กรอบการเข้ารหัสวีดิทัศน์ที่ทนทานต่อความผิดพลาดโดยใช้การจัดเรียงแมโครบล็อกแบบยืดหยุ่นได้และการปกปิดความผิดพลาดสำหรับการส่งวีดิทัศน์ไร้สาย

นางสาว จันทนา ปัญญาวราภรณ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2552

FRAMEWORK OF ERROR-RESILIENT VIDEO CODING USING FLEXIBLE MACROBLOCK ORDERING AND ERROR CONCEALMENT FOR WIRELESS VIDEO TRANSMISSION

Miss Jantana Panyavaraporn

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Doctor of Philosophy Program in Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2009

Thesis Title	FRAMEWORK OF ERROR-RESILIENT VIDEO CODING USING FLEXIBLE MACROBLOCK ORDERING AND ERROR CONCEALMENT FOR WIRELESS VIDEO TRANSMISSION
By	Miss Jantana Panyavaraporn
Field of Study	Electrical Engineering
Thesis Advisor	Assistant Professor Supavadee Aramvith, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Doctoral Degree

..... Dean of the Faculty of Engineering
(Associate Professor Boonsom Lerdhirunwong, Dr.Ing)

THESIS COMMITTEE

.....Chairman
(Professor Prasit Prapingmongkolkarn, Ph.D.)

.....Thesis Advisor
(Assistant Professor Supavadee Aramvith, Ph.D.)

.....Examiner
(Assistant Professor Chaiyachet Saivichit, Ph.D.)

..... Examiner
(Assistant Professor Suree Pumrin, Ph.D.)

.....External Examiner
(Associate Professor Thumrongrat Amornraksa, Ph.D.)

จันทนา ปัญญาวราภรณ์ : กรอบการเข้ารหัสวีดิทัศน์ที่ทนทานต่อความผิดพลาดโดยใช้การจัดเรียงแมโครบล็อกแบบยืดหยุ่นได้และการปกปิดความผิดพลาดสำหรับการส่งวีดิทัศน์ไร้สาย. (FRAMEWORK OF ERROR-RESILIENT VIDEO CODING USING FLEXIBLE MACROBLOCK ORDERING AND ERROR CONCEALMENT FOR WIRELESS VIDEO TRANSMISSION) อ. ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ.ดร. สุภาวดี อร่ามวิทย์, 90 หน้า.

H.264/AVC เป็นมาตรฐานที่เลือกใช้เครื่องมือด้านทานความผิดพลาดที่ตัวเข้ารหัส ที่เรียกว่าการจัดเรียงแมโครบล็อกแบบยืดหยุ่นได้ จุดประสงค์หลักของการจัดเรียงแมโครบล็อกแบบยืดหยุ่นได้ คือใช้ลดผลของความผิดพลาดแบบเบิรสต์โดยใช้การจัดแผนที่กลุ่มสไลซ์สำหรับแต่ละภาพ ในงานวิจัยก่อนหน้าได้ใช้การนับจำนวนบิตของแมโครบล็อกซึ่งเรียกได้ว่าเป็นข้อมูลเชิงพื้นที่และใช้การเข้ารหัสสองรอบในการสร้างแผนที่ตำแหน่งแมโครบล็อก ดังนั้นจึงส่งผลให้เกิดงานวิจัยนี้ขึ้นได้ประยุกต์การใช้ค่าความเพี้ยนของแมโครบล็อกที่อยู่บนพื้นฐานของการปกปิดความผิดพลาดซึ่งเป็นข้อมูลในเชิงเวลาเป็นตัวสร้างรูปแบบแผนที่ตำแหน่งแมโครบล็อก และเนื่องจากงานวิจัยก่อนหน้าทั้งหมดใช้การเข้ารหัสวิดิทัศน์สองรอบ เพื่อหลีกเลี่ยงในเรื่องการหน่วงเวลาและการทำงานที่ซ้ำซ้อนจึงนำมาซึ่งงานวิจัยถัดมา คือการสร้างแผนที่กลุ่มตำแหน่งแมโครบล็อกแบบเข้ารหัสเพียงรอบเดียวในการสร้างแผนที่กลุ่มสไลซ์

นอกจากนี้แล้ว ได้มีการนำเสนอกรอบงานที่รวมการสร้างแผนที่กลุ่มสไลซ์แบบเข้ารหัสรอบเดียวและเทคนิคการปกปิดความผิดพลาด ซึ่งวิธีดังกล่าวนี้สามารถปรับปรุงคุณภาพของวีดิทัศน์เนื่องจากความผิดพลาดในการส่ง การสร้างแผนที่กลุ่มสไลซ์แบบเข้ารหัสรอบเดียวใช้ค่าข้อมูลป้อนกลับที่ได้จากข้อมูลในเชิงพื้นที่และเชิงเวลาในส่วนเข้ารหัส และนำเสนอวิธีการปกปิดความผิดพลาดในส่วนถอดรหัส วิธีการปกปิดความผิดพลาดขึ้นอยู่กับค่าพารามิเตอร์ที่ได้จากข้อมูลจากส่วนเข้ารหัสซึ่งค่าที่ได้มาจากการสร้างแผนที่ตำแหน่งแมโครบล็อกแบบเข้ารหัสครั้งเดียว ผลการจำลองการส่งวีดิทัศน์ผ่านช่องสัญญาณไร้สายแบบเฟดดิงช้าและเฟดดิงเร็วพบว่า วิธีที่นำเสนอสามารถลดจำนวนแมโครบล็อกที่ไม่สามารถถอดรหัสมากถึง 80.54% และปรับปรุงค่า PSNR มากถึง 6.09 dB เมื่อเปรียบเทียบกับวิธีที่ไม่ใช้การจัดเรียงแมโครบล็อกแบบยืดหยุ่นได้และใช้การปกปิดความผิดพลาดแบบไม่ชดเชยการเคลื่อนที่

ภาควิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่อนิสิต.....จันทนา ปัญญาวราภรณ์.....

สาขาวิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก.....

ปีการศึกษา.....2552.....

4971845621 : MAJOR ELECTRICAL ENGINEERING DEPARTMENT
KEY WORD: FMO / ERROR CONCEALMENT / H.264 / FEC / INTERLEAVING

JANTANA PANYAVARAPORN: FRAMEWORK OF ERROR-RESILIENT VIDEO CODING USING FLEXIBLE MACROBLOCK ORDERING AND ERROR CONCEALMENT FOR WIRELESS VIDEO TRANSMISSION. THESIS ADVISOR: ASST. PROF. SUPAVADEE ARAMVITH, Ph.D.,90 pp.

The H.264/AVC standard adopts a robust error resilience tool at the encoder known as Flexible Macroblock Ordering (FMO). The main goal of this tool is to provide a macroblock-level interleaving tool to spread out consecutive burst errors in a frame. Past research proposed the use of macroblock-coded bit-count which acts as spatial information as indicator of macroblock importance, and uses a two-pass encoding process to generate the macroblock-address map. In this dissertation, we propose to use a distortion measure based on concealment error which acts as temporal information as an indicator for a choice of macroblock-address-map of each picture. To avoid the incurred delay and complexity computing of two-pass encoding, we also propose a one-pass encoding scheme to generate the macroblock-address maps.

Furthermore, we present a framework that combines one-pass FMO map generation and error concealment algorithms to improve the video quality due to transmission errors. The one-pass FMO map generation is accomplished by using feedback in terms of spatial and temporal information to simulate spatial and temporal error concealment at the encoder. The choice of error concealment method at decoder is applied according to parameter derived from residual information obtained at the encoder during the map generation process. Our simulation results performed under slow and fast fading channel confirm that the proposed technique can reduce the number of undecodable macroblock up to 80.54% and PSNR improvement are up to 6.09 dB when compared with methods that don't use FMO and uses a simple non-motion compensated error concealment.

Department.....Electrical Engineering.....Student's signature.....Jantana P.....
Field of study... Electrical Engineering.....Advisor's signature.....
Academic year.....2009.....

ACKNOWLEDGMENTS

First, I would like to thank my advisor Asst. Prof. Dr. Supavadee Aramvith for her encouragement, support, and invaluable guidance throughout the entire course of this study. By being up to date with the new trends in video processing and other research activities in the lab gives me motivation and inspiration that I need with out which this dissertation would not have been possible.

I also would like to thank the other members of my thesis committee, Assoc. Prof. Dr. Thumrongrat Amornraksa, Asst. Prof. Dr. Chaiyachet Saivichit, Asst. Prof. Dr. Suree Pumrin and the chairperson, Prof. Dr. Prasit Prapingmongkolkarn, for giving critical reviews of this work and for their advice on my dissertation.

I would like to thank CHE-AUN/Seed-Net Scholarship for which this dissertation has been supported. Also, I appreciate all members of the Video Research Group for their support during my research work and any assistance and warm collaborations.

Finally, I would like to express the highest gratitude to my parents, everyone in my family, and all of my friends for their help, their unfailing understanding and affectionate encouragements my graduate journey.

CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER I

INTRODUCTION

1.1 Motivation and Significance of the Research Problem

Wireless transmission has been growing tremendously in the last decades and has been used to transmit/receive multimedia such as images, sound, digital television broadcasts, video stream and video conferences which can provide an acceptable quality of service to the end user. However, the bandwidth of the channel is limited. Thus, the use of efficient coding schemes to compress the video signal, such as H.263, MPEG-2, MPEG-4 and H.264/AVC (ITU-T Recommendation H.264, 2005) is needed. The H.264/AVC is the best candidate codec for wireless multimedia applications. But if some packets are lost or some of the coded bits are corrupted, channel errors during transmission, an appropriate data recovering process is required to obtain acceptable visual quality. Several methods for error recovery techniques, including Automatic Retransmission reQuest (ARQ) (Lin, 1984), Forward Error Correction (FEC) (Puri, 2001) and interleaving technique which have been proposed to solve the above mentioned problem. ARQ may aggravate the channel congestion and can cause a system to drop more data. FEC usually requires redundant parity bits for error detection and correction. The interleaving technique may require considerable time delay and modification of video coding.

The H.264/AVC aims for high compression capability and support enhanced error-resilient features. Error resilient tools available in H.264/AVC include parameter sets, redundant slice, data partition and flexible macroblock ordering (FMO). Parameter set provides for robust and efficient conveyance of header information. Redundant slice is based on sending the same slice predicted from different frame. Data partitioning creates more bit string per slice and allocates all symbols of a slice into an individual partition that have a close semantic relationship with each other. Flexible macroblock ordering (FMO) (Lamber, 2006) allows assigning Macroblocks (MB) into slices in an order other than raster scan order. The H.264/AVC video structure divides an image into regions called slice groups. These macroblocks are processed in scan order from left to right and top to bottom. Each slice can be decoded independently from other slices. FMO scheme allows the

rearrangement of each macroblock into different slice group called FMO mapping. FMO in H.264/AVC have six pre-defined mapping as defined in the standard and an explicit mapping. In explicit FMO mapping, each macroblock can be assigned freely to any slice group using an MBAmap (MacroBlock Allocation map). The MBAmap consists of an identification number for each macroblock of the image that specifies to which slice group that macroblock belongs. The main goal of FMO is to provide a macroblock-level interleaving tool to spread out consecutive burst errors to several random ones in a frame.

However, by adapting error-resilience tools in the encoder such as FMO, does not guarantee acceptable video quality in high error rate channel condition. Even used with channel coding, practical channel encoding and decoding schemes for video transmission do not provide perfect error recovery from transmission errors because this would require large bandwidth overhead which is impractical in low bit-rate channel. In practice, a certain amount of errors can be tolerated at the decoder since the human visual perception can tolerate some degrees of distortion and visual artifacts. In order to further alleviate quality degradation caused by those errors, error concealment techniques at the decoder are necessary. The choice of error concealment techniques used will contribute to the improvement of the received video quality at certain extent. Error concealment can be classified into three categories: spatial, temporal and hybrid (Tsekeridou, 2000). For spatial error concealment (Wang, 1998), the information from surrounding correctly received or concealed blocks are used for reconstructing the damaged area. In temporal error concealment (Zhang, 2000), the information of the related blocks from the blocks in the previous frame is used to conceal lost blocks. Hybrid error concealment employs both spatial and temporal information for error concealment.

The main contributions are divided two parts. In the first part, the rationale to design the proposed framework based on our previous investigation on utilizing explicit FMO in wireless video transmission in (Hantanong, 2005). To design explicit FMO, the important consideration is to find a meaningful indicator for MB importance such that the insightful map could be generated. The second consideration is that the map is generated through one-pass or two-pass encoding in which this factor reflects the encoding delay. The work in (Hantanong, 2005) proposed the use of the encoder’s macroblock coded-bit-count which acts as spatial information as

indicators for a choice of macroblock-address-map of each picture. Therefore, this dissertation proposes distortion measure based on the concealment error which acts as temporal information as indicators for a choice of macroblock-address-map of each picture. And different methods to measure MB importance have been used to generate explicit FMO map but these methods use only a simple non-motion compensated error concealment at the decoder. We investigated the use of distortion measure that considers spatial and temporal error concealment schemes simulated at the encoder in designing explicit FMO map. However all previously proposed methods (Hantanong, 2005; Aramvith, 2006) used a two-pass encoding scheme. In the first pass, spatial and temporal information are collected to generate explicit FMO maps. In the second pass, the video sequence is encoded using the explicit FMO maps. To avoid an incurred delay from two-pass encoding, in this dissertation, we propose to generate explicit map using one-pass encoding using spatial and temporal information as well as side information extracted from error concealment at the encoder. To improve the video quality at the decoder, the error concealment method is chosen depending on the residual information that is correlated with the FMO map generation process.

In the second part, we extend our previously proposed works on FMO (Aramvith, 2006) by using bit-count information for generating FMO map and to further maximize the effectiveness of using FMO in conjunction with FEC and interleaving for wireless video transmission. We add FEC and a bit-level interleaving scheme as an added layer of protection together with the MB level interleaving scheme of the FMO. We predict the future condition of the channel to be good or bad state using a simplified Gilbert channel model. The parameters of the model are determined at the encoder from the observed feedback from the decoder. FMO combined with FEC and Interleaving will be applied to frames that have bad channel conditions as predicted by the channel model. And during good channel conditions only FMO is applied to provide some minimum level of protection for that frame. Due to the burst nature of the wireless channel, we develop a forward error correction and interleaving method that dynamically adapts the interleaving depth of the current frame according to the mean average burst length observed in the previous frames. In this method using two-pass Explicit FMO, therefore we modify to use one-pass explicit FMO, FEC and adaptive interleaving depth.

1.2 Objective

1. To investigate the error-resilient tools in H.264/AVC for wireless video transmission.
2. To propose one-pass explicit Flexible Macroblock Ordering (FMO) map based on spatial/temporal information for wireless video transmission.
3. To investigate the application of Forward Error Correction (FEC) code and interleaving to use with explicit FMO map.
4. To develop a framework for one-pass explicit FMO map and error concealment for H.264/AVC wireless video transmission

1.3 Working Scopes

1. Design a framework for combined explicit FMO map, FEC and adaptive interleaving depth in H.264/AVC wireless video transmission.
2. Design a framework of one-pass explicit FMO map scheme that take into consideration feedback spatial/temporal information and combined with error concealment method.
3. Evaluate the performance of the system by comparing with JM 9.2 at different explicit FMO maps.

1.4 Expected Benefits

1. Understand H.264/AVC video coding standard.
2. Understand the operation of H.264/AVC, FMO and error concealment.
3. Understanding methods for error recovery techniques, such as FEC and interleaving, used in wireless channel.
4. Develop algorithm for basic understanding combined FMO and error concealment video coding scheme.
5. Improve video quality in wireless video transmission which can apply to application such as video conference and real-time wireless video transmission.

1.5 Research Procedure

This dissertation is organized as follows. Some backgrounds on H.264/AVC, FMO in H.264/AVC, spatial/temporal error concealment schemes and wireless channel model are presented in Section II. Section III describes our proposed framework. In Section IV, the simulation results are discussed. Conclusions are presented in Section V.

The steps of our research are a follow.

1. Study encoder and decoder in H.264/AVC video coding standard.
2. Review relevant literatures about FMO map design, wireless channel modeling and error concealment in H.264/AVC.
3. Test JM reference model and previous work (Hantanong, 2005) by adaptive encoder and decoder parameters.
4. Design two-pass explicit FMO map using distortion measure based on concealment error which acts as temporal information.
5. Design two-pass explicit FMO map using distortion measure based on simulated spatial and temporal error concealment which act as spatial and temporal information
6. Design a framework of one-pass explicit FMO map from feedback information and apply appropriate error concealment method.
7. Simulate the transmission of video sequences of the wireless video transmission system.
8. Summarize and discuss the result compare with JM reference model and previous work (Hantanong, 2005).

CHAPTER II

BACKGROUND AND LITERATURE REVIEWS

BACKGROUND

2.1 Video Processing Theory

2.1.1 Nature Video Scenes and Capture

A typical real world or natural video scene is composed of multiple objects each with their own characteristic shape, depth, texture and illumination. The color and brightness of a natural video scene changes with varying degrees of smoothness throughout the scene, as call continuous tone. Characteristics of a typical natural video scene that are relevant for video processing and compression include spatial characteristics includes texture variation within scene, number and shape of objects, color and temporal characteristics such as object motion, changes in illumination and movement of the camera or viewpoint.

A natural visual scene is spatially and temporally continuous. Representing a visual scene in digital form involves sampling the real scene spatially usually on a rectangular grid in the video image plane and temporally as a series of still frames or components of frames sampled at regular intervals in time as shown in Figure 2.1. Digital video is the representation of a sampled video scene in digital form. Each spatial-temporal sample is represented as a number or set of numbers that describes the brightness or luminance and color of the sample.

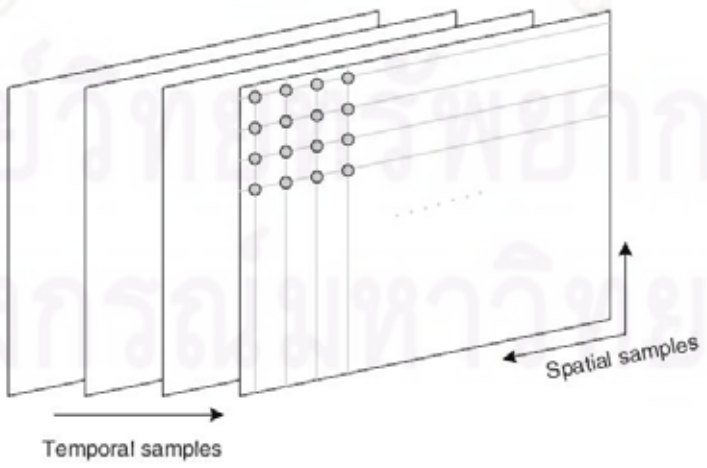


Figure 2.1 Spatial and temporal sampling of a video sequence

Spatial Sampling

The output of a CCD array is an analogue video signal, a varying electrical signal that represents a video image. Sampling the signal at a point in time produces a sampled image or frame that has defined values at a set of sampling points. The most common format for a sampled image is a rectangle with the sampling points positioned on a square or rectangular grid. A continuous-tone frame with two different sampling grids superimposed upon it. Sampling occurs at each of the intersection points on the grid and the sampled image maybe reconstructed by representing each sample as a square picture element. The visual quality of the image is influenced by the number of sampling points.

Temporal Sampling

A moving video image is captured by taking a rectangular ‘snapshot’ of the signal at periodic time intervals. Playing back the series of frames produces the appearance of motion. A higher temporal sampling frame rate gives apparently smoother motion in the video scene but requires more samples to be captured and stored. Frame rates below 10 frames per second are sometimes used for very low bit-rate video communications because the amount of data is relatively small but motion is clearly jerky and unnatural at this rate. Between 10 and 20 frames per second is more typical for low bit-rate video communications; the image is smoother but jerky motion may be visible in fast-moving parts of the sequence. Sampling at 25 or 30 complete frames per second is standard for television pictures with interlacing to improve the appearance of motion, see below; 50 or 60 frames per second produces smooth apparent motion at the expense of a very high data rate.

Frames and Fields

A video signal may be sampled as a series of complete frames, as progressive sampling, or as a sequence of interlaced fields, as interlaced sampling. In an interlaced video sequence, half of the data in a frame is sampled at each temporal sampling interval. A field consists of either the odd-numbered or even-numbered lines within a complete video frame and an interlaced video sequence as shown in Figure 2.2 contains a series of fields, each representing half of the information in a complete video frame The advantage of this sampling method is that it is possible to send twice as many fields per second as the number of frames in an equivalent progressive

sequence with the same data rate, giving the appearance of smoother motion. For example, a PAL video sequence consists of 50 fields per second and, when played back, motion can appears smoother than in an equivalent progressive video sequence containing 25 frames per second.

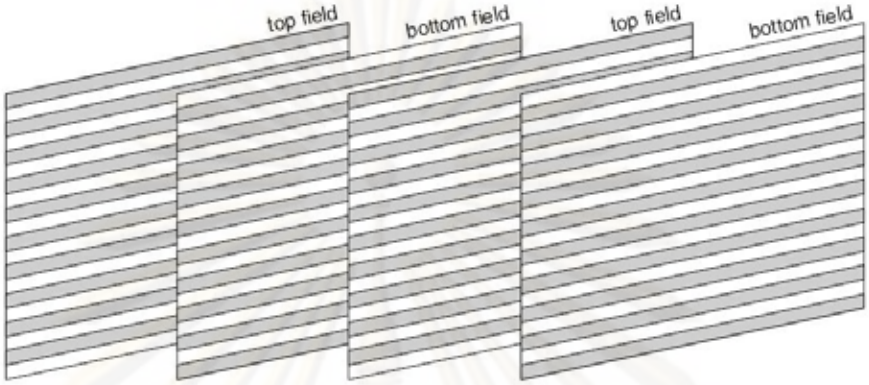


Figure 2.2 Interlaced video sequence

2.1.2 Color Space

Most digital video applications rely on the display of color video and so need a mechanism to capture and represent color information. A monochrome image requires just one number to indicate the brightness or luminance of each spatial sample. Color images, on the other hand, require at least three numbers per pixel position to represent color accurately. The method chosen to represent brightness or luminance or luma and color is described as a color space.

RGB

In the RGB color space, a color image sample is represented with three numbers that indicate the relative proportions of Red, Green and Blue. Any color can be created by combining red, green and blue in varying proportions. The red component consists of all the red samples, the green component contains all the green samples and the blue component contains the blue samples. The RGB color space is well-suited to capture and display of color images. Capturing an RGB image involves filtering out the red, green and blue components of the scene and capturing each with a separate sensor array. Color Cathode Ray Tubes (CRTs) and Liquid Crystal Displays (LCDs) display an RGB image by separately illuminating the red, green and blue components of each pixel according to the intensity of each component. From a

normal viewing distance, the separate components merge to give the appearance of true color.

YCbCr

The human visual system (HVS) is less sensitive to color than to luminance. In the RGB color space the three colors are equally important and so are usually all stored at the same resolution but it is possible to represent a color image more efficiently by separating the luminance from the color information and representing luma with a higher resolution than color. The YCbCr color space and its variations, sometimes referred to as YUV, is a popular way of efficiently representing color images. Y is the luminance component and can be calculated as a weighted average of R, G and B as shown in eq. (2.1):

$$Y == k_r R + k_g G + k_b B \tag{2.1}$$

where *k* are weighting factors.

The color information can be represented as color difference, chrominance or chroma components, where each chrominance component is the difference between *R*, *G* or *B* and the luminance *Y* as shown in eq. (2.2):

$$\begin{aligned} Cb &= B - Y \\ Cr &= R - Y \\ Cg &= G - Y \end{aligned} \tag{2.2}$$

The complete description of a color image is given by *Y* and three color differences *Cb*, *Cr* and *Cg* that represent the difference between the color intensity and the mean luminance of each image sample. The chroma components only have significant values where there is a large difference between the color component and the luma image. Note the strong blue and red difference components.

So far, this representation has little obvious merit since we now have four components instead of the three in RGB. However, *Cb* + *Cr* + *Cg* is a constant and so only two of the three chroma components need to be stored or transmitted since the third component can always be calculated from the other two. In the YCbCr color space, only the luma (*Y*) and blue and red chroma (*Cb, Cr*) are transmitted. YCbCr has an important advantage over RGB, that is the *Cr* and *Cb* components may be represented with a lower resolution than *Y* because the HVS is less sensitive to color

than luminance. This reduces the amount of data required to represent the chrominance components without having an obvious effect on visual quality. To the casual observer, there is no obvious difference between an RGB image and a YCbCr image with reduced chrominance resolution. Representing chroma with a lower resolution than luma in this way is a simple but effective form of image compression. An RGB image may be converted to YCbCr after capture in order to reduce storage and/or transmission requirements. Before displaying the image, it is usually necessary to convert back to RGB. The equations for converting an RGB image to and from YCbCr color space and vice versa are given eq.(2.3) and (2.4). There is no need to specify a separate factor k_g (because $k_b + k_r + k_g = 1$) and that *G* can be extracted from the YCbCr representation by subtracting *Cr* and *Cb* from *Y* , demonstrating that it is not necessary to store or transmit a *Cg* component as shown eq.(2.3) and (2.4).

$$\begin{aligned} Y &= k_r R + (1 - k_b - k_r) G + k_b B \\ Cb &= \frac{0.5}{1 - k_b}(B - Y) \\ Cr &= \frac{0.5}{1 - k_r}(R - Y) \end{aligned} \tag{2.3}$$

$$\begin{aligned} R &= Y + \frac{1 - k_r}{0.5} Cr \\ G &= Y - \frac{2k_b(1 - k_b)}{1 - k_b - k_r} Cb - \frac{2k_r(1 - k_r)}{1 - k_b - k_r} Cr \\ B &= Y + \frac{1 - k_b}{0.5} Cb \end{aligned} \tag{2.4}$$

ITU-R recommendation defines $k_b = 0.114$ and $k_r = 0.299$. Substituting into the above equations gives the following widely-used conversion equations as shown eq.(2.5):

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ Cb &= 0.564(B - Y) \\ Cr &= 0.713(R - Y) \end{aligned} \tag{2.5}$$

$$\begin{aligned} R &= Y + 1.402Cr \\ G &= Y - 0.344Cb - 0.714Cr \\ B &= Y + 1.772Cb \end{aligned} \tag{2.6}$$

2.1.3 Video Format

The video compression standards can compress a wide variety of video frame formats. In practice, it is common to capture or convert to one of a set of 'intermediate formats' prior to compression and transmission. The Common Intermediate Format (CIF) is the basis for a popular set of formats listed in Table 2.1. The choice of frame resolution depends on the application and available storage or transmission capacity. For example, 4CIF is appropriate for standard-definition television and DVD-video; CIF and QCIF are popular for videoconferencing applications; QCIF or SQCIF are appropriate for mobile multimedia applications where the display resolution and the bit rate are limited. Table 2.1 shows lists the number of bits required to represent one uncompressed frame in each format.

A widely-used format for digitally coding video signals for television production is ITU-R recommendation conversion to digital format and does not imply compression). The luminance component of the video signal is sampled at 13.5 MHz and the chrominance at 6.75 MHz to produce a 4:2:2 Y:Cb:Cr component signal. The parameters of the sampled digital signal depend on the video frame rate, 30 Hz for an NTSC signal and 25 Hz for a PAL/SECAM signal. The higher 30 Hz frame rate of NTSC is compensated for by a lower spatial resolution so that the total bit rate is the same in each case. The actual area shown on the display, the active area, is smaller than the total because it excludes horizontal and vertical blanking intervals that exist outside the edges of the frame. Each sample has a possible range of 0 to 255. Levels of 0 and 255 are reserved for synchronization and the active luminance signal is restricted to a range of 16 (black) to 235 (white).

Table 2.1 Video frame formats

Format	**Luminance resolution (horiz. x vert.)**	**Bit per frame (4:2:0, eight bits per sample)**
Sub-QCIF	128x96	147456
Quarter CIF(QCIF)	176x144	304128
CIF	352x288	1216512
4CIF	704x576	4866048

2.1.4 Quality

In order to specify, evaluate and compare video communication systems it is necessary to determine the quality of the video images displayed to the viewer. Measuring visual quality is a difficult and often imprecise art because there are so many factors that can affect the results. Visual quality is inherently subjective and is influenced by many factors that make it difficult to obtain a completely accurate measure of quality. For example, a viewer's opinion of visual quality can depend very much on the task at hand, such as passively watching a movie, actively participating in a videoconference, communicating using sign language or trying to identify a person in a surveillance video scene. Measuring visual quality using objective criteria gives accurate, repeatable results but as yet there are no objective measurement systems that completely reproduce the subjective experience of a human observer watching a video display.

2.1.4.1 Subjective Quality Measurement

Our perception of a visual scene is formed by a complex interaction between the components of the Human Visual System (HVS), the eye and the brain. The perception of visual quality is influenced by spatial fidelity and temporal fidelity. However, a viewer's opinion of quality is also affected by other factors such as the viewing environment, the observer's state of mind and the extent to which the observer interacts with the visual scene. A user carrying out a specific task that requires concentration on part of a visual scene will have a quite different requirement for good quality than a user who is passively watching a movie. For example, it has been shown that a viewer's opinion of visual quality is measurably higher if the viewing environment is comfortable and non-distracting. Other important influences on perceived quality include visual attention and the so-called recency effect. All of these factors make it very difficult to measure visual quality accurately and quantitatively.

2.1.4.2 Objective Quality Measurement

The complexity and cost of subjective quality measurement make it attractive to be able to measure quality automatically using an algorithm. Developers of video compression and video processing systems rely heavily on so-called objective quality measures. The most widely used measure is Peak Signal to Noise Ratio (PSNR) but

the limitations of this metric have led to many efforts to develop more sophisticated measures that approximate the response of real human observers.

PSNR

Peak Signal to Noise Ratio (PSNR) as shown in eq. (2.7) is measured on a logarithmic scale and depends on the mean squared error (MSE) of between an original and an impaired image or video frame, relative to $\left(2^n-1\right)^2$. The square of the highest-possible signal value in the image, where *n* is the number of bits per image sample.

$$PSNR_{dB} = 10\log_{10}\frac{\left(2^n-1\right)^2}{MSE} \tag{2.7}$$

MSE is the mean square error for the component for which PSNR is calculated. It is defined as in eq. (2.8)

$$MSE = \frac{1}{M \square N}\sum_{i=1}^{N}\sum_{j=1}^{M}\left[F\left(i,j\right)-F_o(i,j)\right]^2 \tag{2.8}$$

Where $M \square N$ is the size of the frame, $F\left(i,j\right)$ is the reconstructed frame and $F_o(i,j)$ is the original frame of the color components.

PSNR can be calculated easily and quickly and is therefore a very popular quality measure, widely used to compare the quality of compressed and decompressed video images. The PSNR measure suffers from a number of limitations. PSNR requires an unimpaired original image for comparison but this may not be available in every case and it may not be easy to verify that an original image has perfect fidelity. PSNR does not correlate well with subjective video quality measures. For a given image or image sequence, high PSNR usually indicates high quality and low PSNR usually indicates low quality. However, a particular value of PSNR does not necessarily equate to an absolute subjective quality.

2.1.5 Video Encoder and Decoder

Compression is the process of compacting data into a smaller number of bits. Video coding is the process of compacting or condensing a digital video sequence into a smaller number of bits. Raw or uncompressed digital video typically requires a

large and compression is necessary for practical storage and transmission of digital video.

Compression involves a complementary pair of systems, a encoder and a decoder. The encoder converts the source data into a compressed form prior to transmission or storage and the decoder converts the compressed form back into a representation of the original video data. The encoder and decoder pair is often described as a CODEC encoder and decoder as shown in Figure 2.3 (Richardson, 2003).

Data compression is achieved by removing redundancy, i.e. components that are not necessary for faithful reproduction of the data. Many types of data contain statistical redundancy and can be effectively compressed using lossless compression, so that the reconstructed data at the output of the decoder is a perfect copy of the original data. Unfortunately, lossless compression of image and video information gives only a moderate amount of compression. The best that can be achieved with current lossless image compression standards such as JPEG-LS is a compression ratio of around 3–4 times. Lossy compression is necessary to achieve higher compression. In a lossy compression system, the decompressed data is not identical to the source data and much higher compression ratios can be achieved at the expense of a loss of visual quality. Lossy video compression systems are based on the principle of removing subjective redundancy, elements of the image or video sequence that can be removed without significantly affecting the viewer's perception of visual quality.

Figure 2.3 Encoder and decoder

Most video coding methods exploit both temporal and spatial redundancy to achieve compression. In the temporal domain, there is usually a high correlation between frames of video that were captured at around the same time. Temporally adjacent frames are often highly correlated, especially if the temporal sampling frame rate is high. In the spatial domain, there is usually a high correlation between pixels that are close to each other, i.e. the values of neighboring samples are often very similar.

2.2 H.264/AVC video Coding Standard Overview

The Moving Picture Expert Group and the Video Coding Expert Group (MPEG and VCEG) have developed a new standard that promises to outperform the earlier MPEG-4 and H.263 standards, providing better compression of video images. The new standard is entitled Advanced Video Coding (AVC) and is published jointly as Part 10 of MPEG-4 and ITU-T Recommendation H.264

2.2.1 Terminology and H.264 Codec

2.2.1.1 Terminology

Some of the important terminology adopted in the H.264 standard is as follow:

A field of interlaced video or a frame of progressive or interlaced video is encoded to produce a coded picture. A coded frame has a frame number, which is not necessarily related to decoding order, and each coded field of a progressive or interlaced frame has an associated picture order count, which defines the decoding order of fields. Previously coded pictures may be used for inter prediction of further coded pictures. Reference pictures are organized into one or two lists, described as list 0 and list 1.

A coded picture consists of a number of macroblocks, each containing 16x16 luma samples and associated chroma samples, as 8x8 Cb and 8x8 Cr samples in current standard. Within each picture, macroblocks are arranged in slices, where a slice is a set of macroblocks in raster scan order. An I slice may contain only I macroblock types, a P slice may contain P and I macroblock types and a B slice may contain B and I macroblock types. There are two further slice types SI and SP.

I macroblocks are predicted using intra prediction from decoded samples in the current slice. A prediction is formed either for the complete macroblock or for each 4x4 block of luma samples and associated chroma sample in the macroblock.

P macroblocks are predicted using inter prediction from reference pictures. An inter coded macroblock may be divided into macroblock partitions, i.e. blocks of size 16x16, 16x8, 8x16 or 8x8 luma samples and associated chroma sample. If the 8x8 partition size is chosen, each 8x8 sub-macroblock may be further divided into sub-macroblock partitions of size 8x8, 8x4, 4x8 or 4x4 luma samples and associated chroma sample. Each macroblock partition may be predicted from one picture in list

0. If present, every sub-macroblock partition in a sub-macroblock is predicted from the same picture in list 0.

B macroblocks are predicted using inter prediction from reference pictures. Each macroblock partiotion may be predicted from one or two reference picture, one picture in list 0 and/or one picture in list 1. If present, every sub-macroblock partition in a sub-macroblock is predicted from one or two reference pictures, one picture in list 0 and/or one picture in list 1.

2.2.1.2 The H.264 Codec

H.264 does not explicitly define a CODEC but rather defines the syntax of an encoded video bit-stream together with the method of decoding this bit-stream. A compliant encoder and decoder are likely to include the functional elements shown in Figure 2.4 and Figure 2.5. With the exception of the de-blocking filter, most of the basic functional elements, as prediction, transform, quantization, entropy encoding, are present in previous standards, such as MPEG-1, MPEG-2, MPEG-4, H.261 and H.263, but the important changes in H.264 occur in the details of each functional block.

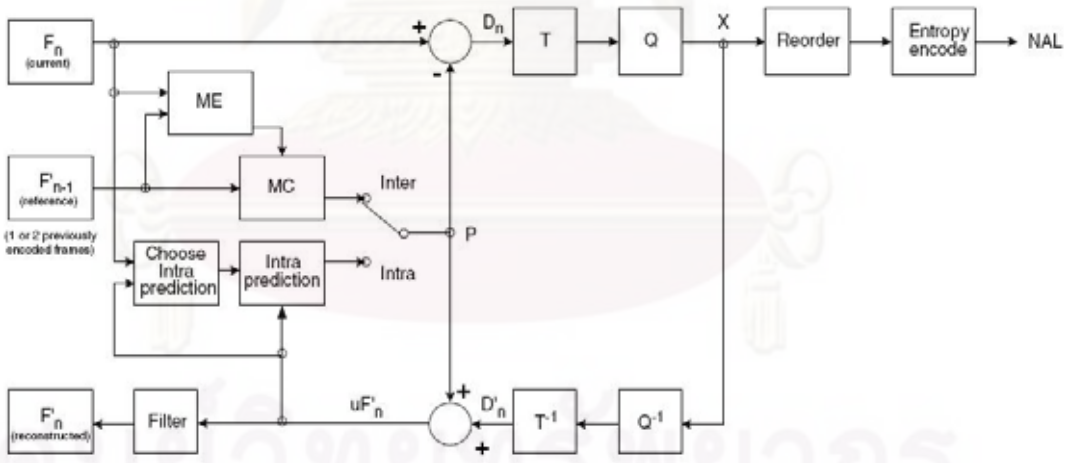


Figure 2.4 H.264 encoder

Figure 2.4 shows the Encoder (Richardson, 2003) includes two dataflow paths, a forward path and a reconstruction path. The dataflow path in the decoder shown in Figure 2.5 is shown from right to left to illustrate the similarities between encoder and decoder. Before examining the detail of H.264, we will describe the main steps in encoding and decoding a frame of video. The following description is simplified in order to provide an overview of encoding and decoding. The term block is used to

denote a macroblock partition or sub-macroblock or a 16 × 16 or 4× 4 block of luma samples and associated chroma samples.

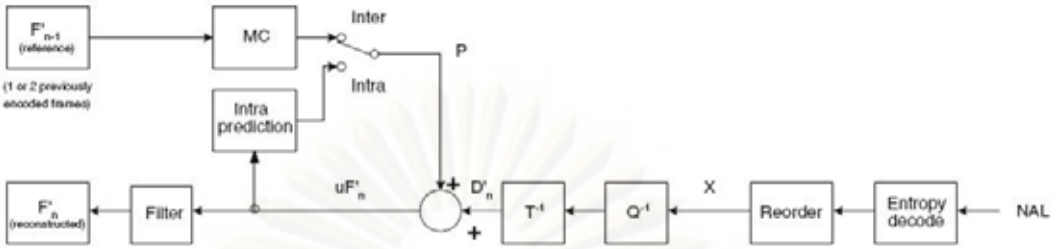


Figure 2.5 H.264 Decoder

Encoder (forward Path)

An input frame or field F_n is processed in units of a macroblock. Each macroblock is encoded in intra or inter mode and, for each block in the macroblock, a prediction PRED, marked P in Figure 2.4, is formed based on reconstructed picture samples. In Intra mode, PRED is formed from samples in the current slice that have previously encoded, decoded and reconstructed, uF_n' in the figures, In Inter mode, PRED is formed by motion-compensated prediction from one or two reference picture selected from the set of list 0 and/or list 1 reference pictures. In the figures, the reference picture is shown as the previous encoded picture F_{n-1}' but the prediction reference for each macroblock partition in inter mode may be chosen from a selection of past or future pictures in display order that have already been encoded, reconstructed and filtered.

The prediction PRED is subtracted from the current block to produce a residual block D_n that is transformed using a block transform and quantized to give X, a set of quantized transform coefficients which are reordered and entropy encoded. The entropy-encoded coefficients, together with side information required to decode each block within the macroblock from the compressed bit-stream which is passed to a Network Abstraction Layer (NAL) for transmission or storage.

Encoder (Reconstruction Path)

As well as encoding and transmitting each block in a macroblock, the encoder decodes (reconstructs) it to provide a reference for further predictions. The

coefficients X are scaled (Q^{-1}) and inverse transformed (T^{-1}) to produce a difference block D'_n. The prediction block PRED is added to D'_n to create a reconstructed block uF'_n a decoded version of the original block; u indicates that it is unfiltered. A filter is applied to reduce the effects of blocking distortion and the reconstructed reference picture is created from a series of blocks F'_n.

Decoder

The decoder receives a compressed bit-stream from the NAL and entropy decodes the data elements to produce a set of quantized coefficients X. These are scaled and inverse transformed to give D'_n. Using the header information decoded from the bit-stream, the decoder creates a prediction block PRED, identical to the original prediction PRED formed in the encoder. PRED is added to D'_n produce uF'_n which is filtered to create each decoded block F'_n.

2.2.2 H.264 Structure

2.2.2.1 Profile and Level

H.264 defines a set of three Profiles, each supporting a particular set of coding functions and each specifying what is required of an encoder or decoder that complies with the Profile. The Baseline Profile supports intra and inter-coding using I/P-slices and entropy coding with context-adaptive variable-length codes (CAVLC). The Main Profile includes support for interlaced video, inter-coding using B-slice, inter coding using weighted prediction and entropy coding using context-based arithmetic coding (CABAC). The Extended Profile does not support interlaced video or CABAC but adds modes to enable efficient switching between coded bit-streams (SP- and SI-slices) and improved error resilience (Data Partitioning). Potential applications of the Baseline Profile include video telephony, video conferencing and wireless communications; potential applications of the Main Profile include television broadcasting and video storage; and the Extended Profile may be particularly useful for streaming media applications. However, each Profile has sufficient flexibility to support a wide range of applications and so these examples of applications should not be considered definitive.

Figure 2.6 shows the relationship between the three Profiles and the coding tools supported by the standard. It is clear from this figure that the Baseline Profile is a subset of the Extended Profile, but not of the Main Profile. Performance limits for CODECs are defined by a set of levels, each placing limits on parameters such as sample processing rate, picture size, coded bit rate and memory requirements.

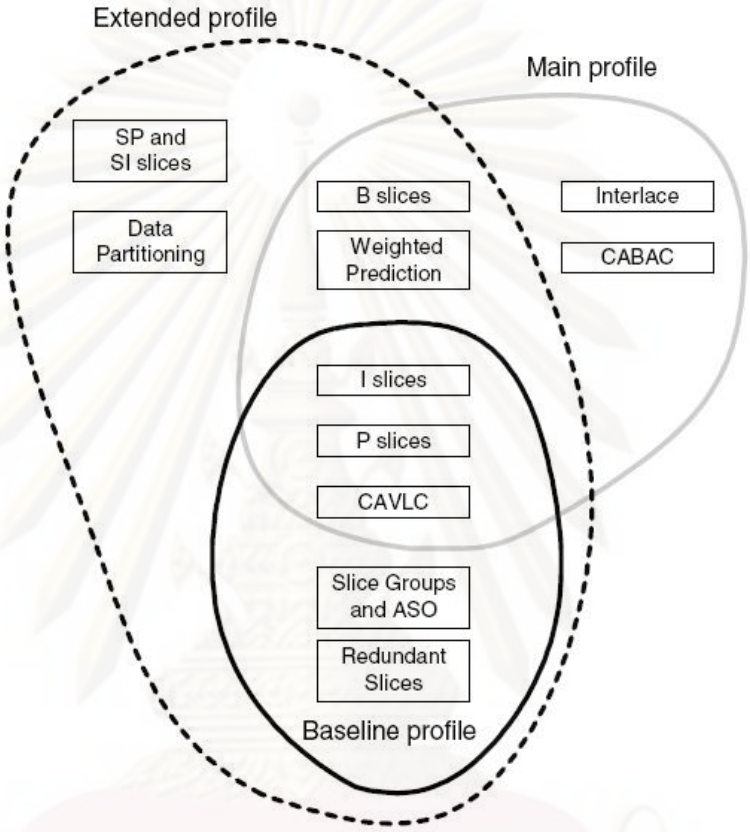


Figure 2.6 H.264 baseline, main and extended profiles

2.2.2.2 Video Format

H.264 supports coding and decoding of 4:2:0 progressive or interlaced video1 and the default sampling format for 4:2:0 progressive frames is shown in Figure 2.7. In the default sampling format, chroma (Cb and Cr) samples are aligned horizontally with every 2nd luma sample and are located vertically between two luma samples. An interlaced frame consists of a top field and a bottom field separated in time and with the default sampling format shown in Figure 2.8.

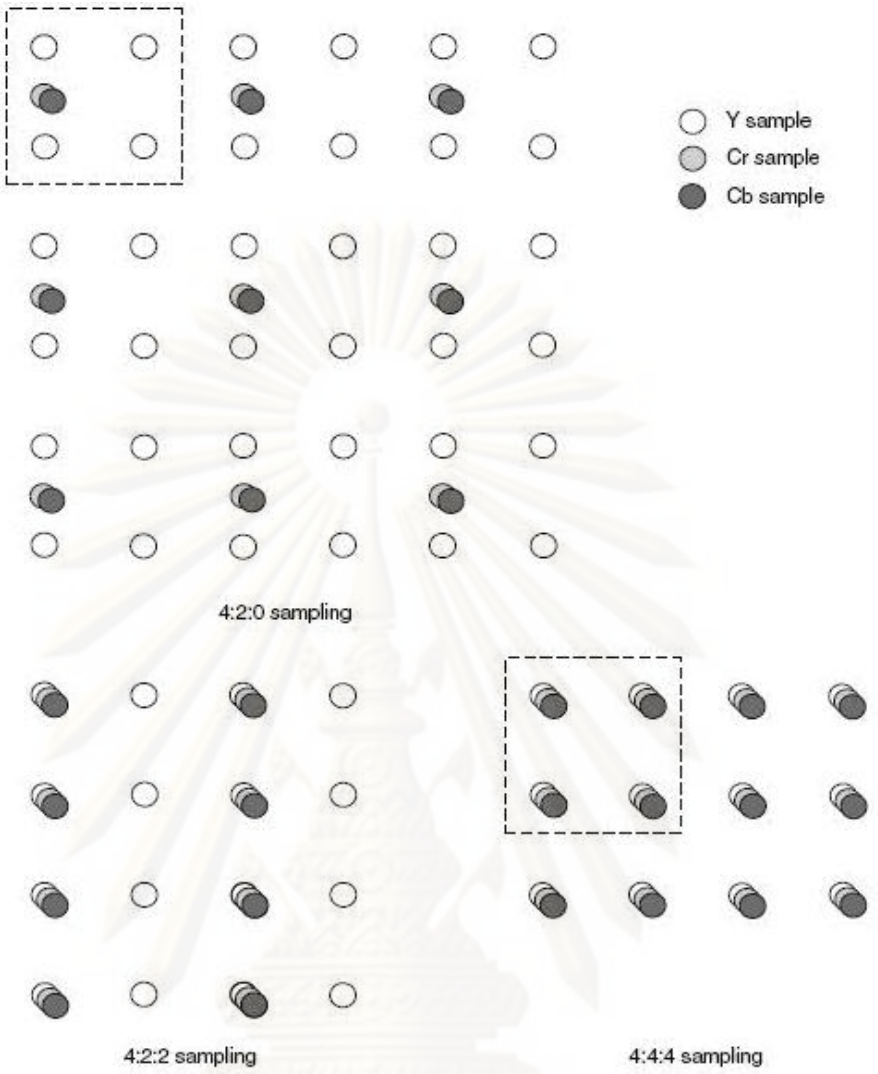


Figure 2.7 4:2:0, 4:2:2 and 4:4:4 sampling pattern (pregressive)

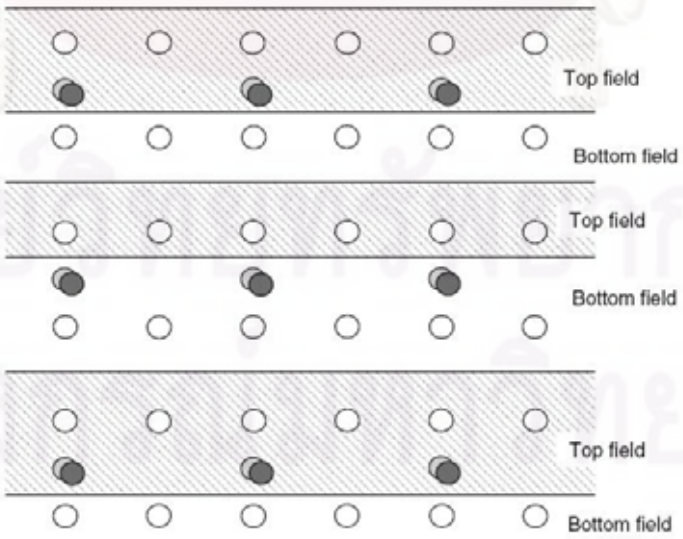


Figure 2.8 Allocation of 4:2:0 sampler to top and bottom fields

2.2.2.3 Coded Data Format

H.264 makes a distinction between a Video Coding Layer (VCL) and a Network Abstraction Layer (NAL). The output of the encoding process is VCL data, a sequence of bits representing the coded video data, which are mapped to NAL units prior to transmission or storage. Each NAL unit contains a Raw Byte Sequence Payload (RBSP), a set of data corresponding to coded video data or header information. A coded video sequence is represented by a sequence of NAL units, shown in Figure 2.9, that can be transmitted over a packet-based network or a bit-stream transmission link or stored in a file. The purpose of separately specifying the VCL and NAL is to distinguish between coding-specific features at the VCL and transport-specific features at the NAL.

Figure 2.9 Sequence of NAL units

2.2.2.4 Reference Pictures

An H.264 encoder may use one or two of a number of previously encoded pictures as a reference for motion-compensated prediction of each inter coded macroblock or macroblock partition. This enables the encoder to search for the best ‘match’ for the current macroblock partition from a wider set of pictures than just the previously encoded picture. The encoder and decoder each maintain one or two lists of reference pictures, containing pictures that have previously been encoded and decoded occurring before or after the current picture in display order. Inter coded macroblocks and macroblock partitions in P slices are predicted from pictures in a single list, list 0. Inter coded macroblocks and macroblock partitions in a B slice may be predicted from two lists, list 0 and list 1.

2.2.2.5 Slices

A video picture is coded as one or more slices, each containing an integral number of macroblocks from 1 MB per slice to the total number of macroblocks in a picture. The number of macroblocks per slice need not be constant within a picture. There is minimal inter-dependency between coded slices which can help to limit the propagation of errors. There are five types of coded slice, as shown in Table 2.2, and a

coded picture may be composed of different types of slices. For example, a Baseline Profile coded picture may contain a mixture of I and P slices and a Main or Extended Profile picture may contain a mixture of I, P and B slices.

Table 2.2 H.264 slice modes

Slice	Description	Profiles
I (Intra)	Contain only I macroblocks (each block/macroblock is predicted from previously coded data within the same slice).	All
P (Predicted)	Contain only P macroblocks (each macroblock/ macroblock partition is predicted from one list 0 reference picture) and/or I macroblock.	All
B (bi-predictive)	Contain only B macroblocks (each macroblock/ macroblock partition is predicted from one list 0 and/or a list 1 reference picture) and/or I macroblock.	Extended and Main
SP (Switching P)	Facilitates switching between coded streams; contains P and/or I macroblocks	Extended
SI (Switching I)	Facilitates switching between coded streams; contains SI macroblocks (a special type of intra coded macroblock).	Extended

Figure 2.10 shows a simplified illustration of the syntax of a coded slice. The slice header defines the slice type and the coded picture that the slice belongs to and may contain instructions related to reference picture management. The slice data consists of a series of coded macroblocks and/or an indication of skipped macroblocks. Each MB contains a series of header elements, shown in Table 2.2, and coded residual data.

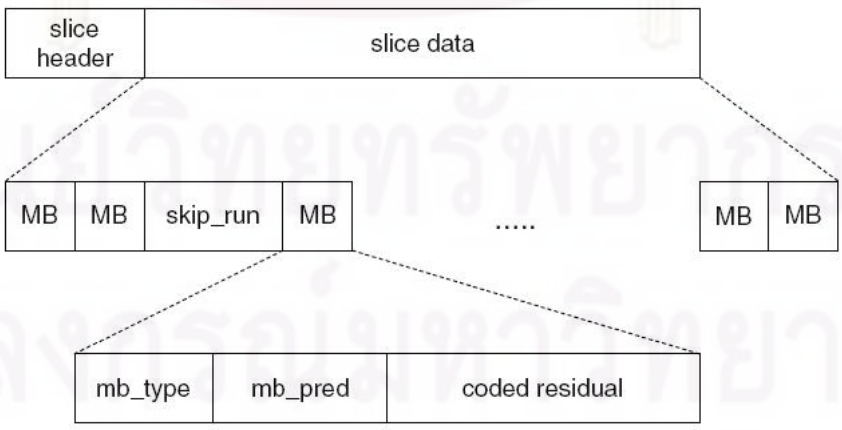


Figure 2.10 Slice syntax

2.2.2.6 Macroblocks

A macroblock contains coded data corresponding to a 16×16 sample region of the video frame (16×16 luma samples, 8×8 Cb and 8 × 8 Cr samples) and contains the syntax elements described in Table 2.3. Macroblocks are addressed in raster scan order within a frame.

Table 2.3 Macroblock syntax elements

mb_type	Determines whether the macroblock is coded in intra or inter (P or B) mode; determines macroblock partition size.
mb_pred	Determines intra prediction modes (intra macroblocks): determines list 0 and/or list 1 references and differentially coded motion vectors for each macroblock partition (inter macroblocks, except for inter macroblocks with 8x8 macroblock partition size).
sub_mb_pred	(Inter MBs with 8x8 macroblock partition size only) Determines sub-macroblock partition size for each sub-macroblock: list 0 and/or list1 reference for each macroblock partition: differentially coded motion vector for each macroblock sub-partition.
coded_block_pattern	Identifies which 8x8 blocks (luma and chroma) contain coded transform coefficients.
mb_qp_delta	Changes the quantize parameter.
residual	Coded transform coefficients corresponding to the residual image sample after prediction.

2.3 Flexible Macroblock Ordering (FMO) in H.264

H.264/AVC is a new standard for digital video compression jointly developed by ITU-T's Video Coding Experts Group (VCEG) and ISO/IEC's Moving Picture Experts Group (MPEG). Aside from having efficient coding algorithms, the H.264/AVC specifications define a set of error resilience tools. The most striking tool is Flexible Macroblock Ordering (FMO).

FMO is one of the new characteristics of the H.264/AVC standard; it allows the possibility of dividing an image into regions called slice groups. Each slice group can also be divided in several slices; here a slice is defined as a sequence of macroblocks that belong to the same slice group. These macroblocks are processed in a scan order and a slice can be decoded independent of other slices. FMO consists of deciding to which slice each macroblock of the image belongs. Each macroblock can be assigned freely to a slice group using an MBAmap. The MBAmap consists of an

identification number for each macroblock of the image that specifies to which slice group that macroblock belongs. The number of slice groups is limited to 8 for each picture to prevent complex allocation schemes. If FMO is deactivated, the images will be composed of a single slice with the macroblocks in a scan order. The use of FMO is totally compatible with any type of inter-frame prediction.

With this technique, errors can be corrected easily by exploiting the spatial redundancy of the images. It's a good idea to choose the slice groups in a way that no macroblock and its neighbors belong to the same group. Therefore, if a slice is lost during transmission, it's easy to reconstruct the lost blocks with the information of the neighboring blocks. By considering the transmission characteristics of these slices: each slice is transmitted independently in separate units called packets and each packet contains in its own header the information to decode itself without any other packet's information (if the images used as reference are the same in the encoder and the decoder side).

2.3.1 Type of FMO

When using FMO, the image can be divided into different scan patterns of macroblocks. FMO consists of 7 different types, labeled Type 0 to Type 6 as shown in Figure 2.11. These patterns can be exploited when storing and transmitting the MBAmap.

Type 0 (interleave): uses run lengths which are repeated to fill the frame. Therefore only those run lengths have to be known to rebuild the image on the decoder side.

Type 1 (dispersed): also known as scattered slices; it uses a mathematical function, which is known in both the encoder and the decoder, to spread the macroblocks. The distribution in the figure, in which the macroblocks are spread forming a chess board, is very common.

Type 2 (foreground and Background): is used to mark rectangular areas, so called regions of interest. In this case the coordinates top-left and bottom-right of the rectangles is saved in the MBAmap.

Type 3-5 (Box-out, Raster and Wipe): are dynamic types that let the slice groups grow and shrink over the different pictures in a cyclic way. Only the growth rate, the direction and the position in the cycle have to be known.

Type 6 (Explicit): is the most random one and allows full flexibility to the user. All the other ones contain a certain pattern.

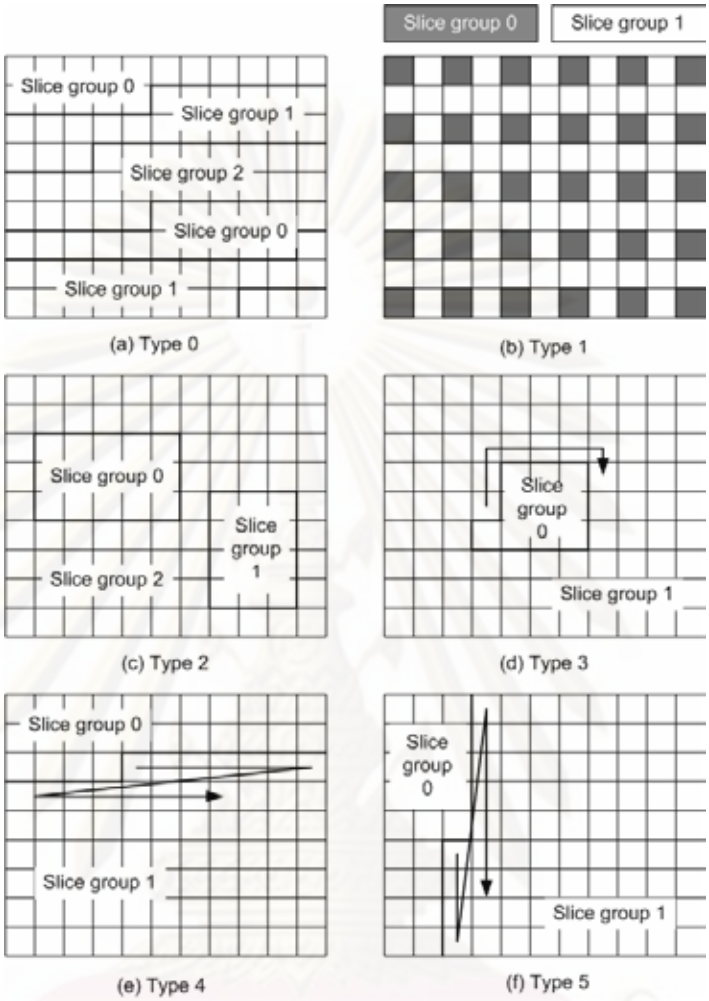


Figure 2.11 Type of flexible macroblock ordering

2.4 Error Propagation

The visual artifact caused by the bit stream error has different shapes and ranges depending on which part of video data stream is affected by the transmission error. Therefore these artifacts can be described in 2 levels: Slice level and GOP level

2.4.1 Slice Level

In the slice level these artifacts are caused by either desynchronization of the variable length code or the loss of the reference in a spatial prediction.

2.4.1.1 Variable Length Code

The quantized transform coefficients are entropy coded using a variable length code (VLC) (T. Thaipanich, 2008) which means that the codewords have variable lengths. The advantages of this kind of code consist in the fact that they are more efficient in the sense of representing the same information using fewer bits on average, reducing therefore the bit rate. That is possible if some symbols are more probable than others. The most frequent symbols will correspond to the shorter codewords, and the rare symbols will correspond to the longer codewords. However, variable length codes between the codewords may be determined in a wrong way, and the decoding process may desynchronize. Figure 2.12 describes how just one erroneous bit shown in red can desynchronize the whole sequence.

Encoding:

C B D B E E F A C

1100 10 1101 10 1110 1110 1111 0 1100

Bit error:

1100 10 0101 10 1110 1110 1111 0 1100

Decoding:

1100 10 0 10 1101 1101 1101 1110 1100

C B A B D D D E C

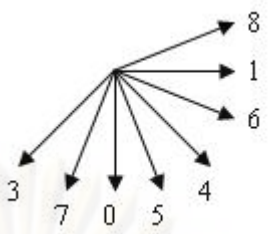
Symbol	Codeword
A	0
B	10
C	1100
D	1101
E	1110
F	1111

Figure 2.12 Example of a VLC desynchronization

2.4.1.2 Spatial Prediction

The H.264/AVC performs intra prediction in the spatial domain. Even for an intra picture, every block of data is predicted from its neighbors before being transformed and coefficients generated for inclusion in the bit stream. As a first step in coding of a macroblock in intra mode, spatial prediction is performed on either 4x4 or 16x16 luminance blocks. Although, in principle, 4x4 block prediction will offer more efficient prediction compared to a 16x16 block, in reality, taking into account the mode decision overhead, sometimes the 16x16 block based prediction may offer overall better coding efficiency. Figure 2.13 and Figure 2.14 shows two types of luminance intra coding (ITU-T recommendation H.264, 2003).

Num	Intra 16x16 Pred Mode
0	vertical
1	horizontal
2	dc
3	diagonal_down_left
4	diagonal_down_right
5	vertical_right
6	horizontal_down
7	vertical_left
8	horizontal_up



M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Figure 2.13 Left: intra 4x4 predictions are conducted for samples a-p of a block by 9 diferent modes. right: 8 prediction direction for intra 4x4 prediction.

Num	Intra 16x16 Pred Mode
0	vertical
1	horizontal
2	dc
3	plane

Figure 2.14 Intra 16x16 predictions modes.

There are two 8x8 blocks of chroma in a macroblock one corresponding to each of the components, Cb and Cr. Each 8x8 block of chroma is subdivided into 4, 4x4 blocks such that each 4x4 block depending on its location uses a pre-fixed prediction using decoded pixels of corresponding chroma component. Figure 2.15 illustrate variable size of macroblocks.

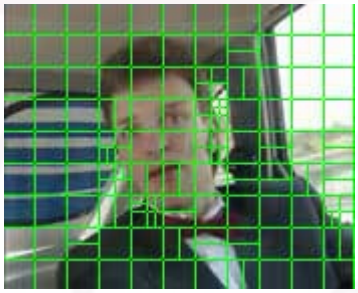


Figure 2.15 Frame divided into multiple macroblock of 16x16, 8x8, 8x4, 4x8 and 4x4 variable sizes to present different coding profiles.

Inter prediction: The inter prediction block includes both motion estimation (ME) and motion compensation (MC). It generates a predicted version of a rectangular array of pixels, by choosing similarly sized rectangular arrays of pixels

from previously decoded reference pictures and translating the reference arrays to the positions of the current rectangular array. Figure 2.16 depicts inter-prediction.

In Figure 2.16, a half-pel is interpolated from neighboring integer-pel samples using a 6-tap Finite Impulse Response filter with weights (1, -5, 20, 20, -5, 1) / 32, quarter-pel is produced using bilinear interpolation between neighboring half- or integer-pel samples (Kwon, 2006).

In the AVC, the rectangular arrays of pixels that are predicted using MC can have the following sizes: 4x4, 4x8, 8x4, 8x8, 16x8, 8x16, and 16x16pixels. The translation from other positions of the array in the reference picture is specified with quarter pixel precision. In case of 4:2:0 format, the chroma MVs have a resolution of 1/8 of a pixel. They are derived from transmitted luma MVs of 1/4 pixel resolution, and simpler filters are used for chroma as compared to luma. Figure 2.17 illustrates the partitioning of the macroblock for motion compensation (Kwon, 2006)

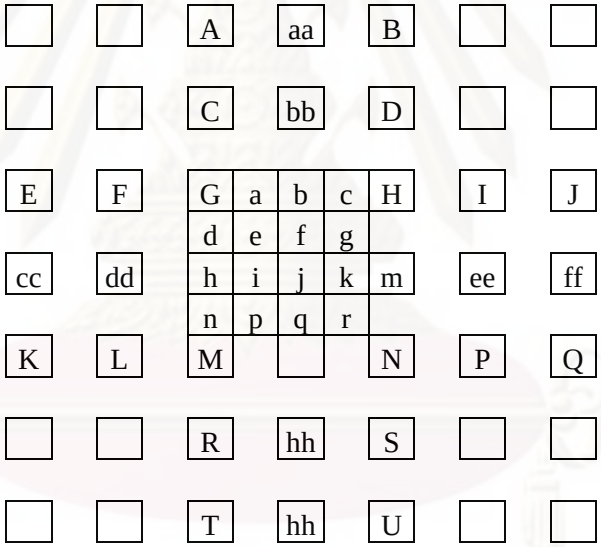


Figure 2.15 Inter predictions in H.264

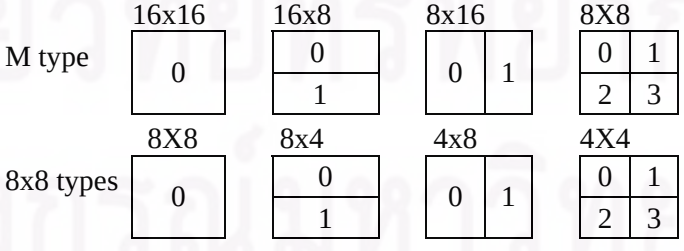


Figure 2.17 Segmentations of the macroblock for motion compensation.

H.264/AVC standard supports multi-picture motion-compensated prediction. That is, more than one prior-coded picture can be used as a reference for motion-

compensated prediction as shown in Figure 2.18. In addition to the motion vector, the picture reference parameters (Δ) are also transmitted. Both the encoder and decoder have to store the reference pictures used for Inter-picture prediction in a multi-picture buffer. The decoder replicates the multi-picture buffer of the encoder, according to the reference picture buffering type and any memory management control operations that are specified in the bit stream (Schäfer, 2003).

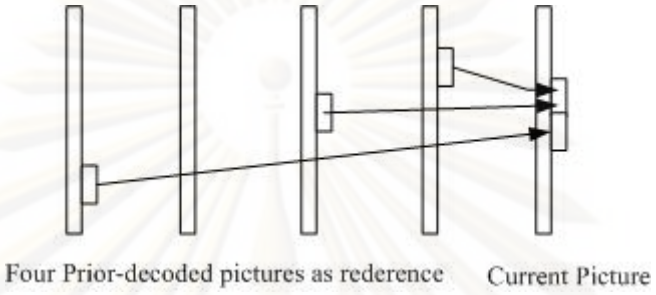


Figure 2.18 Multi-frame motion compensation in H.264.

The H.264/AVC decoder takes in the encoded bit stream as input and gives raw YUV (Y Luminance, (U, V)-Chrominance) video frames as output. The header or syntax information and slice data with motion vectors is extracted by the entropy decoder block through which the bit stream is passed. Next the residual block data is extracted by means of inverse scan and inverse quantizer. An inverse transform is carried out on all the blocks in order to map them from the transform domain to the pixel domain. A predicted block is formed using motion vectors, and previously decoded reference frames if the block is found to be inter coded. Then the predicted block and residual block are combined to reconstruct the complete frame. This decoded frame is then presented to the user after it is passed through a de-blocking filter.

2.4.2 GOP Level

Due to the temporal and spatial predictions of the images, the image distortion caused by an erroneous MB is not restricted to that MB. Since MBs are spatially and/or temporally dependent on neighboring MBs, the errors can also propagate in following frames and in the same frame. Error propagation represents a problem for error concealment because if the error concealed picture differs from the original picture, the error will propagate until the next I frame occurs or until the beginning of the next GOP. If more frames per GOP are used to improve compression, there will be degradation in video quality since the error can propagate over more frames.

2.5 Error Concealment

The loss of transmitted data packets influences the quality of the received video. This problem is caused by the band limited channel used by the mobile communication networks. Since the real time transmission of video stream limits the channel delay, it is not possible to retransmit all erroneous or lost packets. Therefore there is a need for post processing methods, which try to reduce the visual artifacts caused by bit stream error after locating the missing or defective parts of video data (Liu, 2005). Error concealment methods which will be implemented on the receiver side restore the missing and corrupt video content by using the previously decoded video data. The error concealment benefits from the spatial and temporal correlations between the video blocks within one frame or more than one frame within the video sequence. Therefore the error concealment methods are implemented both in the spatial domain and temporal domain. The spatial domain based error concealment uses the video information from the neighboring blocks to restore the missing pixels within a specified area. The temporal domain based error concealment uses the video information from the blocks lying in the previous and next frames to restore the missing pixels within a specified area (Kumar, 2006) and (Thaipanich, 2008). There are some assumptions adopted in this dissertation to concentrate and limit the efforts on the presentation of the error concealment methods:

- The missing part of a video content is limited to one macroblock
- The location of the missing macroblocks is known
- Features like data partitioning belonging to one macroblock such as motion vectors, prediction mode and residuals are lost.

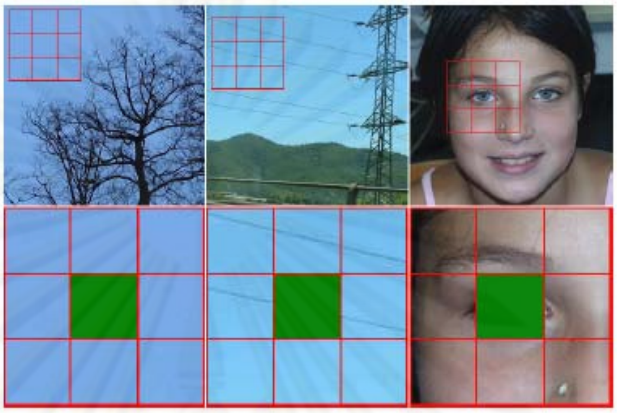
2.5.1 Error Concealment in Spatial Domain

Error concealment in spatial domain is presented

- *Smoothness of the neighborhood*

The smoothness of the neighborhood of the erroneous macroblock will determine the difficulty of the spatial concealment. In Figure 2.19 there are three situations. In Figure 2.19(a), it is going to be easy to reconstruct the lost macroblock because the neighborhood is very smooth with almost no difference between the neighboring macroblocks. In Figure 2.19(b), it is going

to be a little bit more difficult; we have to look for the edges and then, recover the line. The Figure 2.19(c) is an example where the neighbors can not help us to recover the macroblock because they do not give any information about the lost part (in this case, the eye).



a	b	c
d	e	f

Figure 2.19 Different smoothness of the neighborhood

The spatial redundancy in image and video signals is always present. Here the interpixel difference between adjacent pixels for an image is determined. The interpixel difference is defined as the average of the absolute difference between a pixel and its four surrounding pixels. This property has been exploited to perform error concealment. All error concealment methods in the spatial domain are based on the same idea which says that the pixel values within the damaged macroblocks can be recovered by a specified combination of the pixels surrounding the damaged macroblocks.

2.5.2.1 Weighted Averaging

The simplest method of spatial error concealment is called weighted averaging (ITU-T Recommendation H.264, 2001), because the missing pixel values can be recovered by calculating the average pixel values from the four pixels in the four 1-pixel wide boundaries of the damaged macroblock weighted by the distance between the missing pixel and the four macroblocks boundaries (upper, down, left and right boundaries) as shown in Figure 2.20. Let x and y are the vertical and horizontal coordinate of MB. Let i and j are the vertical and horizontal pixel coordinate in MB. Macroblock size is 16x16 pixels. Let $mb_T(x,y), mb_B(x,y), mb_L(x,y), mb_R(x,y)$ are the neighboring MBs of $mb(x,y)$ in the top, bottom, left and right directions.

$p_T(j), p_B(j), p_L(j), p_R(j)$ are neighboring vector for spatial error concealment can be represented in eq.(2.9):

$$\begin{aligned} p_T(j) &= mb_T(x, y; 15, j), \quad 0 \le j \le 16 \\ p_B(j) &= mb_B(x, y; 0, j), \quad 0 \le j \le 16 \\ p_L(i) &= mb_L(x, y; i, 15), \quad 0 \le i \le 16 \\ p_R(i) &= mb_R(x, y; i, 0), \quad 0 \le i \le 16 \end{aligned} \tag{2.9}$$

The weight for pixels is calculated in eq.(2.10)

$$\begin{aligned} w_T(i) &= 16 - i \\ w_B(i) &= i + 1 \\ w_L(i) &= 16 - j \\ w_R(i) &= j + 1 \end{aligned} \tag{2.10}$$

The reconstructed pixel ($mb_s(x, y, i, j)$), as shown in Figure 2.28, in the lost MB is then obtained as in eq.(2.11):

$$mb_s(x, y, i, j) = \frac{w_T(i)\, p_T(j) + w_B(i)\, p_B(j) + w_L(j)\, p_L(i) + w_R(j)\, p_R(i)}{w_T(i) + w_B(i) + w_L(j) + w_R(j)} \tag{2.11}$$

Only correctly received MBs are used for concealment if at least two such MBs are available. Otherwise, concealed MBs are also used.

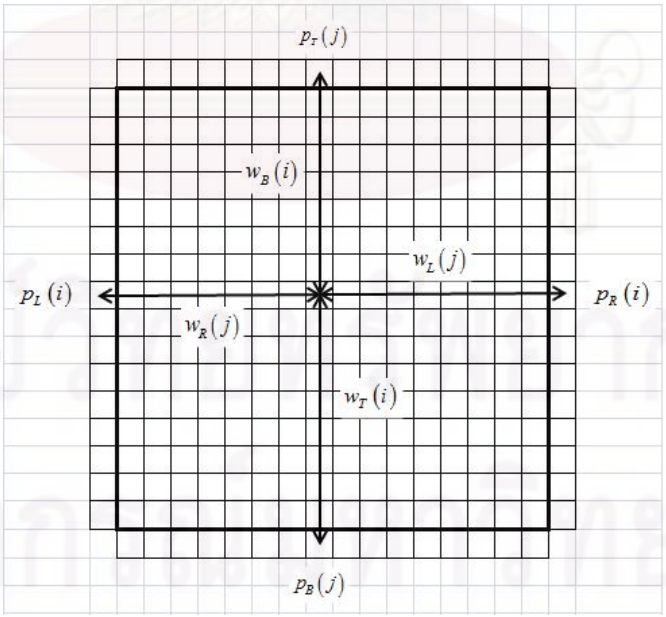


Figure 2.20 Weighted averaging

Another way to implement weighted averaging is called block based weighted averaging (Sun, 2001). The damaged macroblock is split into four independent blocks; each pixel within a block is interpolated from two pixels in its two nearest boundaries. The weighted averaging method based on macroblock approach good results in cases where the missing macroblock lies within a smooth area.

2.5.3 Error Concealment in Temporal Domain

Temporal information is presented movement and speed characteristic.

- *Movement characteristics*

  It is easier to conceal linear movements in one direction because we can predict pictures from previous frames; the scene is almost the same. If we have movements in many directions or scene cuts, finding a part of the previous frame that is similar is going to be more difficult, or even impossible e.g. in case of scene cuts. In Figure 2.21, we can see an example of this. The figure represents a sequence of five frames, with a step of three frames between every one, of three different sequences: a football match in Figure 2.21(a), a village panorama in Figure 2.21(b) and a music video clip in Figure 2.21(c). In the music video sequence we have two scene cuts in the same amount of frames than the village sequence, where we have a smooth movement in one direction. Obviously, it will be easier to conceal the village sequence.

- *Speed characteristic*

The slower the movement of the camera, the easier will be to conceal an error. We can see an example of two different video speeds if we compare the village sequence as shown in Figure 2.21(b) with the football sequence in Figure 2.21(a)

Figure 2.21 Sequences with different movement and speed characteristics

This kind of error concealment seizes on temporal correlation of the video sequence to conceal the error. Motion estimation using previous frames is performed to reconstruct the missing data. In this section, two different error concealment techniques based on the temporal domain are presented.

2.5.3.1 Macroblock Copy

Macroblock copy is the simplest non-motion compensated temporal error concealment method. Here the missing blocks of one frame, F_n, are replaced by the spatially corresponding blocks of the previous frame, F_{n-1}

$$F_n(i, j) = F_{n-1}(i, j) \tag{2.12}$$

This method only performs well for a low motion sequence, but the advantages lie in its low complexity. Better performance is provided by the motion compensated interpolation methods.

2.5.3.2 Recovery of Inter Prediction Side Information (MV copy)

The H.264/AVC decoder needs the inter prediction side information and the DCT coefficients of the residuals. The Inter prediction side information includes the motion vectors and the corresponding reference frame number. The loss of motion vectors degrades the decoded image. This degradation propagates to the subsequent inter frames until an intra frame is decoded. The decoding of the n^{th} inter frame, x_n, is given by eq. (2.13):

$$x_n = x_{n-1}(i + V_x, j + V_y) + r_n(i, j) \tag{2.13}$$

Where x_{n-1} is the decoding of the $(n-1)^{th}$ inter frame, the (V_x, V_y) represent the x and y-component of the motion vector for the $(i, j)^{th}$ pixel and $r_n(i, j)$ denotes the residual value. Note that in opposition to luma and chroma values of a pixel, a motion vector is assigned to block at least 4 x 4 = 16 pixels. Therefore all pixels belonging to the same 4x4 block have the same motion vector.

As mentioned before, the H.264/AVC encoder applies the compression to the motion vector information by taking the difference between the current motion vector and the motion vector of an already encoded neighboring macroblock. The

information, to which differential value of the neighboring macroblock, is added to the other inter prediction side information. Therefore, the loss of a macroblock motion vector propagates the following macroblocks in the frame or in the slice, which depend on the motion vector prediction from the affected macroblock. By dealing with a video sequence containing the slow motion scenes, then the value of motion vector of the macroblock are close to zero.

For example, when a video bit stream is been received in the decoder after it is transposed in the wireless channel. During the reconstruction of video frames have occur loss motion vectors, which may have a different displacement in motion from the original position within the frame in all the following inter frames. The simplest way to recover the lost motion vectors of a damaged macroblock is to set its value to zero. The degree of visual artifacts that might be produced by this method depends on the maximal detected motion.

In some video scenes a homogenous movement of all objects within the video frame can be recognized, such a scene is created by a moving camera shot. The difference between the motion vectors of adjacent macroblocks is near zero. This difference value is extracted by the video compressor; a misinterpretation of this value on the receiver side means automatically a misinterpretation of the actual motion vector and leads to a global displacement of a group of macroblocks within the actual frame. Similar behavior can be recognized in case of homogenous movement of a group of macroblocks within a moving object in the video scene; all macroblocks belonging to this object have the same motion vector, a transmission failure of the motion vector belonging to the first decoded macroblock of this group could cause a local displacement of the object. In these two cases the simplest way to recover motion vectors is to use the motion vector, to which the corrupted motion vector is related to. This can be implemented by setting the differential motion vector value which has been affected by the bit-stream error to zero. With that the resulting motion vector value is the same as the reference motion vector value. This method can also be applied to the video frames of low motion video scenes.

2.6 Characteristic of Wireless Channel

Wireless channel measurements under different possible environments can be difficult and time consuming. However, a wireless channel can be reasonably modeled using a multipath Rayleigh fading model. The effects of the channel under different environments can then be easily studied through computer simulations by varying the appropriate parameters. An example of the wireless channel simulator that can be used for this purpose is described in (Chen, 1995). A schematic of the wireless channel simulator is outlined as shown in Figure 2.22.

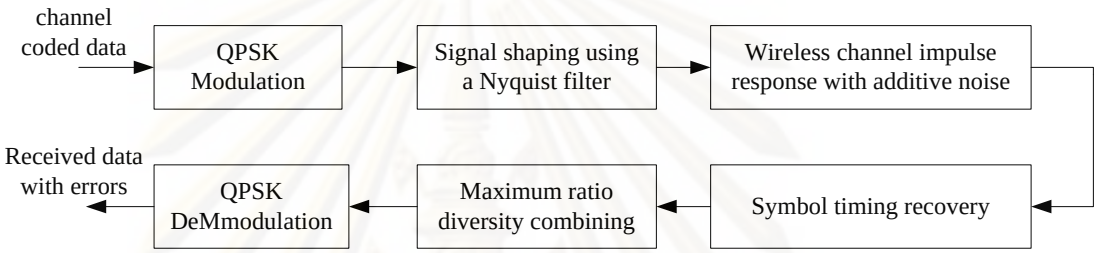


Figure 2.22 Schematic of the wireless channel simulator

The parameters that can be changed in the simulation are the maximum Doppler frequency f_D, the propagation power delay profile modeled as *n*-rays with different inter-path delay and power, signal power and antenna diversity. The data rate varies from 32 kbps to 256 kbps depending on the number of time slots allocated for the user. A coherent receiver, with optimal symbol timing recovery and perfect carrier recovery is assumed. An ideal maximal ratio combiner for antenna diversity combining is used.

The use of the wireless channel simulator will greatly simplify the characterization of the statistical nature of the wireless transmission channel. This research will make extensive use of the wireless channel simulator to model the transmission system.

2.7 Forward Error Correction and Interleaving

FEC coding is a common error control technique used to detect and correct bitstream errors due to the bursty nature of wireless channels. The concept of FEC is to insert redundancy bits to generate codeword such that the probability of correct

decoding could be increased. There are many types of FEC coding. For this dissertation, for simplicity, Bose, Chaudhury, Hocquengham (BCH) code is selected. In BCH code, the information to be transmitted is segmented into blocks of k symbols. To each block of k symbols, r symbols of redundancy are added to make it an n symbol code word and the code is referred to as (n,k) code. An BCH (n,k) code with $q = 2^m$ has the following properties:

$$n = 2^m - 1 \text{ symbols, } t < 2^m - 1 \text{ symbols,}$$

$$k = n - r \text{ symbols (symbol contain m bits)}$$

The code BCH(14,10) which is capable of correcting t or fewer symbol error in an n symbol code word with $r = 2t$ symbols. Note that the choice of FEC code and its corresponding error correction capability depends on the constraints and limitation of the system studied such as complexity, time delay, and channel bandwidth, etc.

Interleaving is used in digital video transmission technology to protect the transmission against burst errors. These errors overwrite a lot of bits in a row, so a typical error correction scheme that expects errors to be more uniformly distributed can be overwhelmed. Interleaving is used to help stop this from happening.

Video is often transmitted with error control bits that enable the receiver to correct a certain number of errors that occur during transmission. If a burst error occurs, too many errors can be made in one codeword, and that codeword cannot be correctly decoded. To reduce the effect of such burst errors, the bits of a number of codeword are interleaved before being transmitted. This way, a burst error affects only a correctable number of bits in each codeword, and the decoder can decode the codeword correctly. Interleaving is popular because it is a less complex and cheaper way to handle burst errors than directly increasing the power of the error correction scheme. In this dissertation, we will adapt BCH codes as the choice of FEC coding and combined with interleaving.

LITERATURE REVIEW

2.8 Literature Review about Flexible Macroblock Ordering

Flexible Macroblock Ordering (FMO) is the most striking new tool for error resilience within the H.264/AVC specification (Horowitz, 2002 , P. Lamber, 2006). It allows great flexibility to define the coding order of macroblocks within a picture. H.264/AVC specifies seven types of FMO. FMO types 0-5 are pre-defined FMO map and special cases where the coded representation of the MBAmap is much smaller due to certain patterns in the MBAmap. In (Sood, 2006), analyze the quality of H.264 video received. Using two slice groups with FMO, the box-out mapping shows a high value of PSNR. The significant observation here is the fact that the division of the frame into slices groups and assigning the macroblocks these specific groups significantly improves the error resilience and results in better PSNR.

FMO type 6 is called explicit FMO map and is the most general type where the entire MBAmap is actually coded into a picture parameter set. Some recent works in explicit FMO studied different methods on how to quantify the importance of a MB. In (katz, 2006), combining the Chessboard-like pattern and the Spiral into one superior FMO type, named ESI. ESI scatters the error over a wider area within the frame more than any other FMO type. ESI effectively enhances the perceptual quality when the sequence is characterized by one important region located at the center of the frame, making it an attractive candidate for video conferencing and video tracking applications. But this method does not relate information of video sequence.

There are two types of indicators that can be used to spatial and temporal indicators. For spatial information, (S.K. Im, 2007) is a new mapping algorithm for categorization of frame’s macroblock into two or more classes. The classification metric considers the impact of the macroblock data on its own pixels and also the improvement that it makes to the error concealment of its adjacent macroblocks. The other method is popular as call Region of Interest (ROI), (Luo, 2008) proposed a new hierarchical FMO type the ROI region can be represented in one or more rectangles, and each ROI rectangle can be further divided into several dispersed sub slice groups. Meanwhile, the background of the frame is kept in only one slice group. And (Sivanantharasa, 2006) proposed to use the FMO Type 6 to code the relatively important areas of a video frame (foreground) into one NAL unit, and the less

important areas (background) into another NAL unit. One of these NAL units carries the background, and the other contains macroblocks from the foreground. In (Dhondt, 2006), a MB impact factor is computed which depends on some information derived from the used pixels. In (Hantanong, 2005), using number of macroblock coded bit-count has been investigated as an indicator for a choice of FMO map of each frame. And in (Aramvith, 2006), the work in (Hantanong, 2005) is combined with FEC and showed the effectiveness in further reducing the number of undecodable macroblock.

For temporal information, (Thomos, 2005) a distortion measure based on the mean square error of the original and reconstructed pixels are used. In (Im, 2005) a MB importance factor is computed based on two distortion measures, a distortion of the coded MB and a distortion if the MB is lost and concealed and the number of bits of a particular MB.

2.9 Literature Review about Error Concealment

The choice of error concealment techniques used will contribute to the improvement of the received video quality at certain extent. Error concealment can be classified into three categories: spatial, temporal and hybrid (Tsekeridou, 2000). For spatial error concealment (Wang, 1998), the information from surrounding correctly received or concealed blocks are used for reconstructing the damaged area. In temporal error concealment (Zhang, 2000), the information of the related blocks from the blocks in the previous frame is used to conceal lost blocks. Hybrid error concealment employs both spatial and temporal information for error concealment. The H.264/AVC reference software uses frame copy and motion vector copy to recover from whole frame loss (ITU-T Recommentation H.264, 2005) and temporal error concealment based on boundary matching algorithms (BMA) (Chen, 2003) to improve the video quality. In (Xu, 2004), an adaptive spatial and temporal estimation method with low complexity is presented. The temporal concealment involves a method of subblock-based refined motion compensated concealment using weighting boundary matching and the spatial concealment scheme that involves an algorithm with refined direction weighted spatial interpolation. In (Suh, 2002), proposed the motion vector recovery method using optical flow in MPEG2. Many parameters have been studied for error concealment such as temporal error concealment based on

optical flow in (Kim, 2006) and using Lagrange interpolation in (Zheng, 2004). The work in (Kim, 2006) proposed to recover the motion vector of a 4x4 block included in a lost macroblock using the weighted average of obtained flow velocities. In (Zheng, 2004), Lagrange interpolation formula is used to compute a polynomial function that is used to recover the lost motion vector. A recursive Optimal Per-pixel Estimate algorithm is estimated by end-to-end distortion at a given packet error rate (PER) for error concealment (Lie, 2007).

2.10 Literature Review Combined FMO and Error Concealment

Some previous paper combined FMO and error concealment for improving both quality and quantity of video sequence. In (Jung, 2004), a method using FMO mode 2 or Interleaved and selective temporal error concealment depending on whether a lost MB is a background or foreground is used. In (Nasiopoulos, 2005), a technique that combines FMO mode 1 or dispersed and error concealment algorithm for intra-coded frame is used.

CHAPTER III

ERROR-RESILIENT VIDEO CODING USING FLEXIBLE MACROBLOCK ORDERING AND ERROR CONCEALMENT METHODS

3.1 Generating Explicit FMO map

3.1.1 Two-Pass Explicit FMO Map Algorithm

Wireless video transmission system is characterized by the burst error nature caused by the time varying channel. FMO can possibly alleviate the effect of data losses by managing the spatial relationship between regions that are coded in each slice and also serves as a macroblock level interleaving tool that can spread out consecutive burst error. From previous work (Hantanong, 2005), bit-count is used as the spatial information to indicate the importance of the macroblock in a video frame. To generate macroblock-to-slice group maps, two-pass encoding is used. In the 1st pass encoding, bit count information is collected to generate the corresponding map by sorting the bit count of each macroblock to the different slice groups. In the 2nd pass encoding, the explicit FMO map generated from the 1st pass encoding is used in the video, as shown in Figure 3.1.

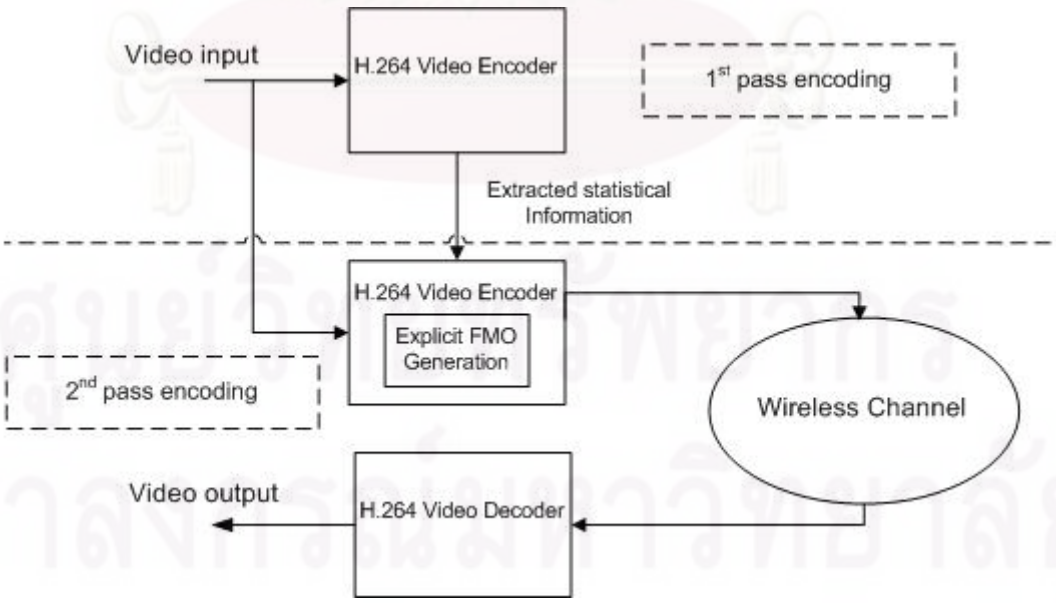


Figure 3.1 Block diagram of two-pass explicit FMO

In section 3.1.1.1, we investigated the use of temporal information for macroblock importance. Distortion measure obtained from concealment error by assuming a loss of each macroblock and using non motion-compensated error concealment in the 1st pass encoding. In section 3.1.1.2, spatial and temporal information are both considered for macroblock importance. By assuming a loss of each macroblock could contribute in quality loss and distortion in terms of using spatial and temporal error concealment.

3.1.1.1 Distortion Measure

We propose and analyze another indicator, called distortion measure from concealment error, as the temporal information in video frames to indicate macroblock importance.

Given a packet corresponded to area of pixels in macroblocks is error, the undecodable macroblock at the decoder will then be concealed using non-motion-compensated error concealment method. The distortion due to the error concealment, D_{CE}, based on the sum of absolute difference (SAD) and can be computed at the encoder during the first pass encoding, as shown in eq.(3.1),

$$D_{CE} = \sum_{(x,y \in L)} \left| f_k(x,y) - f_{k-1}(x,y) \right| \tag{3.1}$$

where frame k and k-1 are the current frame and previous frame, respectively, $f(x,y)$ is the reconstructed pixel value at the coordinate (x,y) and L is the damaged area due to the error packet.

The computed distortion measure per macroblock is then sorted in descending order. The macroblocks with high distortion measures are assigned to different slice groups, in order to interleave the seemingly important macroblock into different slice groups.

The steps of our proposed technique are outlined as follows.

1. Calculate D_{CE} for the corresponding macroblocks using eq.(3.1).
2. Sort the macroblock address of each frame based on the computed distortion measure per MB and assigned slice group id 0 to 7 to the sorted list to assign important macroblock to different slice groups.

3.1.1.2 Distortion by Simulating Spatial and Temporal Error Concealment

We propose to use the explicit map type of FMO and compute distortion measure from spatial and temporal error concealment simulation at the encoder to determine the importance of a particular macroblock. The important macroblocks will then be interleaved to different slice groups to improve error resiliency. To generate MB-to-slice group map, a two pass encoding scheme is used. During the first pass the necessary information of the macroblocks are collected such as the motion vector and the residual mean absolute difference (MAD). This information is used to generate the appropriate MB-to-slice group map. In the second pass, the video is encoded with FMO enabled using the generated explicit FMO map.

From previous work in (Hantanong, 2005), the number of MB coded bit count is used as a spatial indicator of MB importance. In section 3.1.1.1, a distortion measure due to the error concealment is used as a temporal indicator of MB importance. In this method, we propose another indicator, a residual information, derived from the mean absolute difference (MAD) of the macroblock due to motion estimation and the potential distortion measure of a given error concealment technique.

The residual information is computed after the first pass of the encoding sequence. First, we assume that each MB in the frame will be lost and concealed by either spatial or temporal error concealment. If the MB is lost and concealed then a certain amount of distortion is incurred, measure as *Dis* using eq.(3.2). We compute two types of distortion, one is if spatial error concealment (weight interpolation) is used, and the other if a temporal error concealment (motion vector copy) is used. After computing the distortion due to error concealment, we then compare the distortions with the MAD of the MB, this is the residual information, *res*, computed using eq.(3.3). If the resulting residual information is less when weight interpolation is used, then it means the MB can be better concealed by using spatial error concealment.

For each frame, the distortion of using spatial or temporal error concealment of each MB is compared with the MAD and the resulting residual information is computed as in eq.(3.2) and eq.(3.3).

$$Dis = \sum_{(x,y)} \left| \tilde{f}_k(x,y) - f_k(x,y) \right| \tag{3.2}$$

$$res = \sum_{(m,n \in L)} \left| Dis_k(m,n) - MAD_k(m,n) \right| \tag{3.3}$$

Where frame k is the current frame, $\tilde{f}_k(x,y)$ is the concealed pixel value at coordinate (x,y), $f_k(x,y)$ is original video sequence, $Dis(m,n)$ is the distortion if a MB is lost and concealed at coordinate (m,n), res is a residual information and L is the damaged area due to packet loss.

For each frame, the number of MB that can be better concealed (lower residual information) using spatial or temporal error concealment is determined. If the majority of the MB for that frame can be better concealed using spatial error concealment, then the residual information for that frame is computed assuming spatial error concealment will be used for the entire frame. The same is also done if the majority of MBs in the frame can be better concealed using temporal error concealment.

The residual information, *res*, is then used to sort the macroblocks in descending order. The macroblocks are then assigned to different slice groups beginning with the macroblock with the highest residual information and following the order or the sorted list.

The steps of generate FMO map.

1. Simulate spatial error concealment (SEC), compute $\tilde{f}_k(x,y)$ and calculate distortion of spatial error concealment, Dis_{SEC}, by eq.(3.2)

2. Simulate temporal error concealment (TEC), compute $\tilde{f}_k(x,y)$ and calculate distortion of temporal error concealment, Dis_{TEC}, by eq.(3.2)

3. Compare Dis_{SEC} and Dis_{TEC} for each MBs. Count the number of MBs with $Dis_{SEC} < Dis_{TEC}$, denoted as $MBcnt_{SEC}$. Also count the number of MBs with $Dis_{TEC} < Dis_{SEC}$ and denote as $MBcnt_{TEC}$

4. Generate a new explicit FMO map based on $MBcnt_{SEC}$ and $MBcnt_{TEC}$ of each frame:

IF

$$MBcnt_{SEC} > MBcnt_{TEC}, \text{ use Spatial-FMO}$$

and use Dis_{SEC} in eq.(3.3) to compute res

ELSE

$$MBcnt_{SEC} < MBcnt_{TEC}, \text{ use Temporal-FMO}$$

and use Dis_{TEC} in eq.(3.3) to compute res.

5. Sort the MB address for each frame based on the residual information, res, and assign slice group id 0 to 7 to the sorted list to assign important MB to different slice groups.

3.1.2 One-Pass Explicit FMO map Algorithm

We analyze a method to generate one-pass explicit FMO map based on spatial and temporal information as shown diagram of one pass explicit FMO map in Figure 3.2.

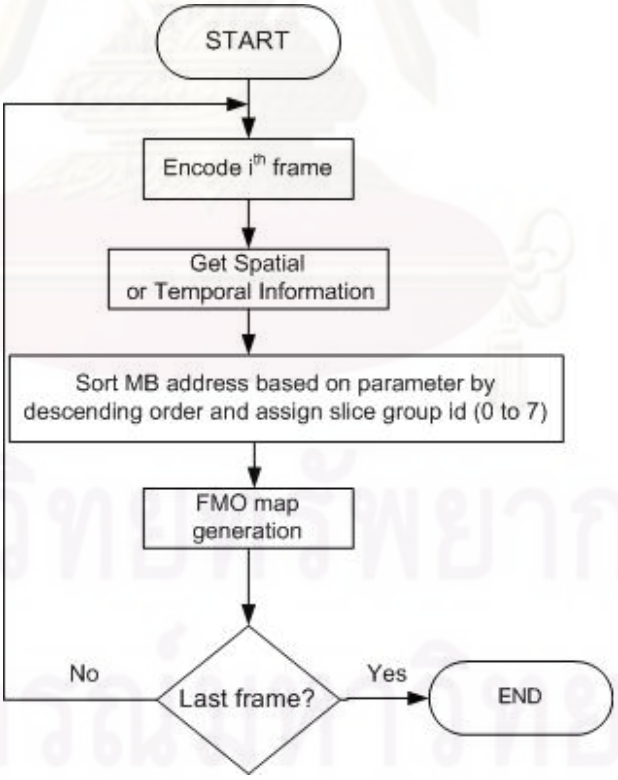


Figure 3.2 Diagram of one-pass explicit FMO map

For the spatial information, we use the encoder's macroblock coded-bit-count information as an indicator for a choice of MBAmap of each picture (in section 3.1.2.1). For the temporal information we compute a distortion measure based on a non-motion compensated error concealment (in section 3.1.2.2). Furthermore, we propose a method to generate one-pass explicit FMO map based on distortion by simulating spatial and temporal error concealment (in section 3.1.2.3).

3.1.2.1 Bit-Count Information

We propose to use one-pass explicit FMO mapping using bit-count information of each macroblock collected during the encoding stage as an indicator of macroblock importance. By using one-pass explicit mapping, the macroblock-to-slice-group mapping will be set differently for each picture. Macroblock bit-count information at encoder of the previous frame is collected to generate explicit map of the current frame. The basic idea of our approach is that the macroblocks that use a higher number of bits are the more important macroblock due to the properties of motion-compensated prediction. Therefore, we try to interleave consecutively the macroblock with high bit-count to be in a different slice. The steps to generate one-pass explicit FMO map using bit count information is as in Figure 3.2.

3.1.2.2 Distortion Measure

We propose and analyze another indicator, called a distortion measure from concealment error, as the temporal information in video frames to indicate macroblock importance. To generate MBAmap, one-pass encoding is used. Pixel values of the previous frame are collected to generate explicit FMO map each frame.

Given a packet corresponding to an area of pixels in macroblocks that have an error, the undecodable macroblock at the decoder will then be concealed using non-motion-compensated error concealment method. The distortion due to the error concealment, D_{CE}, based on the sum of absolute difference (SAD) and can be computed, as shown in eq. (3.4),

$$D_{CE} = \sum_{(x,y \in L)} \left| f_{k-1}(x,y) - f_{k-2}(x,y) \right| \qquad (3.4)$$

where a frame k-1th and k-2th are the previous frame and the second previous frame, respectively, $f_{k-1}(x,y)$ is the reconstructed pixel value at the coordinate (x,y) of a

previous frame, $f_{k-2}(x, y)$ is the reconstructed pixel value at the coordinate (*x*,*y*) of a second previous frame and *L* is the damaged area due to the error packet.

The computed distortion measure per macroblock is then sorted in descending order. The macroblocks with high distortion measures are assigned to different slice groups, in order to interleave the seemingly important macroblock into different slice groups. The step is the same as in Figure 3.4 but change the parameter to the distortion due to the error concealment.

3.1.2.3 Distortion by Simulating Spatial and Temporal Error Concealment

This method proposes to use the one-pass explicit FMO mapping by computing a distortion measure from spatial and temporal error concealment at the encoder to determine the importance of a particular macroblock. The important macroblocks will then be interleaved to different slice groups to improve error resiliency. To generate a MB-to-slice group map, a one pass encoding scheme is used. The feedback information is sent to a current frame such as the motion vector and the residual mean absolute difference (MAD). This information is used to generate the appropriate MB-to-slice group map. In this work, we propose another indicator, a residual information, derived from the mean absolute difference (MAD) of the macroblock due to motion estimation and the potential distortion measure of a given error concealment technique.

The residual information is gathered from the feedback information. First, we assume that each MB in frame will be lost and concealed by either spatial or temporal error concealment. If the MB is lost and concealed then a certain amount of distortion is incurred, measure as *Dis* using eq.(3.5). We compute two types of distortion, one is if spatial error concealment (weight interpolation) is used, and the other if a temporal error concealment (motion vector copy) is used. After computing the distortion due to error concealment, we then compare the distortions with the MAD of the MB, this is the residual information, *res*, computed using eq.(3.6). If the resulting residual information is less when weight interpolation is used, then it means the MB can be better concealed by using spatial error concealment. For each frame, the distortion of using spatial or temporal error concealment of each MB is compared with the MAD and the resulting residual information is computed as in (3.5) and (3.6).

$$Dis = \sum_{(x,y)} \left| \tilde{f}_{k-1}(x,y) - f_{k-1}(x,y) \right| \tag{3.5}$$

$$res = \sum_{(m,n \in L)} \left| Dis_{k-1}(m,n) - MAD_{k-1}(m,n) \right| \tag{3.6}$$

Where a frame k-1 is the previous frame, $\tilde{f}_{k-1}(x,y)$ is the concealed pixel value at coordinate (x,y) of a previous frame, $f_{k-1}(x,y)$ is original video sequence at encoder of a previous frame, $Dis_{k-1}(m,n)$ is the distortion if a MB is lost and concealed at coordinate (m,n) of a previous frame, $MAD_{k-1}(m,n)$ is Mean Absolute Different of a previous frame at coordinate (m,n), res is a residual information and L is the damaged area due to packet loss.

For each frame, the number of MB that can be better concealed, i.e, lower residual information, using spatial or temporal error concealment is determined. If the majority of the MB for that frame can be better concealed using spatial error concealment, then the residual information for that frame is computed assuming spatial error concealment will be used for the entire frame. The same is also done if the majority of MBs in the frame can be better concealed using temporal error concealment.

The residual information, res, is then used to sort the macroblocks in descending order. The macroblocks are then assigned to different slice groups beginning with the macroblock with the highest residual information and following the order or the sorted list. The step of one-pass FMO map is shown in Figure 3.4 However, the parameter is changed following certain conditions.

The steps of generation explicit FMO map from error concealment are as follow.

1. Simulate spatial error concealment (SEC), compute $\tilde{f}_{k-1}(x,y)$ and calculate distortion of spatial error concealment, Dis_{SEC}, by eq.(3.5)
2. Simulate temporal error concealment (TEC), compute $\tilde{f}_{k-1}(x,y)$ and calculate distortion of temporal error concealment, Dis_{TEC}, by eq.(3.5)

3. Compare Dis_{SEC} and Dis_{TEC} for each MBs. Count the number of MBs with $Dis_{SEC} < Dis_{TEC}$, denoted as $MBcnt_{SEC}$. Also count the number of MBs with $Dis_{TEC} < Dis_{SEC}$ and denote as $MBcnt_{TEC}$

4. Generate a new explicit FMO map based on $MBcnt_{SEC}$ and $MBcnt_{TEC}$ of each frame

   IF

   $MBcnt_{SEC} > MBcnt_{TEC}$, use Spatial-FMO

   and use Dis_{SEC} in eq.(3.6) to compute res.

   ELSE

   $MBcnt_{SEC} < MBcnt_{TEC}$, use Temporal-FMO

   and use Dis_{TEC} in eq.(3.6) to compute res.

5. Sort the MB address for each frame based on the residual information, res, from feedback information and assign slice group id 0 to 7 to the sorted list to assign important MB to different slice groups.

3.2 Framework of One-Pass Explicit FMO map and Error Concealment Algorithm

The error concealment method at the decoder is applied according to the residual information derived from the distortion calculated at the encoder that generated the FMO map. A block diagram in Figure 3.3 shows the relationship between encoder and decoder. We generate the MBAmap at the encoder by using feedback information from previous frame in the decoder. The feedback information consists of reconstruction data, motion vector (MV) data and bit-count data at the decoder. At the encoder, the residual information from error concealment that is used to generate the MBAmap is sent to the decoder to compute the value of the reconstruction pixel of the error concealment method. The proposed method is being evaluated in 2 scenarios: assuming perfect feedback channel and feedback channel having random errors (5 % and 10%).

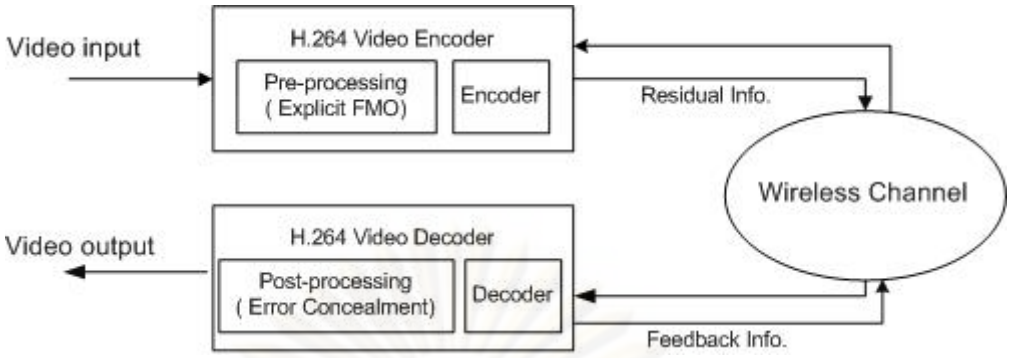


Figure 3.3 Block Diagram of One-Pass Explicit FMO

For error concealment, we propose an error concealment scheme to recover from MB loss and modify an algorithm by using residual information, res, in eq.(3.9) and calculate pixel loss from eq.(3.10).

$$\tilde{f}_k(x,y) = p \cdot \left(res^i_{k-1} + \hat{f}_k(x,y) \right) + (1-p) \cdot \tilde{f}_{k-1}(x,y) \tag{3.10}$$

When p is packet error rate, $\hat{f}_k(x,y)$ denote its encoder reconstruction at the coordinate (x,y), $\tilde{f}_{k-1}(x,y)$ is the decoder reconstructed pixel from the previous frame at the coordinate (x,y), and res^i_{k-1} is residual information from generate explicit FMO mapping.

3.3 Joint Explicit FMO map, Adaptive Interleaving depth and Forward Error Correction Algorithm

Using FMO as the source coding layer of protection is often not enough to protect the video packets especially when the wireless channel is experiencing deep fades resulting in burst errors extending to several packets that can affect several video frames. Using FEC and interleaving on the other hand can provide better error protection in the presence of burst errors but at the expense of reduced effective channel throughput that can result in degraded video quality. But applying FEC and interleaving for all frames is not very efficient since the condition of the channel is not always bad, the bits allocated for the FEC is wasted when the channel is good. Since the overhead incurred by using FMO is relatively small and to be able to make more efficient use of bits allocated for FEC, we add FEC and interleaving to the FMO only when the predicted conditions of the channel is bad.

The block diagram of the video transmission system used is shown in Figure 3.4. The H.264/AVC video encoder is configured with FMO enabled for explicit map types and we used the bit-count information (Hantanong, 2005; Aramvith, 2006) to determine the MB-to-slice-group mapping of every frame. Prior to the transmission of the packetized H.264 encoded frames, a decision is made by the interleaver control block to add FEC and interleaving as additional channel protection on top of the protection provided by FMO. The decision to add FEC and interleaving depends on the predicted condition of the wireless channel to be good or bad. A low-bandwidth error free dedicated feedback channel is assumed in our transmission system to provide feedback information from the receiver. Based on the received information from the feedback channel and the assumed channel model, we compute the probability that the channel will be in the good state. The FEC and interleaving layer of protection will be added to the FMO during the transmission of the current frame if the predicted condition of the channel in the previous frames is bad. Otherwise only FMO will be used to protect the frame.

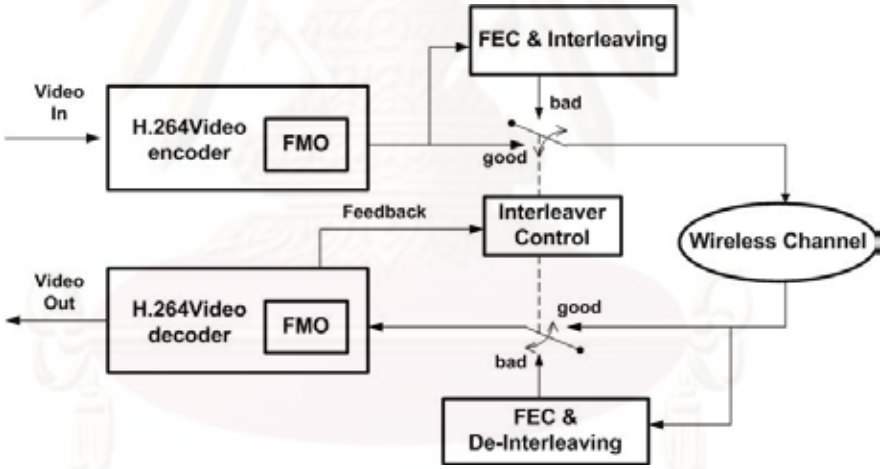


Figure 3.4 Block diagram of the interleaving depth system

For the FMO, we used the explicit map type using the bit-count as the spatial indicator of MB importance (Hantanong, 2005; Aramvith, 2006). The MB bit-count information is collected at the first pass encoding of a two-pass encoding process. The MBs are sorted in descending order according to the number of bits used to code the MBs. The sorted sequence of MBs is used to interleave the MB with high bit-count to different slice group maps, and this determines the MB-to-slice group mapping that used for FMO in every frame.

The design of the FEC and Interleaver differs from previous works (Hantanong, 2005; Aramvith, 2006) because here we vary the interleaving depth dynamically. Using Interleaving is not always helpful as suggested in (Aramvith, 2006). If the FEC has the capability to correct, *t*, symbol errors and the degree of interleaving is of depth, *d*, then Interleaving is useful if *td* is larger than the mean burst length, otherwise interleaving becomes harmful leading to uncorrectable error patterns in multiple codewords. In this work, for simplicity, we used the Bose, Chaudhury, Hocquengham (BCH) code, with $n = 14$ and $k = 10$. The BCH(14,10) code is capable of correcting *t* or fewer symbol errors in an *n* symbol codeword with a redundancy of *2t* symbols, the value $t = 1$ is used in our simulations. In order to adapt the interleaving depth to the burst error nature of the wireless channel, we vary the interleaving depth to be equal to the mean burst length, as suggested in (Zorzi, 1997).

The parameters of the wireless channel model as well as the interleaving depth depend on the observed mean burst error length in packets and the packet error rate (PER). We update the computation of the mean burst length and PER every frame based on the feedback information received from the previous frames. In order to determine whether a packet error is treated as a burst or as a single packet error we defined a *guard space* as the maximum distance between two packet errors. If the distance between two packet errors is less than the *guard space*, then packet errors are grouped as a burst, otherwise it is considered as a single packet error. In our simulation we used a *guard space* equal to 10 packets to compute the mean burst length.

The parameters of the wireless channel model is computed every frame, then we compute the probability that the channel will be in the good state given an initial state following the framework in (Howard, 1971). For this method, we use a two-state Markov model that is a simplified Gilbert-Elliot channel model (Gilbert, 1960) at the packet level; this model has been shown to be sufficient in modeling the burst nature of packet errors (Zorzi, 1997). The model has two states, a good state (G) and a bad state (B) as shown in Figure 3.5. If the channel is in the good state, it is assumed that a packet can be transmitted successfully, and when the channel is in the bad state, a transmitted packet will experience some error.

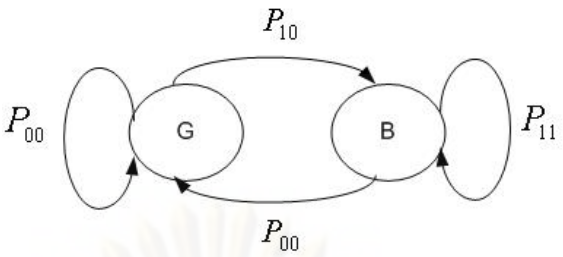


Figure 3.5 Two-state markov model

The channel state-transition probability matrix for this two-state Markov model can be set up as in eq.(3.7), where $P_{00}, P_{01}, P_{10}, P_{11}$ are the state transition probabilities.

$$P = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 1-P_{01} & P_{01} \\ P_{10} & 1-P_{10} \end{bmatrix} \tag{3.7}$$

From (Ronald A. Howard, 1971) the state transition probabilities, P_{01} and P_{10}, can be derived from the burst statistics of the channel as in eq.(3.8) and eq.(3.9). And the assumption is that the channel follows a Markov process having geometric distributions of run lengths error-bursts with mean $1/P_{10}$ (Gilbert, 1960).

$$P_{10} = \frac{1}{M} \tag{3.8}$$

$$P_{01} = \frac{P_{10} * PER}{(1-PER)} \tag{3.9}$$

Where M is mean burst length and PER is the packet error rate computed from the error pattern generated by the wireless channel simulator. The Markov model will be used as a tool to predict future channel behaviors.

We predict that the channel is good if the computed probability of being in the good state $P_g(i)$ in the current frame is greater than the probability of being in the good state in the previous frame $P_g(i-1)$.

Steps in predicting good and bad channel conditions

1. Calculate the mean burst lengths and packet error rate of the previous frames from feedback information.

2. Compute the state transition matrix and calculate the probability of that the next state of the channel is good (P_g), with the initial condition that the channel is in the good state.
3. Compute the P_g and $P_g(i-1)$

   If $P_g(i) \geq P_g(i-1)$, channel is good.

   - Use FMO for frame *i*

   Else $P_g(i) < P_g(i-1)$, channel is bad

   - FMO, FEC and interleaving for frame *i*.
   - Set *Interleaving depth* = mean burst length
4. Repeat steps (1) until (3) all frames have been processed

CHAPTER IV

RESULTS AND DISCUSSIONS

4.1 Simulation Setup

To analyze the effect of FMO for wireless video transmission, the modification of a reference software and wireless channel simulator are needed. In the simulation, the H.264/AVC JM codec version 9.2 references software is used. No modification of the encoded bit-stream is needed. However, the encoder in this version does not fully support encoding of explicit slice-group-map (i.e. slice-group-map-type = 6). We modify the source code such that it can read the entire slice group configuration file, and it can resend the updated picture parameter set (PPS) information for each encoded picture. Note that, the encoded bit-stream can be decoded by the decoder. However, some modifications are needed at the decoder to discard undecodable macroblock and to continue the decoding process. We encode video sequences using the baseline profile at level 3.0 in our simulation. Each sequence is encoded for a total of 100 frames with frame rate of 10 frames per second. The rate-control is enabled at fixed bit rate of 32 kbps. The default encoder parameters (S. Wenge, 2002) are used with the exception on the following FMO related parameters.

Slice_group_map_type (6)

Num_slice_group_minus_one (0 to7)

SliceGroupConfigFileName (proposed MBAmap).

Detailed description of H.264/AVC JM codec parameter usage can be found in (Sühring) and (Michael, 2004).

To investigate the benefits of using FMO on wireless channel, the Rayleigh fading wireless channel simulator is used in our simulation. The detail of the simulator can be found in (Chen, 1995) as shown in Table 4.1. To simulate the effects of slow and fast fading channel, the maximum Doppler frequency parameter is set to 1 Hz and 40 Hz for slow and fast fading, respectively. We set up the simulation by assuming that errors may not attack the PPS header. The average bit error rate (BER)

and average packet error rate (PER) for 80 bit packet, are 0.06 and 0.09, respectively (Chen, 1995).

Table 4.1 Wireless channel parameter

Multiple Access	TDMA
Modulation	QPSK
Channel Rate	32 kbps
Maximum Doppler Frequency	1 Hz (slow), 40 Hz (fast)
Transmitted Signal Power	15 dB
Time Delay Spread	¼ of symbol period
Power Delay Profile	2-ray with equal power
Antenna Diversity	1

4.2 Quality Metrics

This research shows the results of using two-pass, one-pass explicit FMO and error concealment. Due to the complex nature of error propagation in video transmission which can be both spatial and temporal direction, we decide to use both subjective and objective measurement of video quality to evaluate the performance of our proposed technique. The objective measure is the amount of undecodable macroblocks and Peak-Signal to Noise Ratio (PSNR), which depends on both spatial and temporal error propagation and can vary on different error concealment method. Having a lower number of undecodable MB indicates improved visual quality, but it does not directly translate to a better average PSNR because not all MBs have the same effect on the PSNR. The subjective measure is visual video quality.

4.3 Experimental Results

4.3.1 Two-Pass Explicit FMO Map

4.3.1.1 Comparison of Bit-Count and Distortion Measure

We test 4 video sequences: akiyo, claire, foreman and carphone. Each sequence is encoded for a total of 100 frames with frame rate of 10 fps. We compare the results of using no FMO, using the previously proposed explicit FMO using bit-count information and using the recently proposed explicit FMO using distortion measure from concealment error.

Figure 4.1 and 4.2 shows the amount of undecodable macroblock for each mapping types for slow and fast fading channels respectively. On the average, explicit FMO mapping using distortion measure gives comparable performance, in some cases better performance, in terms of the reduction in the number of undecodable macroblock as that of our previously proposed explicit FMO using bit count information. The reduction on the number of undecodable macroblock is up to 63% compared to no FMO mapping. Figure 4.3, 4.4, 4.5 and 4.6 show the PSNR curve of Carphone sequence for slow and fast fading and Akiyo sequence for slow and fast fading, respectively. We observe that in the slow fading case, the PSNR is significantly increased in most parts of the sequence compared with no FMO mapping. The PSNR improvement is comparable when compared between explicit FMO using bit count information and distortion measure. In some cases such as the carphone sequence, explicit FMO using distortion measure yields much better PSNR. This is because the distortion measure generates slice-group mappings utilizing temporal information which is considered as an advantage for sequences with fast movement like carphone. For the fast fading case, the overall PSNR improvement is less significant than that of the slow fading case. Nevertheless, we can still observe the improvement and the PSNR improvement is comparable when compared between explicit FMO using bit count information and distortion measure. As already noted, PSNR curve depends heavily on the methods of error concealment used. By using this method, the minimum undecodable macroblock picture may not have the maximum value of PSNR because of the unequal importance of each macroblock. To gain higher PSNR performance, an unequal data protection or more complicated error concealment might be applied.

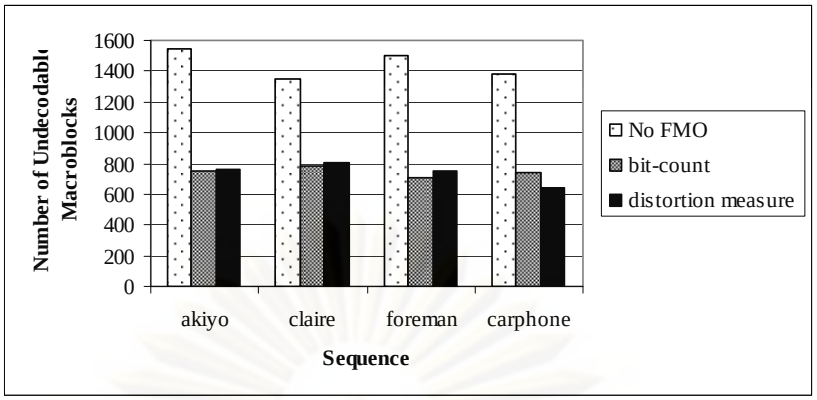


Figure 4.1 Total amount of undecodable macroblock for each map type using 8 slice groups (slow fading)

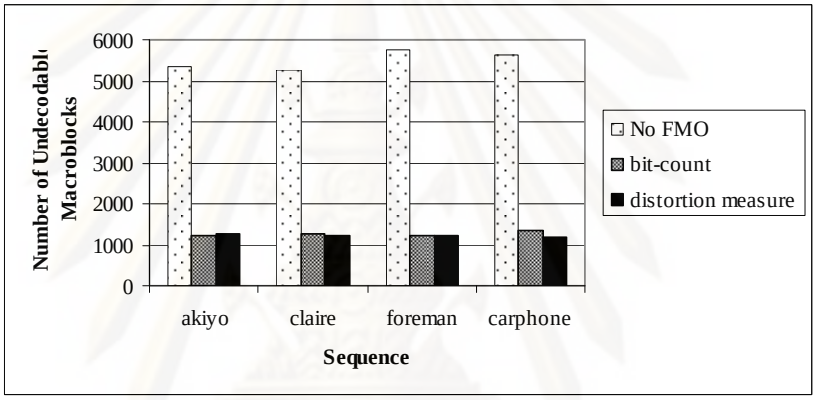


Figure 4.2 Total amount of undecodable macroblock for each map type using 8 slice groups (fast fading)

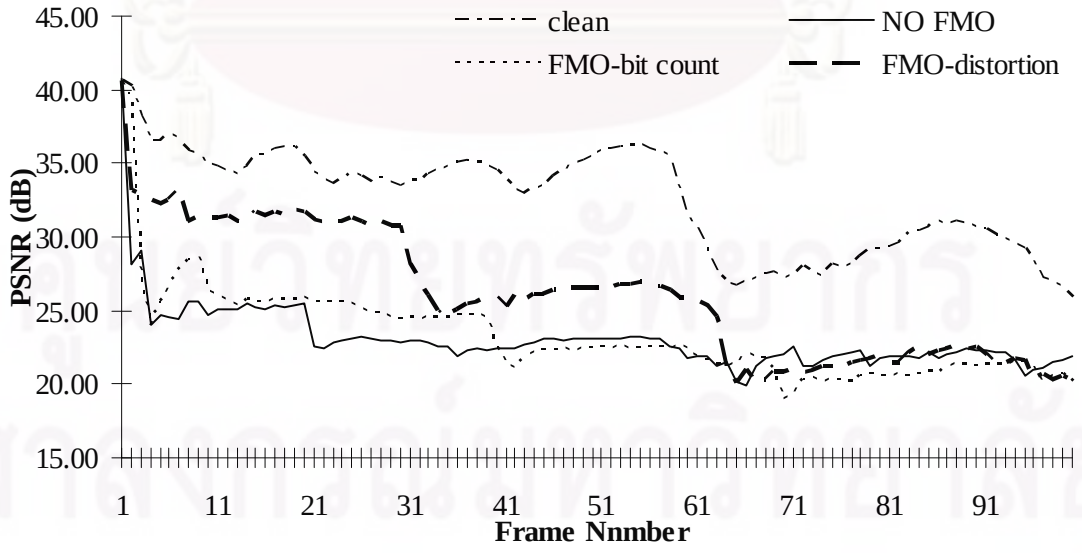


Figure 4.3 PSNR Comparison of carphone sequence for slow fading channel

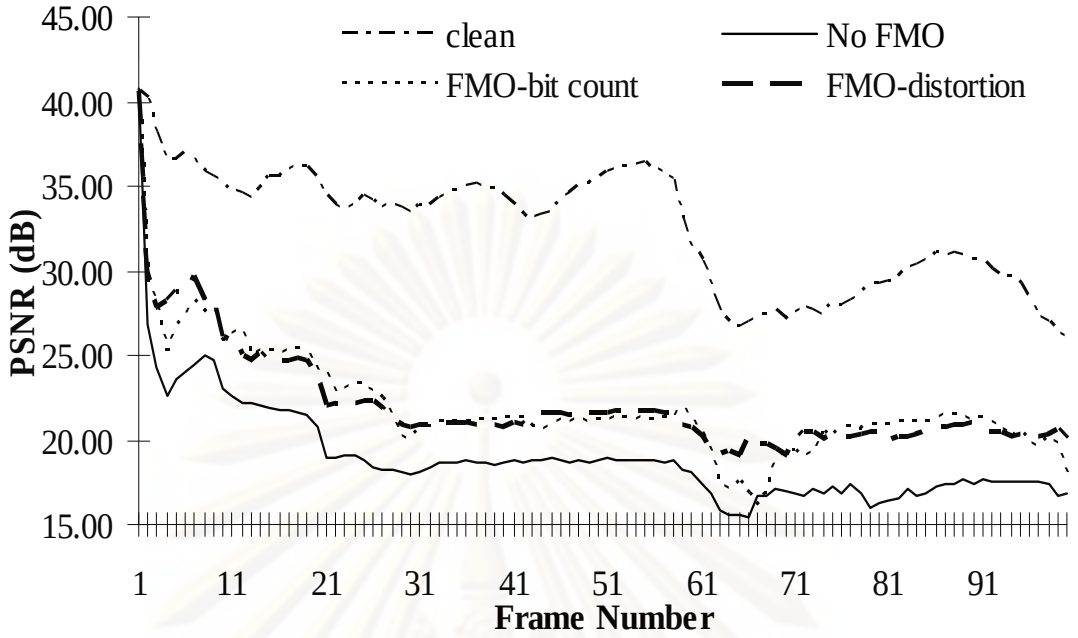


Figure 4.4 PSNR comparison of carphone sequence for fast fading channel

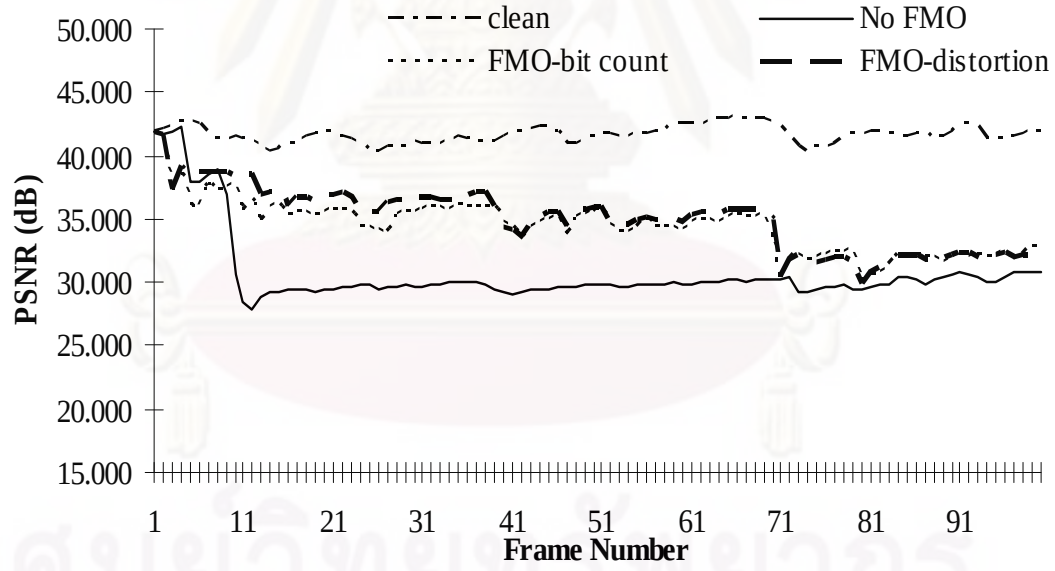


Figure 4.5 PSNR comparison of akiyo sequence for slow fading channel

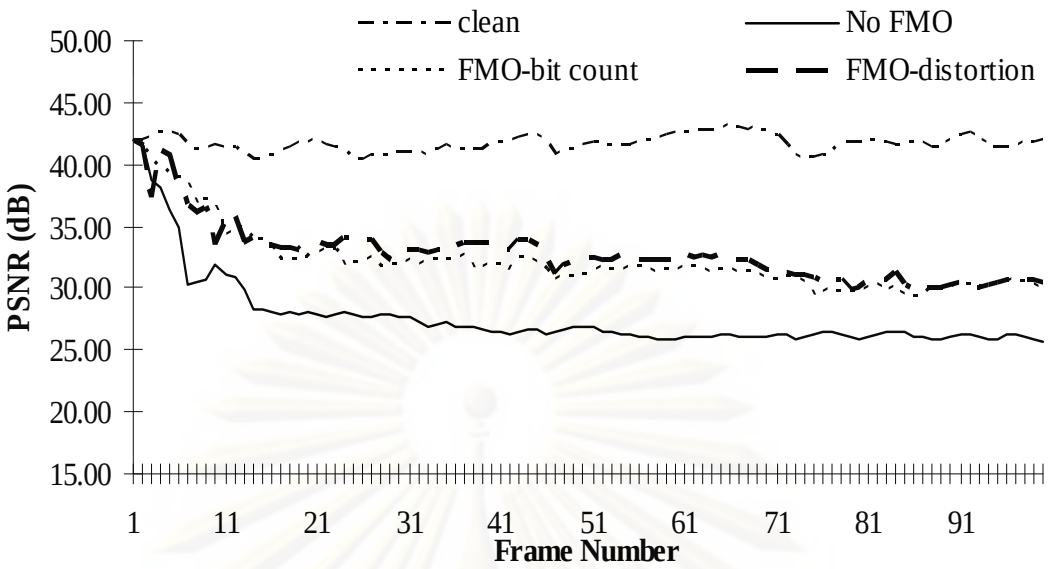


Figure 4.6 PSNR comparison of akiyo sequence for fast fading channel

4.3.1.2 Comparison of Bit-Count, Distortion Measure and Distortion from Simulated Error Concealment

We test 4 video sequences namely, claire, mobile, news and salesman. Each sequence is encoded for a total of 100 frames with frame rate of 10 fps. We compare the results of using no FMO, using the previously proposed explicit FMO using bit-count information and using the recently proposed explicit FMO using distortion measure from concealment error.

Table 4.2 summarizes the average PSNR of video in the scenario of slow and fast fading channels. For the slow fading case the results show that the PSNR improvement of the proposed method is up to 4.60 dB compared to No FMO, (Hantanong, 2005), and (Aramvith, 2006). For fast fading case, the proposed method can improve PSNR of up to 4.98 dB. As mentioned, the quality measurement of video in term of the PSNR depends heavily on the location of the bit errors and also the error concealment method applied. Nevertheless, there is no direct relation between the number of undecodable macroblock and the PSNR as several other factors have to be taken into account

Table 4.3 shows the number of undecodable macroblocks for slow and fast fading channel conditions. For slow fading case, the proposed method can reduce, on the average, the number of undecodable macroblocks of up to 77% compared to No FMO. While in the case of fast fading, the average number of undecodable

macroblocks could be reduced of up to 81%. Figure 4.7-4.8 shows the average PSNR curve of the “news” test sequence under slow and fast fading conditions. From the curves, it can be clearly observed the PSNR improvement of the proposed method. In some cases, the number of undecodable MBs with explicit FMO using bit count information is very close with the proposed technique for sequences with low motion content. If there is little motion, such as news, bit count information based on spatial information can better reduce the number of undecodable MB more than the proposed technique, but the difference is slight. Overall, using explicit FMO based on temporal information on the average can improve the PSNR. The number of undecodable MB are comparable. This is because of the unequal important property of each MB.

Table 4.2 Comparisons of average PSNR(dB)

	claire		**mobile**		**news**		**salesman**	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	30.78	24.28	15.30	12.65	24.00	20.56	25.89	24.59
FMO_bitcnt	31.12	**30.07**	**16.53**	14.99	26.73	23.31	29.15	**27.87**
FMO_Dist	30.66	29.59	16.50	14.91	25.77	23.39	29.73	27.12
FMO_SimDist	**31.61**	29.26	16.52	**15.21**	**27.78**	**24.47**	**30.50**	27.36

Table 4.3 Comparisons of number of undecodable MBs

	claire		**mobile**		**news**		**salesman**	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	1346	5269	1382	4168	1424	5754	1383	5011
FMO_bitcnt	780	1257	394	860	**651**	**1220**	563	**1115**
FMO_Dist	833	**1182**	470	933	692	1298	**547**	1164
FMO_SimDist	**760**	1257	**316**	**810**	658	1318	558	1124

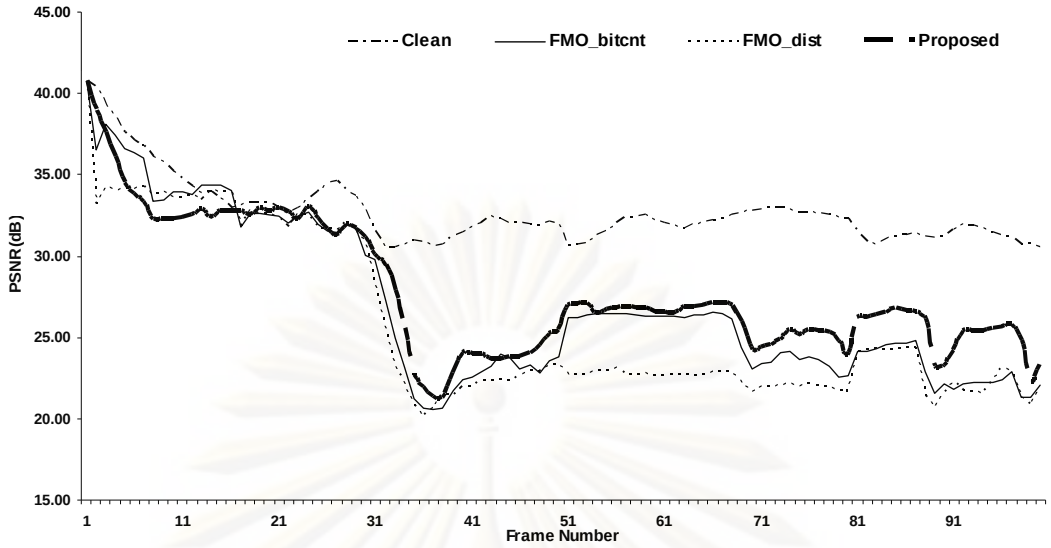


Figure 4.7 Comparison of PSNR among 4 methods with respect to clean channel for news sequence under slow fading channel

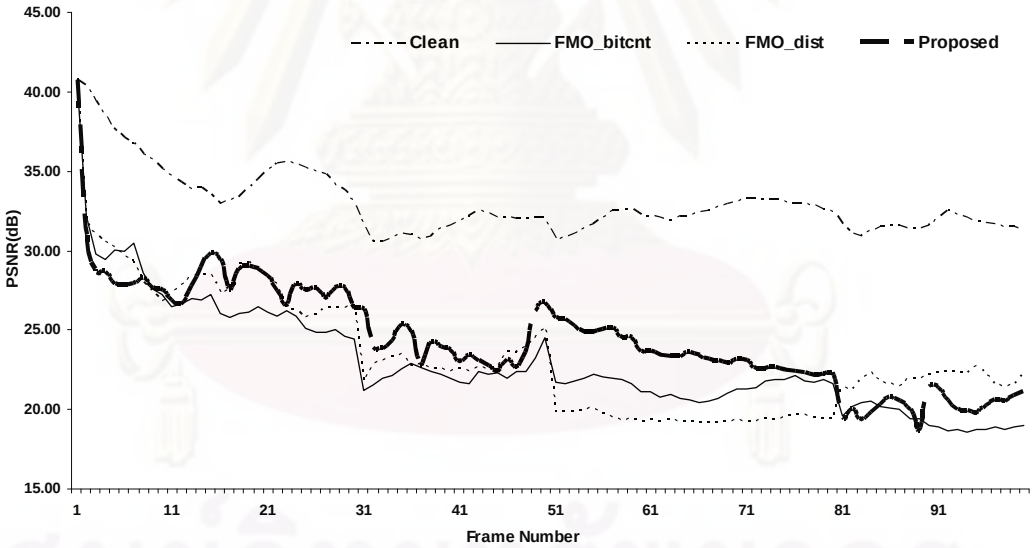


Figure 4.8 Comparison of PSNR among 4 methods with respect to clean channel for news sequence under fast fading channel

4.3.2 One-Pass Explicit FMO map and Error Concealment

We compare the improvement of our proposed one-pass FMO generation scheme and the proposed error concealment method with some previous works that use two-pass encoding and using a non-motion compensated error concealment scheme.

First we discuss the case where perfect feedback channel, i.e., no error in feedback channel, is assumed. Table 4.4 summarizes the average PSNR of the test videos in the scenario of slow and fast fading channels, our proposed scheme is labeled as 1P_bitCnt, 1P_Dist and 1P_SimDis+EC respectively. 1P_FMO_bitCnt is FMO map generation using spatial information, 1P_FMO_Dist is FMO map generation using temporal information and 1P_SimDis+EC is FMO map from error concealment measure combined with error concealment method.

For the slow fading case, the results show that the PSNR improvement of 1P_FMO_bitCnt, 1P_FMO_Dist ,and 1P_SimDis+EC are up to 4.47 dB, 4.30 dB and 5.68 dB in comparison to No FMO. For the fast fading case, 1P_FMO_bitCnt, 1P_FMO_Dist and 1P_SimDis+EC can improve PSNR of up to 6.23 dB, 5.97 dB and 6.09 dB, respectively. As previously mentioned, the qualitative measurement of video in terms of the PSNR depends on the location of the bit errors and also the error concealment method applied.

Table 4.5 shows the total number of undecodable macroblocks for slow and fast fading channel conditions. For the slow fading case, the average number of undecodable macroblocks of 1P_FMO_bitCnt, 1P_FMO_Dist and 1P_SimDis+EC is reduced of up to 65.99%, 67.66% and 66.79 % in comparison to No FMO. In the case of fast fading, the average number of undecodable macroblocks can be reduced of up to 80.58%, 79.11% and 80.54%, as shown in Table 4.5 Under the fast fading conditions, the wireless channel experiences frequent burst errors with small burst durations. Therefore, using spatial information is advantageous because spatial information considers only the current frame but not the previous frame and burst error. This is the cause of PSNR improvement and the reduction in the number of undecodable MBs. In the case of slow fading conditions, the wireless channel experiences less frequent burst errors with longer burst durations.

In case there are random errors in feedback channel, Table 4.6 shows a comparison of the summarizes the average PSNR of the test videos in the scenario of slow and fast fading channels, our proposed scheme is labeled as 1P_SimDis+EC, 1P_SimDis+EC(RandErr5%) and 1P_SimDis+EC(RandErr10%) respectively. 1P_SimDis+EC(RandErr5%) and 1P_SimDis+EC(RandErr10%) is FMO map from error concealment measure combined with error concealment method with feedback have random errors of 5% and 10%. For the slow fading case, the results show that the PSNR improvement of 1P_SimDis+EC, 1P_SimDis+EC(RandErr5%) and 1P_SimDis+EC(RandErr10%) are up to 5.68 dB 5 dB and 5.78 dB in comparison to No FMO. For the fast fading case, 1P_SimDis+EC, 1P_SimDis+EC(RandErr5%) and 1P_SimDis+EC(RandErr10%) can improve PSNR of up to 6.09 dB, 6.59 dB and6.79 dB, respectively.

Table 4.7 shows a comparison of the total number of undecodable macroblocks for slow and fast fading channel conditions. For the slow fading case, the average number of undecodable macroblocks of 1P_SimDis+EC, 1P_SimDis+EC(RandErr5%) and 1P_SimDis+EC(RandErr10%) is reduced of up to 66.79%, 69.18% and 66.50% in comparison to No FMO. In the case of fast fading, the average number of undecodable macroblocks can be reduced of up to 80.54%, 80.19% and 80.13%, as shown in Table 4.7.

The percentage of packet error affects only the FMO map generation process, this changes the FMO map that will be generated every frame. This change in FMO map will change the location of macroblocks that will be affected by error given the same packet error pattern. Considering the effects of error propagation; the effect of change in FMO will result in different number of undecodable macroblocks. In general, if the number of undecodable macroblocks lost increase. This will also increase the number of macroblocks that will be concealed depending on the error concealment strategy used. This could result in the changes of PSNR. The proposed error concealment can increase the PSNR by around 1 dB compare to MB copy.

Figure 4.9(a)-(h) shows the comparison of the decoded image quality of the proposed method with other FMO mappings at the 16^{th} frame of the carphone sequence during fast fading. Fig 4.9(h) shows that the one-pass explicit FMO mapping method is the closest to the original image and has the best visual quality compared with the other methods. Figure 4.10(a)-(h) is the decoded image quality of

the proposed method with other FMO mappings at the 16th frame of the carphone sequence during slow fading. The result is also the closest to the original and the best visual quality. For the 32nd frame of akiyo sequence, a comparison of the decoded image quality of the proposed method under fast fading are shown in Figure 4.11(a)-(h). Akiyo sequence has low motion content, therefore in Figure 4.11(h) the decoded image can be very close to the original image. For 32nd frame of Akiyo sequence under slow fading, as shown in Figure 4.12(a)-(h), 1P_SimDis+EC is the closest to the original image and has the best visual quality. In Figure 4.13 – 4.18 are the decoded quality of the proposed method with other FMO mappings at mother, hall and coastguard sequences. Figure 4.19-4.20 shows the average PSNR curve of the carphone test sequence under slow and fast fading conditions. From the curves, it can be clearly observed that the effect of PSNR improvement on the decoded image of proposed method.

In some cases, the number of undecodable MBs of 1P_FMO_bitCnt is very close to the 1P_FMO_Dist for the sequences with low motion content. If there is little motion, such as news or akiyo, bit count information based on spatial information is better as it can reduce the number of undecodable MB more than the one-pass explicit FMO, but the difference is slight. In case of 1P_SimDis+EC, which takes into consideration spatial and temporal error concealment schemes simulated at the encoder and combine with error concealment method at the decoder using residual information from the simulated one-pass explicit FMO mapping at the encoder side, the average PSNR of this method is better than previous method because this method use error concealment from a correlation parameter at generate FMO mapping.

Overall, using one-pass explicit FMO from error concealment and combine with the proposed error concealment method can improve the PSNR with the number of undecodable MB being comparable. This is because of the unequal importance of each MB. The one-pass explicit FMO technique also has less processing time than section 4.3.2.1, 4.3.2.2 because section 4.3.2.1, 4.3.2.2 used two passes encoding.

The additional complexity to the encoder is considered negligible, because only the FMO map generation process is changed, the encoder structure is the same as the JM reference encoder. The added complexity to the decoder is minimal when the proposed error concealment algorithm is implemented. The added complexity is only in the implementation of solving Eq. 3.10. This has the minimal impact when in

comparison to the other more time consuming task in the decoder, such as motion compensation. In summary, this method is suitable for low resolution video sequence such as mobile communication and video conference.

Table 4.4 Comparison of average PSNR(dB)

	akiyo		carphone		claire		foreman		moblie		news		mother		coastguard		salesman		container		hall		highway		silent	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	30.71	27.63	23.04	19.01	30.78	24.28	17.11	15.12	15.30	12.65	24.00	20.56	28.48	25.11	21.79	17.15	25.89	24.59	26.34	22.43	26.81	25.06	29.45	25.57	23.88	22.36
2P_FMO_bitcnt	34.46	32.17	23.29	21.94	31.12	30.07	20.61	18.19	16.53	14.99	26.73	23.31	31.58	28.09	21.48	19.79	29.15	**27.87**	30.81	28.47	30.48	26.74	28.79	29.10	**28.80**	25.64
2P_FMO_Dist	34.75	31.58	25.49	22.62	30.66	29.59	20.62	**19.81**	16.50	14.91	25.77	23.39	31.96	27.38	21.18	17.97	29.73	27.12	30.49	**28.95**	31.15	26.25	**30.47**	28.45	28.49	25.68
2P_FMO_SimDis	34.43	31.57	25.41	22.08	31.61	29.26	18.53	18.87	16.52	15.21	**27.78**	**24.47**	32.16	28.46	20.99	18.69	30.50	27.36	30.87	28.40	30.54	27.04	29.42	29.39	25.25	25.23
1P_FMO_bitCnt	32.94	32.17	26.13	22.42	31.75	29.39	20.48	18.23	16.60	**15.46**	26.81	24.40	31.74	28.46	21.29	19.39	29.85	27.31	30.66	28.65	31.55	27.66	29.41	29.59	26.39	**25.90**
1P_FMO_Dist	34.93	31.83	26.09	22.08	**31.66**	**29.97**	20.72	18.25	16.69	15.05	27.19	24.01	32.38	28.84	21.75	19.60	30.19	27.19	28.44	28.40	28.88	26.17	29.73	29.71	25.21	25.40
1P_FMO_SimDis	34.56	32.37	26.10	22.67	30.59	28.67	20.45	18.42	16.45	14.39	25.79	23.61	32.26	28.55	22.04	19.97	30.29	26.90	30.83	27.85	31.90	27.08	29.34	29.72	27.53	24.73
1P_FMO_Dis+EC	**35.24**	**33.11**	**26.64**	**23.46**	31.63	29.58	**21.27**	19.33	**16.77**	14.99	26.46	24.18	**32.80**	**29.04**	**22.45**	**20.42**	**30.57**	27.67	**31.50**	28.52	**32.50**	**27.87**	30.19	**30.42**	28.10	25.39

Remark : Two-pass Explicit FMO map

- *bit count (spatial information) as call* "2P_FMO_bitcnt"
- *distortion measure (temporal information) as call* "2P_FMO_Dist"
- *distortion by simulated spatial/temporal EC as call* "2P_FMO_SimDis"

One-pass Explicit FMO map

- *bit count (spatial information) as call* "1P_FMO_bitCnt"
- *distortion measure (temporal information) as call* "1P_FMO_Dist"
- *distortion by simulated spatial/temporal EC as call* "1P_FMO_SimDis"
- *combined distortion by simulated spatial/temporal EC and EC as call* "1P_Dis+EC"

Table 4.5 Comparison of number of undecodable macroblocks

	akiyo		**carphone**		**claire**		**foreman**		**moblie**		**news**		**mother**		**coastguard**		**salesman**		**container**		**hall**		**highway**		**silent**	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	1547	5367	1380	5634	1346	5269	1499	5740	1382	4168	1424	5754	1038	4978	1499	5659	1383	5011	1547	7045	1499	5776	1134	3569	1447	5175
2P_FMO_bitcnt	758	**1210**	742	1329	780	1257	**712**	1243	394	860	**651**	1220	764	**1039**	968	1321	563	1115	801	1681	659	1278	616	923	546	1071
2P_FMO_Dist	701	**1210**	651	1202	833	**1182**	746	1187	470	933	692	1298	777	1130	1068	1284	**547**	1164	818	**1487**	537	1149	633	**908**	597	1159
2P_FMO_SimDis	690	1264	692	**1163**	760	1257	757	**1120**	**316**	**810**	658	1318	759	1076	1017	**1255**	558	1124	763	1567	**416**	**1081**	588	930	741	1086
1P_FMO_bitCnt	805	1213	685	1284	799	1221	728	1224	470	878	680	1363	762	1115	912	1370	559	1141	810	1517	614	1122	**584**	935	566	1093
1P_FMO_Dist	**674**	1290	692	1248	**757**	1241	740	1261	447	854	661	1291	735	1159	980	1278	580	**1079**	1045	1646	562	1251	625	971	728	1138
1P_FMO_SimDis	687	1212	**634**	1203	798	1253	716	1279	459	863	702	**1185**	**728**	1109	**819**	1337	579	1102	**732**	1510	509	1124	627	916	**536**	**1019**
1P_FMO_Dis+EC	687	1212	**634**	1203	798	1253	716	1279	459	863	702	**1185**	**728**	1109	**819**	1337	579	1102	**732**	1510	509	1124	627	916	**536**	**1019**

Table 4.6 Comparison of average PSNR(dB)

	akiyo		**carphone**		**claire**		**foreman**		**moblie**		**news**		**mother**		**coastguard**		**salesman**		**container**		**hall**		**highway**		**silent**	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	30.71	27.63	23.04	19.01	30.78	24.28	17.11	15.12	15.30	12.65	24.00	20.56	28.48	25.11	21.79	17.15	25.89	24.59	26.34	22.43	26.81	25.06	29.45	25.57	23.88	22.36
RT_Dis+EC	35.24	**33.11**	26.64	23.46	31.63	29.58	21.27	**19.33**	16.77	14.99	26.46	24.18	32.80	29.04	22.45	20.42	30.57	27.67	**31.50**	28.52	32.50	27.87	30.19	**30.42**	28.10	25.39
RT_Dis+EC (RandomErr 5%)	33.17	32.34	23.97	**23.73**	**31.83**	29.82	20.84	18.95	**16.98**	**15.50**	27.43	**25.57**	32.23	28.68	**22.65**	**20.90**	29.61	28.05	31.34	**29.02**	31.40	27.99	**30.60**	30.29	27.14	24.81
RT_Dis+EC (RandomErr 10%)	**35.30**	32.44	25.85	23.61	31.49	**31.03**	**21.42**	19.10	16.85	15.48	**27.84**	23.74	**32.99**	**29.33**	21.63	20.18	**30.67**	**28.08**	29.14	28.92	**32.59**	28.46	29.21	29.52	**28.66**	**25.77**

Table 4.7 Comparison of number of undecodable macroblocks

	akiyo		**carphone**		**claire**		**foreman**		**moblie**		**news**		**mother**		**coastguard**		**salesman**		**container**		**hall**		**highway**		**silent**	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
No FMO	1547	5367	1380	5634	1346	5269	1499	5740	1382	4168	1424	5754	1038	4978	1499	5659	1383	5011	1547	7045	1499	5776	1134	3569	1447	5175
RT_Dis+EC	**687**	**1212**	**634**	1203	798	1253	**716**	1279	459	863	702	**1185**	**728**	1109	**819**	1337	579	**1102**	**732**	**1510**	**509**	**1124**	627	**916**	**536**	**1019**
RT_Dis+EC (RandomErr 5%)	821	1232	774	**1149**	805	1251	743	1348	**426**	856	670	1228	783	**1108**	**819**	1418	**549**	1132	803	1554	515	1147	608	937	590	1025
RT_Dis+EC (RandomErr 10%)	717	1237	685	1151	**709**	**1171**	720	**1255**	463	**828**	**637**	1303	759	1168	922	**1254**	553	1202	995	1531	537	1154	**599**	931	621	1120

Figure 4.9 16th carphone sequence (fast fading)

Figure 4.10 16th carphone sequence (slow fading)

(a) Original (b) No FMO (c) FMO spatial Information,

(d) FMO temporal Information (e) FMO Simulated distortion by Error Concealment

(f) One-pass FMO spatial Information (g) One-pass FMO temporal Information

(h) One-pass FMO Simulated distortion by Error Concealment

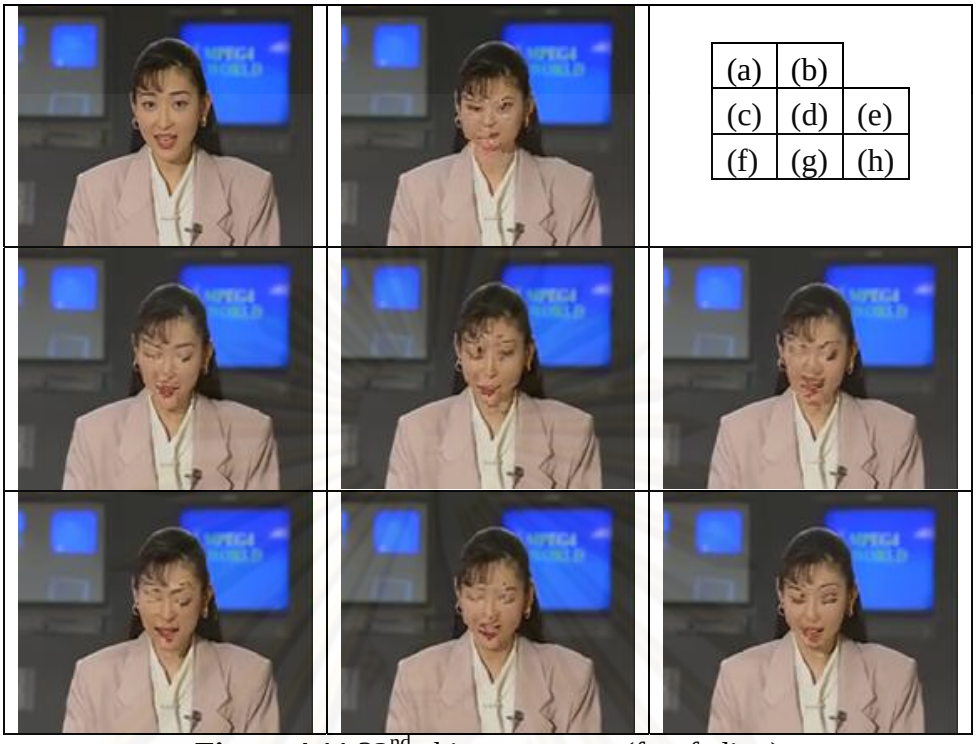


Figure 4.11 32nd akiyo sequence (fast fading)

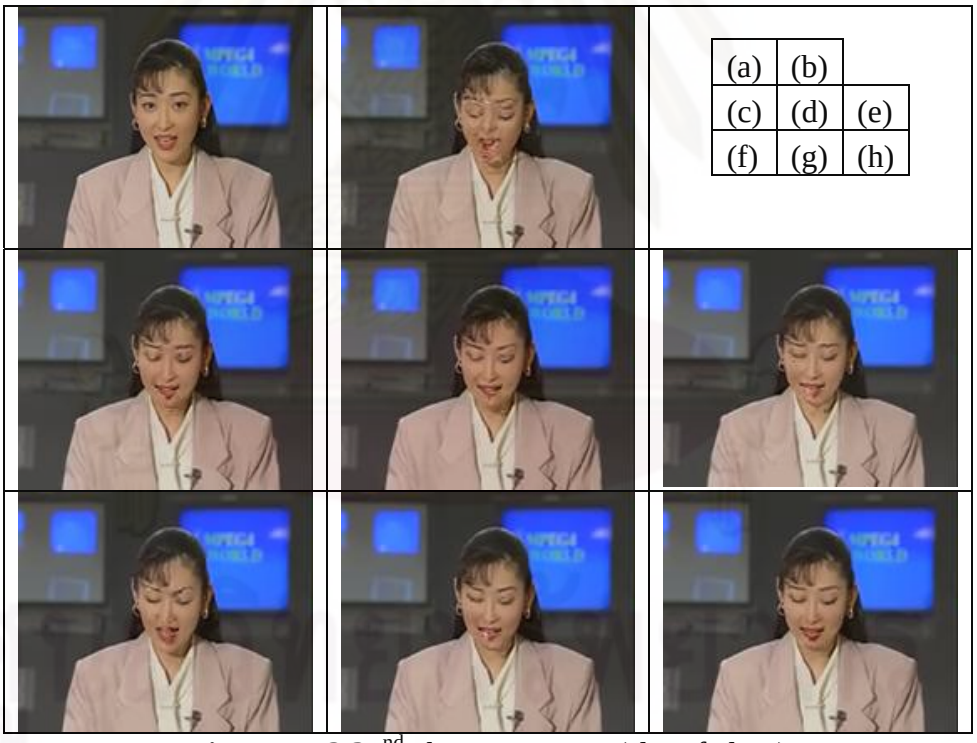


Figure 4.12 32nd akiyo sequence (slow fading)

(a) Original (b) No FMO (c) FMO spatial information,

(d) FMO temporal information (e) FMO Simulated distortion by error concealment

(f) One-pass FMO spatial information (g) One-pass FMO temporal information

(h) One-pass FMO Simulated distortion by error concealment

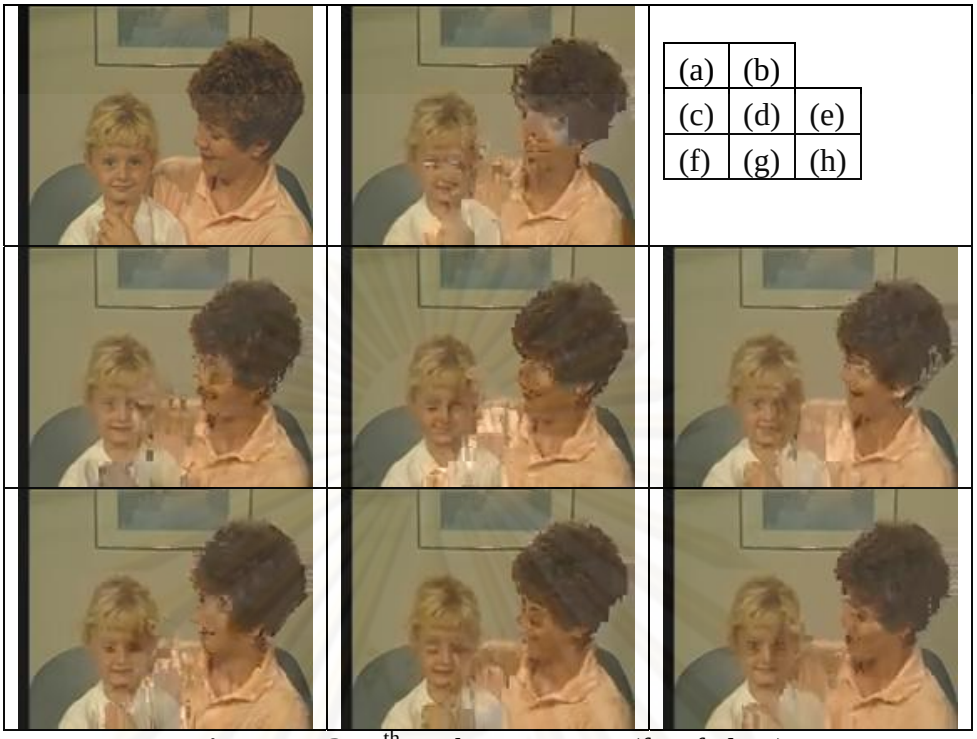


Figure 4.13 55th mother sequence (fast fading)

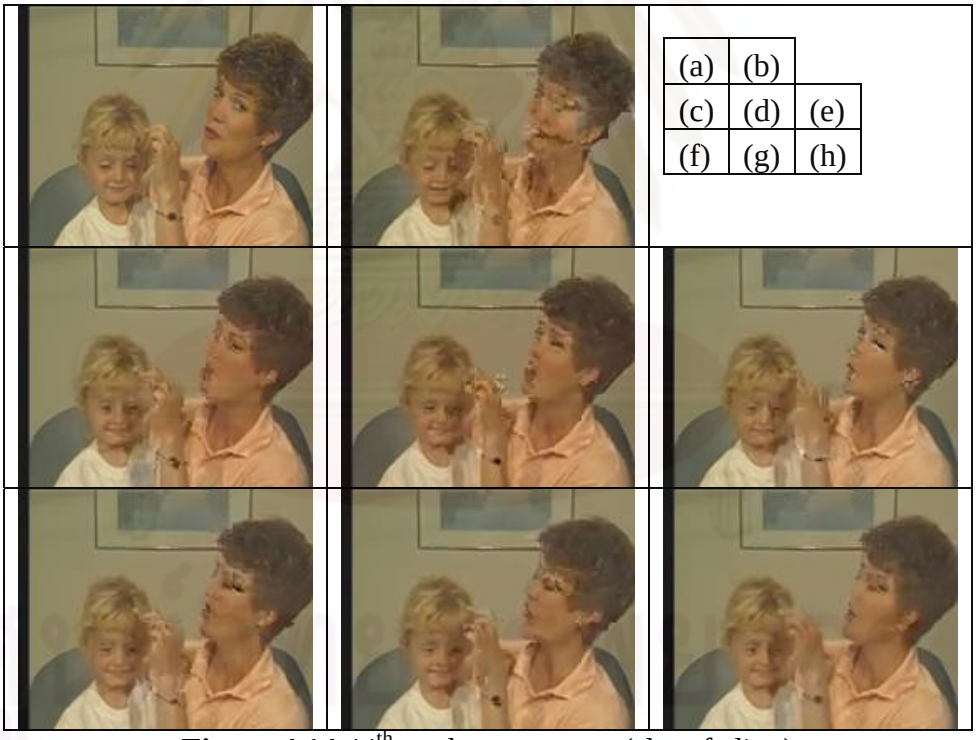


Figure 4.14 44th mother sequence (slow fading)

(a) Original (b) No FMO (c) FMO spatial information,

(d) FMO temporal information (e) FMO Simulated distortion by error concealment

(f) One-pass FMO spatial information (g) One-pass FMO temporal information

(h) One-pass FMO Simulated distortion by error concealment

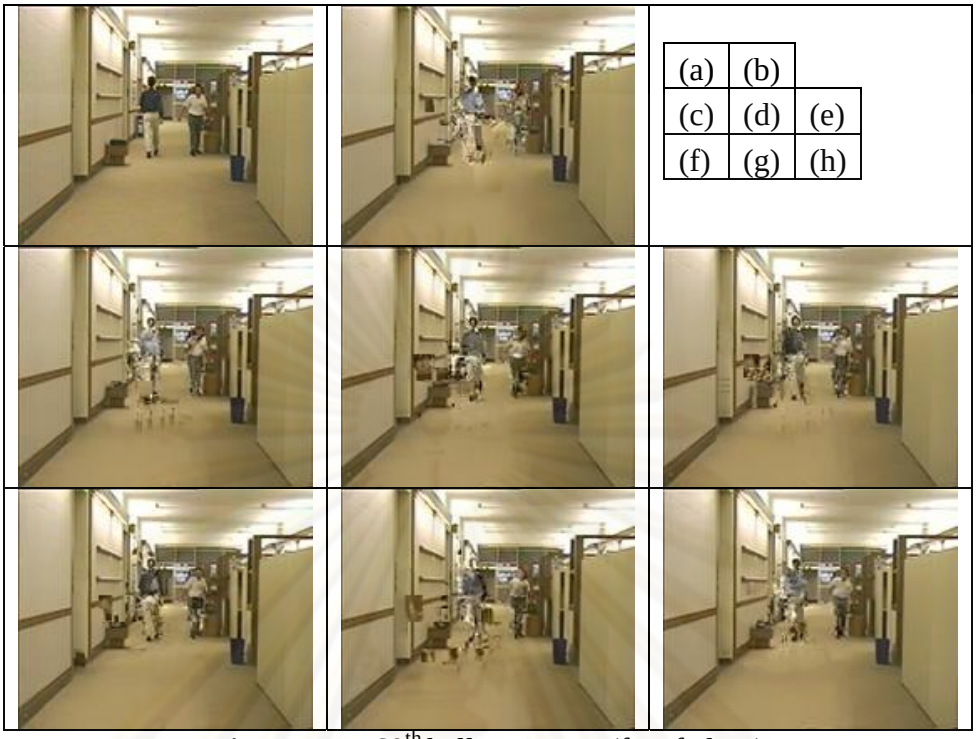


Figure 4.15 60th hall sequence (fast fading)

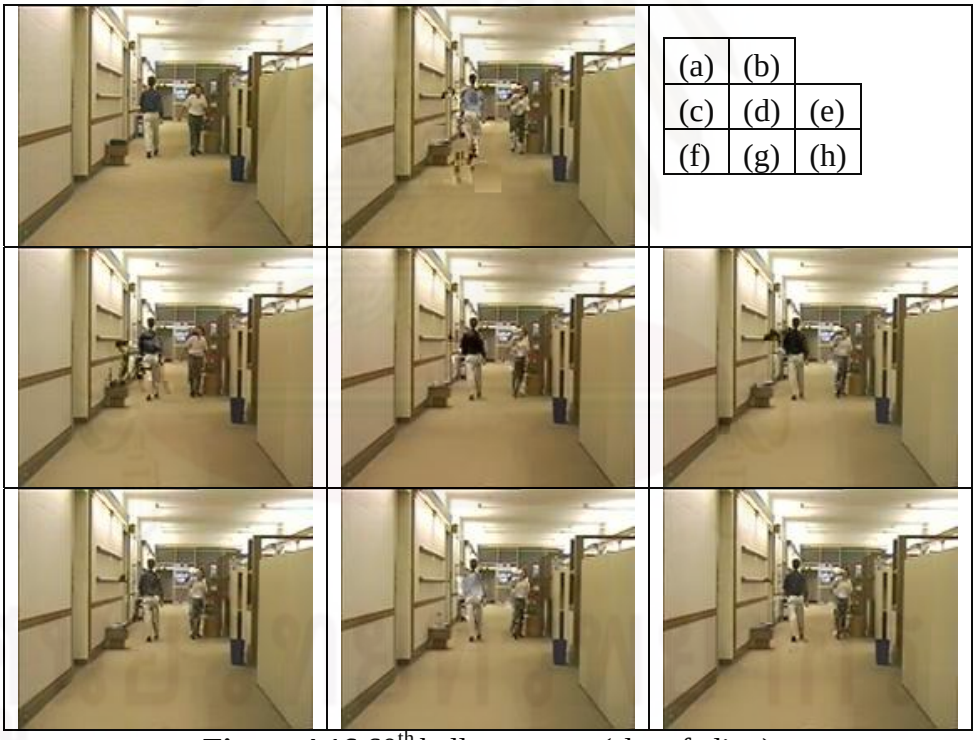


Figure 4.16 60th hall sequence (slow fading)

(a) Original (b) No FMO (c) FMO spatial Information,

(d) FMO temporal Information (e) FMO Simulated distortion by Error Concealment

(f) One-pass FMO spatial Information (g) One-pass FMO temporal Information

(h) One-pass FMO Simulated distortion by Error Concealment

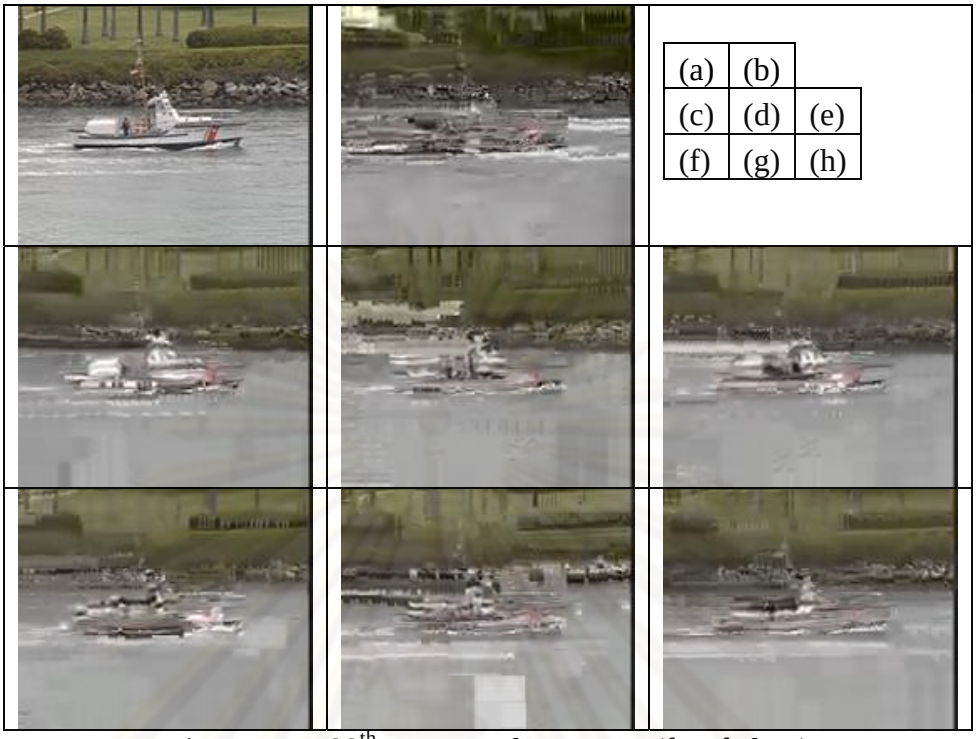


Figure 4.17 88th coastguard sequence (fast fading)

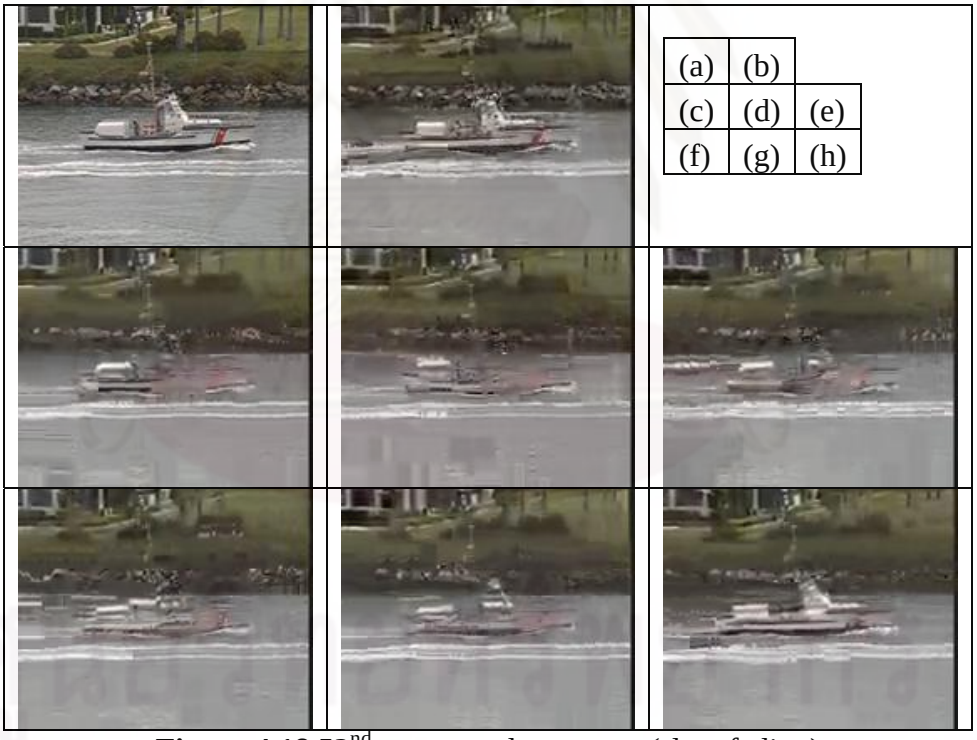


Figure 4.18 52nd coastguard sequence (slow fading)

(a) Original (b) No FMO (c) FMO spatial information,

(d) FMO temporal information (e) FMO Simulated distortion by error concealment

(f) One-pass FMO spatial information (g) One-pass FMO temporal information

(h) One-pass FMO Simulated distortion by error concealment

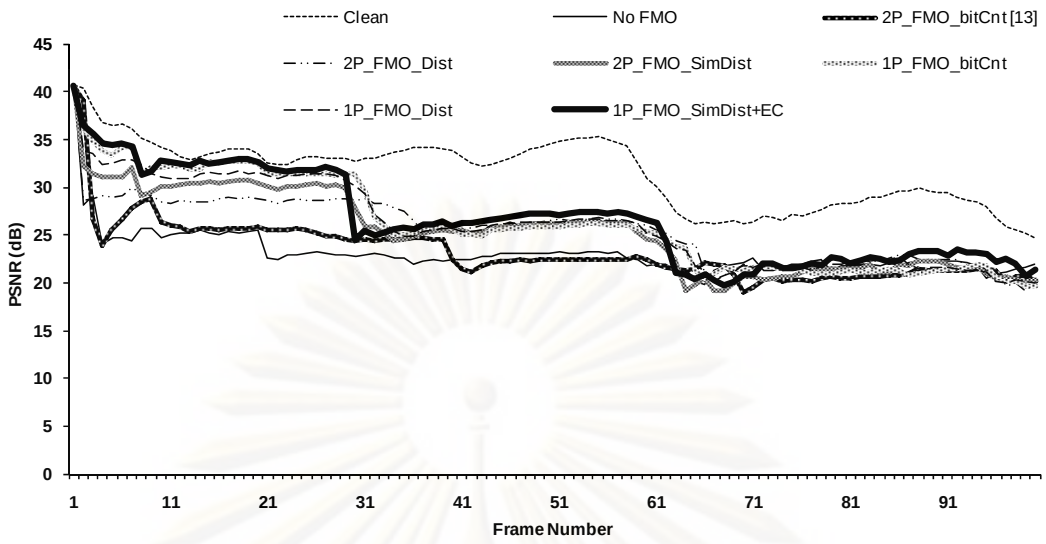


Figure 4.19 Comparison of PSNR among 7 methods with respect to clean channel for carphone sequence under slow fading channel.

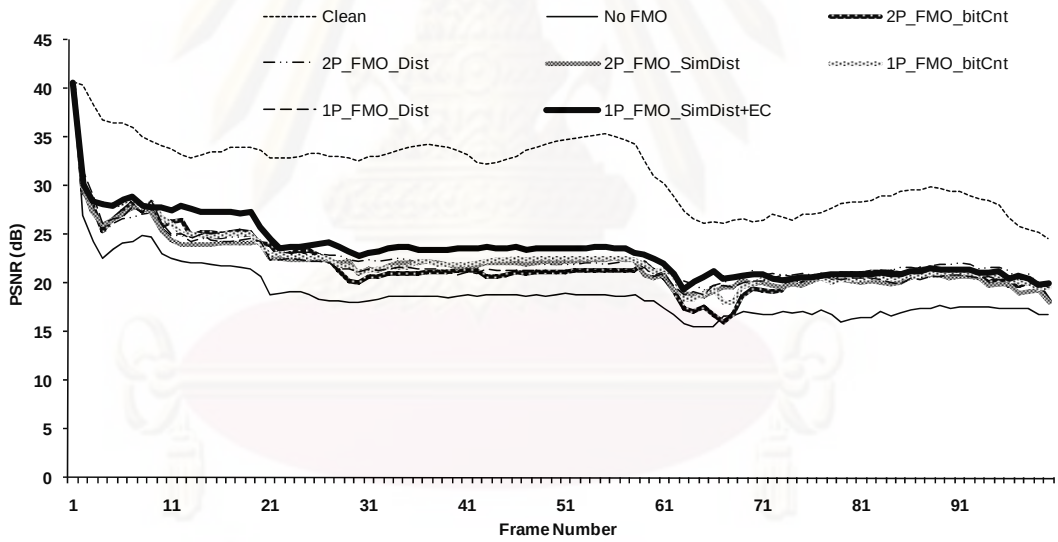


Figure 4.20 Comparison of PSNR among 7 methods with respect to clean channel for carphone sequence under fast fading channel.

4.3.3 Joint Explicit FMO map, Adaptive Interleaving depth and Forward Error Correction

This section shows the results of using joint FMO, FEC and interleaving. The simulation conditions are stated in Section II. We use 4 standard test video sequences, namely, Akiyo, Carphone, Mobile, and News. We used the average PSNR and the number of undecodable MBs to evaluate the performance of our proposed technique. Having a lower number of undecodable MB indicates improved visual quality but it does not directly translate to a better average PSNR because not all MBs have equal the same effect on the PSNR.

4.3.3.1 Two-Pass Explicit FMO Map

Table 4.6 shows the number of undecodable macroblocks for slow and fast fading channels. By incorporating FMO with FEC and adaptive interleaving depth, we can observe the considerable reduction of undecodable macroblock in every test sequences when compared with no FMO and FEC and using FMO with no FEC. For slow fading case, the proposed method could reduce the average of the number of undecodable macroblocks of up to 70%. While in the case of fast fading case, the average of the number of undecodable macroblocks could be reduce up to 88%. The proposed algorithm also outperforms when compared to the methods of using FMO only and in some cases of FMO and FEC coding.

Table 4.7 shows the average PSNR of video in the scenario of slow and fast fading channels. The simulation results show that for the case of slow fading, the PSNR improvement of the proposed method is increased up to 5 dB. For fast fading case, the proposed method can achieve PSNR improvement of up to 7.5 dB. If compared to our previous work in (Aramvith, 2006), the improvement of average PSNR for slow fading case is up to 1.7 dB. While the improvement of average PSNR for fast fading case is up to 2 dB. As already mentioned, the quality measurement of video in term of PSNR depends heavily on the location of bit errors and also the error concealment method applied. As simple non-motion compensated error concealment is used in this study, we expect that the higher PSNR improvement could be achieved if more sophisticated methods of error concealment technique is applied in further study. Nevertheless, there is no direct relation between the number of undecodable macroblock and the PSNR as several other factors have to be taken into account.

Figures 4.23-4.26 show the average PSNR curve of carphone and akiyo test sequence in the slow and fast fading case, respectively. From the curves, it can be clearly observed the PSNR improvement of this method.

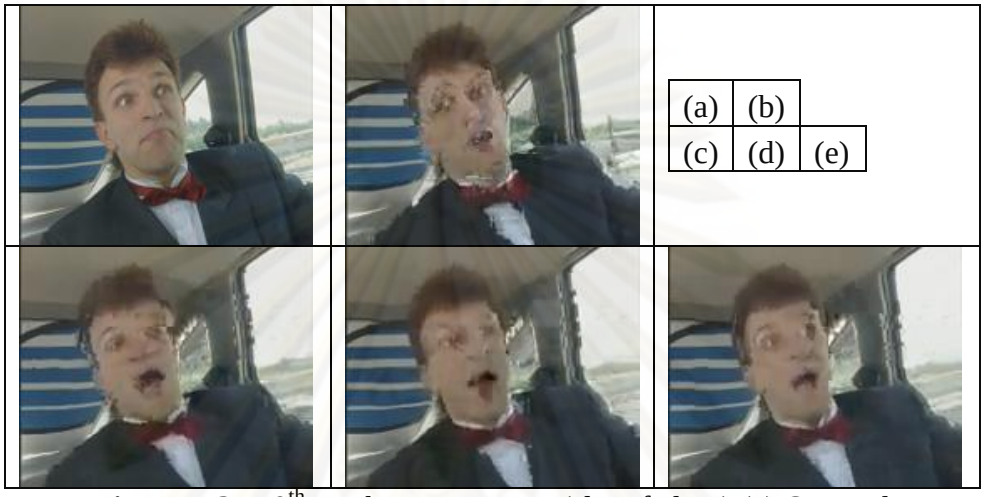


Figure 4.21 40th carphone sequence (slow fading) (a) Original (b) No FMO+ No FEC (c) FMO+ No FEC (d) FMO + FEC (e) Our proposed

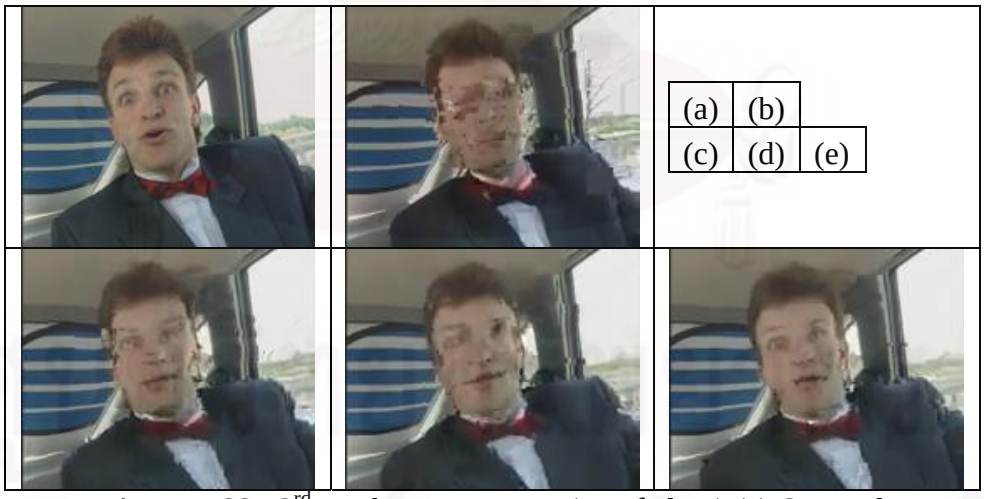


Figure 4.22 23rd carphone sequence (sast fading) (a) Original (b) No FMO+ No FEC (c) FMO+ No FEC (d) FMO + FEC (e) Our proposed

Table 4.6 Comparison of average PSNR (two-pass explicit FMO)

	akiyo		carphone		mobile		news	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
NoFMO+NoFEC	30.71	27.63	23.04	19.01	15.30	12.65	24.00	20.56
FMO+NoFEC	**34.46**	32.17	23.29	21.94	16.53	14.99	26.73	23.31
FMO+FEC	33.76	33.13	23.22	23.76	17.43	15.48	28.05	24.31
Our proposed	34.07	**35.13**	**24.92**	**24.98**	**17.68**	**15.69**	**29.02**	**24.87**

Table 4.7 Comparison of number of undecodable MBs (two-pass explicit FMO)

	akiyo		carphone		mobile		news	
	slow	**fast**	**slow**	**fast**	**slow**	**fast**	**slow**	**fast**
NoFMO+NoFEC	1547	5367	1380	5634	1382	4168	1424	5754
FMO+NoFEC	758	1210	742	1329	394	860	607	1318
FMO+FEC	**489**	**504**	**396**	**551**	199	639	419	**564**
Our proposed	504	555	418	**551**	**153**	**607**	**378**	607

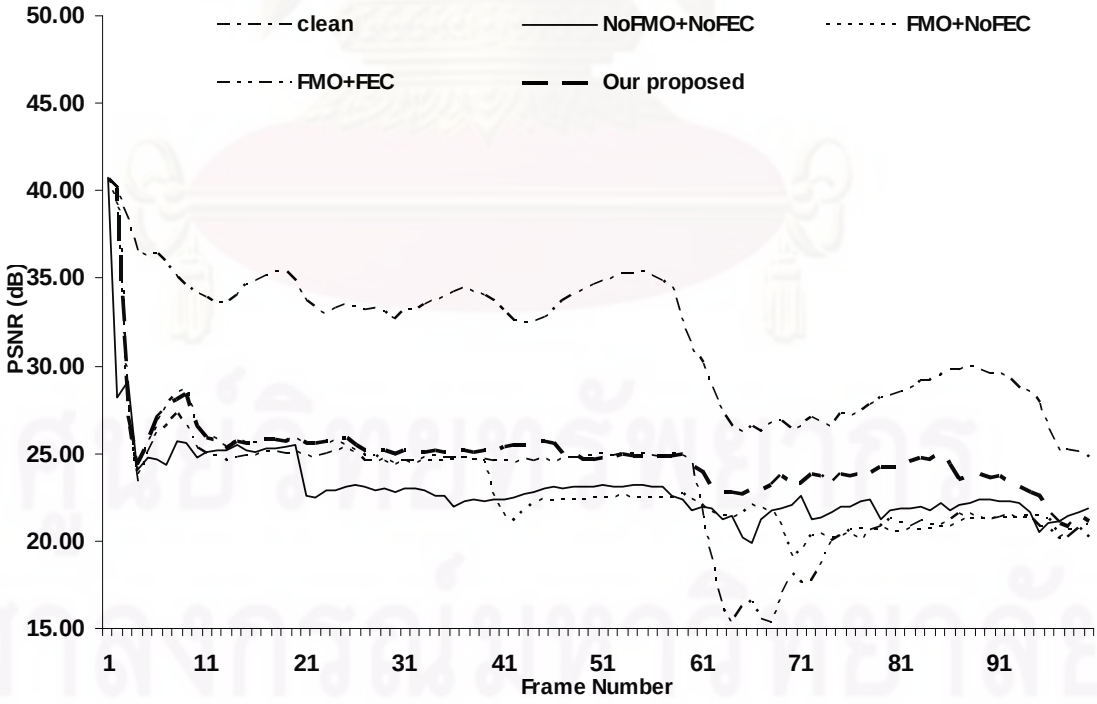


Figure 4.23 Comparison of PSNR among 4 methods with respect to clean channel for carphone sequence under slow fading channel (two-pass explicit FMO)

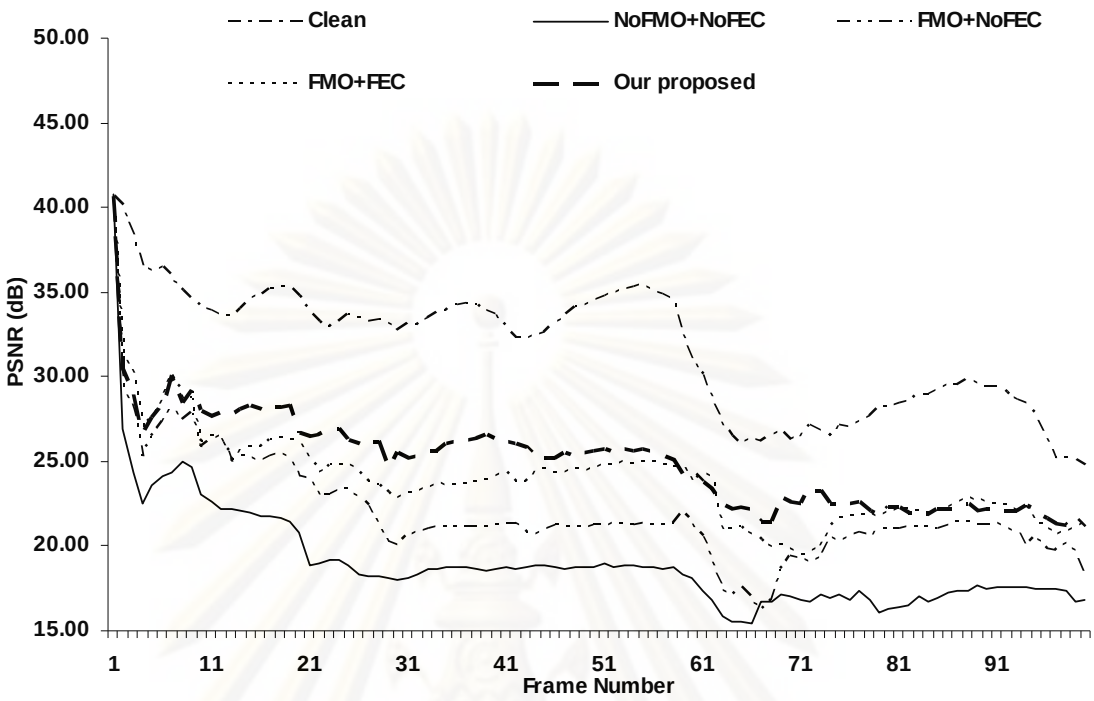


Figure 4.24 Comparison of PSNR among 4 methods with respect to clean channel for carphone sequence under fast fading channel (two-pass explicit FMO)

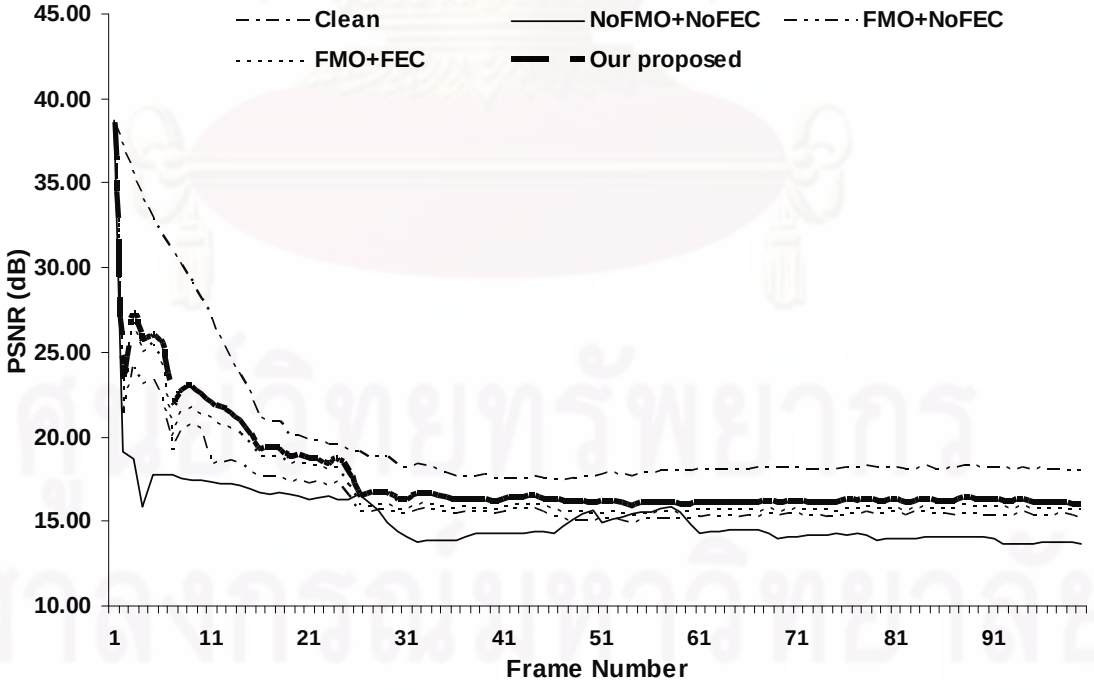


Figure 4.25 Comparison of PSNR among 4 methods with respect to clean channel for mobile sequence under slow fading channel (two-pass explicit FMO)

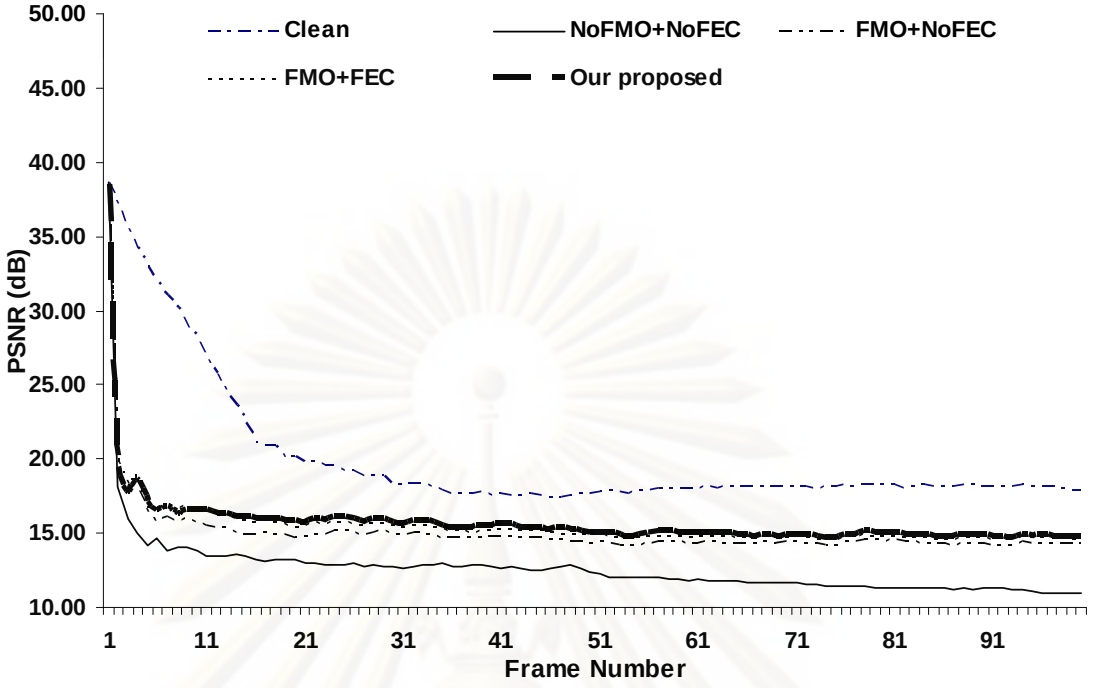


Figure 4.26 Comparison of PSNR among 4 methods with respect to clean channel for mobile sequence under fast fading channel (two-pass explicit FMO)

4.3.3.2 One-Pass Explicit FMO Map

Table 4.8 shows the number of undecodable macroblocks for slow and fast fading channels. By incorporating FMO with FEC and adaptive interleaving depth, we can observe the considerable reduction of undecodable macroblock in every test sequences when compared with no FMO and FEC and using FMO with no FEC. For slow fading case, the proposed method could reduce the average of the number of undecodable macroblocks of up to 89%. While in the case of fast fading case, the average of the number of undecodable macroblocks could be reduce up to 90%. The proposed algorithm also outperforms when compared to the methods of using FMO only and in some cases of FMO and FEC coding.

Table 4.9 shows the average PSNR of video in the scenario of slow and fast fading channels. The simulation results show that for the case of slow fading, the PSNR improvement of the proposed method is increased up to 5.4 dB. For fast fading case, the proposed method can achieve PSNR improvement of up to 7.45 dB.

Table 4.8 Comparison of average PSNR (one-pass explicit FMO)

	akiyo		**Carphone**		**moblie**		**news**	
	Slow	**Fast**	**Slow**	**Fast**	**Slow**	**Fast**	**Slow**	**Fast**
NoFMO+NoFEC	30.71	27.63	23.04	19.01	15.30	12.65	24.00	20.56
FMO+NoFEC	**34.46**	32.17	23.29	21.94	16.53	14.99	26.73	23.31
FMO+FEC	33.52	33.02	23.16	23.92	17.13	15.48	27.94	24.56
FMO+FEC+adpInv	34.07	**35.13**	24.92	**24.98**	17.68	15.69	29.02	24.87
1PFMO+FEC+adpInv	31.34	35.09	**28.45**	24.83	**17.86**	**16.01**	**29.25**	**25.48**

Table 4.9 Comparison of number of undecodable MBs (one-pass explicit FMO)

	akiyo		**Carphone**		**moblie**		**news**	
	Slow	**Fast**	**Slow**	**Fast**	**Slow**	**Fast**	**Slow**	**Fast**
NoFMO+NoFEC	1547	5367	1380	5634	1382	4168	1424	5754
FMO+NoFEC	758	1210	742	1329	394	860	607	1318
FMO+FEC	525	**463**	439	**477**	251	**568**	456	**520**
FMO+FEC+adpInv	**504**	555	418	551	**153**	607	**378**	607
1PFMO+FEC+adpInv	576	624	**303**	549	155	610	400	637

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

In this section, conclusions our algorithm and recommendation for future research in develop error-resilient video coding techniques for wireless video transmissions.

5.1 Conclusions

In this dissertation divide our algorithm to four parts.

First, we investigate the use of the two-pass explicit FMO option of H.264/AVC for wireless video transmission in the slow and fast fading cases. This method addresses this issue and proposes the use of the encoder's macroblock coded-bit-count which acts as spatial information and distortion measure based on the concealment error which acts as temporal information as indicators for a choice of macroblock-address-map of each picture. Our simulation results indicate the benefit of using both of our proposed explicit FMO technique can decrease the amount of undecodable macroblocks of up to 63%.

Second, the extension of integrating the two-pass explicit FMO using spatial and temporal information to generate more suitable maps. Also investigate the integration of utilizing temporal indicator in term of distortion measure. The framework of using joint two-pass explicit FMO map based on distortion simulated from spatial and temporal error concealment at the encoder and a new error concealment at decoder. Our simulation results show the benefit of using explicit FMO especially when using higher number of slice group per picture and apply new error concealment scheme. The amount of undecodable MBs is decreased of up to 81% compared to no FMO and the PSNR improvement is up to 5 dB. The use of error concealment at the decoder and its effectiveness combined with explicit FMO is currently under study.

Third, we investigate the framework of one-pass explicit FMO of H.264/AVC for wireless channel in the slow and fast fading cases, such as one-pass explicit FMO map using bit-count information, one-pass explicit FMO map using distortion measure and one-pass explicit FMO map from the simulated error concealment as

indicators for a choice of macroblock-address-map of each picture. One-pass use feedback information of previous frame for generates MBAmap of current frame. We propose a new error concealment method using residual information from the simulated error concealment as generating one-pass explicit FMO map. Our simulation results indicate that the benefit of using one-pass explicit FMO using a higher number of slice group per picture and combine a new error concealment method. For slow fading, the amount of undecodable MBs of one-pass explicit FMO from error concealment are decreased of up to 66.79% compared with No FMO and the PSNR improvement are up to and 5.68 dB. For fast fading, the amount of undecodable MBs of one-pass from error concealment are decreased of up to 80.54% compare with No FMO and the PSNR improvement are up to 6.09 dB. The results show one-pass explicit FMO map from error concealment is the best method because the indicator used correlates with the error concealment method. If we implement more sophisticated error concealment techniques the PSNR will be improved and decoded image will be better.

Forth, the framework of joint explicit map FMO using bit count information, FEC coding, and adaptive interleaving depth has been proposed. The scheme has been adapted according to the predicted channel condition derived from the probability that next state of channel is good. If the channel condition is good, we use only FMO as an error mitigation scheme. If the channel condition is bad, we use FMO, FEC, and interleaving. The interleaving is adjusted for each frame according to the mean burst length statistics obtained from feedback information of previous frames. The simulation results under slow and fast fading wireless channel scenarios show that our proposed scheme help reduce the number of undecodable macroblock of up to 70% and achieve the PSNR improvement of up to 88%. Thus, the proposed framework is feasible error protection scheme for wireless video transmission.

5.2 Recommendations

- We can modify different error concealment method in the spatial and temporal domain for improve the best quality of video.
- MBAmap can be generating from another parameter.

REFERENCES

Aramvith, S., and Hantanong, W. Joint Flexible Macroblock Ordering and FEC for H.264 Wireless Video Transmission. Proceeding of IEEE International Symposium on Intelligent Signal Processing and Communications, pp. 139-142. 2006.

Chen, T.C., and et. al. A Real-Time Software Based End-To-End Wireless Visual Communications Simulation Platform. Proceeding SPIE Visual Communications and Image Processing, , pp. 1068-1074. 1995.

Chen, T., Zhang, X., and Shi, Y.Q. Error Concealment using Refined Boundary Matching Algorithm. IEEE Transactions on Circuits and System for Video Technology 2003.

Dhondt, Y., Lambert, P., and Walle, R.V. A Flexible Macroblock scheme for Unequal Error Protection. International Conference on Image Processing, pp. 829-839. 2006.

Gilbert, E. N. Capacity of a Burst-Noise Channel. Bell System Technology Journal 1960: 1253-1266.

H.264/AVC Reference Software Download [Online]. Available from: http://iphome.hhi.de/suehring/tml/download/.

Hantanong, W., and Aramvith, S. Analysis of Macroblock-to-slice Group Mapping for H.264 Video Transmission over Packet-Based Wireless Fading Channel. 48th Midwest Symposium on Circuits and Systems, pp. 1541-1544. 2005.

Horowitz, S., and Wenger, M. FMO: Flexible Macroblock Ordering. Fairfax(USA): Joint Video Team, 2002.

Im, S.K., and Pearmain. A.J. Error Resilient Video Coding with Priority Data Classification using H.264 Flexible Macroblock Ordering. IET Image Precessing 1, 2 (2007): 197-204.

ITU-T Recommendation H.264 . Non-normative Error Concealment Algorithms. 2001.

ITU-T Recommendation H.264. JVT Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification [Online]. 2003. Avaliable from: http://ip.hhi.de/imagecom_G1/assets/pdfs/JVT-G050.pdf [2003, March]

ITU-T Recommendation H.264. Advance Video Coding for Generic Audiovisual, 2005.

ITU-T Recommentation H.264. Frame Loss Error Concealment for H.264/AVC, 2005.

Jung, B., Jeon, B., Kim, M., Suh, B., and Choi, S. Selective Temporal Error Concealment Algorithm for H.264/AVC. IEEE International Conference on Multimedia and Expo, pp. 411-414. 2004.

Katz, B., and Greenberg, S. Spiral-Interleaved New Explicit Flexible MacroBlock Ordering Type. International Conference on Informatin Technology: Research and Education, pp. 238-241. 2006.

Kumar, S., and et at. Error Resiliency Schemes in H.264/AVC Standard. Journal of Visual Communication and Image Representation 17 (2006): 425-450.

Kwon, S.K., Tamhankar, A., and Rao, K.R. Overview of H.264/MPEG-4 Part 10. Journal of Visual Communication and Image Representation 2006: 186-216.

Kim, D., Kim, J., and Jeong, J. Temporal Error Concealment on Optical Flow in the H.264/AVC Standard. Advanced Concepts for Intelligent Vision Systems, pp.253-262. 2006.

Lamber, P., Neve, W. D., Dhondt, Y., and Walle, R. V.. Flexible Macroblock Ordering in H.264/AVC. Journal of Visual Communication and Image Representation 17, 2 (2006): 358-375.

Luo, R., and Chen, B. A Hierarchical Scheme of Flexible Macroblock Ordering for ROI based 11.264/AVC Video Coding. 10th International Conference on Advanced Communication Technology, pp. 1579-1582. 2008.

Im, S., and Pearmain, A.J. Unequal Error Protection with the H.264 Flexible Macroblock Ordering. Proceedings of SPIE Visual Communucation. and Image Processing, 2005.

Lie, W.N., and Lin, T.C. Prescription-based Error Concealment Technique for Video Transmission on Error-Prone Channels. Journal of Visual Communication and Image Representation 8, 4 (2007): 310-321.

Lin, S., Costello, D. J., and Miller, M. J. Automatic Repeat Error Control Schemes. IEEE Communications Magazine 1984: 5-17.

Liu, L., Zhang, S., Ye, X., and Zhang, Y. Error Resilience Schemes of H.264/AVC for 3G Conversational Video. Proceeding of IEEE Conference Computer and Information Technology, pp. 657- 661. 2005.

Michael, A., Sühring, K., and Sullivan, G. Proposed H.264/MPEG-4 AVC Reference Software Manual, Plama (Spain): Joint Video Team, 2004.

Nasiopoulos, P., Mendoza, L.C., Mansou, H., and Golikeri, A. An Improved Error Concealment Algorithm for Intra-frame In H.264/AVC. IEEE International Symposium on Circuits ans Systems, pp. 320- 323. 2005.

Puri, R., Ramchangran, K., Lee, K.W., and Bharghavan, V. Forward Error Correction (FEC) Codes based Multiple Description Coding for Internet Video Streaming and Multicast. Signal Processing: Image Communication 2001: 745-762.

Richardson, E.G. H.264 and MPEG-4 Video Compression. Video Coding for Next-generation Multimedia, UK: Wiley, 2003.

Schäfer, R., Wiegand, T., and Schwarz, H. The Emerging H.264/AVC Standard, 2003.

Sivanantharasa, P., Fernando, W.A.C., and Arachchi, H. K. Region of Interest Video Coding with Flexible Macroblock Ordering. First International Conference on Industrial and Information Systems, pp. 596-599. 2006.

Sood, A., Chilamkurti, N. K., and Soh, B. Study and Analysis of an Error Resilient Technique in H.264 Video Using Flexible Macroblock Grouping. International Conference on Wireless Communication Networking and Moblie Computing, pp. 1-4. 2006.

Suh, J.W., and Ho. Y.S. Error Concealment Technique for Digital TV. IEEE Transactions Broadcasting 2002: 299- 306.

Sühring, K. H.264/AVC Software Coordination[Online]. Avaliable from: http://bs.hhi.de/~suehring/tml/

Sun, M.T., and Reibman, A.R. Compressed Video over Networks, Nex York: Marcel Dekker, 2001.

Thomos, N., Argyropoulos, S., Boulgouris, N.V., and Strintzis, M.G. Error Resilient Transmission of H.264/AVC Streams using FMO. Second European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies, 2005.

Tsekeridou, S., and Pitas, I. MPEG-2 Error Concealment based on Block Matching Principles. IEEE Transactions Circuits and System Video Technology 10 (2000): 646-658.

Thaipanich, T., Wu, P.H., and Kuo, C.C. Low-Complexity Mobile Video Error Concealment using OBMA. IEEE International Conference on Consumer Electronics, pp. 753-761. 2008.

Wang, Y., and Zhu, Q.F. Error Control and Concealment for Video Communication. Preceeding of IEEE, pp. 974-997. 1998.

Xu, Y., and Zhou, Y. H.264 Video Communication Based Refined Error Concealment Schemes. IEEE Transactions Consumer Electronics, 2004.

Zhang, J., Arnold, J.F., and Frater, M.R. A cell-loss Concealment Technique for MPEG-2 Coded Video. IEEE Transactions Circuits and System Video Technology 10 (2000): 659-665.

Zheng, J., and Chau , L. A Temporal Error Concealment Algorithm for H.264 using Lagrange Interolation. IEEE International Symposium on Circuits and Systems, 2004.

Zorzi, M., Rao, R. R., and Milstein L. B. ARQ Error Control for Fading Mobile Radio Channels. IEEE Transactions on Vehicular Technology 46 (1997): 445-455.

VITA

Miss Jantana Panyavaraporn was born on 26 February, 1982 in Nakornsawan, Thailand. She received a Bachelor's Degree of Electrical Engineering from the Faculty of Engineering, Burapha University in 2003. She worked one year at Thai Samsung Electronics Co. Ltd and then leaved for study master degree. She received Master of Engineering in Telecommunications Engineering program from King Mongkut's Institute of Technology Ladkrabang in 2006. She subsequently completed the requirements for a Doctor of Philosophy in Electrical Engineering at the Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University in 2010.