

$$MV^1_{t-3\to t} = MV^4_{t-3\to t-2} + MV^1_{t-2\to t-1} + MV^1_{t-1\to t}. \tag{2.4}$$

Various vector selection algorithms provide promising short distance MV choices in consecutive frames for MV composition. However, when the number of MV composition steps is increased, their selection criteria may lead to an inaccurate resultant MV since the relevant area in their considerations is diminished after several composition steps and no longer fully matches with the dominant blocks. Therefore, our proposed vector selection algorithms in MV composition aims to tackle this problem and the solution will be provided in Chapter 3.

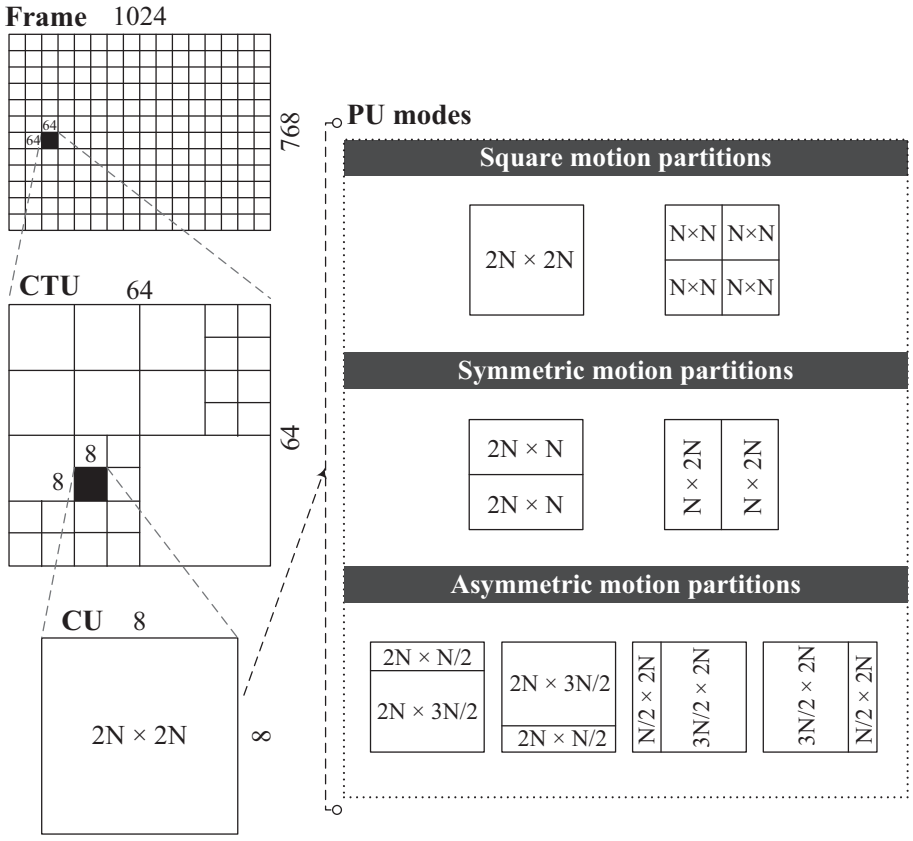


Figure 2.12: Flexible block partitioning in HEVC: CTUs, CUs, and PUs.

2.4 Complexity Reduction in HEVC

The latest HEVC standard is targeted for efficient compression of high resolution (720p and 1080p) and 3D videos [95]. Compared with the H.264 standard, HEVC can reduce the bit rate by almost 50% with the similar perceptual video quality [37]. HEVC adopts the same block-based hybrid video coding scheme [14] used in the prior video compression standards. The achievement in coding gain results mainly from its more flexible block partition mechanism at the cost of high computational complexity [51–53]. In the encoding process as shown in Figure 2.12, each picture

is divided into coding tree units (CTUs), which is the base unit in HEVC [55–57]. The size of a CTU can be chosen as 64×64, 32×32, 16×16, or 8×8. A CTU is composed of a luma coding tree block (CTB), two chroma CTBs, and the associated syntax elements. The luma and chroma CTBs can be further partitioned into smaller blocks using a quad-tree structure [96]. The leaves of the CTBs are specified as coding blocks (CBs). One luma CB, and its corresponding two chroma CBs, together with the syntax elements form a coding unit (CU). The CU shares the identical prediction mode (intra, inter, skip, or merge), and it acts as the root for a prediction unit (PU) partitioning structure. Figure 2.12 lists out all possible PU modes. The PU is composed of prediction blocks (PBs) where the same prediction process is applied for its luma and chroma PBs. In the PU partitioning structure of HEVC, each luma/chroma CB can be further partitioned into one, two, or four rectangular shaped PBs. In HEVC, it adopts square motion partitions, symmetric motion partitions, and asymmetric motion partitions [97], as shown in Figure 2.12. It means that every CU undergoes motion predictions by various types of PU partitions. With this flexible block partitioning mechanism, inter prediction consumes about 60-70% of the whole encoding time [23, 34].

Recently, many researchers have devoted their efforts to expedite the inter prediction process using some fast mode decision, early mode termination and fast search approaches in HEVC [98–103]. This section will categorize them and deliver the literature reviews on them. They tackle the problem of complexity increment in ME but may induce various kinds of drawbacks by themselves.

2.4.1 Early Termination and Fast ME Algorithms for Complexity Reduction

Early termination based on various coding information was suggested in [98–101]. For instance, zero coded block [98,99] and the selection of SKIP mode [100,101] are employed to trigger early termination of CU size decision. The works in [102, 103] further exploited the spatio-temporal analysis, motion homogeneity, and RD cost to determine the condition of early termination. These methods focus on reducing the computational complexity of selecting the best CU and PU, which are also highly related to the ME algorithm.

Fast ME algorithms always restrict the number of search locations. Test Zone Search (TZS) is one of the popular methods implemented in the HEVC test model (HM) [104, 105]. TZS starts with a diamond or square search pattern with different stride lengths of 1, 2, 4, 8, 16, 32, and 64 to locate an initial search point. This initial search point is taken as the center search point for the possible raster search and refinement. In [106], a rotating hexagonal grid with alternate horizontal/vertical hexagonal patterns was suggested for TZS to locate the global minima with early termination. However, the multiple initial search point decision is still a major burden on TZS [107]. In addition to TZS, other search strategies such as directional search were suggested in [108, 109]. These works focus on applying specific search patterns to reduce search points within a fixed search range. Nevertheless, various search patterns are not preferable for hardware implementation due to their irregular data flow [110].

2.4.2 Adaptive Search Range for Complexity Reduction

In this circumstance, full search with an adaptive search range (ASR) can provide both search point reduction and regular data flow in hardwares. Besides, ASR can also be applied to various search patterns in software implementation to further reduce the number of search points. In [111, 112], the search range is modeled by the Cauchy distribution and Laplace distribution, which exhibit good results in terms of quality and complexity in H.264. Other ASR algorithms proposed in H.264 correlates the search range of the current block with the motion characteristics of its neighbors. Examples of these motion characteristics include MVPs [113,114], sum of absolute difference [115], and motion activities [116] from neighboring blocks, MV differences in previous frames [117], etc. Recently, these concepts of ASR have been directly extended to support the flexible block partition in HEVC [110, 118–121]. The three most recent algorithms are Maximum Likelihood Estimation Laplace Distribution Algorithm (MLELD) [117], Large and Small Motion Frame Algorithm (LSMF) [118], and Linear Adaptive Model for Adaptive Search Range Algorithm (LAMASR) [119].

MLELD [117] models the MV differences of the previous frame by the zero-mean Laplace distribution where the parameters are solved by maximum likelihood estimation (MLE) for a motion estimator. The estimated distribution model is then used to set the final ASR. In LSMF [118], it classifies the current block either in a small motion frame (SMF) or a large motion frame (LMF) by the distribution of MV differences in the previous frame. Consequently, it differentiates the current block into two sub-classes of different degrees of motion activity by the average MV

difference of the co-located CTU. Larger ASR is assigned in high motion activity blocks and vice versa. The algorithm of LAMASR [119] adopts a linear adaptive search range model including an overdetermined equation system. The parameters in the system can be solved by PU size, MV difference and motion vector predictors. The ASR is then finalized with a fixed scale factor.

The recursive CU partitioning mechanism suffers from the expensive computations in ME and fast search pattern approaches in a fixed search range may not suitable for the regular data flow in hardware configurations. This motivates us to propose adaptive search range for complexity reduction instead of suggesting any new search patterns. Furthermore, there are only a few of literatures trying to adopt the new features provided in bitstreams for ASR. For example, the ASR algorithm given in [121] uses the correlation between views in 3D-HEVC for ASR adjustment. However, the disparity among views might reduce the correlation between the search range of the view being coded and the MVs of its neighboring views. In [122], the authors tried to reveal the usage of depth information for fast mode decision. The depth information is useful for revealing the relatively movements between objects along a period of time. This algorithm makes the fast decision on selecting SKIP, inter-mode, and intra-mode in H.264 coding only. To the best of our knowledge, there is no existing work of ASR considering the depth information of 3D videos, which has gained great attention recently. In this thesis, the depth information brings new room for designing an efficient ASR algorithm in HEVC. Depth intensity variance in depth maps are revealed for motions along z-direction and formulating the probable motion ranges for x- and y-directions. As a result, adaptive search range can be figured out such that complexity reduction can be achieved.

2.5 Chapter Summary

In this chapter, we initially gave an overview of the hybrid motion-compensated predictive coding. It aims for coding efficiency by reducing the video data redundancies. Only the residual signal is coded by the motion-compensated predictive coding. Therefore, the residual signal form spatial dependency between blocks and temporal dependency between frames in the coded bitstream. By these dependencies, some limitations are raised during encoding among blocks and frames of the coded data. For example, coding order and display order cannot be arbitrary changed. Otherwise, more effort is paid for normal encoding process. Literature reviews were made in MV composition for defining the motion locus on the coding frame and its far away reference. Apart from the MV composition technique, its vector selection algorithms such as median vector selection, FDVS, and its enhanced version E–FDVS were introduced.

Furthermore, since the computational complexity reduction is the scope of our research, literature reviews on existing fast algorithms in various categories for HEVC complexity reduction were presented in this chapter. The ASR approach raises our motivation on motion locus prediction in an earlier stage than ME such that less computations in ME can be conducted. The three most recent ASR algorithms were reviewed. In the following chapters, we examine the possibility of improving the MV composition algorithms by novel vector selection techniques. Nevertheless, motion locus prediction from depth map and the spatial and temporal correlation of the coded information will be illustrated for HEVC complexity reduction.

Chapter 3

Determinations on Motion Locus by Motion Vectors Composition

3.1 Introduction

MV composition is proposed to be used in HP coding since short distance MVs can be summed to form motion locus for a long distance MV. By this arrangement, large search range can be avoided in coding a frame with a long distance reference, especially in the lowest temporal layer. Therefore, computational complexity can be reduced in ME. The organization of this chapter is as follows. In Section 3.2, the motivation of this research work on HP coding is stated. Section 3.3 describes the adoption of the proposed MV composition in coding videos with the HP structure. Section 3.4 evaluates the idea of the proposed MV composition scheme in HP coding and a popular vector selection algorithm in MV composition steps. Section 3.5 reveals the impact of the vector selection algorithm on the proposed scheme.

Based on this observation, a tailor-made motion vector selection algorithm is further adopted in MV composition. This section then provides simulation results in Bjontegaard (BD) measurement [123] and computational complexity among various MV selection algorithms in the HP coding scheme. Finally, conclusions are given in Section 3.6.

Parts of the contents of this chapter are extracted from our published work in [124] ©2013 Elsevier and [125] ©2013 IEEE:

- **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Motion Estimation in Low-delay Hierarchical P-frame Coding Using Motion Vector Composition," *Journal of Visual Communication and Image Representation*, 24 (8) (2013) 1243-1251.

- **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Motion Vector Composition in Low-delay Hierarchical P-Frame Coding," in *Proceedings of IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP 2013)*, Beijing, China, July, 2013, pp. 551-555.

3.2 Motivation for MV Composition Scheme in HP Coding

Chapter 2 has mentioned a straightforward way to improve the coding efficiency for the HP structure in lower temporal layers by employing larger search range (SR). However, the computational complexity will also be increased significantly. Simu-

Table 3.1: RD performances of HP coding in various search ranges versus size of $R = 8$.

Sequences	$R = 8$		$R = 16$		$R = 32$	
	Bitrate (kbits/s)	PSNR (dB)	Bitrate (kbits/s)	PSNR (dB)	Bitrate (kbits/s)	PSNR (dB)
Bus	5364.9	35.10	-1.57%	+1.18	-5.85%	+1.50
Football	2804.4	34.54	-9.75%	+0.32	-14.99%	+0.56
Stefan	2687.5	34.34	-5.99%	+0.14	-8.56%	+0.22
Rush hour	20686.2	40.55	-15.20%	-0.18	-26.49%	-0.02
Spin-calendar	53684.9	35.59	-5.49%	+0.22	-21.85%	+0.54
RD Performances in Average			-6.89%	+0.41	-12.77%	+0.79
Computational Complexity Increase			+276.80%		+1361.94%	

lations in Table 3.1 were conducted in the H.264 JM17.2 codec [126]. All sequences were encoded in 100 frames. Table 3.1 shows that the improvement on RD performance of the base and the first layers, denoted as T_0 and T_1, respectively when the SR increases in ME. In the table, $R = x$ represents the size of the SR in $\pm x$, where x =8, 16, and 32 in this experiment. In average, Table 3.1 exhibits that a larger SR used in ME improves RD performance in a great extent at the expense of remarkable increment in computational complexity. From this table, bitrate reduction by about 6% in average and PSNR improvement of video quality by about 0.4 dB can be achieved once SR is doubled from R =8 to R =16. However, the computational

complexity increases by 276.80% and even by 1361.94% when SR expands in two dimensions by a factor of 2 and 4, respectively. In conclusion, a large SR is useful to maintain or get better RD performance in the lower temporal layers. However, it demands a significant increase in computational complexity. Therefore, a novel MV composition technique is proposed to be adopted in HP coding so as to reduce the computational complexity while maintaining the coding efficiency.

3.3 Proposed MV Composition Scheme in HP Coding

To start with, some symbols for the sake of illustration of the proposed scheme are defined. In the HP structure with the GOP size of L, each current frame F_t to be coded has its corresponding reference frame F_r which has different prediction distances according to temporal layers, T_k, where $r = 0, 1, .., log_2 L$. The indexes t and r are related as

$$r = t - 2^{(log_2 L)-k}. \tag{3.1}$$

In a ME process between frames, let $B_t^{corner} = (x, y)$ be the top left coordinate of the current block being coded and $MV_{r\to t}$ be a MV of a frame F_t referenced by a frame F_r . The sum of absolute differences (SAD) of the block with the size of $N \times N$ pixels is calculated as

$$f_{SAD}(B_t^{corner}, MV_{r\to t}) = \sum_{i=0}^{i=N-1} \sum_{j=0}^{j=N-1} \mid F_t(B_t^{corner} + (i,j)) - F_r(B_t^{corner} + MV_{r\to t} + (i,j)) \mid \quad . \tag{3.2}$$

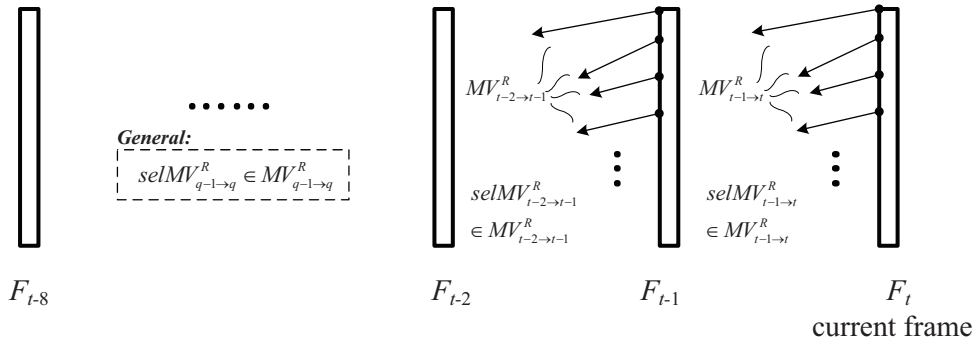


Figure 3.1: Notations of the proposed HP coding scheme.

The proposed scheme involves MV composition as a new coding framework in HP coding. It is the approach of MV reuse in composition. The proposed scheme consists of three core steps. First, all MVs between adjacent frames are estimated through ME with a relatively small SR. Second, we obtain composed MVs for lower layers using the computed MVs in the previous step. It also involves selection algorithm between the computed MVs. Third, the final MV can be obtained by performing refinement of the composed MV over a narrow SR.

3.3.1 ME in Consecutive Frames with a Smaller SR

By minimizing $f_{SAD}(B_t^{corner}, MV_{r\to t})$ in (3.2), through the conventional full-search (FS) ME with a small SR, MVs of all blocks in F_t pointing to the previous frame, F_{t-1}, where t is the current frame index, are obtained. From the illustration of Figure 3.1, all these MVs form a set of MV between F_t and F_{t-1}, and it is denoted as a set of $MV^R_{t-1\to t}$ where R will be the amplitude of the SR. It is noted that the MVs in the highest layer, T_3, are obtained in this step. They are the vectors between each pair of consecutive frames.

3.3.2 Optimal MV Selection and Composition

In lower layers, MV composition is conducted with the help of $MV^R_{t-1\to t}$ obtained above in layer T_3. We are the first to propose MV composition in HP coding as a new coding framework. To compute the MV of F_t in lower layers with its corresponding reference F_r, $MV_{r\to t}$, they are composed by a general form as

$$MV_{r\to t} = \sum_{q=r+1}^{t} selMV^R_{q-1\to q}, \tag{3.3}$$

where $selMV^R_{q-1\to q} \in MV^R_{q-1\to q}$.

For each MV composition step, $selMV^R_{q-1\to q}$ is a MV for a selected block from $MV^R_{q-1\to q}$, which consists of motion vectors between consecutive frames. For a clear illustration, one example is shown in Figure 3.1 where a set of MVs between F_t and F_{t-1} is denoted as one of $MV^R_{t-1\to t}$. Then, the $selMV^R_{t-1\to t}$ is selected from $MV^R_{t-1\to t}$ for a block between this pair of consecutive frames. Furthermore, according to (3.3), the composed $MV_{t-4\to t}$ for F_t in T_1, which uses F_{t-4} as the reference frame, can be computed as

$$MV_{t-4\to t} = selMV^R_{t-4\to t-3} + selMV^R_{t-3\to t-2} + selMV^R_{t-2\to t-1} + selMV^R_{t-1\to t}. \tag{3.4}$$

The way to choose $selMV^R_{q-1\to q}$ in each composition step is of great importance to the proposed HP coding scheme. One straightforward way is to adopt the forward dominant vector selection (FDVS) [93], which is a well-known technique used in video transcoding for MV composition. Figure 3.2 also illustrates an example of

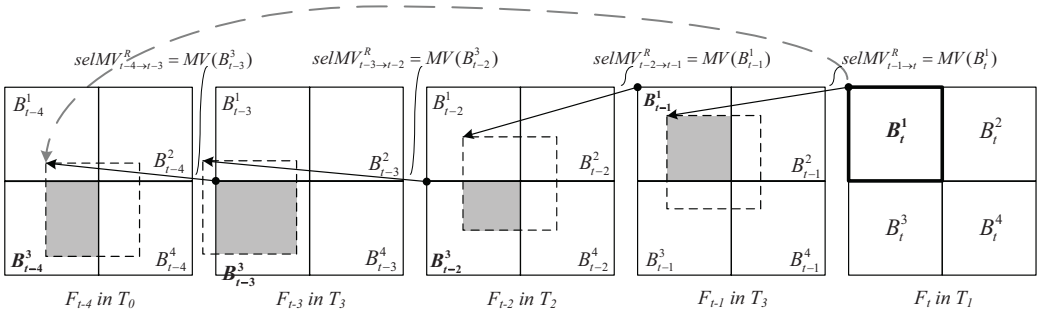


Figure 3.2: Vector selection algorithm FDVS adopted in the proposed HP coding.

using FDVS in the HP coding scheme for layer T_1 where the distance between the current frame and the reference frame is 4 frames. Only four neighboring blocks within a frame are shown in Figure 3.2. In this example, B_t^n represents the n^{th} block in F_t, and its MV referencing to its previous frame is denoted by $MV(B_t^n)$. To compute the new composed $MV_{t-4\to t}$ of $MV(B_t^1)$ using FDVS as shown in Figure 3.2, one dominant MV carried by a dominant block is chosen in each pair of consecutive frames. The dominant block is the one that has the largest overlapping segment with the motion compensated block of $MV(B_t^1)$ in the previous reference frame. In Figure 3.2, the motion compensated block (dotted square in F_{t-1}) overlaps with four blocks, B_{t-1}^1, B_{t-1}^2, B_{t-1}^3, and B_{t-1}^4, in F_{t-1}. B_{t-1}^1 is selected in the first step of FDVS as the dominant block while $MV(B_{t-1}^1)$ is the dominant MV. Therefore, $selMV^R_{t-2\to t-1}$ is set to $MV(B_{t-1}^1)$. This dominant vector selection process is repeated until the desired reference frame is reached, i.e. F_{t-4} in this example. From the example shown in Figure 3.2, the new $MV_{t-4\to t}$ of B_t^1 is then composed by summing up the selected dominant MVs and can be written as

$$MV_{t-4\to t} \text{ of } B_t^1 = MV(B_{t-3}^3) + MV(B_{t-2}^3) + MV(B_{t-1}^1) + MV(B_t^1). \quad (3.5)$$

3.3.3 Refinement after MV Composition

After MV composition, a small refinement vector $\vec{\epsilon}$ is added to the resultant MV, which is expressed as

$$\vec{\epsilon} = (\epsilon_x, \epsilon_y) \text{ where } \epsilon_x, \epsilon_y \in [-1, 0, 1]\,. \quad (3.6)$$

The refinement operation provides fine tunings for the MV composition. It adds an optimum $\vec{\epsilon}_{opt}$ which can yield a minimum value of f_{SAD}, such that

$$\vec{\epsilon}_{opt} = \arg\min_{\vec{\epsilon}} f_{SAD}(B_t^{corner}, MV_{r\to t} + \vec{\epsilon}). \quad (3.7)$$

With (3.7), a final MV, denoted as $\hat{MV}_{r\to t}$ with refinement is calculated as

$$\hat{MV}_{r\to t} = \sum_{q=r+1}^{t} selMV_{q-1\to q}^{R} + \vec{\epsilon}_{opt}. \quad (3.8)$$

From the above illustration, the forward coded MVs between adjacent frames are utilized in each composition step. Only ME among adjacent frames is necessary. In this case, a small SR is sufficient. The computational complexity thus can be significantly reduced by eliminating the need to search for temporal remote reference frames. It can be seen that the coding order of MV composition is also the same as

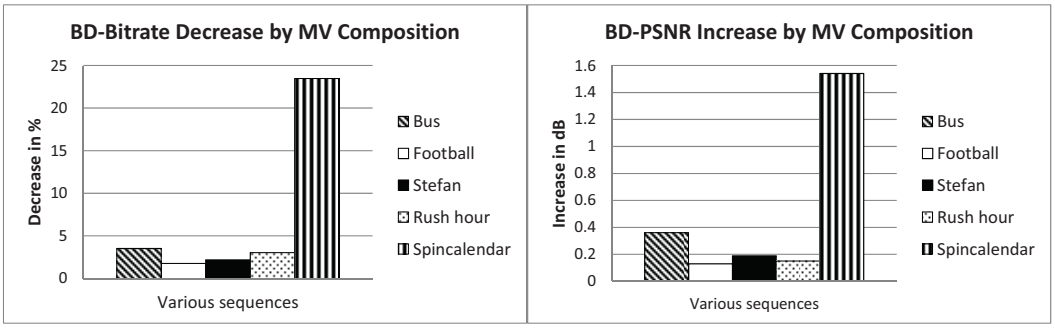
the display order shown in HP coding structure, which does not impose extra delay in the coding process.

3.4 Simulation Results on the Proposed HP Coding Scheme

In this section, we present some simulation results to evaluate the performance of adopting FDVS for MV composition in HP coding. All the simulations were conducted based on the H.264 JM17.2 codec [126]. The test sequences used in the simulations were in yuv format with 4:2:0 sampling. They included "Bus (CIF)", "Football (CIF)", "Stefan (CIF)", "Rush hour (720p)", and "Spincalendar (720p)". They are representative for simulations in video coding. It is because they comprise various common characteristics among natural sequences, such as rich texture, camera and objects movements, which provide a wide range of compression difficulty and subject matter. Each sequence encodes 100 frames. In HP coding, a quantization parameter (QP) is increased monotonically for the cascaded quantization [127] for each k^{th} temporal layer (QP_k), as

$$QP_k = QP_{base} + k \text{ where } k = 0, 1, 2, 3. \tag{3.9}$$

Here, QP_k depends on the QP for the base layer denoted as QP_{base}. For RD performance analysis, QP_{base} was set to 20, 24, 28, and 32. The bitstreams were encoded with the HP structure by different algorithms. The evaluation in BD measurement of the proposed HP coding algorithm versus the full-search (FS) algorithm



(a) (b)

Figure 3.3: Results by the proposed HP coding scheme with FDVS: (a) BD-bitrate decreases, and (b) BD-PSNR increases.

was conducted. The SR amplitude of the FS and the proposed algorithm between the frames were set to 8 and 16 for the CIF and 720p sequences, respectively. To evaluate the proposed HP coding scheme effectively, comparisons were focus on frames in T_0 and T_1 that have distant references in a GOP.

From Figure 3.3, the proposed HP coding scheme with FDVS shows a significant BD improvement. BD-bitrate decrease usually reaches 3% to 5%, and up to 23% in video sequences with high motion activities such as "Spincalendar (720p)" as depicted in Figure 3.3(a). In addition, BD-PSNR increase shown in Figure 3.3(b) has been obtained from 0.4 dB to 1.5 dB among various sequences. It is proved that the proposed HP coding scheme can provide remarkable performance in the view point of rate and distortion in comparison with the FS algorithm in coding with distant reference frames. More details of evaluation in complexity of the proposed HP coding scheme will be shown in Section 3.5.2.

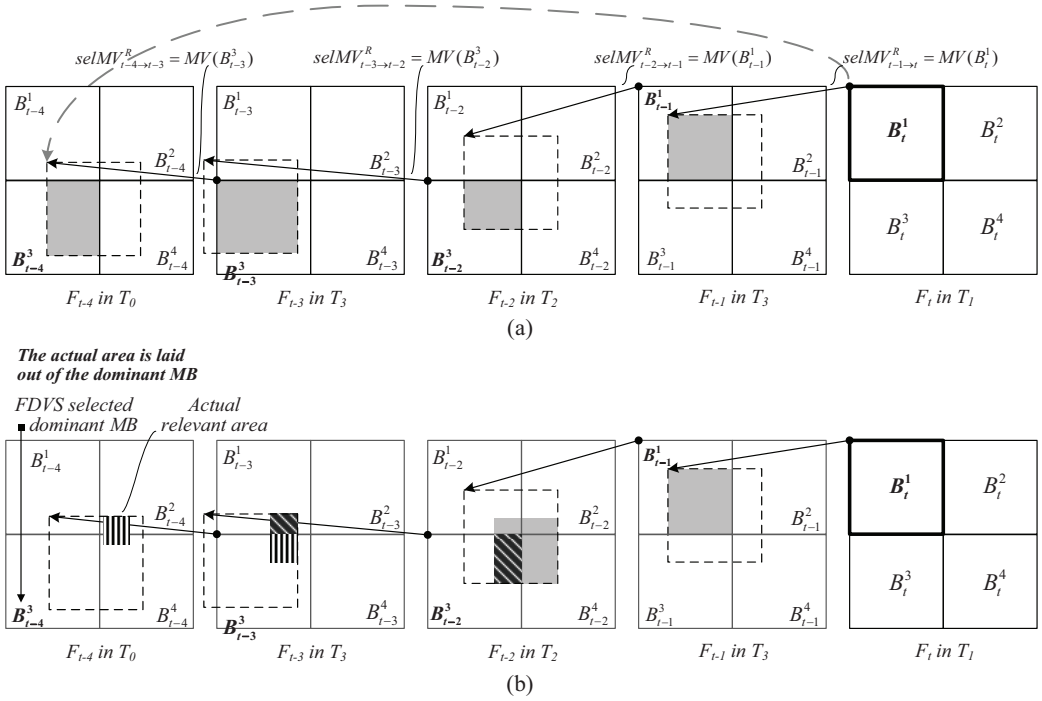


Figure 3.4: (a) FDVS adopted in HP coding, and (b) relevant area in hatched and shaded regions by FDVS in (a).

3.5 Impact of Vector Selection Algorithms on The Proposed MV Composition Scheme in HP Coding

The success of the proposed HP coding scheme depends on the reliability of the MV selection and composition algorithm in coding the current frame with its reference frame. Owing to a very long prediction distance in the low temporal layers, FDVS cannot guarantee to obtain promising results. This can be explained by Figure 3.4, in which Figure 3.4(a) is redrawn from Figure 3.2 for better illustration and comparison. Figure 3.4(b) highlights the relevant area of the target block, B_t^1 , during the MV composition process when FDVS is used for the vector selection algorithm. In

F_{t-1} of Figure 3.4(b), the dot-bordered area shows the motion compensated area of B_t^1. According to the area size, B_{t-1}^1 is the defined dominant the block.

In B_{t-1}^1, only the shaded area is actually relevant to B_t^1. After that, B_{t-1}^1 is defined the dominant block by its largest overlapping segment size. This shaded area is combined with the non-shaded area in B_{t-1}^1, which is an irrelevant area to the target coding block, B_t^1, to determine the dominant block in F_{t-2}. The selection of the dominant block is continuous till the desired reference frame, F_{t-4} in Figure 3.4(b). It can be observed that the relevant area of B_t^1 further diminishes in the selection process by FDVS. The relevant area of B_t^1 in F_{t-4} (shown in vertical hatched region) is even laid out of the dominant block, B_{t-4}^3 as depicted in F_{t-4} of Figure 3.4(b). It incurs inaccuracy of the composed MVs since a large irrelevant area to the target coding block is used to decide the dominant block. And it may lead to a worse coding performance when the number of composition steps increases. The reason is that the dot-bordered area under consideration for each frame is not only the content relevant to area (grey region) of the target block in the previous frame but also a large portion of the non-relevant area (white region) which may reduce the reliability of the selected dominant MVs.

In [128], the concept of utilizing only the relevant areas in the target block and maximizing these areas was suggested by us, and referred to as a multiple candidate vector selection (MCVS) algorithm. Although MCVS in [128] aims at improving the accuracy of MV composition in video transcoding for fast-forward playback, it motivates us to modify MCVS such that it is best suited for our new HP coding framework when the temporal distances between the reference and current frames are far away in lower temporal layers.

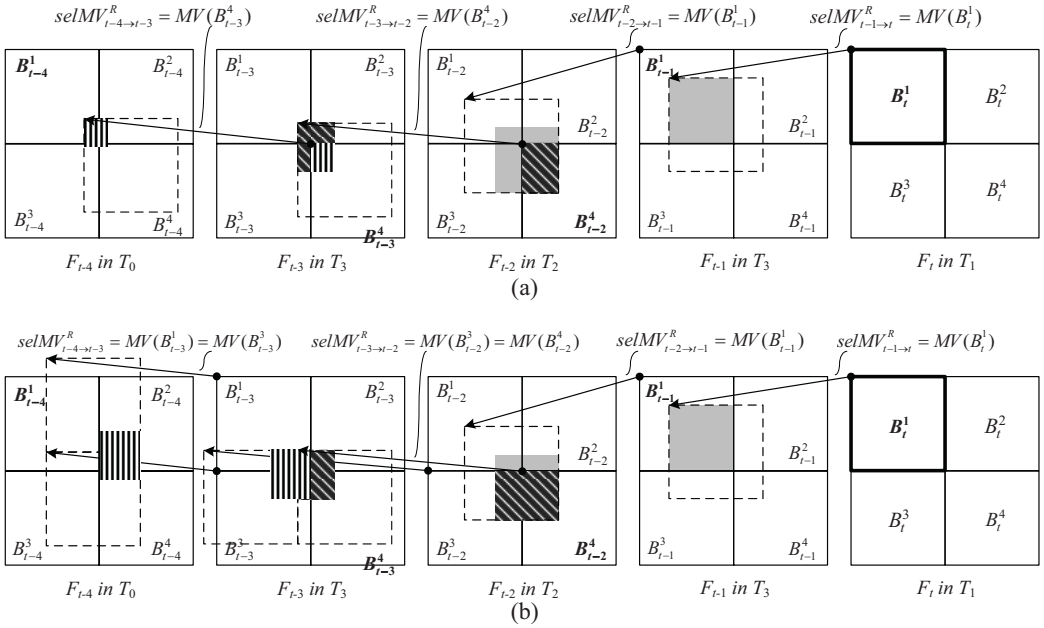


Figure 3.5: Example of adopting AMCVS in HP coding: (a) only relevant area to determine dominant block for MV selection, and (b) Merge process in MV selection.

3.5.1 Proposed Adaptive-Multiple Candidate Vector Selection

The proposed adaptive-multiple candidate vector selection (AMCVS) algorithm may lead to different resultant composed MVs from FDVS. In FDVS, $selMV^R_{q-1\to q}$ is defined as the MV from the dominant compensated block which gets the largest size of compensated area, i.e. $MV(B^1_{t-1})$ in F_{t-1} of the example shown in Figure 3.4(a). It will further select B^3_{t-2} as the dominant block for $selMV^R_{t-3\to t-2}$ by the largest dot-bordered area in F_{t-2}. This criterion for selection may lead to a worse coding performance when the number of composition steps increases. The reason is that the dot-bordered area in F_{t-2} under consideration is not only the content relevant to area (grey region) of the target block in F_{t-1} but also a large portion of non-relevant

area (white region) which may reduce the reliability of$selMV^{R}_{t-3\rightarrow t-2}$. In comparison to FDVS, AMCVS is no longer simply utilizing the whole dot-bordered area to select a dominant block in each MV composition step. The main philosophy of the proposed AMCVS is exemplified in Figure 3.5.

3.5.1.1 Actual Relevant Area Utilization

In the example of Figure 3.5(a), AMCVS only uses the actually relevant dominant compensated area which is the grey area or hatched area instead of the dot-bordered area in each composition step to determine the next dominant block and its dominant MV. Therefore, the result of dominant block determination in F_{t-2} is B^4_{t-2}, The $selMV^{R}_{t-3\rightarrow t-2}$ becomes $MV(B^4_{t-2})$, which is different from that obtained by FDVS ($MV(B^3_{t-2})$). By only taking the relevant areas to the target coding block into consideration, the resultant $MV_{t-4\rightarrow t}$ of B^1_t by AMCVS shown in Figure 3.5(a) can be computed as

$$MV_{t-4\rightarrow t} \text{ of } B^1_t = MV(B^4_{t-3}) + MV(B^4_{t-2}) + MV(B^1_{t-1}) + MV(B^1_t). \quad (3.10)$$

Although the above selection process guarantees only the relevant area of the target coding B^1_t is used in MV composition, the area for deciding the dominant block still becomes smaller and smaller along the frames, as shown in F_{t-4} of Figure 3.5(a). The situation is more serious for MV composition in lower temporal layers, which involves more composition steps with the dominant block and MV determinations for a composed MV in a distant reference. It results in reducing the accuracy of the resultant MVs.

3.5.1.2 Maximization of Dominant Area by Merging

To further increase their accuracy, the relevant area to the target block should be kept as large as possible during MV composition by also considering non-dominant areas, but relevant to the target coding B_t^1. To achieve this, AMCVS can entirely make use of the homogeneity of MVs in which compensated areas in blocks with the same MV are merged.

For the example in Figure 3.5(b), the areas in B_{t-2}^3 and B_{t-2}^4 denoted by the diagonal hatched pattern in F_{t-2}, which have the same MV, are merged for dominant block determination in F_{t-3}. Under this merging process, the final selected dominant MV for $selMV_{t-4\rightarrow t-3}^R$ is picked according to this merged area, and the resultant $MV_{t-4\rightarrow t}$ of B_t^1 can be formed in

$$MV_{t-4\rightarrow t} \text{ of } B_t^1 = MV(B_{t-3}^3) + MV(B_{t-2}^4) + MV(B_{t-1}^1) + MV(B_t^1). \quad (3.11)$$

From Figure 3.5(a) and Figure 3.5(b), it can be observed that the areas determining the block selection increase significantly when homogeneity of MVs is taken into consideration. Consequently, the final resultant MV should be more reliable. This merging process is specifically more appropriate for areas with homogeneous motion such as blocks in the background and inside the moving objects.

3.5.1.3 Multiple Candidates Selection

For object boundary of a video object, the merging process cannot help since the neighboring MVs of blocks are not identical. In fact, AMCVS is a greedy algorithm that only considers the maximum size of relevant compensated area in each

composition step and selects the optimal vector in the particular step for the entire composition. For each composition step, it makes the locally optimal choice with the hope of locating the global optimum. Thus, the MV selection in the rest of composition steps has been decided by the initial step. To increase the probability of obtaining the locally optimal solution that is closer to the global optimum, the AMCVS algorithm also keeps more than one candidate block for MV composition. The candidates are ranked by their relevant segment size. Using more candidates in each step provides more combinations of composition paths to go further. In our proposed AMCVS, we heuristically use different number of candidates, $NCand$, in different temporal layers, T_k, which have different prediction distances, and $NCand$ can be given by

$$NCand = \begin{cases} NCand^{max}, \ k = 0 \\ \frac{NCand^{max}}{2}, \ k = 1 \\ \frac{NCand^{max}}{4}, \ k \geq 2 \end{cases}, \tag{3.12}$$

where $NCand^{max}$ is set to 4 because the maximum number of overlapping blocks is 4 in the first composition step. From (3.12), it is seen that four candidates are used in T_0, which has the longest prediction distance. Two candidates are then enough for T_1 while one candidate is used for other higher layers.

Among these multiple candidates, the final optimal composed MV for the current block being encoded, B_t^n, is selected by minimizing the Lagrangian cost function, J_{motion} as

$$J_{motion}(\hat{MV}_{r\to t(Cand)}, \lambda_{motion}) = \\ f_{SAD}(B_t^{corner}, \hat{MV}_{r\to t(Cand)}) + \lambda_{motion} \cdot R_{motion}(\hat{MV}_{r\to t(Cand)} - MVP) \quad , \tag{3.13}$$

where B_t^{corner} is the top left coordinate of B_t^n, MVP is the MV used for prediction, $\hat{MV}_{r\to t(Cand)}$ is one of the refined composed MV candidates where $Cand$ is the index presenting a candidate, λ_{motion} is the Lagrangian multiplier for motion estimation (ME), $R_{motion}(\hat{MV}_{r\to t(Cand)} - PMV)$ is the estimated number of bits for coding $\hat{MV}_{r\to t(Cand)}$, and f_{SAD} is the sum of of absolute differences between the block B_t^n being coded and its reference block, which is defined in (3.2). The one with smallest J_{motion} is determined to be the final composed MV. The flowchart of the new AMCVS for coding frames in T_k is then shown in Figure 3.6.

3.5.2 Simulation Results of AMCVS

The simulation results we present in this section were also obtained with the "Bus (CIF)", "Football (CIF)", "Stefan (CIF)", "Rush hour (720p)", and "Spincalendar (720p)" sequences, using the parameters and performance criterion described in Section 3.4. More evaluations in BD measurement of the proposed HP coding scheme in various vector composition algorithms versus FS were conducted. The vector composition algorithms between consecutive frames include FDVS [93], its enhanced version (E–FDVS) [94], median algorithm (MEDIAN) [91] and the AMCVS proposed in Section 3.5.1. To evaluate the impact of various vector composition algorithms in HP coding in details, the BD measurement was conducted in two groups, and they are listed in Table 3.2 and Table 3.3.

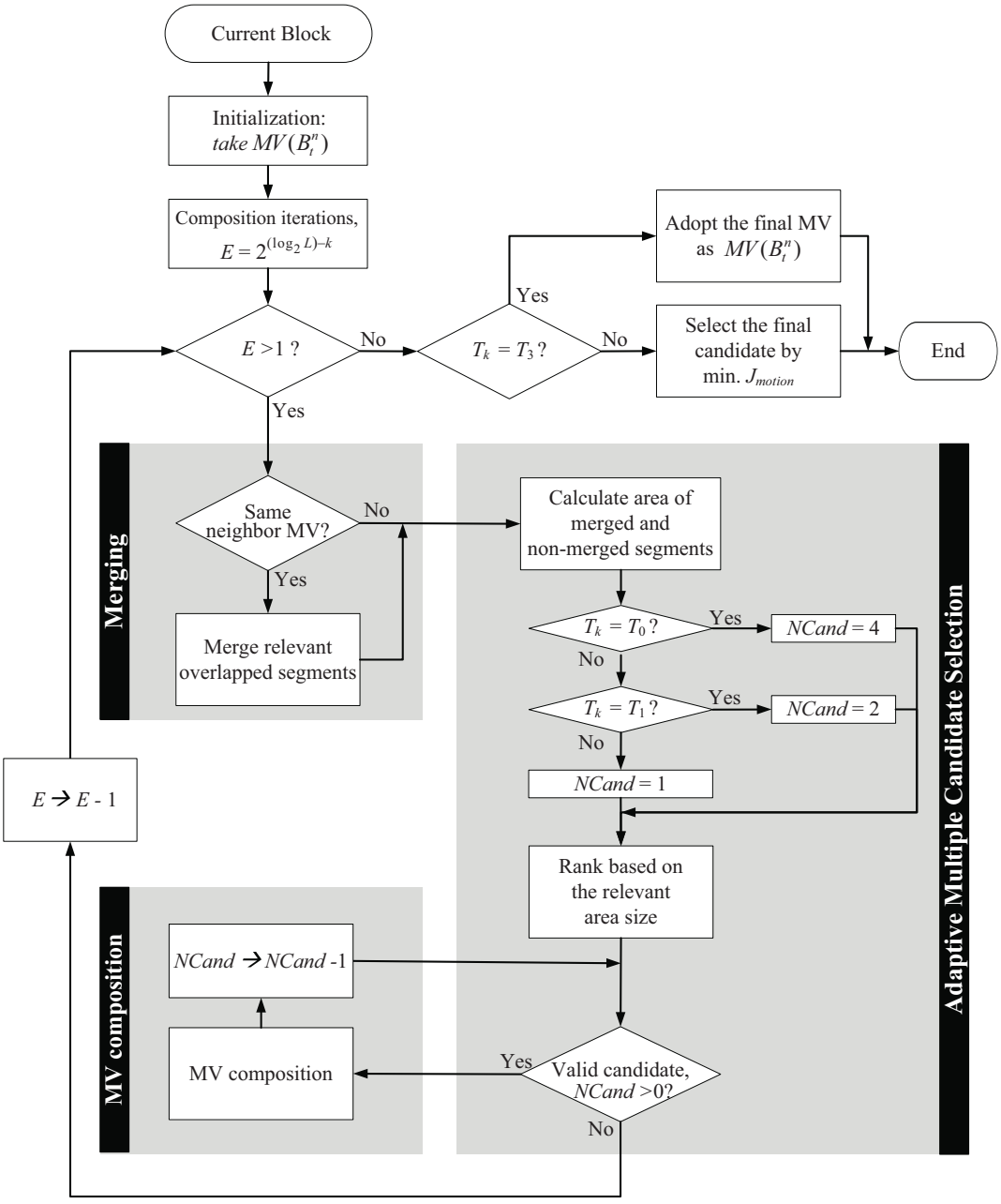


Figure 3.6: Flowchart of AMCVS for coding frames in T_k.

Table 3.2: BD and complexity measurement for T_0 & T_1: MV composition of $selMV^8_{q-1\rightarrow q}$ versus $FS_{R=8}$.

Sequences	Measurement(s)	Vector Selection Algorithms in MV composition				
		FDVS	E–FDVS	MEDIAN	AMCVS	$FS_{R=16}$
Bus	BD-Bitrate(%)	-3.52	-6.01	-0.81	-10.15	-13.55
	BD-PSNR(dB)	+0.36	+0.63	+0.09	+1.07	+1.49
	ΔComplexity(%)	+2.77	+2.77	+2.77	+2.77	+276.80
Football	BD-Bitrate(%)	-1.76	-2.78	-2.14	-4.91	-14.08
	BD-PSNR(dB)	+0.13	+0.20	+0.17	+0.36	+1.11
	ΔComplexity(%)	+2.77	+2.77	+2.77	+2.77	+276.80
Stefan	BD-Bitrate(%)	-2.17	-3.59	-0.30	-3.97	-8.21
	BD-PSNR(dB)	+0.19	+0.31	+0.02	+0.34	+0.70
	ΔComplexity(%)	+2.77	+2.77	+2.77	+2.77	+276.80

1. Table 3.2 is the BD measurement for CIF sequences. The SR of the FS algorithm was set to 8 and 16, and they are denoted by $FS_{R=8}$ and $FS_{R=16}$, respectively. For the proposed scheme with various MV composition algorithms, the SR of conducting ME in the consecutive frames was set to 8.

2. Table 3.3 is the BD measurement for 720p sequences. The SR of the FS algorithm was set to 16 and 32, and they are denoted by $FS_{R=16}$ and $FS_{R=32}$, respectively. For the proposed scheme with various MV composition algorithms, the SR of conducting ME in the consecutive frames was set to 16.

Table 3.3: BD and complexity measurement for T_0 & T_1: MV composition of $selMV^{16}_{q-1\rightarrow q}$ versus $FS_{R=16}$.

Sequences	Measurement(s)	Vector Selection Algorithms in MV composition				
		FDVS	E–FDVS	MEDIAN	AMCVS	$FS_{R=32}$
Rush hour	BD-Bitrate(%)	-3.04	-9.18	-0.36	-14.76	-16.17
	BD-PSNR(dB)	+0.15	+0.51	0	+0.88	+1.08
	ΔComplexity(%)	+0.73	+0.73	+0.73	+0.73	+280.00
Spincalendar	BD-Bitrate(%)	-23.47	-23.62	-15.94	-24.49	-23.67
	BD-PSNR(dB)	+1.54	+1.56	+1.02	+1.56	+1.43
	ΔComplexity(%)	+0.73	+0.73	+0.73	+0.73	+280.00

From Table 3.2, all MV composition algorithms in HP coding for CIF sequences in T_0 and T_1 outperform $FS_{R=8}$ by bitrate reduction and quality increment. A similar trend appears in all MV composition selection algorithms for 720p sequences when they are compared with $FS_{R=16}$ in Table 3.3. This implies that MV composition algorithms can enhance ME in T_0 and T_1 when temporal remote reference frames are used. It shows that MV composition can be successfully adopted in HP coding.

Moreover, among the MV composition algorithms in HP coding, AMCVS can provide higher coding efficiency comparing to FDVS, E–FDVS and MEDIAN. In comparison to FDVS, E–FDVS and MEDIAN, the performance in both BD-PSNR and BD-Bitrate of AMCVS for CIF sequences gets closer to $FS_{R=16}$, as shown in Table 3.2. For 720p sequences in Table 3.3, AMCVS even outperforms $FS_{R=32}$ in one testing sequence. The reason for AMCVS outperforming other MV composition

algorithms is that it only utilizes related partition of compensated area to the target block for MV selection. AMCVS enlarges the relevant partition as large as possible in every MV composition step by merging homogeneous area while ensuring the resultant MV is highly correlated to the target block in the current frame, which cannot be achieved by FDVS, E–FDVS and MEDIAN.

In Table 3.2 and Table 3.3, ΔComplexity represents the computational complexity change in percentage. The positive values mean increment whereas negative values mean decrement in complexity against $FS_{R=8}$ and $FS_{R=16}$ for CIF sequences and 720p sequences, respectively. All MV composition algorithms only require a small amount of extra complexity compared to FS with the same SR in adjacent frames by 2.77% and 0.73% for $FS_{R=8}$ and $FS_{R=16}$, respectively. On the other hand, they can successfully reduce the number of search points compared with $FS_{R=16}$ for CIF and $FS_{R=32}$ for 720p sequences without sacrificing the coding efficiency.

3.6 Chapter Summary

In this chapter, we have proposed to adopt MV composition algorithms in the HP coding framework so as to avoid using a large search window in ME in lower temporal layers. This strategy can reduce computational burdens without sacrificing the coding efficiency. Furthermore, the proposed HP coding scheme can provide better performance when utilizing the new proposed AMCVS for vector selection since it is more reliable in composing MV in low temporal layers. Simulation results showed that the proposed scheme with MV composition performs well in HP

structure coding, in which our AMCVS has been proven to outperform other MV composition algorithms. Under the same searching conditions, our entire MV composition scheme can even perform better for a long distance reference as compared with the FS algorithm under the same size of the search range.

Chapter 4

Determinations on Motion Locus by Spatial Correlation and Depth Variations

4.1 Introduction

HEVC outperforms H.264 by providing a bitrate reduction of about 50% while having almost the same perceptual quality. It adopts more flexible partitioning in motion estimation (ME) which gains higher coding efficiency at a cost of increased coding complexity. This chapter exploits depth maps in the emerging multi-view plus depth (MVD) videos. By the depth variation and the spatial correlation between blocks, the proposed algorithm determines the motion locus before ME. By the proposed motion locus, the adjustment of the search range (SR) in ME for HEVC is conducted for complexity reduction. With the aid of depth intensity variations among

spatial neighboring blocks, the proposed algorithm in this work derives weights to the neighboring blocks and establishes an adaptive search range (ASR) according to the weighted sum of the motion vectors from the neighboring blocks. The rest of this chapter is organized as follows. Section 4.2 reveals the advantage for the adoption of ASR among the existing fast ME strategies. Section 4.3 illustrates the motivation of using depth maps for SR determination. The proposed ASR determination by neighboring depth intensity weighted sum is introduced in Section 4.4. Simulation results of the proposed algorithm applied to the full-search (FS) and the fast Test Zone Search (TZS) algorithm are provided in Section 4.5. Finally, Section 4.6 summarizes this chapter.

4.2 Adaptive Search Range and Fast ME Strategies

The coding gain of HEVC is mainly from its more flexible block partitioning in ME, which is especially crucial for coding high resolution 3D videos in the MVD format [130]. However, the flexible block partitioning mechanism in HEVC induces more ME computations. In hybrid video coding, ME performs block-based search for every location within a pre-defined search range [130]. With the motion vector

predictor (MVP) from a neighbouring block as the search centre, the optimal MV is selected by minimizing the RD cost within the pre-defined search range. The true motion vector (TMV) of the current block is then formed by

$$\text{TMV} = \text{MVP} + \text{MV}. \tag{4.1}$$

HEVC utilizes an advanced motion vector predictor (AMVP) for the determination of MVP to a block as an initial search centre [14]. With a fixed search range of 64 pixels for both FS and TZS integer-pixel ME, MV is obtained from a range of [-64, +64]. TZS is one of the fast ME algorithms adopted in the HEVC test model [104] by restricting the number of search locations. In TZS, a diamond or square search pattern with various sizes is used for its centre search point initialization. However, the multiple initial search point selection is still a major burden on TZS.

Irregular Search Patterns: Other works focus on applying specific search patterns or directional search to reduce search points within a fixed search range [109]. Nevertheless, various search patterns bring irregular data flow which is not preferable for hardware implementation [110]. Besides, spatial neighbouring blocks contain highly homogenous contents to the current block; AMVP is therefore selected among their MVs. It implies if MVP is very similar to TMV, MV becomes very small as stated in (4.1). In this circumstance, the search range can be reduced adaptively. Unnecessary search point computations can therefore be avoided for saving coding time. An adaptive search range (ASR) algorithm can then deliver both search point reduction and regular data flow.

Adaptive Search Range in ME: As discussed in Section 2.4.2, some existing ASR algorithms correlate the search range of the current block with the motion characteristics of its neighboring blocks. In [110], Cauchy distribution is used to model the search range for one frame and MV differences in the neighboring blocks are used to adjust the search range for the block being encoded. In [120], the maximum difference of the estimated true MV and the optimal AMVP is used to give the ASR in HEVC. Such ASR, however, can only be determined from the results of AMVP selection. In [118], MV in the co-located block is used to define the ASR without considering whether the co-located block is within the same object. The most recent ASR algorithm in [119] adopts a linear adaptive search range model (LAM) with an overdetermined equation system. The parameters in the system can be solved if the size of PU, MVs, and predictors are given. The ASR is then adjusted by a fixed scale factor.

To the best of our knowledge, no work has noted so far to adopt the new features provided in MVD videos for defining an ASR. In this thesis, we propose to make use of depth maps in MVD videos and MVs from neighbouring blocks to yield the ASR algorithm for HEVC.

4.3 Adaptive Search Range Adjustment by Motion Locus Prediction

An object in a video frame always occupies a region covered by several blocks. It is obvious that spatial neighboring blocks contain highly homogeneous contents to

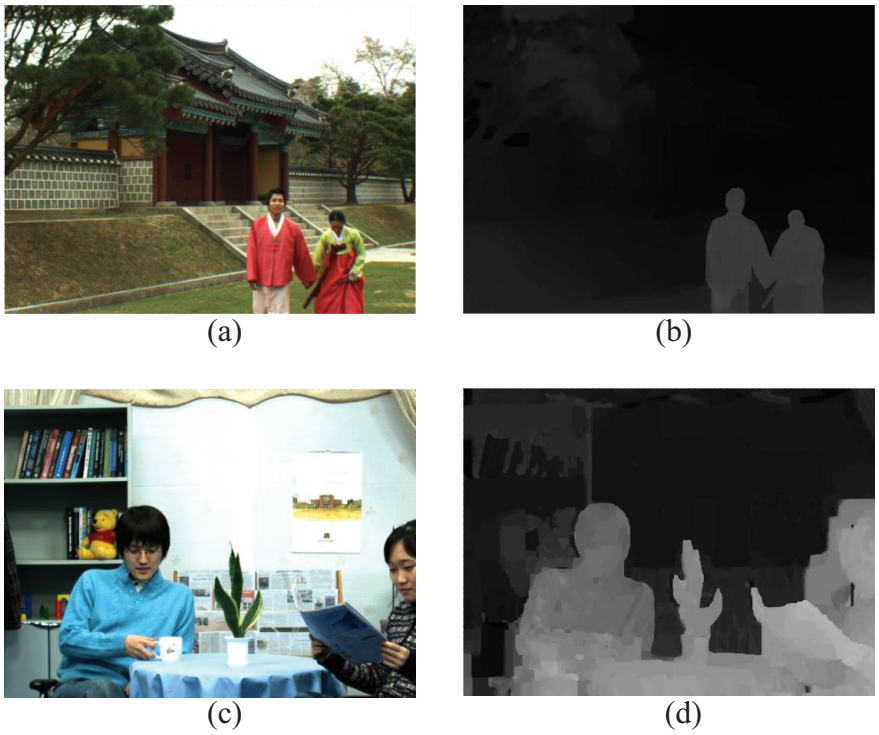


Figure 4.1: Color texture and its associated depth map for a frame. (a) Color texture in “Lovebird1”. (b) Depth map in“Lovebird1”. (c) Color texture in “Newspaper”. (d) Depth map in “Newspaper”.

the current block. Consequently, MVs of spatial neighboring blocks can be utilized to estimate the motion range of the current block. This formulates the motion locus prediction such that a suitable SR can be assigned before ME. In the proposed ASR algorithm, a SR is adaptively adjusted by determining whether the current block and its spatially neighboring blocks belong to the same object. The correlation among MVs in the same object can then be employed to specify the new SR adaptively.

In Section 1.3.4, we have reviewed MVD [39] video format which is one of the emerging 3D video representations. The MVD is composed of the color texture

and the depth map of the captured scene. Two MVD sequences are illustrated in Figure 4.1. Figures 4.1(a) and 4.1(c) show the texture streams for "Lovebird1" and "Newspaper", respectively. In depth maps of Figures 4.1(b) and 4.1(d), gray scale intensity values are assigned to represent the distance of an object from the capturing camera in a 3D scene.

In [122], the authors utilized depth information for fast mode decision. By using depth information, a video scene is divided into near, middle, and far regions. Various mode candidates of a macroblock are chosen according to the classified region the macroblock belongs to. Instead of fast mode decision, the work on this chapter proposes a depth information based ASR algorithm to adjust the SR to speed up ME. Depth maps are able to provide the additional intimation on areas/pixels belonging to objects in the same distance. In Figure 4.1(b) and Figure 4.1(d), the distinguished objects can be obviously figured out by the depth map since different video objects should have the distinct distance in the scene. In other words, the partitions which share similar motion activities in the same object can easily be identified by the depth maps. Once the spatial correlation of an object is located, a particular SR can be applied to an object in order to expedite ME. Therefore, in this chapter, depth information is suggested to be a good feature to exploit the correlation between MVs of spatially neighboring blocks for generating adaptive search range.

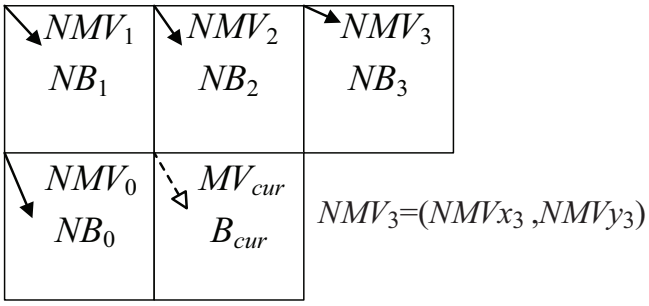


Figure 4.2: Illustration of spatial neighboring blocks with high motion homogeneity to current block and their associated MVs.

4.4 Proposed ASR by Neighboring Depth Intensity Weighted Sum for HEVC Texture Coding

The ASR algorithm proposed in this chapter takes a weighted sum of the neighboring blocks MVs to predict the SR of the current block B_{cur} in order to reduce unnecessary computations in ME. In Figure 4.2, NMV_i is the MV with two components ($NMVx_i$,$NMVy_i$) in the horizontal and vertical directions respectively from a neighboring block NB_i where i = 0, 1, 2, and 3. The new ASR of the current block denoted by $Rx(B_{cur})$ and $Ry(B_{cur})$ for the horizontal and vertical directions, respectively, can then be estimated from NMV_i in Figure 4.2, and can be written as

$$Rx(B_{cur}) = \lceil \sum_{i=0}^{3} |NMVx_i| \cdot \frac{\omega_i}{\sum_{j=0}^{3} \omega_j} \rceil, \tag{4.2}$$

and

$$Ry(B_{cur}) = \lceil \sum_{i=0}^{3} |NMVy_i| \cdot \frac{\omega_i}{\sum_{j=0}^{3} \omega_j} \rceil. \tag{4.3}$$

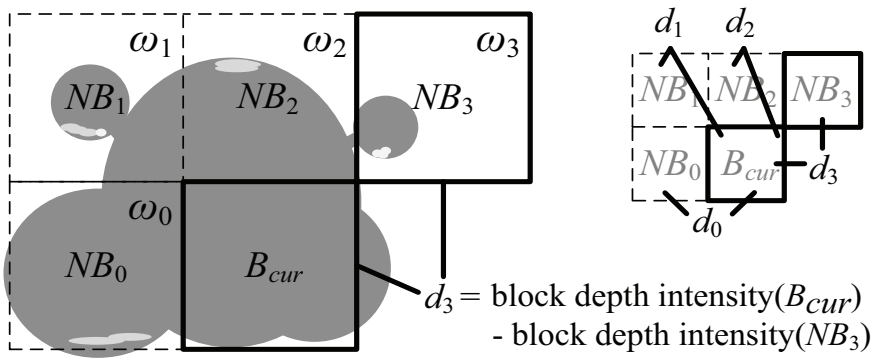


Figure 4.3: Neighboring blocks with higher similarity in block depth intensity are very likely representing the same object.

In (4.2) and (4.3), a weighted factor ω_i of each NB_i reflects the relevance of its $NMVx_i$, and $NMVy_i$ on $Rx(B_{cur})$ and $Ry(B_{cur})$ of the current block respectively, and it is formed by making use of depth information in MVD videos. By considering the characteristics of the depth maps, ω_i can be formulated by

$$\omega_i = e^{-|d_i|}, \tag{4.4}$$

where ω_i is the output of the exponential decay function of which e is Euler's number with the decay rate $\frac{1}{e}$. Using the exponential decay function aims to suppress the non-dominant blocks. The exponent $|d_i|$ is the absolute difference in average depth intensity values between the neighboring block NB_i and the current block B_{cur}. Values with small $|d_i|$ output an exponentially high ω_i and vice versa. Smaller depth intensity difference hints that the probability of NB_i and B_{cur} comprising the same object is high, which reflects this NB_i is more correlated to B_{cur}. The rationale is that a depth map can presumably reveal the object distance within a 3D space.

Therefore, it is a piece of indicative information to decide which blocks belong to the same object as illustrated in Figure 4.3. In the proposed algorithm, a higher weight ω_i will be issued to MV in which its associated block NB_i represents higher content similarity to B_{cur} (i.e. a smaller value of $|d_i|$). From the example in Figure 4.3, $|d_0| < |d_2| < |d_1| < |d_3|$ is observed. It can be concluded that $\omega_0 > \omega_2 > \omega_1 > \omega_3$. Therefore NB_0 is closest to B_{cur} in terms of depth distance and contents. The ASR is then determined based on the amplitude of the weighted MVs of the neighboring blocks (i.e. NMV_0 has a stronger influence than other MVs in this example). Finally, MV of B_{cur}, MV_{cur} , can be obtained by ME with the horizontal and vertical SRs as [- $Rx(B_{cur})$, + $Rx(B_{cur})$] and [- $Ry(B_{cur})$, + $Ry(B_{cur})$], respectively.

4.5 Simulation Results and Discussions

4.5.1 Simulation Conditions

The proposed ASR algorithm using neighboring depth intensity weighted sum has been integrated into the HM 14.0 reference software, and is referred to as NDIWS. Its asymmetric ASR for FS was compared with the conventional FS using a fixed SR of [-64, +64] and the most recent LAMASR algorithm for ASR [119]. It is noted that TZS is designed for squared search windows. Therefore, the SR using NDIWS was computed as max($Rx(B_{cur})$,$Ry(B_{cur})$) when TZS with NDIWS (TZS+NDIWS) was tested, where max( ) is the maximum function aiming at bounding all probable movement among the x and y directions. TZS+NDIWS was further compared to the conventional TZS with the fixed SR and the ASR determined by

LAMASR [119]. All tested algorithms were evaluated with four QPs of 22, 27, 32, and 37 under the low-delay P configuration specified in the common test condition of HEVC [104]. Full quad-tree structure for all CU, PU, and TU was utilized. Bjontegaard (BD) measurement in terms of BD-rate (%) and BD-PSNR (dB) were used to measure the average coding efficiency, and Δtime (%) represents coding time change in percentage as compared with the benchmarking algorithms. Positive and negative values denote increments and decrements, respectively. The test platform used for simulations was a 64-bit MS Windows 8.1 OS running on an Intel Core i7-4770 CPU of 3.4 GHz and 16.0 GB RAM.

4.5.2 Performance evaluation of proposed NDIWS in FS

Table 4.1 lists the performance of NDIWS compared to FS with the fixed SR. It averagely saves 95% coding time over FS while its BD-PSNR drops 0.02dB and its BD-rate increases by 0.64%. In comparison to LAMASR [119], NDIWS saves encoding time by 31% while only introducing an insignificant BD-rate increase of 0.06%. From Table 4.1, the proposed NDIWS saves more time as its asymmetric SR considers movement in the horizontal and vertical directions separately. It is due to the fact that most of the objects do not move diagonally.

4.5.3 Performance evaluation of proposed NDIWS in TZS

By integrating the proposed ASR into TZS, Table 4.2 showed that TZS+NDIWS reduces 65% time on average compared to the conventional fixed search range TZS. The corresponding BD-PSNR decreases by 0.02dB while the BD-rate increases by

Table 4.1: Performance evaluation of proposed NDIWS to conventional fixed search range FS and existing fast algorithm LAMASR in HM14.0.

Sequences	NDIWS compared to					
	FS using Fixed SR			FS using LAMASR		
	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
Balloons	-96.04	-0.01	+0.38	-68.18	-0.01	+0.25
Kendo	-98.50	-0.03	+0.85	-39.71	0	+0.06
Lovebird1	-99.26	-0.01	+0.24	+5.13	0	-0.07
Newspaper	-98.47	-0.01	+0.43	-40.47	0	+0.13
Poznan_Hall2	-92.22	-0.01	+0.56	-21.56	0	+0.02
Undo_Dancer	-91.06	-0.05	+1.43	-38.19	0	-0.03
GT_Fly	-89.65	-0.02	+0.60	-20.17	0	+0.05
Average	-95.03	-0.02	+0.64	-31.88	0	+0.06

0.53%. As compared with TZS using LAMASR [119], TZS+NDIWS reduces averagely 26% of coding time while only introducing an insignificant BD-rate increase of 0.12%. The reason is that LAMASR [119] only formulates a fixed relationship of the motion information by offline trainings. Instead, TZS+NDIWS utilizes the depth intensity correlation for the current block from its neighboring blocks adaptively in order to reduce unnecessary computations with insignificant BD loss.

Table 4.2: Performance evaluation of proposed NDIWS to conventional fixed search range TZS and existing fast algorithm LAMASR in HM14.0.

Sequences	TZS+NDIWS compared to					
	TZS using Fixed SR			TZS using LAMASR		
	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
Balloons	-73.07	-0.01	+0.36	-37.86	0	+0.07
Kendo	-67.89	-0.03	+1.09	-29.63	-0.01	+0.42
Lovebird1	-84.71	-0.02	+0.49	-54.29	-0.01	+0.37
Newspaper	-82.83	-0.02	+0.54	-55.63	-0.01	+0.40
Poznan_Hall2	-67.81	0	+0.13	-33.45	0	+0.12
Undo_Dancer	-43.30	-0.03	+0.90	-18.44	+0.01	-0.32
GT_Fly	-40.87	-0.01	+0.19	+40.69	+0.01	-0.19
Average	-65.78	-0.02	+0.53	-26.94	0	+0.12

4.6 Chapter Summary

In this chapter, an ASR algorithm has been proposed by considering depth information. The ASR is determined by a weighted sum of the neighboring blocks MVs in which their weights depend on the absolute difference of depth intensity values between the neighboring blocks and the current block. It results in a complexity reduction. The proposed ASR is compatible with FS and other fast search algorithms such as TZS in HEVC. Simulation results demonstrated that it is able to reduce 65% of coding time on average in the fast TZS with negligible BD loss.

Chapter 5

Depth-based Motion Locus for Texture Coding

5.1 Introduction

We have observed from the previous chapter that the use of the spatial correlation in depth maps is able to adjust a search range for motion estimation (ME) in HEVC. As aforementioned, the depth map provides an intimation of the objects' distance from the projected screen in a 3D scene, which is very suitable to explore in adaptive search range (ASR) determination. Chapter 4 utilizes all the MV amplitudes from the most spatially nearest blocks to formulate the ASR. Different from the work in Chapter 4, the work in this chapter selectively maps the most correlated candidate blocks in temporal domain by the depth intensity values. It further exposes the usage of the temporal correlation in depth maps for relieving the computational burden of HEVC. By utilizing this correlation, a depth/motion relationship map is built for

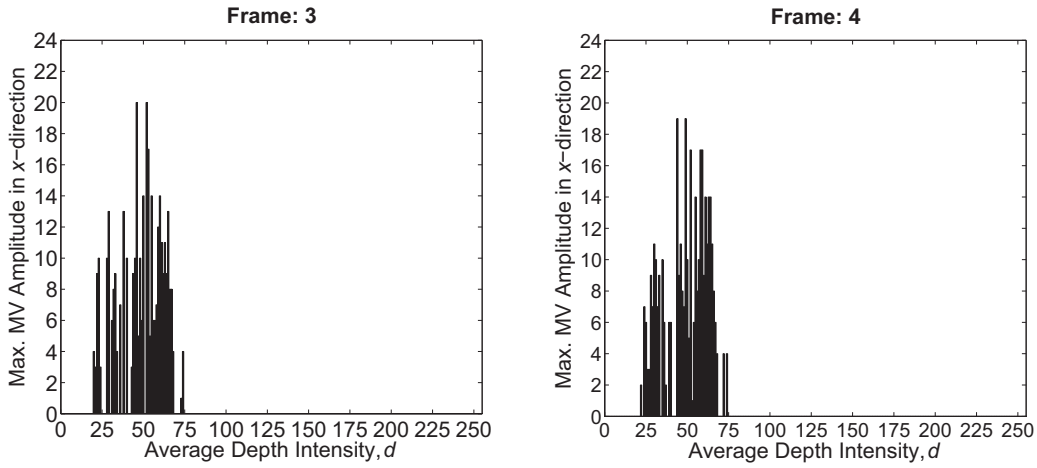
a mapping process. For each block, this forms a tailor-made search range with a motion-aware asymmetric shape to skip unnecessary search points in ME. The obtained search range (SR) can be further adjusted by taking the influence of 3D-to-2D projection into consideration. Besides, the proposed SR determination can work well with other fast search ME algorithms in the literature.

The rest of this chapter is organized as follows. Section 5.2 exploits the temporal correlation between the depth map and the motion in texture. The correlation makes the development of a new ASR algorithm possible for speeding up the ME process in HEVC. The proposed idea of linkage between depth maps and motions for the purpose of SR adjustment is then introduced in Section 5.2. First, we describe the construction of a depth/motion relationship map (DMRMap) based on the correlation between the depth map and the motion in texture. Second, by making use of the DMRMap, the retrieval of ASR for the block being encoded is presented. Furthermore, the final adjustment of the SR due to the influence of the 3D space to the 2D image plane projection is discussed in Section 5.3. The entire proposed depth-based ASR algorithm based on the construction of DMRMap and the SR adjustment due to 3D-to-2D projection is conveyed in Section 5.4. Simulation results of the proposed algorithm are provided in Section 5.5. Finally, Section 5.6 concludes this chapter.

2016)

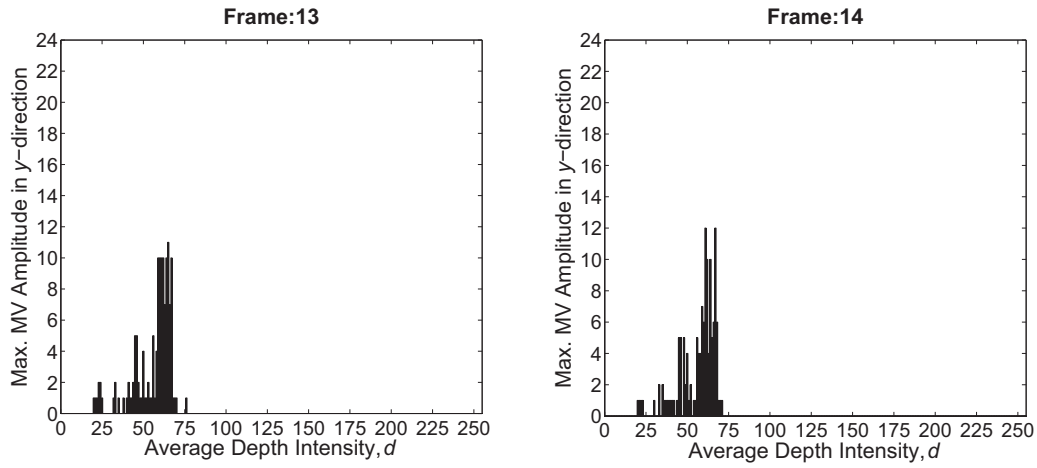
- **Tsz-Kwan Lee**, Yui-Lam Chan, and Wan-Chi Siu, "Depth-based Adaptive Search Range Algorithm for Motion Estimation in HEVC," in *Proceedings of International Conference on Digital Signal Processing (DSP 2014)*, Hong Kong, Aug. 2014, pp.919-923.

5.2 Temporal Correlation between Depth Map and Motion in Texture Streams



(a) (b)

Figure 5.1: The maximum amplitude of x-component motion vectors MV in quarter pixel of color texture for various average depth intensity values between consecutive frames, (a) Frame 3 and (b) Frame 4 of "Lovebird1".



(a) (b)

Figure 5.2: The maximum amplitude of y-component motion vectors MV in quarter pixel of color texture for various average depth intensity values between consecutive frames, (a) Frame 13 and (b) Frame 14 of "Lovebird1".

The motivation of using temporal correlation between the depth maps and motion in texture are depicted in Figures 5.1(a) and 5.1(b), which plot the maximum amplitude of MVs of color texture in the x-direction for various average depth intensity values of all blocks in two consecutive frames. Meanwhile, Figures 5.2(a) and 5.2(b) show the maximum amplitude of MV of color texture in the y-direction for various average depth intensity. From these graphs, it can be seen that they have very similar distribution. It is because the depth information of an object not only represents the physical object position but also exhibits the motion activities of the object itself on each frame. It reflects that blocks with similar average depth intensity value will usually have similar MVs over a period of time. By making use of this temporal correlation between the depth map and motion in texture, we establish

depth and motion relationship for each frame, and it is referred to as a depth/motion relationship map (DMRMap).

With the aid of the relationship map, motion activities of objects between consecutive frames could be roughly predicted by depth maps. In this chapter, the SR is adopted according to the proposed DMRMap. Therefore, unnecessary search points within the pre-defined SR can be removed.

5.2.1 DMRMap Construction in Reference Frame

This chapter proposes a framework to obtain and maintain the DMRMap of a reference frame, which can be used to determine the SR of the current frame. The DMRMap captures the relationship between motion activity and average depth intensity of all blocks in a reference frame. The proposed algorithm should start with any frame other than the first inter-frame because the reference frame of the first inter-frame is intra-coded, and no MVs from this reference frame can be obtained. Therefore, the first inter-frame will go through a conventional full rate-distortion optimization (RDO) inter coding. Once the MVs of all blocks in the frame are obtained, its DMRMap is constructed for ME of the next frame. Let d be the average depth intensity values of a block in the reference frame, where $0 \leq d \leq 255$. It is noted that depth maps are always estimated using stereo matching methods [133], which induces slight variation or noise of depth values within the same object. To tolerate the variation of depth values in an object, d is divided into an appropriate number of ranges, each containing many similar values of d. To do so, d is quantized uniformly by a quantization factor Q into $\hat{d}$, where $0 \leq \hat{d} \leq \lceil 255/Q \rceil$. Note that $\lceil\ \rceil$

is the ceiling function. The DMRMap relates the largest MVs to all possible values of $\hat{d}$ in the reference frame. Assume that $S_x^{\hat{d}}$ and $S_y^{\hat{d}}$ are the sets of MVs in the x- and y-direction, respectively, with the blocks in which their quantized average depth intensity value is $\hat{d}$. The largest MV amplitudes, $(MVx^{\max}(\hat{d}), MVy^{\max}(\hat{d}))$, in the x- and y-directions are respectively the maximum values in $S_x^{\hat{d}}$ and $S_y^{\hat{d}}$ as

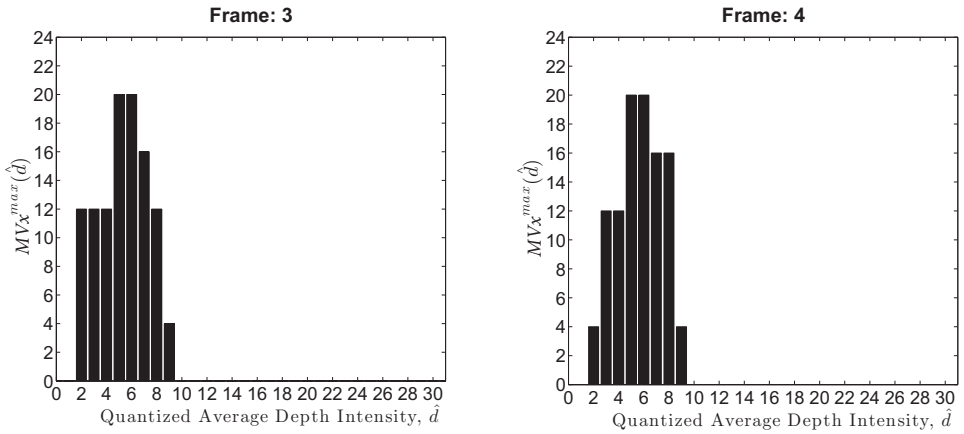
$$MVx^{\max}(\hat{d}) = \max(S_x^{\hat{d}}) \tag{5.1}$$

and

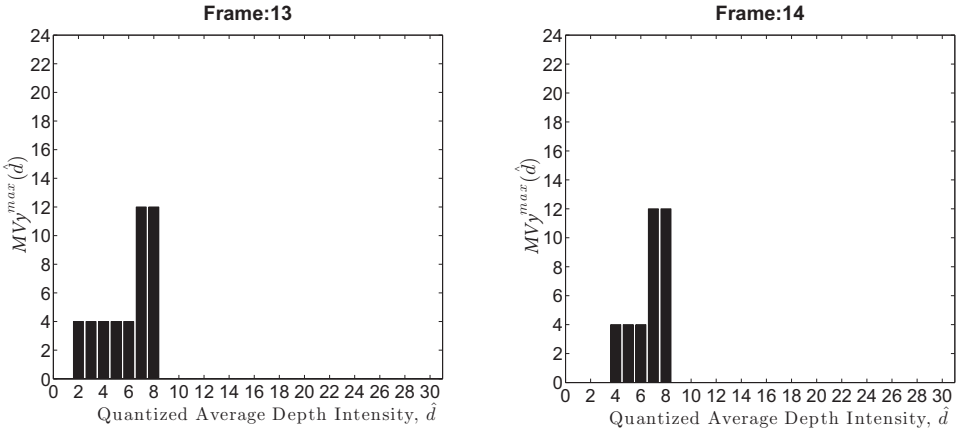
$$MVy^{\max}(\hat{d}) = \max(S_y^{\hat{d}}), \tag{5.2}$$

where $\max(S)$ gives the maximum value of the set S. Actually, $MVx^{\max}(\hat{d})$ and $MVy^{\max}(\hat{d})$ can be used to describe the DMRMap, which constructs the relationship between the largest MVs and $\hat{d}$. In other words, the largest MV in both of the x- and y-directions can be determined for the given $\hat{d}$.

The DMRMap will be updated frame by frame. Two relationship maps in the x- and y-directions constructed from a pair of consecutive frames for "Lovebird1" are illustrated in Figures 5.3(a) and 5.3(b), respectively, in which Q is set to 8. They record $MVx^{\max}(\hat{d})$ and $MVy^{\max}(\hat{d})$ for each $\hat{d}$ within the frame, where $\hat{d}$ is from 0 to 31. The value of Q depends on the level of noise in the depth map. The more the noise in the depth map, the larger the value of Q is used to absorb depth variation in the same object. However, a large Q results in affecting the precision of DMRMap, and detailed discussion will be given in Section 5.5. Setting Q to 8 is always appropriate for the quality of most depth maps recommended by the ISO/IEC and ITU-T JCT-3V group with the reasonably good DMRMap. From Figures 5.3(a) and 5.3(b)



(a)



(b)

Figure 5.3: The largest motion vector amplitudes (a) $MVx^{\max}(\hat{d})$ in the x-direction, and (b) $MVy^{\max}(\hat{d})$ in the y-direction from a pair of consecutive frames for "Lovebird1".

for the DMRMap, it can be observed that the distributions are very similar to each other for consecutive frames in both of the horizontal and vertical movements.

5.2.2 ASR Decision based on Mapping Process using DMRMap

By utilizing the temporal correlation of DMRMaps between two consecutive frames, the mapping process from the average depth intensity for the n^{th} block being encoded in frame t, B_t^n, to its SR, denoted as $Rx(B_t^n)$ and $Ry(B_t^n)$ in the x- and y-directions, respectively, is conducted. The mapping is based on the DMRMap in the reference frame as defined in (5.1) and (5.2). Let $QDepth(B_t^n)$ be the average depth intensity values after quantization for B_t^n. From (5.1) and (5.2), $Rx(B_t^n)$ and $Ry(B_t^n)$ can respectively be computed as

$$Rx(B_t^n) = MVx^{\max}(QDepth(B_t^n)) \tag{5.3}$$

and

$$Ry(B_t^n) = MVy^{\max}(QDepth(B_t^n)). \tag{5.4}$$

This mapping process is to correlate the temporal information by the average depth intensity value of B_t^n being encoded to those blocks in the reference frame. Since depth maps indicate the location of an object in the video scene from the image plane, the average depth intensity value could therefore be a criterion for distinguishing various objects with different distances from the camera in a video scene. Based on this observation, it is likely that the blocks belonging to one particular video object across consecutive frames have consistent motion associated with the similar average depth intensity values. Once the average depth intensity value of the current block, B_t^n, is matched to those in the same intensity value of the reference frame, ASR decision can be made from the DMRMap in the reference frame. It is noted that, if $QDepth(B_t^n)$ is empty in the DMRMap of the reference frame, the

SRs of B_t^n in both x- and y-directions are set to 64. It is the default SR of the main profile in HEVC, which is larger or equal to the values obtained in (5.3) and (5.4). After all MVs of the frame being encoded are determined, the DMRMap is updated for both x- and y-directions for the next frame.

5.3 Influence of 3D-to-2D Projection on Motion Activity on 2D Image Plane

The working principle of the mapping process in (5.3) and (5.4) is based on the very strong temporal correlation of DMRMaps between the current and reference frames. However, an object moving towards and away from the camera, or zoom effect from the camera changes the distance between the object being captured and the camera between frames. This motion activity along the camera axis (z-axis) has the potential to weaken the degree of this correlation, which reduces the prediction accuracy of the SR in (5.3) and (5.4). Taking this into consideration, a scale factor for the proposed ASR of B_t^n, $\rho(B_t^n)$, is added to offer extra flexibility in the determination of the SR in (5.3) and (5.4). The SR prediction is then scaled as

$$Rx^{\rho}(B_t^n) = \rho(B_t^n) \times MVx^{\max}(QDepth(B_t^n)) \tag{5.5}$$

and

$$Ry^{\rho}(B_t^n) = \rho(B_t^n) \times MVy^{\max}(QDepth(B_t^n)). \tag{5.6}$$

Note that $\rho(B_t^n) = 1$ is the case for the scene without motion along the z-axis. In this case, (5.5) and (5.6) are equal to (5.3) and (5.4), respectively. The change in

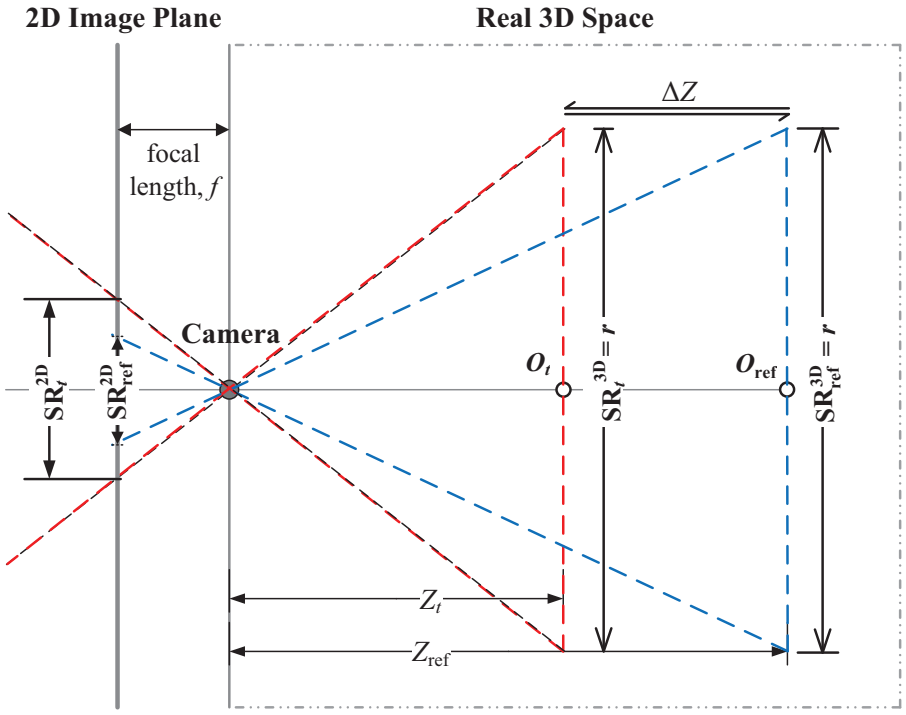


Figure 5.4: Geometric relationship between depth of object and motion activity on the 2D image plane.

depth intensity between frames actually reflects the degree of z-axis motion, which in turn gives a good estimation of $\rho(B_t^n)$. The way to determine $\rho(B_t^n)$ is underlying on the motion parallax. It states that, given the same horizontal or vertical motions of objects in the 3D space, objects that are closer to the camera move faster on the 2D image plane than the objects that are farther. In other words, the degree of the projected displacement of an object on the 2D image plane is always influenced by how close it is located to the camera in the 3D space. When the object is closer to the camera, projected displacement on the 2D image plane is larger. This situation is illustrated in Figure 5.4. In this figure, an example of the geometric relationship

between the depth information of an object moving towards the camera and its displacement variation on the 2D image plane is depicted. Let O_{ref} denote a 3D position of an object in the 3D space. It is noted that the subscript "ref"' represents the reference position in the following discussion. The actual distance value between O_{ref} and the camera is Z_{ref}, and its SR is assumed to be $SR_{\text{ref}}^{\text{3D}}$ in the 3D space. Assume that O_{ref} moves to O_t with the actual distance value of Z_t at time t. In this case of the object moving towards the camera, Z_t is smaller than Z_{ref} since O_t is closer to the camera than O_{ref}, as shown in Figure 5.4. Similarly, the SR of O_t is $SR_{\text{ref}}^{\text{3D}}$. In Figure 5.4, $SR_{\text{ref}}^{\text{2D}}$ and SR_t^{2D} are therefore the projections of $SR_{\text{ref}}^{\text{3D}}$ and SR_t^{3D} on the 2D image plane, respectively. For the same SR for O_{ref} and O_t in the 3D space (i.e., $SR_{\text{ref}}^{\text{3D}} = SR_t^{\text{3D}} = r$), $SR_{\text{ref}}^{\text{2D}}$ is smaller than SR_t^{2D} on the 2D image plane after projection, as illustrated in Figure 5.4. Consequently, this phenomenon can be used to determine $\rho(B_t^n)$. In (5.5) and (5.6), $\rho(B_t^n)$ is a factor to scale the SR projected on the 2D image plane at time t due to the motion along the z-axis, which is the ratio of SR_t^{2D} to $SR_{\text{ref}}^{\text{2D}}$ defined by

$$\rho(B_t^n) = \frac{SR_t^{\text{2D}}}{SR_{\text{ref}}^{\text{2D}}}. \tag{5.7}$$

The dotted lines in Figure 5.4 indicate the trajectory of the projections through the camera lens onto the 2D image plane. Using triangular similarity, the relationship between $SR_{\text{ref}}^{\text{2D}}$ and SR_t^{2D} can be correlated to the actual distances, Z_{ref} and Z_t in the 3D space as

$$\frac{SR_{\text{ref}}^{\text{2D}}}{f} = \frac{r}{Z_{\text{ref}}} \tag{5.8}$$

and

$$\frac{SR_t^{\text{2D}}}{f} = \frac{r}{Z_t}, \tag{5.9}$$

where f is the focal length of the camera, and r is the SR amplitude of the object in the 3D space. By combining (5.7), (5.8), and (5.9), $\rho(B_t^n)$ can be formulated as

$$\rho(B_t^n) = \frac{Z_{\text{ref}}}{Z_t} = 1 + \frac{\Delta Z}{Z_t}, \tag{5.10}$$

where ΔZ is the change in the actual distance between O_{ref} and O_t in the 3D space due to the z-axis motion of the object, as shown in Figure 5.4. The positive ΔZ means the object moving towards the camera since physically $Z_t < Z_{\text{ref}}$ while the negative Z means the object moving away from the camera due to $Z_t > Z_{\text{ref}}$. In addition, there is no z-axis motion when ΔZ is equal to zero.

In (5.10), it introduces the scale factor based on the changes in the actual distance between the current and reference blocks. The actual distances, Z_t and Z_{ref}, in the 3D space can be computed from the average depth intensity values without quantization in the depth maps, $Depth(B_t^n)$ and $Depth(B_{\text{ref}}^n)$, for B_t^n in the current frame and its co-located block B_{ref}^n in the reference frame, respectively, as

$$Z_t = 1/\left[\frac{Depth(B_t^n)}{255} \times \left(\frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}}\right) + \frac{1}{Z_{\text{far}}}\right] \tag{5.11}$$

and

$$Z_{\text{ref}} = 1/\left[\frac{Depth(B_{\text{ref}}^n)}{255} \times \left(\frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}}\right) + \frac{1}{Z_{\text{far}}}\right], \tag{5.12}$$

where Z_{near} and Z_{far} are, respectively, the smallest and the largest actual distances among all points captured by the camera, which are recorded in the camera configure

Table 5.1: Values of Z_{near} and Z_{far} in various sequences

Sequences	Z_{near} (cm)	Z_{far} (cm)
Balloons	448.251214	11206.280350
Kendo	448.251214	11206.280350
Lovebird1	-2228.745812	-156012.206815
Newspaper	-2715.181648	-9050.605493
Poznan_Street	-34.506386	-2760.510889
Poznan_Hall2	-23.394160	-172.531931
Undo_Dancer	2289	213500
GT_Fly	changes every frame between a range of $\{Z_{\text{near}}, Z_{\text{far}}\}$ ={3156.3, 1000000000}	

files of the test sequences recommended by the ISO/IEC and ITU-T JCT-3V group. Their values of Z_{near} and Z_{far} are listed in Table 5.1 in which positive or negative values denote the viewing direction of the camera. It is noted that the values of Z_{near} and Z_{far} are signaled with the 3D videos for a correct geometric displacement in synthesized intermediate views [134] at the decoder side. By putting (5.11) and (5.12) into (5.10), $\rho(B_t^n)$ is expressed as

$$\rho(B_t^n) = \frac{Depth(B_t^n)(Z_{\text{far}} - Z_{\text{near}}) + 255 \times Z_{\text{near}}}{Depth(B_{\text{ref}}^n)(Z_{\text{far}} - Z_{\text{near}}) + 255 \times Z_{\text{near}}}, \tag{5.13}$$

and it can be summarized as

$$\begin{cases} \text{case 1} : \rho(B_t^n) > 1, & Depth(B_t^n) > Depth(B_{\text{ref}}^n) \\ \text{case 2} : \rho(B_t^n) = 1, & Depth(B_t^n) = Depth(B_{\text{ref}}^n) \\ \text{case 3} : \rho(B_t^n) < 1, & Depth(B_t^n) < Depth(B_{\text{ref}}^n) \end{cases}, \tag{5.14}$$

where case 1 represents a scenario that the object moving towards the camera, case 2 represents the object without z-axis motion, and case 3 represents the object moving away from the camera.

5.4 The Proposed DMRMap-based ASR Algorithm

Figure 5.5 shows the flowchart of the proposed DMRMap-based ASR algorithm to encode a frame in HEVC. The proposed ASR algorithm has three new features: (a) DMRMap construction of the reference frame; (b) ASR determination using DMRMap; and (c) ASR update based on 3D-to-2D motion projection. Combining these three techniques, the proposed DMRMap-based ASR algorithm can be applied to the block B_t^n being encoded as follows.

Step (i): Construct the DMRMap of the reference frame by (5.1) and (5.2).

Step (ii): Compute $QDepth(B_t^n)$ for the mapping process.

Step (iii): If $QDepth(B_t^n)$ is available in DMRMap, go to Step (iv); otherwise, go to Step (vii).

Step (iv): Obtain the horizontal $Rx(B_t^n)$, and the vertical $Ry(B_t^n)$ based on the mapping processing in (5.3) and (5.4), respectively.

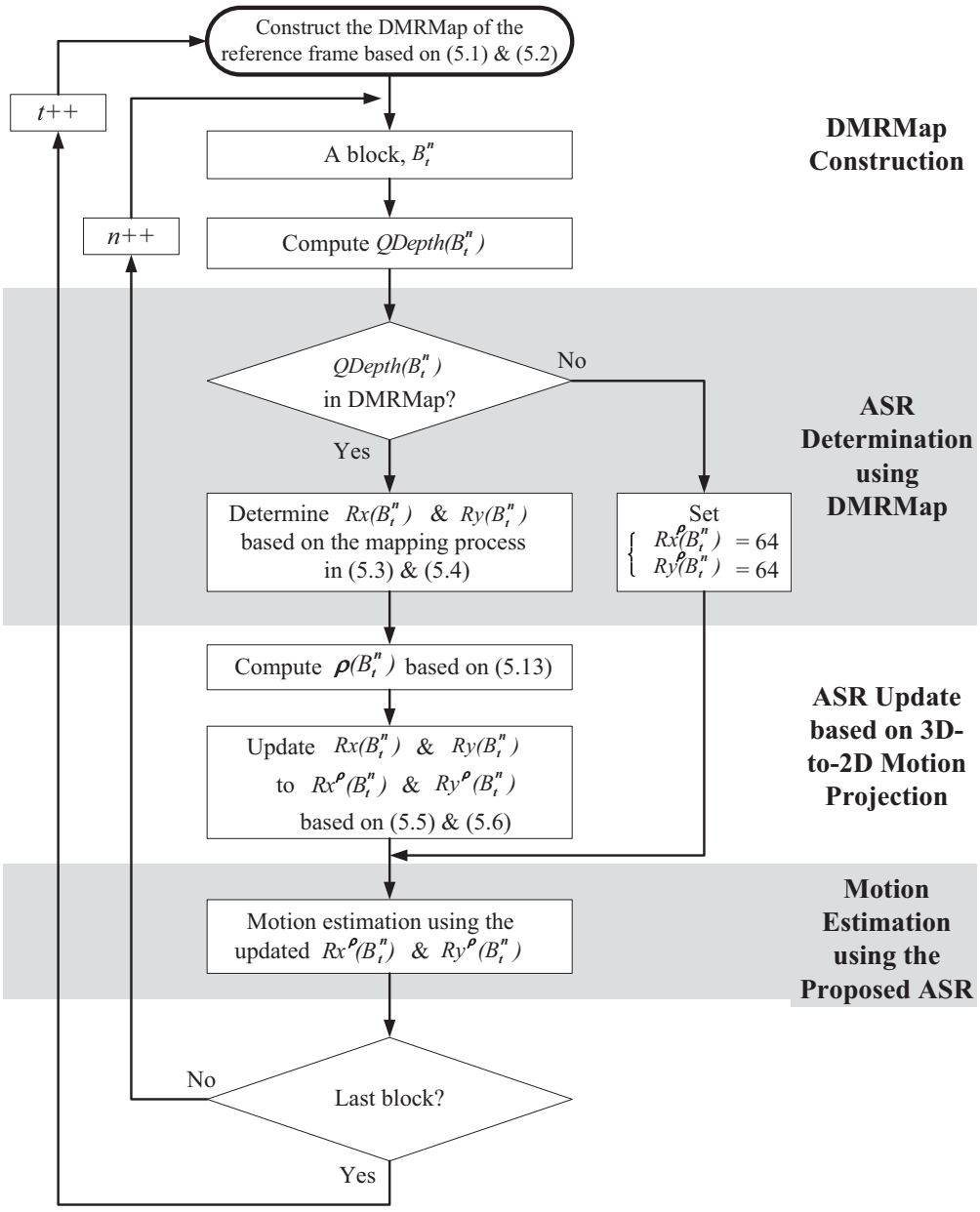


Figure 5.5: Flowchart of the proposed DMRMap-based ASR algorithm.

Step (v): Determine the scale factor $\rho(B_t^n)$ as (5.13).

Step (vi): Update the horizontal $Rx(B_t^n)$ to $Rx^{\rho}(B_t^n)$ and the vertical $Ry(B_t^n)$ to $Ry^{\rho}(B_t^n)$ according to (5.5) and (5.6), respectively by $\rho(B_t^n)$. Go to Step (viii).

Step (vii): Set both of the horizontal $Rx^{\rho}(B_t^n)$ and the vertical $Ry^{\rho}(B_t^n)$ to 64.

Step (viii): Perform ME using $Rx^{\rho}(B_t^n)$ and $Ry^{\rho}(B_t^n)$.

5.5 Simulation Results and Discussions

To evaluate the performance of the DMRMap-based ASR algorithm, the techniques proposed in Section 5.2 and Section 5.3 have been integrated into the HM 14.0 reference software [135], and tested under the low-delay P configuration specified in the common test condition [136] of the HEVC standardization in which the main profile of HEVC was used. An I-frame was allowed in the first frame only, and the rest were encoded as P-frames. All CU-level of 64×64, 32×32, 16×16, and 8×8 were enabled. For PU and TU, a full quad-tree structure was utilized. All tested algorithms were evaluated with four QPs of 22, 27, 32, and 37 using eight test sequences with two resolutions of 720p and 1080p.

5.5.1 Simulation Conditions

Two sets of experiments were performed to evaluate the overall efficiency of applying our proposed DMRMap-based ASR algorithm to various ME search strategies. First, the proposed DMRMap algorithm with and without the scale factor

$\rho(B_t^n)$ in (5.13) have been incorporated into the conventional full-search (FS) in order to provide ASR for ME, and let us call them FS+DMRMap+Scaling and FS+DMRMap, respectively. Q was set to 8 in both of FS+DMRMap+Scaling and FS+DMRMap. Three most recent ASR algorithms [117–119] have also been implemented for comparisons, and they are referred to as FS+MLELD [117], FS+LSMF [118], and FS+LAMASR [119], respectively. Second, we demonstrate the performance of the proposed DMRMap applied to the Test Zone Search (TZS), named as TZS+DMRMap+Scaling. TZS was employed in the H.264 joint scalable video model (JSVM) [34], and TZS is also the only fast method adopted in the HEVC reference software [104, 105]. It can be proved that our proposed DMRMap-based algorithm is compatible to a fast ME algorithm.

In all simulations, Bjontegaard (BD) measurement [123] in terms of BD-rate (%) and BD-PSNR (dB) were used to measure the average coding efficiency of various algorithms, and Δtime (%) represents coding time change in percentage as compared with the benchmarking algorithms. Positive and negative values denote increments and decrements, respectively. Note that the coding time includes the computational cost for all CU quad-tree levels. The test platform used for simulations was a 64-bit MS Windows 8.1 OS running on an Intel Core i7-4770 CPU of 3.4 GHz and 16.0 GB RAM.

5.5.2 Results of Applying DMRMap to FS

The full-search (FS) algorithm gives the best and optimal rate-distortion (RD) performance in block-based ME since it searches all points inside the predefined SR.

Table 5.2: Bjontegaard (BD) measurement and coding time change of FS+LSMF, FS+MLELD, FS+LAMASR, FS+DMRMap, and FS+DMRMap+Scaling for ASR against FS in HEVC

Seq.	QP	FS		FS+LSMF			FS+MLELD			FS+LAMASR			FS+DMRMap			FS+DMRMap+Scaling		
		PSNR (dB)	Bitrate (kbps)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
720p																		
Balloons	37	38.38	335.94	-40.54	+0.01	+0.13	-63.17	0.00	+0.07	-74.01	0.00	-0.06	-93.43	-0.01	+0.18	-89.63	-0.01	+0.26
	32	41.24	593.35															
	27	43.56	1171.64															
	22	45.46	3134.29															
Kendo	37	39.66	372.48	-53.53	0.00	+0.07	-80.74	-0.01	+0.23	-89.99	-0.01	+0.25	-93.01	-0.01	+0.30	-94.01	-0.01	+0.26
	32	42.27	654.66															
	27	44.44	1236.53															
	22	46.29	2900.38															
Lovebird1	37	34.32	164.72	-71.10	-0.01	+0.35	-87.87	0.00	+0.12	-94.96	0.00	+0.06	-99.42	0.00	+0.11	-99.46	0.00	+0.08
	32	37.22	353.62															
	27	40.29	830.90															
	22	43.64	2069.84															
Newspaper	37	35.71	242.51	-54.77	-0.02	+0.45	-73.84	-0.02	+0.47	-90.94	-0.01	+0.23	-95.82	-0.01	+0.39	-97.23	-0.01	+0.31
	32	38.45	451.86															
	27	41.08	960.40															
	22	43.81	2711.91															
1080p																		
Poznan_Street	37	35.10	461.62	-39.99	-0.01	+0.55	-42.87	-0.01	+0.51	-69.47	-0.02	+0.74	-96.38	-0.01	+0.48	-97.15	0.00	+0.17
	32	37.48	1142.23															
	27	39.94	4223.58															
	22	43.24	24872.19															
Poznan_Hall2	37	39.59	237.06	-22.98	0.00	+0.05	-41.25	-0.01	+0.28	-40.37	-0.01	+0.22	-85.24	-0.01	+0.50	-86.72	-0.01	+0.41
	32	41.09	501.60															
	27	42.25	1568.65															
	22	44.24	13711.65															
Undo_Dancer	37	33.01	1525.83	-25.14	-0.01	+0.42	-36.78	-0.01	+0.39	-34.74	-0.02	+0.69	-94.59	-0.01	+0.29	-91.74	-0.01	+0.30
	32	35.76	4303.43															
	27	39.02	11217.01															
	22	42.69	25191.12															
GT_Fly	37	35.82	1166.74	-25.41	0.00	0.00	-27.32	0.00	+0.03	-36.41	0.00	+0.14	-96.56	-0.01	+0.21	-95.47	-0.01	+0.27
	32	38.44	3023.78															
	27	41.10	7445.85															
	22	43.85	16927.17															
Average:				**-41.68**	**-0.01**	**+0.25**	**-56.73**	**-0.01**	**+0.26**	**-66.36**	**-0.01**	**+0.28**	**-94.31**	**-0.01**	**+0.31**	**-93.93**	**-0.01**	**+0.26**

The objective of the proposed DMRMap-based ASR algorithms is to provide a suitable and reasonable SR for both vertical and horizontal directions per block for ME in HEVC. As a result, unnecessary search points can be skipped such that better resource utilization in ME can be achieved.

Table 5.2 lists the BD measurement and Δtime of our proposed FS+DMRMap and FS+DMRMap+Scaling against FS for the eight depth-enhanced sequences. FS undergoes its fixed SR of 64 pixels, which means that 16641 search points are used for each block in ME. From Table 5.2, FS+DMRMap and FS+DMRMap+Scaling can averagely save 94.31% and 93.93% of coding time over FS, respectively. The SRs obtained by FS+DMRMap and FS+DMRMap+Scaling are always smaller than that of FS since they utilize the high temporal correlation of motions revealed by depth intensity mapping. A significant time reduction of around 99% by the proposed FS+DMRMap and FS+DMRMap+Scaling can be observed at "Lovebird1" sequence. This can be explained by the fact that "Lovebird1" consists of large portion of slow movement so that the proposed techniques can offer remarkable reduction in SR in the mapping process. As a result, both proposed algorithms, FS+DMRMap and FS+DMRMap+Scaling, only consume about 1% encoding time of FS. While significant coding time reduction can be achieved, the coding efficiency of the proposed FS+DMRMap and FS+DMRMap+Scaling can be maintained as compared to FS. From the results of Table 5.2, FS+DMRMap obtains negligible loss on BD-PSNR by 0.01dB as compared to FS while only 0.31% of BD-rate is raised. With the help of the proposed scale factor $\rho(B_t^n)$ on SR due to the 3D-to-2D projection, FS+DMRMap+Scaling also attains negligible loss on BD-PSNR by 0.01dB as compared to FS. At the same time, it only costs an increment of 0.26% in BD-rate.

Table 5.2 further lists out the results of FS+LSMF [118], FS+MLELD [117], and FS+LAMASR [119]. It can be observed that FS+LSMF, FS+MLELD, and FS+LAMASR reduce the computational complexity by averagely 41.68%, 56.73%,

Table 5.3: SR dimension and average number of search points per CU of FS, FS+LSMF, FS+MLELD, FS+LAMASR, and FS+DMRMap+Scaling

	Dimension (Dn, integer pixel)						Average number of search points per CU (Sp)					
Sequences	FS	FS+LSMF	FS+MLELD	FS+LAMASR	FS+DMRMap+Scaling		FS	FS+LSMF	FS+MLELD	FS+LAMASR	FS+DMRMap+Scaling	
720p	$Dn_x = Dn_y$	$Dn_x = Dn_y$			Dn_x	Dn_y	2D search window					
Balloons	64	47	38	32	23	14	16641	9025	5929	4225	1363	
Kendo	64	42	27	20	13	18	16641	7225	3025	1681	999	
Lovebird1	64	34	21	14	3	6	16641	4761	1849	841	91	
Newspaper	64	41	32	19	15	8	16641	6889	4225	1521	527	
1080p	$Dn_x = Dn_y$	$Dn_x = Dn_y$			Dn_x	Dn_y	2D search window					
Poznan_Street	64	48	45	35	14	10	16641	9409	8281	5041	609	
Poznan_Hall2	64	55	48	50	28	22	16641	12321	9409	10201	2565	
Undo_Dancer	64	54	50	52	11	25	16641	11881	10201	11025	1173	
GT_Fly	64	54	53	51	17	11	16641	11881	11449	10609	805	
Average:	64	47	39	34	16	14	16641	9174	6796	5643	1017	

and 66.36%, respectively while the proposed FS+DMRMap+Scaling reduces the complexity by 93.93%. The proposed FS+DMRMap+Scaling can save more computational time by about 52%, 37%, and 27%, respectively, as compared with the algorithms in the literature, FS+LSMF, FS+MLELD, and FS+LAMASR. Meanwhile, these algorithms obtain very similar BD-rate deterioration. The reason is that FS+DMRMap+Scaling considers the SR in the x- and y-directions separately for tracing the true MVs. Furthermore, FS+DMRMap+Scaling utilizes an adaptive scale factor for ASR adjustment. On the other hand, all the algorithms in [117–119] consider the SR in the x- and y-directions jointly. In addition, FS+LAMASR simply multiplies a fixed scale factor to the sum of the amplitude for ASR.

From the results of Table 5.2, it can be found that the gain in computational time for 1080p sequences is less significant compared to that of 720p sequences in FS+LSMF, FS+MLELD, and FS +LAMASR. This can be explained by the fact that FS+LSMF and FS +MLELD adopt the MV difference distribution of the previous frame to determine the SR, and FS+LAMASR uses the sum of amplitude differences among all motion vector predictors of the current block to initialize the ASR. In general, the motion vector differences among blocks are used as the hint to guide the SR determination for these three ASR algorithms. However, motion activities in 1080p test sequences are always richer than those in 720p test sequences [137]. In other words, MV differences between blocks are more likely to have abrupt change such that FS+LSMF, FS+MLELD, and FS+LAMASR will have a larger SR in 1080p sequences (i.e. less time reduction as a result). On the contrary, our proposed FS+DMRMap and FS+DMRMap+Scaling can obtain the consistent gain in computational time for both 1080p and 720p sequences, as shown in Table 5.2, since

they only use the depth map for SR determination, which is insensitive to the video resolution.

Table 5.3 further compares the average sizes of the SR in the x- and y-directions of FS, FS+LSMF, FS+MLELD, FS+LAMASR, and the proposed work, FS +DMRMap+Scaling. This table records the x- and y-dimensions of the SR (Dn_x and Dn_y, respectively) and the average number of search points (Sp) per block for the five tested algorithms. For FS, FS+LSMF, FS+MLELD, and FS+LAMASR, Dn_x is equal to Dn_y, and they are 64, 47, 39, and 34 on average, respectively, as shown in Table 5.3. It implies that all FS, FS+LSMF, FS+MLELD, and FS+LAMASR obtain a search window with aspect ratio of 1. For FS+DMRMap+Scaling, the SRs in the x- and y-directions are computed independently. Along the sequences, the aspect ratio of the search window is no longer equal to 1, and it depends on the motion characteristics of the sequence. The proposed FS+DMRMap+Scaling therefore can adopt the search window with various aspect ratios for well fitting the true motion. As a result, the average number of search points for each CU is computed as (5.15) and listed in Table 5.3.

$$Sp = (2Dn_x + 1) \times (2Dn_y + 1). \tag{5.15}$$

In (5.15), Sp is defined as the number of search points in a search window based on Dn_x and Dn_y. Finally, Table 5.3 shows that FS+LAMASR only requires around one third of search points per CU compared to FS whereas the proposed FS+DMRMap+Scaling only occupies averagely less than one tenth of search points for compared to FS.

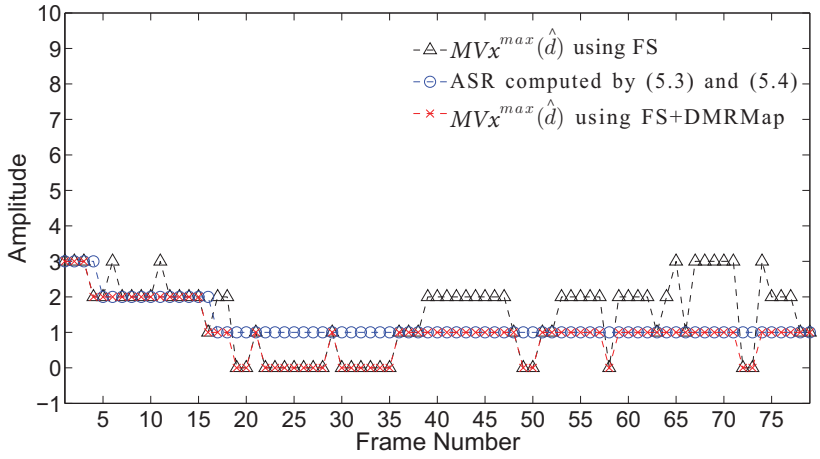


Figure 5.6: The maximum absolute amplitude of motion vectors, $MVx^{\max}(\hat{d})$ using FS and FS+DMRMap, and ASR with $\hat{d} = 7$ along frames for color texture of "Lovebird1".

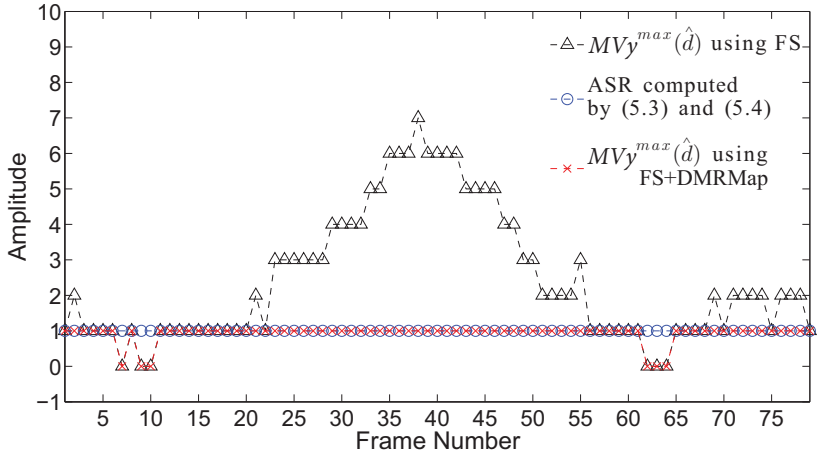


Figure 5.7: The maximum absolute amplitude of motion vectors, $MVy^{\max}(\hat{d})$ using FS and FS+DMRMap, and ASR with $\hat{d} = 14$ along frames for color texture of "Newspaper".

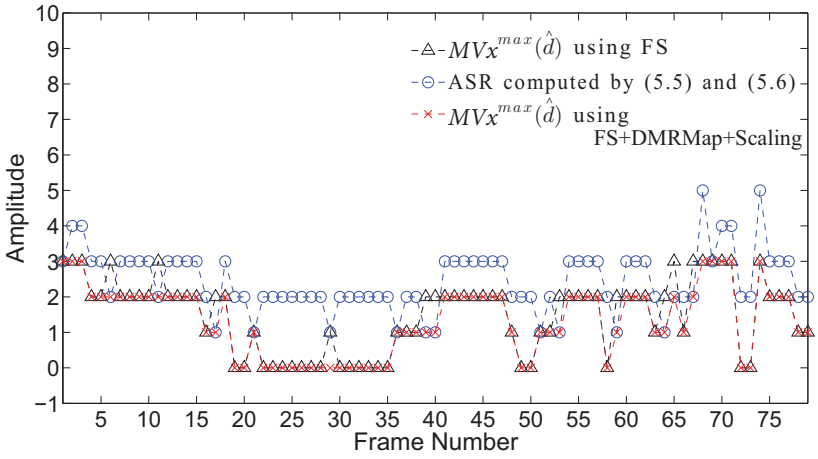


Figure 5.8: The maximum absolute amplitude of motion vectors, $MV_x{}^{\max}(\hat{d})$ using FS and FS+DMRMap+Scaling, and ASR with $\hat{d} = 7$ along frames for color texture of "Lovebird1".

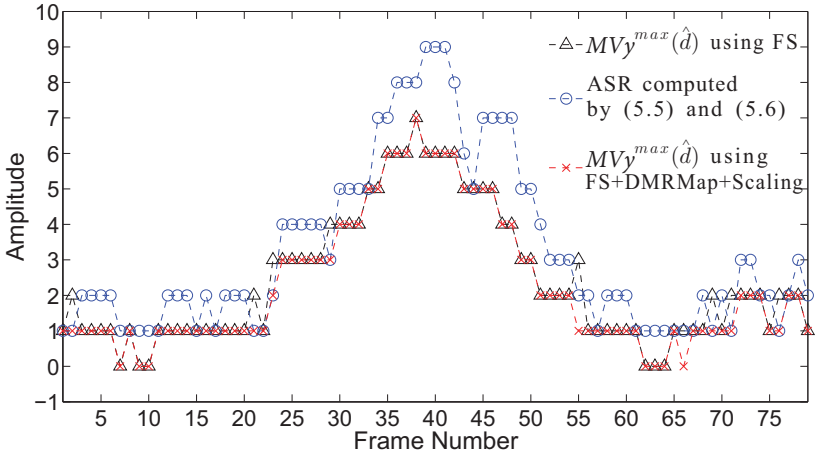


Figure 5.9: The maximum absolute amplitude of motion vectors, $MV_y{}^{\max}(\hat{d})$ using FS and FS+DMRMap+Scaling, and ASR with $\hat{d} = 14$ along frames for color texture of "Newspaper".

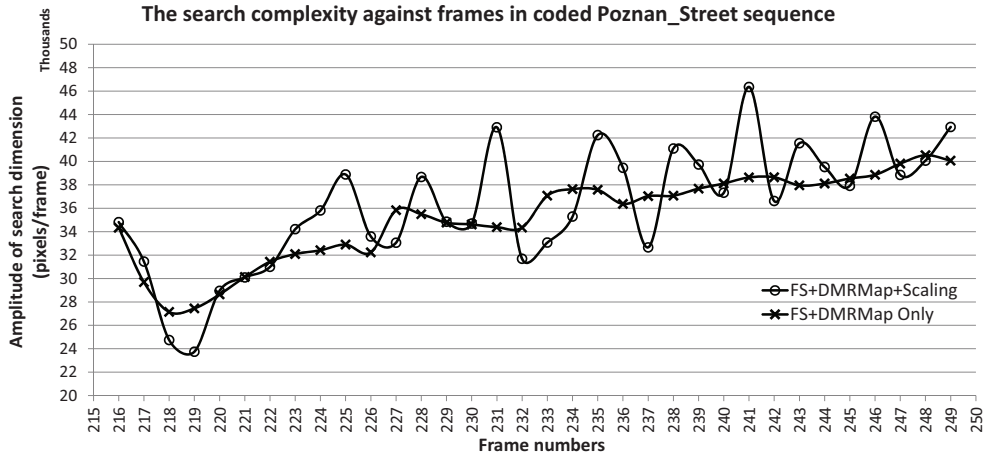
5.5.3 Gains of Scaling Technique on DMRMap

From the results in Table 5.2, we can see that FS+DMRMap+Scaling obtains a slight decrease in BD-rate compared with FS+DMRMap. The gain is contributed from the scale factor $\rho(B_t^n)$ in (5.13) that can adjust the final ASR based on z-axis motion. Figure 5.6 and Figure 5.7 exemplify the inefficiency in FS+DMRMap. In these figures, $MVx^{\max}(\hat{d})$ and $MVy^{\max}(\hat{d})$ represent the largest MVs in the x- and y-directions at the quantized depth value $\hat{d}$, respectively, obtained by FS and FS+DMRMap. Figure 5.6 displays $MVx^{\max}(\hat{d})$ at $\hat{d} = 7$ in "Lovebird1" from frame 1 to frame 80. In most of the time, $MVx^{\max}(\hat{d})$ of FS+DMRMap is the same as that of FS. However, there are some discrepancies in a number of frames. It can be observed that $MVx^{\max}(\hat{d})$ of FS+DMRMap cannot follow the increase in $MVx^{\max}(\hat{d})$ of FS. This is because ASR decision of FS+DMRMap makes use of depth/motion relationship in the reference frame. It implies that the ASR at particular $\hat{d}$ of the current block cannot be larger than $MVx^{\max}(\hat{d})$ of the reference frame.

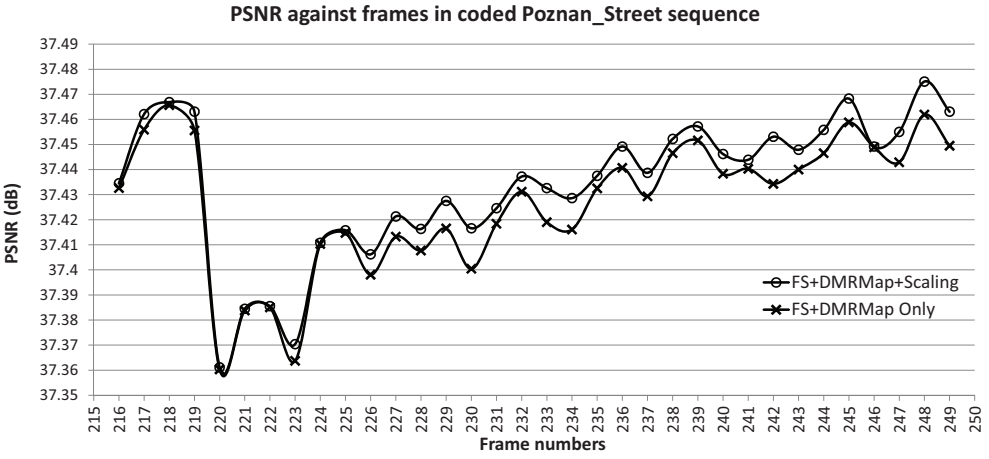
The ASR decision based on the DMRMap of the reference frame is also plotted in Figure 5.6 (the blue curve marked with circle dots). It is clearly shown that the resultant ASR is non increasing along frames due to the use of the DMRMap in the reference frame. This situation is more obvious in Figure 5.7 where $MVx^{\max}(\hat{d})$ at $\hat{d} = 14$ in "Newspaper" is shown. For instance, starting from frame 20 in "Newspaper", there is motion of an object along the z-axis, which results in reducing the temporal correlation of DMRMaps between the current and reference frames, as discussed in Section 5.3. As a consequence, FS+DMRMap may not catch the actual motions for the moving object, and may lead to RD deterioration. By contrast,

FS+DMRMap+Scaling utilizes $\rho(B_t^i)$ to provide additional flexibility in ASR decision for fitting the maximum texture motion. Figure 5.8 and Figure 5.9 illustrate how $\rho(B_t^n)$ can contribute the prediction accuracy of ASR. From these figures, it can be seen that FS+DMRMap+Scaling is able to catch up with $MVx^{\max}(\hat{d})$ and $MVy^{\max}(\hat{d})$ of FS along frames. It is due to the reason that the derivation of $\rho(B_t^n)$ from (5.13) complies with the influence of 3D-to-2D projection such that the ASR can be enlarged or diminished accordingly. In other words, $\rho(B_t^n)$ allows ASR to rebound to a larger value, as depicted in Figure 5.8 and Figure 5.9.

The advantage shown in Figure 5.8 and Figure 5.9 of FS+DMRMap+Scaling cannot be fully depicted in the results of Table 5.2 as the phenomenon in Figure 5.6 and Figure 5.7 only happens in a very short period of most sequences. However, both of the BD-rate and BD-PSNR in Table 5.2 measure the whole sequence in which the gain of FS+DMRMap+Scaling as compared with FS+DMRMap might be averaged out. To demonstrate this benefit of FS+DMRMap+Scaling, Figure 5.10 further shows the performance of FS+DMRMap+Scaling over FS+DMRMap in very short time period. Figure 5.10(a) shows the variation of the average search complexity depending on the SR size for all $\hat{d}$ with respect to the frame number from 216 to 249 in "Poznan_Street". During this period, "Poznan_Street" contains a car moving forward along the z-axis and a man walking away to the background, as shown in Figure 5.11. Figure 5.12 also shows the corresponding depth maps that exhibit remarkable changes in the moving object. Those changes in depth maps can be detected by FS+DMRMap+Scaling. In contrast to FS+DMRMap, the SR obtained by FS+DMRMap+Scaling is then enlarged or diminished accordingly, as shown in Figure 5.10(a). This mechanism allows FS+DMRMap+Scaling to successfully



(a)



(b)

Figure 5.10: Performance of FS+DMRMap+Scaling over FS+DMRMap (from frame 216 to frame 249) in "Poznan_Street". (a) Search complexity in term of amplitude of search dimensions. (b) Resultant PSNR.

(a) (b) (c)

Figure 5.11: Sample texture frames of "Poznan_Street". (a) Frame 216. (b) Frame 232. (c) Frame 248.

(a) (b) (c)

Figure 5.12: Sample depth frames of "Poznan_Street". (a) Frame 216. (b) Frame 232. (c) Frame 248.

provide a more adaptive SR for ME. Furthermore, PSNR results for coding "Poznan_Street" are plotted against the same series of frames in Figure 5.10(b). From the results, FS+DMRMap+Scaling achieves a better quality of coded frames over FS+DMRMap. The observed PSNR gains by FS+DMRMap+Scaling verify that the proposed scaling scheme can provide a proper adjustment of the SR. In conclusion, FS+DMRMap+Scaling can balance the SR prediction accuracy, the complexity of ME, and the PSNR performance with the help of the scale factor $\rho(B_t^n)$.

Table 5.4: Bjontegaard (BD) measurement and coding time change of TZS+LSMF, TZS+MLELD, TZS+LAMASR and TZS+DMRMap+Scaling for ASR against TZS in HEVC

Seq.	QP	TZS		TZS+LSMF			TZS+MLELD			TZS+LAMASR			TZS+DMRMap+Scaling		
		PSNR (dB)	Bitrate (kbps)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
720p															
Balloons	37	38.36	337.13	-12.07	-0.01	+0.11	-15.48	-0.01	+0.14	-14.24	0.00	+0.02	-48.92	0.00	-0.01
	32	41.24	594.51												
	27	43.56	1171.83												
	22	45.46	3131.69												
Kendo	37	39.64	372.91	-16.90	0.00	+0.03	-17.46	0.00	+0.12	-25.92	-0.01	+0.30	-50.14	0.00	+0.13
	32	42.27	655.30												
	27	44.44	1238.99												
	22	46.29	2901.65												
Lovebird1	37	34.32	164.99	-19.75	-0.01	+0.14	-29.30	-0.01	+0.26	-37.90	0.00	+0.02	-63.06	0.00	+0.10
	32	37.22	353.52												
	27	40.30	831.54												
	22	43.64	2073.03												
Newspaper	37	35.71	242.02	-21.80	-0.02	+0.34	-26.98	-0.02	+0.45	-22.34	0.00	+0.13	-54.50	-0.01	+0.24
	32	38.44	452.42												
	27	41.09	960.82												
	22	43.80	2710.81												
1080p															
Poznan_Street	37	35.09	462.86	-14.89	-0.01	+0.16	-22.98	-0.01	+0.30	-21.36	-0.01	+0.64	-56.63	-0.01	+0.23
	32	37.47	1144.32												
	27	39.94	4223.44												
	22	43.24	24864.72												
Poznan_Hall2	37	39.56	238.55	-28.10	0.00	+0.22	-39.54	0.00	+0.04	-31.37	0.00	0.00	-54.25	0.00	+0.14
	32	41.07	502.38												
	27	42.24	1562.84												
	22	44.24	13723.17												
Undo_Dancer	37	32.99	1532.64	-5.70	-0.01	+0.17	-12.82	-0.01	+0.15	-5.98	-0.02	+0.58	-55.56	-0.01	+0.19
	32	35.75	4327.77												
	27	39.01	11262.33												
	22	42.68	25270.98												
GT_Fly	37	35.78	1169.20	-26.96	0.00	+0.02	-20.29	0.00	+0.01	-30.14	0.00	+0.09	-42.61	0.00	+0.12
	32	38.41	3040.38												
	27	41.08	7481.00												
	22	43.84	16976.31												
Average:				**-18.27**	**-0.01**	**+0.15**	**-23.11**	**-0.01**	**+0.18**	**-23.66**	**-0.01**	**+0.22**	**-53.21**	**0.00**	**+0.14**

5.5.4 Results of Applying DMRMap to Fast TZS

Section 5.5.2 and Section 5.5.3 demonstrate that the proposed DMRMap-based ASR algorithm with the scaling factor is successful in FS for complexity reduction. It is

worth noting that our proposed algorithm for SR determination is not only applied to FS, it is also compatible with other fast search algorithms in HEVC. To validate this, our proposed ASR determination has been also used by the fast TZS in HEVC, named as TZS+DMRMap+Scaling. TZS only searches points on the vertexes of the blocks and the diamond patterns with various sizes inside the fixed SR. Instead of using fixed SR in TZS, an ASR is determined by TZS+DMRMap+Scaling. It is noted that the search strategy of TZS is only suited for a squared search window. However, the DMRMap-based ASR algorithm can handle the horizontal and vertical search ranges separately. For the sake of simplicity, based on (5.5) and (5.6), the SR of the squared search window is then computed by $\max(Rx^{\rho}(B_t^n), Ry^{\rho}(B_t^n))$.

Table 5.4 shows the BD measurement and the coding time change of the proposed TZS+DMRMap+Scaling compared to TZS. As far as TZS+DMRMap+Scaling concerned, 42.61% to 63.06% coding time can be saved. Meanwhile, the coding efficiency almost has no loss in terms of BD-PSNR and BD-rate (0.14% increment). The above result indicates that the proposed DMRMap-based ASR algorithm is well compatible with the fast search strategy in HEVC and provides up to around 53.21% time saving on average with only negligible loss in BD measurements. Besides, Table 5.4 also shows the results when LSMF in [118], MLELD in [117], and LAMASR in [119] are used in TZS for ASR determination as denoted by TZS+LSMF, TZS+MLELD, and TZS+LAMASR, respectively. On average, TZS+DMRMap+Scaling attains a better BD performance comprising 0.01 dB BD-PSNR gain and a range from 0.01% to 0.08% BD-rate decrement compared to others. As shown in Table 5.4, TZS+DMRMap+Scaling can reduce more coding time from 29% to 34%. It shows that an accurate ASR determination by

TZS+DMRMap+Scaling is very crucial in the fast search ME process. The above experimental results demonstrate the proposed ASR scheme based on the temporal correlation of depth/motion relationship maps and the 3D-to-2D projection can figure out a more precise range for ME. As a result, motion vectors are obtained quickly.

5.5.5 Influence of Q on DMRMap Accuracy

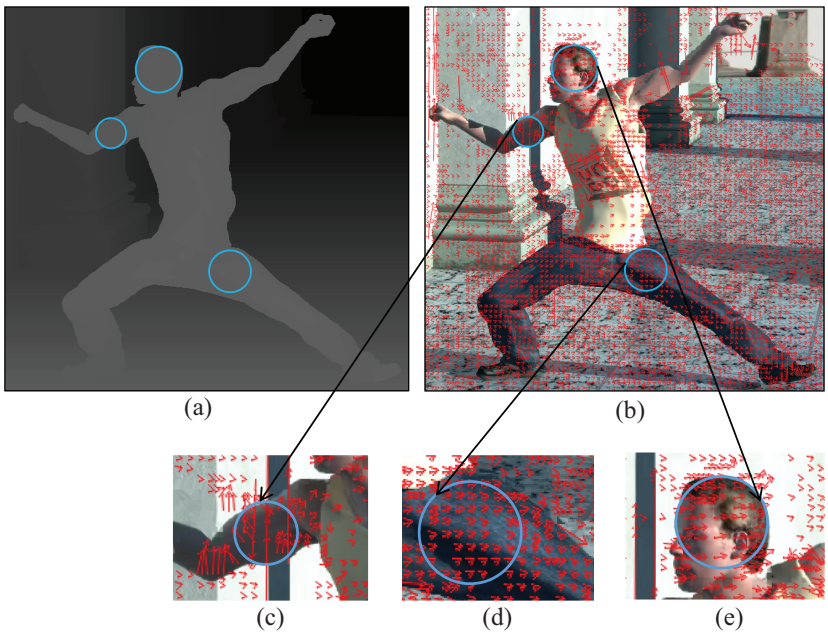


Figure 5.13: (a) Depth map, and (b) the corresponding texture of "Undo_Dancer". Magnified regions with similar depth intensity values in parts of (c) hand, (d) leg, and (e) head with different amplitudes of MVs.

In the following, we discuss the influence of the quantization factor Q on the performance of the proposed DMRMap-based algorithm. As mentioned in Section 5.2.2, Q is used to absorb depth variation in DMRMap construction due to the

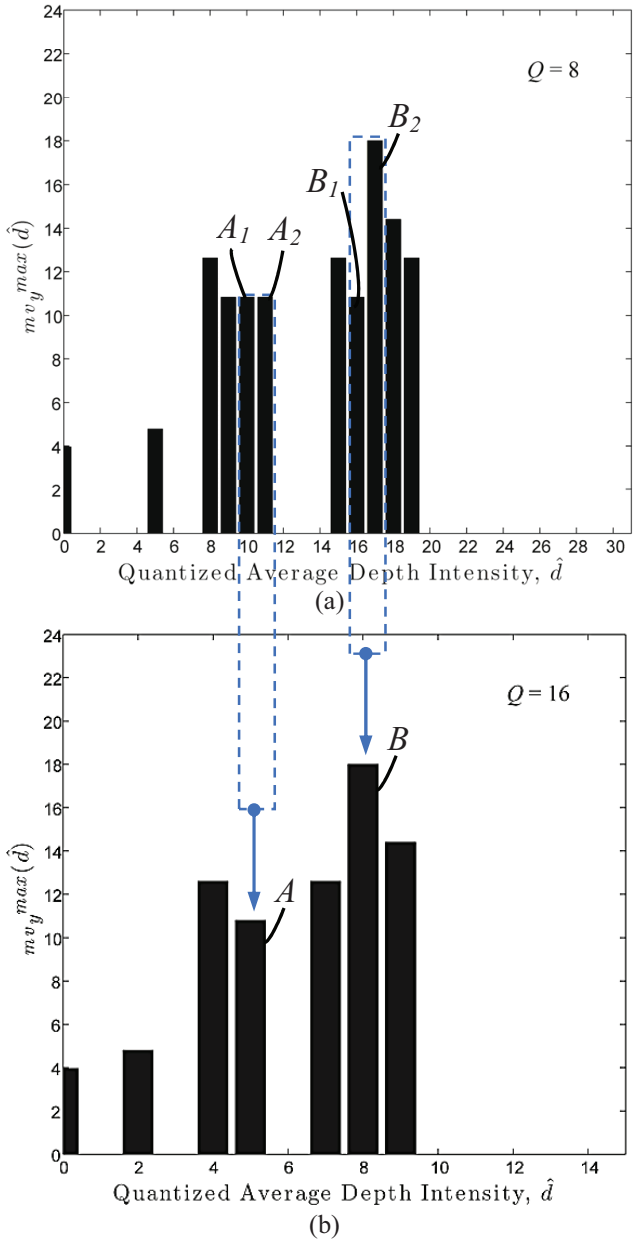


Figure 5.14: Illustration of DMRMaps with various Q, (a) $Q = 8$, and (b) $Q = 16$, in "Undo_Dancer".

Table 5.5: Bjontegaard (BD) measurement and coding time change of the proposed FS+DMRMap+Scaling with various Q against FS in HEVC

Sequences	FS+DMRMap+Scaling								
	$Q = 4$			$Q = 8$			$Q = 16$		
	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
720p									
Balloons	-93.43	-0.01	+0.34	-89.63	-0.01	+0.26	-88.57	-0.01	+0.14
Kendo	-95.09	-0.01	+0.34	-94.01	-0.01	+0.26	-92.71	-0.01	+0.16
Lovebird1	-99.51	0	0	-99.46	0	+0.08	-99.42	0	+0.01
Newspaper	-97.85	-0.01	+0.38	-97.23	-0.01	+0.31	-95.82	-0.01	+0.32
1080p									
Poznan_Street	-98.07	-0.01	+0.25	-97.15	0	+0.17	-96.08	0	+0.17
Poznan_Hall2	-87.42	-0.01	+0.44	-86.72	-0.01	+0.41	-85.32	0	+0.24
Undo_Dancer	-95.04	-0.02	+0.42	-91.74	-0.01	+0.30	-87.75	-0.01	+0.27
GT_Fly	-97.19	-0.01	+0.34	-95.47	-0.01	+0.27	-93.07	-0.01	+0.19
Average:	-95.45	-0.01	+0.31	-93.93	-0.01	+0.26	-92.34	-0.01	+0.19

noise of a depth map. Figure 5.14 illustrates two DMRMaps using different Q. For the example in Figure 5.14(b) where $Q = 16$, group A is exactly equivalent to group A_1 and group A_2 ($Q = 8$) in Figure 5.14(a) with the same largest MV. It implies that the DMRMaps using $Q = 8$ and $Q = 16$ will not affect the accuracy of the mapping processing. In contrast, group B_1 and group B_2 in Figure 5.14(a) of $Q = 8$ associate with different largest MVs while they are combined to group B in Figure 5.14(b) of $Q = 16$. It means that a large SR is required for large Q, but has a chance to achieve better BD-rate in this scenario. It is also the tradeoff between the computational complexity and BD performance of the proposed FS+DMRMap+Scaling. The evidence can be seen in Table 5.5 where the performances in terms of the BD measurement and the coding time change for various Q are shown. As expected, the complexity reduction increases as Q decreases for all sequences. Nevertheless, it only shows little variation for nearly all sequences, except "Balloons" and "Undo_Dancer".

It is interesting to note that the depth map of "Undo_Dancer", as shown in Figure 5.13(a), is different from most of other sequences. Its depth map is computer generated sequence using 3D models and its depth map is ground truth without noise. Besides, the dancer contains diverse motion activities in different parts of his body, as shown in Figure 5.13(b) to Figure 5.13(e). However, these different parts of his body have very close depth values. The quantization process in the construction of DMRMap might merge parts with different motions of the hand, leg, and head into one group if Q is large, which leads to the increase in the computational complexity of "Undo_Dancer", as shown in Table 5.5. It also happens in "Balloons" where the balloons in the foreground have similar depth values, but diverse mo-

Table 5.6: BD performances and Δtime obtained by work in Chapters 4 and 5

	NDIWS in Chapter 4			DMRMap+Scaling in Chapter 5		
Average performance	Δtime (%)	BD-PSNR (dB)	BD-rate (%)	Δtime (%)	BD-PSNR (dB)	BD-rate (%)
Applying on FS	-95.03	-0.02	+0.64	-93.93	-0.01	+0.26
Applying on TZS	-65.78	-0.02	+0.53	-53.21	0	+0.14

tions. In conclusion, for sequences having noiseless depth maps and complicated motion with similar depth value, it is beneficent to adopt small Q for the DMRMap construction.

5.5.6 Evaluation on Depth-based ASR with Spatial Correlation in Chapter 4 and Temporal Correlation in this chapter

In Table 5.6, it shows the encoding performance with the proposed ASR applying on both FS and TZS by NDIWS algorithm in Chapter 4 and DMRMap+Scaling algorithm in this chapter, respectively. The proposed NDIWS algorithm utilizes depth map difference to formulate decaying weights to the neighbouring blocks and the ASR is their weighted sum on the MV amplitudes. It facilitates the search point reduction on FS and TZS such that about 12% of encoding time can be saved as compared to DMRMap+Scaling algorithm in this chapter. It is because spatial correlation is specially suitable for obtaining a huge search points reduction in local homogeneous area. However, it obtains a trade-off of larger RD deterioration since

it could not follow the motion's trail along time. Instead, from Table 5.6, DMRMap+Scaling algorithm in this chapter obtains better coding efficiency comparatively by around 0.4% of BD-rate decrement. It utilizes the temporal correlation linkage which can attain more accurate motion locus along time since it always selects the maximum probable motion ranges by the mapped MV amplitudes within the same object by the depth intensity.

5.6 Chapter Summary

In this chapter, we have proposed an efficient ASR algorithm for HEVC to reduce the computational complexity of ME by exploiting the temporal correlation between the depth map and motion in texture. The new depth/motion relationship map (DMRMap) is then established, and is interpreted to control the ASR for each block. DMRMap builds the linkage on the same object among consecutive frames which reflects the probable range of movements for the object. Based on this, a depth intensity mapping is contrived to form an asymmetric SR for ME. It results in reducing unnecessary search points in ME. Furthermore, the impact of the depth intensity variations of the block in 3D-to-2D projection on ASR has been analyzed. By taking this into account, a scale factor has been proposed to comply with the impact of 3D-to-2D projection. The proposed DMRMap could be jointly worked with FS and other fast search algorithms such as TZS in HEVC for complexity reduction. Simulation results demonstrated that the proposed DMRMap-based ASR algorithm is able to reduce up to 53% of average coding time among various sequences in fast ME algorithms. In the meantime, the coding efficiency can be maintained compared

to FS and TZS in terms of the BD measurement.

We further evaluate the encoding performance of the proposed ASR applying on both FS and TZS by NDIWS algorithm in Chapter 4 and DMRMap+Scaling algorithm in this chapter, respectively. It is found that NDIWS algorithm in Chapter 4 can save more encoding time comparatively up to about 12% but it induces around 0.4% of BD-rate increment, compared to DMRMap+Scaling algorithm in this chapter.

Chapter 6

Conclusions and Future Work

In this thesis, we have investigated the motion locus prediction before the computational intensive motion estimation (ME) process in video coding. Intensive ME search is not only required in remote reference frames in the hierarchical P (HP) structure, but also adopted in the state-of-the-art HEVC recursive block partitioning structure. By the motion locus prediction, unnecessary motion search can be skipped such that coding complexity can be reduced. In each chapter, motivation in various applications were revealed followed by analyses in details. The proposed algorithms with the corresponding rationales were introduced with illustrations. Afterwards, simulation results were provided with analyses and discussions. In this final chapter, the main contributions of this thesis are summarized. After that, we discuss some possible directions that could be the focus for future research.

6.1 Contributions of the Thesis

Our contributions mainly include a comprehensive study with a series of literature review in Chapter 2 and constructive proposals of (1) a new motion vector composition (MV composition) algorithm and its vector selection algorithm in the support of the low-delay HP structure; (2) adaptive search range (ASR) determination by depth-weighted sum of neighboring MVs; and (3) depth-based ASR adjustment by depth/motion relationship maps. All the work are in the objective of computational complexity reduction in HEVC.

In particular, our conclusions are:

- The adoption of a MV composition technique in the HP video coding structure was explored in Chapter 3. This HP structure is used in many emerging video applications like delay sensitive video conferencing. In the HP structure, the prediction distance of frames in the low temporal layer is very large. To prevent from using a large search window in ME at the low temporal layer, we are the first to consider the MV composition algorithms in HP coding. Furthermore, the proposed Adaptive Multiple-Candidate Vector Selection (AMCVS) in MV composition results in good capability to carry out ME to a remote reference frame. Simulation results have proven that the AMCVS succeeds in achieving better coding efficiency when ME is conducted on a temporal remote reference frame in MV composition.

- Our results in Chapter 4 demonstrated that the proposed ASR algorithm is able to reduce averagely 65% of coding time with negligible loss on BD performance as compared to the fast test zone search (TZS) algorithm. The pro-

posed ASR is determined by considering the absolute difference in average depth intensity with neighboring blocks and formulating the weighted sum of their MVs. It results in a complexity reduction in ME by this ASR. The proposed ASR is compatible with full-search (FS) and other fast search algorithms such as TZS in HEVC.

- A further usage of depth information in multi-view plus depth (MVD) video was proposed in Chapter 5 to construct the depth and motion relationship map (DMRMap) between frames. By making use of the novel DMRMap, the temporal correlation between the depth map and motion in texture is exploited. It constructs the linkage on the same object among consecutive frames. This can give the object an expected search range. Afterwards, an asymmetric ASR for ME is established for skipping unnecessary search. Furthermore, a scale factor has been designed to alleviate the impact of the depth intensity variations of the block in 3D-to-2D projection on ASR. The proposed technique is well suited for both FS and other fast search algorithms such as TZS in HEVC for complexity reduction. Simulation results reveal that, compared to other fast approaches, the proposed algorithm can reduce the complexity up to 53% on average whereas the coding efficiency can be maintained.

- Having research and evaluation work in Chapter 4 focusing the spatial correlation and Chapter 5 focusing the temporal correlation, it is further revealed that using depth information to weight the spatial motion ranges could saving more encoding time in Chapter 4. However, it obtains a trade-off of larger RD deterioration. The work utilizing the temporal correlation linkage in Chapter 5

can attain more accurate motion locus along time since it always selects the maximum probable motion ranges by the mapped MV amplitudes within the same object by the depth intensity.

- In our present work, several techniques have been investigated that can reduce the computational complexity of the new tools introduced in HEVC. We believe that the results achieved in this work contribute remarkably to the efficient realization of the modern HEVC coding system.

6.2 Future Work

With the successful techniques proposed and implemented in this thesis and well proved by a wide range of simulation works, we now give some opinions on the trend for the future development of our related studies.

- The hierarchical B (HB) for random access, which includes the use of future frames as a reference, is also well established in the HEVC and 3D video coding standards. The motion vector composition technique in HP has been investigated here. It is worth investigating the video coding scheme that the HB structure is considered. While the MV composition problem becomes much more complicated, a new vector selection algorithm should be pursued in the future.

- In Chapter 5, we proposed a depth-based ASR adjustment algorithm by DMRMap construction. The proposed DMRMap-based algorithm is now independent from quantization parameters (QPs), it only involves a fixed step size

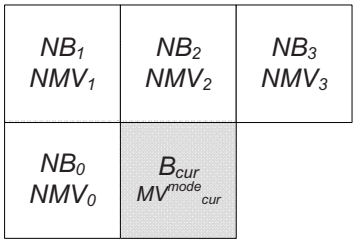


Figure 6.1: AMVP selection from MV among neighbouring blocks.

quantization, Q of average block depth intensity in DMRMap construction for better classification. As mentioned in Chapter 5, Q in DMRMap construction depends on the noise level of depth maps, the amount of complicated motion activities in a same object, the diversity of motion activities in different objects with similar depth values, etc. Since the QPs also play an important role on HEVC mode decision for accuracy, some efficient ways to determine the QP together with the Q in DMRMap construction will be studied. A challenging research topic is to generalize our DMRMap under different kinds of depth maps by a sequence-dependent Q. This could be a point for our immediate future work.

- Advanced motion vector prediction (AMVP) is another key inter prediction coding tool adopted in the emerging HEVC standard, which provides great coding efficiency. AMVP in HEVC is the extension work of finding the predictor in SKIP mode of H.264. In HEVC, MV for SKIP mode of a current block is equal to the advanced motion vector predictor (AMVP) [138], i.e. $MV_{cur}^{skip} = AMVP$. The AMVP is selected from one of the candidates

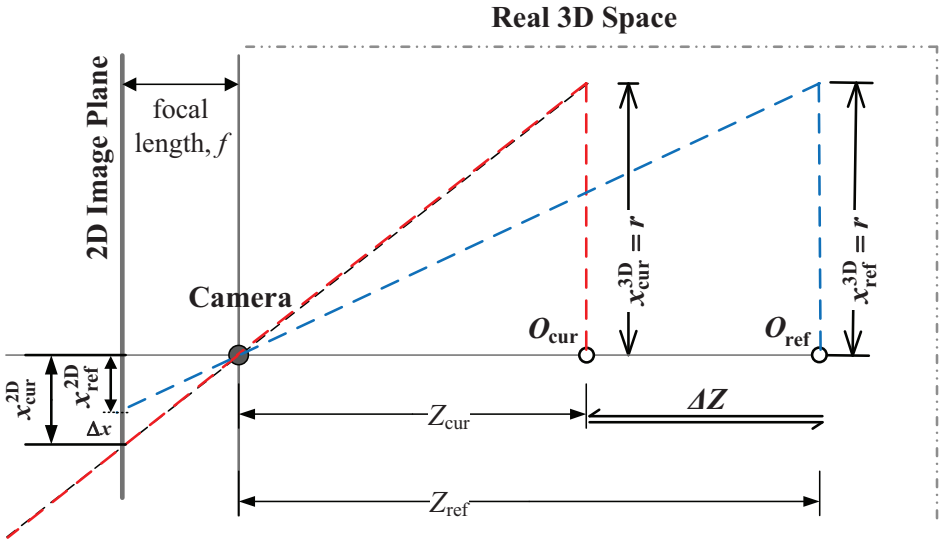


Figure 6.2: Geometric relationship of the projected dimension change by depth intensity difference between two objects.

who obtains the least RD cost among a set of neighboring blocks [139]. The AMVP candidates from neighboring blocks are NMV_0, NMV_1, NMV_2, and NMV_3 as shown in Figure 6.1. In SKIP mode, the current block uses the identical MV amplitude by AMVP directly. However, if the current block and the AMVP block are in different distance in a 3D space, the projected blocks of them will be in distinct dimensions as illustrated in Figure 6.2.

From Figure 6.2, assume there are two objects, O_{cur} and O_{ref} with their corresponding distance from the camera in the real 3D space, Z_{cur} and Z_{ref}, respectively. When they have the same dimension in the real 3D space denoted as $x_{\text{cur}}^{\text{3D}} = x_{\text{ref}}^{\text{3D}} = r$, the projected dimension lengths are formed on the 2D image plane as depicted in Figure 6.2. Their projected dimension lengths of O_{cur} and O_{ref} are represented by $x_{\text{cur}}^{\text{2D}}$ and $x_{\text{ref}}^{\text{2D}}$ respectively. By this geometric

relationship, even the same dimension of objects are applied in the 3D space, two objects with different distance from camera (i.e. $Z_{\text{cur}} \neq Z_{\text{ref}}$) will form different projected dimension lengths (i.e. $x_{\text{cur}}^{\text{2D}} \neq x_{\text{ref}}^{\text{2D}}$) in the 2D image plane. If the neighboring blocks have different depth intensities, AMVP should not be directly used for the predictor of the current block. Thus, extending our depth-based HEVC framework to AMVP will be of great interest to the research community and industry.

- Previous work in merge mode [140] has shown that redundant sets of coding information like motion vectors can be successfully reduced by merging the leaf nodes of the particular quad-tree structure. Following the same concept of using depth information, the study of merge mode together with depth maps should provide an enhanced performance of HEVC.

Bibliography

[1] G. Gualdi, A. Prati, and R. Cucchiara, "Video streaming for mobile video surveillance," *IEEE Trans. Multimedia*, vol. 10, no. 6, pp. 1142–1154, Oct 2008.

[2] S. Bachu and K. M. Chari, "A review on motion estimation in video compression," in *2015 Int. Conf. Signal Process. Commun. Engineer. Syst. (SPACES)*, Jan 2015, pp. 250–256.

[3] S. Hong and S. Kim, "Joint video coding of MPEG-2 video programs for digital broadcasting services," *IEEE Trans. Broadcast.*, vol. 44, no. 2, pp. 153–164, June 1998.

[4] I. E. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia*, 1st ed. Chichester, UK: John Wiley & Sons Ltd., 2003.

[5] ISO/IEC, *Information technology - Coding of audio-visual objects - Part 10: Advanced Video Coding (2nd Edition)*, ISO/IEC Std. 14 496-10, 2004.

[6] ITU-T and ISO/IEC, *Advanced video coding for generic audiovisual services*, ITU-T Rec. H.264 and ISO/IEC 14496-10 Std., March 2010. [Online]. Available: http://www.itu.int/rec/T-REC-H.264

[7] I. E. Richardson, *The H.264 advanced video compression standard*, 2nd ed. Chichester, UK: John Wiley & Sons Ltd., 2010.

[8] ITU-T and ISO/IEC, *SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS - Infrastructure of audiovisual services Coding of moving video, High efficiency video coding*, ITU-T H.265 and ISO/IEC 23008-2 Std., April 2013. [Online]. Available: http://www.itu.int/rec/T-REC-H.265-201304-S

[9] F. Kossentini, N. Mahdi, H. Guermazi, M. Horowitz, S. Xu, B. Li, G. J. Sullivan, and J. Xu, *Informal Subjective Quality Comparison of Compression Performance of HEVC Working Draft 5 with AVC High Profile, document JCTVC-H0562*, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC) Std., February 2012.

[10] R. Schafer and T. Sikora, "Digital video coding standards and their role in video communications," *IEEE Proc.*, vol. 83, no. 6, pp. 907–924, June 1995.

[11] T. Sikora, "MPEG digital video-coding standards," *IEEE Signal Process. Mag.*, vol. 14, no. 5, pp. 82–100, September 1997.

[12] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.

[13] D. Marpe, T. Wiegand, and G. J. Sullivan, "The H.264/MPEG-4 advanced video coding standard and its applications," *IEEE Commun. Mag.*, vol. 44, no. 8, pp. 134–143, August 2006.

[14] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, December 2012.

[15] J. R. Ohm and G. J. Sullivan, "High efficiency video coding: the next frontier in video compression [standards in a nutshell]," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 152–158, Jan 2013.

[16] B. Bing, *Next-Generation Video Coding and Streaming*. Canada: John Wiley & Sons, Inc., 2015.

[17] H. Schwarz, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, D. Marpe, P. Merkle, K. Mller, H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D video coding using advanced prediction, depth modeling, and encoder control methods," in *Proc. IEEE Picture Coding Symp. (PCS2012)*, May 2012, pp. 1–4.

[18] G. Sullivan and T. Wiegand, "Video compression from concepts to the H.264/MPEG-4 AVC video coding standard," *IEEE Proc.*, vol. 93, no. 1, pp. 18–31, January 2005.

[19] H. Schwarz, D. Marpe, and T. Wiegand, *Hierarchical B pictures*, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Std. Std. JVT-P014, 2005.

[20] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE Int. Conf. on Multimedia and Expo*, Canada, July 2006, pp. 1929–1932.

[21] A. Leontaris and P. C. Cosman, "Compression efficiency and delay tradeoffs for hierarchical B-pictures and pulsed-quality frames," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1726–1740, July 2007.

[22] H. A. Q. Maarif, T. S. Gunawan, and A. U. Priantoro, "Complexity evaluation in scalable video coding," *J. Adv. Multimedia*, vol. 1, no. 1, pp. 12–25, May 2010.

[23] I. Kim, J. Min, T. Lee, W. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1697–1706, December 2012.

[24] ITU-T, *Video codec for audiovisual services at px64 kbit/s*, ITU-T Rec. Std., 1993.

[25] ITU-T, *Video coding for low bit rate communication*, ITU-T Rec. Std., 2005.

[26] ITU-T and ISO/IEC, *Information technology - Generic coding of moving pictures and associated audio information: Video*, ITU-T Rec. and ISO/IEC 13818-2 Std., February 2000.

[27] ISO/IEC, *Information technology - Coding of audio-visual objects - Part 2: Visual.*, ISO/IEC 14496-2 Std., April 2010.

[28] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sept 2007.

[29] B. Lee and M. Kim, "An efficient inter-prediction mode decision method for temporal scalability coding with hierarchical B-picture structure," *IEEE Trans. Broadcast.*, vol. 58, no. 2, pp. 285–290, June 2012.

[30] D. Hong, M. Horowitz, A. Eleftheriadis, and T. Wiegand, "H.264 hierarchical P coding in the context of ultra-low delay, low complexity applications," in *Proc. IEEE Picture Coding Symp. (PCS2010)*, Japan, December 2010, pp. 146–149.

[31] ITU-T and ISO/IEC, *Common test conditions and software reference configurations*, Joint Collaborative Team on Video Coding document JCTVC-K1100 of JCT-VC Std., January 2013. [Online]. Available: http://www.itu.int/rec/T-REC-H.264

[32] P. Merkle, A. Smolic, K. Mller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, p. 14611472, November 2007.

[33] S. A. Fezza, K. M. Faraoun, and S. Ouddane, "A comparison of prediction structures for multi-view video coding based on the H.264/AVC standard," in *Proc. Int. Workshop on Syst., Signal Process. Appl. (WOSSPA)*, Algeria, May 2011, pp. 111–114.

[34] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1685–1696, Dec 2012.

[35] P. Hanhart, M. Rerabek, F. D. Simone, and T. Ebrahimi, "Subjective quality evaluation of the upcoming HEVC video compression standard," in *Proc. SPIE Optics Photon.*, Aug. 2012, pp. 1871–1884.

[36] M. Horowitz, F. Kossentini, N. Mahdi, S. Xu, H. Guermazi, H. Tmar, B. Li, G. J. Sullivan, and J. Xu, "Informal subjective quality comparison of video compression performance of the HEVC and H.264/MPEG-4 AVC standards for low-delay applications," *SPIE Proc.*, vol. 8499, pp. 84 990W–84 990W–6, 2012. [Online]. Available: http://dx.doi.org/10.1117/12.953235

[37] J. Vanne, M. Viitanen, T. Hmlinen, and A. Hallapuro, "Comparative rate-distortion-complexity analysis of HEVC and AVC video codecs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1885–1898, Dec 2012.

[38] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," *IEEE Proc.*, vol. 99, no. 4, pp. 626–642, April 2011.

[39] K. Muller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *IEEE Proc.*, vol. 99, no. 4, pp. 643–656, April 2011.

[40] F. Shao, G. Jiang, M. Yu, K. Chen, and Y. S. Ho, "Asymmetric coding of multi-view video plus depth based 3-D video for view rendering," *IEEE Trans. Broadcast.*, vol. 14, no. 1, pp. 157–167, February 2012.

[41] S. Khattak, R. Hamzaoui, S. Ahmad, and P. Frossard, "Low-complexity multiview video coding," in *Proc. IEEE Picture Coding Symp. (PCS2012)*, Poland, May 2012, p. 97100.

[42] L. Shen, Z. Liu, P. An, R. Ma, and Z. Zhang, "Low-complexity mode decision for MVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 6, p. 837843, June 2011.

[43] H. Schwarz and K. Wegner, "Test model under consideration for HEVC based 3D video coding," ISO/IEC JTC1/SC29/WG11, San Jose, CA, USA, Feb 2012.

[44] A. Vetro, Y. Chen, and K. Mueller, "HEVC-compatible extensions for advanced coding of 3D and multiview video," in *Proc. 2015 Data Compression Conference (DCC)*, Apr 2015, pp. 13–22.

[45] W. Wan, Y. Chen, Y. K. Wang, M. M. Hannuksela, and H. Li, "Efficient hierarchical inter picture coding for H.264/AVC baseline profile," in *Proc. IEEE Picture Coding Symp. (PCS2009)*, USA, May 2009, pp. 1–4.

[46] G. J. Sullivan and J. R. Ohm, "Recent developments in standardization of high efficiency video coding (HEVC)," *SPIE Proc.*, pp. 1–7, August 2010.

[47] J. Kim, K. Yoo, and K. Lee, "Analysis and complexity reduction of high efficiency video coding for low-delay communication," in *Proc. IEEE Second Int. Conf. on Consumer Electronics, (ICCE-Berlin)*, Germany, September 2012, pp. 11–12.

[48] A. J. D. Honrubia, J. L. Martnez, and P. Cuenca, "HEVC: A review, trends and challenges," in *Proc. Workshop on Multimedia Data Coding and Transmission, (EMDCT)*, Spain, September 2012, pp. 1–6.

[49] K. Ugur, K. Andersson, A. Fuldseth, G. Bjontegaard, L. Endresen, J. Lainema, A. Hallapuro, J. Ridge, D. Rusanovskyy, C. Zhang, A. Norkin, C. Priddle, T. Rusert, J. Samuelsson, R. Sjoberg, and Z. Wu, "High performance, low complexity video coding and the emerging HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1688–1697, Dec 2010.

[50] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, *Scalable video coding*, Joint Draft ITU-T Rec. H.264 ISO/IEC 14496-10 Std., 2007. [Online]. Available: http://www.itu.int/rec/T-REC-H.264

[51] W. J. Han, J. Min, I. K. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, M. S. Cheon, N. Shlyakhov, K. McCann, T. Davies, and J. H. Park, "Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1709–1720, Dec 2010.

[52] G. Correa, P. Assuncao, L. Agostini, and L. da Silva Cruz, "Performance and computational complexity assessment of high-efficiency video encoders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1899–1909, Dec 2012.

[53] 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. Consum. Electron.*, vol. 58, no. 4, pp. 1375–1383, November 2012.

[54] B. Bross, P. Helle, S. Oudin, T. Nguyen, D. Marpe, H. Schwarz, and T. Wiegand, "Quadtree structures and improved techniques for motion representation and entropy coding in HEVC," in *Proc. IEEE Second Int. Conf. on Consumer Electronics, (ICCE-Berlin)*, Sept 2012, pp. 26–30.

[55] M. Karczewicz, P. Chen, R. Joshi, X. Wang, W.-J. Chien, R. Panchal, Y. Reznik, M. Coban, and I. S. Chong, "A hybrid video coder based on extended macroblock sizes, improved interpolation, and flexible motion representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1698–1708, Dec 2010.

[56] J. Lin, Y. Chen, Y. Huang, and S. Lei, "Motion vector coding in the HEVC standard," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 957–968, Dec 2013.

[57] D. Marpe, H. Schwarz, S. Bosse, B. Bross, P. Helle, T. Hinz, H. Kirchhoffer, H. Lakshman, T. Nguyen, S. Oudin, M. Siekmann, K. Sühring, M. Winken,

and T. Wiegand, "Video compression using nested quadtree structures, leaf merging, and improved techniques for motion representation and entropy coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1676–1687, Dec 2010.

[58] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *SPIE Stereoscopic Displays Virtual Reality Syst. X*, vol. 5291, no. 1, pp. 93–104, 2004. [Online]. Available: http://dx.doi.org/10.1117/12.953235

[59] T. Zhao, Z. Wang, and C. W. Chen, "Adaptive quantization parameter cascading in HEVC hierarchical coding," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 2997–3009, July 2016.

[60] G. Correa, P. Assuncao, L. Agostini, and L. A. da Silva Cruz, "Complexity control of HEVC through quadtree depth estimation," in *2013 IEEE EUROCON*, July 2013, pp. 81–86.

[61] X. Deng, M. Xu, L. Jiang, X. Sun, and Z. Wang, "Subjective-driven complexity control approach for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 91–106, Jan 2016.

[62] T. Sikora, "Mpeg digital video-coding standards," *IEEE Signal Processing Magazine*, vol. 14, no. 5, pp. 82–100, Sep 1997.

[63] *Information Technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s*, ISO/IEC Std. 11 172, 1993.

[64] *ITU-T Recommendation H.261: Video codec for audiovisual services at p x 384 kbit/s*, ITU-T Std., 1993.

[65] *ITU-T Recommendation H.263: Video coding for low bit rate communication (3rd Edition)*, ITU-T Std., 2005.

[66] A. M. Bock, Ed., *Video Compression Systems. From first principles to concatenated codecs*. London, UK: The Institution of Engineering and Technology, 2009.

[67] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transfom," *IEEE Trans. Comput.*, vol. C-23, no. 1, p. 90, January 1974.

[68] F. Arguello and E. Zapata, "Fast cosine transform based on the successive doubling method," *Electron. Lett.*, vol. 26, no. 19, p. 1616, September 1990.

[69] K. Shanmugam, "Comments on discrete cosine transform," *IEEE Trans. Comput.*, vol. C-24, no. 7, p. 759, July 1975.

[70] T. Eude, R. Grisel, H. Cherifi, and R. Debrie, "On the distribution of the DCT coefficients," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP 1994)*, vol. 5, April 1994, pp. 365–368.

[71] A. Johnson, J. Princen, and M. H. Chan, "Frequency scalable video coding using the MDCT," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP 1994)*, vol. 5, April 1994, pp. 477–480.

[72] V. Sze, M. Budagavi, and G. J. Sullivan, Eds., *High Efficiency Video Coding (HEVC): Algorithms and Architectures*. Switzerland: Springer, 2014.

[73] J. Zhu, Z. Liu, and D. Wang, "Fully pipelined DCT/IDCT/Hadamard unified transform architecture for HEVC codec," in *Proc. IEEE Int. Symp. on Circuits and Syst. (ISCAS 2013)*, May 2013, pp. 677–680.

[74] P. K. Meher, S. Y. Park, B. K. Mohanty, K. S. Lim, and C. Yeo, "Efficient integer DCT architectures for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 1, pp. 168–178, Jan 2014.

[75] P. Pirsch, "Design of DPCM quantizers for video signals using subjective tests," *IEEE Trans. Commun.*, vol. 29, no. 7, pp. 990 – 1000, July 1981.

[76] H. Y. Lin, Y. C. Chao, C. H. Chen, B. D. Liu, and J. F. Yang, "Combined 2-D transform and quantization architectures for H.264 video coders," in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS 2005)*, vol. 2, May 2005, pp. 1802–1805.

[77] T. Wedi and S. Wittmann, "Quantization offsets for video coding," in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS 2005)*, vol. 1, May 2005, pp. 324 – 327.

[78] B. Lee, J. Jung, and M. Kim, "An all-zero block detection scheme for low-complexity HEVC encoders," *IEEE Trans. Multimedia*, vol. 18, no. 7, pp. 1257–1268, July 2016.

[79] D. Huffman, "A method for the construction of minimum-redundancy codes," *Institute of Radio Engineers*, vol. 40, no. 9, pp. 1098 –1101, September 1952.

[80] G. Langdon and J. Rissanen, "Compression of black-white images with arithmetic coding," *IEEE Trans. Commun.*, vol. 29, no. 6, pp. 858 – 867, June 1981.

[81] D. Tian, W. Chen, P. S. Chang, G. Al Regib, and R. Mersereau, "Hybrid variable length coding for image and video compression," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP 2007)*, vol. 1, April 2007, pp. 1133–1136.

[82] V. Sze and M. Budagavi, *CE11- parallelization of HHI TRANSFORM CODING fixed diagonal scan*, Joint Collaborative Team on Video Coding document JCTVC-F129 Std., July 2011.

[83] Y. Zheng, M. Coban, X. Wang, J. Sole, R. Joshi, and M. Karczewicz, *CE11-mode dependent coefficient scanning*, Joint Collaborative Team on Video Coding document JCTVC-D393 Std., July 2011.

[84] C. Fogg, D. J. LeGall, J. L. Mitchell, and W. B. Pennebaker, *MPEG video compression standard*. New York: Springer, 2002.

[85] K. R. Namuduri, "Motion estimation using spatio-temporal contextual information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 8, pp. 1111–1115, Aug 2004.

[86] G. Laroche, J. Jung, and B. Pesquet-Popescu, "RD optimized coding for motion vector predictor selection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 12, pp. 1681–1691, Dec 2008.

[87] V. Sze and M. Budagavi, "High throughput CABAC entropy coding in HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1778–1791, Dec 2012.

[88] G. Tian, R. Hu, Q. Liu, and Z. Wang, "Improved temporal scalable video coding based on low-delay hierarchical dual reference p-picture prediction structure," in *Proc. Int. Conf. on Information Technol. Computer Science. (ITCS 2009)*, vol. 1, July 2009, pp. 433–436.

[89] R. Kumar and V. Patil, "An efficient motion vector composition scheme for arbitrary frame down-sampling video transcoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 9, pp. 1148–1152, Sept 2006.

[90] F. Lonetti and F. Martelli, "Motion vector composition algorithm in h.264 transcoding," in *14th Int. Workshop Syst., Signal. Image Process. 2007 and 6th EURASIP Conf. Speech Image Process., Multimedia Commun. Services.*, June 2007, pp. 401–404.

[91] Y. P. Tan and Y. Liang, "A unified transcoding approach to fast forward and reverse playback of compressed video," *IEEE Trans. Consum. Electron.*, vol. 49, no. 4, pp. 1098–1105, November 2003.

[92] J. Youn and M. T. Sun, "A fast motion vector composition method for temporal transcoding," in *Proc. IEEE Int. Symp. on Circuits and Syst. (ISCAS 1999)*, Orlando, USA, June 1999, pp. 243–246.

[93] J. Youn, M. T. Sun, and C. W. Lin, "Motion vector refinement for high-performance transcoding," *IEEE MultiMedia*, vol. 1, no. 1, pp. 30–40, March 1999.

[94] S. Yang, D. Kim, Y. Jeon, and J. Jeong, "An efficient motion re-estimation algorithm for frame-skipping video transcoding," in *Proc. IEEE Int. Conf. on Image Process. (ICIP 2005)*, September 2005, pp. 668–671.

[95] M. S. Goldman, L. Litwic, and O. Baumann, "ULTRA-HD content acquisition and exchange using HEVC range extensions," *J. SMPTE Motion Imaging*, vol. 124, no. 3, pp. 28–36, April 2015.

[96] T. Schierl, M. M. Hannuksela, Y. K. Wang, and S. Wenger, "System layer integration of high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1871–1884, Dec 2012.

[97] J. Vanne, M. Viitanen, and T. Hmlinen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1579–1593, Sept 2014.

[98] R. H. Gweon and Y. L. Lee, "Early termination of CU encoding to reduce HEVC complexity," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Torin, Italy, July 2011.

[99] K. Lee, H.-J. Lee, J. Kim, and Y. Choi, "A novel algorithm for zero block detection in high efficiency video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1124–1134, Dec 2013.

[100] J. Kim, J. Yang, K. Won, and B. Jeon, "Early determination of mode decision for HEVC," in *Proc. Picture Coding Symp. (PCS 2012)*, May 2012, pp. 449–452.

[101] K. Choi, S. H. Park, and E. S. Jang, "Coding tree pruning based CU early termination," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Turin, Italy, July 2011.

[102] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465–470, Feb 2013.

[103] J. H. Lee, C. S. Park, B. G. Kim, D. S. Jun, S. H. Jung, and J. S. Choi, "Novel fast PU decision algorithm for the HEVC video standard," in *Proc. 20th IEEE Int. Conf. on Image Process. (ICIP 2013)*, Sept 2013, pp. 1982–1985.

[104] K. McCann, B. Bross, W. J. Han, I. K. Kim, K. Sugimoto, and G. J. Sullivan, "High efficiency video coding (HEVC) test model 14 (HM14) encoder description," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, San Jose, US, March 2014.

[105] F. Bossen, D. Flynn, and K. Shring, "HM software manual," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, April 2014.

[106] N. Purnachand, L. Alves, and A. Navarro, "Fast motion estimation algorithm for HEVC," in *Proc. IEEE Int. Conf. on Consumer Electronics (ICCE 2012)*, Sept 2012, pp. 34–37.

[107] Z. Pan, Y. Zhang, S. Kwong, X. Wang, and L. Xu, "Early termination for TZSearch in HEVC motion estimation," in *Proc. IEEE Int. Conf. on Acous. Speech and Signal Process. (ICASSP 2013)*, May 2013, pp. 1389–1393.

[108] C. M. Kuo, Y. H. Kuan, C. H. Hsieh, and Y. H. Lee, "A novel prediction-based directional asymmetric search algorithm for fast block-matching motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 6, pp. 893–899, June 2009.

[109] S. H. Yang, J. Z. Jiang, and H. J. Yang, "Fast motion estimation for HEVC with directional search," *Electron. Lett.*, vol. 50, no. 9, pp. 673–675, April 2014.

[110] W. Dai, O. Au, S. Li, L. Sun, and R. Zou, "Adaptive search range algorithm based on cauchy distribution," in *Proc. IEEE Int. Visual Commun. and Image Process. (VCIP 2012)*, Nov 2012, pp. 1–5.

[111] C. C. Lou, S. W. Lee, and C. C. J. Kuo, "Adaptive motion search range prediction for video encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1903–1908, Dec 2010.

[112] Y. H. Ko, H. S. Kang, and S. W. Lee, "Adaptive search range motion estimation using neighboring motion vector differences," *IEEE Trans. Consum. Electron.*, vol. 57, no. 2, pp. 726–730, May 2011.

[113] S. Na and C. M. Kyung, "Activity-based motion estimation scheme for H.264 scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1475–1485, Nov 2010.

[114] Y. Liu, J. Wang, and F. Fu, "Adaptive search range adjustment and multi-frame selection algorithm for motion estimation in H.264/AVC," *J. Electron. Imaging.*, vol. 22, no. 2, pp. 023 031–023 031–8, June 2013.

[115] S. Lee, "Fast motion estimation based on adaptive search range adjustment and matching error prediction," *IEEE Trans. Consum. Electron.*, vol. 55, no. 2, pp. 805–811, May 2009.

[116] Z. Chen, Q. Liu, T. Ikenaga, and S. Goto, "A motion vector difference based self-incremental adaptive search range algorithm for variable block size motion estimation," in *Proc. 15th IEEE Int. Conf. on Image Process. (ICIP 2008)*, Oct 2008, pp. 1988–1991.

[117] L. Jia, O. C. Au, C. ying Tsu, Y. Shi, R. Ma, and H. Zhang, "A diamond search windowbased adaptive search range algorithm," in *Proc. 2013 IEEE Int. Conf. Multimedia and Expo Workshops (ICMEW)*, July 2013, pp. 1–4.

[118] S. Kim, D. K. Lee, C. B. Sohn, and S. J. Oh, "Fast motion estimation for HEVC with adaptive search range decision on CPU and GPU," in *Proc. IEEE China Summit & Int. Conf. on Signal and Information Process. (ChinaSIP2014)*, July 2014, pp. 349–353.

[119] L. Du, Z. Liu, T. Ikenaga, and D. Wang, "Linear adaptive search range model for uni-prediction and motion analysis for bi-prediction in HEVC," in *Proc. 20th IEEE Int. Conf. on Image Process. (ICIP 2014)*, Oct 2014, pp. 3671–3675.

[120] W. D. Chien, K. Y. Liao, and J. F. Yang, "Enhanced AMVP mechanism based adaptive motion search range decision algorithm for fast hevc coding," in *Proc. 21st IEEE Int. Conf. on Image Process. (ICIP 2014)*, Oct 2014, pp. 3696–3699.

[121] H. Tohidypour, M. Pourazad, P. Nasiopoulos, and V. Leung, "A content adaptive complexity reduction scheme for HEVC-based 3D video coding," in *Proc. 18th IEEE Int. Conf. on Digital Signal Process. (DSP 2013)*, July 2013, pp. 1–5.

[122] Y. H. Lin and J. L. Wu, "A depth information based fast mode decision algorithm for color plus depth-map 3d videos," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 542–550, June 2011.

[123] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M33, March 2001.

[124] T. K. Lee, Y. L. Chan, and W. C. Siu, "Motion estimation in low-delay hierarchical P-frame coding using motion vector composition," *J. Visual Commun. Image Represent.*, vol. 24, no. 8, pp. 1243–1251, 2013.

[125] T. K. Lee, Y. L. Chan, and W. C. Siu, "Motion vector composition in low-delay hierarchical P-frame coding," in *Proc. IEEE China Summit Int. Conf. Signal Info. Process. (ChinaSIP 2013)*, Aug 2013, pp. 551–555.

[126] JVT, "H.264/AVC reference software JM17.2," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16

WP3 and ISO/IEC JTC1/SC29/WG11, 2013. [Online]. Available: http://iphome.hhi.de/suehring/tml/download/

[127] M. Winken, H. Schwarz, D. Marpe, and T. Wiegand, "Joint optimization of transform coefficients for hierarchical B picture coding in H.264/AVC," in *Proc. 13th IEEE Int. Conf. on Image Process. (ICIP 2007)*, vol. 4, Sept 2007, pp. IV – 89–IV – 92.

[128] C. H. Fu, T. K. Lee, Y. L. Chan, and W. C. Siu, "An efficient motion vector composition algorithm for fast-forward playback in a video streaming system," *J. Visual Commun. Image Represent.*, vol. 21, no. 8, pp. 939–947, 2010.

[129] T. K. Lee, Y. L. Chan, and W. C. Siu, "Adaptive search range by neighbouring depth intensity weighted sum for HEVC texture coding," *Electron. Lett.*, vol. 52, no. 12, pp. 1018–1020, 2016.

[130] G. Sullivan, J. Boyce, Y. Chen, J.-R. Ohm, C. Segall, and A. Vetro, "Standardized extensions of high efficiency video coding (HEVC)," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1001–1016, Dec 2013.

[131] T. K. Lee, Y. L. Chan, and W. C. Siu, "Adaptive search range for HEVC motion estimation based on depth information," *IEEE Trans. Circuits Syst. Video Technol.*, 2016.

[132] T. K. Lee, Y. L. Chan, and W. C. Siu, "Depth-based adaptive search range algorithm for motion estimation in HEVC," in *Proc. Int. Conf. Digital Signal Process. (DSP2014)*, Aug 2014, pp. 919–923.

[133] J. Jiao, R. Wang, W. Wang, S. Dong, Z. Wang, and W. Gao, "Local stereo matching with improved matching cost and disparity refinement," *IEEE MultiMedia*, vol. 21, no. 4, pp. 16–27, Oct 2014.

[134] L. Zhang and W. J. Tam, "Stereoscopic image generation based on depth images for 3D TV," *IEEE Trans. Broadcast.*, vol. 51, no. 2, pp. 191–199, June 2005.

[135] JCT-VC, "HEVC software repository (main at HHI)," 2014. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/branches/HM-14.0-dev/

[136] F. Bossen, "Common HM test conditions and software reference configurations, jctvc-l1100," JCT-VC, 2013.

[137] Q. Zhang, M. Chen, X. Huang, N. Li, and Y.Gan, "Low-complexity depth map compression in HEVC-based 3D video coding," *EURASIP J. Image and Video Process.*, pp. 1–14, 2015.

[138] K. McCann, W. J. Han, I. K. Kim, J. H. Min, E. Alshina, A. A. annd Tammy Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, M. S. Cheon, and N. Shlyakhov, "Samsungs response to the call for proposals on video compression technology," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Dresden, Italy, Apr 2010.

[139] L. Zhao, X. Guo, S. Lei, S. Ma, and D. Zhao, "Simplified AMVP for high efficiency video coding," in *Proc. IEEE Visual Communications and Image Processing (VCIP 2012)*, Nov 2012, pp. 1–4.

[140] P. Helle, S. Oudin, B. Bross, D. Marpe, M. O. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for quadtree-based partitioning in HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1720–1731, Dec 2012.