Tampereen teknillinen yliopisto. Julkaisu
Tampere University of Technology. Publication

Jin Li

Advances on Video Coding Algorithms for Next Generation Mobile Applications

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 5th of August 2011, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2011

Abstract

Visual communication via mobile devices is becoming more and more popular. However, current video coding standards are not initially designed for this type of application area. Traditionally, high compression efficiency is usually achieved at the expense of increasing computations. The newest standard, H.264/AVC significantly outperforms the others in terms of coding efficiency, however, the complexity is greatly increased. This sets challenges on implementation, especially on mobile devices, which typically possess constrained computational resources. Thus, there is great significance and interest in reducing the computations for video coding. Currently, the emerging standard, i.e., High Efficiency Video Coding (HEVC), claims an aggressive goal to realize high efficiency video coding with low complexity.

This thesis deals with the problems of advancing video coding algorithms for real-time high performance, focusing on applications for the next generation mobile devices. The goal of the research is to develop new techniques in order to reduce the coding complexity and improve the real-time performance while maintaining competitive video quality.

The research work presented in this thesis can be categorized into two areas. First, different techniques are provided for low complexity video encoding and/or decoding for standardized video codec, e.g., H.264/AVC and MPEG-4 Visual. The efforts particularly focus on the prediction of zero-quantized DCT coefficients which can be skipped for complexity reduction. Second, new algorithms and methods are applied to 3-D DCT video coding for improvements of both the coding efficiency and complexity reduction.

Compared to existing techniques, the proposed algorithms in this thesis extend the prediction from 2-D DCT to 1-D DCT on the row and column directions. Thus, they are particularly efficient for partially zero-quantized DCT blocks. In addition, the proposed algorithms lend themselves the butter-fly implementation structure which is commonly used in video coding. The experiments show that the proposed method outperforms the earlier methods in terms of complexity reduction. As good supplement to the effort on inter transformation, complexity reduction for intra 2-D DCT computations has also been intensively researched in this thesis.

In addition, variable size 3-D DCT video coding methods are proposed in the thesis. Compared to competing techniques, the proposed approaches improve the 3-D DCT video coding scheme in both the complexity reduction and coding efficiency.

In addition, the evaluation criterion on this topic is further improved by lending the algorithms to the implementation structure. For instance, since the DCT coefficients are not individually computed, the discussion has been focused on the complexity reduction in terms of 1-D transforms and arithmetic operation numbers instead of individual coefficients. Furthermore, PSNR and bitrate have been separately compared in the related references and thus “negligible video quality degradation” is drawn as conclusion. However, rate-distortion comparisons in the thesis show a comparable overall video quality to the original codec, which means the proposed models do not necessarily result in any additional visual distortion.

Preface

The research presented in this thesis has been carried out during the years 2006-2010 at the Department of Signal Processing and the Department of Computer Systems at Tampere University of Technology.

First and foremost, I would like to express my deepest gratitude to my supervisors Prof. Moncef Gabbouj and Prof. Jarmo Takala for encouragement and scientific guidance throughout the years as well as careful review of the thesis.

I am very grateful to the pre-examiners, Prof. C.-C. Jay Kuo from the University of South California and Dr. Huifang Sun from Mitsubishi Electric Research Labratories, for their efforts and constructive comments which no doubt have improved the thesis.

Most papers included in this thesis were prepared in a collaboration project between Tampere University of Technology and Prof. Hexin Chen's team at Jilin University, China. I owe my thanks to Prof. Chen for the countless hours spent for this work and setting up the collaboration in China.

The discussions and cooperation with my colleagues have contributed substantially to this work. Many thanks go in particular to Jari Koivusaari. I have especially benefited by the guidance and friendship of Jari who granted me his time and efforts for my research and life.

In addition, I owe my thanks to my colleagues and friends, especially to Harri Sorokin, Petteri Laakkonen, Xizhi Zhu, Chenghao Liu, Cixun Zhang, and many others.

I would like to thank Virve Larmila, Ulla Siltaloppi, Elina Orava, Johanna Reponen, and Irmeli Lehto for their great help on important administration work.

Financial support from the Academy of Finland and Nokia Foundation is gratefully acknowledged.

Last but not least, I wish to express my warmest thanks to my parents, my wife, and my daughter for their constant love and support in my life.

Contents

List of Publications

This thesis is composed of a summary part and eight publications listed below as appendices. The publications are referred to as [P1], [P2], *etc.* in the thesis.

[P1] J. Li, M. Gabbouj, and J. Takala, “Zero-Quantized Inter DCT Coefficient Prediction for Real-Time Video Coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, accepted 2011.

[P2] J. Li, M. Gabbouj, and J. Takala, “Merged Inverse Quantization and IDCT for Optimized Decoder Implementation,” Proceedings of *European Signal Processing Conference*, pp. 948-952, Glasgow, UK, Aug. 24-28, 2009.

[P3] J. Li, J. Takala, M. Gabbouj, and H. Chen, “A Detection Algorithm for Zero Quantized DCT Coefficients in JPEG,” Proceedings of *IEEE International Conference on Acoustic, Speech and Signal Processing*, pp. 1189-1192, Las Vegas, USA, Mar. 30-Apr. 4, 2008.

[P4] J. Li, M. Gabbouj, and J. Takala, “Hybrid Modeling of intra DCT Coefficients for Real-Time Video Encoding,” *EURASIP Journal on Image and Video Processing,* Volume 2008, Article ID 749172, 13 pages, doi:10.1155/2008/749172.

[P5] J. Li, J. Takala, M. Gabbouj, and H. Chen, “Use of Adaptive Resizing in 3-D DCT Domain for Video Coding,” Proceedings of *Picture Coding Symposium*, Lisbon, Portugal, Nov. 7-9, 2007.

[P6] J. Li, M. Gabbouj, J. Takala, and H. Chen, “Direct 3-D DCT-to-DCT Resizing Algorithm for Video Coding,” Proceedings of *International Symposium on Image and Signal Processing and Analysis*, pp. 109-114, Salzburg, Austria, Sept. 16-18, 2009.

[P7] J. Li, J. Takala, M. Gabbouj, and H. Chen, “Variable Temporal Length 3D DCT-DWT Based Video Coding,” Proceedings of *IEEE International Symposium on Intelligent Signal Processing and Communication Systems,* pp. 506-509, Xiamen, China, Nov. 28- Dec. 1, 2007.

[P8] J. Li, J. Takala, M. Gabbouj, and H. Chen, “Modeling of 3-D DCT Coefficients For Fast Video Encoding,” Proceedings of *IEEE International Symposium on Communications, Control and Signal Processing,* pp. 634-638, Malta, Mar. 12-14, 2008.

List of Acronyms

ADD	Addition
AVC	Advanced Video Coding
CAVLC	Context-Adaptive Variable-Length Coding
CABAC	Context-Adaptive Binary Arithmetic Coding
CHC	Conversational High Compression Profile of H.263
CR	Compression Ratio
DCT	Discrete Cosine Transform
DHT	Discrete Haar Transform
DPCM	Differential Pulse-Code Modulation
HEVC	High Efficiency Video Coding
IDCT	Inverse Discrete Cosine Transform
MUL	Multiplication
PSNR	Peak Signal-to-Noise Ratio
QDCT	Quantized Discrete Cosine Transform
SAD	Sum of Absolute Differences
SSD	Sum of Square Differences
TMN5	Test Model Number 5
TMN8	Test Model Number 8
ZQDCT	Zero-Quantized DCT Coefficient
3-D DCT	Three-Dimensional Discrete Cosine Transform

Chapter 1

Introduction

MOST video compression schemes to date are based on the principle of *Hybrid Video Coding* [89]. This definition refers to two different techniques used in order to exploit spatial redundancy and temporal redundancy. Spatial compression is indeed obtained by means of a transform based approach, which makes use of the Discrete Cosine Transform (DCT) [2][19][53][97], or its variations. Temporal compression is achieved by computing a *Motion Compensation* prediction of the current frame and then encoding the corresponding prediction errors. A general scheme of a hybrid encoder is given in Fig.1.1.

All international video coding standards – chronologically, H.261 [32], MPEG-1[29], MPEG-2/H.262 [33], H.263 [34], MPEG-4 Visual [30], and H.264/AVC [35], share this basic structure and have been the engines behind the commercial success of digital video compression. Since 1990s, when the technology was in its infancy, these video coding standards have played pivotal roles in spreading the technology by providing powerful interoperability among various products, while allowing enough flexibility for ingenuity in optimizing and fitting a given application. Nowadays, various digital video communication systems, which deploy video coding standards, are used in every-day life by many people.

The newest video coding standard, H.264/AVC is currently the state-of-the-art compression tool and significantly outperforms previous standards. According to [69][91][92], for video streaming applications, H.264/AVC main profile allows an average bit rate saving of about 63% compared to MPEG-2 video and about 37% compared to MPEG-4 Visual. For video conferencing applications, H.264/AVC baseline profile achieves an average bitrate saving of about 40% compared to H.263 baseline and about 27% compared to H.263 conversational high compression profile (CHC). For entertainment-quality application, the bitrate saving of H.264/AVC compared to MPEG-2 is about 45% on average [91]. Interestingly, experiments in [60] show that the measured rate-distortion performance of H.264 main profile is even better than that of the state-of-the-art in still image compression as exemplified by JPEG2000 [72] when test sequence *Canoe* is encoded in *intra* mode only, i.e., each field of the whole sequence is coded in *intra* mode only. Other test cases [54] also show that for up to 1280 × 720 pel HDTV signals, the pure *intra* coding performance of H.264 main profile is comparable or better than that of Motion-JPEG2000 [31]. Therefore, the advanced video coding standard, i.e., H.264/AVC, has become a widely deployed coding technique in numerous products and services, such as Blu-ray Disc, Adobe Flash, video conference and mobile television.

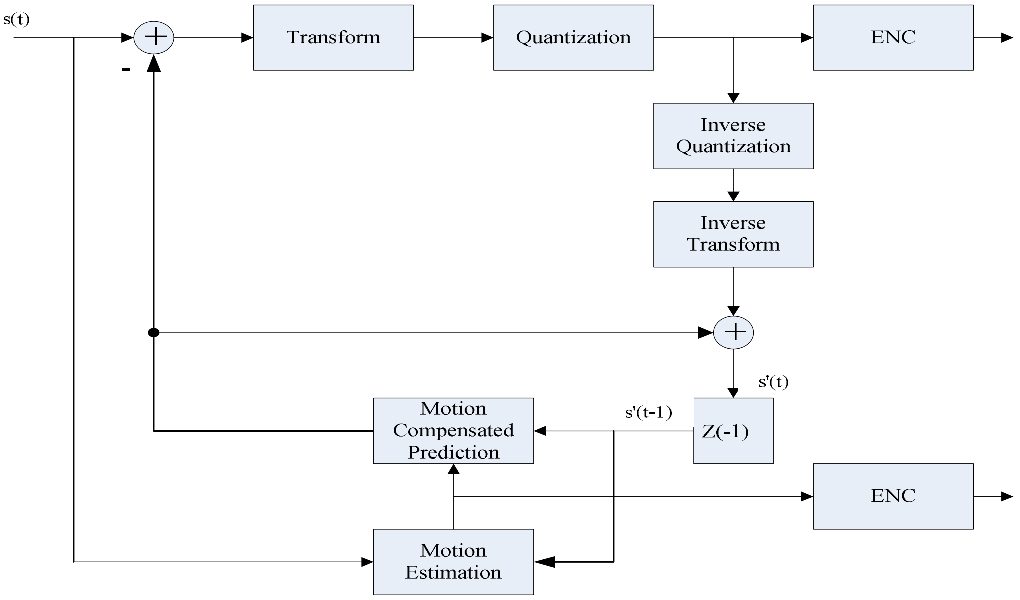


Fig.1.1 General scheme of a hybrid video encoder, where the current image to be coded $s(t)$ is predicted by a motion compensated prediction from an already transmitted image $s'(t-1)$. $e(t)$ is the prediction error.

However, the improvements of coding efficiency are usually achieved at the expense of increasing computational complexity. The initial experiments carried out in [46] indicate that a highly optimized H.26L encoder (designation of H.264 in the early development phase) is at least 3.8 times more complicated than the equivalently optimized H.263 encoder. As the most powerful compression technique, H.264/AVC also exceeds previous standards in terms of computational complexity. A lot of experiments in [65] show that the complexity increases with more than one order of magnitude between MPEG-4 Visual simple profile and H.264/AVC main profile and with a factor of two for the decoder. Currently, the emerging standard, i.e., H.265 [26], claims an aggressive goal to reduce the bit rate by half with the same coding efficiency compared to H.264/AVC. However, the overall complexity can be expected to increase three times or more compared to H.264/AVC [79]. On the other hand, visual communications via mobile devices become more and more popular. However, the video coding standards, e.g., MPEG-4 and H.264/AVC, are not initially designed for this kind of scenario. High complexity in video coding standards limits the application on resource-constrained battery-powered mobile devices as well as the real-time performance of the encoder. Thus, there is a great significance and interest to reduce the computations for video coding.

Previously, as the most computationally intensive part, motion estimation has received a number of investigations [28][16][39][40][41] [81][93]. For instance, experiments performed on H.263 TMN5 [74] show that about 82% of the encoding time is spent on motion estimation when encoding Miss American sequence at 20 Kbit/s. In addition, it occupies about 70% of the overall encoding computations for H.264/AVC codec [1][13][52][99]. However, as the time on motion estimation continues to decline, the time spent on the other stages such as transform and quantization become relatively more significant. The percentage of processing time spent on DCT, inverse DCT, quantization and inverse quantization increases from 9% in TMN5 to 26% in TMN8 [75]. As the motion estimation is optimized, the portion of the transform and quantization in H.264/AVC becomes approximately 24% of the total computations [57]. Therefore, it is of great importance to reduce the complexity of transform and quantization.

The complexity reduction of DCT and quantization plays an important role in improving the real-time performance of video encoding. So far, many algorithms have been developed for fast discrete cosine transform calculation, e.g., [19][23][44][53]. These algorithms can be classified in two categories: direct and indirect algorithms. In 1991, Kou *et al.* [43] proposed a direct computation method that slightly reduces the number of multiplications and additions. On the other hand, indirect algorithms exploit the relationship between DCT and other transforms to reduce complexity. These algorithms include the calculations of DCT through the Hartley transform [53], polynomial transform [19] and by paired transform [23]. Currently, many fast algorithms for multi-dimensional DCT computation are also emerging [11][15]. However, these algorithms still contain redundant computations as they do not take into account the zero-quantized DCT (ZQDCT) coefficients.

When reducing redundant computations due to the ZQDCT coefficients, two types of prediction methods are considered: the prediction for the ZQDCT coefficients of intra blocks and the prediction of the residual pixel blocks. Nishida proposed a zero-value prediction for fast DCT calculation in [59]. If consecutive zero elements are produced during the DCT operation, the remaining transform and quantization are skipped. This method can be directly applied to the DCT and quantization. Although it reduces the total computations of DCT by 29% and quantization by 59% when applied to MPEG-2, the video quality is degraded by 1.6 dB on the average.

However, as inter DCT accounts for most of the discrete cosine transforms compared to intra DCT, complexity reduction regarding inter DCT has a more significant effect on improving real-time performance.

The existing methods can reduce the complexity of inter DCT and quantization, particularly in terms of all-zero-quantized DCT blocks. However, since the DCT is usually implemented in a row and column structure, the efficiency of these algorithms is in practice highly lowered for non-all-zero-quantized DCT blocks. Moreover, another drawback in the Laplacian and Gaussian thresholds existing in the prediction of ZQDCT coefficients for the non-all-zero-quantized DCT blocks. Since the thresholds are symmetric along the diagonal line from the top-left to the bottom-right, the prediction results also have a symmetric property when uniform quantization is applied for transforms in inter frames. However, this is not always the case.

1.1. OBJECTIVE AND SCOPE OF THE RESEARCH

The thesis presents methods for reducing the computational complexity of transform and quantization in standardized video coding and other coding scenario. Other factors affecting the end-user satisfaction, such as rate-distortion performance, are also taken into consideration when evaluating the algorithms. Particular emphasis is given to video compression applications that are relevant for battery-powered resource-constrained mobile devices. The goal of the research is to reduce the coding complexity and improve the real-time performance without degrading the video quality or at the expense of negligible quality degradation.

This thesis primarily focuses on complexity reduction of standardized video coding, such as MPEG-4 Visual and H.264/AVC. It is evident, however, that other coding scenarios also play a very important role for advancing video compression and may be standardized in the future. Thus, the thesis also pays attention to relevant video coding techniques, i.e., three-dimensional (3-D) DCT video coding and considers optimization of both coding efficiency and complexity.

The research work presented in this thesis can be categorized into two areas. First, different techniques are provided for low complexity video encoding and video decoding for standardized video codec, e.g., H.264/AVC and MPEG-4 Visual. Second, new algorithms and methods are applied to 3-D DCT video coding for improvements of both the coding efficiency and complexity reduction compared to competing techniques.

One of the most important objectives of this thesis is to reduce the computational complexity of video encoding and video decoding while, keeping the video quality at the same level or at least comparable to that of the original coding schemes. Therefore, both the overall complexity of encoding and/or decoding and the rate-distortion performance are evaluated before drawing conclusions for each proposed technique in the thesis.

1.2. MAIN RESULTS

1.2.1 Prediction for Inter Coding Mode in Video Coding Standards

An improved prediction method is proposed in the thesis to reduce the complexity of the transform and quantization, especially for those predicted as non-all-zero-quantized DCT blocks. First, a Gaussian distribution based model is employed to predict all-zero-quantized DCT blocks prior to the transform and quantization as commonly done in the literature. Subsequently, a new prediction method based on the Gaussian distribution is analytically computed and used to predict all zero-quantized coefficients along the row and column transforms. Then, sufficient conditions for the DCT coefficients to be quantized to zeros are developed to further reduce the complexity. Finally, a hybrid multi-threshold based model is proposed for the all-zero-quantized DCT blocks and the all-zero-quantized rows and columns. Compared to competing techniques, we extend the prediction to 1-D DCT on the row and column directions by developing a new prediction method based on Gaussian distribution. This algorithm lends itself to the traditional row and column transform structure commonly used in video coding. The results show that the proposed method outperforms the earlier methods in terms of complexity reduction. The advantage is particularly evident for non-all-zero-quantized DCT blocks.

Even though the decoder has less complexity than the encoder in video standards, low complexity in the decoder is equally, if not more important than in the encoder. An efficient method to reduce the decoding complexity is proposed in this thesis. The proposed technique merges the inverse quantization and inverse DCT into a single procedure. Thus, the computations related to inverse quantization are skipped. Compared to the reference encoder, the proposed method is able to further reduce the multiplications for inverse transform and quantization and thus, lends itself to battery-powered embedded systems with limited resources. Moreover, the experiments show that the proposed method does not cause any video quality degradation.

1.2.2 Prediction for Intra Coding Mode in Video Coding Standards

Transform and quantization for intra mode are only a minor field in video coding, thus research on complexity reduction of inter transform and quantization not been very intensive. However, it is still worthwhile to work in this topic, particularly when video sequences are encoded with intra mode only or a lot of images are compressed in mobile applications.

This thesis proposes a sufficient condition based prediction method for the intra ZQDCT coefficients. Experiments on JPEG codec show that the computations for 2-D DCT and quantization are decreased by 10-50% compared to the original codec depending on the quantization parameters and the image textures. Moreover, no video quality degradation is observed.

Later on, we extend existing techniques for inter coding mode to intra DCT and quantization, aiming to simplify the encoding complexity and achieve better real-time performance with minimal video quality degradation. Although the proposed model is implemented based on the 8×8 DCT, it can be directly applied to other DCT-based image/video coding standards. As for the transform and quantization including intra and inter modes, the overall complexity is reduced by 1.82 - 5.63% on average. Although the number of addition operations remains high, the required multiplication operations are reduced by approximately 53 - 90%. Therefore, the overall processing time for intra DCT and quantization is reduced compared to the reference encoder and the original XVID encoder.

1.2.3 New Algorithms for 3-D DCT Video Coding

A 3-D DCT algorithm with variable cube size is proposed in this thesis. The mode selection is determined relying on the local motion activity within each cube. Totally, three modes are utilized to perform the transform in temporal direction using the traditional resizing algorithm. Experiments show a promising improvement in terms of coding efficiency. This resizing algorithm is further optimized in our research work. The DCT-to-DCT base resizing algorithm is proposed to reduce the required number of operations. For 8 × 8 DCT, this improved method only requires six multiplications (MUL) and nine additions (ADD). While the inverse transform and the computations in the spatial domain in the traditional resizing methods take 8 multiplications and 14 additions. Therefore, the proposed method is superior to other approaches in terms of computational complexity.

All existing variable lengths of 3-D DCT algorithms use thresholds. Although the selection of the threshold considerably affects the algorithm performance, there is no framework to choose them effectively. For this problem, an adaptive hybrid algorithm for 3-D DCT based video coding is proposed. Compared to other variable length 3-D DCT schemes, the proposed model adaptively finds the optimal mode decision.

1.2.4 Summary of The Contributions

Publication [P1] proposes a prediction method to reduce the complexity of inter transform and quantization. Compared to existing techniques, we extend the prediction to 1-D DCT on the row and column directions by developing a new prediction method based on Gaussian distribution. This algorithm lends itself to the traditional row and column transform structure commonly used in video coding. The advantage is particularly evident for non-all-zero-quantized DCT blocks. Besides, some misconceptions on evaluation criterion on this topic are clarified and more consistent to the implementation structure. Experiments on both MPEG-4 Visual and H.264/AVC show that the proposed algorithm outperforms other approaches.

Publication [P2] proposes an efficient technique to reduce the complexity of implementation for inverse discrete cosine transform (IDCT)-based video decoders. The proposed method merges the inverse quantization and IDCT into a single procedure. Thus, the computations related to inverse quantization are omitted. This approach and does not change the standard bitstream syntax, so it can be applied to other DCT and quantization based video coding.

Publication [P3] presents an early detection algorithm to predict the zero-quantized DCT coefficients for fast image encoding. This approach can terminate the calculation for high frequency DCT coefficients if they are predicted as zeros. Therefore, the computations for both 2-D DCT and quantization can be partially saved. This algorithm is very beneficial for power-constrained portable devices when a huge number of images are encoded.

Publication [P4] proposes a hybrid statistical model used to predict the zero-quantized DCT (ZQDCT) coefficients for intra transform and to achieve better real-time performance. Most works on this topic have been focused on the residual pixels in video coding. As a good supplement, this approach extends the prediction to the intra DCT and quantization. Experiments show that it further reduces the overall complexity and benefits the power-constrained devices.

Publication [P5] proposes an adaptive resizing algorithm in DCT domain for 3-D DCT based video codec. Based on the local motion activity, three variable temporal length 3-D DCT modes are developed to improve the compression performance. Compared to other variable size DCT techniques, the proposed approach uses a resized $8 \times 8 \times 4$ cube for mild motions instead of classifying these cubes as stationary background, which only the first frame is encoded and the others are duplicated from the first one at the decoder. Therefore, sudden scene changes can be avoided.

Publication [P6] proposes a direct DCT-to-DCT resizing algorithm for 3-D DCT video coding based on the approach proposed in [P5]. This scheme directly resizes the DCT blocks in the transform domain using the DCT-to-DCT algorithm. Therefore, the inverse transform and other computations in [P5] are skipped. In addition, non-uniform quantization is used to improve the coding efficiency. Simulations show that the approach in [P6] outperforms [P5] in terms of both complexity and coding performance.

Publication [P7] The thresholds for variable size 3-D DCT video coding are empirically determined, but they are not necessarily the optimum values for different video sequences. The proposed model can select the different DCT modes based on the quantized high frequency DCT coefficients. Thresholds are not required any more. Experimental results show that the proposed approach has improvement of coding efficiency over the conventional fixed-length 3-D DCT coding and other variable length 3-D DCT coding.

Publication [P8] The key advantage of 3-D DCT video coding is the low encoding complexity by replacing the motion compensation with an additional transform. Particularly, the transform becomes the most complex part in 3-D DCT video coding. This paper proposes a prediction algorithm to skip the calculations for the zero-quantized DCT coefficients along both spatial and temporal directions. Experiments show that the 3-D DCT has been greatly simplified with this approach and the overall encoding time is significantly faster than the original video coding.

1.2.5 Author's Contributions

The author's contributions to real-time standardized video coding and 3-D DCT video coding are included in the publications, denoted as [P1], [P2], …, [P8]. While all publications have resulted from team work, the author's contribution to each has been essential.

The author is the main author in all the publications. He proposed the methods and carried out the simulations as well as the writing of the publications.

1.3. THESIS OUTLINE

This thesis is composed of two parts: an introduction composed of five chapters followed by eight publications. In Chapter 2, the video coding standards, i.e., MPEG-4 Visual and H.264/AVC are first reviewed, focusing on those features relevant for the thesis. The features of MPEG-4 Visual include object based video coding, the basic transformation and quantization tools in the simple profile and the advanced simple profile. H.264/AVC introduces specified profiles and levels. Particularly, certain features in the baseline profile are reviewed, such as various kinds of transform and quantization.

In Chapter 3, the research on complexity reduction of transform and quantization for video coding standards is reviewed. The competing works on MPEG-4 Visual and H.264/AVC are firstly introduced. Following the introduction of existing methods, the motivation of the research work in this area is revealed and the contributions of the thesis are then briefly described. These techniques can be categorized into two types: complexity reduction for inter mode and for intra mode. All the techniques including the state-of-the-art methods and the proposed methods achieve lower encoding or decoding complexity with the same or comparable rate-distortion performance to the original codec.

Chapter 4 introduces the 3-D DCT video coding. The overview of 3-D DCT video coding is first briefly described, such as the advantages over the motion estimation/compensation based video coding and its applications in mobile devices. Then, the effects to improve the coding efficiency of 3-D DCT video coding are reviewed. The state-of-the-art techniques mainly include the utilization of variable size 3-D DCT along the temporal axis depending on the motion activities. Finally, the contributions of the thesis for 3-D DCT video coding are described regarding improvements of coding efficiency and complexity reduction. Chapter 5 concludes the thesis.

Chapter 2

The Video Coding Standards - MPEG-4 Visual and H.264/AVC

AN understanding of the process of creating the standards can be helpful when interpreting or implementing the documents themselves. In this Chapter, we give an overview of two video coding standards, i.e., MPEG-4 Visual and H.264/AVC baseline profile. We discuss the features and parameters in the standards that are relevant for this thesis. Finally, we briefly compare the two standards and summarize the differences between MPEG-4 Visual and H.264/AVC baseline profile.

2.1. MPEG-4 VISUAL

2.1.1 Overview of MPEG-4 Visual

Compared to previous standards, one of the key contributions of MPEG-4 Visual [63] is the utilization of object-based coding technique, which is a move away from the conventional block-based methods in common use today. For instance, a video scene may be separated into a background object several foreground objects as shown in Fig. 2.1. The separate objects may be coded with different visual qualities and temporal resolutions to reflect their importance to the end users. Therefore, this approach is much more flexible than the rectangular frame structure.

However, so far, the object based representation has not been demonstrated to be practical for several reasons. One reason is simply that the segmentation of video content into arbitrarily shaped and semantically meaningful video objects remains a very difficult task. Another is that adding object handling capability to decoders increases the computational complexity of decoder, although typically existing video applications may not ordinarily substantially benefit from the extra capability. Although a number of investigations have been done so far by many researchers such as [7][24][56][62][78], these features of MPEG-4 Visual have not been widely deployed in products and remain primarily only of interest for academic and research purposes today. However, some other capabilities of MPEG-4 Visual, such as the simple profile and advanced simple profile, have achieved widespread deployment in mobile applications and Internet-based video for PC-based viewing [14][25][50][63].

(a)

(b)

(c)

Fig.2.1 Video scene consisting of four video objects, where (a) the video scene, (b) the objects and (c) the background.

2.1.2 Transform in MPEG-4 Visual

For intra and inter coding mode, blocks of luma and chroma samples in MPEG-4 Visual are transformed using an 8×8 DCT during encoding and inverse DCT during decoding. If $\boldsymbol{f}$ is the 8×8 pixel block and $f(i,j)$ indicates the pixel value at position (i,j), where $0 \leq i,j \leq 7$, and $\boldsymbol{F}$ is the transformed matrix and $F(u,v)$ is the DCT coefficient, the DCT in MPEG-4 Visual are defined as

$$F(u,v) = \frac{1}{4}c(u)c(v)\sum_{i=0}^{7}\sum_{j=0}^{7} f(i,j)\cos\frac{(2i+1)u\pi}{16}\cos\frac{(2j+1)v\pi}{16} \qquad (2.1)$$

where $c(x) = 1/\sqrt{2}$, for $x = 0$, and $c(x) = 1$, otherwise.

Alternatively, the DCT in (2.1) can be expressed in matrix form as

$$\boldsymbol{F} = \boldsymbol{A}\boldsymbol{f}\boldsymbol{A}^{\boldsymbol{T}} \qquad (2.2)$$

where the elements of the DCT matrix $\boldsymbol{A} = \{A[i,j]\}$ are

TABLE 2.1 RELATION BETWEEN dc_s AND Q_p

Block type	$Q_p \le 4$	$5 \le Q_p \le 8$	$9 \le Q_p \le 24$	$25 \le Q_p$
dc_s (luma)	8	$2Q_p$	$Q_p + 8$	$2Q_p - 16$
dc_s (chroma)	8	$(Q_p + 13)/2$	$(Q_p + 13)/2$	$Q_p - 6$

$$A[i,j] = \begin{cases} \frac{1}{\sqrt{8}} & i = 0;\ 0 \le j < 8 \\ \sqrt{\frac{2}{8}} cos \frac{(2j+1)i\pi}{16} & 1 \le i < 8;\ 0 \le j < 8 \end{cases}$$

Similarly, the inverse DCT is defined as

$$f(i,j) = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} c(u)c(v)F(u,v) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \qquad (2.3)$$

The MPEG-4 Visual specifies the inverse quantization for the quantized DCT coefficients in the decoder, which is controlled by a quantization scale parameter, Q_p, ranging from 1 to 31. Larger values of Q_p produce larger quantization step size and therefore leading to higher compression performance. However, the visual quality can be serious degraded.

2.1.3 Two Quantization Methods in MPEG-4 Visual

In MPEG-4 Visual, two inverse quantization methods are defined: the basic method ‘method 1’ and a more flexible method as ‘method 2’. In MPEG-4 Visual, ‘method 2’ is defined as the default method and the simple profile only deploys ‘method 2’ for the inverse quantization.

‘Method 2’ inverse quantization operates as follows. The DC coefficient in an intra coded macroblock is inversed quantized by

$$DC = DC_Q \times dc\_s \qquad (2.4)$$

where DC_Q is the quantized DCT coefficient, DC is the inverse quantized coefficients and dc_s is the parameter defined in the standard. In the short header mode $dc\_s = 8$, i.e., all intra DC coefficients are inverse quantized by a factor of eight. Otherwise, dc_s is calculated according to the value of Q_p as shown in Table 2.1. The DCT coefficients except the intra DC coefficients are inverse quantized as follows

$$|F| = \begin{cases} Q_p(2|F_Q| + 1) & \text{if } Q_p \text{ is } odd \text{ and } F_Q \ne 0 \\ Q_p(2|F_Q| + 1) - 1 & \text{if } Q_p \text{ is } even \text{ and } F_Q \ne 0 \\ 0 & \text{if } F_Q = 0 \end{cases} \qquad (2.5)$$

TABLE 2.2 WEIGHTING MATRIX W

10	20	20	30	30	30	40	40
20	20	30	30	30	40	40	40
20	30	30	30	40	40	40	40
30	30	30	30	40	40	40	50
30	30	30	40	40	40	50	50
30	40	40	40	40	40	50	50
40	40	40	40	50	50	50	50
40	40	40	50	50	50	50	50

where F_Q is the quantized coefficient and F is the rescaled coefficient. The sign of F is made the same as the sign of F_Q.

An alternative inverse quantization method, 'method 1', is supported in the advanced simple profile. Intra DC coefficient remains the same as 'method 2', however, other quantized coefficients may be inverse quantized using 'method 1'.

Quantized coefficient $F_Q(u, v)$ is inverse quantized to produce $F(u, v)$, where u, v are the coordinates of the coefficients in the transform block, as follows

$$F(u,v) = \begin{cases} 0 & \text{if } F_Q(u,v) = 0 \\ \frac{[(2F_Q(u,v)+k)W(u,v)Q_p]}{16} & otherwise \end{cases} \tag{2.6}$$

where

$$k = \begin{cases} 0 & intra\ blocks \\ 1 & F_Q(u,v) > 0, nonintra \\ -1 & F_Q(u,v) < 0, nonintra \end{cases} \tag{2.7}$$

and W is a matrix of weighting factors. In 'method 2', all coefficients except intra DC are quantized and inverse quantized with the same quantizer step size. However, 'method 1' allows an encoder to vary the step size depending on the position of the coefficient, using the weighting matrix W. For instance, better subjective performance may be achieved by increasing the step size for high frequency coefficients and reducing it for low-frequency coefficients [63]. Table 2.2 shows an example of a weighting matrix W.

In fact, the default quantization method, i.e., method 2, is sometimes known as "H.263 quantization", and the alternative 'method 1' as MPEG-4 quantization.

2.2 THE ADVANCED VIDEO CODING STANDARD-H.264/AVC

With significant improvement of compression performance, H.264/AVC is now the state-of-the-art video coding standard and has been widely adopted by industry. The most crucial features deployed in H.264/AVC include multiple reference frames, weighted bi-prediction, context adaptive VLC and CABAC entropy coding, intra prediction, and motion estimation/compensation with multiple partitions. Target applications of H.264/AVC include two-way video communication such as video-

TABLE 2.3 QUANTIZATION STEP SIZE IN H.264 CODEC

Q_p	0	1	2	3	4	5	6	7	8	9	10	11	12
$Qstep$	0.625	0.6875	0.8125	0.875	1	1.125	1.25	1.375	1.625	1.75	2	2.25	2.5
Q_p	…	18	…	24	…	30	…	36	…	42	…	48	…
$Qstep$	…	5	…	10	…	20	…	40	…	80	…	160	…

conferencing and video telephony, coding for broadcast and high quality video and video streaming over packet networks. In addition, H.264/AVC specifies different profiles focusing on various applications and several levels with different sets of constraints such as complexity, latency and memory size. In the following, the thesis will give a brief introduction of the transform and quantization in the baseline profile, which is relevant for the work.

H.264/AVC deploys three transforms depending on the type of residual data to be coded: a Hadamard transform [76] for the 4×4 array of luma DC coefficients in intra macroblocks predicted in 16×16 mode, a Hadamard transform for the 2×2 array of chroma DC coefficients and a DCT-based transform for all other 4×4 blocks in the residual data.

2.2.1 Transform and Quantization for Residual Pixels

In H.264/AVC, if $\boldsymbol{X}$ is an 4×4 pixel block and $\boldsymbol{A}$ defines the transform matrix, a 4×4 discrete cosine transform $\boldsymbol{Y}$ can be calculated by

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}\boldsymbol{A}^{\boldsymbol{T}} = \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{bmatrix} \boldsymbol{X} \begin{bmatrix} a & b & a & c \\ a & c & -a & -b \\ a & -c & -a & b \\ a & -b & a & -c \end{bmatrix} \tag{2.8}$$

where $a = \frac{1}{2}$, $b = \sqrt{\frac{1}{2}}\cos(\pi/8)$ and $c = \sqrt{\frac{1}{2}}\cos(3\pi/8)$.

In H.264/AVC, the transform matrix $\boldsymbol{A}$ is decomposed into a new transform matrix $\boldsymbol{C}$ and a new scaling matrix $\boldsymbol{E}$ as

$$\boldsymbol{Y} = \left(\boldsymbol{C}\boldsymbol{X}\boldsymbol{C}^{\boldsymbol{T}}\right) \otimes \boldsymbol{E} = \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \boldsymbol{X} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \right) \otimes \begin{bmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{bmatrix} \tag{2.9}$$

where $\otimes$ indicates that each element of $(\boldsymbol{C}\boldsymbol{X}\boldsymbol{C}^{\boldsymbol{T}})$ is multiplied by the scaling factor in the same position in matrix $\boldsymbol{E}$. The operation for $\boldsymbol{E}$ can be integrated into the following quantization preventing increased computational payload.

It is worthwhile to point out that this transform deployed in H.264/AVC is only a very close approximation of the 4×4 DCT. That is, the result of the new transform is not identical to the 4×4 DCT.

The inverse transform in H.264/AVC is defined as

TABLE 2.4 MULTIPLICATION FACTOR MF

Q_p	Positions (0,0), (2,0), (0,2) or (2,2)	Positions (1,1), (1,3), (3,1) or (3,3)	Other positions
0	13107	5243	8066
1	11916	4660	7490
2	10082	4194	6554
3	9362	3647	5825
4	8192	3355	5243
5	7282	2893	4559

$$\boldsymbol{X} = \boldsymbol{C^T}(\boldsymbol{Y}\otimes\boldsymbol{E})\boldsymbol{C} = \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1/2 & -1 & -1 \\ 1 & -1/2 & -1 & 1 \\ 1 & -1 & 1 & -1/2 \end{bmatrix} \left(\boldsymbol{Y}\otimes \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix} \right) \otimes \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \frac{1}{2} & -\frac{1}{2} & -1 \\ 1 & -1 & -1 & 1 \\ \frac{1}{2} & -1 & 1 & -\frac{1}{2} \end{bmatrix} \tag{2.10}$$

H.264/AVC assumes a scalar quantization. A basic forward quantization operation is

$$Z_{ij} = round(\frac{Y_{ij}}{Qstep}) \tag{2.11}$$

where Y_{ij} is the transform coefficient, $Qstep$ is the quantization step size and Z_{ij} is the quantized coefficient. A total 52 values of $Qstep$ are supported by the standard, indexed by a quantization parameter, Q_p, as shown in Table 2.3.

As mentioned above, the scaling matrix E is incorporated into the quantization. In integer arithmetic, this can be implemented as follows

$$|Z_{ij}| = (|W_{ij}|MF + f) \gg qbits \quad \text{and} \quad sign(Z_{ij}) = sign(W_{ij}) \tag{2.12}$$

where W_{ij} is the transform coefficient calculated by the matrix $\boldsymbol{C}$ as

$$\boldsymbol{W} = \boldsymbol{CXC^T}$$

and $\gg$ indicates a binary right-shift. In the implementation, f is defined as $2^{qbits}/3$ for intra blocks and $2^{qbits}/6$ for inter blocks. In Table 2.4, the first values of MF used in H.264/AVC encoder [37] are given. For $Q_p > 5$, the factors MF remains unchanged but the divisor 2^{qbits} increases by a factor of 2 for each increment of 6 in Q_p. For example, $qbits = 16$ for $6 \le Q_p \le 11$, and $qbits = 17$ for $12 \le Q_p \le 17$ and so on.

Generally, H.264/AVC can benefits significantly from the DCT based integer transform, which all operations can be carried out using integer arithmetic without loss of accuracy. In addition, since the scaling matrix is integrated in the following quantization process, no computational complexity is increased.

TABLE 2.5 SCALING FACTOR V

Q_p	Positions (0,0), (2,0), (0,2) or (2,2)	Positions (1,1), (1,3), (3,1) or (3,3)	Other positions
0	10	16	13
1	11	18	14
2	13	20	16
3	14	23	18
4	16	25	20
5	18	29	23

H.264/AVC is benefited from the small size of transform blocks in terms of compression performance, particularly when processing the QCIF and CIF video sequences or the contents containing little homogenous areas. However, this is achieved at the expense of increasing computational complexity. In addition, with the population of HD video in everyday life, coding efficiency can be significantly improved by using large size of transform blocks. This is due to the ability of large size of transform blocks to more efficiently exploit the increased spatial correlations at high resolutions. At high resolutions there are more likely to be large homogenous areas that can be efficiently represented by large blocks. In addition, quantization techniques can also be improved by considering the characteristic of human vision system.

2.2.2 Hadamard Transform and Quantization

If the macroblock is encoded in 16×16 intra mode, each 4×4 block is first transformed and then, the DC coefficients at each block is transformed again using a 4×4 Hadamard thransform as follows

$$\boldsymbol{H_D} = \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \boldsymbol{Y_D} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix} \right) / 2 \qquad (2.13)$$

where $\boldsymbol{Y_D}$ is the DC coefficients of the 16 blocks in the macroblock, and $\boldsymbol{H_D}$ is the block after Hadamard transformation. The coefficient $H_{D(ij)}$ is then quantized as

$$|Z_{D(i,j)}| = \left(|H_{D(i,j)}| MF_{(0,0)} + 2f\right) \gg (qbits + 1) \qquad (2.14)$$

$$sign\left(Z_{D(i,j)}\right) = sign(Y_{D(i,j)})$$

where $MF_{(0,0)}$ is the multiplication factor for position (0,0) in Table 2.4 and $0 \leq i,j < 4$.

At the decoding side, an inverse Hadamard transform is performed followed by inverse quantization as

$$\boldsymbol{Y_{QD}} = \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \boldsymbol{Z_D} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix}\right) \tag{2.15}$$

The inverse quantization

$$Y'_{D(i,j)} = Y_{QD(i,j)} V_{(0,0)} 2^{\left\lfloor \frac{Q_p}{6} \right\rfloor - 2} \qquad \text{for } Q_p \geq 12$$

$$Y'_{D(i,j)} = \left[Y_{QD(i,j)} V_{(0,0)} + 2^{1 - \lfloor Q_p/6 \rfloor} \right] \gg \left(2 - \left\lfloor \frac{Q_p}{6} \right\rfloor\right) \qquad \text{for } Q_p < 12 \tag{2.16}$$

where $V_{(0,0)}$ is the scaling factor V at position (0,0) in Table 2.5. Then, the retrieved DC coefficients Y'_D are inserted into their respective blocks and each block is then inverse transformed using the DCT-based transform.

Similarly, the DC coefficients of each 4 × 4 block of chroma components are grouped into a 2 × 2 block $\boldsymbol{W_D}$ and further transformed prior to quantization as

$$H_D = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} Y_D \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2.17}$$

where $\boldsymbol{Y_D}$ is the output block. Then, quantization of the 2 × 2 block $\boldsymbol{H_D}$ is performed the same way as the 4 × 4 block, but $0 \leq i,j < 2$.

At the decoder, the inverse transform is applied before scaling as

$$\boldsymbol{Y_{QD}} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \boldsymbol{Z_D} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2.18}$$

The following inverse quantization is then performed as

$$Y'_{D(i,j)} = \begin{cases} Y_{QD(i,j)} V_{(0,0)} 2^{\left\lfloor \frac{Q_p}{6} \right\rfloor - 2} & \text{for } Q_p \geq 6 \\ \left[Y_{QD(i,j)} V_{(0,0)} \right] \gg 1 & \text{for } Q_p < 6 \end{cases} \tag{2.19}$$

The inverse quantized coefficients are replaced in their respective blocks and each block is then inverse transformed using the DCT-based transform. As with the intra luma DC coefficients, this additional transform helps to de-correlate the 2 × 2 chroma DC coefficients and improves the compression performance.

2.3 OVERVIEW OF MPEG-4 VISUAL AND H.264/AVC

This chapter reviews the MPEG-4 Visual and the H.264/AVC baseline profile, particularly focusing on the transform and quantization, which are relevant for the thesis. In the simple profile and the advanced simple profile of MPEG-4 Visual, the conventional 8 × 8 DCT is deployed to convert the pixels and residual pixels into the transform domain, followed by the quantization. Compared to the previous standards such as H.263, MPEG-4 developed two quantization methods, ‘method 1’ and ‘method 2’. The default quantization is ‘method 2’ and the same as that in H.263. Alternatively, ‘method 1’ can be used in the advanced simple profile to improve the compression efficiency without perceptual distortion.

Despite being generally well received, MPEG-4 Visual has drawn some criticisms. For example, it lacks an in-loop filter to improve the visual quality, the global motion compensation is too computationally intensive, and it does not offer much compression advantage over MPEG-2. In addition, one of the key advantages of MPEG-Visual focuses on the object-based coding techniques, which allow more flexibility for encoding the regions of the interest and the backgrounds. However, this remains primarily only of interest for academic and research purposes and still requires significant work before it is adopted across the industry.

Compared to MPEG-4 Visual, H.264/AVC employs two transformation techniques: the DCT based integer transform and the Hadamard transform, which are applied to the pixels in spatial domain and the DC coefficients, respectively. The DCT based integer transform can be implemented only with additions and shifts by approximating the DCT transform matrix into an integer transform and a scaling matrix which is integrated into the following quantization. The integer transform very benefits the implementation on hardware without accuracy loss at the decoding side. In addition, smaller size of 4×4 or 2×2 transform blocks in H.264/AVC can further improve the compression efficiency compared the 8×8 2-D DCT in common use in previous standards. However, this is achieved at the expense of increasing computational complexity.

On the other hand, as an important tool in video compression transform and quantization lead a number of DCT coefficients to zeros without significant perceptual distortion, which means the calculations for those zero-quantized DCT coefficients are of little meaning. H.264/AVC tends to truncated more coefficients to zero values than previous standards by exploiting the correlations in spatial domain more efficiently. However, this is achieved at the expense of significantly increasing computational complexity. Therefore, it is of great importance and interest to predict these zero-quantized DCT coefficients and skip the calculations without additional perceptual distortion or at the cost of negligible quality degradation. In the following chapter, the existing prediction algorithms for zero-quantized DCT coefficients are reviewed followed by the author’s contributions in the field.

Chapter 3

Low Complexity High Performance Video Coding

THIS chapter discusses the proposed and other complexity reduction algorithms concerning transform and quantization in the video coding standards presented in the earlier chapter, i.e., MPEG-4 Visual and H.264/AVC. The DCT and quantization for video coding typically concentrate the energy into low frequencies and then truncate the other coefficients to zeros so that only a fraction of the quantized coefficients are practically encoded. In this way, compression is realized. However, since many DCT coefficients are truncated to zeros in the following quantization stage, the computations for these DCT coefficients become redundant. Thus, it is of importance and significance to develop such algorithms that predict the zero-quantized DCT (ZQDCT) coefficients and skip the calculations of transform and quantization for these coefficients. In this way, computational complexity is reduced and the video quality remains competitive compared to original methods without prediction.

The chapter starts with a review of competing techniques and the proposed algorithms for complexity reduction of transform and quantization in inter mode in Section 3.1. Compared to existing techniques, the proposed schemes extend the prediction for ZQDCT coefficients to 1-D transforms and lend themselves to the row and column computational structure. Therefore, computational complexity was further reduced. As a good supplement, complexity reduction for DCT and quantization in intra mode is briefly presented in Section 3.2. The proposed algorithms are developed by extending the Gaussian distribution based prediction models for inter frames. The approaches proposed in Section 3.2 can further reduce the overall complexity of video encoding and specially benefits the power constrained portable devices when processing a huge number of images. Finally, Section 3.3 summarizes this chapter.

3.1 REAL-TIME CODING FOR INTER CODING MODE

After motion estimation and motion compensation, the inter-coded blocks are only composed of residual pixels. With the discrete cosine transform and quantization, a lot of DCT coefficients are truncated to zero values. Particularly, many blocks become all zero-quantized DCT blocks, which are directly skipped without encoding. Therefore, it is of great significance to predict the ZQDCT coefficients and skip the calculations for those coefficients in order to reduce the computational complexity. In the following, existing predictions methods are firstly reviewed and then the author's contributions are summarized.

3.1.1. Sufficient Condition Based Prediction Methods

Recent years, sufficient condition based prediction methods have been developed to terminate the calculations for DCT and quantization, by comparing the theoretical maximum thresholds of the DCT coefficients depending on their frequency positions. If the thresholds are by no means higher than the quantizers, the computations for the DCT coefficients and quantization are just skipped. Since the thresholds are calculated from the theoretical maximum DCT values, no video quality degradation is observed while complexity reduction can be expected.

Sousa [68] designed a general method for eliminating the redundant computations for inter coding in video coding based on previous techniques [101][102][104]. This method exploits the distortion values computed during the motion estimation phase to detect blocks for which it is useless to go through the subsequent coding/decoding phases. This prediction method was developed based on the following theorem which was proved in [68].

In [68], given the sum of absolute differences SAD for an 8×8 luminance block, all the DCT coefficients will be quantized to zeros if the following condition holds

$$SAD < 8Q/cos^2(\pi/16) \tag{3.1}$$

where Q defines the quantization step size.

In practice, motion estimation is usually calculated based on the sum of absolute differences between the current macroblock and the reference macroblock. Therefore, the values of SAD for each DCT blocks can be pre-computed beforehand and stored in memory for later prediction. That is, this method does not introduce additional overhead computations except for a few comparisons. Experiments on H.263 coder in [102] show that about 6-40% of transformation and quantization operations for inter blocks of QCIF Claire sequence can be omitted. In addition, no video quality degradation is observed.

An early detection algorithm for the all-zero quantized DCT blocks in H.264/AVC video encoding was presented in [57]. The properties of the integer transform and quantization are first theoretically analyzed. Then, a sufficient condition under which each individual quantized DCT coefficient becomes zero is derived. Compared to the previous technique [68], the sufficient condition based thresholds are more precise by investigating the individual condition on different frequency positions of $(1,1)$, $(1,3)$, $(3,1)$ and $(3,3)$ in the 4×4 block. Simulation results show an improvement over Sousa's algorithm [68] when this condition was applied to H.264/AVC codec [37]. The improvements account to 13%-66% in terms of the predicted number of the all-zero-quantized DCT blocks and 10%-35% of the overall computational savings.

A more precise sufficient condition based prediction method for H.264/AVC encoding was developed in [82]. The improvements are achieved by specifying the thresholds into more detailed frequency components. As a result, more redundant computations regarding the transform and quantization are predicted and skipped. Since this algorithm requires more overhead operations than [68] and [57], the advantages are particularly evident at low bitrates. However, at high bitrates this method is not necessarily able to show improvements since the introduced overheads may exceed the reduced com-

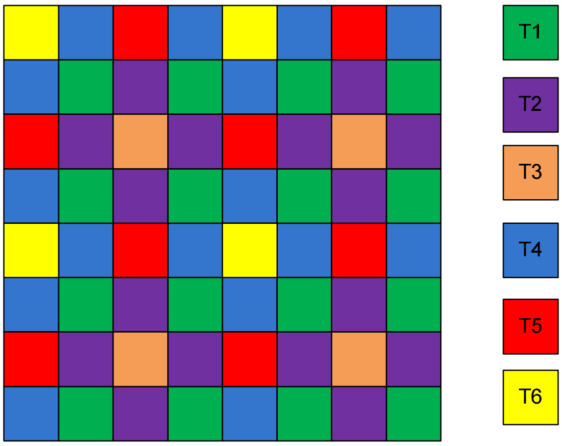


Fig.3.1 The six thresholds for the zero-quantized DCT coefficients at different frequency positions in an 8 × 8 block. $T1 = 4Q/a^2$, $T2 = \frac{4Q}{a}/(ab)$, $T3 = 4Q/b^2$, $T4 = 4\sqrt{2}Q/a$, $T5 = 4\sqrt{2}Q/b$, $T6 = 8Q$, where Q is the quantization step size at the frequency position, $a = \cos(\pi/16)$ and $b = \cos(\pi/8)$.

putations. Our experiments in [P1] show that at very low quantization the algorithm from [57] usually achieves better results than Sousa's algorithm or the method in [82].

Besides the SAD-based prediction methods, a sum of square difference (SSD)-based prediction scheme is developed in [94]. By applying both the distribution property of DCT coefficients and Parseval energy theorem, two effective methods for prediction of all-zero-quantized DCT blocks are developed: the energy conversation method and the DCT distribution method. Experiments show both methods outperform the previous techniques in [57] and [68]. For example, the energy conversation method improves the detection ratio for the all-zero blocks by 2%~23% and 4%~32% when compared to [68] and [57], respectively, without video quality degradation. Although the DCT distribution method results in lower PSNR performance, but the degradation is less than 0.02dB in the experiments. Moreover, the prediction ratio is about 4%~38% higher than [57] and 6%~47% higher than [68].

In addition, various sufficient condition based thresholds for 8 × 8 DCT and quantization are developed in [36][81][83]. Besides the prediction for all-zero blocks, a more general case is also considered in [81] for prediction of the individual ZQDCT coefficient if the block is not recognized as an all-zero block. Various thresholds are developed for the blocks not recognized as all-zero block and totally seven types for DCT and quantization are deployed as shown in Fig. 3.1. Since the 2-D DCT is usually calculated using the butterfly structure, more details about the implementation were supplemented concerning the algorithm in [83].Simulations show that this approach can save about 40% of the computations for DCT and quantization, about 10% higher than [68]. In 2009, Ji *et al* developed an early detection algorithm for 1-D transforms in partially zero-quantized DCT blocks in [36]. This method compares the SAD of the residual pixels in each row and column with the thresholds developed in [36] to determine different types of DCT calculation. This approach lends itself the row and column implementation structure. Experiments show that this approach improves on average 5% over [83] in terms of saving for DCT and quantization computations.

The key advantage of sufficient condition based prediction approaches is that the non zero-quantized DCT coefficients will never be falsely recognized as zeros after quantization. Therefore, these prediction methods do not degrade the video quality at the decoding side. However, since the thresholds are calculated based on the theoretically maximum values of DCT coefficients at different frequency positions, only a fraction of zero-quantized DCT coefficients can be properly predicted and majority of the ZQDCT coefficients are still calculated. For example, the prediction approach proposed in [81] only skips 40% of the computations for DCT and quantization on average. The prediction efficiency is particularly limited as small quantization step size. Therefore, it is of significance to exploit other prediction possibilities with higher efficiency.

3.1.2 Statistical Modeling Based Prediction

A number of experiments [3][21][45][100][105] show that the distribution of the residual pixels following motion compensation can be well modeled by a Laplacian distribution and a Gaussian distribution with a significant peak at zero. Based on this property various statistical modeling based predictions have been developed to predict the ZQDCT coefficients and reduce the redundant computations for the transform and quantization operations.

In [61], Pao *et al.* proposed a Laplacian distribution based model for the prediction of ZQDCT coefficients. Based on this model, an adaptive method with multiple thresholds is mathematically developed to reduce the computations of DCT, quantization, inverse quantization and inverse DCT in a video encoder. The 8×8 DCT is simplified into three types of computations by comparing against the thresholds: no computation, 4×4 low frequency coefficients only and all the 64 DCT coefficients. As a result, the computations of inter DCT and quantization are reduced about 32% ~ 95% at the expense of $0.08 \sim 0.23$ dB of video quality degradation in H.263 codec. The bitrates range from 20 kb/s to 56kb/s. In addition, experiments show that the entire encoding time is 8% ~25% faster than reference codec when the proposed approach was implemented on TMN8.

A Gaussian distribution based model [84] was designed to predict the ZQDCT coefficients for fast video encoding. Using the same implementation method as [61], the computations for transform and quantization operations can be further reduced by about 2% ~ 5% when the algorithm was implemented on MPEG-4 Visual based XVID codec.

Since the above methods are developed for 8×8 DCT and quantization, they cannot be directly applied to H.264/AVC where 4×4 DCT based transform is deployed. A new Gaussian distribution based model is developed in [86] for complexity reduction of 4×4 transform and quantization. In addition, the thresholds are further refined using the sufficient condition based prediction. Finally, four thresholds are applied for five types of implementations of the transform: skip, $1 \times 1, 2 \times 2, 3 \times 3$ and 4×4 transform. Computational savings are obtained for the first four types of the implementation methods. Simulation results show that these thresholds perform better than competing techniques in terms of complexity reduction, while the video quality degradation is less than 0.01 dB on average, which is negligible.

TABLE 3.1 GAUSSIAN THRESHOLD MATRIX $\beta_G(u,v)$

5.54	7.22	9.26	11.86	14.34	16.46	18.07	19.07
7.22	9.44	12.10	15.50	18.73	21.51	23.61	24.91
9.26	12.10	15.51	19.87	24.02	27.58	30.26	31.93
11.86	15.50	19.87	25.45	30.76	35.32	38.76	40.90
14.34	18.73	24.02	30.76	37.18	42.69	46.85	49.44
16.46	21.51	27.58	35.52	42.69	49.02	53.79	56.76
18.07	23.61	30.26	38.76	46.85	53.79	59.03	62.29
19.07	24.91	31.93	40.90	49.44	56.76	62.29	65.73

TABLE 3.2 LAPLACIAN THRESHOLD MATRIX $\beta_L(u,v)$

5.54	7.26	9.37	12.00	14.41	16.63	18.25	19.26
7.26	9.48	12.09	15.63	18.85	21.67	23.80	25.11
9.37	12.20	15.63	20.07	24.20	27.83	30.54	32.16
12.00	15.63	20.07	25.71	30.95	35.58	39.12	41.24
14.41	18.85	24.20	30.95	37.51	43.05	47.28	49.80
16.63	21.67	27.83	35.58	43.05	49.43	53.79	57.27
18.25	23.80	30.54	39.12	47.28	53.79	59.49	62.81
19.26	25.11	32.16	41.24	49.80	57.27	62.81	66.24

In 2007, a hybrid statistical model was developed in [87] to reduce the redundant computations for inter transform and quantization. Compared to the Gaussian distribution based prediction model proposed in [84], the threshold for the DC coefficient is further optimized by exploiting the property of sufficient condition based prediction model. In addition, the algorithm is implemented as three types: skip, only the 4×4 low frequency coefficients are computed, and calculate the transform and quantization as usual. Experimental results on XVID codec show that this hybrid model provides higher savings in complexity compared to [61] [68] [104] with similar video quality.

An 8×8 block is usually predicted as the all-zero-quantized DCT block if the sum of absolute residuals is smaller than the minimum threshold among all the 64 thresholds. In [87], the threshold for the DC coefficients is improved to eight, which is larger than the thresholds for the DCT coefficients at the position of (1,0) and (0,1) in an 8×8 block, the thresholds at (1,0) and (0,1) become the smallest. However, the approach proposed in [87] fails to explain why the threshold for the DC coefficients is still employed to predict the all-zero-quantized DCT blocks.

Since the Hadamard transform is deployed in H.264/AVC, prediction for the Hadamard transformed coefficients is also developed in [88]. The thresholds are derived based on the sum of absolute transformed difference instead of the SAD in previous motion estimation. Therefore, only a few overhead comparisons are introduced. Experiments show that this method can reduce the transform and quantization operations by 30% ~ 40% by average than the reference codec JM 9.5. The entire encoding time is reduced by 1.78% averagely.

However, it is worthwhile pointing out that Gaussian distribution based prediction model is not necessarily better than the Laplacian distribution based model according to the experiments in [P1]. Since the thresholds in the two models are calculated depending on the confidence level parameters, adjusting of the confidence level parameters in the two models could result in different thresholds. Therefore, the prediction results are changeable. With the same confidence level parameter the Gaussian distribution based model tends to generate higher thresholds for prediction of the ZQDCT coefficients, thus it has higher prediction efficiency. However, when the confidence level parameter in the Laplacian distribution based model is adjusted so that the first threshold is equivalent to the first threshold in the Gaussian distribution based model, the differences between the other thresholds in the two models are very small as shown in Table 3.1 and Table 3.2. In this case, experiments show both models achieve almost the same prediction results and the video quality degradation is comparable. In fact, it is impossible to draw a conclusion that which model is better than the other for different video sequences and quantization parameters.

3.1.3 Other Prediction methods for Transform and Quantization Operations

The quantized DCT (QDCT) is a nonlinear transform obtained by embedding the quantization stage into the DCT transformation stage. The main idea is to pre-compute and store a set of coefficients for each quantizer used in the encoder [38]. For the case of uniform quantization, one QDCT routine is designed for each possible value of the quantization step. This effectively replaces the need for computing power with little additional memory. Uniform quantization in video coding can use the same quantization step or different quantization steps for the DCT coefficients.

For two-dimensional QDCT, if $\boldsymbol{Y}$ is the DCT coefficients in matrix format calculated from the pixel block $\boldsymbol{X}$ and the 2-D transform matrix $\boldsymbol{C}$, this can be defined as

$$\boldsymbol{Y} = \boldsymbol{C}\boldsymbol{X}\boldsymbol{C}^{\boldsymbol{T}} \tag{3.2}$$

Following a constant quantization step size, Q, the quantized DCT coefficients, $\boldsymbol{Y}^{\boldsymbol{Q}}$, can be calculated as

$$\boldsymbol{Y}^{\boldsymbol{Q}} = [\boldsymbol{Y}/Q] = \left[\boldsymbol{C}^{\boldsymbol{Q}}\boldsymbol{X}\left(\boldsymbol{C}^{\boldsymbol{Q}}\right)^{\boldsymbol{T}}\right] \tag{3.3}$$

where $\boldsymbol{C}^{\boldsymbol{Q}} = \boldsymbol{C}/q$, $q = \sqrt{Q}$,and [ ] indicates the rounding operation. Then the constant quantization is merged into the DCT transform matrix $\boldsymbol{C}$.

When a constant quantization step is used, the same uniform quantization scheme is applied to all DCT coefficients. This is the case in H.263 coding and MPEG-2 inter mode. The quantization step value for each macroblock is obtained via the rate control mechanism. This choice of a constant quantization step simplifies the quantization circuitry or software implementation. A practical method for calculation of QDCT was presented in [17] to skip the division operations performed by the quantization.

To further reduce the complexity an early termination QDCT algorithm is developed in [17]. If the input data after motion compensated prediction is regarded as an all-zero block, the computations of QDCT are just skipped. In addition, a more practical order in which the column and row QDCTs are performed is presented to further optimize the prediction.

Experiments in [17] were performed on MPEG-2 and H.263 codec using the CCIR 601 (720 × 480) sequences and CIF 352 × 288 sequences at various bitrates. Experimental results show that up to 70% of 1-D transforms can be saved in both cases with a PSNR loss of less than 0.3dB. In addition, the QDCT algorithm shows a significant improvement in terms of the execution time over the traditional method which transform and quantization are separately computed.

However, QDCT in [17][38] is only applicable for uniform quantization. As 2-D DCT operation, the coefficients are usually computed as the row-column structure, it is feasible that the constant quantization step size, Q, is equally embedded into both stages, i.e., row transform and column transform. That is, the quantization operations can be merged into the transformation processing by scaling the transform matrix $\boldsymbol{C}$ scaled to $\boldsymbol{C}/q$. When a variable quantization step is used, different quantization steps are deployed for the DCT coefficients at different frequency positions. In this case, the quantization step size, Q, would vary for each coefficient and cannot be merged into the DCT transform matrix as in (3.3). Therefore, the QDCT in [17] is not applicable if non-uniform quantization is used.

In order to solve this problem, a QDCT (NQDCT) method is presented in [85], which can be applied to both uniform and non-uniform quantization. In this approach, the quantization operations are only embedded into the second stage of transform, i.e., the column transform if the 2-D DCT is implemented in the row and column order. The quantized DCT coefficients $\boldsymbol{Y^Q}$ can be calculated as

$$\boldsymbol{Y^Q} = [\boldsymbol{Y}/Q] = \left[\boldsymbol{CX}\left(\boldsymbol{C^{Q'}}\right)^{\boldsymbol{T}}\right] \tag{3.4}$$

where $\boldsymbol{C^{Q'}} = \boldsymbol{C}/\boldsymbol{Q}$ and $\boldsymbol{Q}$ is the quantization matrix composed of non-uniform quantization step sizes in DCT blocks.

Therefore, the original quantization operations can be directly merged to the transform calculation and both uniform and non-uniform quantization are applicable. Experiments in [85] were carried out on MPEG-4 Visual using ‘Method 2’ quantization and it is shown that the NDCT algorithm achieves comparable results to the QDCT method. For ‘Method 1’ quantization where the QDCT algorithm is not applicable any more, the NQDCT still reduces the encoding time of DCT and quantization routine by 58% and the entire encoding time by 17% on the average, while achieves almost the same rate-distortion performance as the separate DCT and quantization method.

A general method for detecting all-zero-quantized DCT blocks prior to DCT and quantization was proposed for H.264/AVC by Xie *et al.* in [95]. Using this model a number of computations for the all-zero-quantized DCT blocks are skipped. In addition, much less searching points are required for motion estimation. Experiments on H.264/AVC baseline profile show a saving of up to 32% of all the all-zero blocks without video quality degradation and up to 42% of detection ratio by changing the thresholds based on the DCT coefficient distribution. However, this method is only limited to predict the all-zero-quantized DCT blocks and does not reduce the computational complexity for the non all-zero-quantized DCT blocks.

Even though the decoder has less complexity than the encoder in video coding standards, the importance of low decoding complexity is equally important to or even more important than low encoding complexity. The reason is that, in many applications such as DVD players, digital TV receivers *etc*, the end-user equipment has only the decoder implemented and the decoder block is the only codec related functional block adding complexity to the system.

Many algorithms have been proposed to reduce the decoding complexity for optimized decoder implementation. Navarro *et al.* [58] proposed a fast integer IDCT to speed up the decoding by using more efficient transform structure. In 2006, an algorithm is proposed by Ugur *et al.* [77] to relieve the computational load in the decoder of H.264. The method is able to generate decoder-friendly bit stream at the encoder and reduce the decoding complexity at the expense of negligible video degradation. Recently, Lee *et al.* [47] proposed a complexity reduction model for H.264/AVC decoder. The H.264 encoder integrated with the proposed model selects a proper coding mode to minimize the distortion while satisfying the decoding complexity constraints. In addition, many efforts have been done for the optimized implementation of video decoding on various processors [51]. All the methods have lower decoding complexity than the H.264/AVC reference software. However, since they are utilizing the traditional separate inverse quantization and IDCT method, the calculations for inverse quantization are not reduced.

3.1.4. Proposed Prediction Methods for the ZQDCT Coefficients

The previous methods can reduce the complexity of inter DCT and quantization operations, particularly in terms of all-zero-quantized DCT blocks. However, since the DCT is usually implemented in a row and column structure, the efficiency of these algorithms is in practice highly lowered for non-all-zero-quantized DCT blocks. For instance, if the first 4 × 4 residual pixels are predicted as non-ZQDCT coefficients after quantization within an 8 × 8 block, only four 8-point DCT transforms can be saved according to [61][84][87] using the implementation structure in XVID codec, even though the other 48 DCT coefficients have been directly recognized as zeros.

Moreover, there is another drawback in the Laplacian and Gaussian thresholds in the prediction of ZQDCT coefficients for the non-all-zero-quantized DCT blocks. Since the thresholds are symmetric along the diagonal line from the top-left to the bottom-right, the prediction results also have a symmetric property when uniform quantization is applied for transforms in inter frames. For example, if the third DCT coefficient on the first row in a residual block is recognized as a non-zero value, the third coefficients on the first column is also predicted as a non-zero value. However, this is not always the case. Therefore, it is of great importance to develop such methods, which are able to efficiently predict the ZQDCT coefficients in the non-all-zero-quantized DCT blocks for further complexity reduction.

Even although Ji *et al* [36] extend the prediction for ZQDCT coefficients from the all zero-quantized DCT blocks to partially zero-quantized DCT blocks in accordance with the butterfly implementation structure. For non-all-zero-quantized DCT blocks, a fast transform algorithm is developed to determine the all zero-quantized DCT row or column and skip the calculations for those 1-D transforms by pruning the conventional butterfly based algorithms. This approach is developed based on the sufficient conditions at different frequency positions in the 8 × 8 DCT block. That is, each threshold is determined by the theoretical maximum value at a specific frequency position. This approach does not result in variation in terms of video quality. However, it only considers two cases for 1-D transform saving, i.e., the case of two 1-D transforms (column 0 and 4) simultaneously skipped and the case of five 1-D transforms (column 0, 4 and row 0, 4, 2 or 6) simultaneously skipped in an 8 × 8 DCT block. All the other possibilities are just neglected. Therefore, the prediction efficiency is limited.

By extending the Gaussian distribution based thresholds developed by [84] from 2-D DCT to 1-D DCT, [P1] proposes an improved prediction method to reduce the complexity of the transform and quantization, especially for those predicted as non-all-zero-quantized DCT blocks. First, a Gaussian distribution based model is employed to predict all-zero-quantized DCT blocks prior to the transform and quantization as commonly done in the literature. Subsequently, the 2-D Gaussian distribution thresholds are orthogonally decomposed into a vector, which is used to predict all zero-quantized coefficients along the row and column transforms. Then, the sufficient conditions for DCT coefficients to be quantized to zeros are developed to further reduce the complexity. Finally, a hybrid multi-threshold based model is proposed for the all-zero-quantized DCT blocks and the all-zero-quantized rows and columns. Compared to other existing algorithms, the proposed method extends prediction to 1-D DCT for the blocks recognized as non-all-zero-quantized DCT block. Furthermore, the method fits well the row and column transform structure, and thus it is more implementation friendly.

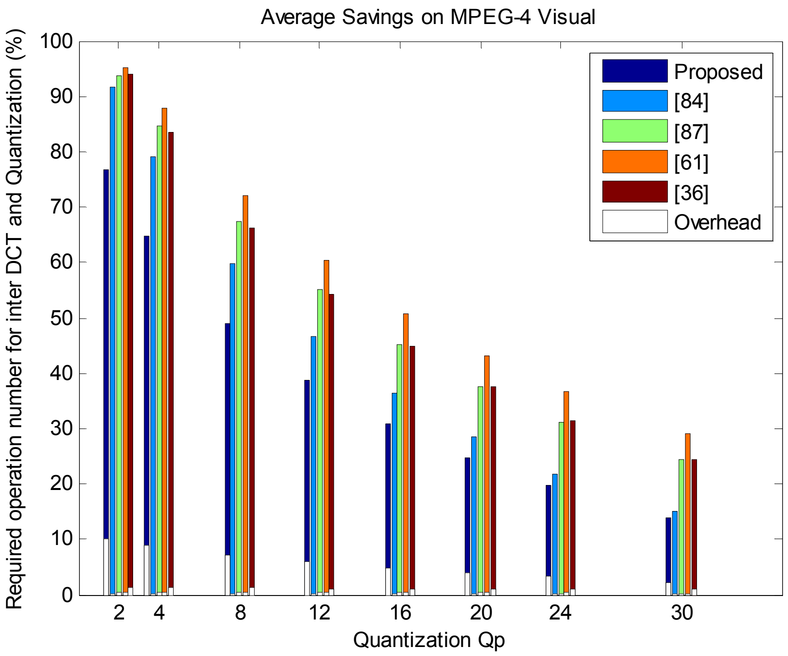
Besides, [P1] lends the evaluation criterion to the row and column implementation structure regarding this topic. For instance, previous works recognize the false recognition ratio (FRR) and the false acceptance ratio (FAR) as two important parameters to evaluate different prediction model. In the references, it is desirable to have both small FAR and FRR for an efficient prediction model and low video quality degradation. However, since the DCT coefficients are not computed individually, the prediction efficiency is highly related to the implementation structure. On the other hand, prediction errors have a negative effect the video quality. But, the quality is not only affected by the number of errors but also the magnitudes of these errors and other factors. Since the complexity analysis of individual coefficient only concerns the number of the "right" predictions and mistakes regardless of the implementation structure, it is unfair to determine the performance of the prediction models.

To evaluate the proposed model, a series of experiments were carried out on MPEG-4 Visual and H.264/AVC against competing methods, among which [36][61][84][87] were implemented on MPEG-4 Visual and [57][82][86] were implemented on H.264/AVC codec. According to the results, the proposed model achieves the best complexity reduction of DCT and quantization in terms of arithmetic operations for both MPEG-4 Visual and H.264/AVC. Fig. 3.2 summarizes the main results performed using MPEG-4 Visual and H.264/AVC.

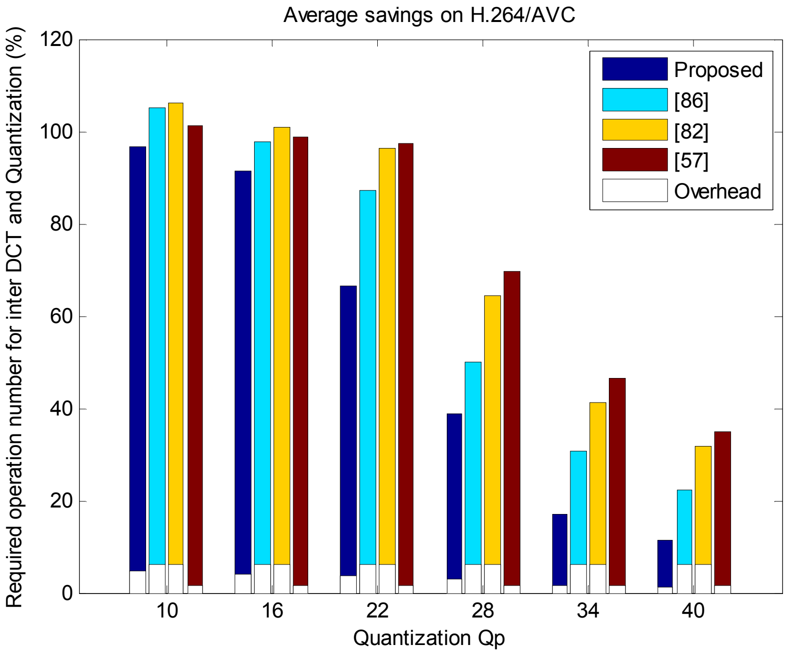
Unlike separate comparisons of PSNR and bitrates in the references, the rate-distortion (R-D) performance is studied in the experiments, thus new conclusions are drawn. According to the results, all the test models achieve comparable R-D performance to the original codec. In addition, based on the results in the magnified local areas it is hard to tell which model performs best in terms of video quality and each model is not necessarily better or worse than the others, which is different from the conclusion in the references stated as "negligible video quality degradation". The R-D performance comparisons for Crew sequence performed on MPEG-4 Visual are illustrated in Fig. 3.3 as an example.

Using statistical modeling based prediction methods, a few non ZQDCT coefficients are falsely predicted as zero values, which results in slight PSNR degradation. However, these falsely predicted zeros could decrease the overall bitrate. The separate comparisons of PSNR and bitrates among different test models are presented in Table 3.3, which summarize the results performed on MPEG-4 Visual for different video sequences. According to experiments, although the PSNR is slightly lower than the original codec, the overall bitrates are also decreased. Therefore, the overall R-D performance is still comparable to the original video codec.

[P2] describes an efficient method to reduce the decoding complexity. The proposed technique merges the inverse quantization and IDCT into a single procedure. This method is similarly a reverse procedure in [85]. The inverse quantization step sizes are embedded into the second stage of the 1-D transform by modifying the inverse transform matrix. Therefore, the computations related to inverse quantization are skipped. Particularly, both uniform quantization and non-uniform quantization are applicable. The proposed method does not change the standard bitstream syntax and can be directly applied to the standard video decoder and other DCT based video decoder.



(a)



(b)

Fig.3.2 Average complexity reduction in terms of operation numbers for CITY, CREW, HARBOUR, SOCCOER on (a) MPEG-4 Visual and (b) H.264/AVC. Each bar is composed of two parts: the "WHITE" area means the overhead operations and the upper part only considers the DCT and quantization.

Overall, the proposed model can reduce the required computations of IDCT and inverse quantization and speed up the decoding process. Compared to the reference encoder where inverse transform and quantization are separately processed, the proposed method is able to reduce the multiplications for inverse quantization as shown in Fig. 3.4. Moreover, the experiments show that the proposed method does not cause any video quality degradation compared to the original MPEG-4 Visual based XVID codec.

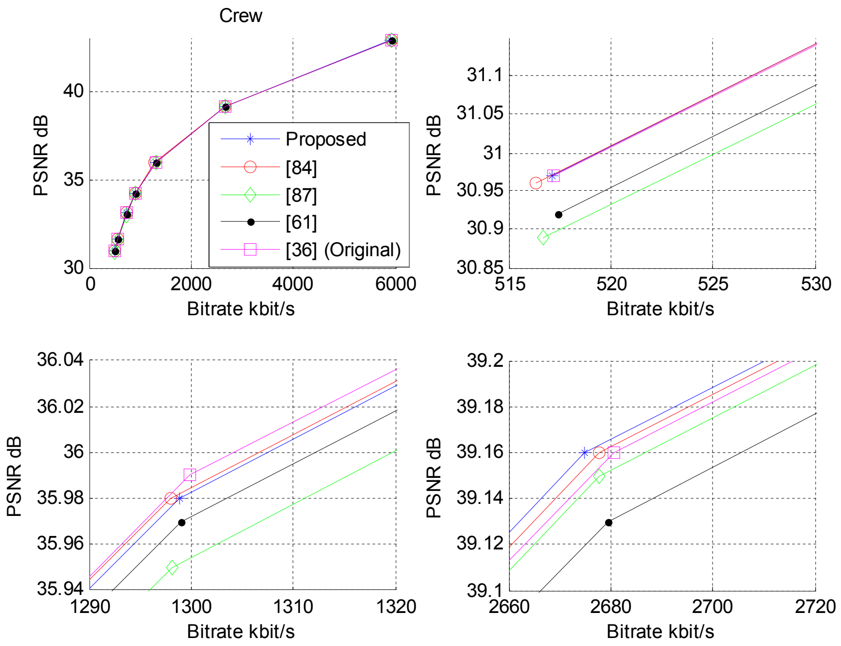


Fig. 3.3 Rate-Distortion performance regarding luminance component of CITY on MPEG-4 Visual. Each sequence consists of an overall comparison and three local comparisons. Please note that because the methodology in [36] does not change the video quality, it shares the same curve with the reference codec, i.e., Original.

TABLE 3.3 COMPARISONS OF AVERAGE PSNR (dB) AND BITRATE (Δ*R* %)

Image	***Q***	**ΔPSNR (dB)**				**Δ*R* (%)**			
		[P1]	**[84]**	**[87]**	**[61]**	**[P1]**	**[84]**	**[87]**	**[61]**
City	4	-0.05	-0.06	-0.05	-0.07	-0.21	-0.31	-0.30	-0.09
	8	-0.06	-0.08	-0.09	-0.06	-0.04	-1.06	-1.06	-0.51
	16	-0.03	-0.14	-0.11	-0.09	-0.17	-2.46	-2.39	-1.19
	30	-0.04	-0.12	-0.06	-0.09	-0.10	-0.85	-1.08	-0.63
Crew	4	-0.07	-0.07	-0.06	-0.05	-0.41	-0.02	-0.02	-0.02
	8	-0.06	-0.04	-0.08	-0.08	-0.09	-0.05	-0.14	-0.07
	16	-0.08	-0.05	-0.14	-0.11	-0.02	-0.06	-0.13	-0.03
	30	-0.07	-0.09	-0.13	-0.09	-0.11	-0.07	-0.09	-0.04
Harbour	4	-0.07	-0.02	-0.03	-0.04	-0.53	-0.07	-0.05	-0.02
	8	-0.05	-0.09	-0.06	-0.07	-0.06	-0.19	-0.59	-0.09
	16	-0.05	-0.08	-0.11	-0.09	-0.10	-0.72	-0.71	-0.41
	30	-0.10	-0.11	-0.13	-0.15	-0.05	-0.61	-0.65	-0.38
soccer	4	-0.08	-0.09	-0.06	-0.06	-0.39	-0.10	-0.09	-0.05
	8	-0.06	-0.08	-0.06	-0.03	-0.15	-0.17	-0.35	-0.09
	16	-0.09	-0.11	-0.09	-0.01	-0.13	-0.57	-0.56	-0.21
	30	-0.11	-0.15	-0.08	-0.08	-0.16	-0.31	-0.24	-0.14
Average		**-0.07**	**-0.09**	**-0.08**	**-0.07**	**-0.17**	**-0.48**	**-0.53**	**-0.25**

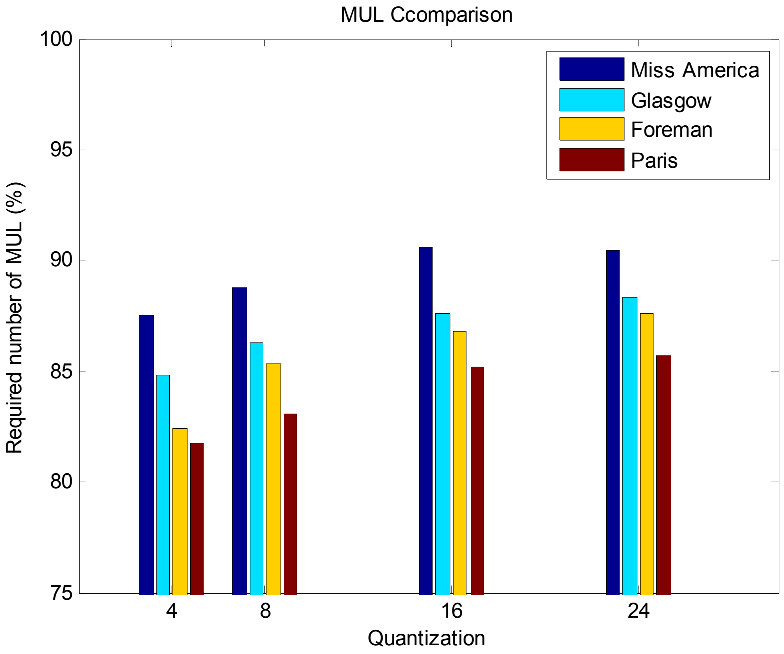


Fig.3.4 Relative multiplicative complexity of the proposed IQIDCT compared to the XVID decoder for Miss America, Glasgow, Foreman and Paris [P6].

3.2 REAL-TIME VIDEO CODING FOR INTRA MODE

3.2.1 Complexity Reduction for DCT and Quantization in Intra Mode

The research on complexity reduction of intra DCT has not been intensive, since this is only a minor issue in video coding. However, computational reduction on intra transform and quantization can still benefit the encoding process. Particularly, it is of great importance when a number of images have to be processed by portable devices with power constraints. A zero-valued prediction technique for fast DCT computation was developed in [59] based on the observations made for the DCT operation and quantization. The DCT in video coding has two important properties: (a) the high frequency components at the DCT output are concentrated nearby zero values, and (b) the frequency of the data appearance nearby these zero values becomes high as DCT progresses from a low frequency component to a high frequency component. That is, the quantized DCT coefficients at high frequency positions have higher probability to become zeros than those coefficients at low frequency positions. Before computing the DCT coefficient at a specific position in a given block, the values of N most recently computed elements for this block are checked. If all of the coefficients are zeros, the remaining DCT coefficients are directly predicted as zeros without transform and quantization. The developed method tries to incorporate effects of quantization into DCT by predicting the DCT coefficients which become zeros after quantization and omitting computation of these coefficients. The approach is controlled by N, i.e., the number of consecutive zeros, which evidently affects the total number of arithmetic operations and the image quality. With a small number N, some computations can be omitted while the image quality will be deteriorated. When N is large, the image quality remains almost the same compared to the original codec, the computational savings become very insignificant. In addition, four modes are developed for implementation of this zero-valued prediction for fast DCT computation.

In the experiments in [59], different values of N are tested against the original codec. For $N = 2$, up to 44% of computations for transform and quantization are saved for low-motion pictures and up to 37% savings are achieved for other pictures, e.g., Foreman. However, the PSNR is seriously degraded. For $N = 20$, the PSNR of the developed method is worse than that of the conventional DCT by 0.20 – 1.07dB, while the computational complexity is only about 7.86 - 17.8% of the original transform and quantization. Comparisons among the implementation modes are also carried out in [58] and the ALL mode is proven to have the best impact on PSNR degradation than other modes with the same computational savings.

3.2.2. Proposed Prediction Methods for Intra Mode

Sufficient condition based prediction methods have been investigated by many researchers, focusing on the computational complexity for inter DCT and quantization. Since the residual pixels in the inter frames are very close to zero with a good prediction process, the sum of the absolute value of the residuals in each DCT block can be very small. If the sum of the absolute residual pixels is smaller than the thresholds calculated from the 2-D DCT transform matrix and the quantization, computations for certain coefficients can be skipped. Thus, complexity reduction is realized. However, as intra frame contains original pixels, the sum of absolute pixel values can be very large. In this case, it cannot be directly applied the intra frame for prediction of DCT and quantization. [P3] proposed a modified sufficient condition based prediction method for the intra ZQDCT coefficients. In each intra block, the pixels at the same row or column in each block at the input of the transform are decomposed into a mean value and eight residual pixels. Then, the first DCT coefficient is just calculated from this mean value and the remaining coefficients are predicted in a similar way as the sufficient condition based prediction methods for inter mode. Finally, computations for these DCT coefficients can be partially skipped if they are predicted as zeros. In order to improve the prediction efficiency, the quantization is embedded into the first stage of the 2-D DCT, i.e., the row transform in this paper. It is actually a trade-off between the prediction efficiency and the video quality degradation. Experiments on JPEG codec show that the computations for 2-D DCT and quantization are decreased by 10 - 50% compared to the original codec depending on the quantization parameters and the image textures. Moreover, no video quality degradation was observed.

In [P4], we extend Pao's [61] and Wang's [84] results to intra DCT and quantization in video coding, aiming to simplify the encoding complexity and achieve better real-time performance with minimal video quality degradation. The pixel block at the input of DCT is decomposed into some mean values and a residual block. Although the proposed model is implemented based on the 8×8 DCT, it can be directly applied to other DCT-based image/video coding standards.

This approach was implemented on MPEG-4 Visual based XVID codec and the codec based on [87]. On average, the computational complexity for the intra transform and quantization is reduced by about 13%~42%. The overall complexity of DCT and quantization including both intra mode and inter mode is reduced by 1.82 - 5.63% on average. In addition, the overall required number of multiplications and additions including the overhead are compared with the original XVID encoder for the calculations of intra DCT and quantization as shown in Fig. 3.5. The comparisons summarize the results performed on different video sequences. Since the number of comparisons is very small in the experiments, they are included into the addition operations. Although the number of addition operations remains high, the required multiplication operations are reduced approximately by 53 - 90%. Therefore, the overall processing time for intra DCT and quantization is reduced compared to the reference encoder [87] and the original XVID encoder. In addition, this method is beneficial as the number of multiplications is reduced.

Experiments show that the falsely classified non-zero coefficients are usually the high frequency coefficients, thus they do not result in obvious PSNR degradation. In addition, the bitrates are slightly deceased as more non-zero valued coefficients are predicted as zeros. Based on the results in Table 3.4 and Fig. 3.6, although the proposed hybrid model has a slightly higher PSNR deterioration than the reference encoder, the overall R-D performance remains competitive and the degradation is tolerable.

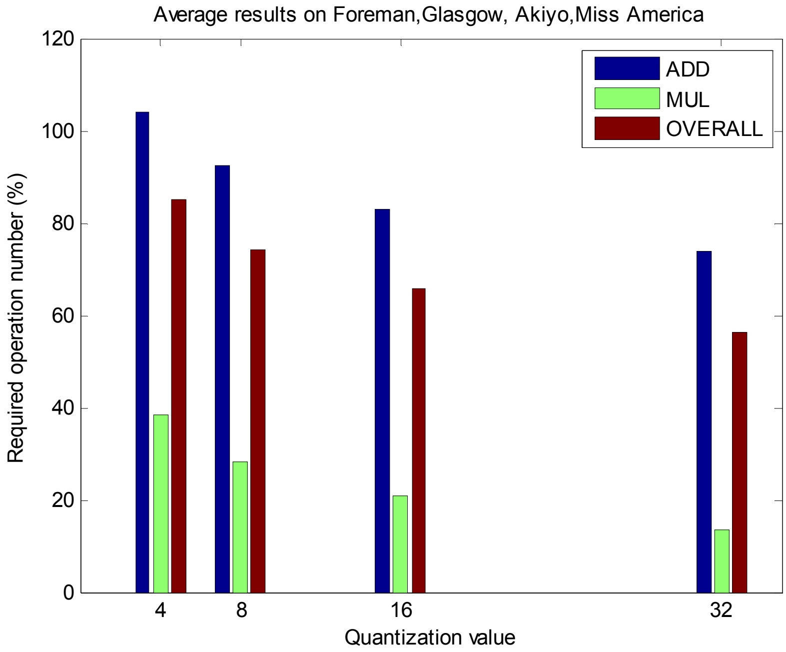
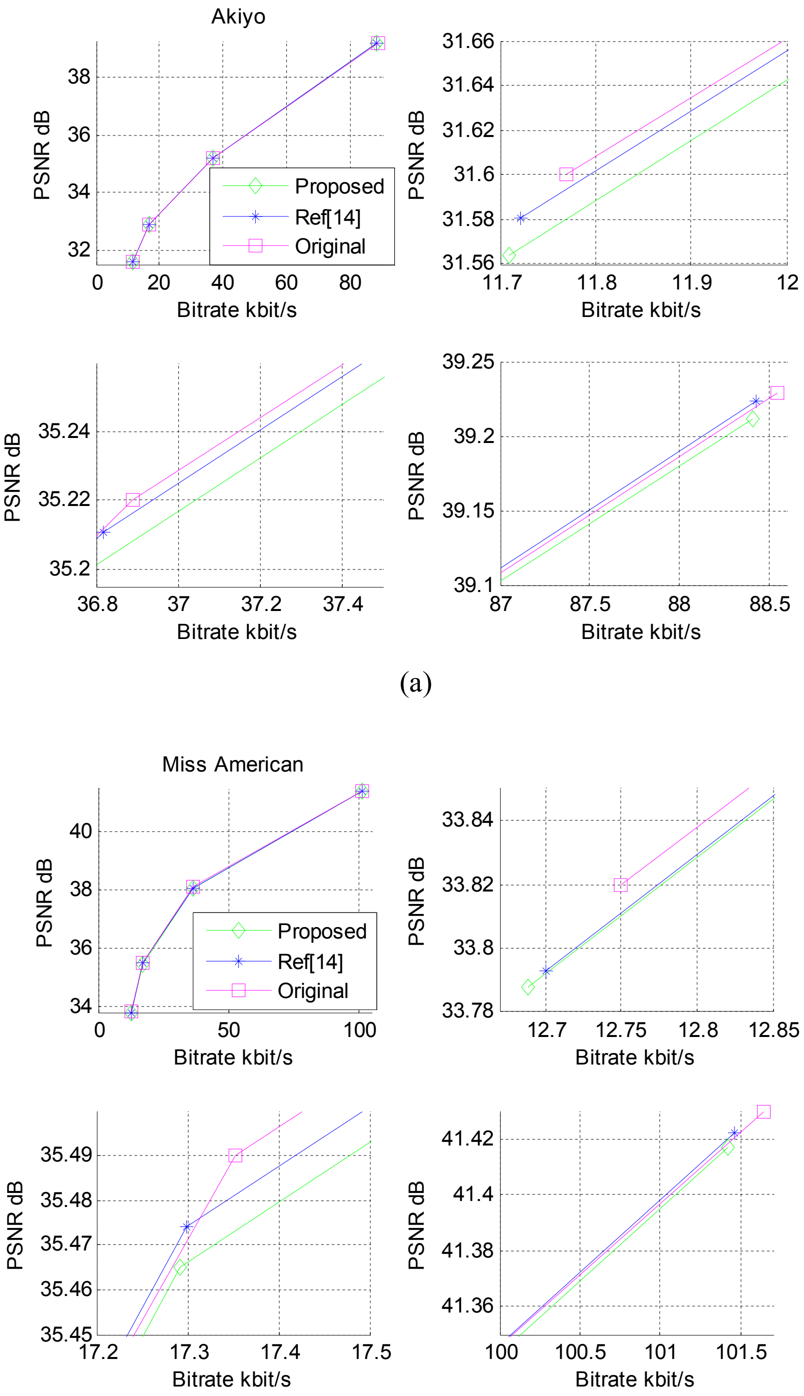


Fig. 3.5 Complexity reduction for intra DCT and quantization including the overhead computations by summarizing the results on different video sequences.

TABLE 3.4 COMPARISON OF PSNR (dB) AND BITRATE (Δ*R* %)

Image	***Q***	**ΔPSNR (dB)**		**Δ*R* (%)**	
		[P4]	**[87]**	**[P4]**	**[87]**
Foreman	4	-0.027	-0.015	-0.10	-0.09
	8	-0.021	-0.017	-0.17	-0.14
	16	-0.016	-0.013	-0.33	-0.27
	32	-0.020	-0.012	-0.50	-0.42
Glasgow	4	-0.019	-0.011	-0.08	-0.07
	8	-0.022	-0.010	-0.21	-0.16
	16	-0.025	-0.013	-0.19	-0.17
	32	-0.022	-0.009	-0.27	-0.22



(a)

(b)

Fig.3.6 Rate-Distortion performance regarding luminance component on XVID codec, (a) Akiyo and (b) Miss American. Each sequence consists of an overall comparison and three local comparisons. Note: Ref[14] refers to [87] in the thesis. Please note that “Original” means the results obtained from the reference codec.

Chapter 4

3-D DCT Video Coding

THE increasing demand for portable digital video applications as cellular video phones or fully digital video cameras is pushing the need for highly integrated real-time compression and decompression solutions. For these types of applications, efficient implementation and low power consumption are the most critical issues.

Most of today's established video coding standards such as MPEG-4 and H.264/AVC rely on the so called motion-estimation/compensation approach to exploit the correlations among inter frames. This is a highly complex process, which requires a large number of operations per pixel and is therefore less appropriate for such an implementation as real-time compression in a portable recording or communication device. Moreover, the algorithm loses much of its efficiency in scenes of morphological character or with extremely fast moving objects.

An alternative approach to prediction-based exploitation of correlations among inter frames is to use a transform-based approach for the encoding of subsequent frames. The goal hereby is to find an appropriate transform that has the desired energy compaction property and therefore allows an efficient subsequent entropy encoding. In practice, the three-dimensional (3-D) DCT video coding [55][90] is proven to be a good solution.

Compared to the motion-estimation/compensation based video coding, the 3-D DCT video coding has three major advantages [8], which are key issues for the realization of mobile video compression systems:

- No motion estimation is required, greatly reducing the number of encoding and decoding operations per pixel;
- Encoder and decoder are symmetric with almost identical structure and complexity, which facilitates their joint implementation;
- The complexity of the implementation is independent of the compression ratio.

So far, a lot of research work has been carried out on 3-D DCT video coding. Along with its advantage in computational complexity, the compression efficiency is also significantly improved. Recent results [6] show that for a given PSNR, the 3-D DCT video coding requires about 23% lower bit rates than MPEG-2 but 14% higher bit rates than MPEG-4 Visual; for a given bit rate, the 3-D DCT video coding outperforms MPEG-2 by 1.83dB while it yields 0.71dB below MPEG-4 Visual.

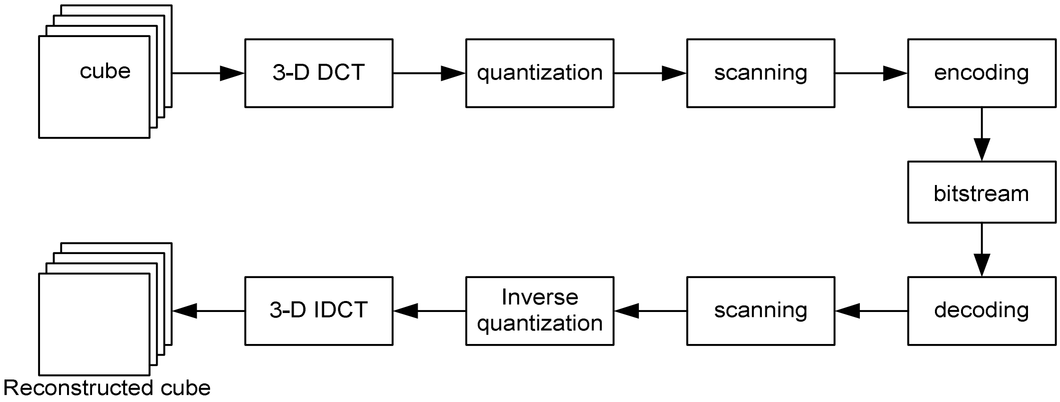


Fig.4.1. Block diagram of typical 3-D DCT based video codec.

In the sequel, the basic structure of 3-D DCT video coding is first described in Section 4.1. Then, the variable size 3-D DCT video coding approaches are briefly reviewed in Section 4.2. Finally, Section 4.3 summarizes the contributions of the thesis in this topic.

4.1 STRUCTURE OF 3-D DCT VIDEO CODING

The 3-D DCT video coding takes a full motion digital video stream and divides it into groups of N frames. Each group of N frames is considered as a three-dimensional image, including two spatial coordinates and one temporal coordinate. Each frame in the group is divided into $N \times N$ blocks, forming a set of $N \times N \times N$ cubes. In practice, the size of cube is usually selected as $8 \times 8 \times 8$, i.e., $N = 8$. Each cube is then independently encoded using the 3-D DCT video coding: 3-D DCT, quantization, reorder, and entropy encoder. A basic diagram of the 3-D DCT video coding is shown in Fig.4.1.

The original unsigned pixels, typically in the range of [0,255], are first shifted to signed integers ranging between -128 and 127. Then each $8 \times 8 \times 8$ cube of the 512 pixels is transformed into the frequency domain using the forward 3-D DCT.

$$F(u,v,w) = C(u,8)C(v,8)C(w,8)\sum_{x=0}^{7}\sum_{y=0}^{7}\sum_{z=0}^{7} f(x,y,z) \frac{\cos(2x+1)u\pi}{16}\frac{\cos(2y+1)v\pi}{16}\frac{\cos(2z+1)w\pi}{16} \tag{4.1}$$

where
$$C(k,8) = \begin{cases} \sqrt{1/8} & k = 0 \\ \sqrt{2/8} & otherwise \end{cases}$$

The transformed 512-point discrete signal is a function of three dimensions containing both spatial and temporal information. Most of the energy in the cube is concentrated in a few low-frequency coef-

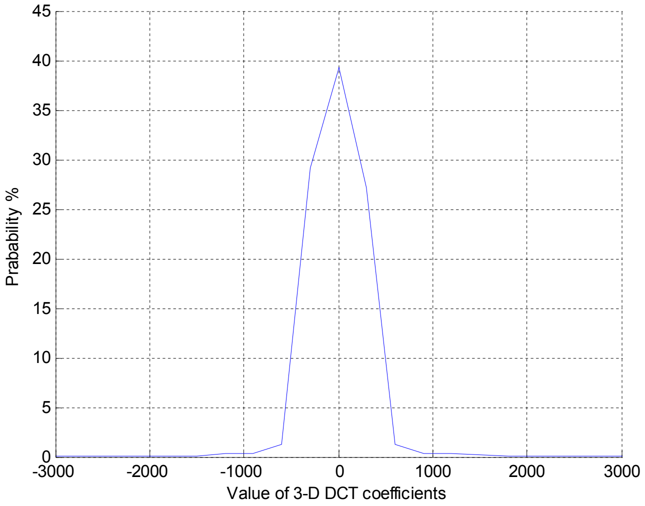


Fig.4.2 Probability density function of 3-D DCT coefficients

ficients, while the majority of the high frequency coefficients have zero or near-zero values as shown in Fig,4.2 , which summarize the statistical results performed on many video sequences.

In the next step, the 512 DCT coefficients are quantized as

$$F_q(u,v,w) = \left\lfloor \frac{F(u,v,w)}{Q(u,v,w)} \right\rfloor \tag{4.2}$$

where $F(u,v,w)$ is the transform coefficient before quantization, $Q(u,v,w)$ is the quantization value, and $F_q(u,v,w)$ is the quantized 3-D DCT coefficient. $\lfloor . \rfloor$ indicates the floor operation.

A quantization technique is developed in [48] after analyzing the distributions of the 3-D DCT coefficients, where an exponential function is used to determine the appropriate quantization volume. The quantization values are generated as

$$Q(u,v,w) = \begin{cases} A_i\left(1 - \frac{e^{-\beta_i f(u,v,w)}}{e^{-\beta_i}}\right) + 1 & for\ f(u,v,w) \leq C \\ A_0(1 - e^{-\beta_o f(u,v,w)}) & for\ f(u,v,w) > C \end{cases} \tag{4.3}$$

and

$$f(u,v,w) = (u+1)(v+1)(w+1) \qquad \text{for } 0 \leq u,v,w < 8$$

where A_0, A_i, β_o, β_i and C are empirically determined fixed parameters, and (u,v,w) indicate the coordinates in the $8 \times 8 \times 8$ cube.

Another important issue is how to order the quantized coefficients for entropy encoding as a last step in the compression. There are many ways to order the coefficients. The work in [9] suggests the use of an isoplane, $u + v + w = K$, as the scan order for quantized 3-D DCT coefficients, where K is a constant value. This method is a direct extension of the 2-D zig-zag which is based on $u + v = K$. This technique is currently the most popular scanning method in 3-D DCT video coding.

Alternatively, a procedure to determine the scan order by examining the quantized DCT coefficients is developed in [48]. It reveals that the dominant coefficients in a cube tend to lie along the major axes and, thus, the zig-zag scan order in [9] fails to effectively account for the distribution of such coefficients. In [48], the quantized 3-D DCT coefficients $F_q(u,v,w)$ inside the region, i.e. $f(u,v,w) \leq C$ in (4.3), are first ordered, followed by other DCT coefficients in the region of $f(u,v,w) > C$. This approach first orders the more dominant coefficients in the selected region before those from outside the region. Thus, more of coefficients that have been quantized to zeros will be packed together, running in an overall compression ratio. As the scan order proposed in [48] is adaptively calculated with the quantization values in (4.3), additional computations are required to calculate the scan orders for each sequence. Experiments show a better performance than the traditional zig-zag method.

For the reconstruction process the above steps are just carried out in reverse order. This similarity of the encoding and decoding process is one of the interesting properties of the 3-D DCT video coding since it greatly facilitates the implementation of joint encoder and decoder.

4.2 VARIABLE SIZES OF 3-D DCT VIDEO CODING ALGORITHMS

In video coding, the application of the discrete cosine transform along the temporal axis is advantageous over the motion estimation/compensation scheme because the structure can be non-recursive, which avoids infinite propagation of transmission errors. Moreover, the 3-D DCT video coding requires much lower computational complexity than the motion estimation/compensation based video coding.

The 3-D DCT video coding is very efficient when the amount of motion is low. This is a typical case that the energy in the high frequency components is very low. Hence, the energy compaction is good and high compression efficiency can be easily reached. However, the performance of the 3-D DCT video coding can be affected by complex scenes with a large amount of motion. In this case, traditional fixed-length 3-D DCT video coding is much inferior to the motion estimation/compensation based video coding in terms of coding efficiency. Compared to the fixed-length 3-D DCT video coding, a fewer frames along the temporal axis in each cube could result in higher video quality for sequences containing high motion activities. On other hand, for those sequences with low motion activities, a longer time window in each cube achieves better compression performance and the video quality could remain the same. Therefore, variable size of 3-D DCT video coding schemes [10][22] [42][49] [67][70][71] have been developed to improve both the compression efficiency and the video quality.

A variable 3-D DCT video coding is developed in [10]. Based on the fact that the DCT performs particularly well when the block contains only similar pixels, this scheme tries to segment a sequence into variable length of cubes with identical or similar scenes within each cube. To do this, a scene detector is used to identify scene changes. And, therefore, construct a set of discontinuity planes with variable length before the transformation. The number of discontinuity planes increases when the sequence contains a large amount of motions. On the other hand, smooth or minor temporal variations of a sequence produce a few planes, which means a longer time window is selected for the 3-D DCT.

Since variable sizes of the 3-D DCT cubes contain different number of coefficients, a uniform quantization table is not applicable any more. Various quantization tables are also developed in [9] corresponding to different types of transformed cubes. The DC coefficient is truncated with a 10-bit uniform quantizer. For AC coefficients, the transform coefficients are shown to have a Gaussian distribution [4]. Therefore, the quantization tables with different temporal lengths are defined based on their coefficient variance using the least square minimization technique [66].

The decoder requires the information about the scene changes so that the inverse 3-D DCT can be performed properly. This side information is recorded with a bit sequence where the number of bits equals to the number of frames in the current time window. A bit of '1' indicates the start of each cube and this bit sequence can be further encoded using arithmetic coding technique.

Experiments show that the variable temporal length 3-D DCT approach can improve the R-D performance over the fixed length 3-D DCT coding scheme. For video sequences such as Salesman containing little motions, the improvements are particularly significant, 4-5dB on average. For those video having a large amount of motions, this approach also improves the performance about 1-2dB averagely.

However, this approach has a high risk to properly reconstruct the video data at the decoder. The side information indicating scene changes is recorded by a sequence of binary bits and not encoded together with the corresponding DCT coefficients. If the side information is lost in transmission the decoder cannot retrieve these cubes even the related 3-D DCT coefficients are properly received. Therefore, the overall video quality would be seriously degraded.

Another variable size 3-D DCT video coding scheme is developed in [22], considering both the homogeneity in a cube along both spatial and temporal directions. The cubes at the input of the 3-D DCT are divided into three types based on the level of motion activities: $16 \times 16 \times 1$, $16 \times 16 \times 8$ and $8 \times 8 \times 8$. The motion analysis is performed in each $16 \times 16 \times 8$ cube with two empirically determined thresholds. For each cube, the normalized pixel difference is calculated between the first frame and the eighth frame. If the normalized pixel difference is less than the first threshold, it is regarded as 'no motion' and only the first frame is transformed in the mode of $16 \times 16 \times 1$. At the decoding side, the other seven frames are directly duplicated. If the normalized pixel difference is larger than the first threshold but smaller than the second threshold, the cube is assumed to contain 'low motion' and it is transformed according to the mode of $16 \times 16 \times 8$. Otherwise, the cube is considered to contain high motion activities, in which case it is divided into four $8 \times 8 \times 8$ cubes before transformation. In addition, two bits are used for each $16 \times 16 \times 8$ cube as side information for the decoder.

Experiments show that this approach proposed in [22] outperforms the fixed length 3-D DCT coding scheme, especially for those video data contains little motion activities and homogenous areas in spatial directions. However, for video sequences containing slow motions, the cubes can be classified as $16 \times 16 \times 1$ type and only the first frame is encoded. That is, duplication is applied to produce the other frames in the block, which are identical to the first frame. In this case, sudden scene changes instead of gradual motions may happen to consecutive cubes in temporal dimension, which in turn degrade the subjective video quality.

The 3-D DCT was also applied to the multiview stereo image compression. An adaptive 3-D DCT scheme is designed in [67] for image sets that form multiview stereo images. This study is limited to multiview image sets containing only eight images. That is, the eight images in the image sets set up the third dimension for multiview stereo images. First, the sum of square difference between the first frame and the last frame in each cube is calculated. Secondly, if the sum of square difference is smaller than a pre-defined threshold, this cube is characterized as background. Otherwise, it belongs to the foreground. Then, this variability map is used in the quantizer unit in order to determine the proper quantization level for the 3-D DCT coefficients. It is evident that coarser quantization can be applied for background without significant degradation in the video quality. Finally, proper quantization and scan order are designed based on the calculated variability map and direction information for each cube. Compared to [10], there are only two types of transforms, therefore it only requires one bit for each cube to indicate the type of transform. However, since the selection of quantization and scan order has to be determined for each cube, the computational complexity is highly increased.

4.3 PROPOSED ALGORITHMS

In [10], the side information that indicates scene changes within a video sequences are recorded in a sequence of binary bits, and then further compressed by arithmetic coding method. Since the side information and the 3-D DCT cubes in the current time window are encoded separately, there is a great risk that if the side information and the related 3-D DCT blocks cannot be packed into the same packet. During transmission, if the packets containing the side information are lost, all the packets having the information of related 3-D DCT blocks would have to be dropped even they are properly received. Therefore, the ability for error resilience would be seriously degraded. Otherwise, if the side information and all the related 3-D DCT blocks are included into a single packet, the packet would be too large and the sizes of different packets are not identical.

To solve this problem, a resizing 3-D DCT algorithm with variable size cubes was proposed in [P5], where three modes are utilized to perform the transform in temporal direction. The mode selection is determined relying on the local motion activity within each cube. Two binary bits are required for each cube to indicate the selection among the three modes. In [P5], the two bits are encoded together with each cube. If a packet is lost during transmission, it does not affect the content in the other packets, no matter how the side information and the related 3-D DCT cubes are packed. Thus, the ability for error resilience is improved compared to the approach proposed in [10].

Based on the approach proposed in [P5], totally three transform modes are employed. In the first mode, if the adjacent 2-D DCT coefficients in a cube are almost equal in the temporal dimension, which means that the cube contains no motion or very low motion, we apply 2-D DCT only to the first block instead of the whole cube. In the second mode, if the cube contains mild motion, we resize the $8 \times 8 \times 8$ cube to a new $8 \times 8 \times 4$ cube. Finally, in the third mode, if the cube contains high motion activity, we take 3-D DCT for the two $8 \times 8 \times 4$ cubes separately.

In the proposed method, the mode decision and the resizing operation are operated in the process of temporal transform, thus the 2-D DCT can be first performed for each block.

A typical resizing algorithm in the DCT transform domain is described in Fig.4.3 as follows

- Take the $N/2$-point DCT for block $\boldsymbol{b_1}$ and $\boldsymbol{b_2}$, respectively;
- Take the $N/4$-point inverse DCT for the $N/4$-point low frequency coefficients in $\boldsymbol{B_1}$, and the same as $\boldsymbol{B_2}$;
- Combine the two $N/4$-point blocks $\boldsymbol{b_1'}$and $\boldsymbol{b_2'}$ into one block $(\boldsymbol{b_1'}, \boldsymbol{b_2'})$, then take its $N/2$-point DCT. Finally, the two $N/2$-point blocks $\boldsymbol{b}_1$and $\boldsymbol{b}_2$ are resized into a $N/2$-point DCT block $\boldsymbol{B}'$.

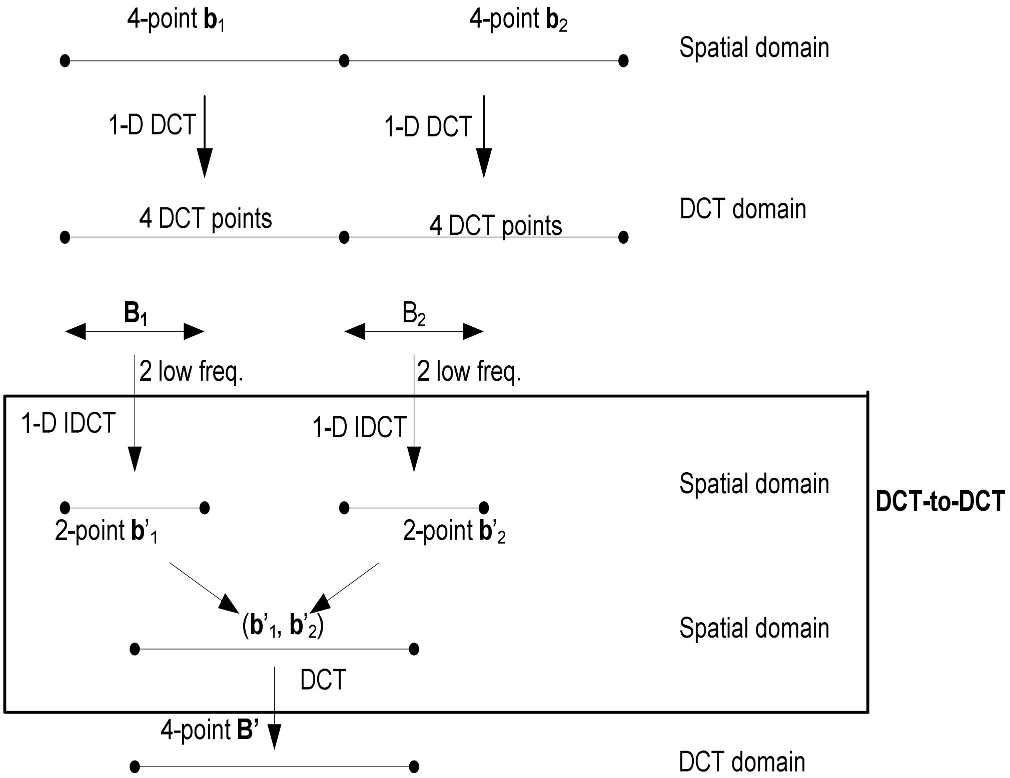


Fig.4.3. The traditional resizing algorithm at $N = 8$ and the proposed method replacing the computations inside the rectangle for the DCT-to-DCT algorithm in [P6].

The proposed 3-D DCT algorithm was tested against the approach proposed in [67] which is implemented on video coding. Compared to [67], the proposed algorithm further refines the cube size into three types. Besides the 8×8 2-D DCT and the original $8 \times 8 \times 8$ transform, an additional $8 \times 8 \times 4$ 3-D DCT is utilized for cubes only containing mild motions. To avoid the sudden scene changes due to the duplications in [67], the threshold for the 8×8 2-D DCT type is set to a relatively small value and some cubes recognized as having no motion activity in [67] are classified into the $8 \times 8 \times 4$ mode in [P5] in order to avoid direct duplication. Therefore, both better objective video quality and subjective video quality can be expected.

In the experiments, different video sequences with various motion activities were encoded and decoded. The proportions of the three modes are presented in Fig.4.4. As the threshold for mode 1 selected as the 8×8 2-D DCT is irrelevant to the quantization, a fixed proportion of mode 1 is selected. However, more cubes are resized to use the $8 \times 8 \times 4$ transform with increase of quantization. In addition, the Peak Signal to Noise Ratio (PSNR) versus compression ratio (CR) curves are plotted based on the obtained results as shown in Fig. 4.5. Experimental results show that the proposed algorithm can give better compression performance for different types of sequences. Significant improvement can be expected for sequences with low motion activity. For other sequences containing high motion activity the proposed algorithm can gain 0.5-1.2dB for luminance components and 0.3-0.8dB for

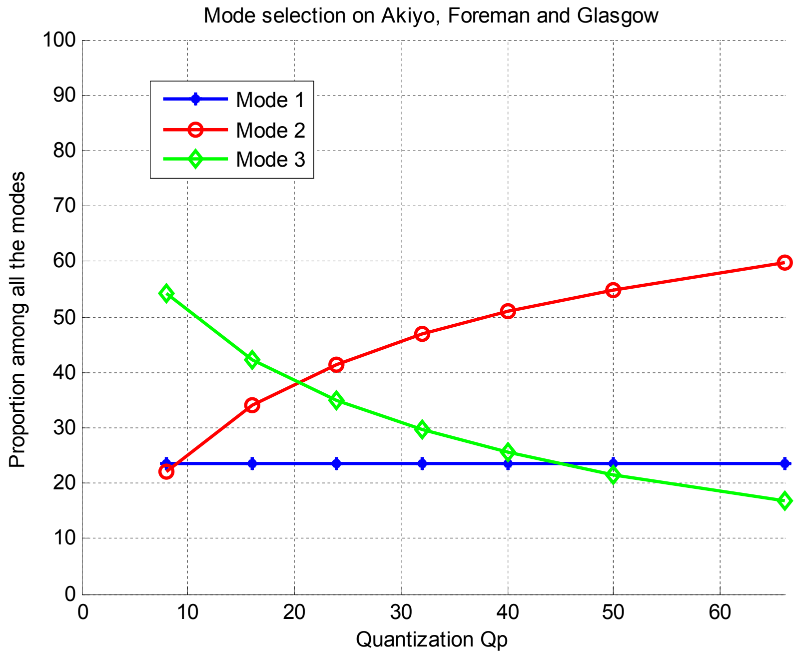


Fig. 4.4 Utilization ratio of proposed modes based on the average results on Akiyo, Foreman and Glasgow.

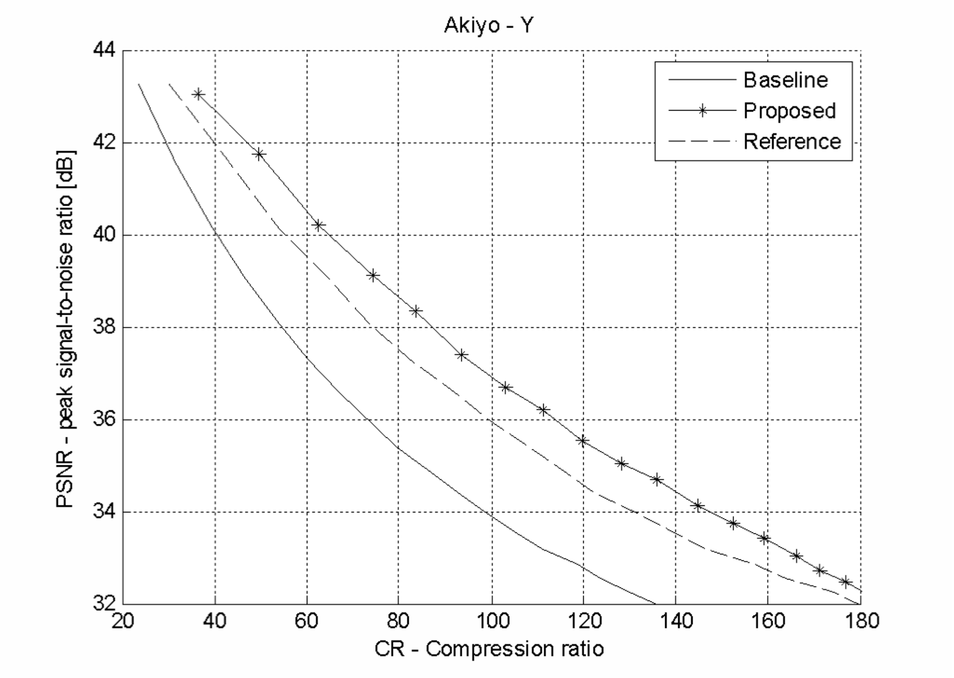
chrominance components. Moreover, the proposed algorithm can even reduce the computations for sequences with low motion activity. Take Akiyo for example, the required computations for 3-D DCT in the proposed codec are less than 84.9% of the reference codec. For Glasgow, the increase does not exceed 18.4% in the worst case.

The key advantage of the 3-D DCT video coding is the low complexity by replacing the motion estimation and compensation with an additional transform in temporal direction. Although the approach [P5] achieves better compression performance than the fixed length 3-D DCT video coding scheme and the method in [67], additional computations are introduced for selection of different transform types. Therefore, it is of great interest to reduce the overhead computations.

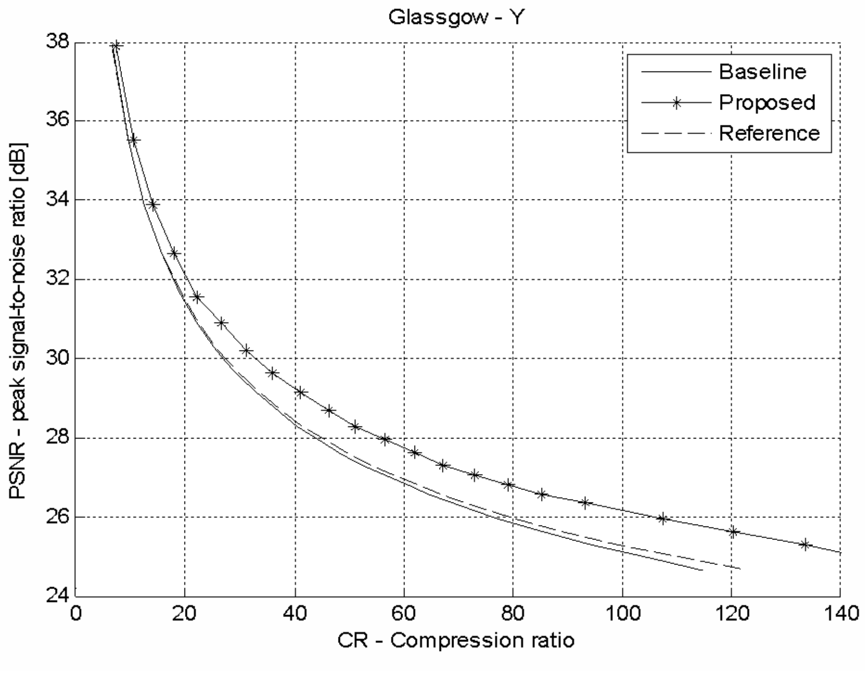
A direct DCT-to-DCT algorithm was proposed in [P6]. Previously, inverse transform has to be performed on both two DCT blocks if they are resized into one block. To alleviate the computational complexity, the proposed method in [P6] enables direct calculation that the DCT block $\boldsymbol{B}'$ is directly obtained from the two blocks $\boldsymbol{b}'_1$ and $\boldsymbol{b}'_2$ without inverse transform. Thus, the calculation for the inverse transform and other computations in the spatial domain are saved. The signal flow graph of traditional resizing method and the proposed resizing algorithm is shown in Fig.4.3.

Besides, this approach in [P6] employs non-uniform quantization method for the 3-D DCT coefficients instead of uniform quantization used in [P5]. Since human vision system is more sensitive to the low frequency information, coarsely quantization for high frequency DCT coefficients can lead to high compression performance but almost the same video quality than uniform quantization.

Therefore, using the same implementation of 3-D DCT and more efficient quantization technique as [P5], experiments show the proposed method in [P6] achieves better coding efficiency as shown in



(a)



(b)

Fig. 4.5 The luminance PSNR curves are plotted versus the compression ratio for (a) Akiyo and (b) Glasgow based on three different codec: the reference 3-D DCT video coding and the references. In this thesis, Baseline refers to the fixed length 3-D DCT video coding, Reference refers to [67].

Fig.4.6. Moreover, it requires less computations compared to [P5], particularly at low bitrates as shown in Fig.4.7.

All of the above variable length 3-D DCT algorithms employ thresholds. Although the selection of these thresholds considerably affects the algorithm performance, there is no framework to choose them effectively. In practice, the thresholds are empirically determined. These thresholds are actually trade-offs among different video sequences, but not necessarily the optimum thresholds for all the video sequences. Therefore, it is of importance to develop an algorithm for variable size 3-D DCT without empirically determining the thresholds.

An algorithm for variable 3-D DCT based video coding is proposed in [P7] to avoid empirical determination of thresholds. This proposed model iteratively utilizes 3D-DCT and discrete Haar transform (DHT) to remove the temporal redundancy and is able to find the different DCT modes for each $8 \times 8 \times 8$ cube.

The DHT is applied to the temporal axis of 3-D DCT for each cube. Given an $8 \times 8 \times 8$ cube, the eight frames are first divided into two $8 \times 8 \times 4$ cubes by applying DHT, which contain the low frequency information and high frequency information, respectively. Then, 3-D DCT is performed on the second $8 \times 8 \times 4$ cube only with high frequency information, followed by quantization. If not all the DCT coefficients are truncated to zeros. Similar transform is performed on the first cube and encode the two cubes separately, which is regarded as mode 1. Otherwise, if all the coefficients are quantized to zeros, this $8 \times 8 \times 8$ cube is recognized as containing low motion activities. Therefore, DHT is applied again to separate the first $8 \times 8 \times 4$ into two $8 \times 8 \times 2$ cubes. If not all the coefficients in the high frequency $8 \times 8 \times 2$ cube are quantized to zeros, the two $8 \times 8 \times 2$ cubes are separately encoded as mode 2. Otherwise, the low frequency $8 \times 8 \times 2$ cube will be further divided into two 8×8 blocks. Similarly, if two blocks have to be encoded, it is regarded as mode 3. Otherwise, only one block is encoded, which is mode 4.

In the proposed model in [P7], the mode decision is totally based on the truncated high frequency information. Only if all high frequency contents in the residual cube are quantized to zeros, the hybrid transform is repeated to form a smaller cube and thus produces a tighter information representation. Compared to other variable length 3-D DCT schemes, the proposed model adaptively determines the mode decision based on the quantization results on the high frequency cubes by applying DHT.

Compared to the approach in [P5], the selection of mode 4 in [P7], i.e., only the first block is transformed in a cube, is dependent on the quantization, since the high frequency coefficients have higher possibilities to be truncated to zeros at larger quantization set sizes. The utilization ration of different modes is presented in Fig.4.8.

The proposed algorithm was tested against the reference 3-D DCT video coding and two approaches proposed in [22] and [67]. Experimental results show that the proposed algorithm can give better compression performance for different types of sequences. Major improvements can be expected for those with low motion activity. In the experiments, we also evaluated the subjective visual quality. The proposed model works well for small QP (e.g. 12, 30, 52) and no temporal coding artifacts are observed. However, as QP increases, visible temporal artifacts appear due to the hard truncation in high frequencies. Thus, future work includes further improvement of quantization strategy.

Although 3-D DCT coding is much superior to the video coding standards such as H.263 in terms of computational complexity, it is still desired to further reduce the computations without visual quality degradation. As the most complex part, three-dimensional DCT transform consumes more than half of the computation in 3-D DCT encoding while a large number of transformed coefficients will be finally quantized to zeros. Thus, if we can reduce these redundant computations, the encoding process will be much faster.

In [P8], we proposed an analytical model to skip redundant DCT and quantization without visual quality degradation through a theoretical analysis of the dynamic range of DCT coefficients. Each $8 \times 8 \times 8$ cube is first decomposed into a mean value and a residual $8 \times 8 \times 8$ cube. The DC coefficient can be directly calculated from the mean value. Then, thresholds for each 1-D transform along spatial and temporal directions are developed based on the theoretically maximum DCT values and the following quantization step size. Finally, if the sum of the absolute residual pixels on each 1-D transform is smaller than the thresholds, the calculation for the 1-D DCT is skipped and the coefficients are predicted as zeros. Although the proposed model is implemented based on the baseline 3-D DCT video coding, it can be used on other fast 3-D DCT schemes such as those in [5][73]. The experimental results show that the proposed model can significantly improve the encoding efficiency without visual quality degradation.

4.4 SUMMARY AND FUTURE WORK

As a potential alternative of hybrid 2-D transform based video coding, the main advantage of 3-D DCT video coding is the considerable computational savings. Since the most complicated motion estimation and motion compensation are replaced by the temporal transformation in 3-D DCT video coding, the overall encoding complexity is significantly reduced. The target applications could be power-constrained portable devices and other applications having priority restrictions on computational complexity.

However, coding efficiency is the main bottleneck that 3-D DCT video coding has not been applied in current video coding standards and industrial products. Even the most recent 3-D DCT video coding scheme is still inferior to H.264/AVC. In addition, the temporal transformation requires both the encoder and decoder to process multiple frames simultaneously, which requires additional memory to store frames for 3-D video coding operation and therefore may increase the frame buffer size.

Moreover, so far most research work on 3-D DCT video coding focuses on improving coding efficiency and a lot of other practical considerations have been neglected. For instance, as a very important tool in video coding standards, error resilience has not been investigated in 3-D DCT video coding so far.

In fact, a lot of work remains to be done to improve the 3-D DCT video coding, which covers a wide range of topics. For example, a new rate-distortion cost scheme may significantly improve the coding efficiency of 3-D DCT video coding. The new rate-distortion cost scheme can be used to select the transform cube size and optimize the quantizer. In addition, more efficient entropy coding should also be investigated to improve 3-D DCT video coding efficiency. Although motion estimation and motion compensation are considered the most complicated part in 2-D transform video coding schemes, it is still worth to investigate the possibility for motion compensation based 3-D DCT coefficients for video sequences with complex motion. From coding efficiency point of view, a kind of combination of 3-D DCT and motion compensation may significantly improve the current 3-D DCT video coding schemes.

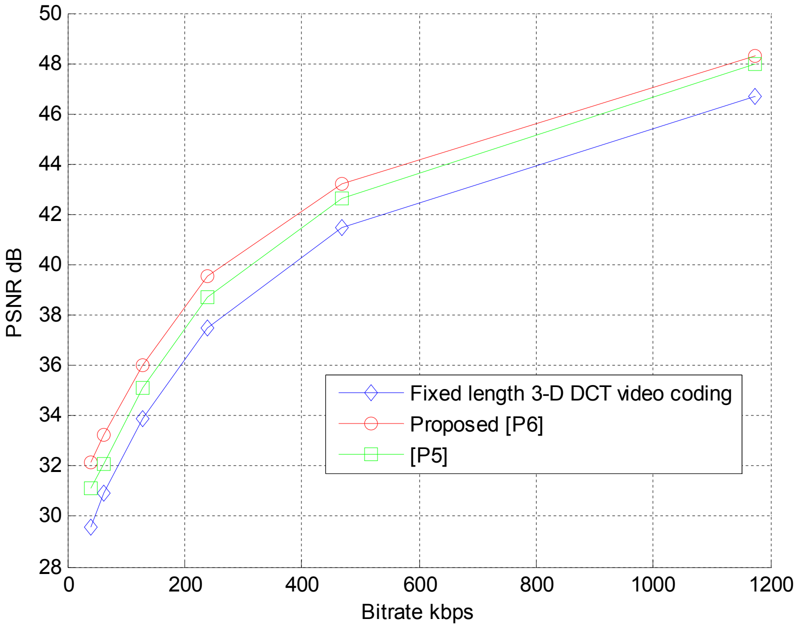


Fig. 4.6 The average PSNR versus Bitrate performed on Akiyo, Claire, Paris and Glasgow based on three different codec: the fixed length 3-D DCT video coding, the proposed codec [P6] and the reference codec [P5].

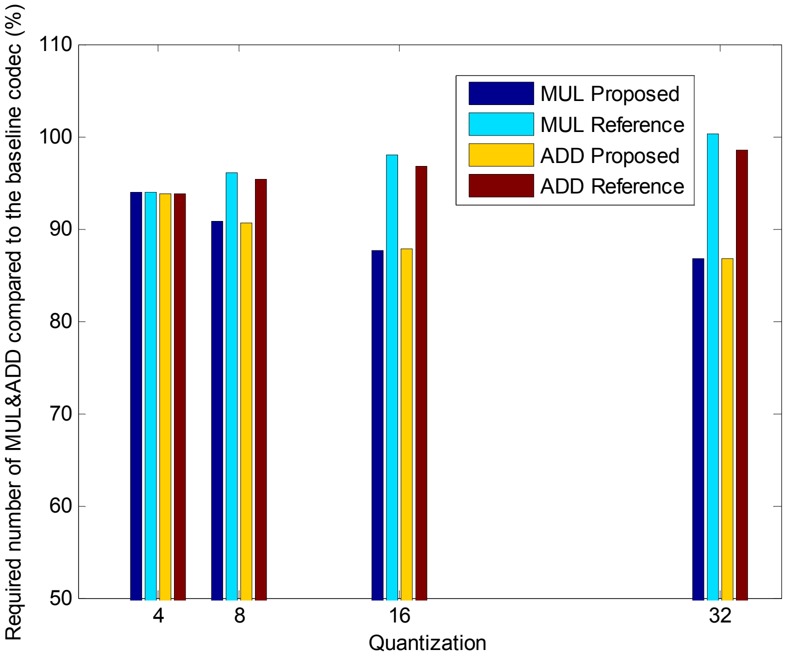


Fig. 4.7 The average MUL and ADD comparison among the proposed model in [P6], the reference [P5] and the fixed length 3-D DCT video coding for Akiyo and Glasgow.

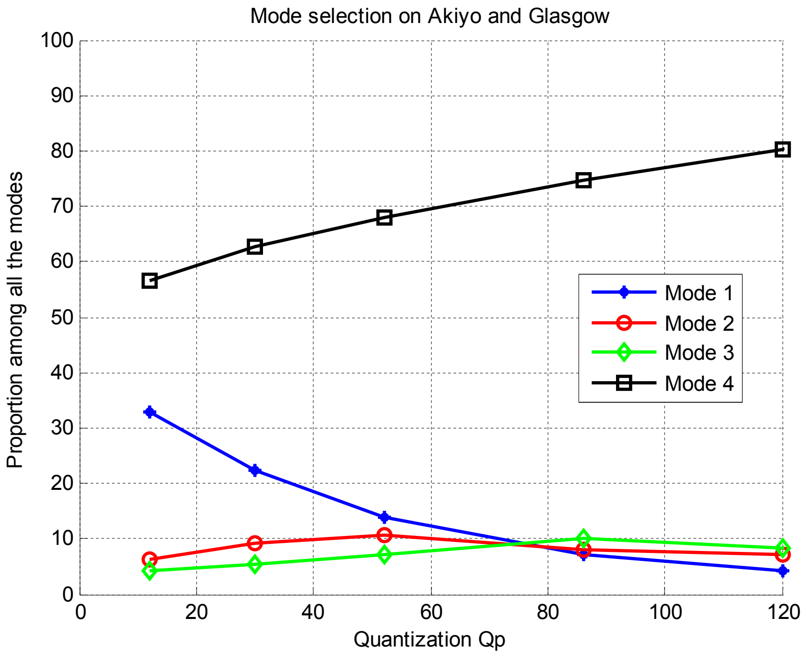


Fig. 4.8 Utilization ratio of proposed modes on Akiyo and Glasgow in [P7]. An 8 × 8 × 8 cube can be calculated with Mode 1: two 8 × 8 × 4 cubes, Mode 2: two 8 × 8 × 2 cubes, Mode 3: 8 × 8 × 1 blocks and Mode 4: one 8 × 8 × 1 block.

Chapter 5

Conclusions and Future Work

VISUAL communication via mobile devices is becoming more and more popular. However, current video coding standards are not initially designed for this type of application area. Traditionally, high compression efficiency is usually achieved at the expense of increasing computations. As the newest standard, H.264/AVC significantly outperforms the others in terms of coding efficiency; its complexity is greatly increased. The high complexity limits its application on mobile devices with low computational power and battery constraint as well as the real-time performance of the encoder. Thus, there is great significance and interest in reducing the computations for video coding. Currently, the emerging standard, i.e., HEVC, claims an aggressive goal to reduce the encoding complexity by 50% with 25% improvement in visual quality.

The specifications of most video coding standards define only the bit-stream syntax and the decoding process. The encoding process is not standardized to allow flexible implementations. The research work proposes algorithms for both the encoder and the decoder with emphasis on complexity reduction for mobile applications.

Several algorithms for real-time encoding/decoding are proposed to reduce the computational complexity for the transform and quantization operations. These techniques can be categorized into two types: complexity reduction for inter mode and for intra mode. All the proposed techniques achieve lower encoding or decoding complexity with the same or comparable rate-distortion performance to the original codec. The targeted applications could be portable devices with constraints on computing power and amount of memory.

In addition, new algorithms for 3-D DCT video coding are proposed for improving the coding efficiency and computational complexity. First, a resizing algorithm was proposed for variable size 3-D DCT depending on the local motion activities. Later on, this algorithm was further improved using direct DCT-to-DCT technique and a more efficient quantization scheme to achieve better performance in terms of both complexity and coding efficiency. All the variable temporal length 3-D DCT schemes employ thresholds for the selection of different transform types, however, the thresholds are usually empirically determined. To solve this problem, a new 3-D DCT approach was proposed to allow adaptive selection of thresholds for variable size of temporal transform. By applying the DHT, the transform mode can be adaptively determined according to the quantization step size. Therefore, the thresholds are not required any more. Finally, a new 3-D DCT video coding was proposed to reduce the encoding complexity by exploiting the maximum thresholds of 3-D DCT coefficients based on the information of quantization. For those DCT coefficients predicted smaller than the thresholds, the computations are just omitted and they are directly set to zeros. Therefore, a lot of computations are saved. In addition, since the thresholds are obtained based on the theoretically maximum DCT values at different frequencies, no video quality degradation is observed.

The proposed algorithms and implementations are successfully utilized in conjunction to other related techniques based on conventional video coding methods in different video coding scenarios. The new approaches can be used for multi-layer scalable video coding as well as the emerging video coding schemes such as stereo video coding.

This thesis has concentrated on the analysis of the transform and quantization operations in video coding, leaving other aspects, such as motion estimation/compensation, filtering, human visual system based lossy coding, outside the scope of the research work. However, since motion estimation and filtering also require intensive computations, it is also of great importance to advance the research in these fields. For 3-D DCT video coding, the future work could focus on a new rate-distortion cost based coding scheme, more efficient entropy coding techniques, reducing the required memories, and even combination of 3-D DCT video scheme and the hybrid 2-D transform based video coding standards. Moreover, the human visual system based lossy coding is expected to provide better subjective video quality to end-users of mobile devices. Thus, potential future work could provide more analysis on aspects beyond transform and quantization.

Bibliography

[1] N. K. Ahmad, S. Masud, and M. A. Maud, “Efficient block size selection in H.264 video coding standard,” *Electronics letters*, vol. 40, no. 1, pp. 19-21, Jan. 2004.

[2] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” *IEEE Trans. Comput.*, Vol. C-23, pp. 90–93, Jan. 1974.

[3] Y. Altunbasak and N. Kamaci, “An analysis of the DCT coefficient distribution with the H.264 video coder,” *IEEE International Conference on Acoustics, Speech, and Signal Processing, pp.* 177-180, Montreal, Canada, May 2004.

[4] M. Bhaskaranand and J. D. Gibson, “Distributions of 3D DCT coefficients for video,” *IEEE International Conference on Acoustics, Speech and Signal Processing,* pp.793-796, Taipei, Taiwan, April 2009.

[5] S. Boussakta and H. O. Alshibami, “Fast algorithm for the 3-D DCT-II,” *IEEE Transactions on Signal Processing*, vol.52, no.4, pp. 992- 1001, April 2004.

[6] N. Bozinovic and J. Konrad, “Motion analysis in 3D DCT domain and its application to video coding,” *Proceedings of Signal Processing: Image Communication*, vol.20, pp. 510-518, 2005.

[7] N. Brady, “MPEG-4 standardized methods for the compression of arbitrarily shaped video objects,” *IEEE Transactions on Circuits Systems for Video Technology*, vol. 9, pp. 1170–1189, Dec. 1999.

[8] A. Burg, R. Keller, J. Wassner, N. Felber, and W. Fichtner, “A 3D-DCT real-time video compression system for low complexity single chip VLSI implementation,” *IEEE International Workshop on Mobile Multimedia Communications ,* Tokyo, Japan, Oct. 2000.

[9] R. K.W. Chan and M.C. Lee, “3D-DCT quantization as a compression technique for video sequences,” *International Conference on Virtual Systems and MultiMedia*, pp.188, 1997.

[10] Y. L. Chan and W.C. Siu, “Variable temporal-length 3-d discrete cosine transform coding,” IEEE *Transactions on Image Processing*, vol 6, No. 5, pp. 758-763, 1997.

[11] X. Chen, Q. Dai, and C. Li, “A fast algorithm for computing multi-dimensional dct on certain small sizes,” *IEEE Trans. Signal Processing*, vol. 51, No. 1, Jan. 2003.

[12] S. D. Chen, A. R. Ramli, and M. R. Mukerjee, "All-zero-AC block detection using energy preservation theorem for H.263 video coding," *IEEE TENCON*, pp.425-430, 2000.

[13] W. Choi, B Jeon, and J. Jeong, “Motion estimation with modified diamond search for variable block sizes,” *IEEE Conference on Image Processing (ICIP)*, pp. 371-374, Barcelona, Spain, Sept. 2003.

[14] H. C. Chuang, C. Y. Huang, and T. Chiang, “A novel adaptive video playout control for video streaming over mobile cellular environment,” *IEEE International Symposium on Circuits and Systems (ISCAS),* pp. 3267- 3270, Kobe, Japan, May 2005.

[15] Q. Dai, X. Chen, and C. Lin, “Fast algorithm for multidimensional DCT-to-DCT computation between a block and its associated subblocks,” *IEEE Trans. Signal Processing*, Vol. 53, No. 8, Aug. 2005.

[16] S. G. Deshpande, and J. N. Hwang, “A new fast motion estimation method based on total least squares for video encoding,” *IEEE conference on Acoustic, Speech and Signal Processing (ICASSP)*, pp. 2797-2800, Seattle, USA, May 1998.

[17] A. Docef, F. Kossentini, K. Nguuyen-Phi, and I. R.Ismaeil, “The quantized DCT and its application to DCT-based video coding,” *IEEE Trans. Circuits Systems Video Technology*, vol. 11 No. 3, pp. 177–187, Mar. 2002.

[18] R. Dugad and N. Ahuja, “A fast scheme for image size change in the compressed domain,” *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11 No. 4, pp.461-474, 2001.

[19] P. Duhamel and C. Guillemot, “Polynomial transform computation of the 2-D DCT,” *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pp. 1515–1518, Albuquerque, U.S.A, Apr.1990.

[20] A. Elnaggar and H. M. Alnuweiri, “A new multidimensional recursive architecture for computing the discrete cosine transform,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 1, pp. 113–119, Feb. 2000.

[21] T. Eude, R. Grisel, H. Cherifi, and R. Debrie, “On the distribution of the DCT coefficients,” *IEEE International Conference on Acoustics, Speech, and Signal Processing,* pp. 365- 368, Apr. 1994.

[22] B. Furht, K. Gustafson, H. Huang, and O. Marques, “An adaptive three-dimensional DCT compression based on motion analysis,” *Proceedings of ACM*, pp. 765-768, 2003.

[23] M. Grigoryan, “An algorithm for calculation of the discrete cosine transform by paired transform,” *IEEE Tran. Signal Processing*, vol. 53, Issue 1, pp. 265–273, Jan. 2005.

[24] C. Huang and P. Salama, “Error concealment for shape in MPEG-4 object-based video coding,” *IEEE Transactions on Image Processing,* vol.14, no.4, pp.389-396, April 2005.

[25] A. Hutter, G. Giebel, and W. Stechele, “A coprocessor architecture implementing the MPEG-4 visual core profile for mobile multimedia applications,” *IEEE International Symposium on Circuits and Systems (ISCAS),* pp.176-179, Geneva, Swizerland, 2000.

[26] http://www.h265.net/ cited: 2011/01/16

[27] http://iphome.hhi.de/suehring/tml/ cited:2011/01/16

[28] Y. Ismail, M. A. Elgamel, and M. A. Bayoumi, “Fast variable padding motion estimation using smart zero motion prejudgment technique for pixel and frequency domains,” *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 19, Issue 5, pp. 609-626, May 2009.

[29] ISO/IEC JTC 1, “Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 2: Video” ISO/IEC 11172 (MPEG-1), Nov. 1993.

[30] ISO/IEC JTC 1, “Coding of audio-visual objects – Part 2: Visual,” ISO/IEC 14496-2 (MPEG-4 Visual), Jan. 1999.

[31] ISO/IEC, “Motion-JPEG2000 (JPEG2000 Part 3),” 15444-3, Geneva, Swizerland, 2002.

[32] ITU-T, “Video codec for audiovisual services at px64 kbits/s,” ITU-T Rec. H.261 v1: Nov. 1990, v2: Mar. 1993.

[33] ITU-T and ISO/IEC JTC 1, “Generic coding of moving pictures and associated audio information – Part 2: Video,” ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2), Nov. 1994.

[34] ITU-T, “Video coding for low bit rate communication,” ITU-T Rec. H.263 v1: Nov. 1995, v2: Jan. 1998, v3: Nov. 2000.

[35] ITU-T and ISO/IEC JTC 1, “Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC),” May 2003.

[36] X. Ji, S. Kwong, D. Zhao, H. Wang, C.-C. J. Kuo, and Q. Dai, “Early determination of zero-quantized 8 × 8 DCT coefficients,” *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 19, no.12, pp.1755-1765, Dec. 2009.

[37] JM H.264 Reference Software Version JM6.1d, http://bs.hhi.de~suehring/tml/, March, 2003

[38] N. P. Khanh, A. Docef, and F. Kossentini, “Quantized discrete cosine transform: a combination of DCT and scalar quantization," *IEEE International Conference on Acoustics, Speech, and Signal Processing,* pp. 3197-3200, Mar. 1999.

[39] J. N. Kim, S. C. Byun, and B. H. Ahn, “Fast full search motion estimation algorithm using various matching scans in video coding,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* vol. 31, Issue 4, pp. 540-548, Nov. 2001.

[40] J. N. Kim, S. C. Byun, Y. H. Kim, and B. H. Ahn, “Fast full search motion estimation algorithm using early detection of impossible candidate vectors,” *IEEE Transactions on Signal Processing,* vol. 50, Issue 9, pp. 2355-2365, Sept. 2002.

[41] K. B. Kim, Y. Jeon, and M. C. Hong, “Variable step search fast motion estimation for H.264/AVC video coder,” *IEEE Transactions on Consumer Electronics,* vol. 54, Issue 3, pp. 1281-1286, Oct. 2008.

[42] J.J. Koivusaari, and J.H. Takala, “Simplified three-dimensional discrete cosine transform based video codec,” *SPIE Multimedia on Mobile Devices*, pp. 11-21, 2005.

[43] J.J. Koivusaari, J.H. Takala, and M. Gabbouj, “Image coding using adaptive resizing in the block-DCT domain,” SPIE *Multimedia on Mobile Devices II*, pp. 1-9, 2006.

[44] W. Kou and T. Fjallbrant, “A direct computation of DCT coefficients for a signal block taken from two adjacent blocks,” *IEEE Trans. Acoustics, Speech and Signal Processing,* Vol. 39, No. 7, pp. 1692-1695, Jul. 1991.

[45] E. Y. Lam, “Analysis of the DCT coefficient distributions for document coding,” *IEEE Signal Processing Letters,* vol.11, no.2, pp. 97- 100, Feb. 2004.

[46] V. Lappalainen, A. Hallapuro, and Timo. D. Hämäläinen, “Performance of H.26L encoder on general-purpose processor,” *IEEE Conference on Consumer Electronics (ICCE),* pp. 266-267, Los Angeles, USA, June 2001.

[47] S.W. Lee and C. C. J. Kuo, “Complexity modeling of H.264/AVC CAVLC/UVLC entropy decoders,” *IEEE ISCAS*, pp.1616-1619, Seattle, USA, 2008.

[48] M. C. Lee, K. W. Chan, and D. A. Adjeroh, “Quantization of 3D-DCT coefficients and scan order for video compression”, *Journal of visual communication and image representation*, vol. 8, Issue 4, pp. 405-422, 1997.

[49] G. Lee, J. Song, and R. Park, “Three-dimensional DCT/WT compression using motion vector segmentation for low bit-rate video coding,” *International Conference on Image Processing,* pp.456-459, Oct 1997.

[50] Y. Liang, F. Chebil, and A. Islam, “Compressed domain transcoding solutions for MPEG-4 visual simple profile and H.263 baseline videos in 3GPP services and appli-

cations," *IEEE Transactions on Consumer Electronics,* vol. 52, no.2, pp. 507- 514, May 2006.

[51] H. Liu, R. Xu, and Z. Liu, "Implementation of H.264 on TMS320DM642," *IEEE ICESS*, pp.605-609, Chengdu, China, 2008.

[52] S. Y. Ma, C. F. Shen, and L. G. Chen, "Analysis and reduction of reference frames for motion estimation in MPEG-4 AVC JVT/H.264," *IEEE conference on Acoustic, Speech and Signal Processing (ICASSP)*, pp. 145-148, Hongkong, China, April 2003.

[53] H. S. Malvar, "Fast computation of the discrete cosine transform through fast Hartley transform," *Electronics Letters*, vol. 22, No. 7, pp. 352–353,1986.

[54] D. Marpe, V. George, H. L. Cycon, and K. U. Barthel, "Performance evaluation of Motion-JPEG2000 in comparison with H.264/AVC operated in intra coding mode," *SPIE conference on Wavelet Applications in Industrial Processing, Photonics East*, Rhode Island, USA, Oct. 2003.

[55] T. Mekky, S. Boussakta, and M. Darnell, "On the computation of the 3-D DCT," *IEEE International Conference on Electronics, Circuits and Systems,* pp. 1141-1143, Dec. 2003.

[56] T. Miki, S. Hotani, and T. Kawahara, "Error resilience features of MPEG-4 audio-visual coding and their application to 3G multimedia terminals," *IEEE Conference on Communication Technology (ICCT),* vol.1, pp.805-808, Beijing, China, 2000.

[57] Y. H. Moon, G. Y. Kim, and J. H. Kim, "An Improved Early Detection Algorithm for All-Zero Blocks in H.264 Video Coding," *IEEE Trans. Circuits Systems Video Technol.,* Vol. 15, No.8, pp. 1053-1057, Aug. 2005.

[58] A. Navarro, A. Silva, and J. Tavares, "MPEG-4 codec performance using a fast integer IDCT," *IEEE Int. Symp. on Consumer Electronics,* pp.1-5, St. Petersburg, Russia, 2006.

[59] Y. Nishida, K. Inoue, and V. G. Moshnyaga, "A Zero-Value Prediction Technique for Fast DCT Computation," *IEEE Workshop on Sig. Proc. Syst.*, pp. 165-170, Aug. 2003.

[60] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: Tools, performance and complexity," *IEEE Circuits and Systems Magazine*, vol. 4, Issue 1, pp. 7-28, 2004.

[61] M. Pao and M. T. Sun, "Modeling DCT Coefficients for Fast Video Encoding," *IEEE Trans. Circuits Systems Video Technology,* Vol. 9, No.4, pp. 608-616, Jun. 1999.

[62] Y. Pourmohammadi-Fallah, K. Asrar-Haghighi, and H. Alnuweiri, "Internet delivery of MPEG-4 object-based multimedia," *IEEE Transactions on Multimedia,* vol.10, no.3, pp. 68- 78, Sept. 2003.

[63] C. Poynton, “Digital video and HDTV algorithms and interfaces,” Morgan Kaufmann, 2002.

[64] Iain E. G. Richardson, “H.264 and MPEG-4 Video Compression - Video Coding for Next-generation Multimedia,” John Wiley & Sons, September 2003.

[65] S. Saponara, C. Blanch, K. Denolf, and J. Barmans, “The JVT advanced video coding standard: Complexity and performance analysis on a tool-by-tool basis,” *Packet Video Workshop*, Nantes, France, April 2003.

[66] A. Segall, “Bit allocation and encoding for vector sources,” *IEEE Transactions on Information Theory*, vol. 22, pp. 162–169, Mar. 1976.

[67] N. P. Sgouros, S. S. Athineos, P. E. Mardaki, A. P. Sarantidou, M. S. Sangriotis, P. G. Papageorgas, and N. G. Theofanous, “Use of an adaptive 3D-DCT scheme for coding multiview stereo images,” *IEEE International Symposium on Signal Processing and Information Technology,* pp.180-185, Dec. 2005.

[68] L. A. Sousa, “General method for eliminating redundant computations in video coding,” *Electronics Letters* , vol. 36, no. 4, pp.306-307, Feb. 2000.

[69] G. J. Sullivan and T. Wiegand, “Rate-distortion optimization for video compression,” *IEEE Signal Processing Magazine*, vol. 15, pp. 74-90, Nov. 1998.

[70] S.C. Tai, Y.G. Wu, and C.W. Lin, “An Adaptive 3-D Discrete Cosine Transform Coder for Medical Image Compression,” *IEEE Transactions on Information Technology in Biomedicine*, Vol.4, pp. 259-264, 2000.

[71] S. Tai, Y. Wu, and C. Lin, “An adaptive 3-D DCT coder for image compression,” *International Computer Symposium workshop on image processing and characters recognition*, pp. 35-40, Taiwan, Dec. 1998.

[72] D. S. Taubman and M. W. Marcellin, “JPEG2000: Image Compression Fundamentals, Standards and Practice,” Kluwer International Series in Engineering and Computer Science.

[73] V. Testoni and M. Costa, "Three-dimensional transforms and entropy coders for a fast embedded color video codec," *Brazilian Symposium on Computer Graphics and Image Processing*, pp.147-154, Oct. 2008.

[74] TMN5, Video codec test model, ITU-T/SG-15, Jan. 1995.

[75] TMN8, Video codec test model, ITU-T/SG15, June. 1997.

[76] C. H. Tseng, H. M. Wang, and J. F. Yang, “Improved and fast algorithms for intra 4×4 mode decision in H.264/AVC,” *IEEE International Symposium on Circuits and Systems,* pp. 2128- 2131, May 2005.

[77] K. Ugur, J. Lainema, A. Hallapuro, and M. Gabbouj, “Generating H.264/AVC compliant bitstreams for lightweight decoding operation suitable for mobile multimedia systems,” *IEEE ICASSP*, Vol. II, pp.33-36, Toulouse, France, 2006.

[78] A. Vetro and H. Sun, “An overview of MPEG-4 object-based encoding algorithms,” *IEEE International Conference on Information Technology: Coding and Computing*, pp. 366-369, Las Vegas, USA, April 2001.

[79] Vcodex, http://www.vcodex.com/h265.html cited: 2011/01/16

[80] H. S. Wang and R. M. Mersereau, “Fast algorithms for the estimation of motion vectors,” *IEEE Transactions on Image Processing*, vol. 8, Issue 3, pp. 435-438, Mar. 1999.

[81] H. Wang, S. Kwong, and C. W. Kok, “Analytical model of zero quantized DCT coefficients for video encoder optimization,” *IEEE International Conference on Multimedia and Expo,* pp.801-804, July 2006.

[82] H. Wang, S. Kwong, and C. Kok, “Efficient prediction algorithm of integer DCT coefficients for H.264/AVC optimization,” *IEEE Trans. Circuits Systems Video Technol.,* Vol. 16, No.4, pp. 547-552, Apr. 2006.

[83] H. Wang, S. Kwong, C. W. Kok, and M. Y. Chan, “Zero-quantized discrete cosine transform prediction technique for video encoding,” *IEE Proceedings of Vision, Image and Signal Processing,* vol. 153, no. 5, pp.677-683, Oct. 2006.

[84] H. Wang, S. Kwong, and C. Kok, “Fast video coding based on Gaussian model of DCT coefficients,” *IEEE ISCAS,* pp. 1073-1076, Island of Kos, Greece, May 2006.

[85] H. Wang, M. Y. Chan, S. Kwong, and C. W. Kok, “Novel quantized DCT for video encoder optimization,” *IEEE Signal Processing Letters,* vol.13, no.4, pp. 205- 208, April 2006.

[86] H. Wang and S. Kwong, “Hybrid Model to Detect Zero Quantized DCT Coefficients in H.264,” *IEEE Transactions on Multimedia,* vol. 9, No. 4, pp. 728-735, Jun. 2007.

[87] H. Wang, S. Kwong, and C. W. Kok, “Efficient Predictive Model of Zero Quantized DCT Coefficients for Fast Video Encoding,” *Image and Vision Computing***,** vol 25 , Issue 6, pp. 922-933, 2007.

[88] H. Wang and S. Kwong, “Prediction of zero quantized DCT coefficients in H.264/AVC using Hadamard transformed information,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 4, pp.510-515, April 2008.

[89] T. Wedi, S. Wittmann, T. Palfner, B. Schuur, and F. Knicker, “Advances on hybrid video voding,” *SPIE Visual Communications and Image Processing*, vol. 6508, pp. 650812, 2007.

[90] R. Westwater and B. Furht, "Real-time video compression – Techniques and algorithms," Kluwer Academic Publishers, USA, 1997.

[91] T. Wiegand, H. Schwarz, A. Joch, and F. Kossentini, "Rate-constraint coder control and comparison of video coding standards," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, vol.13, pp. 608-703, July 2003.

[92] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," *Proceedings of IEEE Conference on Image Processing (ICIP)*, Thessaloniki, Greece, Oct. 2001.

[93] H. M. Wong, O. C. Au, A. Chang, S. K. Yip, and C. W. Ho, "Fast mode decision and motion estimation for H.264 (FMDME)," *IEEE Symposium on Circuits and Systems (ISCAS), pp.* 473-476, Island of Kos, Greece, May 2006.

[94] Z. Xie, Y. Liu, W. Liu, and T. Yang, "AZB Prediction Algorithms in SSD-Based Video Coding," *IEEE International Conference on Signal Processing (ICSP),* pp. 16-20, Beijing, China, 2006.

[95] Z. Xie, Y. Liu, J. Liu, and T. Yang, "A General Method for Detecting All-zero Blocks Prior to DCT and Quantization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, No. 2, pp. 237–241, 2007.

[96] XVID codec, http://www.xvid.org/ cited: 2011/01/16

[97] K. Yamatani and N. Saito, "Improvement of DCT-Based compression algorithms using Poisson's Equation," *IEEE Transactions on Image Processing*, vol. 15, No. 12, pp. 3272–3289, Dec. 2006.

[98] A. Yin, R. Lee, and M. Flynn, "Early detection of all-zero coefficients in H.263," *Proceedings of Coding Symposium*, pp. 159-164, 1997.

[99] P. Yin, H. Y. C. Tourapis, A. M. Tourapis, and J. Boyce, "Fast mode estimation within H.264 codec," *IEEE Conference on Multimedia and Expo (ICME)*, pp. 517-520, Maryland, USA, July 2003.

[100] G. S. Yovanof and S. Liu, "Statistical analysis of the DCT coefficients and their quantization error," *IEEE Conference on Signals, Systems and Computers,* pp.601-605, Nov 1996.

[101] A. Yu, R. Lee, and M. Flynn, "Performance enhancement of H.263 encoder based on zero coefficient prediction," *ACM International Multimedia Conference*, pp. 21-29, 1997.

[102] A. Yu, R. Lee, and M. Flynn, "Early detection of all-zero coefficients in H.263," *Proceedings of Coding Symposium*, pp. 159-164, 1997.

[103] Z. Zhao, H. Chen, X. Wen, and A. Sang, “Real-time Video Compression Based-on Three-Dimensional Matrix DCT,” *IEEE Int. Conf. Signal Processing,* pp. 1155-1158, Istanbul, Turkey, Aug. 2004.

[104] X. Zhou, Z. Yu, and S. Yu, "Method for detecting all-zero DCT coefficients ahead of discrete cosine transformation and quantization," *Electronics Letters*, vol. 34, no. 19, pp.1839-1840, Sep 1998.

[105] F. Zou, Z. Lu, and H. Ling, “Statistical model of quantized DCT coefficients,” *International Conference on Signal Processing,* pp. 2572- 2575, Aug. 2004.