NEW METHODS FOR MOTION ESTIMATION WITH APPLICATIONS TO LOW COMPLEXITY VIDEO COMPRESSION

A Thesis

Submitted to the Faculty

of

Purdue University

by

Zhen Li

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2005

To my parents, Meilian Jin and Yanzong Li; To my wife, Limin Liu; and in memory of my grandmother, Zhugu Cai.

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Edward J. Delp, for his guidance and support. I truly appreciate the research freedom he allowed. And I am deeply grateful for his reminder to keep focused when I was zigzagging with various research topics, and his advice to look deeper into one challenging problem rather than playing around with many problems.

I would also like to thank my Doctoral Committee: Professors Charles A. Bouman, Michael D. Zoltowski, and Dongyan Xu, for their advice, encouragement, and insights.

I am grateful to the organizations which supported the research in this dissertation, in particular the Indiana Twenty-First Century Research and Technology Fund.

I appreciate the support and friendship of all my colleagues in the Video and Image Processing (VIPER) lab. It has been a joyful and memorable journey with these excellent people and scholars: Dr. Eduardo Asbun, Dr. Gregory Cook, Dr. Paul Salama, Dr. Lauren Christopher, Dr. Eugene Lin, Dr. Zoe Yuxin Liu, Dr. Yajie Sun, Dr. Jinwha Yang, Dr. Cuneyt Taskiran, Dr. Sahng-Gyu Park, Hyung Cook Kim, Jennifer Talavage, Hwayoung Um, Anthony Martone, Aravind Mikkilineni, Oriol Guitart, Michael Igarta, Liang Liang, Ying Chen, Fengqing Zhu, Ashok Mariappan and Hakeem Ogunleye. I would like to thank our visiting researchers from abroad for their perspectives: Professor Luis Torres, Professor Hwan Il Kang, Dr. Josep Prades-Nebot, Dr. Alberto Albiol, Dr. Werapon Chiracharit, Rafael Villoria, and Pablo Sabria.

I would like to thank Drs. Ya-Qin Zhang, Shipeng Li, Feng Wu and Guobin Shen for twice offering me the chance to visit Microsoft Research Asia as a summer intern. It was a great working and learning experience. I am grateful for having had

numerous compassionate discussions with these world-class video coding experts. And I truly miss all my friends and colleagues from Microsoft Research Asia.

I would like to thank Mr. Kevin O'Connell and Dr. Faisal Ishtiaq for the summer internship at Motorola Multimedia Research Lab.

I would like to thank all my friends from Tsinghua University for their encouragement and help.

I would like to thank Dr. Guoxiong Jin for his help to my family and me.

I have dedicated this document to my mother and father, and in memory of my grandmother for their support throughout these years. They have made tremendous sacrifices to survive this family in the extremely hard days and managed to support all the children with higher education opportunities. I could never have gone this far without every bit of their help. And I am deeply grateful for my brother and sisters. They are the best.

Finally, I thank my wife, Limin Liu, for her patience and love. She is my best friend and I am eternally grateful for having her on my side. Her encouragement, understanding, care, and love have made my life particularly meaningful.

Towards finishing this dissertation, I am thankful to have the opportunity to work on these research problems. The more I work on them, the more I learn to appreciate an Aristotle's quote, "It is the mark of an educated mind to expect that amount of exactness which the nature of the particular subject admits."

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

ABSTRACT

Li, Zhen. Ph.D., Purdue University, December, 2005. New Methods for Motion Estimation with Applications to Low Complexity Video Compression. Major Professor: Edward J. Delp.

The goal in video compression is to remove the redundancy in a video sequence while preserving its fidelity. Motion estimation can significantly improve video coding efficiency by reducing temporal redundancy. In this thesis we focus on new methods for motion estimation with applications in low complexity video compression.

In this thesis we study the performance of low complexity video coding. Theoretical rate-distortion performance is derived and evaluated for both low complexity video coding and conventional motion-compensated prediction (MCP) based video coding. Based on our analysis of sub-pixel and multi-reference motion search methods used to improve low complexity video coding efficiency, we propose a new refined side motion estimation method to better extract motion information at the video decoder.

Motion side estimation methods in general assume a motion model in a video sequence and use computationally intensive motion search. In this thesis we propose a novel side information estimation method based on universal prediction. Our method does not assume an underlying motion model. Instead, it makes predictions based on its observation of the past video information. The experimental results show that this new method can significantly reduce the computational complexity while achieving comparable performance for many video sequences.

Lossy image and video compression is often accompanied by annoying artifacts. In this thesis we present a transform domain Markov Random Field model to address the artifact reduction. We present two methods, which we denote as TD-MRF and

TSD-MRF, based on this model. We show by objective and subjective comparisons that transform domain post processing methods can substantially reduce the computational complexity compared with conventional spatial domain method and still achieve significant artifact reduction.

1. INTRODUCTION

As we enter into the new millennium, video has been more and more an inseparable element in our society. People are accustomed to video applications that did not even exist merely a decade ago. Just to name a few, the digital versatile disk (DVD), digital video camcorder, and video on demand are common applications used by most consumers. Many more new video applications and devices are emerging and evolving. Advances in hardware and software, as well as standardization activities and research in industry and academia, are making it much easier and affordable to capture, store and transmit video signals.

1.1 Digital Video Coding

Digital video signals are represented and stored as sequences of video frames [1, 2]. Each frame is a rectangular grid of pixels. The size of the grid defines the spatial resolution of a video frame. Different video formats have different spatial resolutions. For instance, the resolution can be as small as QCIF(176×144) for a video camera phone, or it can be as large as 4096×2048 or even larger for digital cinema applications. The pixels are in general represented in the RGB, YUV or other tri-component color spaces. Each component is typically digitized to 8 to 12 bits.

The storage and transmission of uncompressed video requires significant resources. For example, we examine one second of a standard NTSC CCIR601(720×480) sequence at 30 frames per second. Every pixel is represented by three color components, each being digitized to 8 bit. One second of the sequence requires $720 \times 480 \times 3 \times 8 \times 30 = 237$ Mbits. One hour of such a sequence requires 834 Gbits. For digital cinema with 4096×2048 pixel resolution and 12 bits/pixel [3], 8.4 Gbits/second or 30,300.8

Gbits per hour is required. This is beyond the storage capacity of many current entertainment devices. Even for resolution formats such as QCIF(176×144), the transmission of 10 frames/second uncompressed QCIF frames requires a bandwidth of 5.8 Mbits/second, which is still well above the consumer network bandwidth capacity. With the advances in material science and engineering, storage and network bandwidth capacity may dramatically increase in the future. However, the storage and transmission of unnecessary data still wastes resources, especially given the amount of redundancy in a typical natural video sequence.

To reduce the storage and transmission bandwidth requirements of digital video, compression is used. The goal of video compression is to remove redundancy in a video signal. The introduction of video compression also affects many system performance tradeoffs and the design and selection of a video codec is not only based on its ability to compression information. Issues such as data rate, algorithm complexity, transmission channel characteristics, video source statistics, and buffer conformance should all be considered in the design. Various application scenarios may have different priorities in video coding. For instance, for video data archiving applications, highest priority is often given to preserving the original data as much as possible, while for real-time video streaming the channel bandwidth imposes a limit on the available data rate. For mobile video devices, it is important to make the decoder less complex, while for video surveillance, it is more compelling to keep the encoder simple.

With so many video codecs available, one major concern is to assure the compatibility among different codecs produced by various vendors. Many standardization activities have occurred in the last two decades to achieve this goal. The design of a video codec will have to follow certain rules so that an encoded video stream can be decoded by decoders produced by others. In the following we review several widely used video coding structures and video coding standards. We also give a brief introduction to low complexity video coding.

1.2 Hybrid Video Coding

Video signals contain information in three dimensions. These dimensions are modeled by the spatial and temporal domains. Digital video compression seeks to reduce the redundancy in both domains. While there are many methods that can reduce the spatial redundancy, transform coding proves to be very effective as it can compact the energy into a few coefficients and only these coefficients need to be coded.

One early transform-based video coding method is motion-JPEG. It exploits the spatial redundancy inside a frame using the still image coding standard JPEG [4]. Each frame is first divided into blocks of equal size. These blocks are transformed by the Discrete Cosine Transform(DCT). The DCT coefficients are then quantized and entropy coded with a variable length code(VLC). This type of coding does not rely on other frames in the video sequence and is referred to as INTRA coding, or I frame coding. Such INTRA coding methods are still used by professional video editing systems where random access to any frame is of high priority while the constraint on the data rate is relatively minimal [5].

In INTRA coding, only the spatial redundancy is exploited and the temporal dependence is not used. Therefore the coding efficiency is generally very low. Changes in a video sequence are due to the motion of objects with respect to previous frames. Higher coding efficiency can be achieved by taking advantage of this temporal dependence. This is referred to as Predictive, or INTER-frame coding, also known as P frame coding. The ability to exploit the temporal redundancy to improve coding efficiency distinguishes video coding fundamentally from still image coding.

Based on this idea, a general video coding structure, referred to as a hybrid video coder, was proposed. Hybrid video coding performs motion estimation first and then codes the residual frames using 2-D transforms. It is widely used and is the baseline structure for current video coding standards. A typical hybrid video coding structure is shown in Fig. 1.1.

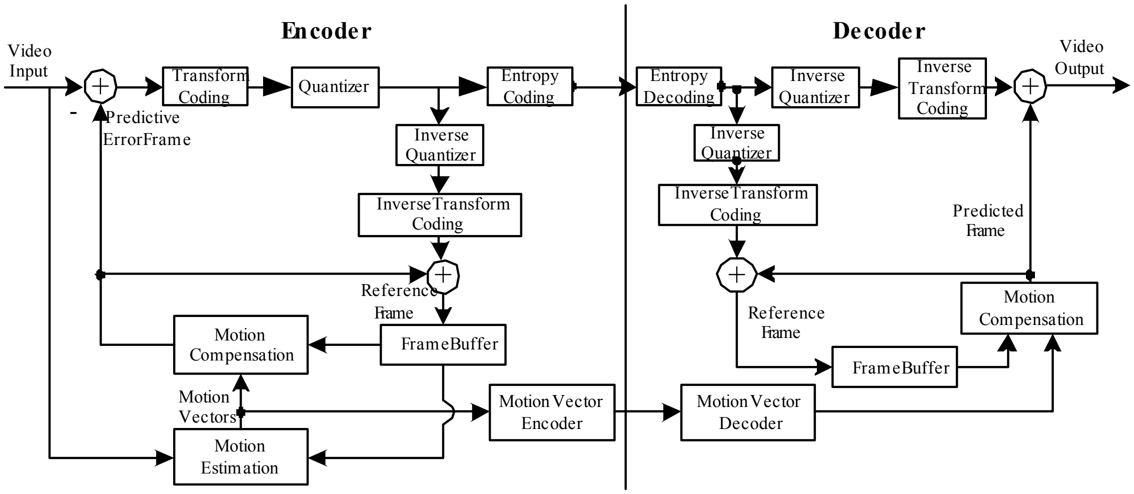


Fig. 1.1. Hybrid video coding framework.

Motion estimation is used to de-correlate the temporal dependence of the input video sequence. A motion estimation unit at the video encoder in general includes the following three steps. In the first step, motion estimation is used to estimate the motion between blocks in the reference frames and a block in the current frame. The second step creates the displaced motion compensated frame and is referred to as motion compensation. The final step obtains the residual frame as the difference between the current frame and the displaced motion compensated frame. A similar procedure is performed at the decoder. The decoder first decodes the residual frame. Then it obtains the motion compensated frame using the motion vectors and the reference frames. Finally the difference frame is added to the motion compensated frame to obtain the reconstructed current frame. This reconstructed frame can be used as the reference frame for the next frames.

The residual frame obtained in the motion estimation step is then sent to a transform coder to de-correlate the spatial redundancy. The transform coder decomposes each block into an orthogonal or near orthogonal basis. The energy of the residual signal is compacted to a few coefficients. Lossless coding can also be used to represent these coefficients. The typical compression ratio of lossless video coding ranges from $5:1$ to $2:1$. Since the human visual system is insensitive to distortions at high frequencies, the distortion requirements can be relaxed leading to lossy video coding. In lossy video coding, the residual frame is transformed and quantized. The quantized coefficients are then entropy coded to form the data stream. The quantization process in general is lossy, while entropy coding is lossless. Depending on the specific transforms used, transform coding can be either lossless or lossy.

One widely used transform is the DCT transform, which is sub-optimal to the optimal Kahunen-Loeve transform(KLT) but simpler in terms of implementation. An input image of size $M \times N$ is divided into blocks of size $L \times L$ and the 2-D DCT transform is done on each block. The reason why the residual frames are broken into blocks is that it not only fits well to most 2-D transforms but also reduces the hardware implementation complexity and memory cost. However, recent research found that a versatile shape can be more suitable and each video sequence may require a specific set of block shapes and sizes, which are later defined as a video object [6].

Besides the DCT transform, wavelet transform coding has also generated a great deal of research interest. One of the reasons that wavelet transforms have been successful is that it is more suitable than DCT for de-correlating the spatial dependence and approximates a piecewise smooth signals. It is also more natural to introduce scalability into wavelet coded coefficients as many wavelet transforms have a multi-resolution interpretation. Such scalability is a desirable feature in real-time streaming video applications. On the downside, many wavelet transform coding methods use a global transform, which requires the buffering of the entire image or video frame before the encoding and leads to coding delay and increases memory cost. In contrast, the DCT is relatively simple for hardware implementation.

Lossy compression inevitably leads to imperfect descriptions of the original videos. Some distortions are invisible or only cause minor quality problems, while others can be very annoying. There are some well-known artifacts in transform-based video coding, for instance, the contouring artifact and the block artifact. By incorporating statistical models of the input video sequence, many post processing methods have been developed to remove the artifacts and improve visual quality. We will discuss more on post processing later in this chapter.

1.3 Overview of Current Coding Standards

Several video coding standards have been adopted in the last two decades.

Two organizations have strongly influenced the development of video coding standards: the Video Coding Experts Group (VCEG) of ITU-T (Telecommunication Standardization Sector of International Telecommunications Union) and the Moving Picture Experts Group (MPEG) of ISO/IEC (International Organization for Standardization). Video coding standards developed by ITU-T are designated by

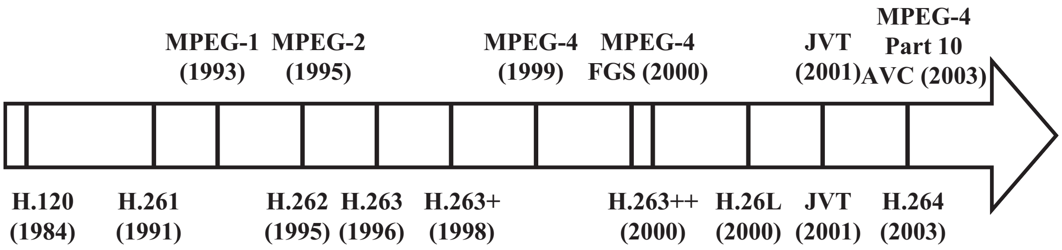


Fig. 1.2. Timeline of video coding standards.

Table 1.1
Current video coding standards.

Title	Organization	Targeted rates	Current Status
H.120	CCITT	1544-2048 Kbps	approved 1984, revised 1988
H.261	ITU-T	64-2048 Kbps	approved 1991, revised 1993
MPEG-1	MPEG	1-2 Mbps	approved 1993
MPEG-2	MPEG	4-30 Mbps	approved 1995
H.263	ITU-T	10-2048 kbps	approved 1996
H.263+	ITU-T	all	approved 1998
MPEG-4	MPEG	all	approved 1999
H.264 (MPEG-4 V-10)	JVT	all	approved 2004

the label “H.26x,” and standards developed by MPEG are designated by the label “MPEG-x.” The major standards are listed in Table 1.1. The timeline of video coding standards development is shown in Fig. 1.2.

- H.120

  H.120 [7] was the first digital coding standard developed by ITU-T to code natural continuous-tone visual content of any kind(both still image and video). It was also the first international video coding standard. H.120 was approved by ITU-T(when it was known as CCITT) in 1984 and revised in 1988. The basic components in H.120 are differential pulse-code modulation(DPCM), scalar quantization, and variable length coding(VLC). The second version of H.120 was approved in 1988. It added motion compensation and background prediction. The operational data rate of H.120 is 1544 and 2048 Kbit/second.

- H.261/MPEG-1

  H.261 [8] was the first practical and widely adopted international digital video coding standard. H.261 was approved in 1991 and revised in 1993. Its operational data rate is $64-2048$ Kbit/second, part of which is practically affordable for typical telecom data rates($64-320$ Kbit/second). The coding framework of H.261 was based on the hybrid video coding structure. The basic coding components of H.261 are 16×16 macro-block(MB)-level motion estimation, full pixel motion search, 8×8 block DCT, scalar quantization, and two-dimensional run-level VLC entropy coding. H.261 also proposed half-pixel motion search, which was not adopted into the final standard due to computational complexity concern. H.261 is still used in many video applications due to its simple structure and hardware implementation.

  Meanwhile, MPEG-1 [9] was developed in the early 1990s and approved in 1993 by ISO/IEC. MPEG-1 is a widely accepted and successful video coding standard. It is capable of delivering VHS videotape quality at about 1.5M bits/second. Its operational data rate is 1-2 Mbit/second. It followed the hybrid coding structure of H.261. Compared with H.261, it added bi-directionally predicted frame(B frame) and half-pixel motion search. The coding efficiency of MPEG-1 is generally better than H.261 at data rates higher than 1M bits/second.

- MPEG-2/H.262

  After the development of H.261 and MPEG-1, both ITU-T and ISO/IEC found that it is necessary to set up a joint project to develop a video coding standard for both standard-definition and high-definition televisions(SDTV and HDTV), which in late 1994 eventually led to the H.262, or better known as MPEG-2 [10]. The target data rate of MPEG-2 is $4 - 30$ Mbit/second. The major new features of MPEG-2 were efficient handling of interlace scanned pictures and hierarchical scalability. Spatial, temporal and SNR (Signal-to-Noise Ratio) scalabilities were first introduced into a video coding standard in MPEG-2 and later became standard feature for many real-time streaming video applications.

- H.263/H.263+, ++/MPEG-4

  The mid-1990s witnessed the Internet boom. The early Internet was operated at low data rates, e.g., below 64 Kbit/second for PSTN (Public Switched Telephone Network). The target data rate of H.263 [11] is hence about $10 - 30$ Kbit/second initially, which leaves some bandwidth for speech, overhead payload and error control. The data rate range was later extended to $10 - 2048$ K bit/second after it became obvious that the new coding standard was superior to H.261 and hence was capable of handling not only internet video streaming but also other video applications. H.263 was approved by ITU-T in 1996. The major new features of H.263 were variable block-size motion compensation, overlapped block motion compensation(OBMC), and 3-D run-length-last VLC entropy coding. Test results confirmed that H.263 could achieve the same video quality as H.261 at low data rates with half or less than half of the data rate, which roughly corresponds to a 3 dB coding gain in terms of peak signal-to-noise ratio (PSNR).

  The second version of H.263, known as H.263+, added many new features to H.263 and was approved in 1998 by ITU-T. Compared to H.263 and other earlier video coding standards, H.263+ is the first standard to provide error resilience capability for video transmission over a wireless channel. Moreover, H.263+ added customized and flexible video formats, scalability, and backward-compatible supplemental enhancement information. The effective operational data rate was extended to any data rate and any frame rate.

Some further improvements of H.263, known as H.263++, were approved in 2000. Several new optional features were added to the standard. A list of available H.263 and H.263+/++ options is shown in Table 1.2.

Meanwhile, ISO/IEC was also developing a new video coding standard, known as MPEG-4 [6]. The development of MPEG-4 adopted many new coding features used in H.263/H.263+/H.263++. It was approved by ISO in 1999. Compared to other standards, it is the first standard to propose object-based video coding. In late 1998, a new scalability mode, namely, the fine granular scalability(FGS) [12] was proposed and finally adopted in MPEG-4 [13], which greatly enhanced MPEG-4 as a streaming video standard.

- H.26L/H.264/AVC

  H.26L was a standard developed by the ITU-T VCEG since August 1999. In December 2001, the Joint Video Team (JVT) was formed between the ITU-T VCEG and the ISO/IEC MPEG to work on H.26L as a joint project. The JVT project was finalized in December 2003, and the new standard was approved as H.264, as a new ITU-T recommendation, and also as MPEG-4 Part 10 AVC (Advanced Video Coding), as a new part of ISO/IEC MPEG-4.

  Many video coding standards contain different configurations of capabilities, which are specified by *profiles* and *levels*. A *profile* is a set of algorithmic features, while a *level* is a degree of capability. H.264/AVC currently has three profiles:

  – Baseline, which addresses a broad range of applications, in particular those requiring low latency;

  – Main, which adds features such as interlace, B-Slices, and CABAC (context-based adaptive binary arithmetic coding) [14];

  – Extended, which targets streaming video, and includes features to improve error resilience and to facilitate switching between different bit streams.

Table 1.2
H.263/H.263+ annex list.

Annex Index	Title
A	Inverse Transform Accuracy Specification
B	Hypothetical Reference Decoder
C	Considerations for multipoint
D	Unrestricted Motion Vector Mode
E	Syntax-based Arithmetic Coding Mode
F	Advanced Prediction
G	PB Frames
H	Forward error correction for coded video signal
I	Advanced Intra Coding mode
J	De-blocking Filter mode
K	Slice Structured mode
L	Supplementary enhancement information specification
M	Improved PB-frames mode
N	Reference Picture Selection mode
O	Temporal, SNR, and Spatial Scalability mode
P	Reference Picture Re-sampling
Q	Reduced-resolution Update Mode
R	Independent Segment Decoding mode
S	Alternative INTER VLC mode
T	Modified Quantization mode
U	Enhanced reference picture selection
V	Data partitioning
W	Enhanced supplementary info
X	Profiles/Levels

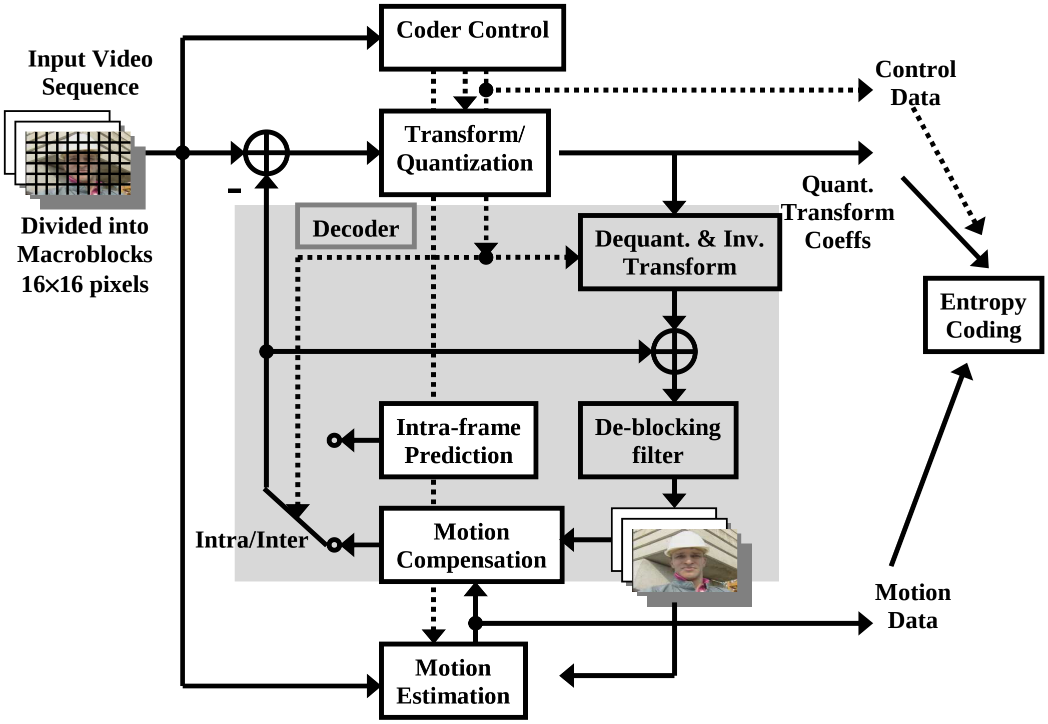


Fig. 1.3. Basic macroblock coding structure in H.26L/H.264.

As in the previous video coding standards, H.26L/H.264 specifies a hybrid video coding implementation using block-based transform coding and motion estimation, as illustrated in Fig. 1.3.

1.4 Motion Estimation

The goal of video compression is to remove the redundancy in the video signal. In still image coding, the main task is to improve the transform and entropy coding efficiency. However, the optimal performance of an individual video frame in video coding is upper bounded by the entropy of its residual frame. Motion estimation plays an important role in video coding by taking advantage of the temporal redundancy across the sequence. The general goals of motion estimation methods are to improve the prediction accuracy, or to reduce the implementation complexity, or a combination of the two. In this Section, we give a survey of current motion estimation methods.

1.4.1 Motion Search Accuracy

Based on Shannon's coding theorem [15], the coding efficiency of an individual video frame is upper bounded by the entropy of its residual frame, which is the difference between the current frame and the motion compensated frame. A more accurate motion estimation will lead to a residual frame with smaller entropy. Theoretically a pixel-wise motion search can achieve the most accurate motion estimation. In practice, side information used to describe the motion information, i.e., the motion vectors, also needs to be transmitted to reconstruct the motion-compensated reference frame at the decoder. Pixel-wise motion vectors may cost many more bits than the saving in residual frame bits. To achieve a compromise between accuracy and side information cost, block based motion estimation is used. The frame is first divided into blocks and a block matching motion search is used. The general process is to compare the current block with all the blocks having the same shape in the reference frames and finding the block with the minimum difference. This method is referred to as full motion search. An example of motion estimation is shown in Fig. 1.4.

Full motion search has high computational complexity. To reduce the computational complexity, full search is seldom used. Instead, simplified motion search, such as the three-step search [11], is used. Such sub-optimal motion search methods can achieve comparable performance to the full motion search results while considerably reducing the computational complexity. An example of the three-step motion search is shown in Fig. 1.5. In this example [2], a 14 × 14 pixel search neighborhood is depicted. The search range size is chosen so that the total search neighborhood can be covered in finding the local minimum. The search areas are square. The length of the search for step 1 is chosen to be greater than or equal to half the length of

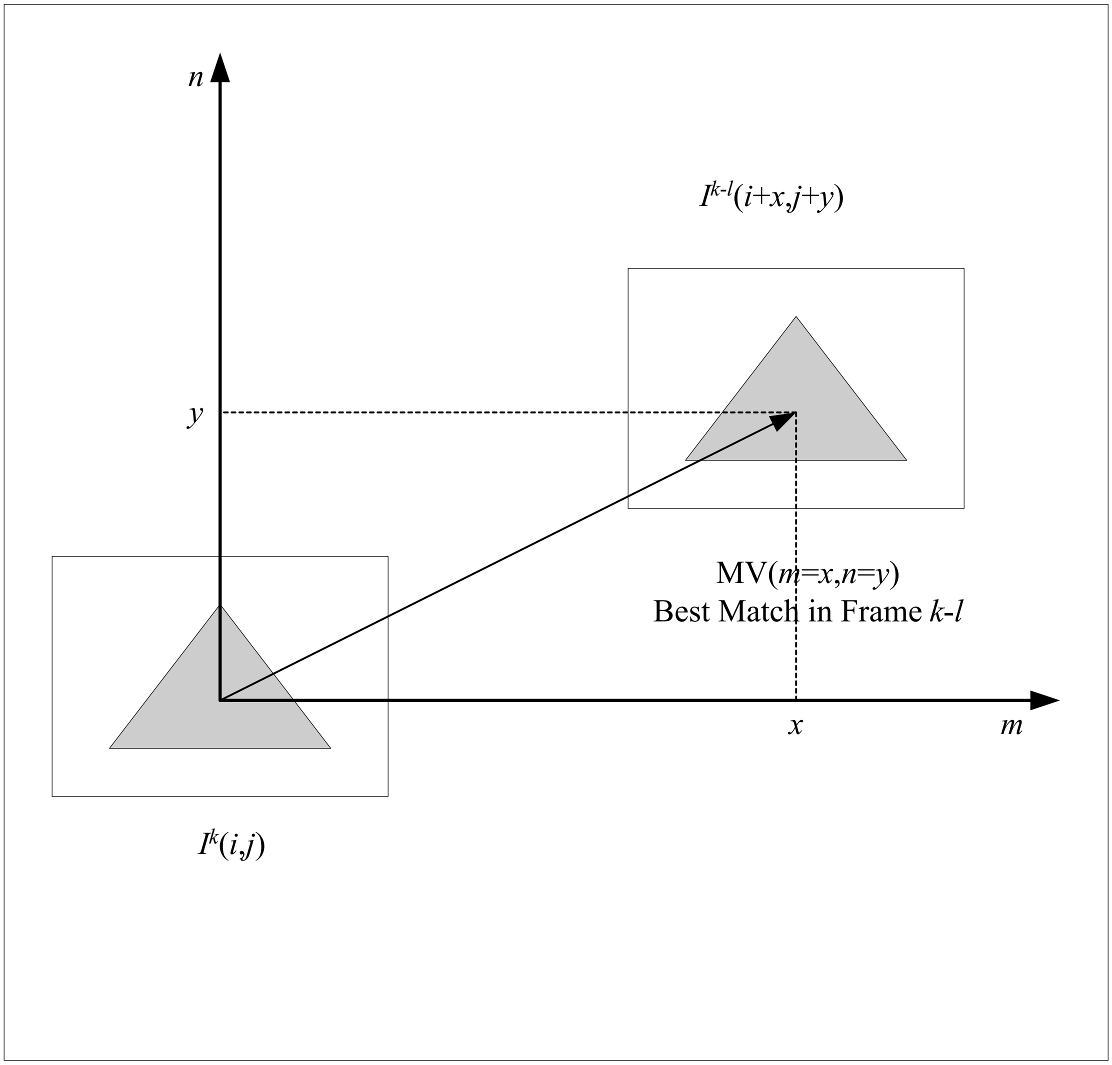


Fig. 1.4. Motion estimation.

the range of the search neighborhood, where in this example the search area is 8×8. The length is reduced by half after each of the first two steps are completed. Nine points for each step are compared using the matching criteria. These consist of the central point and eight equally spaced points along the perimeter of the search area. The search area for step 1 is centered on the search neighborhood. The search proceeds by centering the search area for the next step over the best match from the previous step. The overall best match is the pixel displacement chosen to minimize the matching criteria in step 3. The total number of required comparisons for the three-step algorithm is 25, which is a 87% reduction in complexity versus the full search method for a 14×14 pixel search neighborhood. It should be noted that even with these simplifications, motion estimation still accounts for $40\% - 60\%$ of the entire complexity in the encoder and may even take up to 80% of the complexity in the decoder. Another point, as noted in [16], is that the three-step search assumes the existence of a local minimum. However, in practical video coding, it is possible that a local minimum does not exist or exists but is far from the global minimum, both cases will lead to inaccurate motion estimation.

1.4.2 Motion Estimation Methods Used in Standards

In this Section we discuss motion estimation methods used in the video coding standards.

H.120

Motion estimation was not used in H.120. Only conditional replenishment(CR) is used to reduce the temporal redundancy. CR coding sends a signal to the decoder to indicate which area is repeated in the current frame. For the areas that are not repeated, CR sends the new information to replace the changed areas.

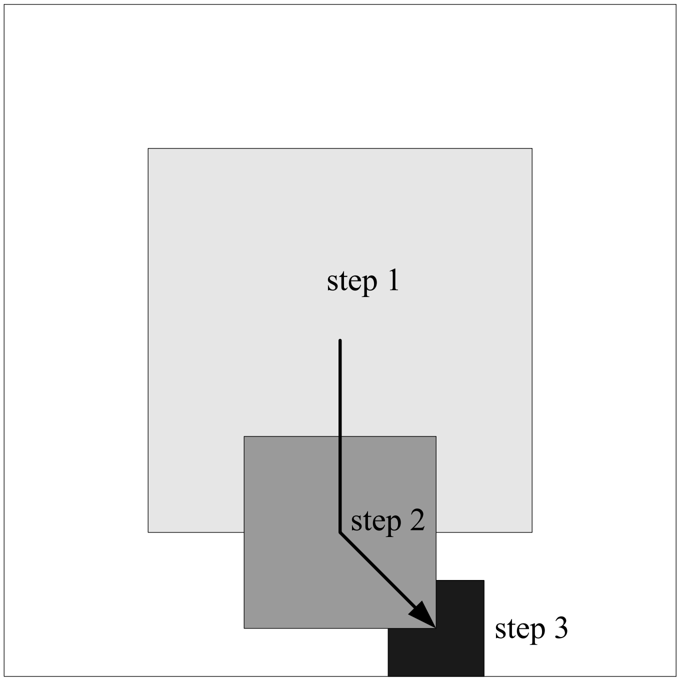


Fig. 1.5. Three-step motion search.

H.261

The idea of a displaced frame difference(DFD), or a residual frame, was first introduced in H.261. The motion search resolution adopted in H.261 is full-pixel. Only forward motion estimation is used. A simple example of forward motion estimation is shown in Fig. 1.6. The search unit in H.261 is a 16×16 macro-block. Full search or other forward motion search, such as the three-step search, can be used.

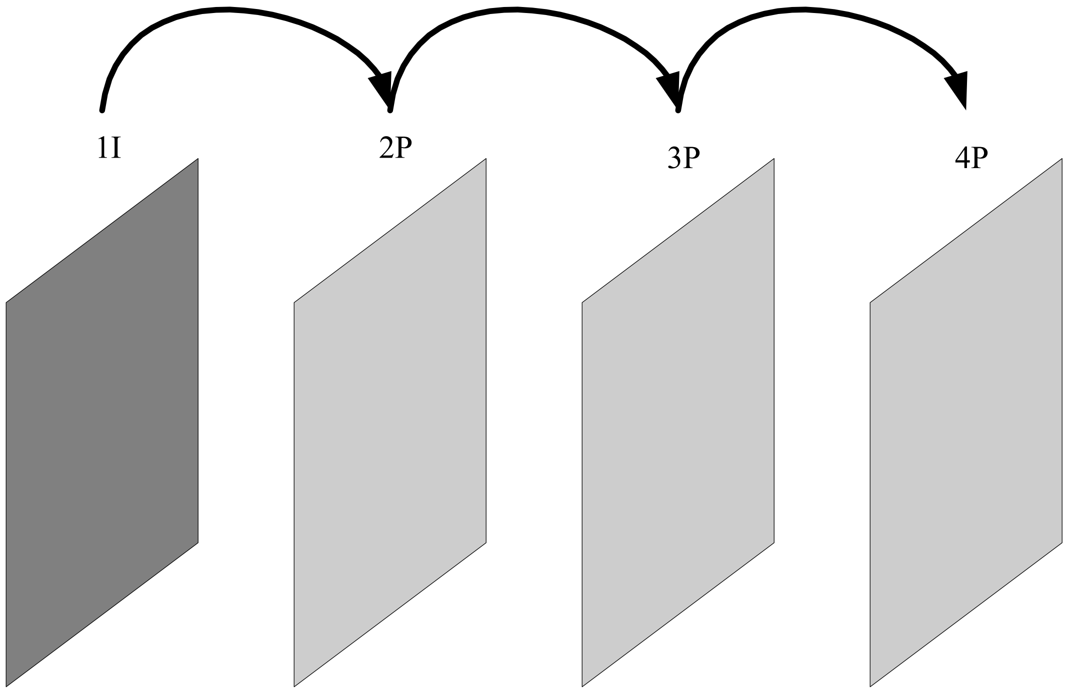


Fig. 1.6. Forward motion estimation.

MPEG-1

In MPEG-1, a new estimation approach, backward motion estimation, was introduced. The bi-directionally predictive frame (B frame) uses references from both the previous and next frames to construct its reference frame. An illustration of bi-directional motion estimation is shown in the Fig. 1.7. B frame motion estimation can significantly improves coding efficiency by constructing a more accurate motion-compensated reference. The B frame method introduces frame delay and increases the memory budget and computational complexity.

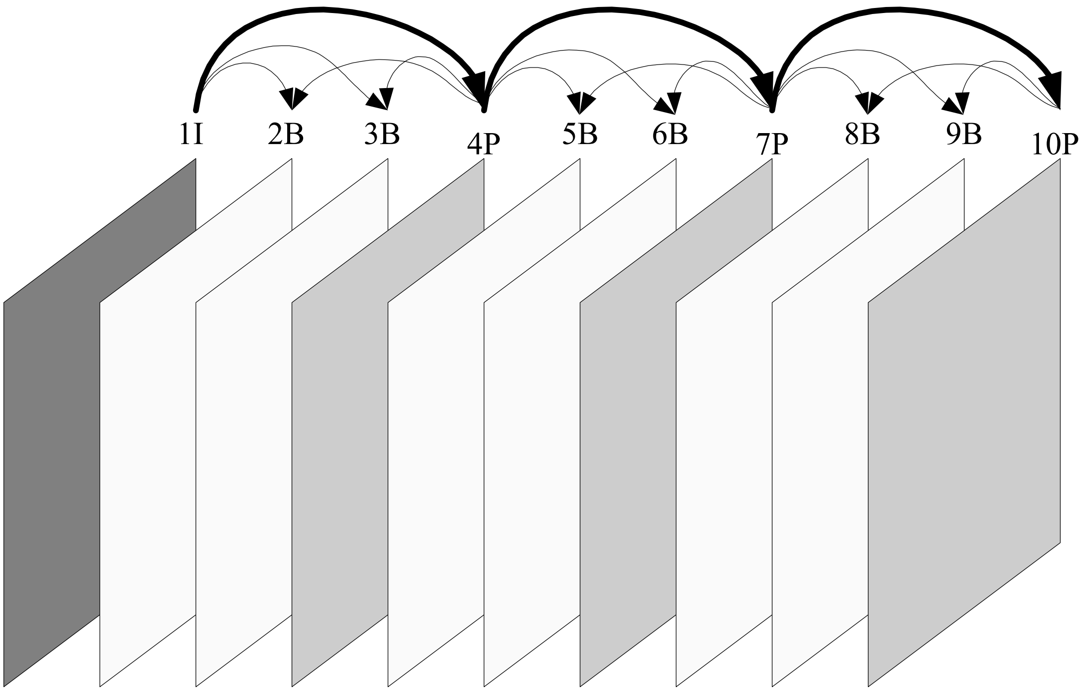


Fig. 1.7. Bi-directional motion estimation.

MPEG-2

MPEG-2 uses the motion estimation methods developed for MPEG-1. It also introduced temporal, spatial and SNR scalability. The motion estimation used in the scalable codec is one-loop motion estimation. The basic structure of the one-loop motion estimation defined in MPEG-2 is shown in Fig. 1.8. Only the base layer reconstructed frames are used for reference. The reason for this relatively conservative motion estimation is to avoid drift error. A drift error is introduced when the encoder and decoder do not use the same reference information for the reconstruction of video sequences [17, 18].

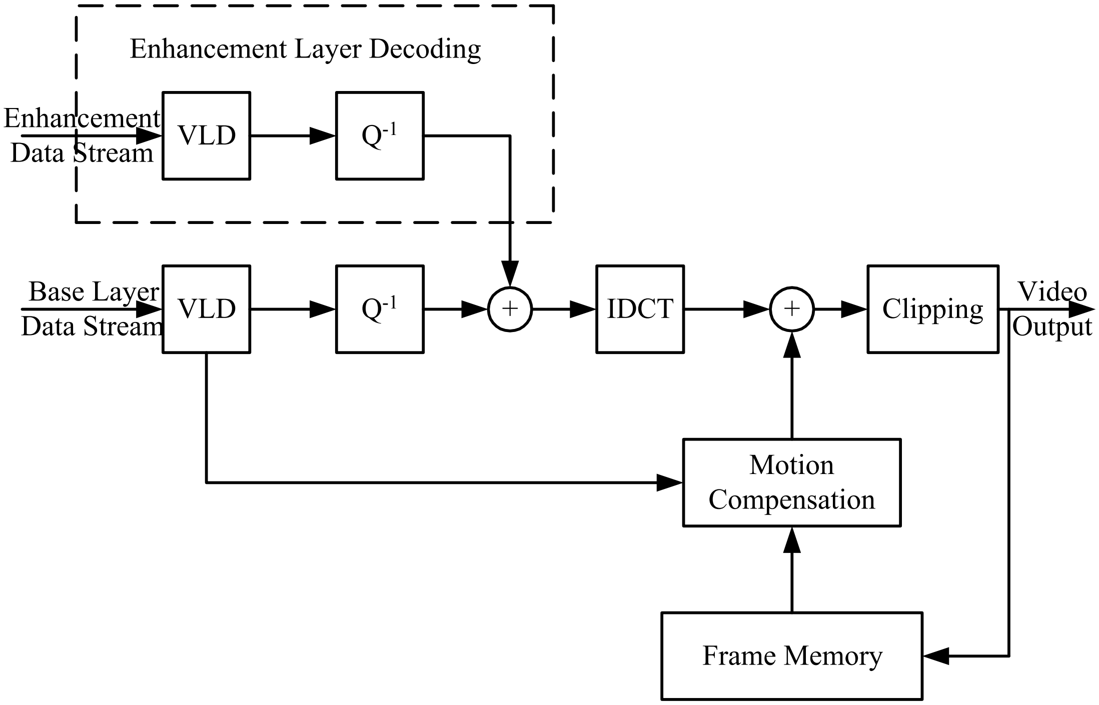


Fig. 1.8. SNR scalability video coding defined in MPEG-2.

H.263, H.263+, H.263++ and MPEG-4

Half-pixel motion search became a standard option in H.263 and MPEG-4. It was also found that motion search in the unit of a 16×16 macro-block(MB) is not efficient enough since many video objects can not be simply represented as a combination of 16×16 MBs. When an object cannot be represented this way, an edge appears in the residual frame. This increases the residual frame's entropy. Hence, variable block size motion search was adopted in the standard. Several other optional motion estimation modes were adopted, including reference picture selection, overlapped block motion compensation(OBMC), and the use of four motion vectors for each macro-block [11].

MPEG-4 FGS

One new feature of MPEG-4 is fine granular scalability(FGS) coding. FGS coding allows truncating and decoding a bit stream at any point with bit-plane coding. This is a desirable property in real-time video streaming, where bandwidth is either heterogeneous or fluctuating. The FGS motion estimation uses the scalable motion estimation structure defined in MPEG-2 as shown in Fig. 1.9 and Fig. 1.10.

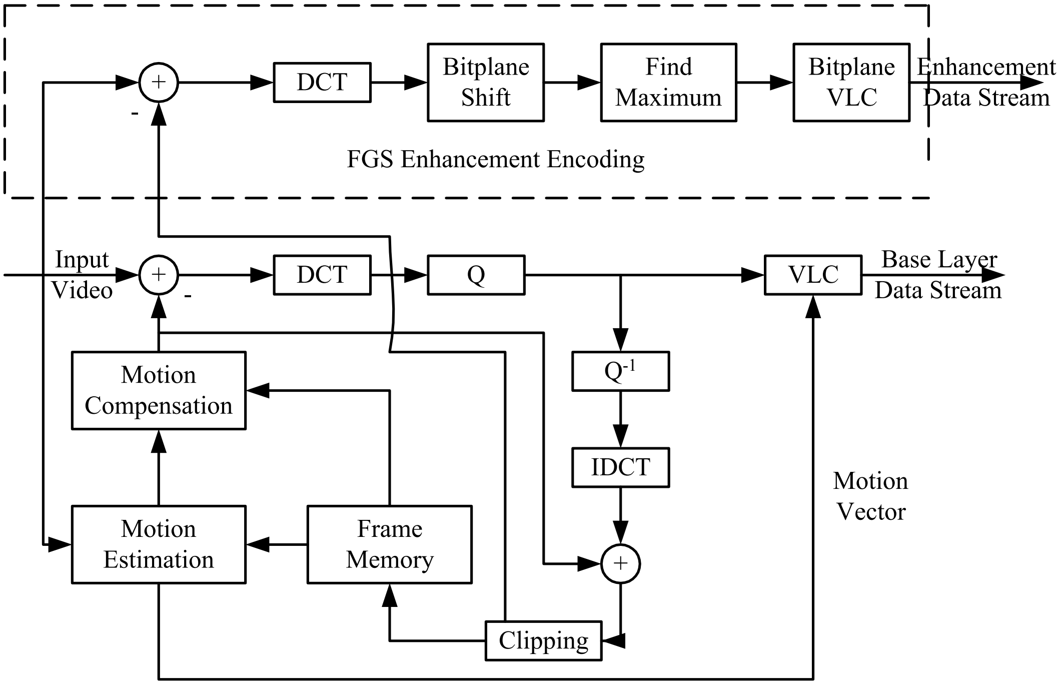


Fig. 1.9. MPEG-4 FGS video encoder structure.

H.26L, AVC, JVT and H.264

H.26L/AVC/JVT/H.264 allow the use of more variable block sizes, as shown in Fig. 1.11, including $16\times16, 16\times8, 8\times16, 8\times8, 8\times4, 4\times8$ and 4×4. Finer resolution samples, such as $\frac{1}{4}$-pel resolution, were adopted as the standard motion search resolution. These video coding standards also support $\frac{1}{8}$, $\frac{1}{16}$ and $\frac{1}{32}$-pel resolution motion

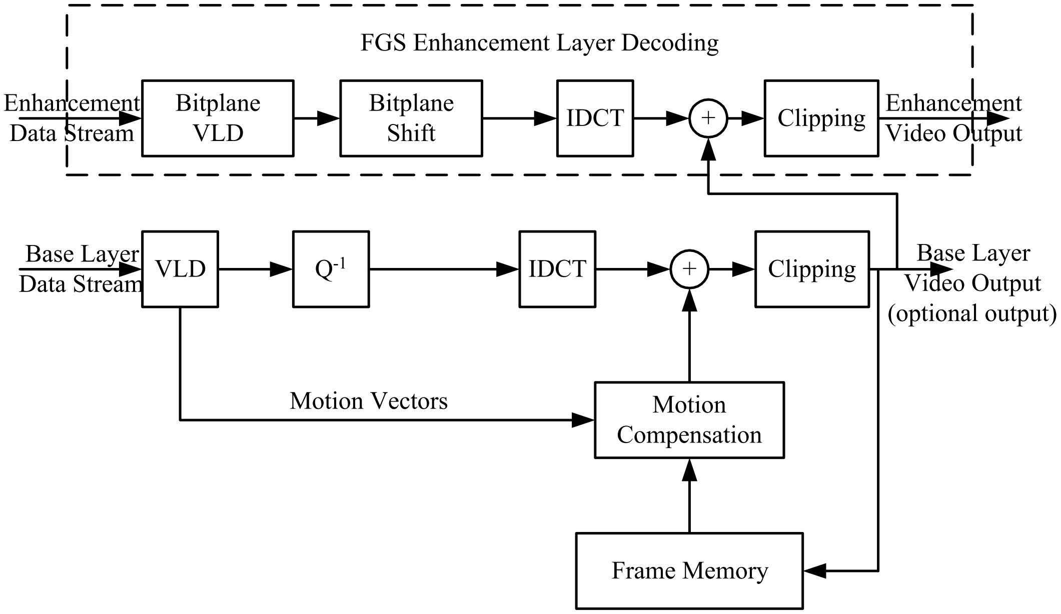


Fig. 1.10. MPEG-4 FGS video decoder structure.

search when needed. Multi-reference motion search is used to further improve the motion estimation accuracy, as illustrated in Fig. 1.12 and Fig. 1.13 [19–21].

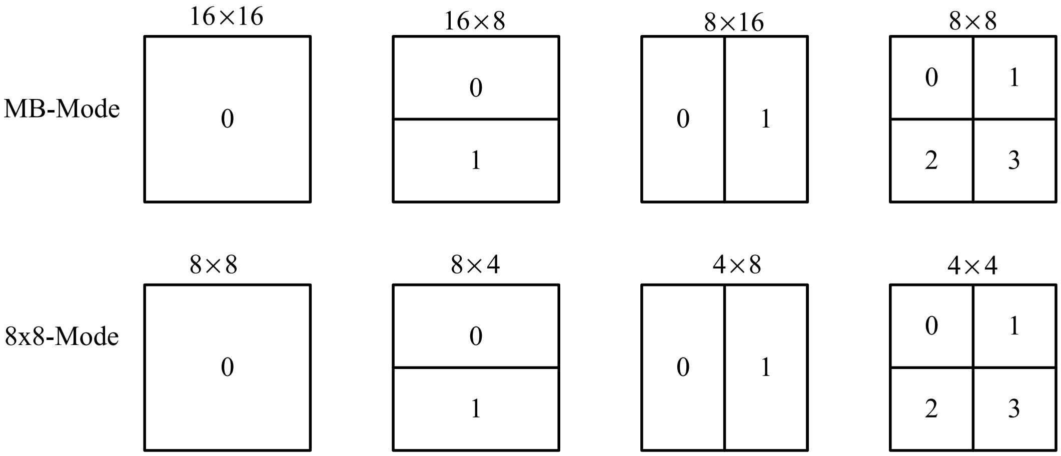


Fig. 1.11. Block modes defined in H.264.

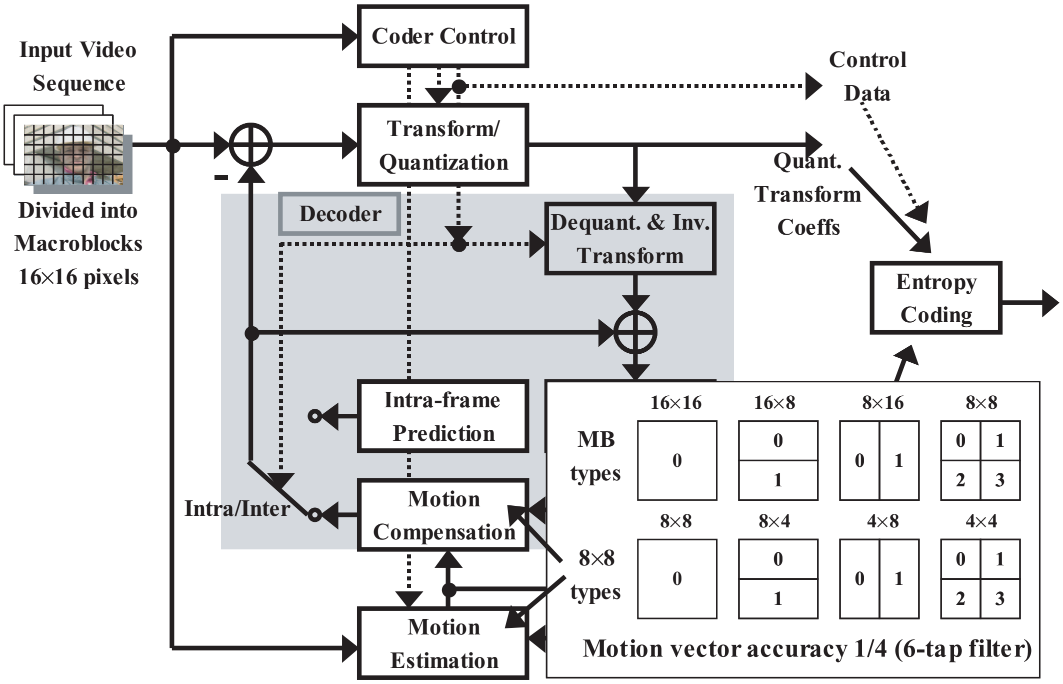


Fig. 1.12. Motion compensation accuracy in H.264.

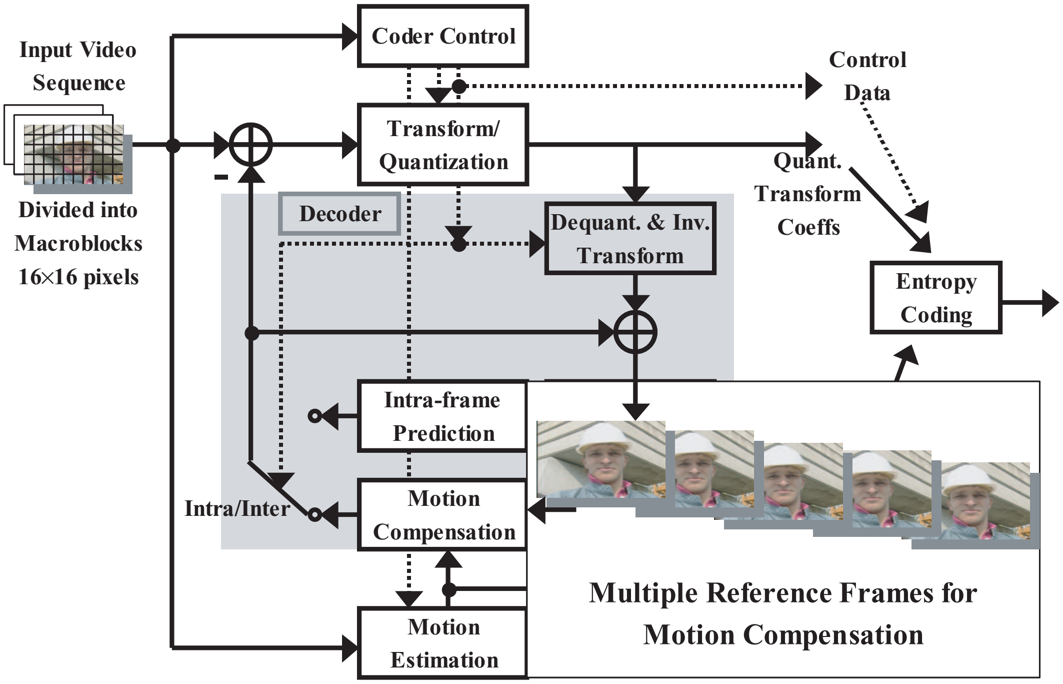


Fig. 1.13. Multiple reference motion search in H.264.

1.4.3 Other Applications of Motion Estimation

Outside of the standards, there have been considerable efforts to improve the quality of motion estimation.

Global Motion Estimation and Sprite Coding

Sprite coding [22], also known as mosaic rendering, is a very effective method to represent and compress a background video object. Fig. 1.14 shows the four basic modules in spirit coding: Global motion estimation(GME) [23–26], image segmentation, image warping and image blending. Global motion is generally related to camera motion such as panning, tilting and zooming, which can be modeled with a parametric geometrical model. To generate a sprite, the GME estimates the global motion parameters between current frame and the sprite. For each frame to be coded, an estimation is made to predict the position of the background. The location parameters and the difference are transmitted to the decoder. And a new estimation is made for the area of background to update the sprite.

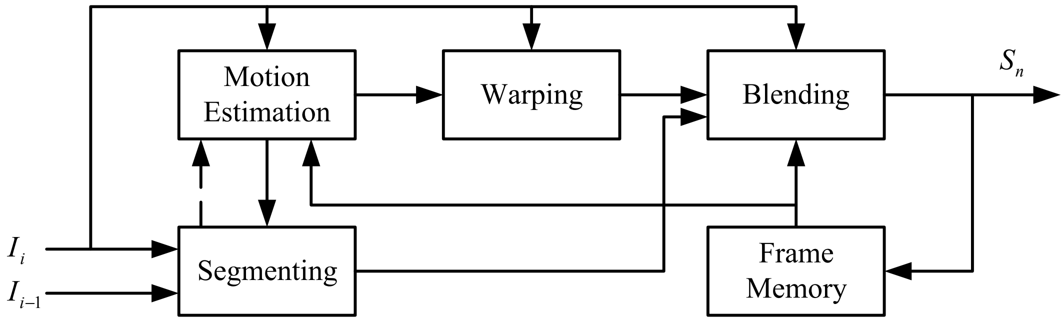


Fig. 1.14. Global motion estimation and Sprite coding.

Progressively Scalable Motion Estimation

In this Section, we discuss one-loop and two-loop motion estimation. As shown in Fig. 1.8, SNR scalable coding uses one-loop motion estimation. This open-loop enhancement layer structure does not use any information from the enhancement layer for the reference and hence is immune to drift error [27]. A drift error occurs when the encoder and decoder use different motion information. This error is further propagated along the sequence until an INTRA frame is used. Despite the drift-free advantage for one-loop design, it is found that the open-loop enhancement layer design may suffer up to 3 dB in PSNR performance loss due to the inferior quality of the reference [28, 29]. The close-loop enhancement layer structure, or two-loop structure, is proposed to address this problem and improve the coding efficiency [28]. As shown in Fig. 1.15, part of the information in the enhancement layer is used to construct the reference. To accommodate rate fluctuation, the base layer and enhancement layer are used as the reference alternately, as shown in Fig. 1.16. This structure is used in PFGS [29] and can achieve a coding gain of 2-3 dB over FGS.

Leaky Prediction

Leaky prediction was proposed to address the drift error in the two-loop motion estimation [30, 31]. As mentioned above, a drift error occurs when the encoder and decoder do not have the same reference information. This can happen when there is error in the transmission channel or the channel is not adequate to accommodate the reference data. The basic idea of leaky prediction comes from the leaky predictor in DPCM [32, 33]. By using only a fraction of the reference frame, the prediction is attenuated by a leaky factor of value between zero and unity. Leaky prediction is a trade-off between error resilience and coding efficiency. Only a fraction of the enhancement layer is used for the reference. This reduces the coding efficiency but improves the drift error robustness. If the enhancement layer information is not available at the decoder, the drift error will be attenuated exponentially across the frames. By varying the leaky factor, the method can model the FGS or PFGS readily [31]. The coding efficiency of leaky prediction lies between FGS and PFGS. An example of motion search with leaky prediction is shown in Fig. 1.17.

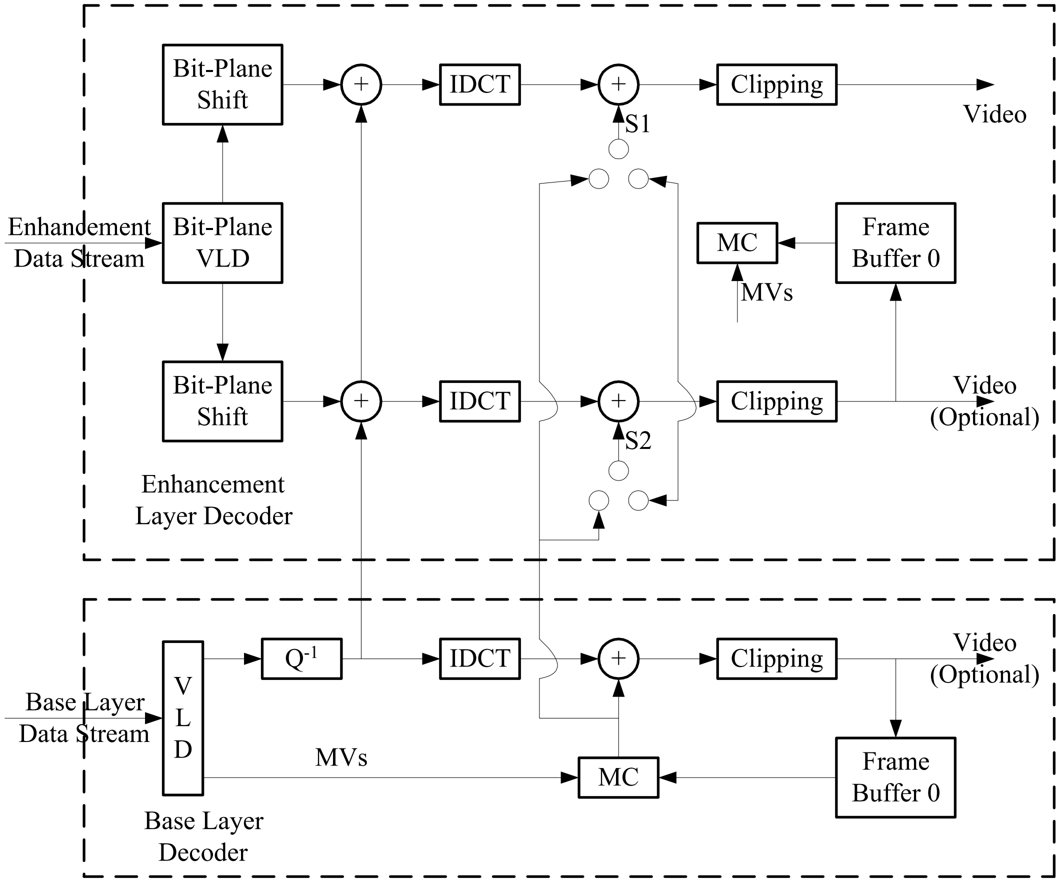


Fig. 1.15. Video decoder with two-loop motion estimation.

Multiple Description

Multiple description (MD) motion esitmation is used to improve error resilience [34,35]. It ensures the reconstruction of video sequences with only part of the stream, referred to as a description. An MD motion estimator is shown in Fig. 1.18. The basic idea is to use both descriptions to form a new and more accurate reference for both descriptions, only the mismatch is transmitted.

Long-Term Memory and Multi-Hypothesis Motion Estimation

In P-frame motion estimation, only one previous reconstructed frame is used as the reference. However, the video content may not be always more correlated with

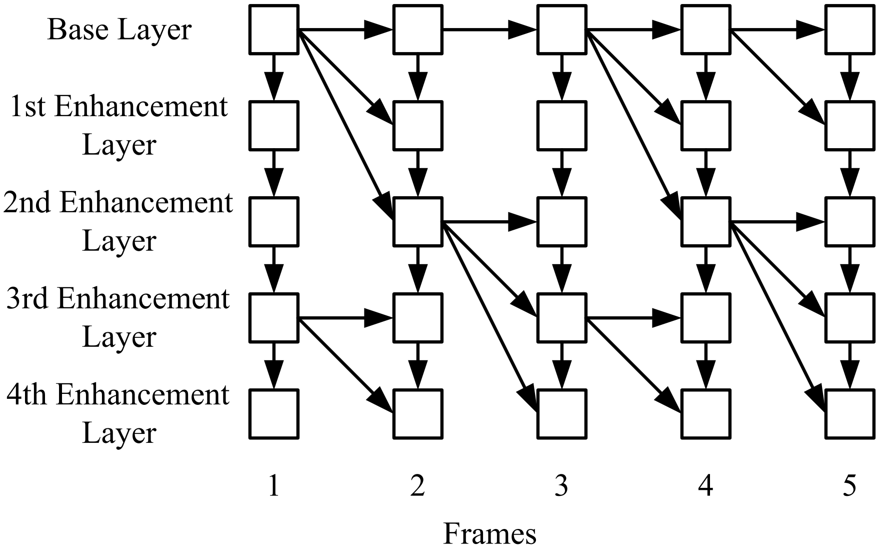


Fig. 1.16. Coding mode in two-loop motion estimation.

the previous frame than other frames. B-frames takes advantage of this feature and includes the next frame in the references [36]. The coding efficiency is significantly improved with B-frame motion estimation. It is shown in [37–39] that by using multiple hypothesis in previous frames and enforcing a rate-distortion optimized selection coding efficiency can be improved. A hypothesis is referred to as one candidate reference information, which can either be a frame or a coding unit such as a block or macro-block. To address the temporal selection of hypothesis, the motion vector overhead increases. However, the improvement in reference accuracy, which results in the reduction in the entropy of the residual frame, may offset this cost and provide additional coding efficiency gains. An illustration of motion search with three hypotheses is shown in Fig. 1.19.

Estimation Theoretic Motion Estimation

Most of the previous motion estimation methods use the reconstructed frame directly, or part of the reconstructed frame (as in FGS), or a linear combination of the reconstructed information as the reference. Despite the simplicity, there is no

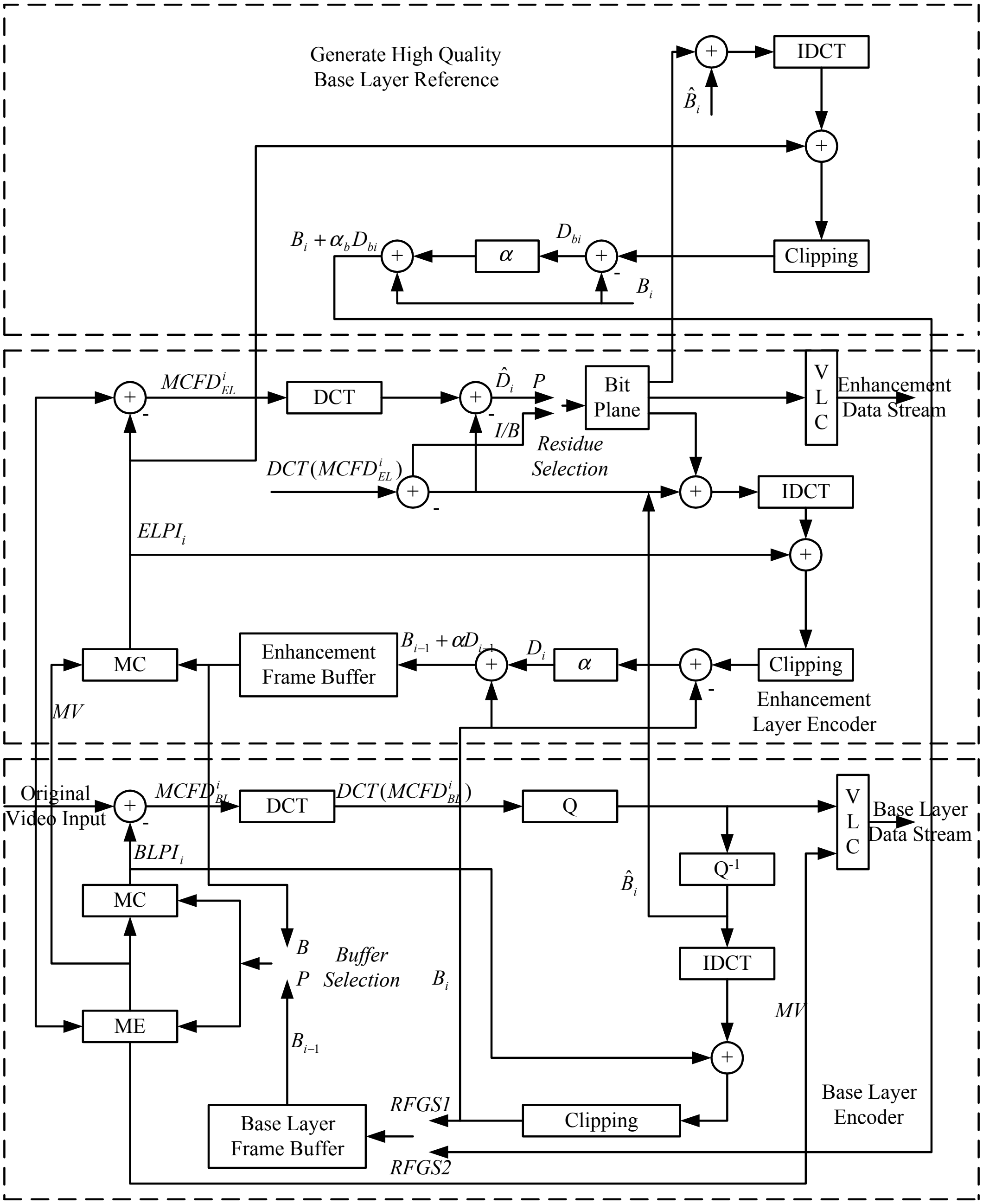


Fig. 1.17. Leaky prediction.

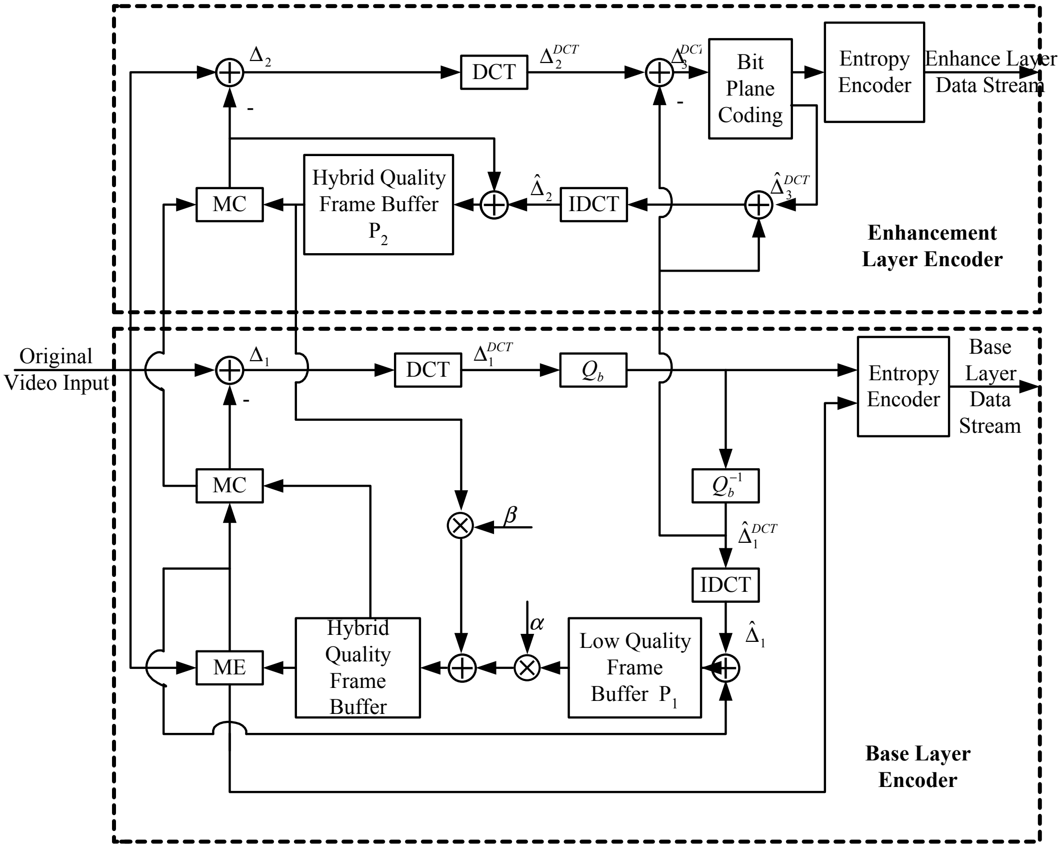


Fig. 1.18. Multiple description motion estimation.

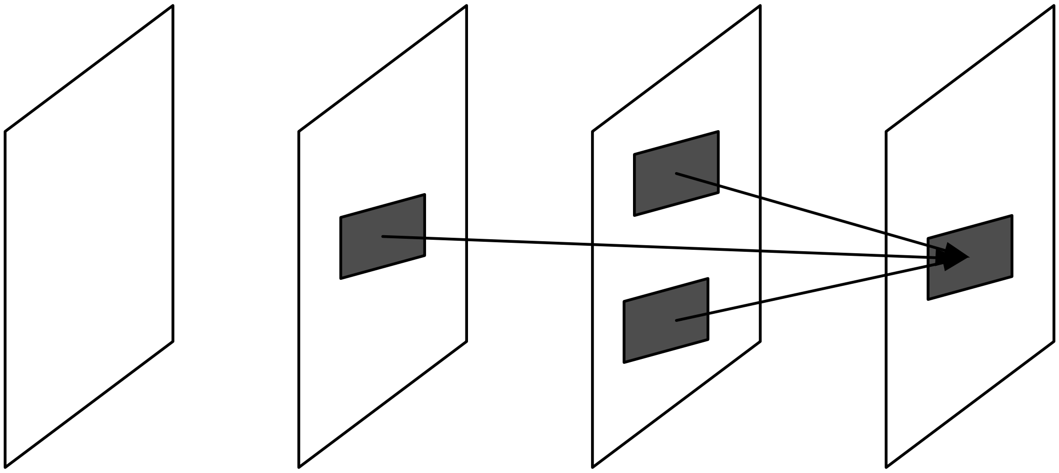


Fig. 1.19. Multi-hypothesis motion estimation and long-term memory motion estimation.

guarantee that such usage of the reconstructed frames would lead to the optimal reference selection. In [40], the motion estimation problem was formulated as an estimation problem based on the received previous reconstructed frames to construct the optimal reference in the minimum mean square error(MMSE) sense. A simple illustration is shown in Fig. 1.20. It is shown in [40] that by optimally estimating the reference in the encoder and decoder, approximately 0.5-1.5 dB performance gain can be achieved.

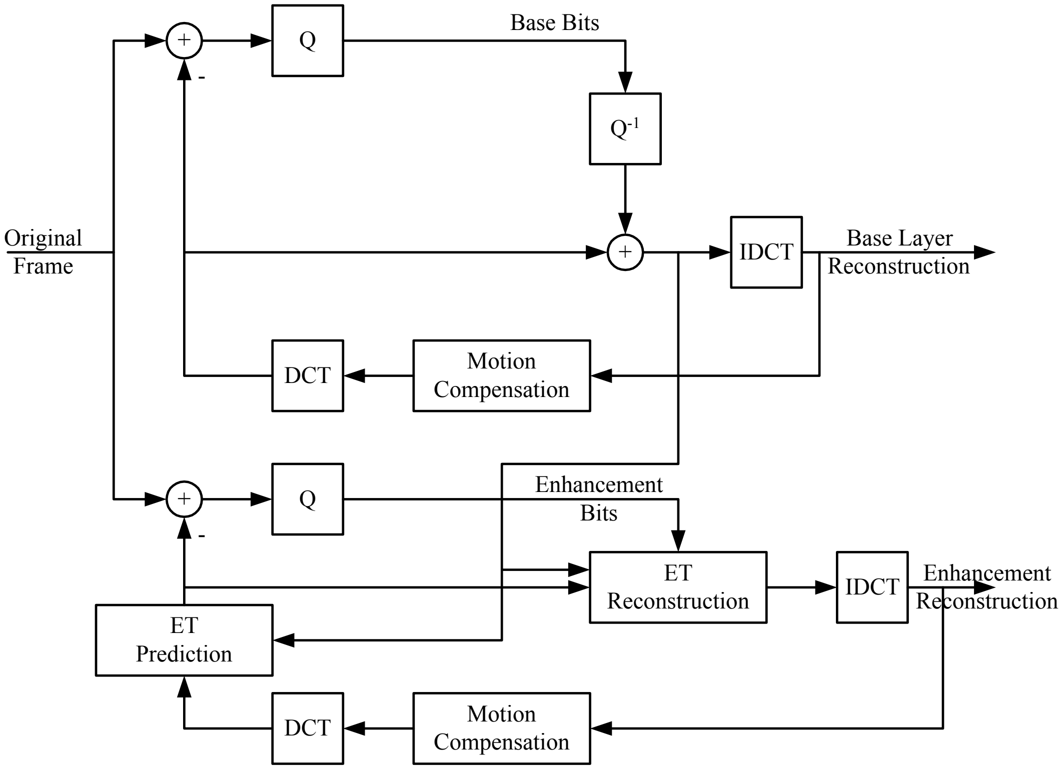


Fig. 1.20. Estimation theoretic motion estimation.

1.4.4 Rate Distortion Analysis of Motion Estimation

Rate distortion analysis for source coding was pioneered in Shannon's work [15]. It was then extended to analyze the performance of still image coding and video coding. An analysis of rate distortion performance of motion estimation for conventional motion compensated prediction (MCP)-based video coding was described in [41–45]. In this research, signal power spectrum and Fourier analysis tools are used to analyze motion estimation. These ideas were used to analyze MCP-based scalable video coding [46–49], multiple descriptions coding and leaky prediction [21, 50–52]. In our work, we extend these methods to study the video coding loss for a low complexity Wyner-Ziv video coder.

1.5 Low Complexity Video Coding

Current video coding standards are highly asymmetrical in terms of complexity. Encoding is typically 5-10 times more complex than decoding. This is due to the use of inter-frame predictive coding, which is desirable for consumer electronics (CE) applications including DVD and DTV (Digital Television), video streaming, and video-on-demand.

New video coding applications have emerged such as wireless sensor networks, wireless PC cameras, and mobile video cameras for video surveillance as shown in Fig. 1.21. A common characteristic of these applications is that resources for memory, computation, and energy are scarce at the video encoder. Since conventional video coding methods cannot meet the requirements of these new emerging applications, it is necessary to develop alternative video coding methods that have low complexity at the encoder. We refer to such video coding methods as low complexity video encoding. One way to implement this is to replace an inter-frame encoder-decoder, which uses both inter-frame encoding and inter-frame decoding as in the case of the conventional video coding, with an intra-frame encoder but inter-frame decoder. This implies that the implementation of the motion estimation is shifted from the encoder to the decoder, and the motion vectors are only used by the decoder.

In conventional video source coding, the encoder usually uses extra information, statistical or syntactic, to encode the source. This extra information is known as "side information." Side information may be supplied to the encoder, or estimated

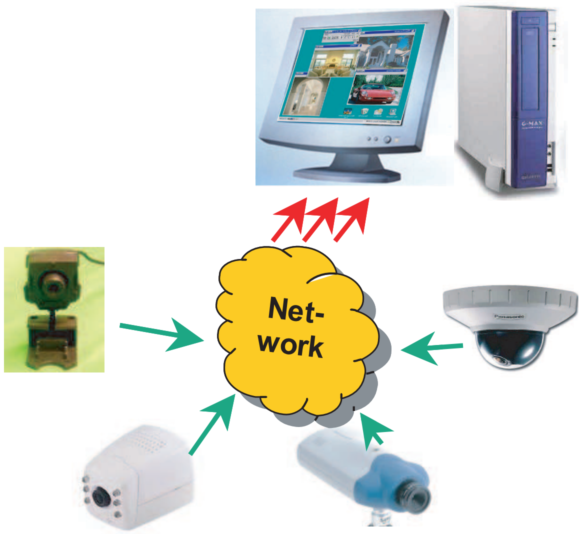


Fig. 1.21. A video surveillance system with distributed cameras.

by the encoder. A classical example of side information in conventional video coding is motion information characterized by motion vectors. Hence, the side information is known to both the encoder and the decoder for conventional video coding. In contrast, low complexity video encoding may require that the side information be known only to the decoder. Therefore, how one obtains the side information at the decoder, instead of at the encoder, is the most essential problem to be addressed for low complexity encoding.

Wyner-Ziv video coding, also known as distributed video coding, has provided a coding paradigm that allows side information to be used only at the decoder. Given two arbitrary sources, one source may serve as the side information to the other. Using Wyner-Ziv video coding, the two sources can be encoded independently but decoded jointly. The ideal situation for Wyner-Ziv video coding is to achieve a coding efficiency the same as that of using joint encoding and joint decoding, i.e., the same as that of conventional source coding.

1.6 Video Post Processing

Transform coding methods have proved very effective in image and video compression. The block-based discrete cosine transform (BDCT) is widely used due to its similarity with the Karhunen-Loeve transform (KLT) and availability of fast implementation algorithms.

However, BDCT introduces the block artifact, especially at low data rates. A block artifact manifests itself as an annoying discontinuity between adjacent blocks. This is a direct result of independent transform and quantization of each block, which fails to take into account the inter-block correlation. Fig. 1.22 shows an image with block artifacts.

A great deal of research have been done to reduce the block artifact. There are basically two approaches. One treats the artifact at the encoder, such as the in-loop filtering in H.264 [53]. The other one uses post processing at the decoder side. The post processing approaches have been actively investigated since it maintains the compatibility with the original decoder structure.

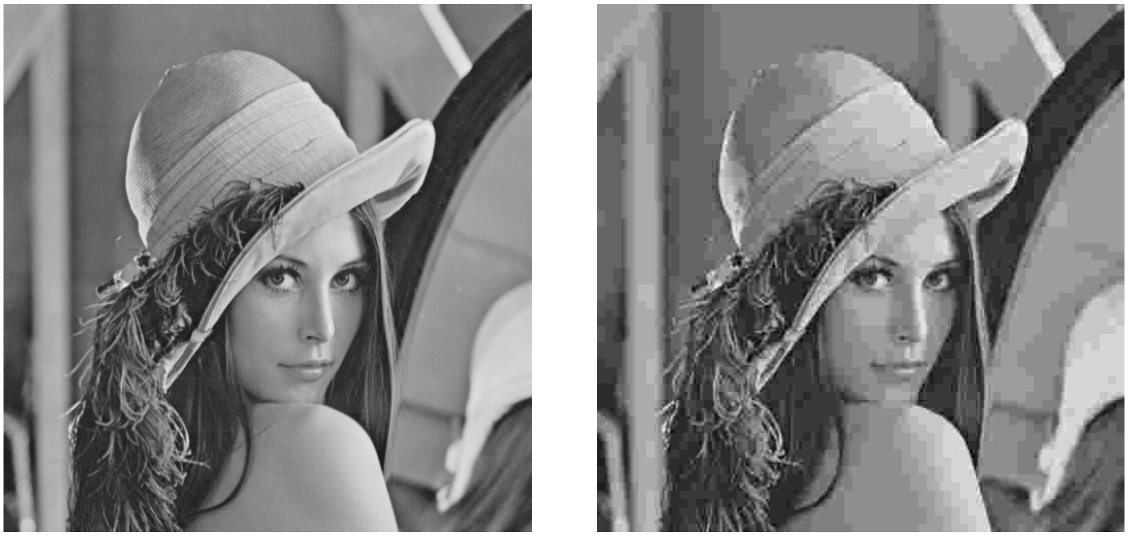


Fig. 1.22. An example of block artifact.

1.6.1 Spatial Domain Post Processing

Post Filtering

Since block artifacts manifest themselves as discontinuities across block boundaries, a low-pass filter can be used to remove these high frequency artifacts. In H.264, the post filter [54], is used across the macro-block boundaries. This method is quite simple in implementation and relatively effective. An example of the filtering defined in H.264 is shown in Fig. 1.23.

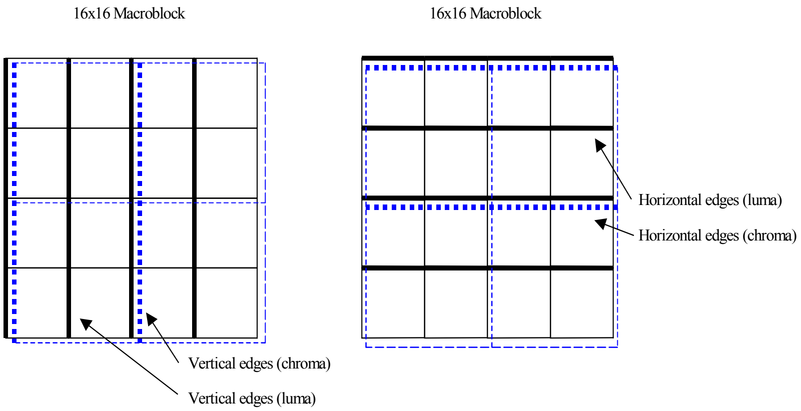


Fig. 1.23. Low-pass post filtering.

POCS

In post filtering methods, the filtered DCT transform coefficients may fall outside of the Quantization Constraint Set(QCS) [55], this can result in over-blurred frames. Many techniques use constrained iterative methods to avoid the violation of QCS. Post-processed images are projected back to the transform domain to examine if the DCT coefficients fall inside the QCS. If any coefficient is found outside of the QCS, either clipping or other measures are used to fix the problem. The transform coefficients are then projected back to the spatial domain using the inverse transform for the next iteration. This leads to a post processing method known as Projections

On Convex Sets (POCS) [55,56]. The general idea behind POCS-based methods is to find several close constraint convex sets, one of which being the QCS, to enforce the convergence objective iteratively. The block diagram of POCS-based methods is illustrated in Fig. 1.24.

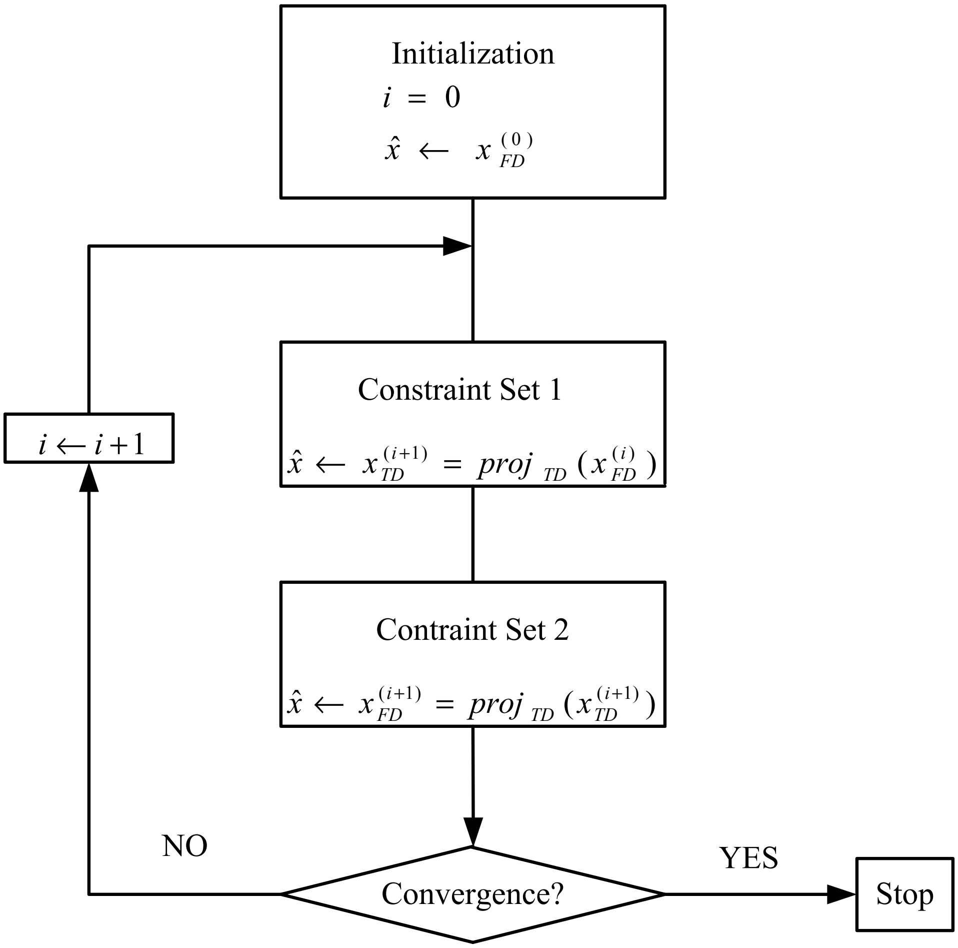

Initialization
$i = 0$
$\hat{x} \leftarrow x_{FD}^{(0)}$
Constraint Set 1
$\hat{x} \leftarrow x_{TD}^{(i+1)} = proj_{TD}(x_{FD}^{(i)})$
Contraint Set 2
$\hat{x} \leftarrow x_{FD}^{(i+1)} = proj_{TD}(x_{TD}^{(i+1)})$
$i \leftarrow i+1$
NO
Convergence?
YES
Stop

Fig. 1.24. POCS post processing.

Over-Complete Wavelet Representation

An over-complete wavelet representation can also be used to reduce the quantization error of BDCT [57]. An example is given in Fig. 1.25. The basic idea is to view the deblocking problem as one that introduces quantization noise in the compressed image. The denoising property of the wavelet transform is then used. The deblocking problem then becomes the problem of smoothing the discontinuities across block boundaries only in smooth background regions but protecting edges that may occur at block boundaries.

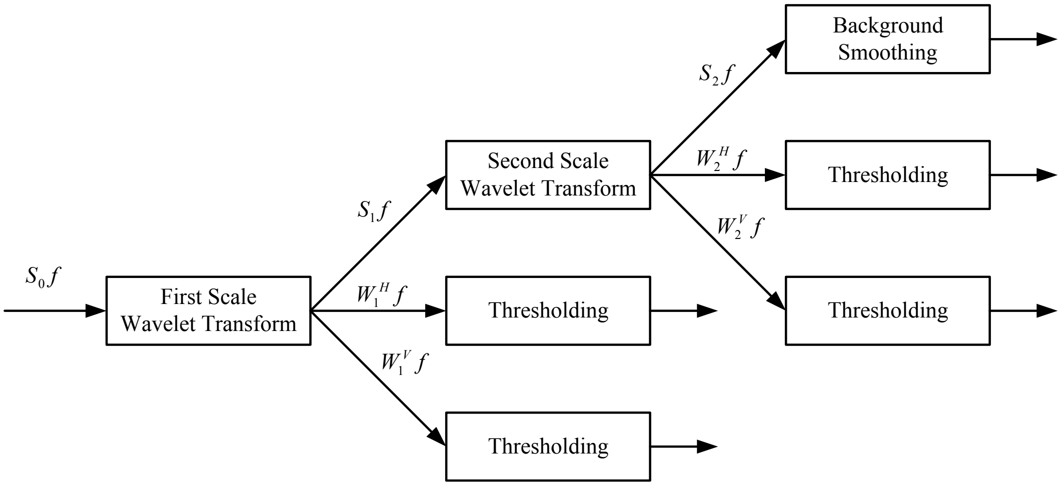


Fig. 1.25. Over-complete wavelet post processing.

MAP Formulation

Another post processing method uses statistical properties of the image. Block artifact reduction is modeled as a maximum *a posteriori* (MAP) estimation problem [58–60], where the image is modeled as a Markov Random Field (MRF), i.e., assume

$$Y = X + N \tag{1.1}$$

where X is the original image, Y is the observed image and N is the noise. Then the MAP estimator is given by

$$\hat{X} = \arg\max_X \{\log P(Y|X) + \log P(X)\} \tag{1.2}$$

And the image X is modeled as a Markov Random Field represented by a Gibbs distribution

$$P(x) = \frac{1}{Z}\exp\{-\sum_{c\in C} V_c(x_c)\} \tag{1.3}$$

where x_c is the value of X at the points in clique C, $V_c(x_c)$ is the potential function of x_c, C is the set of all cliques and Z is the normalizing constant.

The MAP solution for deblocking problems becomes a constrained optimization problem subject to the QCS constraint and the Markov Random Field properties.

1.6.2 Transform Domain Post Processing

Relatively few research has been done in transform domain post processing. In [61], a new metric, referred to as the mean square difference of slope (MSDS), is defined. MSDS is “the square of the difference between the slope across two adjacent blocks, and the average between the slopes of each of the two blocks close to their boundaries” as shown in Fig. 1.26.

$$\begin{aligned} MSDS &= \sum_k \Big([x_{i,j}(k,0) - x_{i,j-1}(k,N-1)] \\ &- \frac{[x_{i,j-1}(k,N-1) - x_{i,j-1}(k,N-2)] + [x_{i,j}(k,1) - x_{i,j}(k,0)]}{2} \Big)^2 \end{aligned} \quad (1.4)$$

Hence the block artifact reduction problem is formulated as an MSDS minimization problem subject to the QCS. This method is followed in [62, 63], where new forms of MSDS and minimization methods were proposed.

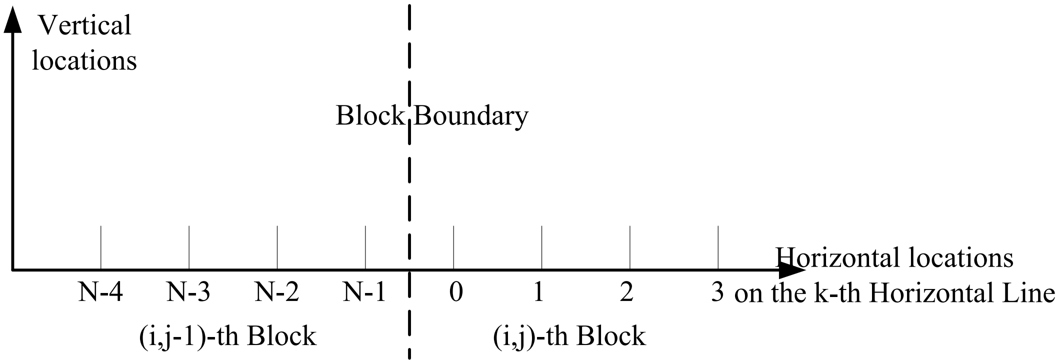


Fig. 1.26. MSDS post processing.

1.7 Contributions of This Thesis

In this thesis, we studied low complexity video compression and post processing [50, 52, 64–77]. The main contributions of this thesis, as described in later chapters, are:

- **Rate Distortion Analysis of Wyner-Ziv Video Coding**

  We addressed the rate-distortion performance of Wyner-Ziv video coding. A theoretic analysis of Wyner-Ziv video coding was developed. We studied the Wyner-Ziv video coding performance and compared it with conventional motion-compensated prediction (MCP) based video coding. We showed that theoretically Wyner-Ziv video coding may outperform by up to 0.9bit/sample DPCM-frame video coding. For most practical sequences with low motion correlation across the sequences, the rate saving is less significant. When compared with INTER-frame video coding, Wyner-Ziv video coding leads to a rate increase of as much as 0.2-0.6bit/sample for sequences with small motion vector variances and 1bit/sample for those with large motion vector variances. The results show that the simple side estimator cannot meet the expectation of practical applications and more effective side estimation is needed.

- **Conventional Motion Search Methods Revisited**

  We studied the use of conventional motion search methods for side estimators. The results showed that the use of sub-pixel motion search does not affect the coding efficiency as much as it does in conventional video coding. Also the use of sub-sampling motion search only results in small coding efficiency loss with Wyner-Ziv video coding. This indicates that for side estimators that cannot locate the rough position of the true motion vectors in the first place, it makes little sense to further refine the motion vector by using sub-pixel motion search. For decoders with limited computing capability, a 2:1 or even coarser subsampling is worth consideration. We also showed that the use of multi-reference motion search can effectively improve coding efficiency. Side estimators with two reference frames can outperform those with one reference frame by 0.2bit/sample or more. More data rate reduction can be achieved with more reference frames.

- **Wyner-Ziv Video Coding with a Refined Side Estimator**

  We presented a new Wyner-Ziv video decoder with a refined side estimator. The goal of this new decoder is to better exploit the motion correlation in the reconstructed video sequence and hence improve the coding efficiency. The idea of refined side estimator is to progressively extract side information from both previous reconstructed frames and the current partially decoded Wyner-Ziv frame. The experimental results showed that our design can improve the coding quality by as much as 2dB in PSNR.

- **Wyner-Ziv Video Coding with Universal Side Estimator**

  We proposed a novel decoding structure with a universal prediction side estimator. The goal of this decoder is to reduce the decoding complexity at the Wyner-Ziv decoder and hence make it possible to design an encoder and decoder with low complexity. This new side estimator uses a non motion-search method to construct the initial estimate at the Wyner-Ziv video decoder. The test results show that for sequences in which the motion can be predicted with previous frames, the universal prediction side estimator is 2-3dB lower than the conventional MCP-based motion side estimators in terms of coding efficiency. However, for other sequences, the coding efficiency is rather close and sometimes universal prediction side estimator even outperforms the MCP-based side estimator. We also showed that this new method can significantly reduce the coding complexity at the decoder.

- **Block Artifact Reduction Using A Transform Domain Markov Random Field Model**

Lossy image and video compression is often accompanied with annoying artifacts. Post processing methods are used to reduce or eliminate the artifacts. Many post processing methods are done in the spatial domain. Since human visual system is not as sensitive to the artifacts in the high frequency bands as to those in the low frequency bands, spatial domain post processing methods may unnecessarily process those artifacts in the high frequency bands and hence lead to high computational complexity. We presented a transform domain Markov Random Field model to address the artifact reduction. We presented two methods, namely TD-MRF and TSD-MRF, based on this model. We showed by objective and subjective comparisons that transform domain post processing method can substantially relax the computational complexity constraint compared with spatial domain method and still achieve significant block artifact reduction. We note that TD-MRF generally cannot achieve as much block artifact reduction as SD-MRF. This is because in highly quantized images, almost all high frequency coefficients are truncated to zero. Hence, loss incurred to high frequency coefficients cannot be fully recaptured if subbands are processed separately as in TD-MRF. Meanwhile, the TSD-MRF framework that combines the TD-MRF and SD-MRF leads to more significant block artifact reduction without over-blurring the image.

2. RATE DISTORTION ANALYSIS OF WYNER-ZIV VIDEO CODING

2.1 Introduction

Wyner-Ziv video coding, also known as distributed video coding, is a new video lossy compression method with side information available only to the decoder but not at the encoder. As new as it might appear, the basic principle behind Wyner-Ziv video coding dates back at least three decades. The information-theoretic work for lossless coding with side information available only to the decoder was described in [78] in 1973 and is referred to as Slepian-Wolf coding. It was recognized that there is a connection between Slepian-Wolf coding and channel codes [79]. The result was extended to lossy compression by Wyner and Ziv [80] in 1976. Despite the initial excitement of the theoretical insight, "the conceptual importance of Slepian-Wolf coding has not been mirrored in practical data compression. Not much progress on constructive Selpian-Wolf schemes has been achieved beyond the connection with error-correcting channel codes" [81] in the following twenty-five years. This is mainly due to the inability of finding a channel code efficient enough to approach the performance claimed in [78] and [80]. When a code is theoretically efficient enough [82], the high decoding complexity is, more often than not, beyond practical considerations.

The latest efforts towards the development of practical Wyner-Ziv coding were accompanied by the advance in high-performance computing and channel coding. Several video lossy compression schemes based on Wyner-Ziv coding have been presented with new applications being discussed. The last twenty-five years also witnessed advances in video coding. The enormous success of conventional video coding, ranging from the various video coding standards, including H.261/3/4 [8, 11, 53] and

MPEG-1/2/4 [6,9,10], to numerous research, has made it possible to deliver video at a reasonable quality and acceptable cost to users in many applications. This has also posed a challenge to any emerging video codec design. A new video codec generally has to meet certain expectations with respect to coding efficiency before receiving serious consideration. Intuitively, given the lack of side information at the encoder, it is acknowledged that "it is unlikely that a distributed video coding algorithm will ever beat conventional video coding schemes in rate-distortion performance, although they might come close " [83]. A quantitative evaluation of Wyner-Ziv video coding performance compared to conventional video coding has not been thoroughly discussed in the literature. It is of interest to consider the following questions,

- How close can Wyner-Ziv video coding come to conventional video coding in terms of rate-distortion performance?
- Will conventional motion search methods improve Wyner-Ziv video coding efficiency as well?
- How can one design an efficient Wyner-Ziv video coder?

In this Chapter we try to address these questions and provide a definitive answer. We evaluate the video coding performance difference between Wyner-Ziv video coding and conventional video coding, in particular H.264, which is regarded as one of the most efficient video coding methods. Both analytical and simulation results are provided. We also investigate the use of sub-pixel and multi-reference motion search methods, both of which significantly improve conventional MCP-based video coding efficiency. By addressing these two questions, our goal is to find not only the performance gap between Wyner-Ziv video coding and conventional video coding but also how to improve Wyner-Ziv video coding efficiency. A new Wyner-Ziv decoding scheme, referred to as Wyner-Ziv video coder with a Refined Side Estimator (WZ-RSE), is presented. WZ-RSE can more effectively extract the side information from the reconstructed information at the decoder. Experimental results show that WZ-

RSE can consistently have a gain of up to 2dB over the Wyner-Ziv video decoders without RSE.

The rest of this Chapter is organized as follows. In Section 2.2 we give an introduction to Wyner-Ziv video coding. In Section 2.3 we formulate the problem and examine the difference between Wyner-Ziv video coding and conventional MCP-based video coding. The use of sub-pixel and multi-reference motion search methods is addressed in Section 2.4. Our new Wyner-Ziv video decoder WZ-RSE is presented in Section 2.5 with experimental results.

2.2 Wyner-Ziv Video Coding

The following problem is considered in [78]. Consider two correlated information sources X and Y, as shown in Fig. 2.1, encoded by two separate encoders A and B, neither has the access to the other source. A rate R is *achievable* if for any $\epsilon > 0$, there exists an encoder and decoder such that $Pr(\hat{X} \neq X) < \epsilon$. If joint decoding is allowed, the Wolf-Slepian Theorem says, by a random coding argument, that the achievable rate region for the system in Fig. 2.1 is

$$R_X \geq H(X|Y), R_Y \geq H(Y), R_X + R_Y \geq H(X,Y) \tag{2.1}$$

Hence, regardless of its access to side information Y, encoder A can encode X with arbitrarily high fidelity as long as the decoder A has access to Y and the rate is equal to or larger than $H(X|Y)$.

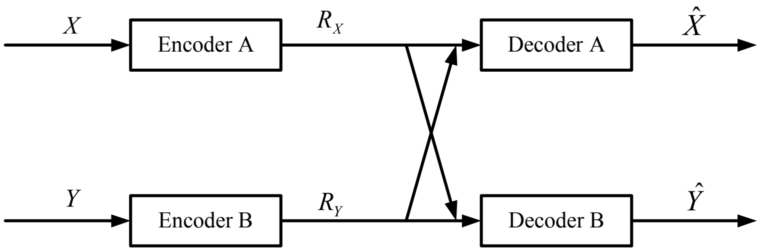


Fig. 2.1. Slepian-Wolf coding.

This result is then extended to lossy compression in [80]. Denote $R^*(d)$ as the rate distortion function of X when the side information Y is only available at the decoder, and $R_{X|Y}(d)$ is the rate distortion function when side information Y is available at both the encoder and decoder. In [80] it is shown that although $R^*(d) \geq R_{X|Y}(d)$, in certain cases the equality can be achieved, e.g. for a Gaussian source and a mean square error distortion metric. Hence, the side information at the encoder is not always necessary to achieve the rate distortion bound in lossy compression.

We now consider the implication of the above results to video coding. Many of the current video coding structures involve the use of motion-compensated hybrid video coding, as shown in Fig. 2.2. Such a coding scheme is generally referred to as motion-compensated prediction(MCP) based video coding. It is used in most video coding standards. In this scheme, each frame is decoded and reconstructed at the encoder, which is then stored in the frame buffer and used to construct a reference for the encoding of the next frame. At the decoder, each decoded frame is also stored in the frame buffer and used to construct the reference for the decoding of

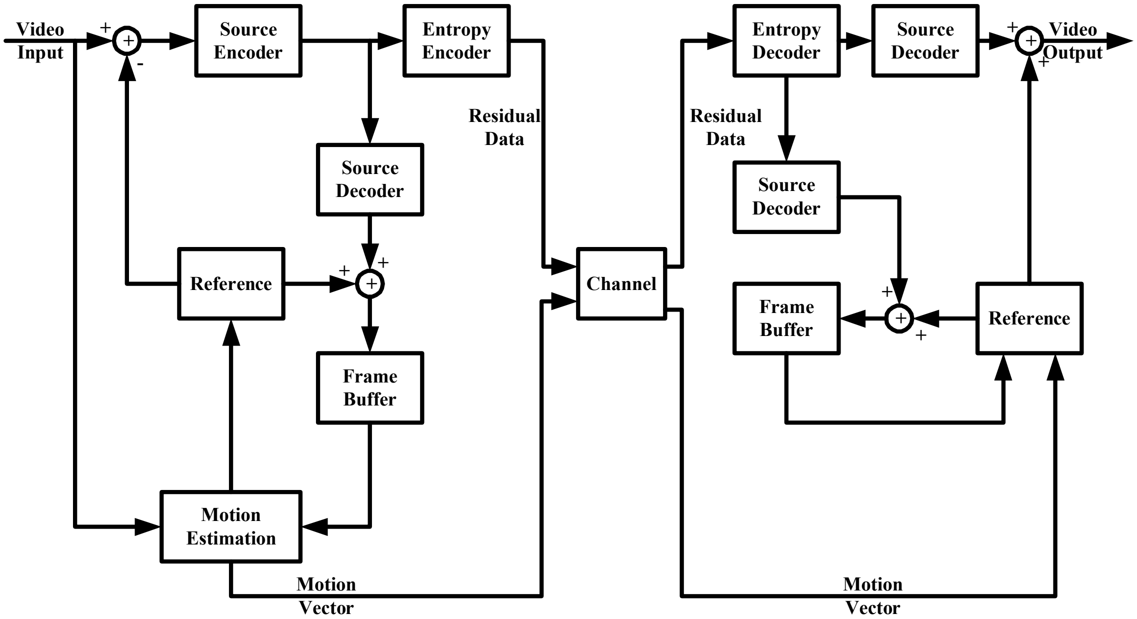


Fig. 2.2. Conventional MCP-based video coding.

the next frame. As long as the frame buffers store the same reconstructed frames at the encoder and decoder, and the motion vectors are correctly transmitted to the decoder, it is guaranteed that both the encoder and decoder have the same reference information. The reference information in MCP-based video coding can be regarded as the side information for the next frame to be coded. Analogous to the system in Fig. 2.1, we can simplify Fig. 2.2 to the Fig. 2.3. The reconstructed frames serve as the source of the side information. The side information is processed by a side estimator g to generate the reference. In conventional MCP-based video coding, the switch K is ON and both the encoder and decoder have access to this reference information. The Slepian-Wolf Theorem and the Wyner-Ziv Theorem suggest that the reference information at the encoder is not necessary to achieve or approach the rate distortion bound. This leads to a substantially different coding structure, referred to as Wyner-Ziv video coding, in that source statistics are only exploited at the decoder. This structure makes it possible for many new applications previously considered difficult with conventional video coding. For instance, in conventional video coding the complexity-intensive motion estimation is done at the encoder. This is not convenient for applications that require simple encoding complexity such as video surveillances or video camera phones. By exploiting the source statistics only at the decoder, the Wyner-Ziv video coder allows the motion search work to shifted to the decoder, where more computational resources are available. With this among

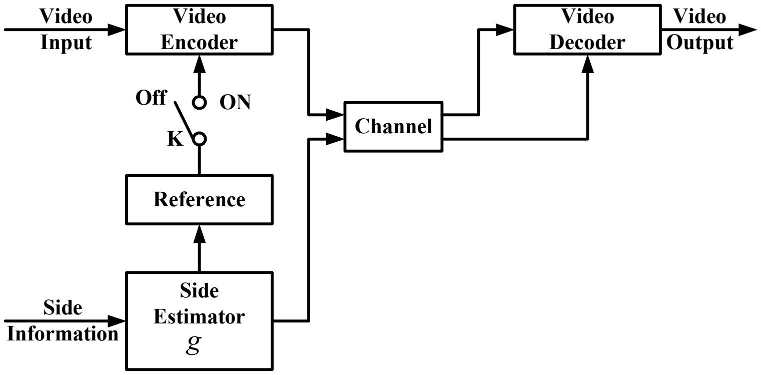


Fig. 2.3. Simplified conventional video coding scheme.

many other advantages, it may sound unusual that Wyner-Ziv video coding had not got much attention until quite recently. The main reason, as pointed out in the literature [83–85], is rather practical. Like many other important information theory discoveries, non-constructive reasoning was used to prove the theoretical results and the practical construction of a Wyner-Ziv coder is very difficult. There are very few channel codes efficient enough to realize the theoretical results claimed in [78] and [80]. Even for codes that can achieve the performance theoretically [82], the decoding complexity is still high enough to frustrate those who attempt to use them.

The advance of high-performance computing in the last two decades and the development of new channel codes, especially the reexamination of low-density parity-check(LDPC) [82] codes and the emergence of turbo codes [86,87], have finally made a Wyner-Ziv coder feasible and promising. Much research has been done in recent years to implement a practical Wyner-Ziv coder. To name a few, a high dimensional lattice code was used in [88] to approach the Wyner-Ziv bound. The use of algebraic channel coding is investigated in [84,89,90]. A nested lattice quantizer with an LDPC Slepian-Wolf coder is developed in [91–93]. Turbo codes are used in [94–96]. Several new applications have also been developed with Wyner-Ziv video coding. A transform domain Wyner-Ziv video coder is proposed in [97]. Sensor network applications were considered in [85] and [98]. Error resilient properties of Wyner-Ziv video coding are discussed in [99, 100]. Interested readers are referred to [83] for a detailed discussion of the current development.

In this work, we focus our discussion of Wyner-Ziv video coding efficiency in terms of rate distortion performance. Despite the mathematical elegance, information-theoretic insight, and potential promising applications, the coding efficiency remains one of the top concerns for any video coding method. To better understand and evaluate this video coding method, it would be helpful to know the performance gap between Wyner-Ziv video coding and conventional video coding. It would also be helpful to understand how conventional video compression techniques work for Wyner-Ziv video coding, and more importantly, how to design a more efficient Wyner-Ziv video coder.

2.3 Wyner-Ziv Video Coding vs. Conventional Video Coding

2.3.1 System Description

We consider the system in Fig. 2.3 again. As indicated by the Wyner-Ziv Theorem,

$$R^*(d) \geq R_{X|Y}(d) \tag{2.2}$$

Hence, the rate distortion performance of X by Wyner-Ziv video coding is lower bounded by conventional video coding with the reference Y available at the encoder. Here we introduce two terms, *the source of side information* and *the reference.* Although the Wyner-Ziv Theorem remains valid independent of the side estimator g, in a practical Wyner-Ziv video coder, the reference Y, which is obtained by applying g to the source of side information S, is the actual side information available to the decoder. To compare the performance difference between Wyner-Ziv video coding and conventional MCP-based video coding, we need to keep in mind that the performance difference is actually caused by the following three terms:

- *System loss* $\Delta R_s(d)$. System loss is the inherent loss due to the lack of side information S at the encoder. As proved in the Wyner-Ziv theorem, $R^*(d) \geq R_{X|S}(d)$, then system loss is

$$\Delta R_s(d) = R^*(d) - R_{X|S}(d) \tag{2.3}$$

- *Source coding loss* $\Delta R_c(d)$. Source coding loss is introduced due to the inefficiency of the channel code and quantization scheme used in the system. Unless a channel code C can achieve the Shannon limit [15], any Wyner-Ziv video coder using C will incur a loss. We denote the real rate as $R'(d)$, then the source coding loss

$$\Delta R_c(d) = R'(d) - R^*(d) \tag{2.4}$$

- *Video coding loss* $\Delta R_v(d)$. Video coding loss results from the loss in reference information. Denote the rate distortion function as $R_{X|S}(d)$ if the side information S is ideally exploited by the decoder, while $R_{X|Y}$ as the rate distortion function when Y is instead used. This apparently changes the upper bound of $R^*(d)$ from $R_{X|S}(d)$ to $R_{X|Y}(d)$. This loss is denoted as video coding loss

$$\Delta R_v(d) = R_{X|Y}(d) - R_{X|S}(d) \tag{2.5}$$

The system loss $\Delta R_s(d)$ has been studied in [101], in which it is shown that the loss can be up to 0.5bit/sample for general source statistics and a mean square error distortion metric. The source coding loss $\Delta R_c(d)$ depends on the specific channel code and quantization scheme used, which we would rather leave to the channel coding research community. In this work we focus on the video coding loss. We note that these three losses are not necessarily uncorrelated or independent.

For conventional MCP-based video coding, the reference information does not necessarily contain all the side information in S. Assume that with the best available motion estimation methods, a subset of side information, denoted as Y_c, is used as a reference for both the encoder and decoder in conventional video coding. Meanwhile, the side estimator g is used in the Wyner-Ziv video coder to extract the actual side information Y_d. The following inequality immediately follows

$$R_{X|Y_d}(d) \geq R_{X|Y_c}(d) \tag{2.6}$$

Proof: The proof of this inequality is trivial. Since all the side information available at the Wyner-Ziv video decoder is also available in the conventional video encoder, if for a supposedly *best available* motion estimation method, $R_{X|Y_d}(d) < R_{X|Y_c}(d)$, we'll simply replace the current *best available* motion estimator by the side estimator g. This inequality echoes the conclusion reached in [83] that Wyner-Ziv video coding may never beat conventional video coding in terms of rate distortion performance, which we cited earlier in Section 2.1.

Hence, to investigate the video coding loss $\Delta R_v(d)$, we need to compare the performance difference between two conventional video coding systems as shown in

Fig. 2.3. One uses references generated by the Wyner-Ziv side estimator g, and the other uses references generated by the conventional motion estimator f. The key difference is that the side estimator g does not have access to the original current frame, while f does.

2.3.2 Related Researches

A comprehensive analysis of rate distortion performance for conventional MCP-based video coding is described in [41–45]. In this work, signal power spectrum and Fourier analysis tools are used to analyze MCP-based video coding. This idea was further developed to analyze MCP-based scalable video coding [46–49], multiple description coding and leaky prediction [21,50–52]. In this thesis we extend this work to investigate the video coding loss for a Wyner-Ziv video coder. Not surprisingly, readers with knowledge of the above work might find familiar tools and sometimes similar conclusions from time to time in this work. The analysis in this work, as we shall show, is however not trivial given the different scenarios.

2.3.3 Problem Formulation

In this thesis conventional video coding means the conventional MCP-based structure shown in Fig. 2.2 henceforward unless otherwise specified. Denote $r_c(n)$ as the reference for the current frame in conventional video coding, $r_w(n)$ as the reference in Wyner-Ziv video coding, then

$$r_c(n) = f(s(n), \hat{S}(n)) \tag{2.7}$$

$$r_w(n) = g(\hat{S}(n)) \tag{2.8}$$

where $s(n)$ is the original current frame and $\hat{S}(n)$ is the previous decoded frames stored in the frame buffer. Here “previous” is used in terms of decoding order rather than actual physical order in the sequences, i.e., it can be the next frame as in the

B-frame case, or a lower quality of the current frame's reconstruction in the SNR scalable case.

There are many side estimators and Wyner-Ziv coders proposed in the literature. It is worth emphasizing at this point that the Wyner-Ziv video coding performance we investigate in this work is by no means the best available performance for Wyner-Ziv video coding, given the fact that there is no single optimal side estimator g available. Instead, we look at several general motion prediction based side estimators, not necessarily optimal but generally accepted and used. The conclusions reached in this work shall be true for the schemes discussed rather than for general structures.

The residual frame in the two systems is

$$e(n) = s(n) - c(n) \tag{2.9}$$

where $c(n) = r_c(n)$ for conventional video coding and $c(n) = r_w(n)$ for Wyner-Ziv video coding.

Consider the power spectrum of the residual frame. As shown in [41,44,45]

$$\begin{aligned}
\Phi_{ee}(\omega) &= \Phi_{ss}(\omega) - 2Re\{\Phi_{cs}(\omega)\} + \Phi_{cc}(\omega) \\
\Phi_{cs}(\omega) &= \Phi_{ss}(\omega)E\{e^{-j\omega^T\Delta}\} = \Phi_{ss}(\omega)e^{-\frac{1}{2}\omega^T\omega\sigma_\Delta^2} \\
\Phi_{cc}(\omega) &= \Phi_{ss}(\omega) \qquad (2.10)
\end{aligned}$$

where $E\{\cdot\}$ is the expectation operator and $\Delta = (\Delta_x, \Delta_y)$ is the motion vector error, i.e., the difference between the motion vector used and the true motion vector. Hence

$$\begin{aligned}
\Phi_{ee}(\omega) &= 2\Phi_{ss}(\omega) - 2\Phi_{ss}(\omega)e^{-\frac{1}{2}\omega^T\omega\sigma_\Delta^2} \qquad (2.11) \\
\frac{\Phi_{ee}(\omega)}{\Phi_{ss}(\omega)} &= 2 - 2e^{-\frac{1}{2}\omega^T\omega\sigma_\Delta^2} \qquad (2.12)
\end{aligned}$$

Then the rate saving over INTRA-frame coding by using motion search is [102,103]

$$\Delta R = \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{\Phi_{ee}(\omega)}{\Phi_{ss}(\omega)}\,d\omega \tag{2.13}$$

Hence the rate difference between two systems using two motion vectors MV_1 and MV_2 is

$$\Delta R_{1,2} = \Delta R_1 - \Delta R_2$$

$$\begin{aligned}
&= \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{\Phi_{ee,1}(\omega)}{\Phi_{ss}(\omega)}\,d\omega - \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{\Phi_{ee,2}(\omega)}{\Phi_{ss}(\omega)}\,d\omega \\
&= \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{\Phi_{ee,1}(\omega)}{\Phi_{ee,2}(\omega)}\,d\omega \\
&= \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{1-e^{-\frac{1}{2}\omega^T\omega\sigma_{\Delta_1}^2}}{1-e^{-\frac{1}{2}\omega^T\omega\sigma_{\Delta_2}^2}}\,d\omega \qquad (2.14)
\end{aligned}$$

where $\Phi_{ee,1}(\omega)$ and $\Phi_{ee,2}(\omega)$ are the residual frame's power spectrum density with motion vectors MV_1 and MV_2 respectively.

For the n-th frame that is to be decoded, denote the motion vector obtained by the side estimator using previous reconstructed frames as $MV'_{n-1} = (x'_{n-1}, y'_{n-1})$. Also let ρ_x and ρ_y be the correlation between x_n and x'_{n-1}, and y_n and y'_{n-1} respectively, where $MV_n = (x_n, y_n)$ is the true motion vector. The motion vector error is then

$$\Delta = (\Delta_x, \Delta_y) = (x_n - \hat{x}_n, y_n - \hat{y}_n) \qquad (2.15)$$

To minimize σ_Δ^2, the optimal linear minimum mean square error (MMSE) estimator is given by

$$\hat{x}_n = \rho_x x'_{n-1}, \hat{y}_n = \rho_y y'_{n-1} \qquad (2.16)$$

And the variance of the motion vector error component is given by

$$\sigma_{\Delta_x}^2 = (1-\rho_x^2)\sigma_x^2, \sigma_{\Delta_y}^2 = (1-\rho_y^2)\sigma_y^2 \qquad (2.17)$$

Without loss of generality, we assume x and y are independent identically distributed, then $\sigma_x^2 = \sigma_y^2 = \frac{1}{2}\sigma_{MV}^2$, hence the variance of the motion vector error is

$$\sigma_\Delta^2 = (1 - \frac{\rho_x^2+\rho_y^2}{2})\sigma_{MV}^2 = (1-\rho^2)\sigma_{MV}^2 \qquad (2.18)$$

where $\rho = \sqrt{\frac{\rho_x^2+\rho_y^2}{2}}$. The motion vector correlation signal-to-noise ratio ($cSNR$) is defined as

$$cSNR = 10\log_{10}\frac{\sigma_{MV}^2}{\sigma_\Delta^2} = 10\log_{10}\frac{1}{1-\rho^2} \qquad (2.19)$$

We now consider the following two conventional MCP-based video coding methods.

The first method, referred to as DPCM-frame coding, does not perform any motion search and let $MV = (0, 0)$, i.e., it simply subtracts the previous reconstructed frame $\hat{s}(n-1)$ from the current frame $s(n)$ and codes the difference. This is similar to Differential Pulse-Code Modulation (DPCM) used in communications. Apparently this encoder has very low complexity, which is also one of the major advantages for Wyner-Ziv video coding. By comparing with this method, we can evaluate the performance gain over the simple conventional low-complexity video coding.

The second method, referred to as INTER-frame coding, performs the best available motion search and the only distortion is introduced due to motion search pixel accuracy. Since half-pixel accuracy is used in H.263/MPEG-2 and quarter-to-eighth pixel accuracy is used in H.264, by comparing this method with Wyner-Ziv video coding, we are able to evaluate the performance gap between Wyner-Ziv video coding and the popular H.263 codec and the state-of-the-art H.264 codec.

2.3.4 Wyner-Ziv Video Coding vs. DPCM-Frame Video Coding

For DPCM-frame video coding, the motion vector error is

$$\Delta = (x, y) - (0, 0) = MV \Rightarrow \sigma_\Delta^2 = \sigma_{MV}^2 \tag{2.20}$$

With (2.18) and (2.20), the rate difference between the Wyner-Ziv video coding and DPCM-frame coding is

$$\Delta R_{WZ,DPCM} = \frac{1}{8\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \log_2 \frac{1 - e^{-\frac{1}{2}\omega^T \omega \sigma_{MV}^2}}{1 - e^{-\frac{1}{2}\omega^T \omega (1-\rho^2) \sigma_{MV}^2}} \, d\omega \tag{2.21}$$

2.3.5 INTER-Frame Video Coding vs. DPCM-Frame Video Coding

For INTER-frame video coding, the variance of the motion vector error is introduced by the pixel inaccuracy. The effect of the fractional pixel accuracy has been studied in the literature [42, 104]. In this work we adopt the analytical results presented in [104]. The motion vector error variance $\sigma_{\Delta_\beta}^2$ for a 8×8 block in a

QCIF sequence is $0.105, 0.090$ and 0.080 for integer-pixel, half-pixel and quarter-pixel motion search accuracy respectively. Here the motion vector error variance is a function of frame size and search block size and, unlike the cases for DPCM-frame video coding, independent of the variance of motion vectors. The rate difference between INTER-frame video coding and DPCM-frame video coding is

$$\Delta R_{INTER,DPCM} = \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2 \frac{1-e^{-\frac{1}{2}\omega^T\omega\sigma^2_{MV}}}{1-e^{-\frac{1}{2}\omega^T\omega\sigma^2_{\Delta_\beta}}}\,d\omega \qquad (2.22)$$

2.3.6 Simulation Results

We now present simulation results based on (2.21) and (2.22). We examine the rate savings vs. $cSNR$ for several different motion vector variances σ^2_{MV}. The rate difference compared with DPCM-frame video coding is shown in Fig. 2.4. The rate difference compared with DPCM-frame coding by using INTER-frame video coding is shown in Table 2.1.

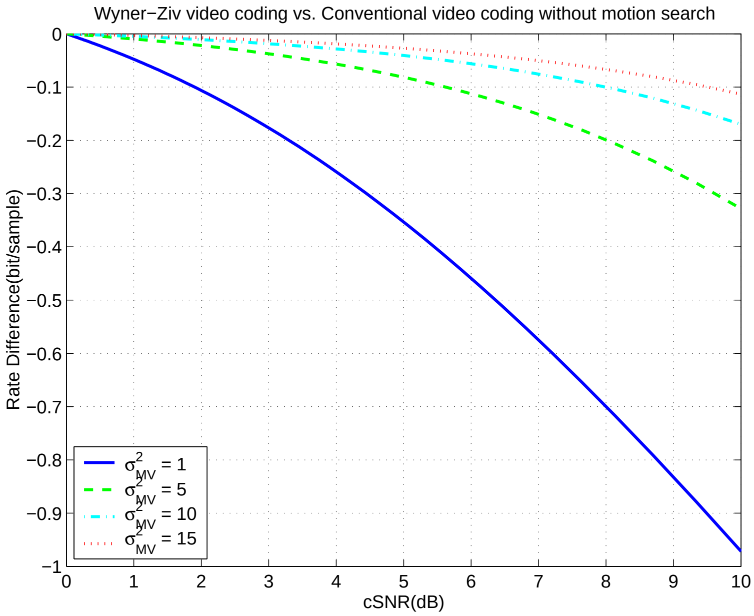


Fig. 2.4. Wyner-Ziv video coding vs. DPCM-frame video coding.

Table 2.1

Rate difference between INTER-frame video coding and DPCM-frame video coding.

	$\sigma^2_{MV} = 1$	$\sigma^2_{MV} = 5$	$\sigma^2_{MV} = 10$	$\sigma^2_{MV} = 15$
Integer-pixel (bit/sample)	0.94	1.09	1.11	1.12
Half-pixel (bit/sample)	1.04	1.19	1.21	1.21
Quarter-pixel (bit/sample)	1.11	1.26	1.28	1.29

Table 2.2

σ^2_{MV} and cSNR (frame size: QCIF, block size: 8 × 8).

Sequence	*Foreman*	*Coastguard*	*Carphone*	*Stefan*	*Table*	*Silent*
σ^2_{MV}	9.46	0.76	7.23	15.40	2.09	1.93
$cSNR$ (dB)	1.54	5.69	0.44	1.58	1.05	0.86

Since the rate savings using Wyner-Ziv video coding varies with σ^2_{MV} and $cSNR$, to have a better idea of the actual performance comparison, we test six standard QCIF sequences and obtain their σ^2_{MV} and $cSNR$. The six standard test sequences we use are *Foreman*, *Coastguard*, *Carphone*, *Stefan*, *Table* and *Silent*. For each frame, we obtain the estimated motion vector MV'_{n-1} by a side estimator and the "true" motion vector using the H.264 reference software JM8.0 with eighth-pixel motion search. In this side estimator, the motion vector $MV'_{n-1} = (x'_{n-1}, y'_{n-1})$ is obtained with forward motion prediction using the co-located block in the previous frame $\hat{s}(n-1)$ as the source and $\hat{s}(n-2)$ as the reference. σ^2_{MV} and $cSNR$ are then computed for each frame and averaged over the whole sequence. $cSNR$ and σ^2_{MV} of each test sequence are given in Table 2.2.

As shown in Fig. 2.4, the rate savings over DPCM-frame video coding by using Wyner-Ziv video coding is more significant when the motion vector variance σ^2_{MV} is small. For $\sigma^2_{MV} = 1$, Wyner-Ziv video coding can save up to 0.96bit/sample, while for $\sigma^2_{MV} = 15$, it can save merely slightly more than 0.1bit/sample even with high $cSNR$. This makes sense since for lower motion vector variance, the side estimator has a better chance of using a motion vector that is not very distant from the true motion vector. Meanwhile, $cSNR$ plays a critical role in rate savings. Table 2.2 shows that most sequences have small $cSNR$ for the side estimator we choose. For sequences with $cSNR \leq 2dB$, the rate saving over DPCM-frame video coding is very marginal. It is hard to justify using Wyner-Ziv video coding in this case, given the fact that the Wyner-Ziv video decoder has a much higher complexity. All the sequences we tested, except *Coastguard*, fall in this $cSNR$ range. Apparently a more effective side estimator is required in order to use Wyner-Ziv video coding on these sequences.

For INTER-frame video coding, we tabulate the results in Table 2.1 for different motion search accuracies. Here the rate difference is also compared with DPCM-frame video coding. Comparing the results in Table 2.1 with Fig. 2.4, we note that except for very small motion vector variance cases, INTER-frame video coding generally has about 1bit/sample gain over Wyner-Ziv video coding. For sequences with small σ^2_{MV}, the difference is lower, ranging from 0.2-0.6bit/sample depending on the specific side estimator used.

For readers who are more comfortable with PSNR comparisons, we can obtain a rough estimate by using $\Delta PSNR = 6.02\Delta R$. The conclusion we reach here is that Wyner-Ziv coding can achieve a gain up to 6dB (for small motion vector variance) or 1 - 2dB (for normal to large motion vector variance) over DPCM-frame video coding. With the side estimator mentioned above, Wyner-Ziv video coding has 0.5 - 1dB gain for most test sequences. On the other hand, INTER-frame video coding outperforms Wyner-Ziv video coding by at least 1dB (for small motion vector variance) or more generally 6dB (for normal to large motion vector variance) even when a highly correlated motion vector is available. It is noted that the conventional coding results are practically achievable. A practical Wyner-Ziv video coder, as mentioned earlier, may suffer additional system loss up to 0.5bit/sample (or equivalently, 3dB) [101] as well as source coding loss.

2.4 Conventional Motion Search Methods Revisited

In the simulation in Section 2.3, we assume that the true motion vector between the previous two reconstructed frames is available for the estimation of the current motion vector. However, this is not always achievable given that the actual motion search pixel accuracy is always limited. Hence the Wyner-Ziv video coding bound reached in Section 2.3 will be compromised by a coarser motion search. It is well known that conventional MCP-based video coding benefits from continually improving motion search methods, a more general question is then how does these conventional motion search methods affect the side estimator performance in a Wyner-Ziv video coder. In this Section, we discuss side estimators using two motion search methods, sub-pixel motion search and multi-reference motion search.

2.4.1 The Sub-Pixel Side Estimator

We model the true motion vector MV_n for the current frame as a random variable with a mean of the previous frame's true motion vector MV_{n-1}, as shown in Fig. 2.5. MV_n is a 2-D random variable, for the sake of simplicity, we show a 1-D slice of the probability density function in Fig. 2.5.

As discussed in Section 2.3, with the motion vector correlation ρ between MV_n and MV_{n-1}, the optimal estimator is

$$M\hat{V}_n = \rho MV_{n-1} \tag{2.23}$$

such that

$$\sigma_{\Delta}^2 = (1-\rho^2)\sigma_{MV}^2 \tag{2.24}$$

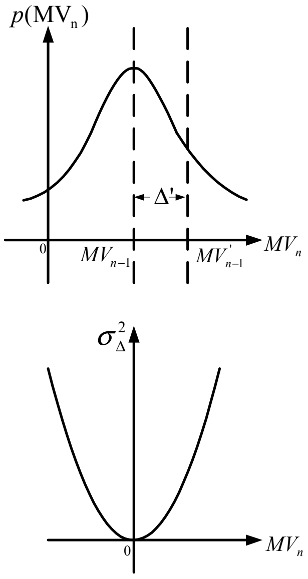


Fig. 2.5. Motion vector model.

Now consider the case that the actual previous motion vector $MV'_{n-1} = MV_{n-1} + \Delta'$, where Δ' is random noise due to the motion search inaccuracy. Then the actual estimate used is

$$\hat{MV}_n = \rho MV'_{n-1} = \rho(MV_{n-1} + \Delta') \tag{2.25}$$

And the actual variance of the motion vector error is

$$\begin{aligned} \sigma^2_{\Delta_a} &= E[(MV_n - \rho(MV_{n-1} + \Delta'))^2] = E[((MV_n - \rho MV_{n-1}) - \rho\Delta')^2] \\ &= \sigma^2_\Delta + \rho^2\sigma^2_{\Delta'} = (1-\rho^2)\sigma^2_{MV} + \rho^2\sigma^2_{\Delta'} \end{aligned} \tag{2.26}$$

where the Δ' is zero mean and independent of the motion vector error Δ. The additional rate incurred due to the use of an inaccurate previous motion vector, compared to the case when a true previous motion vector is used becomes

$$\Delta R = \frac{1}{8\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\log_2\frac{1-e^{-\frac{1}{2}\omega^T\omega[(1-\rho^2)\sigma^2_{MV}+\rho^2\sigma^2_{\Delta'}]}}{1-e^{-\frac{1}{2}\omega^T\omega(1-\rho^2)\sigma^2_{MV}}}\,d\omega \tag{2.27}$$

$\sigma^2_{\Delta'}$ depends on the motion search pixel accuracy and we can use the results in Section 2.3 derived from [104]. We also derive the $\sigma^2_{\Delta'}$ for pixel subsampling based on the analytical method outlined in [104]. The corresponding $\sigma^2_{\Delta'} = 0.142, 0.213, 0.295$ for $2:1, 4:1$ and $8:1$ subsampling respectively. For an $n:1$ subsampling, only one pixel in a $n \times n$ pixel grid is used for motion search.

The rate increase is given in Fig. 2.6 and Fig. 2.7. Compared with the Wyner-Ziv video coding bound discussed in Section 2.3, a practical coder with integer pixel motion search may use 0.02-0.4bit/sample or more depending on the motion vector variance due to the inaccuracy in the previous motion vector.

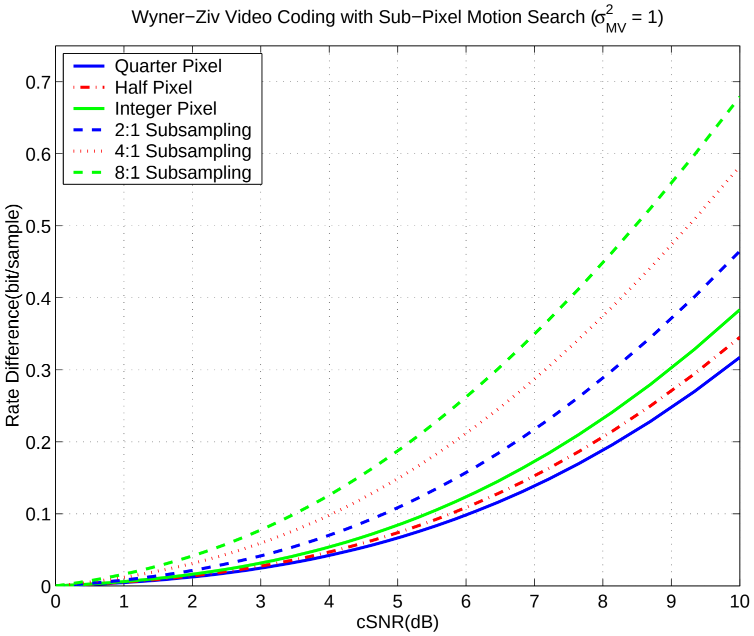


Fig. 2.6. Rate increase due to motion search pixel inaccuracy ($\sigma^2_{MV} = 1$).

Unlike conventional MCP-based video coding, where the motion search pixel accuracy can greatly affect the coding efficiency, Fig. 2.6 and Fig. 2.7 show that Wyner-Ziv video coding is not as sensitive to search accuracy. For small σ^2_{MV}, the difference in using the integer pixel motion search and quarter pixel motion search is less than 0.07bit/sample, where the difference in sequences with larger σ^2_{MV} further reduces to less than 0.005bit/sample.

An interesting observation from Fig. 2.6 and Fig. 2.7 is that the use of subsampling motion search is not always unacceptable in terms of coding efficiency. For

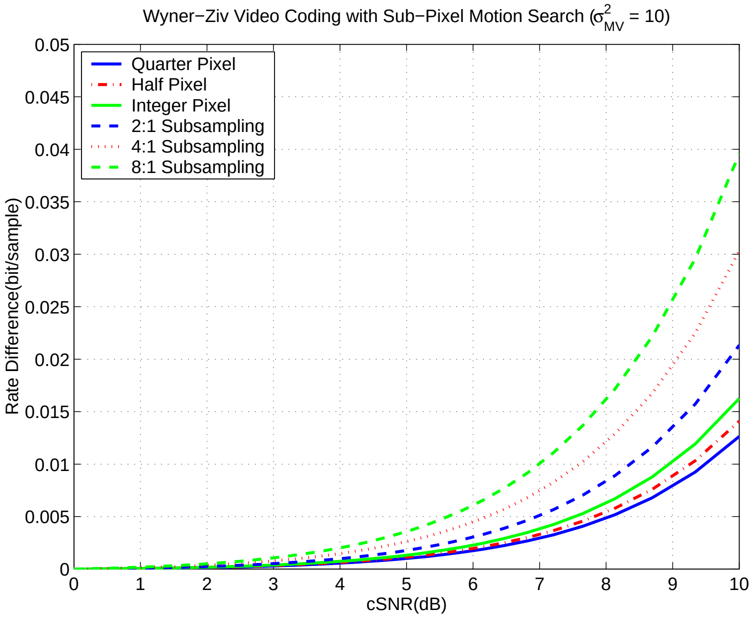


Fig. 2.7. Rate increase due to motion search pixel inaccuracy ($\sigma^2_{MV} = 10$).

small σ^2_{MV}, using $2:1$ subsampling only leads to less than 0.1bit/sample rate increase, while in the large σ^2_{MV} case the difference is 0.01bit/sample, compared with integer pixel motion search. These results show that for side estimators that cannot locate the rough position of the true motion vectors in the first place, it makes little sense to further refine the motion vector by using sub-pixel motion search. For decoders with limited computing capability, a 2:1 or even coarser subsampling is worth considering.

2.4.2 Multi-Reference Motion Search

We now consider the use of multi-reference motion search. As described in [45] for multi-hypothesis motion search in conventional MCP-based video coding, the normalized power spectrum density of the residual frames with the use of multi-reference motion search is

$$\frac{\Phi_{ee}(\omega)}{\Phi_{ss}(\omega)} = \frac{N+1}{N} - 2e^{-\frac{1}{2}\omega^T\omega\sigma^2_{\Delta_a}} + \frac{N-1}{N}e^{-(1-\rho_\Delta)\omega^T\omega\sigma^2_{\Delta_a}} \tag{2.28}$$

where N is the number of references and ρ_Δ is the correlation between two motion vector errors. Consider the case where motion vector errors are mutually uncorrelated, i.e., $\rho_\Delta = 0$, the rate difference in using N references over using only one reference is

$$\begin{aligned} \Delta R &= \Delta R(N) - \Delta R(1) \\ &= \frac{1}{8\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \log_2 \frac{\frac{N+1}{N} - 2e^{-\frac{1}{2}\omega^T\omega\sigma^2_{\Delta_a}} + \frac{N-1}{N} e^{-(1-\rho_\Delta)\omega^T\omega\sigma^2_{\Delta_a}}}{2 - 2e^{-\frac{1}{2}\omega^T\omega\sigma^2_{\Delta_a}}} \, d\omega \qquad (2.29) \end{aligned}$$

Unlike the case in conventional video coding discussed in [45], where $\sigma^2_{\Delta_a}$ is solely due to the motion search pixel inaccuracy, in Wyner-Ziv video coding, as shown in (2.26), $\sigma^2_{\Delta_a}$ is introduced by both the inaccuracy in previous motion vector and also the imperfect correlation between current motion vectors and previous motion vectors. The results are shown in Fig. 2.8 and Fig. 2.9.

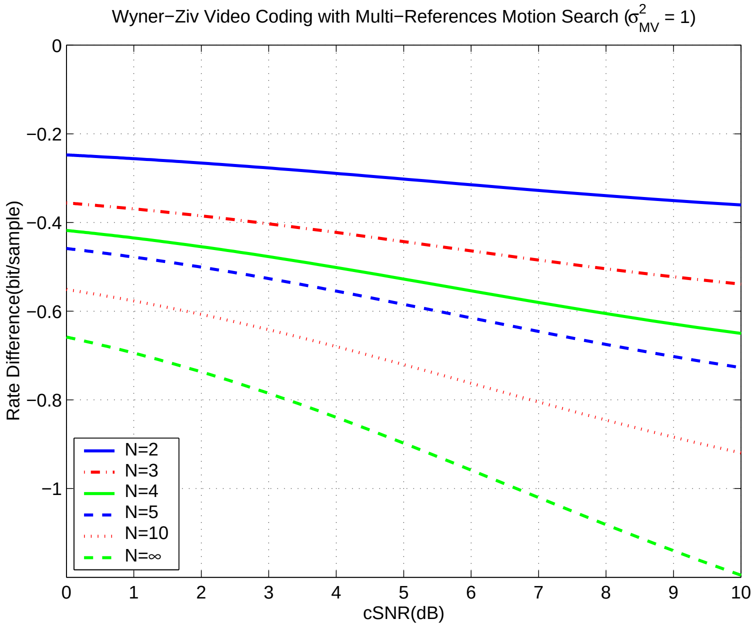


Fig. 2.8. Rate saving with multi-reference ($\sigma^2_{MV} = 1$).

Fig. 2.8 and Fig. 2.9 show that multi-reference motion search can effectively improve Wyner-Ziv video coding efficiency. With two references, it can save more than

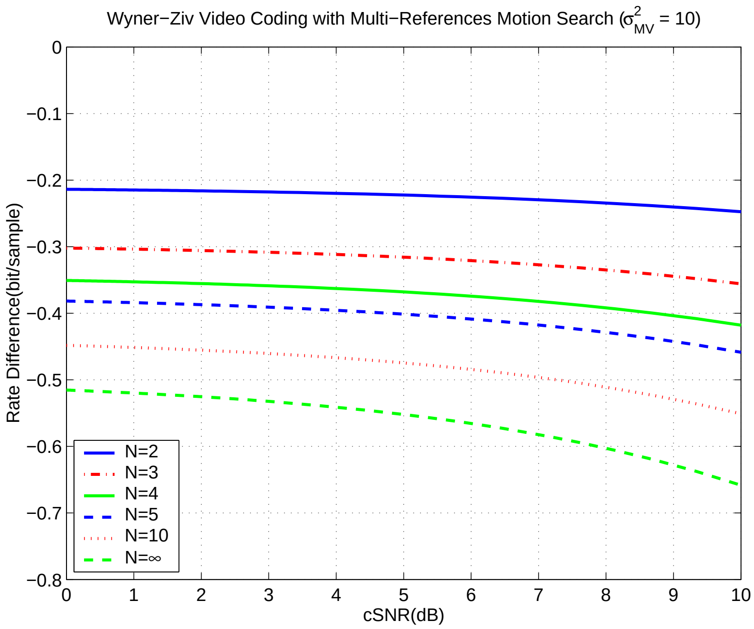


Fig. 2.9. Rate saving with multi-reference ($\sigma^2_{MV} = 10$).

0.2bit/sample regardless of the motion vector variance and $cSNR$. With additional number of references, the coding efficiency can be further improved.

Another observation is $cSNR$ does not affect the rate saving as much in multi-reference motion search as in sub-pixel motion search, i.e., multi-reference motion search remains effective in low correlated sequences. In terms of σ^2_{MV}, for small σ^2_{MV} sequences, multi-reference motion search achieves slightly better results than larger σ^2_{MV} sequences.

2.4.3 Experimental Results

In this Section we implement the side estimator with sub-pixel motion search and multi-reference motion search and show experimental results for the test sequences.

To obtain a fair comparison of different motion search modes, we evaluate the PSNR of the references constructed by the side estimators instead of the final coding efficiency. This is because the final video coding results also factor into the source coding difference that depends on specific codec design, especially the choice of channel coding. As is known in rate-distortion theory [102], we can bound the distortion by

$$d(R) = \sigma^2 2^{-\epsilon R} \tag{2.30}$$

where σ^2 is the coding complexity, or the distortion of the reference frame in our case. Hence the reference quality gives a reliable practical upper bound for the investigated video coding scheme.

We first test the results for motion search pixel accuracy. The decoder side estimator performs forward motion search between the previous two reconstructed frames with different pixel accuracies. The motion vector obtained, along with the previous reconstructed frame, is then used to construct the reference for the current frame. The results of reference quality are shown in Table 2.3. Three motion search pixel accuracies are listed compared with the results of DPCM-frame and INTER-frame video coding. As shown in Table 2.3, by using integer pixel motion search, the reference quality can gain 1.2 - 3.5dB over DPCM-frame coding. The only exception is the *Carphone* sequence, which, as pointed out in Section 2.4.1, is expected to show little gain since its $cSNR$ is very small. On the other hand, the reference quality in conventional video coding with half-pixel motion search is 0.8 - 4.8dB better than the Wyner-Ziv video side estimator. The closest match happens in the *Coastguard* sequence, which has large $cSNR$ and small σ^2_{MV}. The largest difference occurs for the *Carphone* sequence again as a result of low $cSNR$. For the side estimator in Wyner-Ziv video coding, compared with integer pixel motion search, the use of half-pixel motion search can improve 0.3 - 0.7dB in reference quality while 2:1 subsampling motion search only loses 0.2 - 0.5dB. These experimental results are consistent with our analytical results in Section 2.4.1.

We test the use of multi-reference motion search. Consider the following four side estimators.

- SE-1: For each block in the current frame, the side estimator uses the co-located block in the previous reconstructed frame as the source and the second

Table 2.3
PSNR of side estimators with different motion search pixel accuracy, in *dB*.

Sequence	Half Pixel	Integer Pixel	2:1 Subsampling	DPCM-Frame	INTER-Frame (Half Pixel)
Foreman	31.22	30.66	30.38	28.17	33.21
Coastguard	31.16	30.82	30.53	27.32	31.61
Carphone	29.31	29.03	28.62	29.77	33.80
Stefan	23.51	22.88	22.61	19.48	24.84
Table	31.21	30.48	30.00	27.10	33.86
Silent	34.10	33.62	33.41	32.43	38.11

Table 2.4
Side estimators with multi-reference motion search, in *dB*.

Sequence	SE-1	SE-2	SE-3	SE-4 (B)	DPCM-Frame
Foreman	30.66	30.87	30.84	32.56	28.17
Coastguard	30.82	30.37	29.27	32.17	27.32
Carphone	29.03	29.44	29.79	31.72	29.77
Stefan	22.88	23.19	22.66	23.54	19.48
Table	30.48	30.81	30.63	32.20	27.10
Silent	33.62	33.89	34.03	36.09	32.43

previous frame as the reference and performs forward motion prediction. The obtained motion vector is then used on the previous reconstructed frame to find the reference block for the current frame.

- SE-2: The side estimator obtains two motion vectors. The first motion vector is generated in the same way as SE-1, while the other points to the next available best matching blocks. These motion vectors are used on previous reconstructed frames to find the corresponding reference blocks, which are then averaged to construct the reference for the current block.

- SE-3: The side estimator obtains the reference in the same way as SE-2. The only difference is that it searches five motion vectors instead of two.

- SE-4: For each block in the current frame, the side estimator first uses the co-located block in the next reconstructed frame as the source and the previous reconstructed frame as the reference to perform forward motion estimation. Denote the obtained motion vector as MV_F. It then uses the co-located block in the previous frame as the source and next reconstructed frame as the reference to perform backward motion estimation. Denote the obtained motion vector as MV_B. It then uses $\frac{MV_F}{2}$ on the previous reconstructed frame to find the corresponding reference block P_F, and uses $\frac{MV_B}{2}$ on the next reconstructed frame to find the corresponding reference block P_B. P_F and P_B are averaged to serve as the reference for the current block. Note that in this side estimator, the sequence will have to be decoded similar to a B-frame in conventional video coding such that both reference frames are available when a frame is to be decoded.

The results are shown in Table 2.4. The motion search accuracy for all side estimators is integer pixel. This shows that the use of two or more reference blocks in the same frame can hardly improve the side estimator quality. This is because the reference blocks in the same frame do not satisfy the uncorrelated condition $\rho_\Delta = 0$ assumed in Section 2.4.2. Instead, when the reference blocks are low correlated or negatively correlated, which is the case for SE-4, the side estimator can perform fairly well. In SE-4, the side estimator achieved 0.7 - 2.7dB gain over single motion vector case and is 2 - 5.1dB better than DPCM-frame coding.

2.5 Wyner-Ziv Video Decoder with a Refined Side Estimator

The analytical and experimental results in Section 2.3 and 2.4 show that the performance difference of Wyner-Ziv video coding compared with conventional video coding depends on σ^2_{MV} and $cSNR$. While σ^2_{MV} is a sequence intrinsic characteristic, $cSNR$ depends on the motion vector accuracy obtained by the side estimator. In this Section we present a new Wyner-Ziv video decoder with a refined side estimator to improve the Wyner-Ziv video coding efficiency by improving $cSNR$.

Many current side estimators use the information extracted from the previous reconstructed frames. However, in Wyner-Ziv video coding, with the input of Wyner-Ziv frame data, the decoder has a gradually improving reconstruction of the current frame. Hence it is possible to utilize the information from the current frame's lower quality reconstruction as well. This is analogous to SNR scalable video coding in conventional video coding. In that case, a previous frame's reconstruction is first used as a reference, while a lower quality reconstructions of the current frame can later be used as references for the enhancement layers [6, 10, 11].

Based on this idea, we propose a new Wyner-Ziv decoder with a refined side estimator(RSE). We implement a Wyner-Ziv video coder based on the H.264 reference software JM8.0. In the encoder each frame is coded either as an INTRA frame or a Wyner-Ziv frame. H.264 INTRA mode is used to code an INTRA frame. A Wyner-Ziv frame is coded in the spatial domain. After all pixels are quantized, the quantized pixels are coded bitplane by biplane by a turbo coder. The parity bits generated by the turbo coder are sent to decoder. The side estimator at the decoder is shown in Fig. 2.10. The INTRA frame is reconstructed with H.264 INTRA decoder. A Wyner-Ziv frame first estimates a reference using the side estimator. This

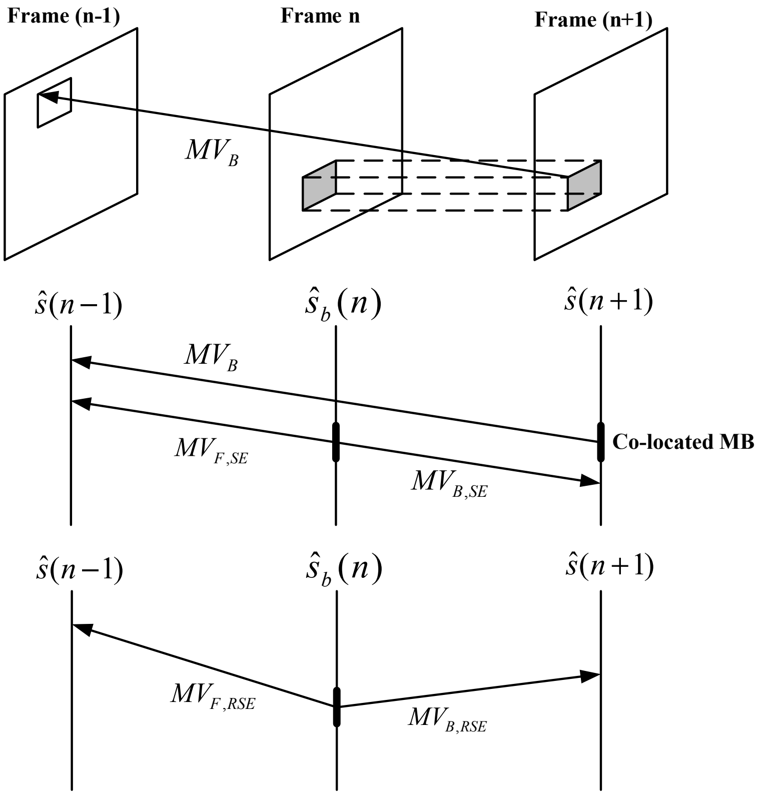


Fig. 2.10. Wyner-Ziv video decoder with a refined side estimator.

reference is used with the additional incoming parity bits to decode the frame with turbo decoder. We compare the following five coding methods in our experiments

- H.263-I: every frame is coded by H.263+ reference software TMN3.1.1 INTRA mode;
- H.264-I: every frame is coded by JM8.0 INTRA mode;
- H.264-IB: every even frame is coded by JM8.0 INTRA mode, while the odd frames are coded by JM8.0 bi-directional mode with quarter-pixel motion search accuracy;
- WZ-SE: every even frame is coded by JM8.0 INTRA mode, while the odd frames are coded as a Wyner-Ziv frame. At the decoder, the side estimator estimates the reference using the forward motion search as defined in B-frame

direct mode in H.264. As shown in Fig. 2.10, for every block to be decoded in the current Wyner-Ziv frame $s(n)$, its co-located block in the next reconstructed INTRA frame $\hat{s}(n+1)$ searches its best match in $\hat{s}(n-1)$ and obtains a motion vector MV_B. In H.264, the B-frame motion search Direct Mode [53], the motion is assumed to be linear. Hence for the block to be decoded, the estimated motion vector $MV_{F,SE}$ from $s(n)$ to $\hat{s}(n-1)$ is $\frac{MV_B}{2}$, and the estimated motion vector $MV_{B,SE}$ from $s(n)$ to $\hat{s}(n-1)$ is $-\frac{MV_B}{2}$. Using these two motion vectors, the two motion compensated blocks in $\hat{s}(n-1)$ and $\hat{s}(n+1)$ are found and averaged, which is then used as the reference. Integer motion search is used.

- WZ-RSE: the encoding is the same as WZ-SE. At the decoder, WZ-RSE also performs the same search and decoding as in WZ-SE. After the incoming parity bits are used along with the reference to reconstructed the current frame $\hat{s}(n)$, WZ-RSE performs a second motion search, as shown in Fig. 2.10. In this refined motion search, for every block in $\hat{s}(n)$, it searches its best match in $\hat{s}(n-1)$ and $\hat{s}(n+1)$ respectively and obtains two new motion vectors $MV_{F,RSE}$ and $MV_{B,RSE}$. The two best matched blocks in $\hat{s}(n-1)$ and $\hat{s}(n+1)$ are then averaged to construct a new reference. The parity bits received before are now used to decode this frame again with this new reference. WZ-RSE also uses integer pixel motion search.

We test these five methods on the six test sequences used in our previous experiments. For each sequence, we compare the coding efficiency of the Wyner-Ziv frames and also that of all the frames. The data rate of H.263-I, H.264-I and H.264-IB is adjusted by the quantization parameter (QP). For WZ-SE and WZ-RSE, we adjust the Wyner-Ziv frame's data rate by setting the number of bitplanes used for decoding, while the QP for the even frames(INTRA frame) is chosen such that the PSNR quality is close to its neighboring Wyner-Ziv frames.

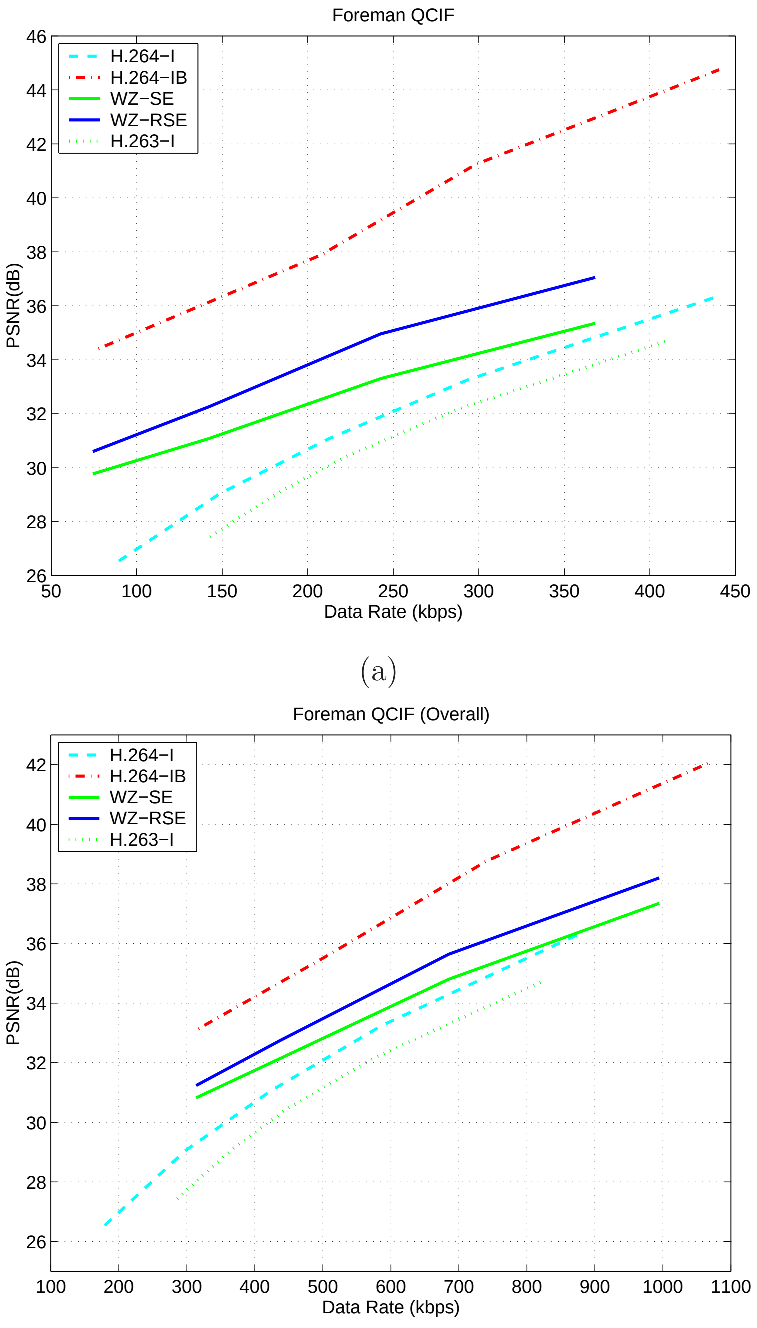
In Fig. 2.11 - 2.16, we show the video coding results. Compared with H.264-IB, WZ-SE generally trails by more than 3dB. The only exception is the *Silent* sequence, where WZ-SE only lags behind by less than 1dB for low to medium data rates. The performance difference can reach as much as 6dB in the *Carphone* sequence or 8dB in the *Stefan* sequence, both of which have a large σ^2_{MV} and low $cSNR$.

Compared with conventional INTRA video coding, for sequences with small σ^2_{MV}, WZ-SE generally outperforms H.264-I by 2-4dB or H.263-I by 3-5dB. For sequences with large σ^2_{MV}, the difference is much less and sometimes INTRA coding is even better than WZ-SE. This is more obvious at high data rates, which shows that the Turbo coding used in our Wyner-Ziv coder is not efficient enough compared with INTRA coding.

Comparing the performance of WZ-SE and WZ-RSE, the results show that in each test sequence, the use of the current frame's reconstruction to construct the reference can have up to a 2dB gain. The tradeoff, however, is to introduce more decoding complexity.

2.6 Conclusions

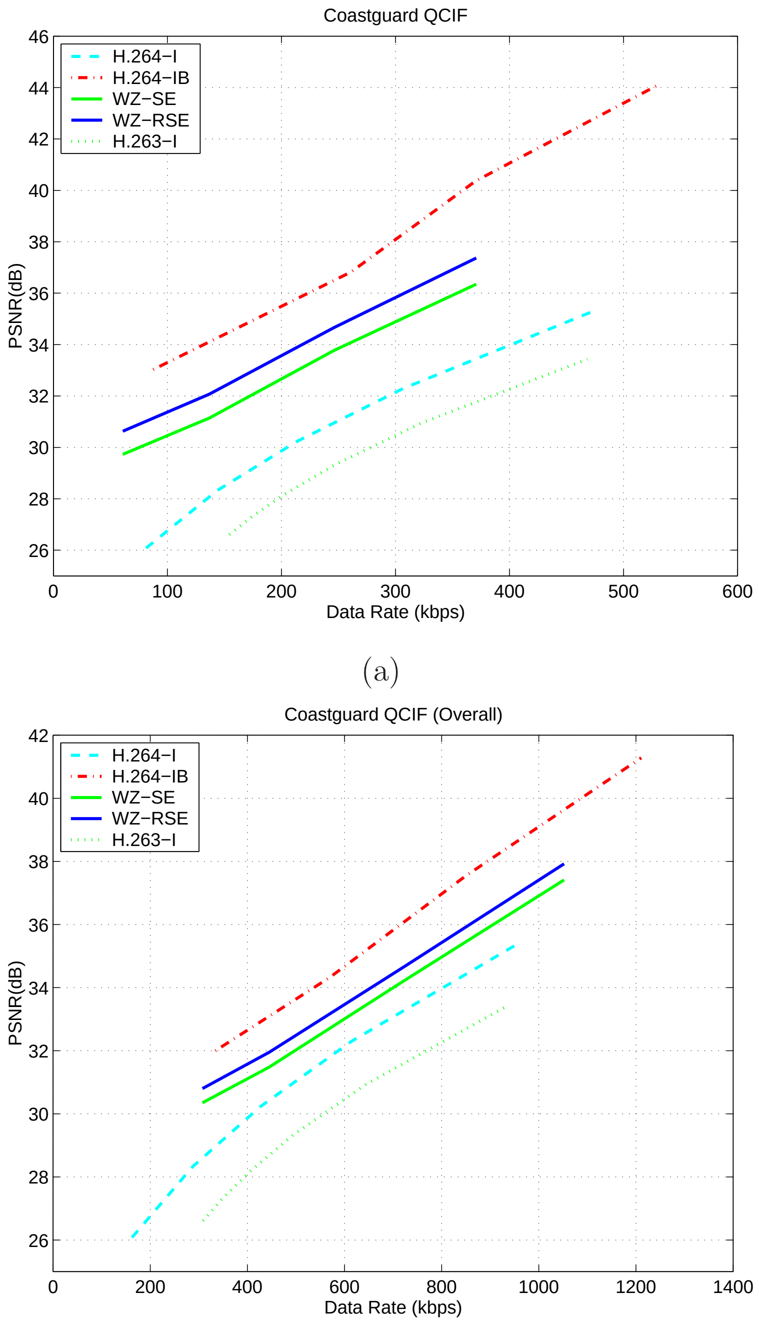
In this Chapter, we discuss the coding efficiency of Wyner-Ziv video coding. Theoretically, with the simple side estimator we discussed in Section 2.3, Wyner-Ziv video coding may outperform by up to 0.9bit/sample over DPCM-frame coding. However, most practical sequences, which have low $cSNR$, the rate saving is less than 0.1bit/sample. When compared with INTER-frame video coding, Wyner-Ziv video coding leads to a rate increase of as much as 0.2-0.6bit/sample for sequences with small motion vector variance and 1bit/sample for those with large motion vector variances. These results show that the simple side estimator cannot meet the expectation of practical applications and the development of more effective side estimation is still an open problem.



(a)

(b)

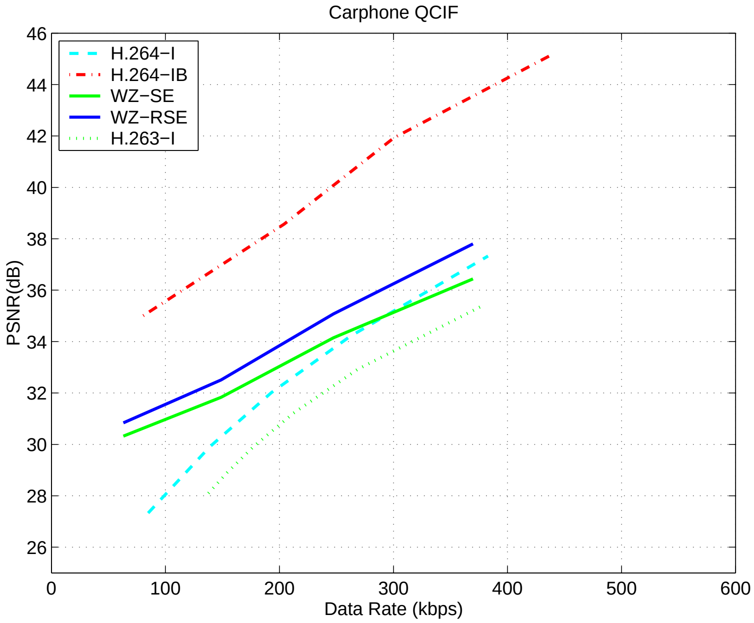
Fig. 2.11. Wyner-Ziv video decoder with a refined side estimator (*Foreman*) (a) Wyner-Ziv frame; (b) All frames.



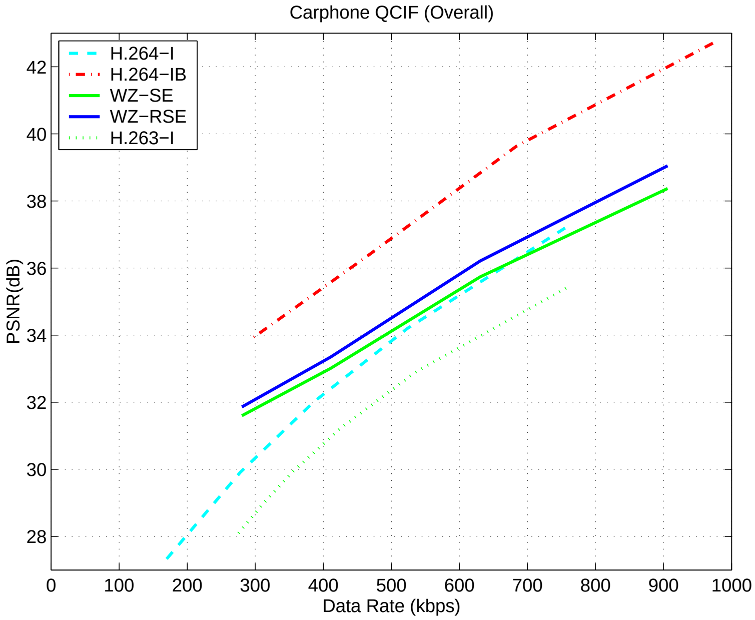
(a)

(b)

Fig. 2.12. Wyner-Ziv video decoder with a refined side estimator (*Coastguard*) (a) Wyner-Ziv frame; (b) All frames.

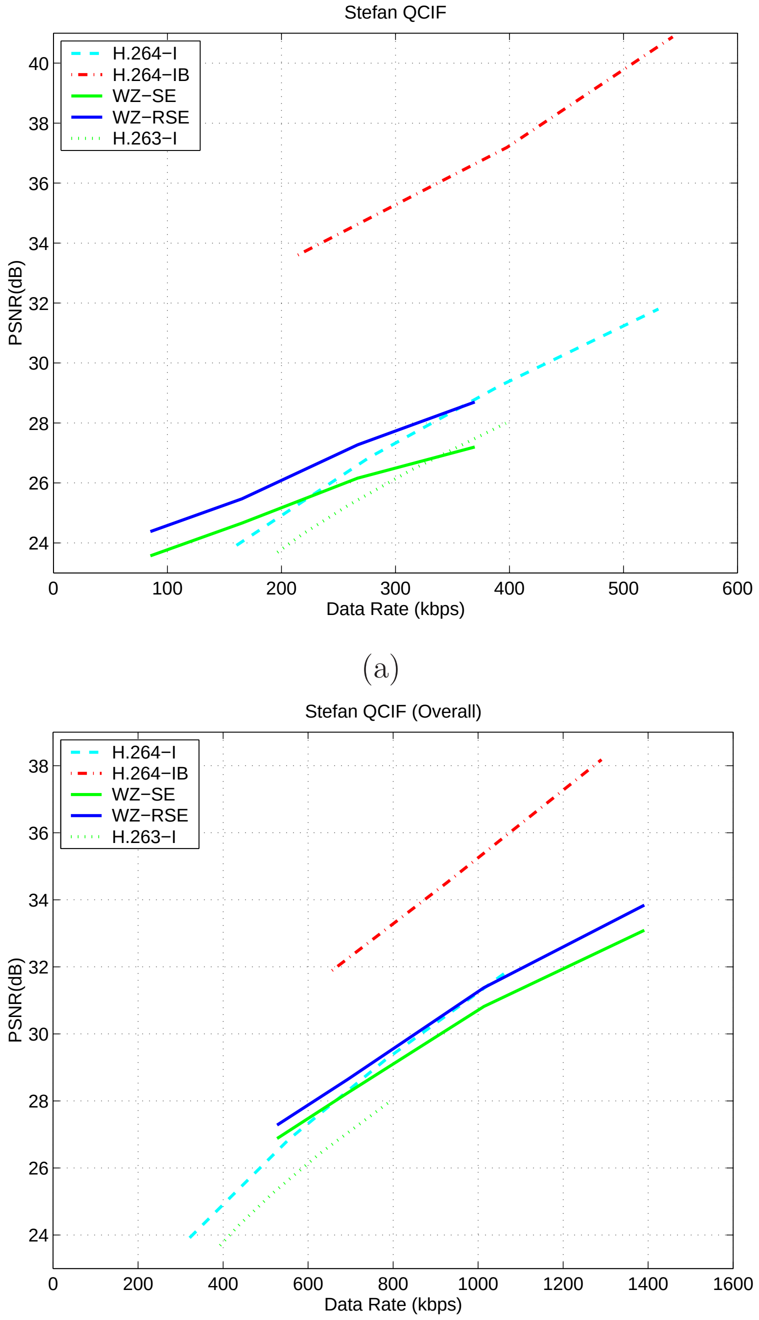


(a)



(b)

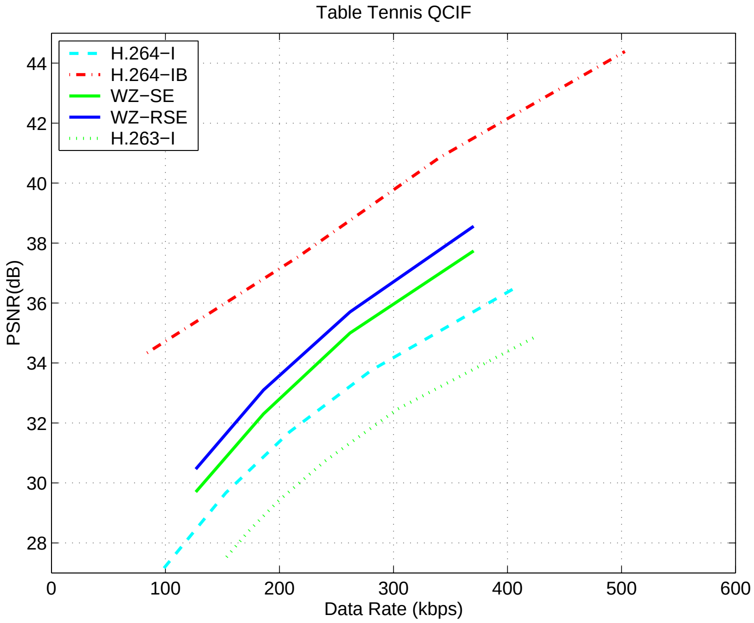
Fig. 2.13. Wyner-Ziv video decoder with a refined side estimator (*Carphone*) (a) Wyner-Ziv frame; (b) All frames.



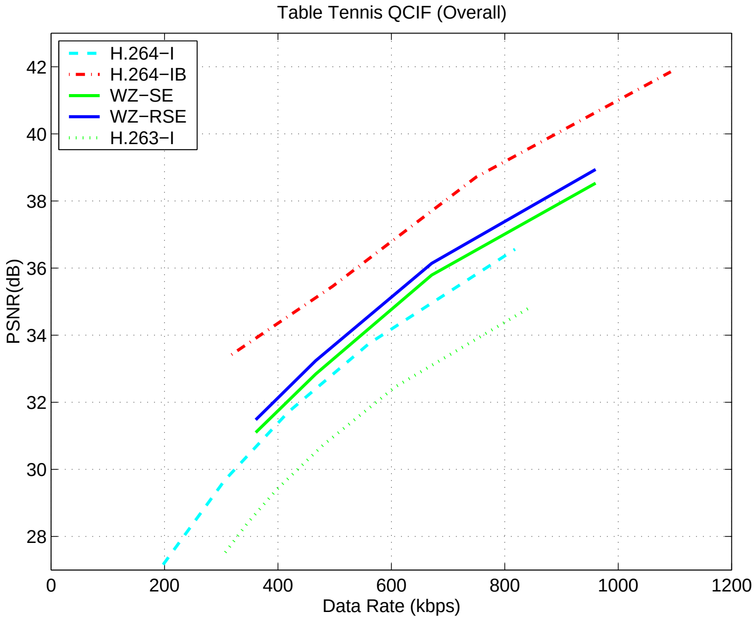
(a)

(b)

Fig. 2.14. Wyner-Ziv video decoder with a refined side estimator (*Stefan*) (a) Wyner-Ziv frame; (b) All frames.

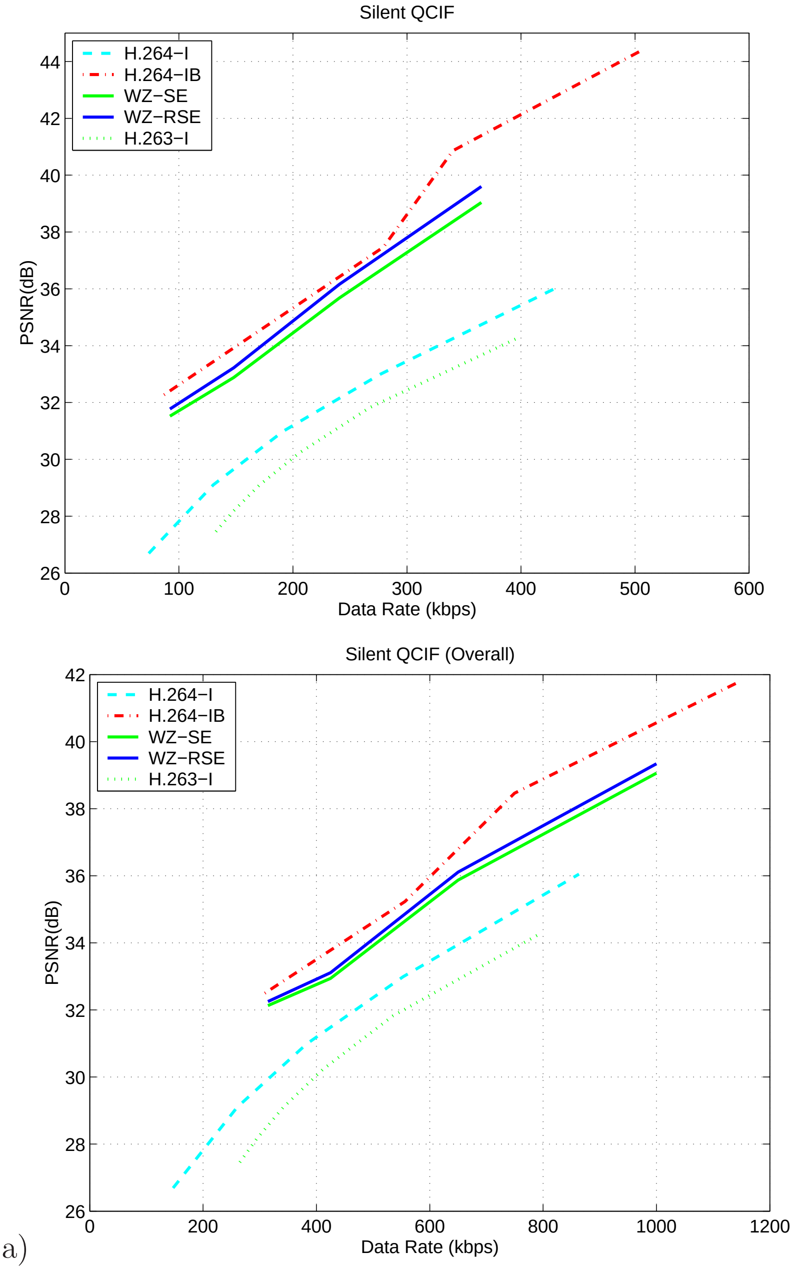


(a)



(b)

Fig. 2.15. Wyner-Ziv video decoder with a refined side estimator (*Table Tennis*) (a) Wyner-Ziv frame; (b) All frames.



(a)

(b)

Fig. 2.16. Wyner-Ziv video decoder with a refined side estimator (*Silent*) (a) Wyner-Ziv frame; (b) All frames.

We further investigate the use of two common motion search methods in side estimators. The results show that the use of sub-pixel motion search does not affect the coding efficiency as much as it does for conventional video coding. The use of sub-sampling motion search only results in small coding efficiency loss for Wyner-Ziv video coding. The use of a multi-reference can noticeably improve coding efficiency. These results are confirmed analytically and experimentally. We also presented a new Wyner-Ziv video decoder with a refined side estimator. The experimental results show that our scheme can improve coding efficiency by up to 2dB.

Current Wyner-Ziv video coding methods fall behind conventional MCP-based video coding. The improvement of the side estimator constitutes one of the most critical aspects in improving compression efficiency. A better understanding of the motion field and the motion vector is essential to improving the side estimator performance. The improvement of the side estimator will, undoubtedly, also bring more insight into motion estimation methods in conventional video coding.

3. WYNER-ZIV VIDEO CODING WITH UNIVERSAL PREDICTION

3.1 Introduction

Wyner-Ziv video coding has created considerable interest in recent years [83–85, 85, 89–100, 105]. As described in Chapter 2, reconstructed frames at the encoder and decoder can be regarded as the source of side information. Unlike conventional motion-compensated prediction (MCP) based video coding, side information is only available to the decoder but not to the encoder in Wyner-Ziv video coding. In MCP-based video coding, side information is analyzed in the encoder to reduce the redundancy in the input sequence. This generally involves a computationally intensive motion search and is hence inconvenient for applications that require a simple encoder such as a video camera phone or video surveillance. In contrast, Wyner-Ziv video coding shifts the side information analysis to the decoder and tries to maintain a comparable coding performance with MCP-based video coding.

To achieve this goal, Wyner-Ziv video coding needs to exploit video source statistics at the decoder. As shown in Chapter 2, the Wyner-Ziv video decoder first constructs a reference to serve as the initial estimate for the decoding of current frames. However, the construction of reference information without the original video sequence is difficult. Existing Wyner-Ziv video decoders, in general, resort to conventional motion estimators to extract motion information from the reconstructed video frames at the decoder. To do this, an underlying motion model needs to be assumed. Motion field research has provided many useful insights into the reconstruction of motion information without the original video sequence. For example, linear motion models can be used, in which it is assumed that the motion in the current frame is a continuous extension of the previous frames' motion. While this is true for some video sequences, the motion of natural video sequences is not well defined and a simple model can be inadequate.

Another downside to this method is the video decoder complexity. Since the motion estimation tasks are shifted to the decoder, Wyner-Ziv video coding requires much higher complexity at the decoder. Although this may not be a problem for video surveillance where computational complexity is less a concern at the decoder, it does pose a challenge for applications such as a wireless video camera phone, where both the sender and the receiver have limited computational resources. Some methods have been proposed to address this problem. For example, one of the methods proposes to use an intermediate transcoding server to process the video stream. The idea, as shown in Fig. 3.1, is to send the Wyner-Ziv encoded video stream to an intermediate server first. At this server the Wyner-Ziv video stream is decoded by a Wyner-Ziv video decoder and then re-compressed by a conventional MCP-based video encoder. The re-compressed video stream is then sent to the receiver to be decoded by the MCP-based video decoder. This method increases the transmission cost and delay, which eventually will lead to increased cost and delay on the receiver side.

In our work, we propose a new side information reconstruction method that is independent of the underlying video sequence model. This method, referred to as Wyner-Ziv video coding with Universal Prediction(WZUP), exploits the source statistics of the reconstructed video sequence and does not assume an underlying model of input sequence. As shown in Fig. 3.2, the goal is to construct a video codec with low coding complexity at both the encoder and the decoder while preserving comparable coding efficiency.

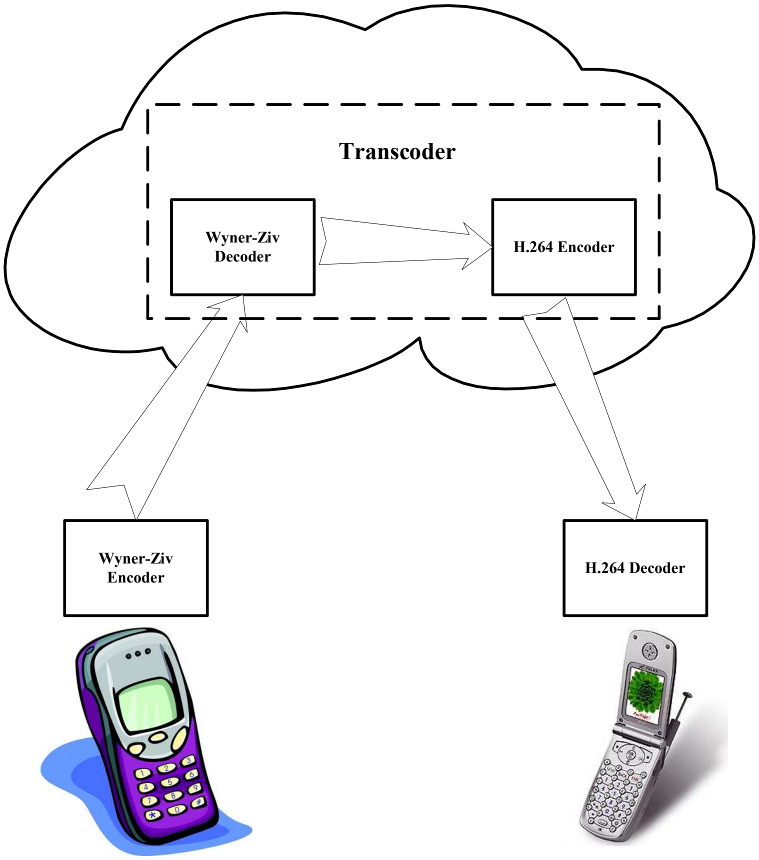


Fig. 3.1. Wyner-Ziv transcoder.

3.2 Wyner-Ziv Video Coding with Universal Prediction

In this Section we first present an introduction to universal prediction and then present our new side estimator with universal prediction.

3.2.1 Universal Prediction

The idea of universal prediction rises from the practice of predicting the next outcome of a sequence. We borrow the introduction of universal prediction from a survey by Merhav and Feder in [106].

Can the future of a sequence be predicted based on its past? If so, how good could this prediction be? These questions are frequently encountered in many applications.

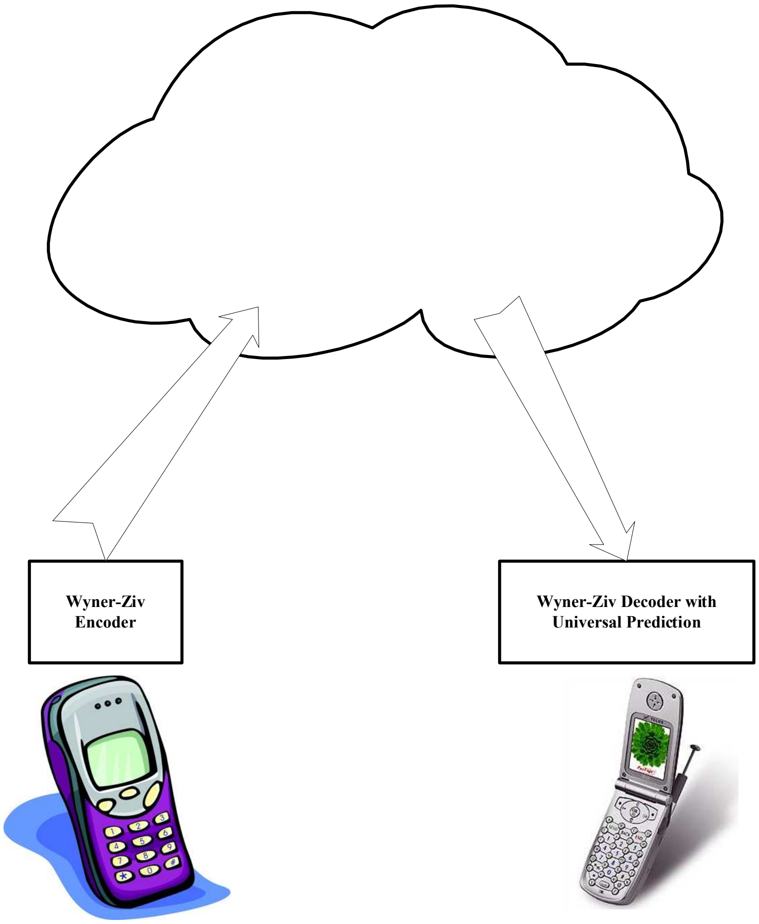


Fig. 3.2. Wyner-Ziv coding with universal prediction.

Generally speaking, one may wonder why should the future be at all related to the past. Evidently, often there is such a relation, and if it is known in advance, then it might be useful for prediction. In reality, however, the knowledge of this relation or the underlying model is normally unavailable or inaccurate, and this calls for developing methods of universal prediction. Roughly speaking, a universal predictor is one that does not depend on the unknown underlying model and yet performs essentially as well as if the model were known in advance.

One early example of universal prediction is Shannon's "mind-reading" machine inspired by Hagelbarger's "penny-matching" game [106]. Hence, universal prediction is closely related to universal lossless source coding, most notably Lemple-Ziv coding [107, 108]. It is also related to the research on universal gambling in [109]. The universal prediction problem was revisited in [110]. Although none of theis work has led to the optimal predictor, some finite-state predictors have been proposed and achieve sub-optimality [111–113]. A more detailed survey is presented in [106]. Applications of universal prediction includes lossless predictive compression, estimation, and denoising. Universal denoising was proposed in [114] with more research described in [115–118].

The universal prediction problem is formulated as follows. Consider an observer who receives a sequence of data $x_1, x_2, \cdots, x_{t-1}$ and wishes to predict the next outcome x_t subject to a loss metric $l(\cdot)$ defined on the predicted outcome $\hat{x}_t$ and the real outcome x_t. If the underlying statistical model of the data source is known and the prediction objective is well defined, classical statistical prediction theory can be used, among which are maximum likelihood prediction, maximum *a posteriori* (MAP) prediction, and Wiener prediction theory. In these cases, it is assumed that the data is generated by a source with a statistical model P. If however the underlying source statistics are not known, which is the case for many natural video sequences, the prediction solution is then not as well defined as the previous case. In this scenario, a universal prediction algorithm tries to estimate the underlying statistical characteristics of the sequence based on its observation of the past data.

In our work, we consider the reconstructed video data at the Wyner-Ziv video decoder as observations and the decoder tries to predict the outcome of the next video frame without knowing the statistical mechanism that generates the video source. This prediction will then serve as the initial estimate for the Wyner-Ziv video decoder.

3.2.2 Wyner-Ziv Video Coding

The Wyner-Ziv video coder we used is shown in Fig. 3.3. At the encoder each frame is coded either as an INTRA frame or a Wyner-Ziv frame. H.264 INTRA mode is used to code an INTRA frame. A Wyner-Ziv frame is coded in the spatial domain. After all pixels are quantized, the quantized pixels are coded bitplane by bitplane by a turbo coder. The parity bits are stored at the encoder and sent to the decoder upon the decoder's request. After receiving the parity bits, the decoder starts the decoding by first constructing an initial estimate of each frame. The initial estimate is constructed by the side estimator. The turbo decoder uses this initial estimate and incoming parity bits to decode the frame. It may request more parity bits from the encoder until a predetermined decoding accuracy is met.

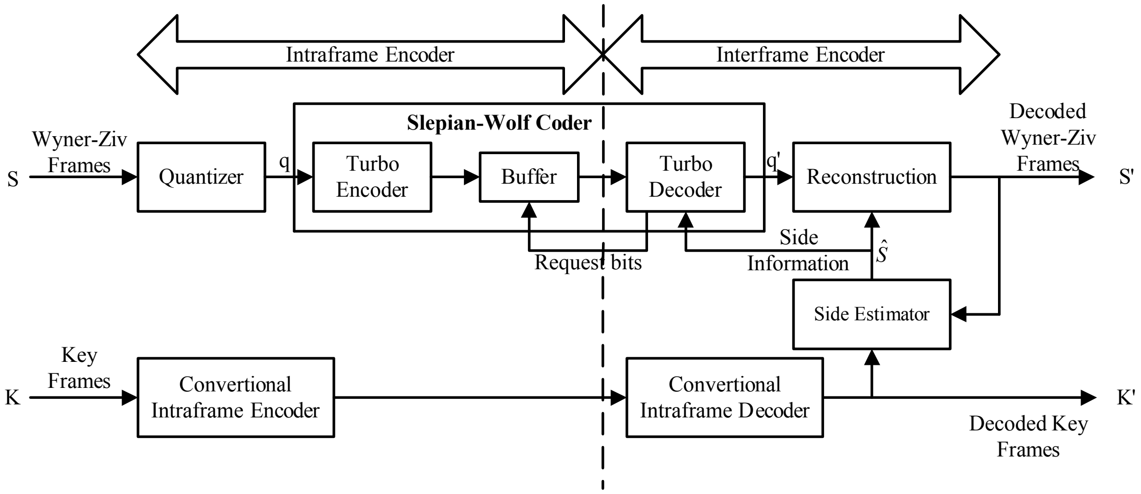


Fig. 3.3. Wyner-Ziv video coding.

3.2.3 A Side Estimator with Universal Prediction

We propose a new side estimator based on the universal prediction formulation. Consider each video frame as a vector and group the pixel values at the same spatial coordinate as $I(k,l)$, where (k,l) is the spatial coordinate inside a video frame. Without loss of generality, consider one of such $I(k,l)$, denoted as $X = x_1, x_2, \cdots, x_{t-1}, x_t$, where i in x_i denotes the temporal order in the sequence. And the loss function $\Lambda : [0,M] \times [0,M] \rightarrow [0,M^2]$ is $\Lambda = \Lambda(i,j), i,j \in [0,M]$, where $\Lambda(i,j)$ is the loss from estimating pixel value i with j. Denote the kth column of Λ as

λ_k and $\Lambda = [\lambda_0|\lambda_1|\cdots|\lambda_M]$. M is the maximum pixel value allowed, which is assumed to be 255. Denote the reconstructed $\hat{I}(k,l)$ at the decoder as $Z = z_1, z_2, \cdots, z_{t-1}, z_t$. z_t is an initial guess of the current reconstructed outcome. Since this initial guess is arbitrary and generally not reliable, the side estimator tries to provide a more accurate estimate of x_t. Therefore the side estimation can be formulated as a denoising problem, which can be solved by the method in [114].

Denote the transition matrix from X to Z as $\Pi = \Pi(i,j)_{i,j\in[0,M]}$ and $\Pi(i,j)$ is the probability when the input in X is i while the corresponding reconstructed value in Z is j.

Denote $P_{X_t|Z}$ as the conditional probability of x_t, whose αth component is $P(x_t = \alpha|Z = z)$. The optimal estimate of x_t is the one that minimizes the expected loss, i.e.,

$$\begin{aligned}\hat{X}^{opt}(z)[t] &= \arg\min_{\hat{x}\in[0,M]} \Sigma_{\alpha\in[0,M]}\Lambda(\alpha,\hat{x})P(x_t = \alpha|Z = z) \\ &= \arg\min_{\hat{x}\in[0,M]} \lambda_{\hat{x}}^T P_{X_t|z} = \arg\min_{\hat{x}\in[0,M]} \lambda_{\hat{x}}^T P_{X_t,z} \qquad (3.1)\end{aligned}$$

where

$$P_{X_t,z} = P_{X_t|z} \cdot P(Z = z) \qquad (3.2)$$

If we know the joint distribution $P_{X_t,z}$ of X_t and the reconstructed context z, the optimal estimator can be found readily by Lagrangian optimization and root finding methods. However, since we do not assume the statistical knowledge of the video sequence model in the decoder, this probability distribution is not available. Therefore we need to find a good estimate of $P_{X_t,z}$. Since

$$P(X_t = x_t, Z_t = z_t, Z^{n\setminus t} = z^{n\setminus t}) = P(X_t = x_t, Z^{n\setminus t} = z^{n\setminus t}) \cdot \Pi(x_t, z_t) \qquad (3.3)$$

where $z^{n\setminus t} = z_1, z_2, \cdots, z_{t-1}$. In vector form

$$P_{X_t,z} = \pi_{z_t} \odot P_{X_t,z^{n\setminus t}} \qquad (3.4)$$

Where $(u \odot v)[i] = u_i v_i$. Marginalize (3.4) with respect to X_t and iterate over all possible $z_t \in [0, M]$

$$P_{Z_t,z^{n\setminus t}} = \Pi^T P_{X_{t,zn\setminus t}} \qquad (3.5)$$

Hence

$$P_{X_t,z^n} = \pi_{z_t} \odot [\Pi^{-T} P_{Z_t,z^{n\backslash t}}] \tag{3.6}$$

And the optimal estimate is

$$\begin{aligned}\hat{X}^{opt}(z^n)[t] &= \arg\min_{\hat{x}\in[0,M]} \lambda_{\hat{x}}^T \pi_{z_t} \odot [\Pi^{-T} P_{Z_t,z^{n\backslash t}}] \\ &= \arg\min_{\hat{x}\in[0,M]} [P_{Z_t,z^{n\backslash t}}]^T \Pi^{-1} [\lambda_{\hat{x}} \odot \pi_{z_t}] \qquad (3.7)\end{aligned}$$

We now consider the case when the distortion is measured by mean square error(MSE), i.e.,

$$\Lambda(i,j) = (i-j)^2 \tag{3.8}$$

Also consider the simple case that the transition probability $\pi(x_t, z_t) = 1$ if $x_t = z_t$ or 0 if $x_t \neq z_t$. In this case the optimal estimator in (3.7) is a minimum mean square error (MMSE) estimator. Using Lagrangian optimization, (3.7) yields the MMSE estimator as

$$\hat{X}^{opt}(z^n)[t] = [P_{Z_t,z^{n\backslash t}}]^T \odot Z_t \tag{3.9}$$

i.e., the optimal estimator is a weighted average of the previous occurrence with the same context. The weighting coefficient is determined by the number of occurrences.

3.3 Experimental Results

In this Section we evaluate the performance of Wyner-Ziv video coding with universal prediction(WZUP). We compare it with the Wyner-Ziv video coding with MCP-based motion side estimator(WZ-MCP) and the conventional MCP- based hybrid video coding.

We first describe the encoder and decoder of WZ-MCP. To estimate the motion vector without using the original frame, one general assumption is to assume a linear motion model in the video sequence. An MCP-based side estimator was shown in Fig. 3.4. The INTRA frame is reconstructed with H.264 INTRA decoder. At the decoder, the side estimator estimates the reference using the forward motion search. As shown in Fig. 3.4, for every block to be decoded in the current Wyner-Ziv frame $s(n)$, its co-located block in the previous reconstructed INTRA frame $\hat{s}(n-1)$ searches for the best match in $\hat{s}(n-2)$ and obtains a motion vector MV_P. Assume the motion is linear, for the block to be decoded in the current frame, the estimated motion vector MV from $s(n)$ to $\hat{s}(n-1)$ is also MV_P. As shown in Chapter 2, conventional motion search methods may be used to further improve the side estimator accuracy.

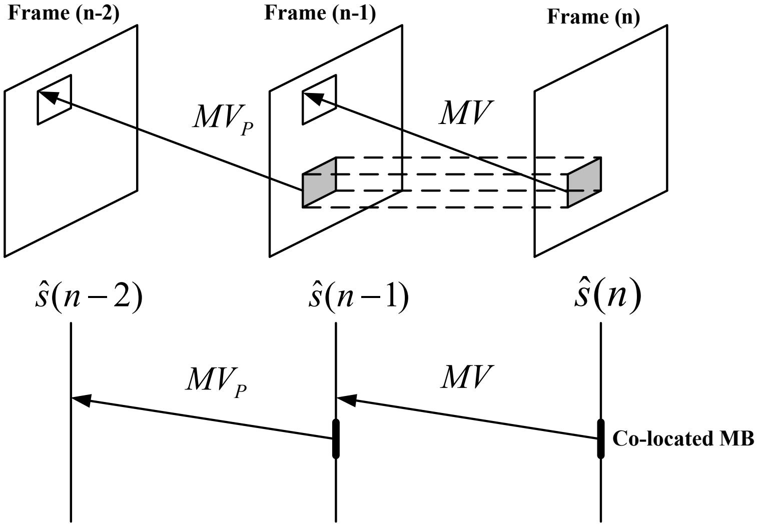


Fig. 3.4. MCP-based side estimator.

The side estimator in WZUP is constructed based on (3.9). For each pixel to be decoded at the decoder, we first collect its contexts in the previous frames. This context is the pixel values at the same spatial coordinate in the previous N frames, as shown in Fig. 3.5. In our experiment we set $N = 4$ and the context is $[z_{t-1}, z_{t-2}, z_{t-3}, z_{t-4}]$. The universal prediction side estimator is shown in Fig. 3.6. The decoder searches the occurrence of this context in the previous decoded data. For each matched context, it output $z_{t,i}$, which is the pixel value after this matched context. The initial estimate of the pixel to be decoded is the average of all these $z_{t,i}$'s.

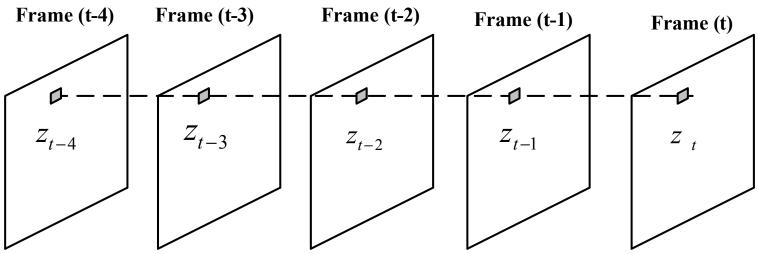


Fig. 3.5. Universal predication side estimator context.

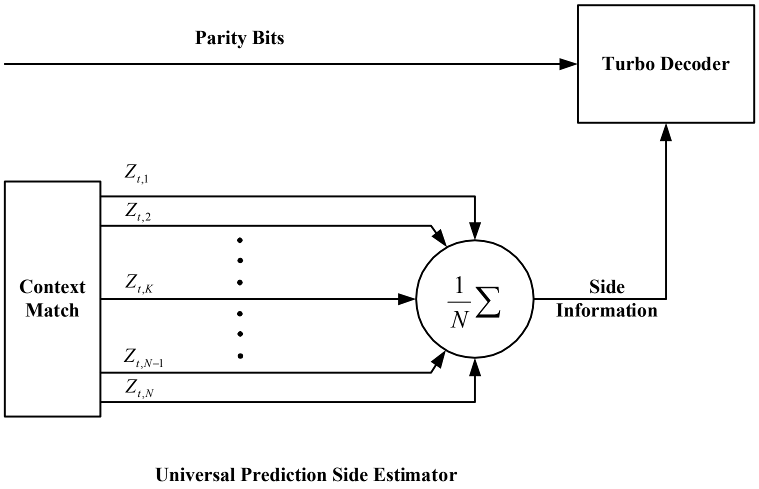


Fig. 3.6. Universal prediction side estimator.

For conventional video coding, we compare WZUP with hybrid video coding with the frame structure IPPPP and integer motion search.

As discussed in Chapter 2, to obtain a fair comparison of different motion search modes, we evaluate the PSNR of the references constructed by side estimators instead of the final coding efficiency. In general only the first three most significant bitplanes are decoded with the turbo coder. This is because the initial estimate only gives a rough guess of the true value. This initial estimate is generally not reliable enough to give a close estimate of the less significant bits. Hence to decode the less significant bitplanes is equivalent to correct data with high bit error rate (BER). The BER can be so large that it becomes inefficient to code these bitplanes. Instead, we can simply send these less significant data bits un-encoded. Therefore only the quality of the first bitplanes matters in the Wyner-Ziv video decoder and we evaluate the PSNR of the side motion estimator quality assuming all the bits in the less significant bitplanes are zero. For each z_{t-k} in the context, we only consider the first three bitplanes.

The results of using the universal prediction side estimator, motion side estimator and H.264 integer motion search are shown in Fig. 3.7-3.12. We compare the side estimators with different quality of reconstructed frames. As we can see, the H.264 integer motion search always outperforms the other two side estimators. This shows the access to the side information in the encoder does improve the estimation accuracy in the practical applications.

Comparing the universal prediction side estimator with the motion side estimator, in the *Foreman* and *Coastguard* sequences, the motion side estimator has a 2-3dB gain over universal prediction side estimator. This shows that in these two sequences the motion information extracted from the previous frames is a good indicator of the following frames. The difference is much smaller for the other two sequences *Mother and Daughter* and *Salesman*, while in the *Mobile* sequence the performance is nearly identical and the universal prediction side estimator even outperforms the motion side estimator in the *Carphone* sequence.

Consider the computational complexity. The universal prediction side estimator needs to maintain a context of length $N = 4$. Therefore there are 8^4 possible entries, where 8 is the number to store 3 bitplanes. For each entry, there are also 8 possible outputs after the matched context, which results in 32768 entries in total. This number is independent of the video frame size. Since the entry can be updated in real time, one does not need to store the previous frames. For the motion side estimator, one needs to store two previous frames, which is $176 \times 144 \times 2 = 50688$. This buffer budget increases with the frame size. For each pixel in the universal prediction side estimator, one needs to do a table lookup to find the 8 entries and compute the weighting average. After the pixel is decoded with the parity bits, one updates this context's entries. This is much faster than the motion side estimator, in which case each pixel has to be compared with every pixel in the search range(typically 8 × 8 or 16 × 16 grid) and then find the optimal motion vector.

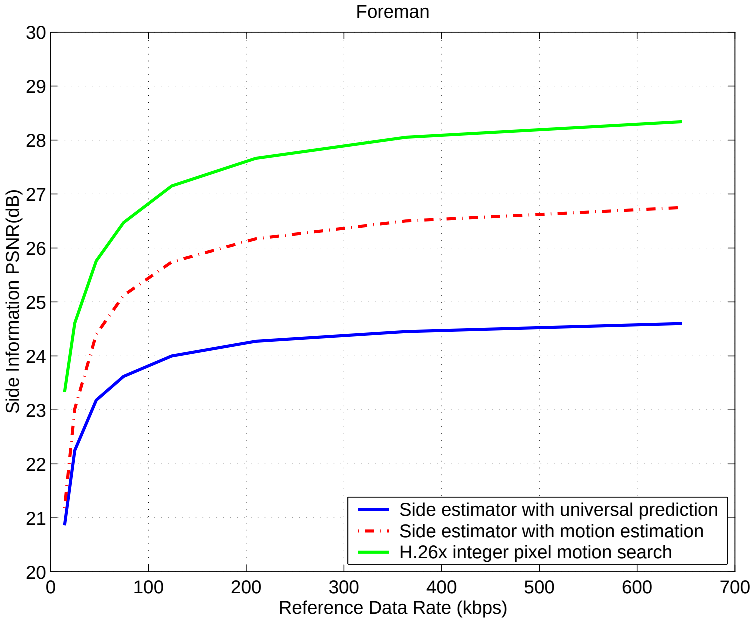


Fig. 3.7. PSNR of side estimator vs. reference data rate comparison (*Foreman*).

3.4 Conclusions

In this thesis, we present a new side estimator for Wyner-Ziv video coding. This new side estimator uses a non motion-search method to construct the initial estimate at the Wyner-Ziv video decoder. Our test results show that for sequences in which the motion can be predicted with the previous frames, the universal prediction side estimator is 2-3dB less than the conventional MCP-based video coding in terms of coding efficiency. However, for other sequences, the coding efficiency is rather close and sometimes the universal prediction side estimator even outperforms the MCP-based side estimator. This new method can significantly reduce the coding complexity at the decoder and therefore make it possible to design a codec with low computational complexity at both the encoder and the decoder.

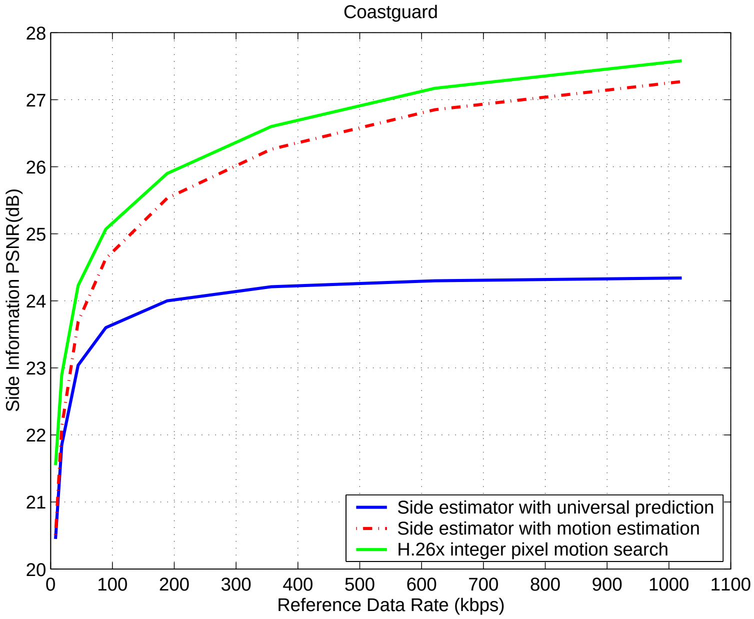


Fig. 3.8. PSNR of side estimator vs. reference data rate comparison (*Coastguard*).

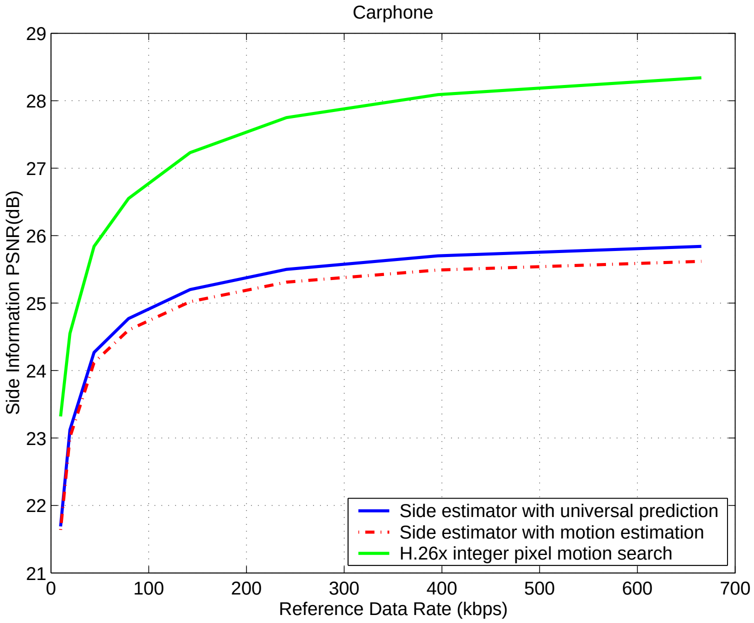


Fig. 3.9. PSNR of side estimator vs. reference data rate comparison (*Carphone*).

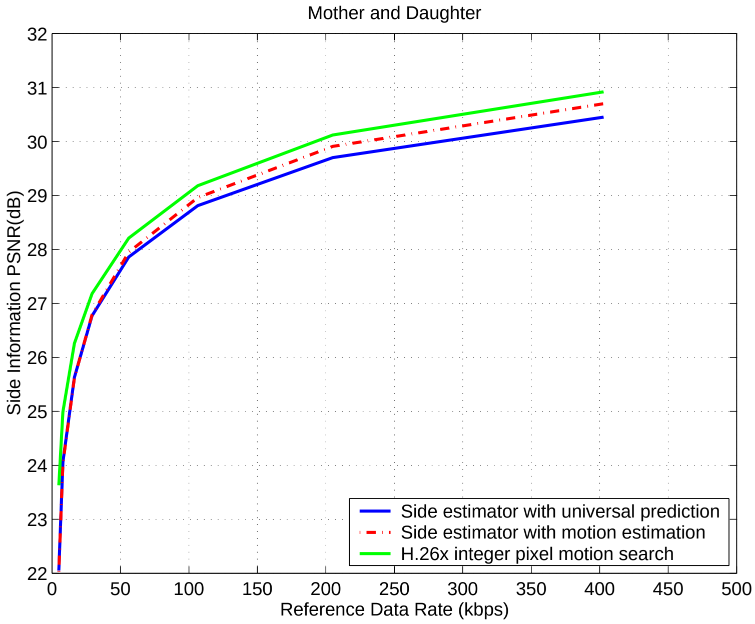


Fig. 3.10. PSNR of side estimator vs. reference data rate comparison (*Mother and Daughter*).

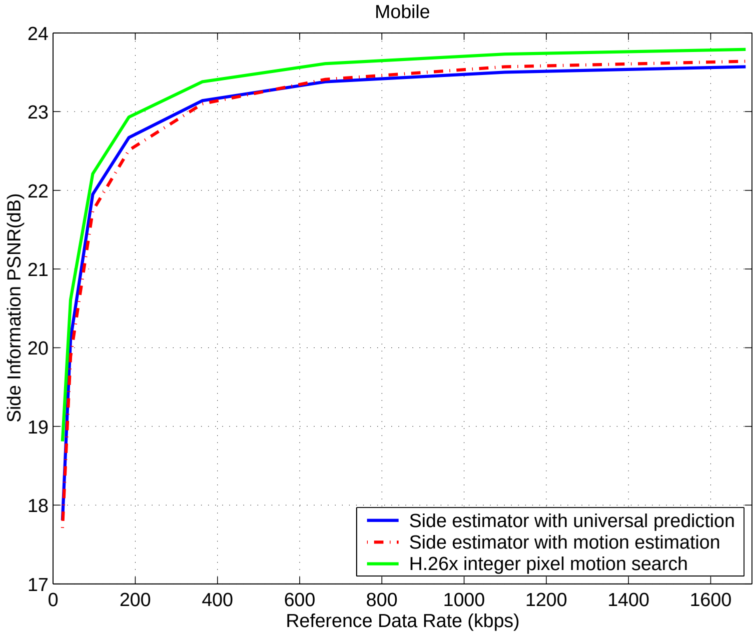


Fig. 3.11. PSNR of side estimator vs. reference data rate comparison (*Mobile*).

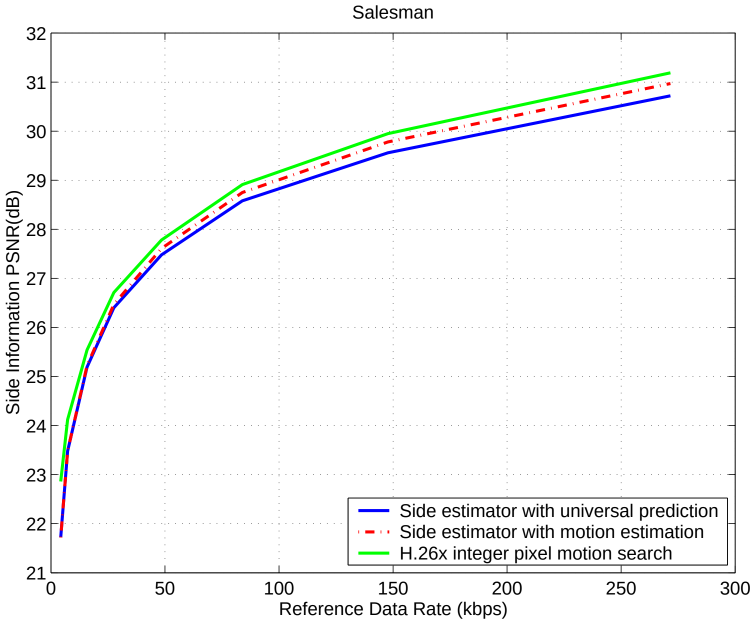


Fig. 3.12. PSNR of side estimator vs. reference data rate comparison (*Salesman*).

4. BLOCK ARTIFACT REDUCTION USING A TRANSFORM DOMAIN MARKOV RANDOM FIELD MODEL

4.1 Introduction

Transform coding is a key component in image and video coding standards. Among the many transforms that exists, the block-based Discrete Cosine Transform (BDCT) is often used [4, 6, 8–11, 53]. It is known that the BDCT may introduce block artifacts at low data rates that manifest themselves as an annoying discontinuity between adjacent blocks. This is a direct result of the independent transform and quantization of each block which fails to take into account the inter-block correlation.

A variety of research has been done to reduce block artifacts. Two types of methodologies are most frequently mentioned. One deals with the artifact inside the coding loop at the encoder, for example the in-loop filtering used in H.264 [53]. The other employs post-processing techniques at the decoder, where the decoded images or videos are post-processed to improve the visual quality. Post-processing approaches have been actively investigated as they maintain compatibility with the original coding structure, which is critical in many applications.

Many post processing methods have been studied to reduce the block artifact in the spatial domain. In recent video standards, such as H.264, low-pass in-loop filters are used across the block boundaries [53]. This technique is simple and effective. However, since the DCT transform coefficients may fall out of the Quantization

Constraint Set (QCS), it frequently results in over-blurred images [55]. Here the QCS of a coefficient y, denoted as $\Omega(y)$, is the set of integers defined by

$$\Omega(y) = \{x : Q(x) = Q(y), x \in Z\} \tag{4.1}$$

where $Q(\cdot)$ is the quantization operation and Z is the set of all integers. Iterative methods have been proposed to avoid violation of the QCS and provide more sophisticated artifact reduction. Spatial domain post-processed images are projected back into the transform domain at each iteration step to check if the DCT coefficients fall inside the QCS. If any post-processed coefficient is found outside of its QCS, then either clipping or other actions are taken. The transform coefficients are then projected back to the spatial domain using the inverse transform for the next iteration. One such example is the iterative post-processing methods based on projections onto convex sets (POCS), with QCS being one of the convex sets [55,56,119,120]. Other approaches also use an over-complete wavelet representation to reduce the block artifact by using multiscale edge analysis [57,121–123]. Finally, block artifact reduction can also be formulated as a maximum *a posteriori*(MAP) estimation problem, where the image is modeled as a Markov Random Field (MRF) [59,60]. This method is referred to as the spatial domain MRF (SD-MRF) MAP solution in this work.

One issue posed by the above spatial methods is computational complexity. The post-processing operations are used on the entire image at each iteration step. As such, the computational resources are distributed over the entire image uniformly. This means that the same computation resources consumed in earlier steps do not necessarily make more contribution for artifact reduction than those in later steps. From the perspective of implementation, such distribution of computational resources may not be efficient. A more desirable goal would be 1) reduce the overall post-processing complexity and/or 2) find a computationally progressive algorithm where the computational resources at early processing steps can contribute more to the visual quality improvement than those in later steps. By doing so, lower end computational devices will be allowed to more effectively address block artifact reduction.

Transform domain-based post-processing methods have also been investigated. Unlike the spatial domain post-processing techniques, the post-processing is performed on the DCT coefficients directly rather than on the pixels [61–63, 124–127]. The transform domain methods can be more efficient than the spatial domain techniques since the DCT compacts most of the energy in each block to the DC coefficients and the first few AC coefficients. Moreover, it is also known that the human visual system (HVS) is more sensitive to the distortions in the low frequencies than in the high frequencies [128]. Hence transform domain post-processing methods allow direct manipulation of only those DCT transform coefficients that affect the visual quality.

In this work, we exploit the advantages of both spatial domain and transform domain post processing methods by investigating a transform domain Markov random field (TD-MRF) model. The $L \times L$ DCT is interpreted as an L^2-subband transform. The transform coefficients at the same subband are grouped together and regarded as a subband image. MAP restoration is then performed on each subband image which is modeled by a MRF. Based on this model, we discuss two methods with different block artifact reduction scenarios in mind. In the first scenario, we target lower end post processing applications where computational complexity is a critical concern. We present a fast progressive TD-MRF solution. By "progressive" we mean that the block artifact is reduced at a faster rate with the same computational resources at earlier steps than that at later steps. Compared with SD-MRF, TD-MRF can reduce the computational complexity up to 90% with comparable visual improvement. For the second scenario, we consider the higher end post processing applications where visual quality is the major issue. In this case we present a framework, referred to as TSD-MRF, that sequentially combines TD-MRF and SD-MRF. Our results show that this new framework achieves better visual quality both objectively and subjectively over SD-MRF methods.

Here we pause to clarify the block artifact evaluation metric we use in this work. Unlike image or video coding algorithms where peak signal-to-noise ratio (PSNR) is widely accepted as a benchmark metric, a major difficulty in evaluating a block artifact reduction method is the lack of a universally accepted metric. PSNR is well recognized as inadequate for measuring block artifact alone. A block artifact metric, referred to as the mean square difference of slope (MSDS), was introduced and is frequently used in the research community [61]. MSDS is “the square of the difference between the slope across two adjacent blocks, and the average between the slopes of each of the two blocks close to their boundaries,” i.e.,

$$\begin{aligned} MSDS &= \sum_k \Big([x_{i,j}(k,0) - x_{i,j-1}(k,N-1)] \\ &- \frac{[x_{i,j-1}(k,N-1) - x_{i,j-1}(k,N-2)] + [x_{i,j}(k,1) - x_{i,j}(k,0)]}{2} \Big)^2 \end{aligned} \quad (4.2)$$

For pixel values $x_{i,j}(k,l)$, i and j are the horizontal and vertical index of the block

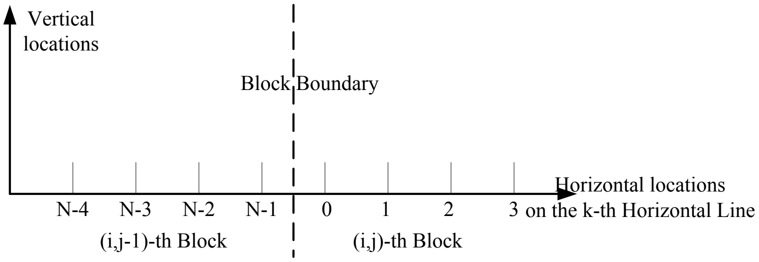


Fig. 4.1. Illustration of the MSDS along adjacent blocks.

in a image, with k and l being the horizontal and vertical index of the pixel inside the block as shown in Fig. 4.1. It can be shown that the MSDS increases after quantizing the DCT coefficients [61]. The MSDS metric was extended to include the four diagonally adjacent blocks [63]. This new metric is referred to as MSDS$_t$. In this work, we use MSDS and MSDS$_t$ for measuring block artifact reduction and PSNR for measuring the fidelity of the processed images. Since we emphasize the complexity reduction of our methods, the processing times are also provided. Finally, we include several test images and processed images to provide a subjective comparison.

The rest of this work is organized as follows. In Section 4.2, we formulate block artifact reduction as a MAP optimization problem with Markov random field model.

With this formulation, in Section 4.3 we revisit SD-MRF method and derive TD-MRF and TSD-MRF solutions respectively. These methods are evaluated and compared in Section 4.4 with experimental results on BDCT-coded test images. Section 4.5 concludes the Chapter with a brief remark on contributions and future work.

4.2 Block Artifact Reduction: MAP Optimizations Using A Markov Random Field Model

Block artifact reduction can be regarded as a Bayesian estimation problem, i.e., assume

$$Y = X + N \tag{4.3}$$

where X is the original image, Y is the observed blocky image and N is the block artifact noise. Then the MAP estimator is given by

$$\begin{aligned}\hat{x}_{MAP} &= \arg\max_x p(x|y) \\ &= \arg\max_x \log\frac{p(y,x)}{p(y)} \\ &= \arg\max_x \{\log p(y|x) + \log p(x)\} \qquad (4.4)\end{aligned}$$

The image probability density function $p(x)$ is modeled as a Markov random field. Here we borrow from [129] a brief tutorial on Markov random field models to facilitate our derivation. Interested readers are referred to [129–131] for a more detailed discussion.

For a pixel x inside a image with certain neighboring rules, a neighborhood system ∂x includes all the neighboring pixels of x. A neighborhood system ∂x is symmetric, i.e.,

$$x \in \partial y \Rightarrow y \in \partial x \tag{4.5}$$

Also $x \notin \partial x$. A clique c is a set of points, which are all neighbors of each other, i.e.,

$$\forall x, y \in c \Rightarrow y \in \partial x \tag{4.6}$$

An example of a 8-point neighborhood is shown in Fig. 4.2 with the corresponding 2-point cliques. The MRF model represents each image as a Gibbs distribution

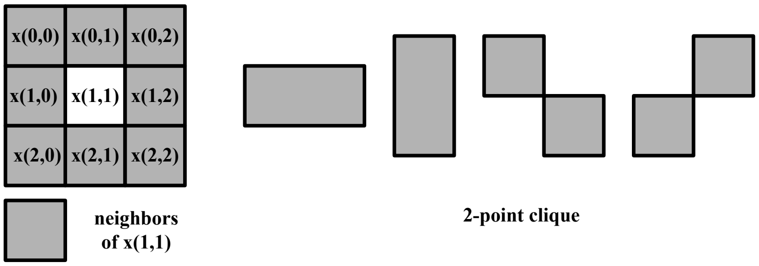


Fig. 4.2. A 8-point neighborhood system and its 2-point clique.

Table 4.1
g-function coefficient table.

$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$
$\frac{1}{6}$	0	$\frac{1}{6}$
$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$

$$p(x) = \frac{1}{Z} \exp\{-\sum_{c \in C} V_c(x_c)\} \tag{4.7}$$

where x_c is the value of pixel points in clique c, $V_c(x_c)$ is the potential function of x_c, C is the set of all cliques containing x and Z is the normalizing constant. There are various choices of potential functions in the literature, each of which defines an MRF model with different properties. In this work, we use an edge-preserving MRF proposed in [132], referred to as the Generalized Gaussian MRF (GGMRF). The GGMRF is defined as

$$\log p(x) = -\frac{1}{p\sigma_x^p} \sum_{(i,j) \in C} g_{(i,j)} |x - x'|^p \tag{4.8}$$

where x and x' are the pixel values at position i and j respectively, and $g_{(i,j)}$ is the Gaussian function coefficients and is normalized and shift invariant, i.e., $\sum_{j \in \partial i} g_{(i,j)} = 1$ and $g_{(i,j)} = g_{(j,i)}$. The following g as shown in Table 4.1 is used in our implementations, where x is at the center of the Table. Here p defines the order of the GGMRF. We use $p = 1.2$ as suggested in [132] to balance the edge-preserving property and block artifact reduction performance. More discussion on the selection of p can be found in [132]. σ_x^p can be estimated by [133]

$$\hat{\sigma}_x^p = \sum_{(i,j)\in C} g_{(i,j)} |y - y'|^p \tag{4.9}$$

4.3 SD-MRF, TD-MRF and TSD-MRF

With the MAP formulation given in (4.4) and the image formation described by the GGMRF model given in (4.8), we introduce two new transform domain solutions in this Section. Before we do this, we first briefly summarize the spatial domain MRF solutions from the literature.

4.3.1 SD-MRF Solution

In SD-MRF solutions, the noise N in (4.3) is assumed to have particular statistical distributions. One of the most commonly used models is to assume N is Gaussian noise with mean 0 and variance σ_N^2. Hence

$$\log p(y|x) = -\frac{1}{2\sigma_N^2}(y-x)^2 + \log\frac{1}{\sqrt{2\pi}\sigma_N} \tag{4.10}$$

Substitute (4.8) and (4.10) into (4.4), we have

$$\begin{aligned} \hat{x}_{MAP} &= \arg\max_x \{-\frac{1}{2\sigma_N^2}(y-x)^2 - \frac{1}{p\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x - x'|^p\} \\ &= \arg\min_x \{\frac{1}{2\sigma_N^2}(y-x)^2 + \frac{1}{p\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x - x'|^p\} \end{aligned} \tag{4.11}$$

By differentiating (4.11) with respect to x, we get

$$-\frac{1}{\sigma_N^2}(y-x) + \frac{1}{\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x-x'|^{p-1} sign(x-x') = 0 \tag{4.12}$$

Root finding techniques can then be used to find the optimal solution.

4.3.2 TD-MRF Solution

In this Section we derive the TD-MRF MAP solution. In classical BDCT coding techniques, the image or video frame of size $M \times N$ is divided into blocks of size $L \times L$ and then the 2-D DCT is obtained for each block $B_{(i,j)}$, where $i = 0, \ldots, \frac{M}{L}$ and $j = 0, \ldots, \frac{N}{L}$ are the horizontal and vertical index of the block respectively. The 2-D DCT coefficients of the (i,j)-th block $B_{(i,j)}$ can be obtained by

$$F^D_{(i,j)}(u,v) = \frac{2C(u)C(v)}{L}\left[\sum_{m=0}^{L-1}\sum_{n=0}^{L-1} f_{(i,j)}(m,n)\cos\left(\frac{2m+1}{2L}u\pi\right)\cos\left(\frac{2n+1}{2L}v\pi\right)\right] \tag{4.13}$$

where $f_{(i,j)}(m,n)$ is the (m,n)-th pixel value in the $B_{(i,j)}$ block for $u, v = 0, \ldots, L-1$, and for $L = 8$

$$C(u) = \begin{cases} \frac{1}{\sqrt{2},} & \text{if } u = 0; \\ 1, & \text{otherwise.} \end{cases} \tag{4.14}$$

After the DCT, the transform coefficients are quantized independently in each block. The quantized coefficients can be determined by

$$F^Q_{(i,j)}(u,v) = round\Big(\frac{F^D_{(i,j)}(u,v)}{\gamma Q(u,v)}\Big) \tag{4.15}$$

where $Q(u,v)$ is a quantization table and γ is a quantization parameter (QP) that controls the overall quality [4].

Correspondingly, the “dequantization” and the inverse DCT transform is defined as:

$$\hat{F}^D_{(i,j)} = \gamma F^Q_{(i,j)}(u,v)Q(u,v) \tag{4.16}$$

$$\hat{f}_{(i,j)}(m,n) = \frac{2}{L}\left[\sum_{u=0}^{L-1}\sum_{v=0}^{L-1} C(u)C(v)\hat{F}^D_{(i,j)}(u,v)\cos\left(\frac{2m+1}{2L}u\pi\right)\cos\left(\frac{2n+1}{2L}v\pi\right)\right] \tag{4.17}$$

It has been observed that DCT coefficients at the same frequency are highly correlated [134]. Hence we can group the DCT coefficients at the same frequency in different blocks together. We define each group as $F(u,v)$, where (u,v) is the frequency order of the coefficients. This DCT subband image formation is also used in [63]. $F(u,v)$ is a subband image of size $\frac{M}{L} \times \frac{N}{L}$. To illustrate the idea of the DCT subband image, we encode a 512×512 *lena* blocky image by a 2×2 DCT and its subband images are shown in Fig. 4.3.

As we can see from Fig. 4.3, when the original image is blocky, its subband images also demonstrate a degree of blockiness. Based on this observation, we conjecture that we may reduce the block artifact of the whole image by reducing the blockiness at each subband image. The basic idea behind TD-MRF is then to use MAP optimization (4.4) on each subband image sequentially.

In SD-MRF, the conditional probability $p(y|x)$ is derived based on the Gaussian noise assumption. We now examine this assumption in the transform domain. Denote the QCS of y as $\Omega(y)$. Notice that $p(y|x) = 0$ for $x \notin \Omega(y)$, and $p(y|x) = 1$ for all $x \in \Omega(y)$, hence $\log p(y|x) = -\infty$ for $x \notin \Omega(y)$ and $\log p(y|x) = 0$ for $x \in \Omega(y)$. Apparently, the Gaussian assumption is no longer valid and SD-MRF solution (4.11) cannot be used on each subband image directly. To maximize $\log p(y|x) + \log p(x)$, we will have to limit our search of $\hat{x}_{MAP}$ to inside $\Omega(y)$.

The image DCT AC coefficients at the same frequency order follow empirical statistical distributions. Many distributions have been studied in the literature, among which the Laplacian distribution remains a popular choice balancing simplicity and fidelity to empirical data [134–138]. The Laplacian model for the (u, v)-th subband image is defined as

$$p(x) = \frac{\lambda(u,v)}{2} \exp\{-\lambda(u,v)|x|\} \tag{4.18}$$

where $\lambda(u, v)$ is defined as

$$\lambda(u,v) = \begin{cases} 0, & \text{if } (u,v) = (0,0); \\ \sqrt{\frac{2}{\sigma_x^2(u,v)}}, & \text{otherwise.} \end{cases} \tag{4.19}$$

where $\sigma_x^2(u, v)$ is the variance of the subband image $F(u, v)$ and can be estimated by (4.9) with $p = 2$.

Combined with (4.4), this leads to the MAP formulation as maximize $\log p(x)$ subject to the QCS constraint $\Omega(y)$ in (4.1), the GGMRF constraint in (4.8) and the Laplacian model in (4.18). With Lagrangian optimization, the MAP estimator is given by:

$$\begin{aligned}\hat{x}_{MAP} &= \arg\max_{x\in\Omega(y)}\{-\lambda(u,v)|x| - \sum_{c\in C} V_c(x_c)\} \\ &= \arg\min_{x\ \in\Omega(y)}\{\lambda(u,v)|x| + \sum_{c\in C} V_c(x_c)\} \qquad (4.20)\end{aligned}$$

For the DC subband image, the MAP estimator is given by

$$\hat{x}_{MAP} = \arg\min_{x\in\Omega(y)}\{\frac{1}{p\sigma_x^p}\sum_{(i,j)\in C} g_(i,j))|x-x'|^p\} \qquad (4.21)$$

By differentiating (4.21) with respect to x, we obtain

$$\frac{1}{\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x-x'|^{p-1} sign(x-x') = 0 \qquad (4.22)$$

For AC subband images,

$$\hat{x}_{MAP} = \arg\min_{x\in\Omega(y)}\{\frac{\sqrt{2}}{\sqrt{\sigma_x^2(u,v)}}|x| + \frac{1}{p\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x-x'|^p\} \qquad (4.23)$$

By differentiating (4.23) with respect to x, we get

$$\frac{\sqrt{2}}{\sqrt{\sigma_x^2(u,v)}} sign(x) + \frac{1}{\sigma_x^p}\sum_{(i,j)\in C} g_{(i,j)}|x-x'|^{p-1} sign(x-x) = 0 \qquad (4.24)$$

A root finding technique can be then used to find the optimal MAP estimator in (4.24). A half interval search is used in our experiments. It is noted that any optimized root finding technique should work with both TD-MRF and SD-MRF solutions [139].

Both the SD-MRF and TD-MRF solutions treat the block artifact reduction as a Bayesian estimation problem with a MRF model. They are distinguished in the following aspects. First, the SD-MRF assumes a Gaussian noise model and uses this constraint on the restored image to balance the image fidelity and smoothness. The TD-MRF assumes a Laplacian model on the DCT AC coefficients and uses this constraint to address the image fidelity. Hence, a more accurate noise model will improve the SD-MRF quality, while a more accurate AC coefficient model will help the TD-MRF solution. Second, in SD-MRF, the MAP optimization is done on the entire image in the spatial domain; while in TD-MRF, the MAP solution is obtained on each subband image sequentially. Since the DCT transform compacts the energy in the first few coefficients and the human visual system is more sensitive to lower frequency than higher frequencies, the TD-MRF can achieve artifact reduction progressively.

4.3.3 TSD-MRF Solution

The TSD-MRF solution is a simple extension by combining the TD-MRF solution and SD-MRF solution serially. Despite its simplicity, our experimental results shows that it can achieve significant visual quality improvement. For the TSD-MRF solution,

- On a $M \times N$ blocky image, use the TD-MRF procedure and obtain a processed image;
- On the TD-MRF processed image, use the SD-MRF procedure.

4.4 EXPERIMENTAL RESULTS

In this Section, we compare these three methods in the following four aspects:

- block artifact reduction measured by MSDS and MSDS$_t$;
- image fidelity measured by PSNR;
- computational complexity in terms of processing time in millisecond on a Pentium III 933MHz computer with 256MB RAM;
- visual subjective quality by showing the original image, the blocky image and the processed image.

The algorithms are implemented by Visual C++ 6.0. We tested the following four 512×512 TIFF images(*Lena, Peppers, Couple, Boat*) and one 256×256(*Cameraman*) from the USC-SIPI data base [140]. Each image is encoded using JPEG by the VcDemo reference software [141]. Two data rates, 0.2bpp and 0.3bpp, are tested for each image. MSDS, MSDS$_t$, PSNR and processing time are compared in Table 4.2 - 4.6. We obtained the processing time for each subband in the TD-MRF solution. Our experimental results show that the MSDS, MSDS$_t$ and PSNR of the spatial domain reconstructed images generally stop changing after the first 20-25 subband images are processed. So the processing time for TD-MRF is accumulated to the subband when these metrics stop changing.

As we can see from the results, SD-MRF can improve the PSNR by 0.2-0.6dB, while TD-MRF and TSD-MRF will lose 0.1-1dB in PSNR. In MSDS and MSDS$_t$, SD-MRF also generally leads to more reductions than TD-MRF. This is particularly true when the JPEG encoded image is less blocky, i.e., at higher data rates such as 0.3bpp. Meanwhile, TSD-MRF leads to lower MSDS and MSDS$_t$ than both SD-MRF and TD-MRF.

In terms of computational complexity, TD-MRF uses less processing time compared with SD-MRF. To quantize this comparison, we normalize the TD-MRF processing time by the entire image's SD-MRF processing time. We also normalize the TD-MRF MSDS reduction with the SD-MRF MSDS reduction. We show the normalized MSDS reduction versus normalized processing time with the increase of number of subband images in Fig. 4.4 and Fig. 4.5. As we can see from Fig. 4.4, TD-MRF uses $10-15\%$ of the processing power but can achieve $75-100\%$ of MSDS reduction compared with the SD-MRF solution. In the case of JPEG coding with a 0.3bpp data rate, the TD-MRF method generally uses $15-25\%$ processing power and achieves $45-75\%$ MSDS reduction. From this comparison, we can conclude that TD-MRF is more effective in reducing complexity for images encoded at a lower data rate. This may serve low-end devices, such as mobile telephones, well in the sense that low-end computational resources are often associated with smaller bandwidth or storage spaces in these applications. It is noted that the PSNR of TD-MRF processed images does not change much with the increase of the subband numbers. For TSD-MRF, its computational complexity is basically the sum of TD-MRF and SD-MRF. Another observation is the time for TD-MRF to process each subband image. Each subband image is $\frac{1}{64}$ of the size of the whole image. Denote the time for SD-MRF to process the whole image as T_s. For DC subband image, it generally takes a little more than $\frac{T_s}{64}$, while each of the first few AC subband images takes a little less than $\frac{T_s}{64}$. The processing accelerates with the increase of frequency. Beyond 20-25 subbands, the subband images are virtually flat with sparse large values and TD-MRF processing generally has marginal effect on these subband images.

Although the MSDS, MSDS$_t$, and PSNR results show that our methods can reduce the block artifact significantly without endangering the fidelity of the original image, we have to be careful here to make sure that the MSDS reduction does not lead to over-blurring, one artifact that many block artifact reduction algorithms frequently suffer from. This concern is especially valid in the case of TSD-MRF, where we observe that MSDS of the TSD-MRF is sometimes even lower than the MSDS of the original image. While this can be interpreted as over-blurring objectively, we find in Fig. 4.8-(d) and 4.10-(d) that this is not necessarily true subjectively. We demonstrate two test images(*Lena* and *Peppers*) with their blocky images and processed images shown side by side. In the 0.2bpp case, the SD-MRF and TD-MRF have comparable visual quality, and TSD-MRF is significantly better. In the 0.3bpp case, SD-MRF and TSD-MRF have very slightly better quality than TD-MRF.

4.5 Conclusions

To conclude, we presented a transform domain Markov Random Field model in this work to address the block artifact reduction problem in BDCT-based image compression. We presented two methods, namely TD-MRF and TSD-MRF, based on this model. We showed by objective and subjective comparisons that TD-MRF can

Table 4.2
MSDS, MSDS$_t$, PSNR and processing time (PT) comparison (*Lena* 512 × 512).

Title	MSDS	MSDS$_t$	PSNR (dB)	PT (ms)
Original	1257	1624	N/A	N/A
JPEG Encoded (0.2 bpp)	5299	6674	28.90	N/A
SD-MRF (0.2 bpp)	3078	4055	29.44	48072
TD-MRF (0.2 bpp)	3241	4071	27.92	8066
TSD-MRF (0.2 bpp)	1694	2229	27.80	57063
JPEG Encoded (0.3 bpp)	3435	4384	31.68	N/A
SD-MRF (0.3 bpp)	1806	2427	32.19	41369
TD-MRF (0.3 bpp)	2511	3192	30.84	11030
TSD-MRF (0.3 bpp)	1220	1641	30.49	51270

Table 4.3
MSDS, MSDS$_t$, PSNR and processing time (PT) comparison (*Peppers* 512 × 512).

Title	MSDS	MSDS$_t$	PSNR (dB)	PT (ms)
Original	2079	2418	N/A	N/A
JPEG Encoded (0.2 bpp)	5150	6341	28.03	N/A
SD-MRF (0.2 bpp)	2863	3663	28.60	42322
TD-MRF (0.2 bpp)	3407	4200	27.44	8510
TSD-MRF (0.2 bpp)	1724	2212	27.45	52030
JPEG Encoded (0.3 bpp)	2799	3524	31.28	N/A
SD-MRF (0.3 bpp)	1357	1799	31.79	41431
TD-MRF (0.3 bpp)	2145	2697	30.53	10531
TSD-MRF (0.3 bpp)	998	1322	30.27	52270

Table 4.4
MSDS, MSDS$_t$, PSNR and processing time (PT) comparison (*Couple* 512 × 512).

Title	MSDS	MSDS$_t$	PSNR (dB)	PT (ms)
Original	3654	4554	N/A	N/A
JPEG Encoded (0.2 bpp)	10102	12273	25.44	N/A
SD-MRF (0.2 bpp)	6318	7867	25.84	41790
TD-MRF (0.2 bpp)	6413	7803	25.32	6660
TSD-MRF (0.2 bpp)	3626	4538	25.33	49400
JPEG Encoded (0.3 bpp)	8108	10097	27.67	N/A
SD-MRF (0.3 bpp)	4808	6210	28.13	41369
TD-MRF (0.3 bpp)	6110	7604	27.59	10350
TSD-MRF (0.3 bpp)	3451	4423	27.59	54430

Table 4.5
MSDS, MSDS$_t$, PSNR and processing time (PT) comparison (*Boat* 512 × 512).

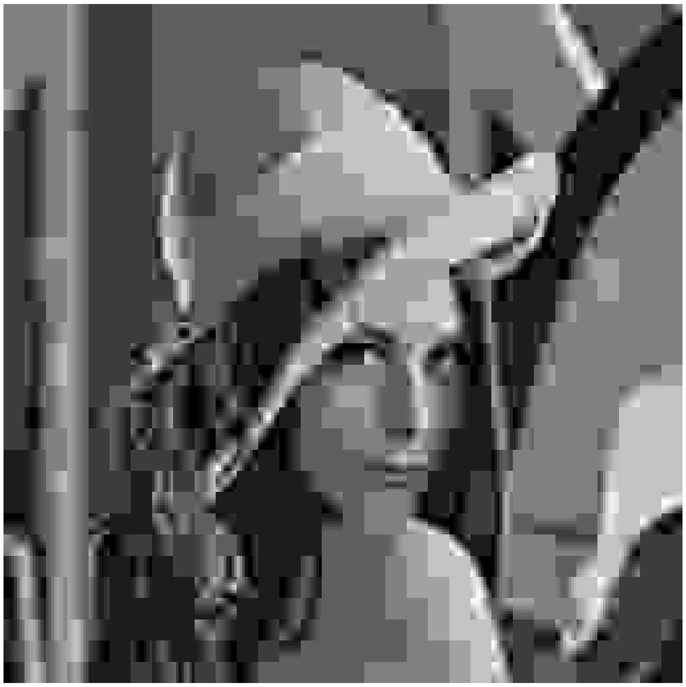
Title	MSDS	MSDS$_t$	PSNR (dB)	PT (ms)
Original	3911	4767	N/A	N/A
JPEG Encoded (0.2 bpp)	8878	10947	26.24	N/A
SD-MRF (0.2 bpp)	5589	7095	26.67	41060
TD-MRF (0.2 bpp)	5498	6755	25.87	8984
TSD-MRF (0.2 bpp)	3129	3969	25.84	51020
JPEG Encoded (0.3 bpp)	7176	8965	28.13	N/A
SD-MRF (0.3 bpp)	4285	5525	28.57	43225
TD-MRF (0.3 bpp)	5315	6620	28.02	10770
TSD-MRF (0.3 bpp)	2989	3844	27.95	54270

Table 4.6

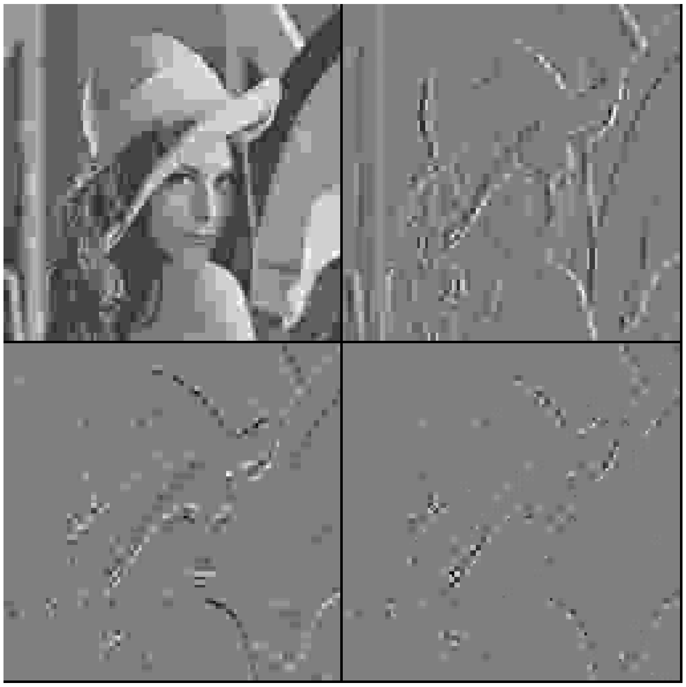
MSDS, MSDS$_t$, PSNR and processing time (PT) comparison (*Cameraman* 256 × 256).

Title	MSDS	MSDS$_t$	PSNR (dB)	PT (ms)
Original	2278	2767	N/A	N/A
JPEG Encoded (0.2 bpp)	4325	5164	22.32	N/A
SD-MRF (0.2 bpp)	3336	4025	22.56	9974
TD-MRF (0.2 bpp)	3232	3866	22.27	1760
TSD-MRF (0.2 bpp)	2399	2904	22.36	11935
JPEG Encoded (0.3 bpp)	3440	4276	25.77	N/A
SD-MRF (0.3 bpp)	2565	3232	26.06	10826
TD-MRF (0.3 bpp)	2787	3456	25.69	1810
TSD-MRF (0.3 bpp)	2032	2554	25.70	12045

substantially relax the computational complexity constraint compared with SD-MRF and still achieve significant block artifact reduction. We noted that TD-MRF generally cannot achieve as much block artifact reduction as SD-MRF. This is because in highly quantized images, almost all high frequency coefficients are truncated to zero. Hence, loss incurred to high frequency coefficients cannot be fully recaptured if subbands are processed separately as in TD-MRF. In the meantime, the TSD-MRF framework that combines the TD-MRF and SD-MRF leads to more significant block artifact reduction without over-blurring the image.



(a)



(b)

Fig. 4.3. Subband representation of the DCT coefficients. (a) Blocky image; (b) Subband representation: (i) top left: DC subband image; (ii) top right: (0,1)-th subband image; (iii) bottom left: (1,0)-th subband image; (iv) bottom right: (1,1)-th subband image.

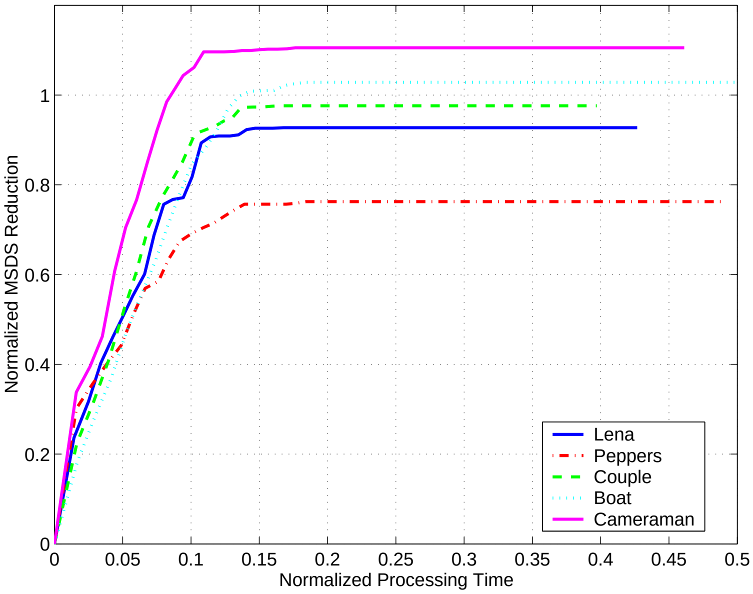


Fig. 4.4. Normalized comparison (0.2bpp).

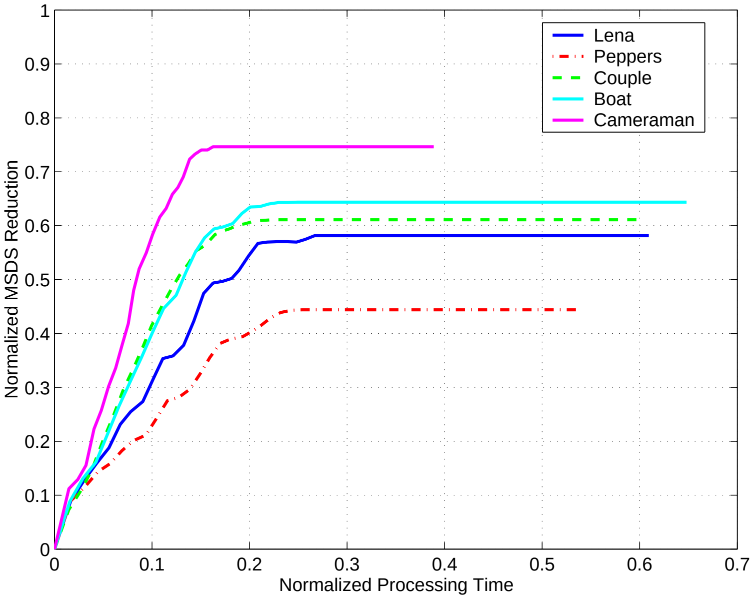
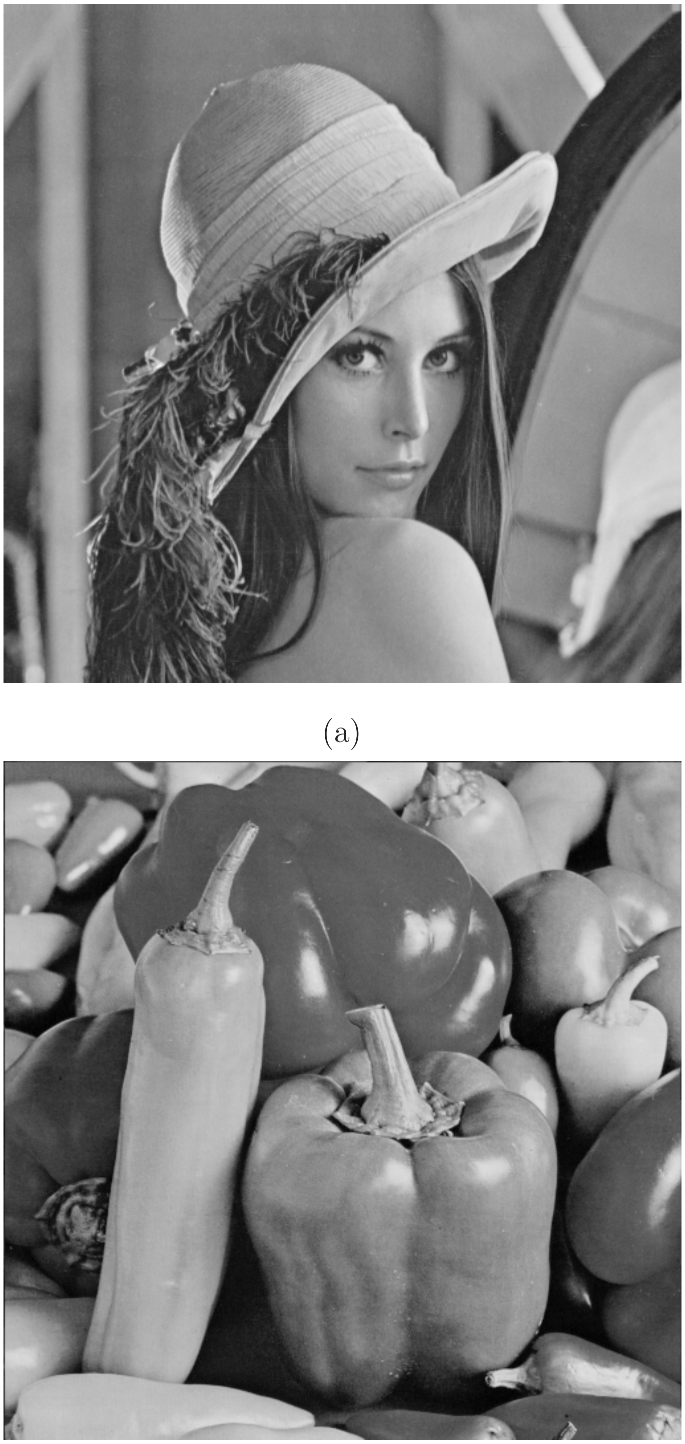


Fig. 4.5. Normalized comparison (0.3bpp).



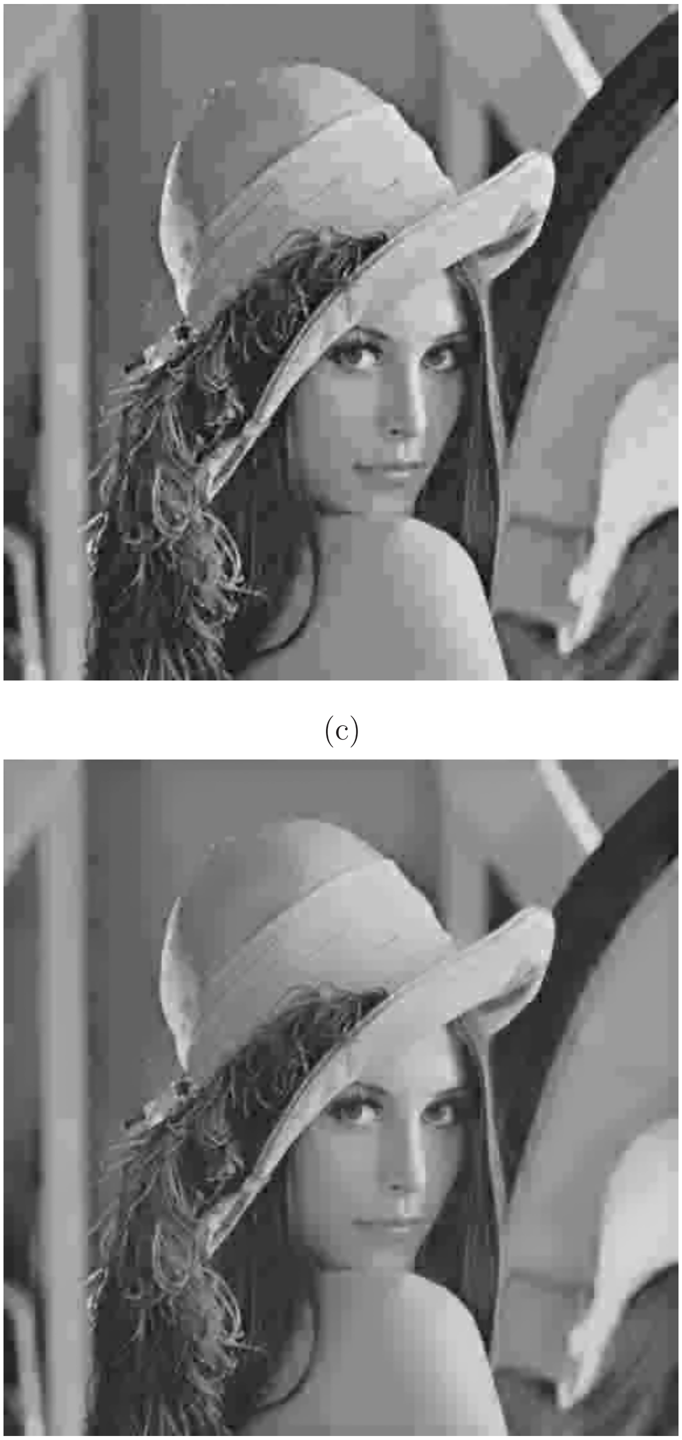
(a)

(b)

Fig. 4.6. Original test images. (a) *Lena* (512 × 512); (b) *Peppers* (512 × 512).

(a)

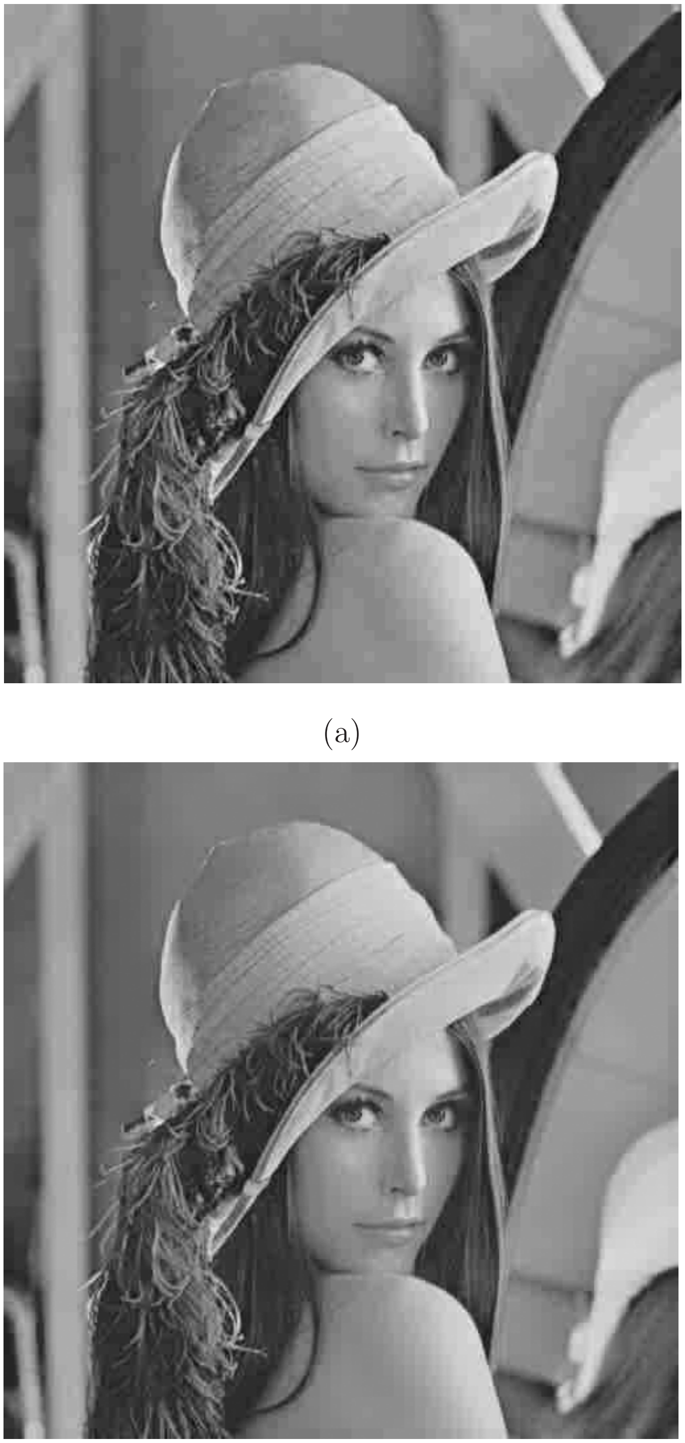
(b)



(c)

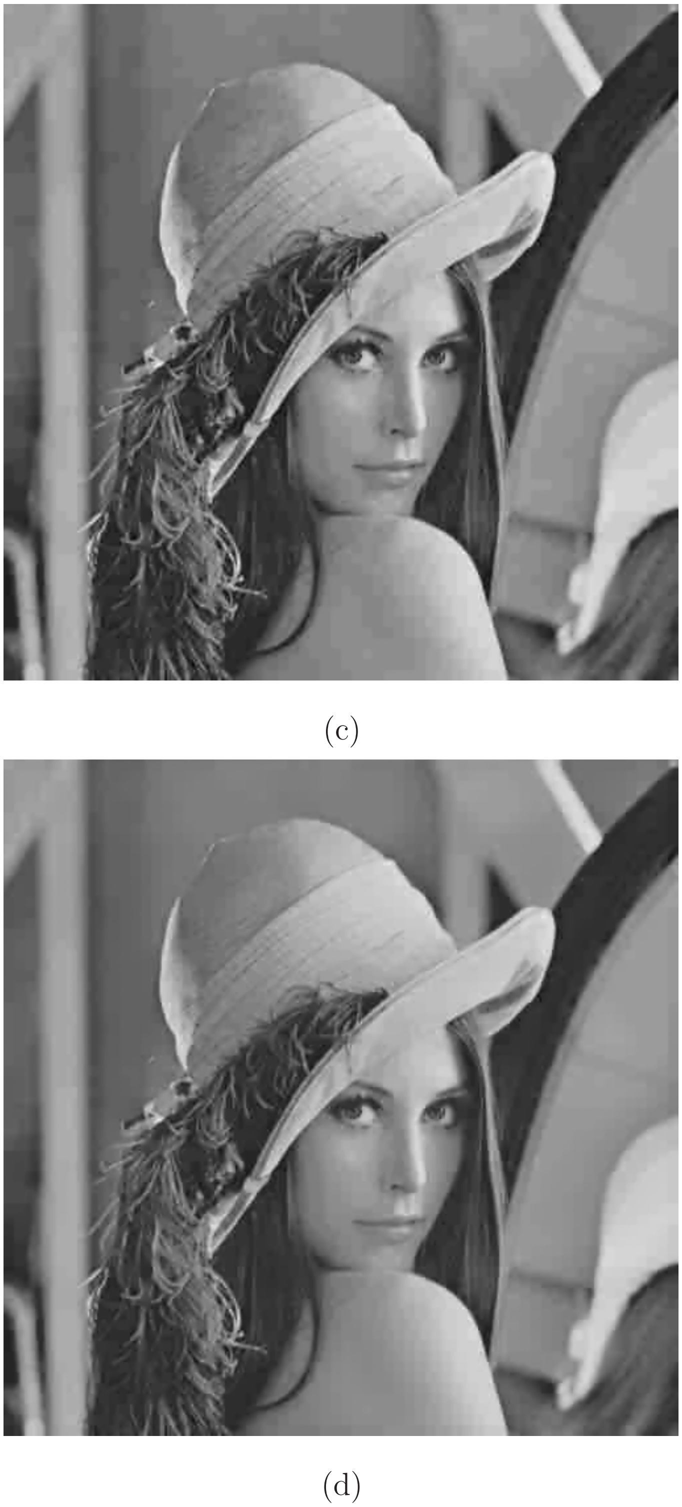
(d)

Fig. 4.7. Comparisons of the SD-MRF, TD-MRF and TSD-MRF methods (*Lena* 0.2bpp). (a) JPEG encoded image(0.2bpp); (b) TD-MRF (c) SD-MRF; (d) TSD-MRF.



(a)

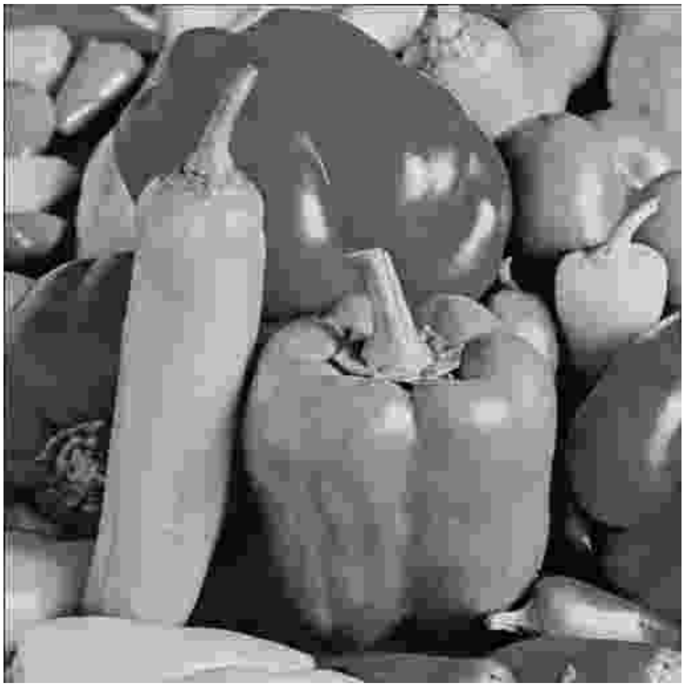
(b)



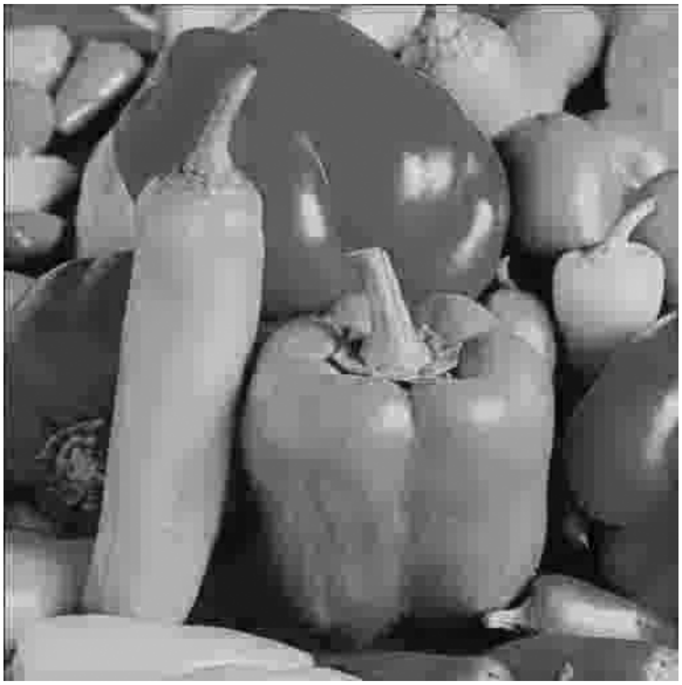
(c)

(d)

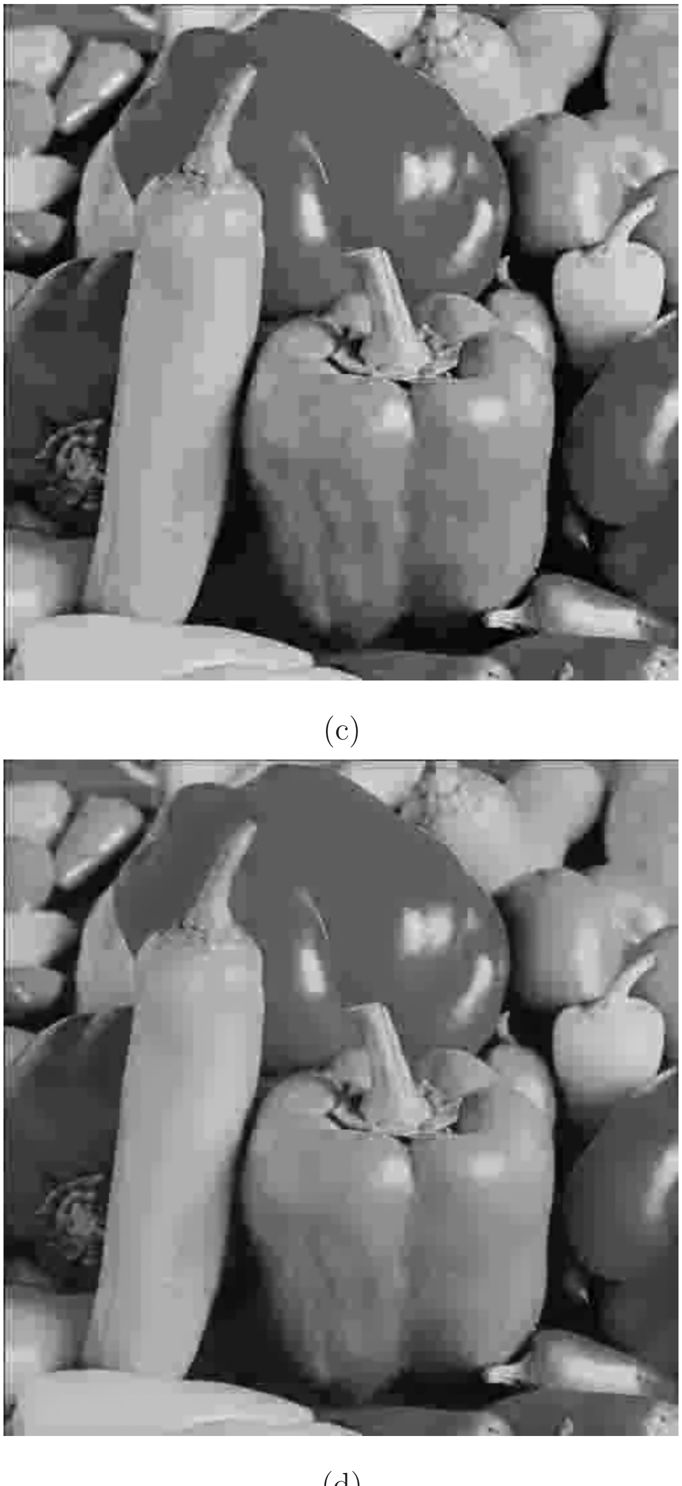
Fig. 4.8. Comparisons of the SD-MRF, TD-MRF and TSD-MRF methods (*Lena* 0.3bpp). (a) JPEG encoded image(0.3bpp); (b) TD-MRF (c) SD-MRF; (d) TSD-MRF.



(a)



(b)



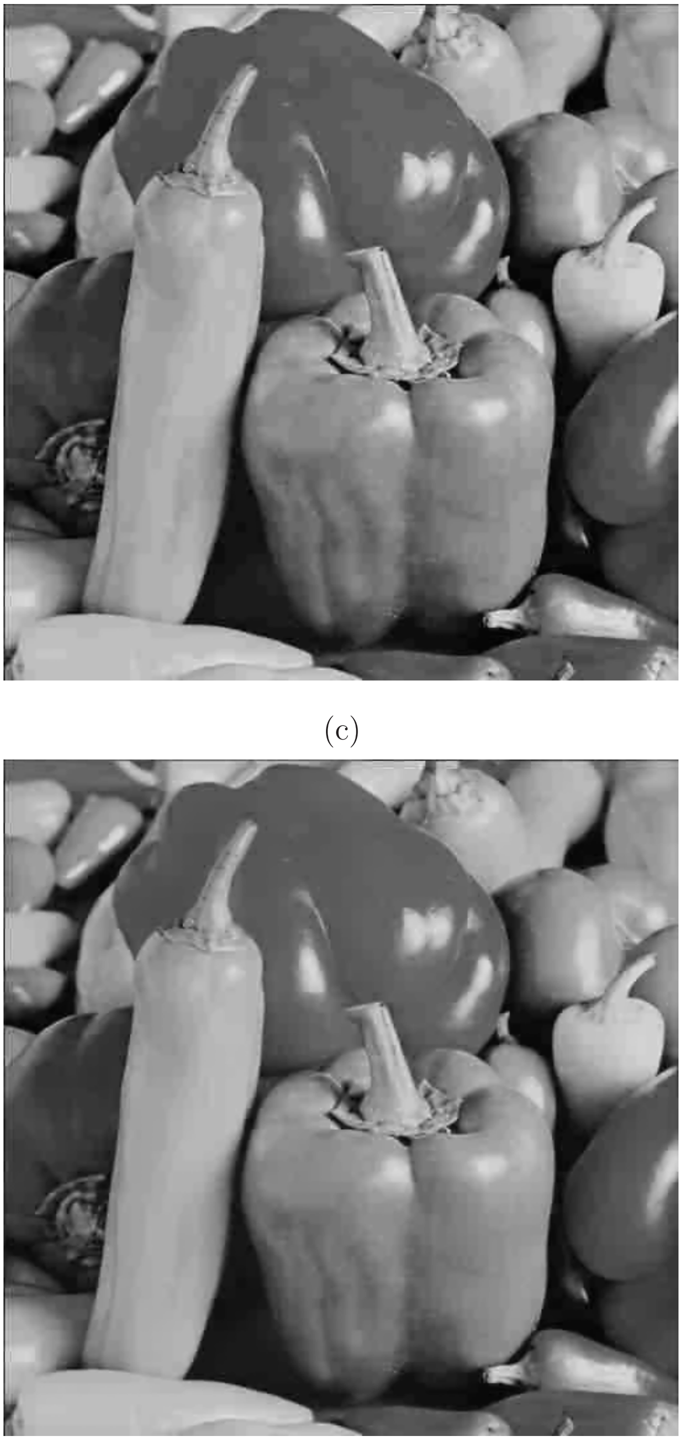
(c)

(d)

Fig. 4.9. Comparisons of the SD-MRF, TD-MRF and TSD-MRF methods (*Peppers* 0.2bpp). (a) JPEG encoded image(0.2bpp); (b) TD-MRF (c) SD-MRF; (d) TSD-MRF.

(a)

(b)



(c)

(d)

Fig. 4.10. Comparisons of the SD-MRF, TD-MRF and TSD-MRF methods (*Peppers* 0.3bpp). (a) JPEG encoded image(0.3bpp); (b) TD-MRF (c) SD-MRF; (d) TSD-MRF.

5. CONCLUSIONS

Advances in video compression and video processing have made it possible to deliver high quality video for many applications. Rate and distortion tradeoffs have long been a major concern in video coding. Within the last two decades, video quality has significantly improved while data rate and cost continues to decrease. Many researchers have pondered the future of video compression. With many more new technologies on the horizon, there are also many new video applications that pose new research challenges. We believe, along with the rate distortion tradeoff, low complexity and low latency will also become a very challenging problem for many applications [50, 52, 64–77]. In this dissertation we examined the design of low complexity video compression and processing methods.

5.1 Contributions of This Dissertation

- **Rate Distortion Analysis of Wyner-Ziv Video Coding**

  We addressed the rate-distortion performance of Wyner-Ziv video coding. A theoretic analysis of Wyner-Ziv video coding was developed. We studied the Wyner-Ziv video coding performance and compared it with conventional motion-compensated prediction (MCP) based video coding. We showed that theoretically Wyner-Ziv video coding may outperform by up to 0.9bit/sample DPCM-frame video coding. For most practical sequences with low motion correlation across the sequences, the rate saving is less significant. When compared with INTER-frame video coding, Wyner-Ziv video coding leads to a rate increase of as much as 0.2-0.6bit/sample for sequences with small motion vector variances and 1bit/sample for those with large motion vector variances.

The results show that the simple side estimator cannot meet the expectation of practical applications and more effective side estimation is needed.

- **Conventional Motion Search Methods Revisited**

  We studied the use of conventional motion search methods for side estimators. The results showed that the use of sub-pixel motion search does not affect the coding efficiency as much as it does in conventional video coding. Also the use of sub-sampling motion search only results in small coding efficiency loss with Wyner-Ziv video coding. This indicates that for side estimators that cannot locate the rough position of the true motion vectors in the first place, it makes little sense to further refine the motion vector by using sub-pixel motion search. For decoders with limited computing capability, a 2:1 or even coarser subsampling is worth consideration. We also showed that the use of multi-reference motion search can effectively improve coding efficiency. Side estimators with two reference frames can outperform those with one reference frame by 0.2bit/sample or more. More data rate reduction can be achieved with more reference frames.

- **Wyner-Ziv Video Coding with a Refined Side Estimator**

  We presented a new Wyner-Ziv video decoder with a refined side estimator. The goal of this new decoder is to better exploit the motion correlation in the reconstructed video sequence and hence improve the coding efficiency. The idea of refined side estimator is to progressively extract side information from both previous reconstructed frames and the current partially decoded Wyner-Ziv frame. The experimental results showed that our design can improve the coding quality by as much as 2dB in PSNR.

- **Wyner-Ziv Video Coding with Universal Side Estimator**

  We proposed a novel decoding structure with a universal prediction side estimator. The goal of this decoder is to reduce the decoding complexity at the

Wyner-Ziv decoder and hence make it possible to design an encoder and decoder with low complexity. This new side estimator uses a non motion-search method to construct the initial estimate at the Wyner-Ziv video decoder. The test results show that for sequences in which the motion can be predicted with previous frames, the universal prediction side estimator is 2-3dB lower than the conventional MCP-based motion side estimators in terms of coding efficiency. However, for other sequences, the coding efficiency is rather close and sometimes universal prediction side estimator even outperforms the MCP-based side estimator. We also showed that this new method can significantly reduce the coding complexity at the decoder.

- **Block Artifact Reduction Using A Transform Domain Markov Random Field Model**

  Lossy image and video compression is often accompanied with annoying artifacts. Post processing methods are used to reduce or eliminate the artifacts. Many post processing methods are done in the spatial domain. Since human visual system is not as sensitive to the artifacts in the high frequency bands as to those in the low frequency bands, spatial domain post processing methods may unnecessarily process those artifacts in the high frequency bands and hence lead to high computational complexity. We presented a transform domain Markov Random Field model to address the artifact reduction. We presented two methods, namely TD-MRF and TSD-MRF, based on this model. We showed by objective and subjective comparisons that transform domain post processing method can substantially relax the computational complexity constraint compared with spatial domain method and still achieve significant block artifact reduction. We note that TD-MRF generally cannot achieve as much block artifact reduction as SD-MRF. This is because in highly quantized images, almost all high frequency coefficients are truncated to zero. Hence, loss incurred to high frequency coefficients cannot be fully recaptured if subbands are processed separately as in TD-MRF. Meanwhile, the TSD-MRF framework that combines the TD-MRF and SD-MRF leads to more significant block artifact reduction without over-blurring the image.

5.2 Future Work

The future work can be explored from the following three perspectives:

- For rate distortion analysis of Wyner-Ziv video coding, we have developed a systemic method to analyze the video coding efficiency. In the future, this method can be extended to analyze the use of other motion search methods for the Wyner-Ziv side estimator, such as motion search with various block shapes, leaky motion prediction, motion vector prediction, and multiple description motion search.

- For universal prediction based side information estimation, we currently only use the co-located pixels in the previous frames as the search context. In the future, one could consider how to optimally select a context to improve video coding efficiency, in particular the size and shape of the context. Our results showed that universal prediction sometimes can better capture the characteristics of a video sequence, hence one could also study the use of universal prediction in conventional video coding.

- In this thesis we addressed the rate distortion tradeoffs in low complexity video compression. A broader problem to study in the future is to explore the theoretic aspects of rate-distortion-complexity tradeoffs and the applications in video compression. This is particularly important for applications that face constraints both in the data rate and computational resources.

LIST OF REFERENCES

LIST OF REFERENCES

[1] A. Murat Tekalp. *Digital Video Processing.* Prentice Hall, 1995. ISBN: 0131900757.

[2] Alan C. Bovik. *Handbook of Image and Video Processing.* Academic Press, 2000. ISBN: 0121197905.

[3] C. Fenimore. Assessment of resolution and dynamic range for digital cinema. In *Proceedings of the SPIE 15th Annual Symposium on Electronic Imaging Science and Technology*, volume 5022, Santa Clara, CA, January 20-24, 2003.

[4] Digital compression and coding of continuous-tone still images: Requirements and guidelines. ISO/IEC and ITU-T, September 1992.

[5] G. J. Sullivan and T. Wiegand. Rate-distortion optimization for video compression. *IEEE Signal processing Magazine*, pages 74–90, November 1998.

[6] Coding of audiovisual objects - part 2: Visual. ISO/IEC 14496-2 (MPEG-4), 1999.

[7] Codec for videoconferencing using primary digital group transmission. ITU-T Recommendation H.120 Version 2, 1988.

[8] Video codec for audiovisual services at $p \times 64$ kbits/s. ITU-T Recommendation H.261 Version 2, March 1993.

[9] Coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbits/s - part 2: Video. ISO/IEC 11172-2 (MPEG-1), March 1993.

[10] Generic coding of moving pictures and associated audio information - part 2: Video. ITU-T Recommendation H.262 and ISO/IEC 13818-2 (MPEG-2), November 1994.

[11] Video coding for low bit rate communications. ITU-T Recommendation H.263 Version 2, January 1998.

[12] W. Li. Overview of fine granularity scalability in MPEG-4 video standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3):301–317, March 2001.

[13] Coding of audiovisual objects - part 2: Visual, amendment 4: Streaming video profile. ISO/IEC 14496-2, July, 2000.

[14] D. Marpe, H. Schwarz, and T. Wiegand. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):620–636, July 2003.

[15] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, July and October 1948.

[16] F. Moschetti. *A Statistical Approach to Motion Estimation*. PhD thesis, Swiss Federal Institute of Technology, Lausanne Switzerland, 2001.

[17] R. Mathew and J. Arnold. Layered coding using bitstream decomposition with drift correction. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(6):882–891, November 1997.

[18] P. Yin, A. Vetro, B. Liu, and H. Sun. Drift compensation for reduced spatial resolution ttranscoding. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(11):1009–1020, November 2002.

[19] T. Wedi and H. G. Musmann. Motion- and aliasing-compensated prediction for hybrid video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):577–586, July 2003.

[20] M. Flierl and B. Girod. Generalized B pictures and the draft H.264/AVC video-compression standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):587–597, July 2003.

[21] Y. Liu. *Layered Scalable and Low Complexity Video Encoding: New Approaches and Theoretic Analysis*. PhD thesis, Purdue University, West Lafayette, IN, 2004.

[22] A. Smolic, T. Sikora, and J. Ohm. Long-term global motion estimation and its applications for sprite coding, content description, and segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(8):1227–1242, December 1999.

[23] F. Dufaux and J. Konrad. Efficient, robust, and fast global motion estimation for video coding. *IEEE Transactions on Imaging Processing*, 9(3):497–501, March 2000.

[24] G. Rath and A. Makur. Iterative least squares and compression based estimations for a four-parameter linear global motion model and global motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(7):1075–1099, October 1999.

[25] G. Giunta and U. Mascia. Estimation of global motion parameters by complex linear regression. *IEEE Transactions on Imaging Processing*, 8(11):1652–1657, November 1999.

[26] D. Wang and L. Wang. Global motion parameters estimation using a fast and robust algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(5):823–826, October 1997.

[27] K. Shen and E. Delp. Wavelet based rate scalable video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(1):109–122, February 1999.

[28] F. Wu, S. Li, R. Yan, X. Sun, and Y. Zhang. Efficient and universal scalable video coding. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 37–40, Rochester, NY, September 22-25, 2002.

[29] F. Wu, S. Li, and Y. Zhang. A framework for efficient progressive fine granularity scalable video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3):332–344, March 2001.

[30] S. Han and B. Girod. Robust and efficient scalable video coding with leaky prediction. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 41–44, Rochester, NY, September 22-25, 2002.

[31] H. Huang, C. Wang, and T. Chiang. A robust fine granularity scalability using trellis-based predictive leak. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(6):372–385, June 2002.

[32] K. Chang and R. Donaldson. Analysis, optimization, and sensitivity study of differential PCM systems operating on noisy communication channels. *IEEE Transactions on Communications*, COM-20(3):338–350, June 1972.

[33] N. Jayant and P. Noll. *Digital Coding of Waveforms*. Prentice Hall, Englewood Cliffs, NJ, 1984. ISBN: 0132119137.

[34] Y. Wang and S. Lin. Error-resilient video coding using multiple description motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(6):438–452, June 2002.

[35] A. Reibman, H. Jafarkhani, Y. Wang, M. Orchard, and R. Puri. Multiple-description video coding using motion-compensated temporal prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(3):193–204, March 2002.

[36] B. Girod. Why B-pictures work: a theory of multi-hypothesis motion-compensated prediction. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 41–44, Chicago, IL, October 4-7, 1998.

[37] M. Flierl, T. Wiegand, and B. Girod. Rate-constrained multihypothesis prediction for motion-compensated video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(11):957–969, November 2002.

[38] T. Wiegand, X. Zhang, and B. Girod. Long-term memory motion-compensated prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(1):70–84, February 1999.

[39] B. Girod. Efficiency analysis of multihypothesis motion-compensated prediction for video coding. *IEEE Transactions on Image Processing*, 9(2):173–183, February 2000.

[40] K. Rose and S. Regunathan. Toward optimality in scalable predictive coding. *IEEE Transactions on Image Processing*, 10(7):965–976, July 2001.

[41] B. Girod. The efficiency of motion-compensating prediction for hybrid coding of video sequences. *IEEE Journal on Selected Areas in Communications*, 5(7):1140–1154, August 1987.

[42] B. Girod. Motion-compensating prediction with fractional-pel accuracy. *IEEE Transactions on Communications*, 41(4):604–612, April 1993.

[43] T. Wiegand, X. Zhang, and B. Girod. Long-term memory motion-compensated prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(1):70–84, February 1999.

[44] B. Girod. Efficiency analysis of multihypothesis motion-compensated prediction for video coding. *IEEE Transactions on Image Processing*, 9(2):173–183, February 2000.

[45] M. Flierl, T. Wiegand, and B. Girod. Rate-constrained multihypothesis prediction for motion-compensated video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(11):957–969, November 2002.

[46] G. Cook. *A Study of Scalability in Video Compression: Rate-Distortion Analysis and Parallel Implementation.* PhD thesis, Purdue University, West Lafayette, IN, 2002.

[47] G. Cook, J. Prades-Nebot, Y. Liu, and E. Delp. Rate-distortion analysis of motion compensated rate scalable video. *submitted to IEEE Transactions on Image Processing*, 2004.

[48] J. Prades-Nebot, G. Cook, and E. Delp. Analysis of the efficiency of snr-scalable strategies for motion compensated video coders. In *Proceedings of the IEEE International Conference on Image Processing*, volume 5, pages 3109–3112, Singapore, October 24-27 2004.

[49] G. Cook, J. Prades-Nebot, and E. Delp. Rate-distortion bounds for motion compensated rate scalable video coders. In *Proceedings of the IEEE International Conference on Image Processing*, volume 5, pages 3121–3124, Singapore, October 24-27 2004.

[50] Y. Liu, P. Salama, Z. Li, and E. Delp. An enhancement of leaky prediction layered video coding. *IEEE Transactions on Circuits and Systems for Video Technology, accepted*, 2004.

[51] Y. Liu, J. Prades-Nebot, P. Salama, and E. Delp. Rate distortion analysis of leaky prediction layered video coding using quantization noise modeling. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 801–804, Singapore, October 24-27 2004.

[52] Z. Li and E. Delp. Channel-aware rate-distortion optimized leaky motion prediction. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 2079–2082, Singapore, October 24-27 2004.

[53] Coding of audio-visual objects - part 10: Advanced video coding. ITU-T Recommendation H.264—ISO/IEC 14496-10 AVC(MPEG-4 Part 10), 2002.

[54] H.26L test model long-term number 8 (TML-8). ITU-T/SG16/VCEG(Q.6), September 24-27, 2001.

[55] A. Zakhor. Iterative procedures for reductions of blocking effects in transform image coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 2(1):91–95, March 1992.

[56] H. Paek, R. Kim, and S. Lee. On the POCS-based postprocessing technique to reduce the blocking artifacts in transform coded images. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(3):358–367, June 1998.

[57] Z. Xiong, M. Orchard, and Y. Zhang. A deblocking algorithm for JPEG compressed images using over-complete wavelet representations. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(5):692–695, August 1999.

[58] R. Stevenson and E. Delp. Fitting curves with discontinuites. In *Proceedings of the First International Workshop on Robust Computer Vision*, pages 127–136, Seattle, WA, October 1-3, 1990.

[59] T. O'Rourke and R. Stevenson. Improved image decomposition for reduced transform coding artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(6):490–499, December 1995.

[60] M. Robertson and R. Stevenson. Reduced-complexity iterative post-filtering of video. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(10):1121–1127, October 2001.

[61] S. Minami and A. Zakhor. An optimization approach for removing blocking effects in transform coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(2):74–82, April 1995.

[62] G. Lakhani and N. Zhong. Derivation of prediction equations for blocking effect reduction. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(3):415–418, April 1999.

[63] G. Triantafyllidis, D. Tzovaras, and M. Strintzis. Blocking artifact detection and reduction in compressed data. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(10):877–890, October 2002.

[64] Z. Li and E. Delp. Block artifact reduction using a transform domain markov random field model. *IEEE Transactions on Circuits and Systems for Video Technology, accepted*, 2005.

[65] Z. Li, L. Liu, and E. Delp. Wyner-ziv video coding: How far away is coding reality? *in submission to IEEE Transactions on Circuits and Systems for Video Technology*, 2005.

[66] Z. Li, L. Liu, and E. Delp. Wyner-ziv video coding with universal prediction. *in submission to IEEE Transactions on Circuits and Systems for Video Technology*, 2005.

[67] Z. Li, L. Liu, and E. Delp. Wyner-ziv video coding: A motion estimation perspective. In *submission to the SPIE International Conference on Video Communications and Image Processing*, San Jose, January 15-19 2006.

[68] Z. Li, L. Liu, and E. Delp. Wyner-ziv video coding with universal prediction. In *submission to the SPIE International Conference on Video Communications and Image Processing*, San Jose, January 15-19 2006.

[69] Z. Li and E. Delp. Wyner-ziv video side estimator: Conventional motion search methods revisited. In *Proceedings of the IEEE International Conference on Image Processing, accepted*, Genova, Italy, September 11-15 2005.

[70] Z. Li and E. Delp. Statistical motion prediction with drift. In *Proceedings of the SPIE International Conference on Video Communications and Image Processing*, volume 5308, pages 416–427, San Jose, CA, January 18-22 2004.

[71] Z. Li and E. Delp. Universal motion prediction. In *Proceedings of the SPIE International Conference on Video Communications and Image Processing*, volume 5308, pages 1295–1304, San Jose, CA, January 18-22 2004.

[72] Z. Li, G. Shen, S. Li, and E. Delp. L-tfrc: An end-to-end congestion control mechanism for video streaming over the internet. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 2, pages 309–312, Baltimore, MD, July 6-9 2003.

[73] Y. Liu, Z. Li, P. Salama, and E. Delp. A discussion of leaky prediction based scalable coding. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 2, pages 565–568, Baltimore, MD, July 6-9 2003.

[74] Z. Li, F. Wu, S. Li, and E. Delp. Performance optimization for motion compensated 2d wavelet video compression techniques. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 2, pages 616–619, Bangkok, Thailand, May 25-28 2003.

[75] Z. Li, F. Wu, S. Li, and E. Delp. Wavelet video coding via spatially adaptive lifting structure. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 93–96, Hong Kong, China, April 6-10 2003.

[76] Z. Li and E. Delp. MAP-based post processing post processing of video sequences using 3-D Huber Markov random field model. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 153–156, Lausanne, Switzerland, August 26-29 2002.

[77] Z. Li and E. Delp. A new progressive block artifact reduction algorithm using a transform domain-based markov random field model. In *Proceedings of the SPIE International Conference on Image and Video Communications and Processing*, volume 5022, pages 1001–1012, Santa Clara, CA, anuary 20-24 2003.

[78] D. Slepian and J. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, IT-19(4):471–480, July 1973.

[79] A. Wyner. Recent results in the shannon theory. *IEEE Transactions on Information Theory*, IT-20(1):2–9, January 1974.

[80] A. Wyner and J. Ziv. The rate-distortion function for source with side information at the decoder. *IEEE Transactions on Information Theory*, IT-22(1):1–10, January 1976.

[81] S. Verdu. Fifty years of shannon theory. *IEEE Transactions on Information Theory*, IT-44(6):2057–2078, October 1998.

[82] Robert G. Gallager. *Low-Density Parity-Check Codes.* The MIT Press, Cambridge, MA, 1963. ISBN: 0262571773.

[83] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero. Distributed video coding. *Proceedings of the IEEE*, 93(1):71–83, January 2005.

[84] S. Pradhan and K. Ramchandran. Distributed source coding using syndromes(DISCUS): Design and construction. *IEEE Transactions on Information Theory*, IT-49(3):626–643, March 2003.

[85] Z. Xiong, A. Liveris, and S. Cheng. Distributed source coding for sensor networks. *IEEE Signal Processing Magazine*, 21(5):80–94, September 2004.

[86] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding: Turbo-codes. In *Proceedings of the IEEE International Conference on Communications*, volume 2, pages 1064–1070, Geneva, Switzerland, May 11-15 1993.

[87] C. Berrou and A. Glavieux. Near optimum error correcting coding and decoding: Turbo-codes. *IEEE Transactions on Communications*, 44(10):1261–1271, October 1996.

[88] R. Zamir and S. Shamai. Nested linear/lattice codes for Wyner-Ziv encoding. In *Proceedings of the IEEE Information Theory Workshop*, pages 92–93, Killarney, Ireland, June 22-26 1998.

[89] P. Ishwar, V. Prabhakaran, and K. Ramchandran. Towards a theory for video coding using distributed compression principles. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 687–690, Barcelona, Spain, September 14-17 2003.

[90] R. Puri and K. Ramchandran. Prism: An uplink-friendly multimedia coding paradigm. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 856–859, Hongkong, China, April 6-10 2003.

[91] C Lan, A. Liveris, K. Narayanan, Z. Xiong, and C. Georghiades. Slepian-Wolf coding of multiple m-ary sources using LDPC codes. In *Proceedings of the Data Compression Conference*, pages 549–549, Snowbird, UT, March 23-25 2004.

[92] Z. Xiong, A. Liveris, S. Cheng, and Z. Liu. Nested quantization and Slepian-Wolf coding: A Wyner-Ziv coding paradigm for i.i.d. sources. In *Proceedings of the IEEE Workshop on Statistical Signal Processing*, pages 399–402, St. Louis, MO, September 28 - October 1 2003.

[93] A. Liveris, Z. Xiong, and C. Georghiades. Compression of binary sources with side information at the decoder using LDPC codes. *IEEE Communications Letters*, 6(10):440–442, October 2002.

[94] A. Liveris, Z. Xiong, and C. Georghiades. A distributed source coding technique for correlated images using turbo-codes. *IEEE Communications Letters*, 6(9):379–381, September 2002.

[95] A. Aaron, S. Rane, and B. Girod. Wyner-Ziv video coding with hash-based motion compensation at the receiver. In *Proceedings of the IEEE International Conference on Image Processing*, Singapore, October 24-27 2004.

[96] A. Aaron, E. Setton, and B. Girod. Towards practical Wyner-Ziv coding of video. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 869–872, Barcelona, Spain, September 14-17 2003.

[97] A. Aaron, S. Rane, R. Zhang, and B. Girod. Transform-domain Wyner-Ziv codec for video. In *Proceedings of the SPIE Visual Communications and Image Processing*, pages 520–528, Santa Clara, CA, January 18-22 2004.

[98] J. Kusuma, L. Doherty, and K. Ramchandran. Distributed compression for sensor networks. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, pages 82–85, Thessaloniki, Greek, October 7-10 2001.

[99] A. Aaron, S. Rane, R. Zhang, and B. Girod. Wyner-Ziv coding of video: Applications to compression and error resilience. In *Proceedings of the Data Compression Conference*, pages 93–102, Snowbird, UT, March 25-27 2003.

[100] A. Sehgal, A. Jagmohan, and N. Ahuja. Wyner-Ziv coding of video: An error-resilient compression framework. *IEEE Transactions on Multimedia*, 6(2):249–258, April 2004.

[101] R. Zamir. The rate loss in the Wyner-Ziv problem. *IEEE Transactions on Information Theory*, IT-42(6):2073–2084, November 1996.

[102] Toby Berger. *Rate distortion theory: A mathematical basis for data compression.* Prentice-Hall, 1971. ISBN: 0137531036.

[103] T. Cover and J. Thomas. *Elements of Information Theory.* Wiley-Interscience, 1991. ISBN: 0471062596.

[104] R. Buschmann. Efficiency of displacement estimation techniques. *Signal Processing: Image Communications*, 10:43–61, 1997.

[105] R. Puri and K. Ramchandran. Prism: An uplink-friendly multimedia coding paradigm. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, pages 617–620, Barcelona, Spain, September 14-17 2003.

[106] N Merhav and M. Feder. Universal prediction. *IEEE Transactions on Information Theory*, 44(6):2124–2147, October 1998.

[107] J. Ziv and A. Lemple. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 23(3):337–343, May, 1977.

[108] J. Ziv and A. Lemple. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5):530–536, September, 1978.

[109] Jr J. L. Kelly. A new interpretation of information rate. *Bell System Technical Journal*, 35:917–926, 1956.

[110] N Merhav, M. Feder, and M. Gutman. Universal prediction of individual sequences. *IEEE Transactions on Information Theory*, 38(4):1258–1270, July 1992.

[111] M. J. Weinberger, J. J. Rissanen, and M. Feder. A universal finite-memory source. *IEEE Transactions on Information Theory*, 41(3):643–652, May 1995.

[112] T. Weissman, N. Merhav, and A. Baruch. Twofold universal prediction schemes for achieving the finite-state predictability of a noisy individual binary sequence. *IEEE Transactions on Information Theory*, 47(5):1849–1866, July 2001.

[113] T. Weissman and N. Merhav. Universal prediction of individual binary sequence in the presence of noise. *IEEE Transactions on Information Theory*, 47(6):2151–1273, September 2001.

[114] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M. J. Weinberger. Universal discrete noising. *IEEE Transactions on Information Theory*, 51(1):5–28, January 2005.

[115] E. Ordentlich, G. Seroussi, S. Verdu, K. Viswanathan, M. J. Weinberger, and T. Weissman. Channel decoding of systematically encoded unknown redundant sources. In *Proceedings of the IEEE International Symposium on Information Theory*, page 165, Chicago, IL, June-July, 2004.

[116] E. Ordentlich, T. Weissman, M. J. Weinberger, A. Somekh-Baruch, and N. Merhav. Discrete universal filtering through incremental parsing. In *Proceedings of the 2004 Data Compression Conference*, pages 352–361, Snowbird, UT, March 2004.

[117] G. Gemelos, S. Sigurjonsson, and T. Weissman. Universal minimax binary imaging denoising under channel uncertainty. In *Proceedings of the IEEE International Conference on Image Processing*, pages 997–1000, Singapore, October 24-27 2004.

[118] R. Zhang and T. Weissman. On discrete denoising for the burst noise channel. In *Proceedings of the 42nd Annual Allerton COnference on Communication, Control, and Computing*, Monticello, IL, September 2004.

[119] Y. Yang, N. Galatsanos, and A. Katsaggelos. Regularized reconstruction to reduce blocking artifact of block discrete cosine transform compressed images. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(6):421–432, December 1993.

[120] C. Weerasinghe, W. Liew, and H. Yan. Artifact reduction in compressed images based on region homogeneity constraints using the projection onto convex sets algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(10):891–897, October 2002.

[121] T. Hsung, D. Lun, and W. Siu. A deblocking technique for block-transform compressed image using wavelet transform modulus maxima. *IEEE Transactions on Image Processing*, 7(10):1488–1496, October 1998.

[122] N. Kim, I. Jang, D. Kim, and W. Hong. Reduction of blocking artifact in block-coded images using wavelet transform. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(3):253–257, June 1998.

[123] A. Liew and H. Yan. Blocking artifacts suppression in block-coded images using overcomplete wavelet representation. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(4):450–461, April 2004.

[124] T. Chen, H. Wu, and B. Qiu. Adaptive postfiltering of transform coefficients for the reduction of blocking artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(5):594–602, May 2001.

[125] B. Jeon and J. Jeong. Blocking artifacts reduction in image compression with block boundary discontinuity criterion. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(3):345–357, June 1998.

[126] B. Zeng. Reduction of blocking effect in dct-coded images using zero-masking techniques. *Signal Processing*, 79(2):205–211, December 1999.

[127] S. Liu and A. Bovik. Efficient dct-domain blind measurement and reduction of blocking artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(12):1139–1159, December 2002.

[128] S. Karunasekera and N. Kingsbury. A distortion measure for blocking artifacts in images based on human visual sensitivity. *IEEE Transactions on Image Processing*, 4(6):713–724, June 1995.

[129] C. Bouman. Markov random fields and stochastic image models. In *Proceedings of the IEEE International Conference on Image Processing,Tutorial Notes*, Washington DC, October 23-26, 1995.

[130] H. Derin, H. Elliot, R. Cristi, and D. Geman. Bayes smoothing algorithms for segmentation of binary images modeled by markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:707–720, November 1984.

[131] J. Zhang, D. Wang, and P. Fieguth. *Random Field Models*. Academic Press, 2001.

[132] C. Bouman and K. Sauer. A generalized Gaussian image model for edge-preserving MAP estimation. *IEEE Transactions on Image Processing*, 2(3):296–310, July 1993.

[133] C. Bouman and K. Sauer. Maximum likelihood scale estimation for a class of Markov random fields. In *Proceedings of the IEEE International Conference on Accoustics, Speech, and Signal Processing*, volume 5, pages 537–540, Adelaide, Australia, April 19-22, 1994.

[134] E. Lam and J. Goodman. A mathematical analysis of the DCT coefficients distributions for images. *IEEE Transactions on Image Processing*, 9(10):1661–1666, October 2000.

[135] R. Reininger and J. Gibson. Distributions of the two-dimensional dct coefficients for images. *IEEE Transactions on Communications*, 31(6):835–839, June 1983.

[136] F. Muller. Distribution shape of two-dimensional dct coefficients of natural images. *Electronics Letters*, 29(22):1935–1936, October 28 1993.

[137] C. Nikias and M. Shao. *Signal Processing with Alpha-Stable Distributions and Applications*. John Wiley and Sons, New York, NY, 1995.

[138] M. Robertson and R. Stevenson. DCT quantization noise in compressed images. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, pages 185–188, Thessaloniki, Greece, October 7-10, 2001.

[139] E. Chong and S. Zak. *An Introduction to Optimization*. Wiley-Interscience, 2001. ISBN: 0471391263.

[140] *The USC-SIPI Image Database*. Available [Online]: http://sipi.usc.edu/services/database/.

[141] *VcDemo: Image and Video Compression Learning Tool*. Available [Online]: http://www-it.et.tudelft.nl/ inald/vcdemo/.

VITA

VITA

Zhen Li was born in Fujian, P. R. China, in 1978. He earned his B.E. degree in electrical engineering in 2000 from Tsinghua University, Beijing, China. He was supervised by Prof. Kun Tang and Prof. Huijuan Cui for his study at Tsinghua on H.263+ video coding with forward error correction for wireless channel.

Since Fall 2000, he has been pursuing his Ph.D. degree in the School of Electrical and Computer Engineering at Purdue University, West Lafayette, Indiana. Since 2001 he has been a research assistant in the Video and Image Processing Laboratory (VIPER) under the supervision of Professor Edward J. Delp. His research work has been funded by a grant from the Indiana Twenty-First Century Research and Technology Fund. His current research interests include image and video compression (JPEG/JPEG2000/MPEG-2/H.263/MPEG-4/H.264/AVC), image and video post processing, multimedia communications and wireless networking, .

During the summers of 2001 and 2002, he worked at Microsoft Reesarch Asia as a summer intern. He worked with Drs. Shipeng Li, Feng Wu and Guobin Shen on rate control for MPEG-4 video streaming, advanced wavelet video coding with adaptive lifting structure, and rate-distortion optimizations for H.26L/AVC. In the summer of 2004, he worked at the Multimedia Communications Research Lab, Motorola Labs, Motorola Inc., Schaumberg, IL, as a summer intern. In Motorola Labs he worked with Dr. Faisal Ishtiaq on JVT/H.264/AVC rate control with Hypothetical Reference Decoder (HRD) conformance.

Zhen Li is a student member of the IEEE professional society.

Zhen Li's publications for his research work at Purdue include:

Journal papers:

1. Zhen Li, Limin Liu, and Edward J. Delp, "Wyner-Ziv Video Coding: How Far Away is Coding Reality?" in submission to the *IEEE Transactions on Circuits and Systems for Video Technology.*

2. Zhen Li, Limin Liu, and Edward J. Delp, "Wyner-Ziv Video Coding with Universal Prediction," in submission to the *IEEE Transactions on Circuits and Systems for Video Technology.*

3. Zhen Li and Edward Delp, "Block Artifact Reduction Using A Transform Domain Markov Random Field Model," accepted by the *IEEE Transactions on Circuits and Systems for Video Technology.*

4. Yuxin Liu, Paul Salama, Zhen Li, and Edward J. Delp, "An Enhancement of Leaky Prediction Layered Video Coding," accepted by the *IEEE Transactions on Circuits and Systems for Video Technology.*

Conference papers:

1. Zhen Li, Limin Liu, and Edward J. Delp, "Wyner-Ziv Video Coding with Universal Prediction," submitted to the *SPIE International Conference on Video Communications and Image Processing (VCIP)*, January 15-19, 2006, San Jose, CA.

2. Zhen Li, Limin Liu, and Edward J. Delp, "Wyner-Ziv Video Coding: A Motion Estimation Perspective," submitted to the *SPIE International Conference on Video Communications and Image Processing (VCIP)*, January 15-19, 2006, San Jose, CA.

3. Zhen Li and Edward J. Delp, "Wyner-Ziv Video Side Estimator: Conventional Motion Search Methods Revisited," accepted by the *IEEE International Conference on Image Processing (ICIP)*, September 11-15, 2005, Genova, Italy.

4. Zhen Li and Edward J. Delp, "Channel-Aware Rate-Distortion Optimized Leaky Motion Prediction," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, October 24-27, 2004, Singapore, Vol. 3, pp. 2079-2082.

5. Zhen Li and Edward J. Delp, "Statistical Motion Prediction with Drift," *Proceedings of the SPIE International Conference on Video Communications and Image Processing (VCIP)*, January 18-22, 2004, San Jose, CA, Vol. 5308, pp. 416-427.

6. Zhen Li and Edward J. Delp, "Universal Motion Prediction," *Proceedings of the SPIE International Conference on Video Communications and Image Processing (VCIP)*, January 18-22, 2004, San Jose, CA, Vol. 5308, pp. 1295-1304.

7. Zhen Li, Guobin Shen, Shipeng Li, and Edward J. Delp, "L-TFRC: An End-to-end Congestion Control Mechanism for Video Streaming Over the Internet," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, July 6-9, 2003, Baltimore, MD, Vol. 2, pp. 309-312.

8. Yuxin Liu, Zhen Li, Paul Salama, and Edward J. Delp, "A Discussion of Leaky Prediction Based Scalable Coding," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, July 6-9, 2003, Baltimore, MD, Vol. 2, pp. 565-568.

9. Zhen Li, Feng Wu, Shipeng Li, and Edward J. Delp, "Performance Optimization for Motion Compensated 2D Wavelet Video Compression Techniques," *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, May 25-28, 2003, Bangkok, Thailand, Vol. 2, pp. 616-619.

10. Zhen Li, Feng Wu, Shipeng Li, and Edward J. Delp, "Wavelet Video Coding Via Spatially Adaptive Lifting Structure," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 6-10, 2003, Hong Kong, Vol. 3, pp. 93-96.

11. Zhen Li and Edward J. Delp, "A New Progressive Block Artifact Reduction Algorithm Using a Transform Domain-Based Markov Random Field Model," *Proceedings of the SPIE International Conference on Image and Video Communications and Processing (IVCP)*, January 20-24, 2003, Santa Clara, CA, Vol. 5022, pp. 1001-1012.

12. Zhen Li and Edward J. Delp, “MAP-based Post Processing Post Processing of Video Sequences Using 3-D Huber Markov Random Field Model,” *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, August 26-29, 2002, Lausanne, Switzerland, Vol. 1, pp. 153-156.

Patent:

1. “L-TFRC: An End-to-end Congestion Control Mechanism for Video Streaming Over the Internet,” European Patent Pending, Assignee: Microsoft Corporation.