國立交通大學

資訊科學與工程研究所

碩士論文

使用多重參考幀
在基於R-D最佳化之H.264錯誤恢復影像編碼

RDO-Based Error-Resilient Coding of H.264 Video
Using Multiple Reference Frames

研究生：林宜政
指導教授：蔡文錦 博士

June 2009

Abstract

RDO-Based Error-Resilient Coding of H.264 Video
Using Multiple Reference Frames

Transmission for high-compressed video coding, such as H.264, over error-prone environment is quite a challenge due to the potential error propagation. In this thesis, we propose an error-resilient scheme for H.264 based on Error Resilient Rate-Distortion Optimization (ER-RDO) and Multiple Reference Frames (MRF). Since the error propagation may substantially degrade the received video quality, we include a Dominant Intra Frame (DIF) as one reference frame to alleviate error propagation. By predicting from the dominant intra frame, it can be more coding efficient and more error resilient. Multiple reference frames technique improves the coding efficiency and suppresses the error propagation. However, it also brings a major drawback that the computational complexity for motion estimation (ME) may increase depending on the number of reference frames. To efficiently reduce the computational complexity of ME, we propose a fast motion estimation algorithm. The experimental results show that the proposed scheme has substantial improvement over the existing schemes in providing error resilience using MRF.

Keywords: Video Transmission, Error Resilience, RDO, Multiple Reference Frames

摘要

使用多重參考幀
在基於R-D最佳化之H.264錯誤恢復影像編碼

對於高壓縮率影像編碼，如 H.264，因為潛在的錯誤傳遞效應，要透過錯誤傾向環境來進行影像傳輸是一項挑戰。在本篇論文中，我們提出了一個架構在 H.264 影像編碼下，並結合 Error Resilient Rate-Distortion Optimization (ER-RDO) 及多重參考幀 (Multiple Reference Frames, MRF) 的編碼策略。因為錯誤傳遞 (Error Propagation) 可能造成接收端影像品質的顯著下降，因此我們加入 Dominant Intra Frame (DIF) 做為在多重參考幀環境下的一個參考幀。藉由 DIF，可以有效地抑制錯誤傳遞，同時也可以確保編碼效率。而藉由 ER-RDO 技術，也可以讓影像編碼具有更高的適應性以符合不同傳輸環境及不同的影像內容。雖然 MRF 在影像編碼的效率及錯誤傳遞的抑制上有不錯的效果，然而隨著參考幀的增加，計算上的複雜度也相對地提升，尤其是在 Motion Estimation (ME) 上，因此為了有效減少計算複雜度，我們也提出了一個快速移動估計演算法 (Fast Motion Estimation Algorithm)。最後的實驗結果顯示，我們提出的方法能比既有架構在 MRF 下的方法提供更多的錯誤恢復效果。

關鍵字: 影像傳輸、錯誤恢復、R-D 最佳化、多重參考幀

Acknowledgments

I am in hearty appreciation of the continuous guidance, discussions, support, and encouragement received from my advisor, Dr. Wen-Jiin Tsai, not only in the development of this thesis, but also in every aspect of my personal growth. Thanks are due to Mr. Ting-Li Chou, Miss Zj-Juan Huang, Mr. Jian-Yu Chen, Mr. Shi-Kai Lin, and Mr. Han-Lun Wu for their valuable discussions, suggestions, and encouragement. Appreciation is also given to the colleagues of the Video Information Processing Laboratory in the Institute of Computer Science and Engineering at National Chiao Tung University for their suggestions and help during my thesis study. Finally, I also extend my profound thanks to my family for their lasting love, care, and encouragement. I dedicate this dissertation to my beloved parents and friend.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Transmission of compressed video over error-prone environment is a challenge task due to potential error spread effect. In the typical video transmission system, as figure 1.1, the video source encoder first encodes source sequence into bit-stream and encapsulates it as packets. Then, these packets are transmitted to destination through the error-prone environment. The video source decoder combines received packets and decodes it. During the stage of transmitting through the error-prone environment, packets might loss due to signal degradation, oversaturated bandwith, or routing issues. The problem results in loss of synchronization between decoder and encoder. With high ratio of compression, especially in H.264/AVC, the loss issue may cause devastatin impact of decoded video quality due to potential error spread effect, the error propagation.

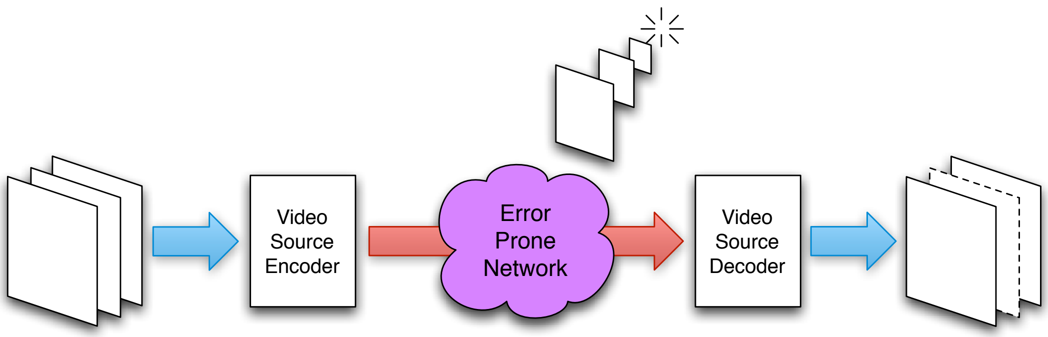

Figure 1.1 A typical video transmission system

1.1 Error Propagation

The main reason causing error propagation is a widely adopted technique in today's video coding standards, Motion Compensated Prediction (MCP). With MCP, video information could be easily predicted according to the previous coded data. This makes a great coding efficiency by removing temporal redundancy. However, it also creates a dependency between current coded and previous coded data. Even if current coded data is completely received, it still inherits the error from previous corrupted data, and becomes corrupted. Then the error propagates frame by frame, as shown in figure 1.2, causing substantial degradation of decoded video quality.

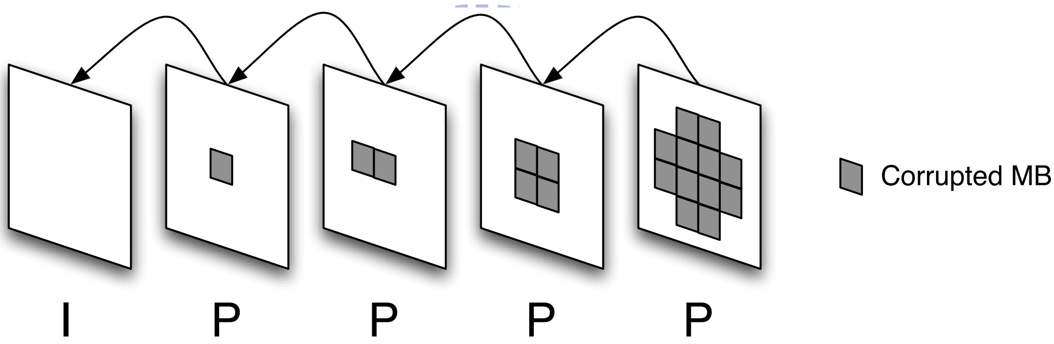

Figure 1.2 The error spread effect - Error Propagation

1.2 Error resilience tools in H.264 standard

In recent years, several error resilience tools [1] for H.264/AVC have been represented to provide robust video transmission over error-prone environment. Some of them were already included in H.264 standard, and others are implemented or introduced recently. These methods include: slices, slice groups with unequal error protection (FMO + FEC), data partitions, intra placement, and Multiple Reference Frames (MRF).

1.2.1 Slice

If a frame is coded as a packet without slice mode, it means that the loss of a packet causes the loss of the whole frame. In slice mode, each slice contains an integral number of macroblocks and will be packetized as, usually if data partitioning is not involved, a packet. By splitting a frame into slices, the probability of whole frame loss could be substantially reduced.

1.2.2 Slice Group with Unequal Error Protection

A frame could be coded as one or more slice groups, each containing an integral number of slices. One and the most important property is that the needed information to the coded MBs in a slice group is limited within the slice group. In other words, coding MBs in a slice group will not cross refer to other slice groups. This property makes a good suppression for spatial error propagation. Flexible Macroblock Ordering (FMO) [2] allows assigning MBs to slices in different orders, e.q. interleaved and dispersed. The main goal of this technique is to limit the scatter possible errors to the whole frame, since the error will not propagate to other slice groups.

Another technique which is well adopted with slice groups for error resilient coding is Unequal Error Protection (UEP) [3]. It first classifies MBs to different slice groups according to specified criteria, and then, each slice group is assigned with different numbers of protection bits. Forward Error Correcting (FEC) [4] such as Reed-Solomon Coding (RS-Code) [5] is a widely adopted method for data protection. FEC guarentees that, if the number of erased packets is less than the decoding threshold for the FEC code, the original data can be recovered perfectly.

1.2.3 Data Partition

Data Partition technique further splits a slice into three partitions, each is encapsulated as a packet. The three partition types are:

- Partition A, containing header information such as MB types, quantization parameters, and motion vectors. This information is the most important part because, without it, the whole slice becomes unusable, even if the other partitions are available.

- Partition B (intra partition), containing intra coded residuals. Since intra information can stop further error propagation, this partition is more important than the inter partition.

- Partition C (inter partition), containing inter coded residuals. Inter partition is the least important part.

If the intra or inter partitions (B or C) are lost, the available header information from partition A can still be used in error concealment.

1.2.4 Intra Placement

H.264 allows intra macroblock prediction in inter frames. It could not only achieve better coding efficiency, but also add robustness into inter frames. However, with more intra information, more coding bits are required.

1.2.5 Multiple Reference Frames (MRF)

Fore old video coding standard such as MPEG II, inter-coded macroblocks in P-frames predict from only one frame immediately preceeding the current frame. The Multiple

Reference Frames technique in H.264 allows block-level prediction from a set of previously encoded frames, called candidate reference frames. As shown in figure 1.3, current encoded MB contains three blocks, the first 8×8 block referencing to F_{n-2}, the second 8×8 block referencing to F_{n-1}, and the 8×16 block referencing to F_{n-3}.

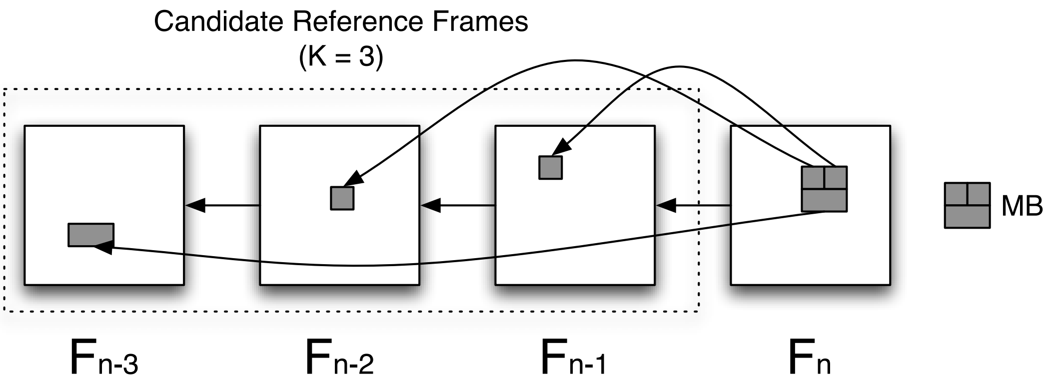

Figure 1.3 Multiple Reference Frames

With MRF, even if some macroblocks in F_{n-2} and F_{n-1} are corrupted, those blocks predicted from F_{n-3} can still be correctly decoded. By involving MRF, coded video becomes more error resilient and coding efficient.

1.3 Rate-Distortion Optimization based ER tools

H.264 standard offers a rate-distortion optimized (RDO) technique which gets a great tradeoff between coding rate and source distortion for motion estimation and mode decision. Since the original RDO doesn't consider the channel distortion during video transmission, several error-resilient RDO (ER-RDO) techniques have been proposed for video transmission in error-prone environment [6, 7].

The R-D optimization technique has been well-studied for the source video coding in error-free environment [8]. However, it is not applicable to error-prone environment due to improper distortion estimation scheme which concerns source distortion only. To encompass error resilience, the R-D optimization scheme should jointly consider the source distortion and the potential channel distortion together so as to achieve the best tradeoff between the overall end-to-end distortion and the rate.

Recently, an Error RoBust Rate-Distortion Optimization method, referred to as ERB-RDO, has been developed for video coding in error-prone environment [6,7], and has been adopted in the H.264/AVC test model [9]. ERB-RDO eastimates the expected end-to-end distortion by calculating the mean value of distortion from K copies, each with a random loss maps. The expected end-to-end distortion can be estimated very accurately if K is large enough. However, this would cost much higher computational complexity. Therefore, a number of approachs have been proposed for accurate end-to-end distortion estimation at low computational complexity.

Recursive Optimal per-Pixel Estimation (ROPE) proposed by [10] has been recognized as an effective method to estimate the expectation of end-to-end distortion. The estimation is integrated into a RD-based scheme for optimal mode selection. Several end-to-end distortion based RDO schemes [10–12] use a similar way to recursively calculate the expected end-to-end distortion. In [10], R. Zhang proposes a way for mode decision that recursively calculates the first and second distortion items, which are repesented as

$$\begin{aligned} d_n^i &= E\{(f_n^i - \tilde{f}_n^i)^2\} \\ &= (f_n^i)^2 - 2f_n^i E\{\tilde{f}_n^i\} + E\{(\tilde{f}_n^i)^2\} \end{aligned}$$

However, the seperation is very sensitive to the approximation errors caused by averaging operations such as subpixel motion compensated prediction. In [11], H. Yang proposes an end-to-end distortion estimation solution that seperates the distortion into three items: source distortion, error-propagated distortion and error-concealment distortion. With their distortion model, they applied error resilient RDO to both motion estimation and mode decision. Based on the end-to-end distortion model in [11], Y. Zhang further proposes a generalized estimation model [12] for fitting multiple reference frames (MRF).

1.4 Multiple Reference Frame based ER tools

Multiple Reference Frames (MRF) has been proven a powerful tool to improve both coding efficiency and suppression of error propagation. Thus, many MRF-based techniques have been proposed for improvement of error resilience [13, 14].

1.4.1 Automatic Repeat reQuest (ARQ)

Automatic Repeat Request is an error transmission mechanism, which creates a feedback channel for transmitting extra information. The feedback information can be used to retransmit corrupted video data [15] or by encoder to adjust the encoding behavior, e.q., to skip the corrupted data area in the motion estimation of succeeding frames [16]. As shown in figure 1.4, the coded data of frame F_n is transmitted through data channel, and then, the decoder receives the data and finds it is corrupted and unrecoverable. Thus, the feedback information about the corrupted data is sent back to the encoder. In motion estimation of $(n+1)^{th}$ frame, it will not refer to the corrupted area of n^{th} frame. With multiple reference frames, the encoder can simply skip frame F_n as the reference frames for succeeding motion estimation.

By an extra feedback channel, the real loss map can be easily obtained by encoder so the encoder can choose a way to prevent error propagation. However, a major drawback of ARQ is the requirement of additional communication time for transmission of feedback infomation. Thus, ARQ is not suitable for low-latency environment.

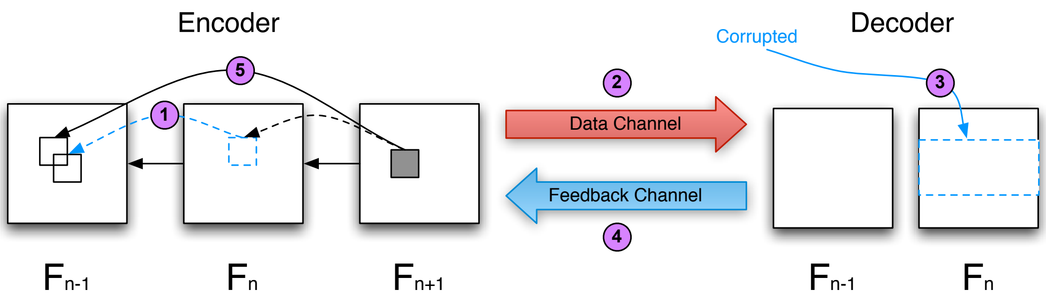

Figure 1.4 Automatic Repeat reQuest in MRF

1.4.2 Periodic Macroblock

In [13], a reference frame selection method is developed, which for every K^{th} frame, selects n macroblocks, called *Periodic Macroblock*, to predict from the frame that is K frames away, as shown in figure 1.5. For other macroblocks, only the immediately previous frame is used as reference.

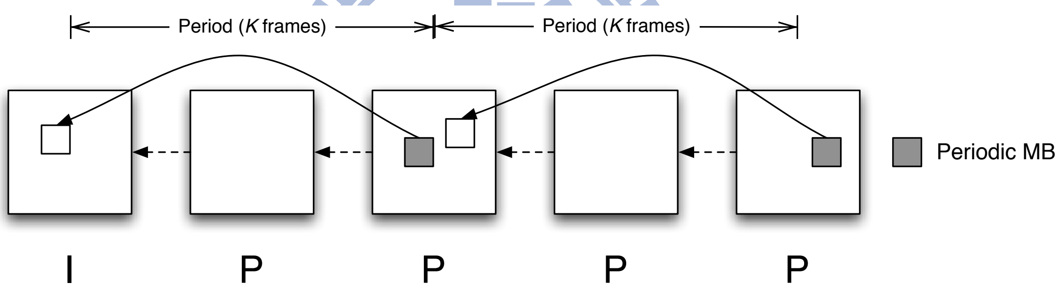

Figure 1.5 Periodic MB with K period

The authors have shown that, by using long-term frames for motion compensated prediction, these Periodic MBs can suppress te error propagation and increase the receipt probability. The selection criteria of Periodic MBs is based on the expected end-to-end distortion. However, the choice of period K and the number of Periodic MBs, n, are predefined constant in their algorithm. This makes their approach hard to adapt to various content characteristics and channel conditions.

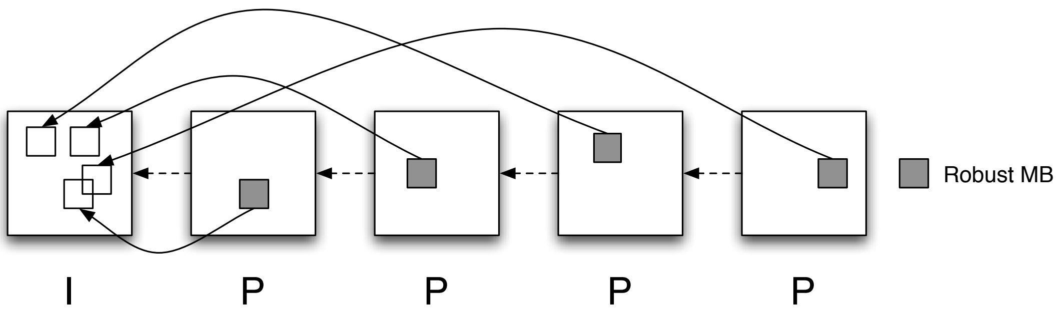

Figure 1.6 Robust MB

1.4.3 Robust Macroblock

Alternatively, robust macroblocks proposed in [14] is another reference frame selection method. As shown in figure 1.6, every P-frame contains a number of macroblocks called *Robust Macroblock*, which predict from the nearest intra frame. The selection criteria of Robust MBs is similar to that of Periodic MB, which is also based on the expected end-to-end distortion. With long-term reference to the intra frame, the authors in [14] have shown that the Robust MBs can get good suppression of error propagation. However, the number of Robust MBs in every P-frame is a predefined constant, which is independent of content characteristics and channel conditions. This makes their approach not adaptive to various video contents and channel conditions.

In this thesis, a MRF-based error resilient scheme is proposed, which includes the nearest dominant frame as one of the reference frames and adopts error resilient RDO on the stages of motion estimation and mode decision for optimal coding parameter selection. The experimental results show that proposed scheme gets a good coding efficiency over different error-prone environments. Besides, since MRF will increase the computational complexity of motion estimation, a fast motion estimation algorithm based on the proposed error-resilient MRF scheme is proposed. The fast algorithm skips unnecessary reference frames according to two key reference frames (the short-term frame and the nearest dominant frame), which determine the dominant factor between source distortion and error propagated distortion. The experimental results show that the proposed fast algorithm can speed up motion estimation of MRF without losing any performance.

The rest of the thesis is orginized as follows. In chapter 2, we describe RDO-based ER tools proposed in recent years. In chapter 3, we summarize the advantages and disadvantages of existing RDO-based and MRF-based methods, and present some observations from experimental results that give motivation to this work. In chapter 4 and chapter 5, we describe the proposed methods in detail. The experimental results shown in chapter 6 demonstrate that our proposed scheme has substantial improvement over existing schemes in providing error resilience using MRF. And finally chapter 7 concludes this work.

Chapter 2

Related Works

In chapter 1, the reasons causing error propagation and three categories of ER tools have been introduced. There are many features in these three categories of ER tools, especially content-adaptibility, network-adaptibility and better coding efficiency over error-prone environment in RDO-based ER tools. Since ERB-RDO scheme uses the average distortion from K random loss map as expected distortion, it comes with a potential drawback, high computational complexity when K is large. Thus, the proposed method chooses end-to-end distortion based RDO scheme to be based upon. In this chapter, we introduce details of the end-to-end distortion based RDO scheme.

2.1 Rate-Distortion Optimization in H.264 Standard

H.264 standard provides a Lagrangian method which optimizes the tradeoff between video quality and bit rate to determine coding parameters. The Lagrangian method is applied in two stages, Motion Estimation and Mode Decision. In the stage of motion estimation, the main concern is to determine the best MV for a certain reference frame; while in the stage of mode decision, the main concern is to decide the best coding mode and reference frame. The Lagrangian formulation for these two stages are written as follows.

- Motion Estimation:

$$J(mv) = D_{enc} + \lambda_{motion} R(mv) \tag{2.1}$$

  where D_{enc}, denotes the block-level prediction error between the current and the reference blocks. It is usually measured as SSD or SAD; $R(mv)$ is the estimate bit rate for specified motion vector; and λ_{motion} is the Lagrange multiplier to control the weight of the bit rate cost.

- Mode Decision:

$$J(mode) = D_{enc} + \lambda_{mode} R(mode) \tag{2.2}$$

  where D_{enc} denotes the macroblock-level difference between the reconstructed MB and the reference one. It is usually measured as SSD; λ_{mode} is the Lagrange multiplier for mode decision; and $R(mode)$ denotes the estimated coding rate for specified mode (reference frame, coding mode, residue, etc.).

The Lagrange multipliers for motion estimation and mode decision can be represented as

$$\lambda_{mode} = {\lambda_{motion}}^2 \tag{2.3}$$

2.1.1 Expected End-to-End Distortion Model in MRF

Commonly, the expected end-to-end distortion (the overall distortion) is defined using SAD or SSD. That is

$$d_n^i = E\{|f_n^i - \tilde{f}_n^i|\} \tag{2.4}$$

, or

$$d_n^i = E\{(f_n^i - \tilde{f}_n^i)^2\} \tag{2.5}$$

where f_n^i and $\tilde{f}_n^i$ denote the original value and the decoder reconstructed value, respectively, for pixel i in frame n. The distortion is measured as the expected difference between the pixels in encoder and the pixels in decoder. In order to effectively calculate the overall distortion, the decorder reconstructed value $\tilde{f}_n^i$ which is unkonwn in the encoder needs to be derived further.

The authors in [12] have derived $\tilde{f}_n^i$ in a way such that d_n^i can recursively calculated at the encoder. We summarize their approach here. Let $\hat{f}_n^i$ and $\hat{r}_n^i$ be the reconstructed value and the reconstructed residue in the encoder, respectively. With a motion vector mv predicted from reference frame ref, $\hat{f}_n^i$ can be represented as $\hat{f}_n^i = \hat{f}_{ref}^{i+mv} + \hat{r}_n^i$. Suppose the transmission error rate is known as p and frame copy is adopted as the error concealment policy. When current pixel is lost during transmission, it copies from the same position in previous frame $n-1$. Then, The decoder reconstructed value $\tilde{f}_n^i$ can be represented as

$$\tilde{f}_n^i = \begin{cases} \tilde{f}_{ref}^{i+mv} + \hat{r}_n^i & , 1-p \\ \tilde{f}_{n-1}^i & , p \end{cases} \tag{2.6}$$

Hence, in [12] the expected end-to-end distortion d_n^i for inter-coded pixel i in frame n

was derived to be

$$\begin{aligned}
d_n^i &= E\{(f_n^i - \tilde{f}_n^i)^2\} \\
&= (1-p)E\{(f_n^i - (\tilde{f}_{ref}^{i+mv} + \hat{r}_n^i))^2\} + pE\{(f_n^i - \tilde{f}_{n-1}^i)^2\} \\
&= (1-p)E\{(f_n^i - \hat{f}_n^i + \hat{f}_n^i - \tilde{f}_{ref}^{i+mv} - \hat{r}_n^i)^2\} + pE\{(f_n^i - \tilde{f}_{n-1}^i)^2\} \\
&= (1-p)E\{(f_n^i - \hat{f}_n^i + \hat{f}_{ref}^{i+mv} - \tilde{f}_{ref}^{i+mv})^2\} + pE\{(f_n^i - \tilde{f}_{n-1}^i)^2\} \\
&= (1-p)(E\{(f_n^i - \hat{f}_n^i)^2\} + E\{(\hat{f}_{ref}^{i+mv} - \tilde{f}_{ref}^{i+mv})^2\}) + pE\{(f_n^i - \tilde{f}_{n-1}^i)^2\} \\
&= (1-p)({d_s}_n^i + {d_{ep}}_{ref}^{i+mv}) + p{d_{ec}}_n^i \qquad (2.7) \\
&= (1-p)({d_s}_n^i + {d_{ep}}_{ref}^{i+mv}) + p(E\{(f_n^i - \hat{f}_{n-1}^i)^2\} + {d_{ep}}_{n-1}^i) \qquad (2.8)
\end{aligned}$$

where d_s denotes the source distortion, d_{ep} denotes the error-propagated distortion and d_{ec} denotes the error-concealment distortion. Since ${d_s}_n^i$ and the first part of ${d_{ec}}_n^i$ are known and can be calculated at the encoder, the estimation of d_n^i in the encoder mainly relies on the calculation of ${d_{ep}}_{ref}^{i+mv}$ and ${d_{ep}}_{n-1}^i$.

Note that ${d_{ep}}_{ref}^{i+mv}$ and ${d_{ep}}_{n-1}^i$ are in the similar style, thus we derive the generalized formula ${d_{ep}}_n^i$ as

$$\begin{aligned}
{d_{ep}}_n^i &= E\{(\hat{f}_n^i - \tilde{f}_n^i)^2\} \\
&= (1-p)E\{(\hat{f}_n^i - (\tilde{f}_{ref}^{i+mv} + \hat{r}_n^i))^2\} + pE\{(\hat{f}_n^i - \tilde{f}_{n-1}^i)^2\} \\
&= (1-p)E\{(\hat{f}_{ref}^{i+mv} - \tilde{f}_{ref}^{i+mv})^2\} + p(E\{(\hat{f}_n^i - \hat{f}_{n-1}^i)^2\} + E\{(\hat{f}_{n-1}^i - \tilde{f}_{n-1}^i)^2\}) \qquad (2.9) \\
&= (1-p){d_{ep}}_{ref}^{i+mv} + p(E\{(\hat{f}_n^i - \hat{f}_{n-1}^i)^2\} + {d_{ep}}_{n-1}^i) \qquad (2.10)
\end{aligned}$$

From (2.10), it is observed that the error-propagated distortion from current frame (i.e., ${d_{ep}}_n^i$) can be recursively calculated by the error-propagated distortion values from previous frames (i.e., ${d_{ep}}_{n-1}^i$ and ${d_{ep}}_{ref}^{i+mv}$). Since the three distortion items d_s, d_{ep} and d_{ec} can be calculated directly or recursively, the expected end-to-end distortion can be estimated at the encoder side.

Chapter 3

Motivation

In chapter 2, we simply described that R-D Optimization in H.264 standard reveals a good Rate-Distortion tradeoff in error-free environment. Some RDO-based ER tools [10–12] has been proposed to modify the distortion estimation model in order to apply the R-D optimization in error-prone environment. These RDO-based ER tools improve error resilience by taking into account the channel conditions and error propagations for optimizing R-D tradeoff in two stages, motion vector determination and mode decision. However, most of them focus on single reference frame only. On the other hand, MRF-based ER tools such as Periodic MB [13] and Robust MB [14] have shown that long-term reference frames can be used to suppress error propagation and improve error resilience. Therefore, reference frame selection plays an important role in the error resilient motion estimation when MRF is adopted.

By combining RDO-based ER tools and MRF techniques, we have conducted some experiments in error-prone environment and the results are shown in figure 3.1, where GOP size is 30, quantization parameter is 28, the number of reference frames is set to 29 (that is, all the frames in the same GOP can be selected as reference frames), and the RDO-based ER tool in [12] is extended to MRF and adopted in both MV and mode decisions. The x-axis of figure 3.1 denotes the reference index. For current frame n, the reference index i ($1 \leq i \leq 29$) means that frame $(n - i)$ is selected as the reference frame and reference index NIF means that the nearest intra frame is used as the reference frame. Figure (a) and (b) show the percentage of references for every reference index for Foreman and Football sequences, respectively. The results are calculated from all the 4×4 blocks of all the frames in the corresponding sequence and three different packet loss rates (1%, 5%, 10%) are adopted.

From figure 3.1, it is interesting to observe that

- The percentage curves are varying for different packet loss rates and for different video sequences.
- About 70% to 80% of blocks reference to the immediately previous frame (i.e., reference index = 1) and the nearest intra frame (NIF, i.e., reference index = -1).
- As the packet loss rate increases, the percentage of blocks that select NIF as the reference frames also increases. This implies that although selecting NIF may increase coding bit rate, it is still beneficial to use it to reduce the impact of error propagation. Such reduction is substantial, especially when the packet loss rate is high.

From the observations above, we propose that the nearest intra frame should be considered as one of the reference frames to alleviate the error propagation. Thus, the proposed method combines the error resilient RDO scheme and the reference frame selection with the nearest intra frame to provide a content-adaptive, network-adaptive and high-coding-efficiency scheme in error-prone environment.

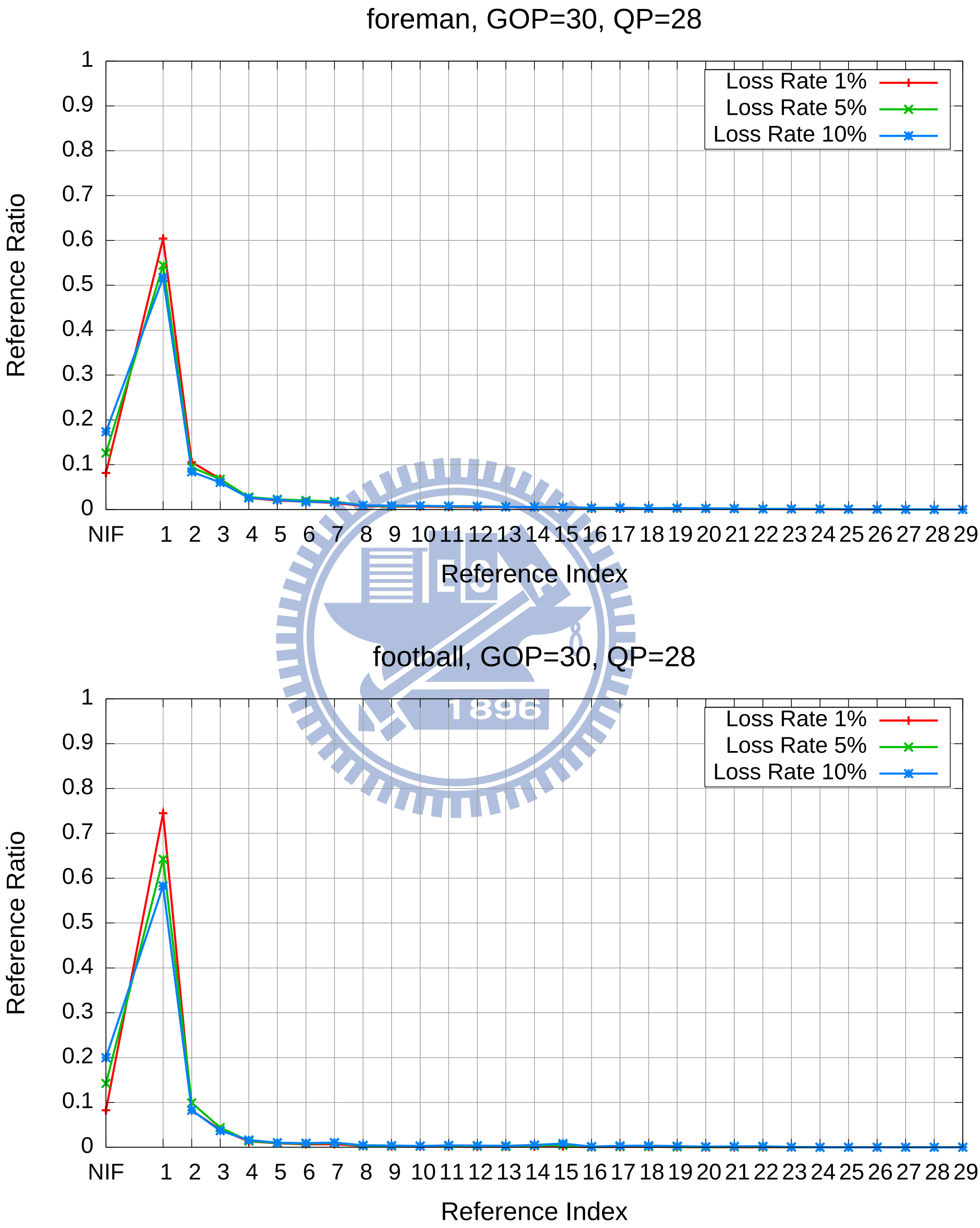

Figure 3.1 Reference distribution: (a) Foreman (b) Football

Chapter 4

Proposed Method

In this chapter, we describe the proposed error resilient method in detail. As mentioned in chapter 3, a better error resilient scheme should consider content-adaptibility, network-adaptibility and coding efficiency in error-prone environment. To achieve these goals, we propose a novel MRF-based error resilient scheme involving Rate-Distortion optimization on motion vector determination, mode decision and reference frame selection. Besides, to reduce the computational complexity caused by motion estimation in MRF, we propose a fast motion estimation algorithm.

4.1 Candidate Reference Frames

In traditional MRF, current frame uses previous K frames as candidate reference frames to predict from as shown in figure 4.1, where $K = 3$. However, since these frames are located closely in the sequence, they have similar characteristics, in terms of error propagation length. Predicting from one of them has similar error resilience. Reference to intra-coded frame, however, can suppress the error propagation. According to 3.1, as the packet loss rate increases, the percentage of the blocks choosed to predict from the intra-coded frame also increases. Thus, we propose to include the nearest intra frame (called NIF) as one of the candidate reference frames. In order to keep the same number of reference frames as usual, the original farest candidate reference frame is excluded. As shown in figure 4.1, the candidate reference frames of F_n will include F_{n-1}, F_{n-2}, and NIF when $K = 3$.

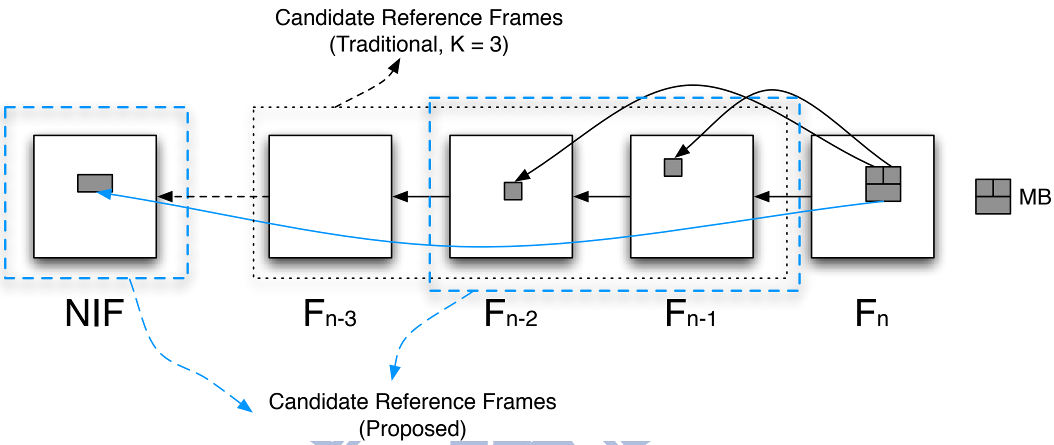

Figure 4.1 Multiple Reference Frames with Nearest Intra Frame

4.2 Error Resilient RDO

In RDO-based ER tools, original Lagrangian minimization formulation is modified as

$$J_{ER} = E\{D\} + \lambda_{ER}(o)R(o) \tag{4.1}$$

to get good coding efficiency over error-prone environment. The formulation can be applied to motion estimation and mode decision to get the best selection of coding options such as motion vector, coding mode, reference frame, etc. Therefore, the Lagrangian minimization formulation can be described as the Lagrange cost function problem.

4.2.1 Lagrange Cost Function in Error-Prone Environment

Let $O(o)$ denote the set of all defined coding options. To get good coding efficiency in error-prone environment, a general cost function for Rate-Distortion optimization can be represented as

$$J_{ER}(o) = E\{D(o)\} + \lambda_{ER}(o)R(o) \tag{4.2}$$

$$o^* = \arg\min_{o \in O}(E\{D(o)\} + \lambda_{ER}(o)R(o)) \tag{4.3}$$

where o denotes the coding option such as motion vector and reference frame in motion estimation or coding mode in mode decision, $E\{D(o)\}$ denotes the overall expected distortion for specified coding option o, $\lambda_{ER}(o)$ denotes the Lagrange multiplier for error-prone environment and $R(o)$ denotes the coding rate for specified coding option o. There are two parts $E\{D(o)\}$ and $\lambda_{ER}(o)$ need to be derived to get the optimal coding option o^*.

According to [12], λ_{ER} is equal to $(1-p)\lambda$, where p denotes the transmission error rate and λ denotes the Lagrange multiplier in error-free environment. As described in chapter 2, the overall end-to-end distortion can be seperated as

$$E\{D\} = (1-p)(E\{D_s\} + E\{D_{ep}\}) + pE\{D_{ec}\} \tag{4.4}$$

where D_s, D_{ep} and D_{ec} denote the source distortion, error-propagated distortion and error-concealment distortion.

According to (4.4), the optimal coding option o^* (4.3) can be selected with

$$\begin{aligned} o^* &= \arg\min_{o \in O}(E\{D(o)\} + \lambda_{ER}(o)R(o)) \\ &= \arg\min_{o \in O}((1-p)(E\{D_s(o)\} + E\{D_{ep}(o)\}) + pE\{D_{ec}(o)\} + (1-p)\lambda(o)R(o)) \end{aligned} \tag{4.5}$$

Since D_{ec} is independent of coding option o, it is unnecessary to be calculated. Therefore, (4.5) can be further derived as

$$
\begin{aligned}
o^* &= \arg\min_{o \in O}((1-p)(E\{D_s(o)\} + E\{D_{ep}(o)\}) + pE\{D_{ec}(o)\} + (1-p)\lambda(o)R(o)) \\
&= \arg\min_{o \in O}((1-p)(E\{D_s(o)\} + E\{D_{ep}(o)\}) + (1-p)\lambda(o)R(o)) \\
&= \arg\min_{o \in O}(E\{D_s(o)\} + E\{D_{ep}(o)\} + \lambda(o)R(o)) \qquad (4.6)
\end{aligned}
$$

The proposed RDO method applies (4.6) to both motion estimation and mode decision, which are represented as follows.

- Motion Estimation (block-based):

$$
J_{ER}(mv, ref) = E\{D_s(mv, ref)\} + E\{D_{ep}(mv, ref)\} + \lambda_{motion}(R(mv) + R(ref)) \qquad (4.7)
$$

$$
(mv, ref)^* = \arg\min_{mv, ref}(E\{D_s(mv, ref)\} + E\{D_{ep}(mv, ref)\} + \lambda_{motion}(R(mv) + R(ref))) \qquad (4.8)
$$

- Mode Decision (macroblock-based):

$$
J_{ER}(mode) = E\{D_s(mode)\} + E\{D_{ep}(mode)\} + \lambda_{mode}(R(mode)) \qquad (4.9)
$$

$$
mode^* = \arg\min_{mode}(E\{D_s(mode)\} + E\{D_{ep}(mode)\} + \lambda_{mode}R(mode)) \qquad (4.10)
$$

4.2.2 End-to-End Distortion Estimation

Since the Lagrange cost function has been defined, the last and most important task to do is to accurately estimate the end-to-end distortion. However, end-to-end distortion is, actually, hard to estimate due to the uncertainty of real channel loss map during transmission. Y. Zhang's distortion estimation model in [12] has been recognized as an effective end-to-end distortion estimation model in MRF. Therefore, the proposed model is based on Y. Zhang's model. Since Y. Zhang et. al applied the distortion estimation for mode decision only, they meansured the distortion using SSD. In this thesis, the distortion estimation will be applied for both motion estimation stage and mode decision stage. In order to reduce the computational complexity in motion estimation, the distortion is measured using SAD (i.e., formula (2.4)) Besides, previous ROPE-based models in [10–12] didn't consider spatial error concealment in slice mode. The proposed model further proposes a more accurate end-to-end distortion estimation model in slice mode.

The proposed model is applied with the following assumptions:

- The video is transmitted over a packet-loss channel.
- The packet loss rate is available at the encoder.
- Slice mode without data patitions, which means a slice is encapsulated in a packet.
- Error concealment policy is Frame Copy, which means that, when a packet is lost, the encoder simply copies the macroblock at the same location from the previous decoded frame.

Since H.264 allows intra-coded blocks in inter frames, these intra-coded blocks are more robust against channel error than inter-coded blocks. However, the distortion estimation model in [12] only considers inter-coded pixels, which may cause overestimation. In the proposed model, we consider both intra-coded and inter-coded pixels for more accurate end-to-end distortion estimation.

Before the derivation of the proposed end-to-end distortion model, we define some notations to be used. Let f_n^i, $\hat{f}_n^i$ and $\tilde{f}_n^i$ be the original value, encoder reconstructed value and decoder reconstructed value, respectively. For intra-coded pixel i in frame n, both encoder and decoder have the same reconstructed value, i.e., $\tilde{f}_n^i = \hat{f}_n^i$, if this pixel is received in the decoder. For inter-coded pixel i in frame n which predicts from pixel $i + mv$ in frame ref with motion vector mv, let $\hat{r}_n^i$ be the reconstructed residue in the encoder, i.e., $\hat{f}_n^i = \hat{f}_{ref}^{i+mv} + \hat{r}_n^i$.

When the current pixel i in frame n is lost in the decoder, it copies from the same location in frame $n-1$. This is applied to both intra-coded and inter-coded pixels. Suppose the packet loss rate is p, the decoder reconstructed value $\tilde{f}_n^i$ can be represented as

- intra-coded pixel

$$\tilde{f}_n^i = \begin{cases} \hat{f}_n^i & , 1-p \\ \tilde{f}_{n-1}^i & , p \end{cases} \tag{4.11}$$

- inter-coded pixel

$$\tilde{f}_n^i = \begin{cases} \tilde{f}_{ref}^{i+mv} + \hat{r}_n^i & , 1-p \\ \tilde{f}_{n-1}^i & , p \end{cases} \tag{4.12}$$

Hence, the expected end-to-end distortion for intra-coded pixels can be estimated by

$$\begin{aligned} d_n^i &= E\{|f_n^i - \tilde{f}_n^i|\} \\ &= (1-p)E\{|f_n^i - \hat{f}_n^i|\} + pE\{|f_n^i - \tilde{f}_{n-1}^i|\} \\ &= (1-p){d_s}_n^i + p{d_{ec}}_n^i \end{aligned} \tag{4.13}$$

where ${d_s}_n^i$ denotes the source distortion and ${d_{ec}}_n^i$ denotes the error-concealment distortion.

Since ${d_{ec}}_n^i$ depends on random variable $\tilde{f}_{n-1}^i$, we further derive it as

$$
\begin{aligned}
{d_{ec}}_n^i &= E\{|f_n^i - \tilde{f}_{n-1}^i|\} \\
&= E\{|f_n^i - \hat{f}_{n-1}^i|\} + E\{|\hat{f}_{n-1}^i - \tilde{f}_{n-1}^i|\} \\
&= E\{|f_n^i - \hat{f}_{n-1}^i|\} + {d_{ep}}_{n-1}^i
\end{aligned} \tag{4.14}
$$

where ${d_{ep}}_{n-1}^i$ denotes the error propagated distortion. Since ${d_{ep}}_{n-1}^i$ depends on random variable $\tilde{f}_{n-1}^i$ which is not available at the encoder, we further derive it as

$$
\begin{aligned}
{d_{ep}}_n^i &= E\{|\hat{f}_n^i - \tilde{f}_n^i|\} \\
&= E\{|\hat{f}_n^i - ((1-p)\hat{f}_n^i + p\tilde{f}_{n-1}^i)|\} \\
&= pE\{|\hat{f}_n^i - \tilde{f}_{n-1}^i|\} \\
&= p(E\{|\hat{f}_n^i - \hat{f}_{n-1}^i|\} + E\{|\hat{f}_{n-1}^i - \tilde{f}_{n-1}^i|\}) \\
&= p(E\{|\hat{f}_n^i - \hat{f}_{n-1}^i|\} + {d_{ep}}_{n-1}^i)
\end{aligned} \tag{4.15}
$$

Since the first item of formula (4.15) is available at the encoder, the calculation of ${d_{ep}}_n^i$ only depends on the availability of ${d_{ep}}_{n-1}^i$. Therefore, ${d_{ep}}_n^i$ can be recursively calculated frame by frame, where the d_{ep} of the first frame will be discussed in section 4.2.3.

For inter-coded pixel, as described in [12], the expected end-to-end distortion is estimated by

$$
\begin{aligned}
d_n^i &= E\{|f_n^i - \tilde{f}_n^i|\} \\
&= (1-p)(E\{|f_n^i - \hat{f}_n^i|\} + E\{|\hat{f}_{ref}^{i+mv} - \tilde{f}_{ref}^{i+mv}|\}) \\
&\quad + p(E\{|f_n^i - \hat{f}_{n-1}^i|\} + E\{|\hat{f}_{n-1}^i - \tilde{f}_{n-1}^i|\}) \\
&= (1-p)({d_s}_n^i + {d_{ep}}_{ref}^{i+mv}) + p{d_{ec}}_n^i \qquad (4.16) \\
&= (1-p)({d_s}_n^i + {d_{ep}}_{ref}^{i+mv}) + p(E\{|f_n^i - \hat{f}_{n-1}^i|\} + {d_{ep}}_{n-1}^i) \qquad (4.17)
\end{aligned}
$$

where the formula (4.17) is derived from formula (4.16) by substituting ${d_{ec}}_n^i$ with formula (4.14). Since the calculation of formula (4.17) only depends on error propagated distortion items (all others are available at the encoder side), we further derive it as

$$ {d_{ep}}_n^i = (1-p){d_{ep}}_{ref}^{i+mv} + p(E\{|\hat{f}_n^i - \hat{f}_{n-1}^i|\} + {d_{ep}}_{n-1}^i) \tag{4.18} $$

Since ${d_{ep}}_n^i$ can be recursively calculated frame by frame, the expectation of end-to-end distortion of inter-coded pixel (i.e., d_n^i) can be obtained at the encoder side.

4.2.3 Initial Value of Expected End-to-End Distortion

ROPE-based models estimate the expected end-to-end distortion in a manner of recursively calculating accumulated distortion from previous frames. This kind of recursive way requires an initial value, which is usually determined by intra frames. In traditional ROPE-based model [10–12], a presupposition is involved to determine the initial value, which assume that each packet contains one complete compressed frame. When the packet containing the intra frame is lost, error concealment simply copies previous decoded frame as current frame. It is reasonable at low bit rate. However, with slice mode enabled, the error concealment for corrupted slices of intra frames can apply spatial interpolation to obtain a better performance. Therefore, we proposed a model to estimate the initial value in slice mode.

Support a frame is seperated to s slices and ${f_{ec}}_0^i(k)$ denotes the error concealment value with k slices are lost. With packet loss rate p, the expected end-to-end distortion is represented as

$$ d_0^i = \sum_{k=0}^{s} \binom{s}{k} p^k (1-p)^{s-k} E\{|f_0^i - {f_{ec}}_0^i(k)|\} \tag{4.19} $$

and the initial value of error-propagated distortion is

$$d_{ep_0}^{\ i} = \sum_{k=0}^{s} \binom{s}{k} p^k (1-p)^{s-k} E\{|\hat{f}_0^i - f_{ec_0}^{\ i}(k)|\} \tag{4.20}$$

Since the initial value only needs to calculated at the first frame of GOP, the frame number is marked as 0.

4.3 Adaptive DIF Replacement for Scene Change

When scene change happened, the difference between the two consecutive frames right before and after the scene cut may increase substantially, resulting in a frame with high-ratio of intra-coded MBs (marked as scene-change frame). For those frames after the scene-change frame, selecting NIF as their reference frames may suffer from large prediction error, which will reduce coding efficiency dramatically. This implies that it would be no longer beneficial to select NIF as reference because the gain from error propagation reduction may not be able to compensate the loss in the coding efficiency. On the other hand, the scene-change frame has high ratio of intra-coded MBs, which provides a certain ability to alleviate error propagation. Compared with NIF, reference to scene change frames provides a better coding efficiency for those frames after scene cut. Thus, we propose a DIF replacement mechanism which changes the candidate reference frame from the nearest intra frame to the scene-change frame. To be more general, we define a dominant intra frame (DIF), which can be either

- A nearest intra frame (NIF), or
- A nearest inter frame with high-ratio intra-coded MBs (i.e., scene-change frame)

As shown in figure 4.2, assume the number of reference frames is 2 and a scene change happened between frames F_{n-4} and F_{n-3}. Without DIF replacement mechanism, candi-

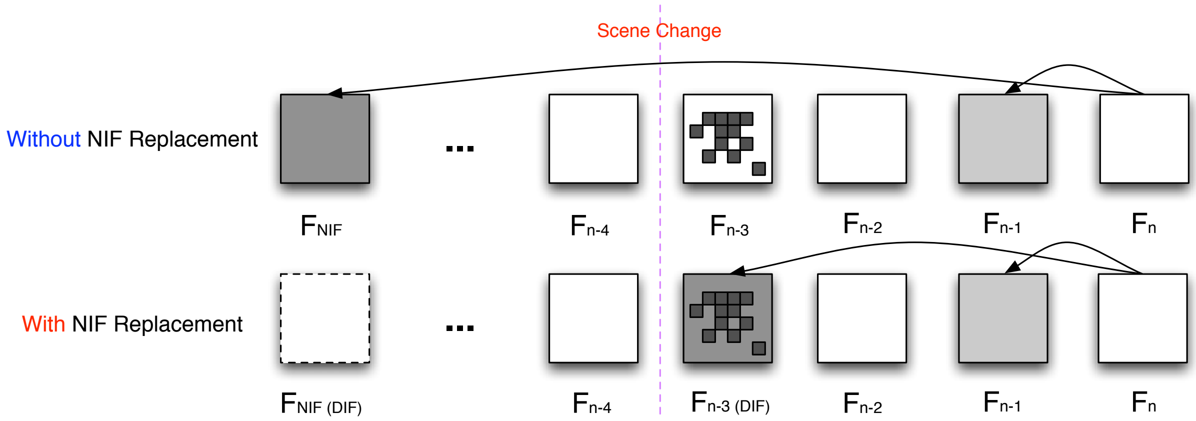

Figure 4.2 DIF Replacement Mechanism for Scene Change

date reference frames of frame F_n will be F_{n-1} and F_{NIF}. In this case, most MBs in F_n would prefer to predict from F_{n-1} because of its high coding efficiency. However, using F_{n-1} as reference frame will suffer from the error propagation from F_{n-3} and F_{n-2}. With DIF replacement mechanism, the candidate reference frames of F_n become F_{n-1} and F_{n-3} since both F_{n-3} and F_n are in the same scene, the loss in coding efficiency of using F_{n-3} as reference is not as high as that using F_{NIF}. The loss in coding efficiency has the propability to be covered by the gain of getting rid of error propagation from F_{n-3} to F_{n-2} if F_{n-3} is selected as the reference frame.

Chapter 5

Fast Motion Estimation

Multiple reference frames can be adopted to increase the coding efficiency and suppress the error propagation. However, this technique requires more computational time for mode decision and, especially, for motion estimation, which comsumes most of computational time. In general, the computational complexity depends on the number of candidate reference frames K. Therefore, we propose a fast motion estimation algorithm for the proposed error resilient MRF scheme in order to reduce the computational complexity.

5.1 Reference Trend Decision

From the experimental results in figure 3.1, it can be seen that

- more than 90% of blocks choose to reference to the first four reference frames as well as the DIF, and
- about 70% to 80% of blocks reference to the two key frames (the immediately previous frame & the DIF)

Thus, we propose to use five reference frames at most, and use two key frames first to decide the trend of prediction because of the different properties of the four previous frames and the DIF. Reference to DIF has good error resilience but poor coding efficiency, while reference to the four previous frames has good coding efficiency but poor error resilience. Assume current frame is n. With two key frames as reference, if frame $n-1$ is selected by proposed ER-RDO formula, it means the gain from coding efficiency is more important than that from error resilience. This could be due to good channel conditions (i.e., low packet loss rate) or video content with high motion. In this case, motion estimation will continue with candidate reference frames $n-2$, $n-3$ and $n-4$. On the contrary, selecting dominant intra frame as reference frame means that error resilience capability is more important than coding efficiency. In this case, candidate reference frames $n-2$, $n-3$ and $n-4$ will be skipped, and motion estimation process can be early terminated.

5.2 Fast Motion Estimation Algorithm

Figure 5.1 shows the H.264 standard ME algorithm.

Step 1. Get next available MB in current frame.

Step 2. Choose next unprocessed mode.

Step 3. Do motion estimation for each candidate reference frame $(F_{n-i}, 1 \leq i \leq 5)$ in selected mode.

Step 4. Apply H.264 standard RDO to ME for each candidate reference frame.

Step 5. If all modes for current MB are done, go to Step 6. Otherwise, go to Step 2.

Step 6. Apply H.264 standard RDO to mode decision to determine the final coding mode.

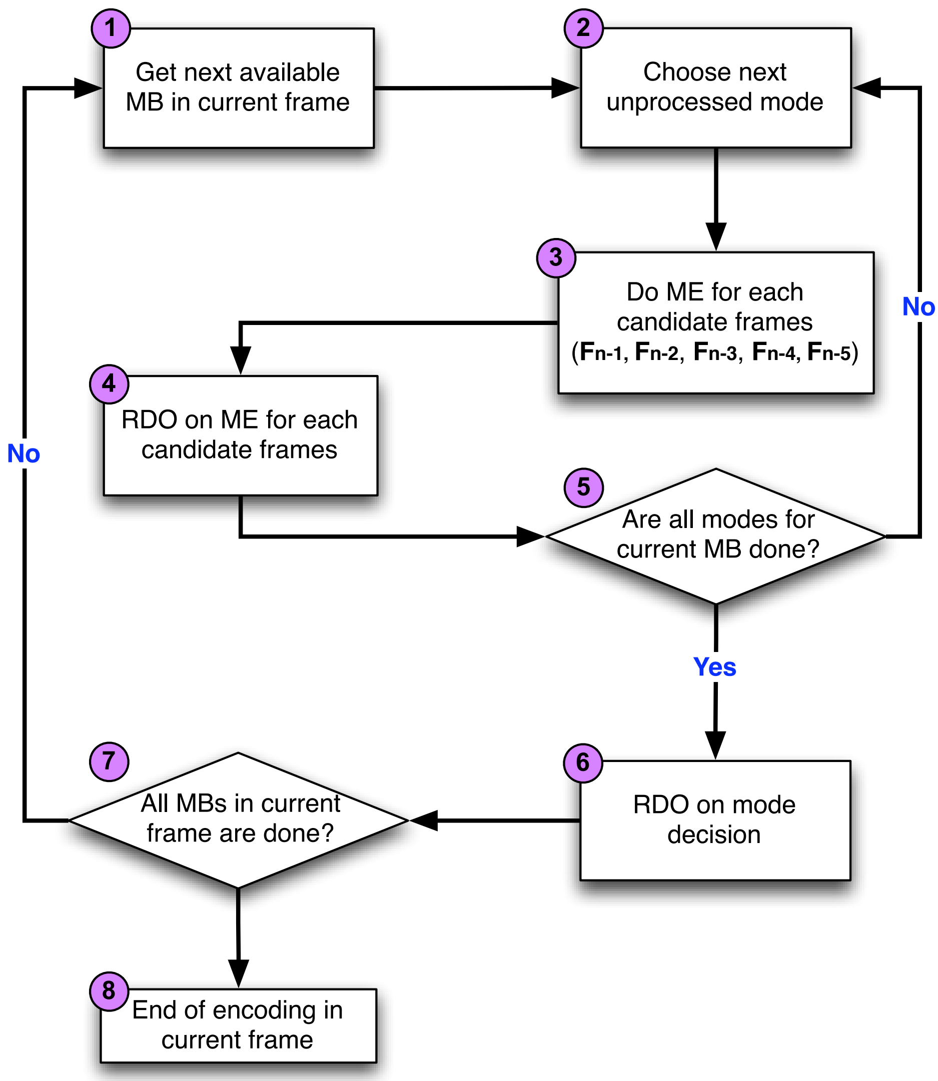

Figure 5.1 H.264 Motion Estimation flow chart

Step 7. If all MBs in current frame are done, go to Step 8. Otherwise, go to Step 1 to get next MB for encoding.

Step 8. End of encoding in current frame.

As described above, the proposed fast motion estimation algorithm changes the order of ME for two key frames as shown in figure 5.2.

Step 1. Get next available MB in current frame.

Step 2. Choose next unprocessed mode.

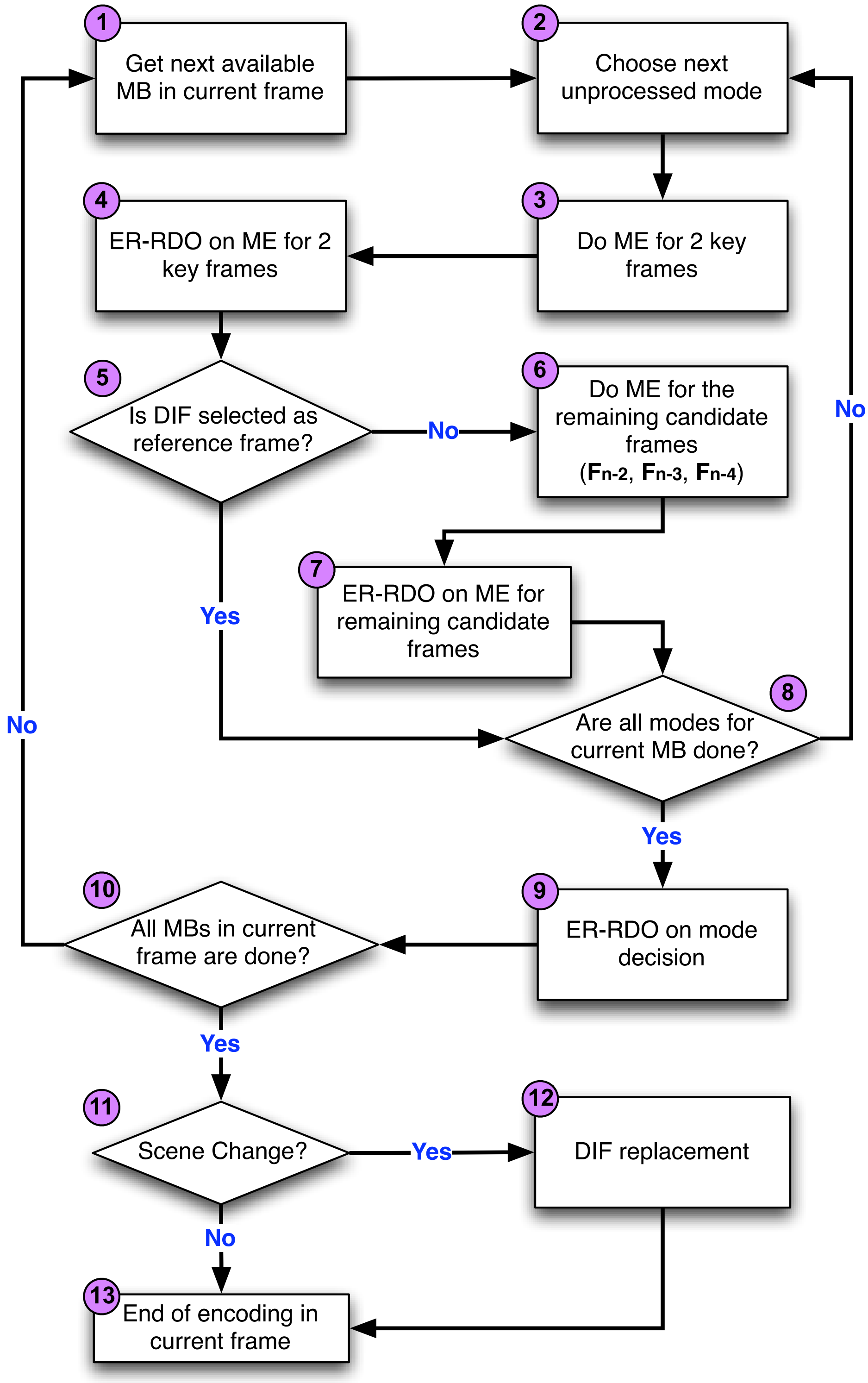

Figure 5.2 Fast Motion Estimation flow chart

Step 3. Do motion estimation for 2 key frames (F_{n-1} and F_{DIF}) in selected mode.

Step 4. Apply Error Resilient RDO to ME for 2 key frames.

Step 5. If DIF is selected as reference frame rather than the immediately previous frame, use it as best reference frame in selected mode and go to Step 8. Otherwise, go to Step 6.

Step 6. Do ME for the remaining candidate frames. (F_{n-2}, F_{n-3}, F_{n-4})

Step 7. Apply Error Resilient RDO to ME for the remaining candidate frames.

Step 8. If all modes for current MB are done, go to Step 9. Otherwise, go to Step 2.

Step 9. Apply Error Resilient RDO to mode decision to determine the final coding mode.

Step 10. If all MBs in current frame are done, go to Step 11. Otherwise, go to Step 1 to get next MB for encoding.

Step 11. Detect whether scene change happened by the ratio of intra-coded MBs in current frame. If scene change happened, go to Step 12. Otherwise, go to Step 13.

Step 12. Apply DIF replacement mechanism.

Step 13. End of encoding in current frame.

With the comparison between figure 5.1 and figure 5.2, it can be seen that the proposed fast motion estimation algorithm, which only changes the order of ME for two key frames without affecting to the ER-RDO applying to both motion estimation and mode decision stages.

5.3 Motion Vector Prediction for DIF

Since the dominant intra frame may be far from current frame, using co-located MB as the center of search window in motion estimation may not be adequate. Thus, we have designed a scheme to predict MV on the DIF. Our approach is based on FDVS (Forward Dominant Vector Selection) [17], which is a MV composition method originally designed for transcoding with frame skipping applications.

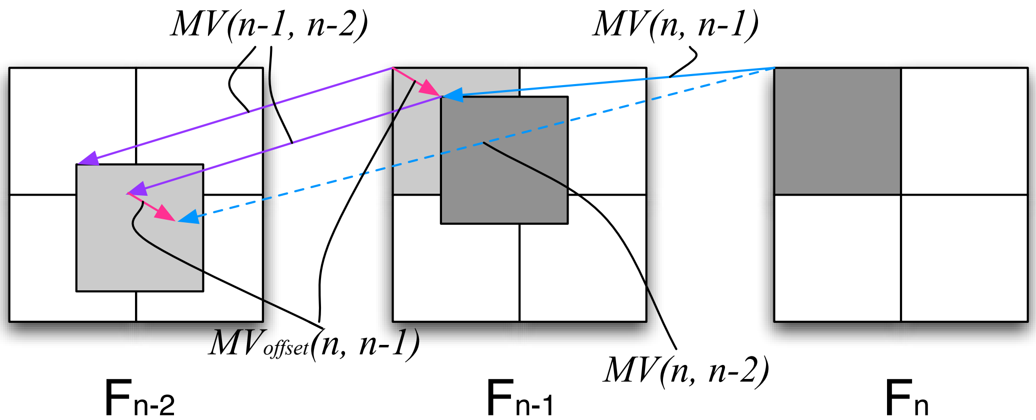

Figure 5.3 FDVS - Fast Dominant Vector Selection

Considering a simple sample as shown in figure 5.3, we have motion $MV(n, n-1)$ between frame n and frame $n-1$, and $MV(n-1, n-2)$ between frame $n-1$ and frame $n-2$, respectively. If frame $n-1$ is skipped, the most common way to represent the corresponding motion between frame n and frame $n-2$ would be $MV(n, n-1) + MV(n-1, n-2)$. However, for each MB in frame n, the area pointed by its MV may not be aligned on the MB boundary of frame $n-1$. In FDVS method, the MV associated with the MB with the largest overlapping area out of the four neighboring MBs on frame $n-1$ is selected. To compensate the shift in frame $n-1$, the authors in [18] suggested to add the offset (denoted by $MV_{offset}(n, n-1)$) to the composed MV. Therefore, the MV from frame n to frame $n-2$ can be predicted as

$$PMV(n, n-2) = MV(n, n-1) + MV(n-1, n-2) + MV_{offset}(n, n-1) \qquad (5.1)$$

This MV prediction technique can be easily extended to across more frames. Assume current frame is frame n and DIF is frame m ($m < n$). The predict MV from frames n to m can be represented as

$$PMV(n,m) = \sum_{i=m}^{n-1} [MV(i+1,i) + MV_{offset}(i+1,i)] \tag{5.2}$$

It can be seen that such MV prediction relies on all the MVs between frame n and frame m. However, all the required MVs may not be available in the encoding buffer, especially when $n-m > k$, where k is the number of reference frames. To rope with this problem, we propose a modified FDVS scheme called Accumulated Forward Dominant Vector Selection (AFDVS) by using Accumulated Motion Vector (AMV).

First of all, we define $AMV(i)$ to denote the predict MV from frame i to DIF $PMV(i, DIF)$. Assume each frame i has only one candidate reference frame, which is its immediately previous frame $i-1$. Therefore, by equation (5.1), $AMV(i)$ can be recursively derived from $AMV(i-1)$ as

$$AMV(i) = MV(i,i-1) + AMV(i-1) + MV_{offset}(i,i-1) \tag{5.3}$$

where the initial $AMV(m) = 0$ if DIF is frame m. From (5.3), it is interesting to notice that once $AMV(i)$ is obtained, $AMV(i-1)$, $MV(i,i-1)$ and $MV_{offset}(i,i-1)$ are unnecessary for calculating $AMV(i+1)$, that is, they can be removed from the encoding buffer. In other words, if AMV in reference frame (frame $n-1$) is available, the predict MV from current frame n to DIF (frame m) can be derived by

$$PMV(n,m) = MV(n,n-1) + AMV(n-1) + MV_{offset}(n,n-1) \tag{5.4}$$

To be more general, we further modify the equation (5.4) for MRF, as shown in figure 5.4. The predict MV $PMV(n,m)$ from current frame n to DIF (frame m) is represented

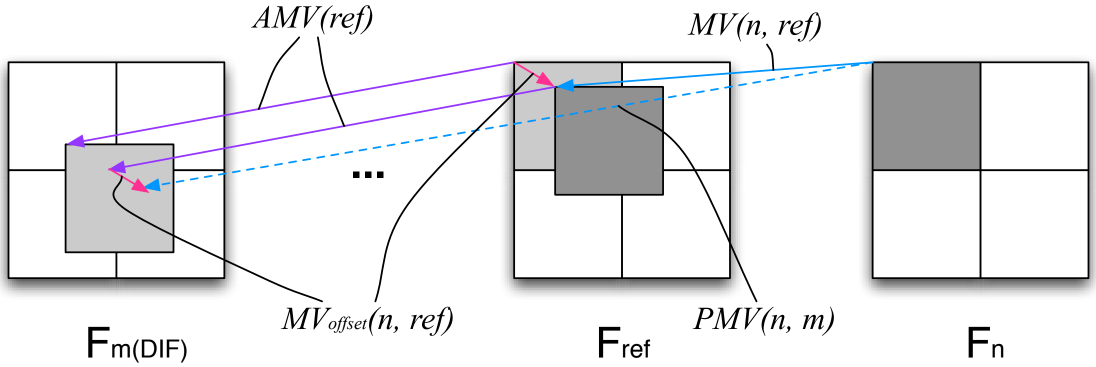

Figure 5.4 AFDVS - Accumulated Fast Dominant Vector Selection

as

$$PMV(n,m) = MV(n,ref) + AMV(ref) + MV_{offset}(n,ref) \tag{5.5}$$

where ref denotes the frame number of reference frame.

The equation (5.5) shows that the current predict MV can be calculated by tracing the reference frame. Since the MV prediction may across all candidate reference frames, even if $AMV(n)$ is obtained, $AMV(ref)$ still cannot be immediately removed from the encoding buffer.

To be more clearly, assume the number of reference frames K is 3, as shown in figure 5.5. At frame 1, it only predicts from DIF (frame 0), thus the predict MV $PMV(1,0)$ is simply equal to $MV(1,0)$. At frame 2, it may predict from previous frame 1 or from DIF, thus these two cases has to be considered to estimate the predict MV $PMV(2,0)$. If it chooses frame 0 as reference frame, the MV $MV(2,0)$ is directly set as $PMV(2,0)$, since $AMV(0)$ is 0, otherwise, it combines $MV(2,1)$ and $AMV(1)$, which is calculated at frame 1, as $PMV(2,0)$. At frames 3 to 5, there are only 3 cases needs to be considered ($K = 3$), and by using the same way at frame 2, the predict MVs $PMV(3,0)$, $PMV(4,0)$ and $PMV(5,0)$ can be easily composed. Thus, a general formula is proposed to represent the accumulated predict MV $PMV(n,0)$ from frames n to DIF

$$PMV(n,0) = \begin{cases} MV(n,0) & \text{, if select } DIF \text{ (frame 0)} \\ MV(n,n-1) + AMV(n-1) & \\ \quad +MV_{offset}(n,n-1) & \text{, if select frame } n-1 \\ MV(n,n-2) + AMV(n-2) & \\ \quad +MV_{offset}(n,n-2) & \text{, if select frame } n-2 \end{cases} \tag{5.6}$$

From equation (5.6), it can be seen that, to obtain predict MV $PMV(n,0)$ from current frame n to DIF, we only need 2 AMVs from frame $n-1$ and from frame $n-2$ when $K = 3$. That is, at most of $K-1$ AMVs need to be keep in encoding buffer.

Moreover, an accurate predict MV to DIF can also shrink the search range of motion estimation to reduce the computational complexity. In our thesis, the search range on DIF is set as 4.

5.4 Time Complexity of Fast Motion Estimation

Assume the following

- number of reference frames: k
- search range of DIF: from 32 to 4
- reference ratio of DIF (skip ME of $k-2$ reference frames): r (about 14% to 26%)

Thus, the time complexity ratio of ME is estimated as

$$r\left(\frac{2}{k}\right)\left(\left(\frac{4}{32}\right)^2\frac{1}{2}+\frac{1}{2}\right)+(1-r)\left(\left(\frac{4}{32}\right)^2\frac{1}{k}+\frac{k-1}{k}\right) \tag{5.7}$$

For 5 reference frames ($K = 5$), it performs about 28% to 35% reduction.

By applying fast motion estimation algorithm to proposed scheme, the computational complexity in motion estimation can be substantially reduced.

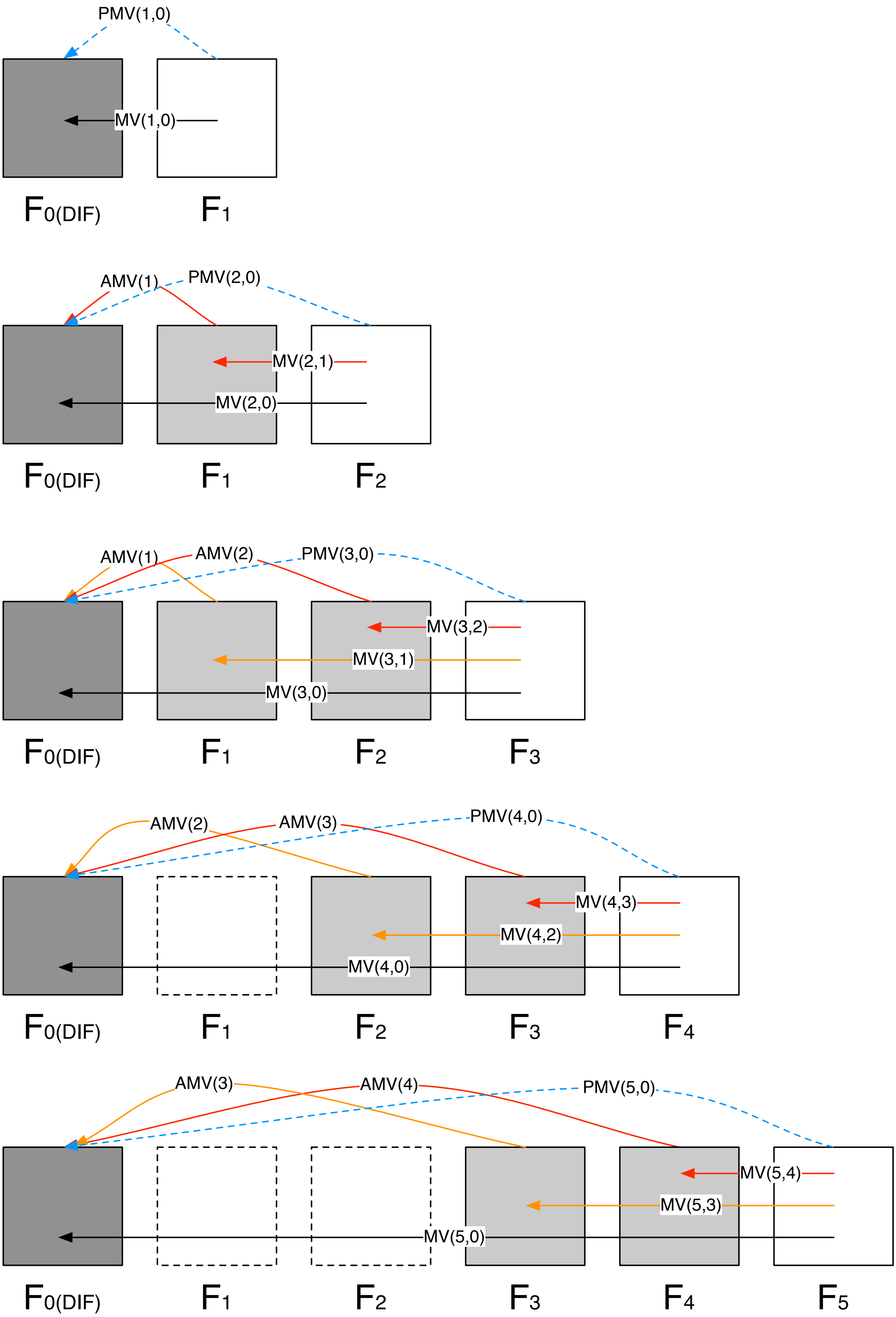

Figure 5.5 AFDVS with 3 Reference Frames

Chapter 6

Experimental Results

The proposed ER-RDO-based MRF method is integrated into the latest JVT reference software JM15.1 [19]. The parameters of our experimental environment are set as follows.

- Test sequence: *Foreman*, *Football*, *News*
- Number of frames: 100 frames
- Frame rate: 30 fps
- Structure of the Group of Picture (GOP): I P P P P ...
- GOP size: 30 frames
- Frame format: CIF (352×288 pixels)
- Number of slices per frame: 6 slices
- 100 random loss patterns for each of different packet loss rates (1%, 5%, 10%)

The methods used for comparison are listed as follows.

- **H.264**: the original RDO method in H.264 standard.

- **Periodic MB**: Periodic Macroblock proposed by J. Zheng in [13], where the period $K = 5$.
- **Robust MB**: Robust Macroblock proposed by Q. Zhang in [14], where the number of robust MBs per frame is 120.
- **ER-RDO**: The ER-RDO model in [12]. Note that, in [12], the ER-RDO is applied to mode decision stage only.
- **ER-RDO [12] + ME**: The ER-RDO model in [12] is used, but it is applied to both the mode decision stage and the motion estimation stage.
- **Proposed w/o Fast ME**: the proposed method without fast motion estimation mechanism.
- **Proposed w/o IMB**: the proposed method without end-to-end distortion estimation for intra-coded MBs.
- **Proposed**: the full version of proposed method.

6.1 Bit Rate v.s. Average PSNR

The measured average PSNR results of sequences *Foreman*, *Football*, *News* with packet loss rates 1%, 5% and 10% are shown in figure 6.1, figure 6.2, figure 6.3 and table 6.1. We can see that the proposed method is better than methods proposed in [12–14]. It is observed that for sequence *Foreman* in figure 6.1, the Robust MB scheme in [14] achieves good performance at packet loss rates 1% and 5%, however, it has about 0.7 dB lower than the proposed method at packet loss rate 10%; And for sequence *News* in figure 6.3, it can achieve good performance at packet loss rates 5% and 10%, however, it has about

0.8 dB lower than the proposed method at packet loss rate 1%. The results are due to that although the scheme in [14] choosed Robust MBs according to estimated distortion, the number of Robust MBs for each frame is a fixed constant which makes it hard to adopt to different channel conditions. On the other hand, the scheme in [14] achieved good performance for sequences *Foreman* and *News*, but it has about 2.5 dB lower than the proposed method for sequence *Football*. The results again show that a fixed number of Robust MBs per frame is not suitable for various sequences. The proposed method adopts ER-RDO at the stage of ME, which will select the best MV and reference frame for each MB, resulting in an optimal number of MBs that reference to intra-coded MBs. Therefore, the proposed method is adaptive to varying channel conditions and various video sequences.

According to the results of 'Proposed w/o IMB' and 'ER-RDO [12]', by involving DIF as one of reference frames, 'Proposed w/o IMB' can gain about 1.2 dB higher than 'ER-RDO [12]' for sequence *Foreman* and about 0.7 dB higher for sequence *News*. The reason that the gain for sequence *Football* is relatively low (less than 0.5 dB) is due to that *Football* is a high-motion sequence and, referencing to DIF will cause too much increase in the coding bits. As a consequence, a relative high ratio of MBs choose intra-coding to alleviate error propagation. This makes the performance difference between 'Proposed w/o IMB' and 'ER-RDO [12]' become small. By involving end-to-end distortion estimation for intra-coded MBs, the proposed method can still gains more than 0.7 dB in low packet loss rate, and especially, more than 2 dB in high packet loss rate.

	Loss Rate		1%					5%					10%				
	QP		30	29	28	27	26	30	29	28	27	26	30	29	28	27	26
Foreman	Proposed (5 Ref)	BR	478.54	558.38	672.85	801.63	925.52	615.96	726.59	871.34	1048.37	1226.96	741.82	876.60	1048.78	1250.61	1445.71
		PSNR	33.69	34.17	34.73	35.24	35.60	30.63	30.87	31.52	31.88	32.20	28.48	28.78	29.45	29.69	29.84
	H.264	BR	420.51	488.40	581.65	688.73	788.36	420.51	488.40	581.65	688.73	788.36	420.51	488.40	581.65	688.73	788.36
		PSNR	32.15	32.52	32.72	33.27	33.38	27.94	28.08	28.33	28.32	28.42	25.18	25.20	25.35	25.39	25.41
	Periodic MB [13] (Period 5)	BR	503.16	590.63	710.95	847.18	980.75	503.16	590.63	710.95	847.18	980.75	503.16	590.63	710.95	847.18	980.75
		PSNR	33.54	34.01	34.56	34.96	35.42	29.47	29.62	29.83	29.92	30.15	26.54	26.58	26.70	26.71	26.81
	Robust MB [14]	BR	595.12	682.82	820.31	962.12	1104.80	595.12	682.82	820.31	962.12	1104.80	595.12	682.82	820.31	962.12	1104.80
		PSNR	34.02	34.52	35.18	35.68	36.22	30.63	30.97	31.33	31.60	31.86	27.78	28.03	28.22	28.39	28.57
	ER-RDO [12] (5 Ref)	BR	459.01	534.34	642.54	758.73	878.86	555.06	650.45	777.01	926.23	1072.47	627.80	731.27	869.33	1027.60	1185.56
		PSNR	32.26	32.86	33.11	33.49	33.83	29.11	29.49	29.61	30.15	30.33	26.77	27.06	27.24	27.50	27.59
	[12] + ME (5 Ref)	BR	464.06	540.69	644.44	764.82	884.90	567.85	666.89	800.34	946.59	1090.68	652.83	762.64	908.26	1066.61	1229.08
		PSNR	32.25	32.69	32.82	33.48	33.67	29.19	29.56	29.75	30.26	30.16	26.84	26.85	27.08	27.58	27.58
Football	Proposed (5 Ref)	BR	1561.42	1742.15	1990.05	2246.20	2475.27	1743.87	1924.36	2187.41	2450.12	2678.87	1819.74	2002.47	2270.71	2541.25	2775.20
		PSNR	34.00	34.56	34.87	35.79	36.65	30.78	31.46	31.98	32.66	33.23	28.77	29.32	29.80	30.25	30.94
	H.264	BR	1413.31	1574.89	1796.35	2025.08	2216.43	1413.31	1574.89	1796.35	2025.08	2216.43	1413.31	1574.89	1796.35	2025.08	2216.43
		PSNR	32.16	32.87	33.28	33.74	33.92	26.20	26.37	26.44	26.87	26.45	23.07	23.24	23.19	23.44	23.12
	Periodic MB [13] (Period 5)	BR	1452.26	1614.98	1840.68	2073.20	2271.72	1452.26	1614.98	1840.68	2073.20	2271.72	1452.26	1614.98	1840.68	2073.20	2271.72
		PSNR	33.10	33.45	34.18	34.63	35.19	27.01	27.19	27.52	27.46	27.77	23.66	23.81	24.09	24.01	24.26
	Robust MB [14]	BR	1590.18	1751.79	1987.37	2222.81	2433.85	1590.18	1751.79	1987.37	2222.81	2433.85	1590.18	1751.79	1987.37	2222.81	2433.85
		PSNR	33.86	34.23	34.92	35.47	36.04	28.53	28.65	28.99	29.27	29.66	25.22	25.09	25.42	25.64	25.96
	ER-RDO [12] (5 Ref)	BR	1491.89	1662.10	1896.45	2137.11	2348.33	1566.64	1738.76	1978.30	2226.08	2437.10	1598.62	1769.74	2014.96	2261.93	2476.05
		PSNR	32.64	33.21	34.12	34.71	35.22	27.82	28.15	28.50	29.11	29.22	24.66	24.92	24.94	25.32	25.36
	[12] + ME (5 Ref)	BR	1497.06	1664.85	1901.07	2138.51	2349.88	1586.71	1757.63	2003.59	2247.24	2459.89	1636.38	1808.78	2056.69	2307.52	2516.06
		PSNR	32.79	33.29	34.07	34.58	35.05	27.93	28.30	28.54	28.89	29.08	25.01	25.22	25.58	25.62	25.68
News	Proposed (5 Ref)	BR	275.90	309.95	355.23	402.50	449.78	307.47	348.30	403.25	460.43	520.42	347.76	392.22	454.35	520.36	586.65
		PSNR	35.68	36.33	36.97	37.58	38.15	32.39	32.82	33.37	33.78	34.19	29.73	30.10	30.56	30.87	31.24
	H.264	BR	266.02	298.40	340.54	385.44	428.55	266.02	298.40	340.54	385.44	428.55	266.02	298.40	340.54	385.44	428.55
		PSNR	35.13	35.82	36.41	36.94	37.42	30.70	31.12	31.35	31.52	31.79	27.67	27.91	28.04	28.12	28.25
	Periodic MB [13] (Period 5)	BR	291.02	326.38	372.74	419.35	465.54	291.02	326.38	372.74	419.35	465.54	291.02	326.38	372.74	419.35	465.54
		PSNR	35.78	36.36	37.04	37.61	38.21	32.02	32.33	32.79	33.11	33.41	28.97	29.10	29.42	29.64	29.84
	Robust MB [14]	BR	338.89	379.21	430.13	484.33	539.83	338.89	379.21	430.13	484.33	539.83	338.89	379.21	430.13	484.33	539.83
		PSNR	35.77	36.43	37.04	37.66	38.21	32.56	33.04	33.44	33.82	34.18	29.74	30.07	30.28	30.56	30.85
	ER-RDO [12] (5 Ref)	BR	271.43	304.92	350.41	397.48	442.27	291.01	327.16	377.02	430.54	483.54	309.66	347.49	401.18	458.72	513.13
		PSNR	35.38	35.93	36.64	37.22	37.59	31.45	31.82	32.30	32.62	32.96	28.58	28.97	29.27	29.59	29.54
	[12] + ME (5 Ref)	BR	271.75	305.00	349.76	396.54	442.50	293.81	330.49	380.03	432.59	483.60	314.31	354.39	407.67	462.49	514.38
		PSNR	35.28	35.99	36.66	37.17	37.57	31.48	31.94	32.28	32.74	33.09	28.63	29.07	29.30	29.63	29.66

Table 6.1 Table of Bit Rate v.s. Avg. PSNR in different packet loss rates $p = 0.01, 0.05, 0.10$ and $QP = 26, 27, 28, 29, 30$

Sequence	Method	Percentage of intra-coded MBs		
		$p = 1\%$	$p = 5\%$	$p = 10\%$
Foreman	Proposed (5 Ref)	10%	22%	32%
	Proposed w/o IMB (5 Ref)	8%	16%	20%
Football	Proposed (5 Ref)	53%	72%	80%
	Proposed w/o IMB (5 Ref)	42%	49%	53%
News	Proposed (5 Ref)	3%	6%	8%
	Proposed w/o IMB (5 Ref)	3%	4%	5%

Table 6.2 Intra-coded MB Rates with different packet loss rates $p = 0.01, 0.05, 0.10$ and $QP = 28$ for sequences *Foreman*, *Football* and *News*

6.2 The Effects of Intra-coded MBs in End-to-End Distortion Estimation

As described in chapter 4, our proposed method considers intra-coded MBs in end-to-end distortion estimation to get accurate expectation of end-to-end distortion. Therefore, in our experiment, we compare methods 'Proposed' with 'Proposed w/o IMB'. Table 6.2 shows the intra-coded MB rate for different sequences and different packet loss rates with $QP = 28$. It can be seen that, the inclusion of intra-coded MBs for end-to-end distortion causes increase of the intra-coded MB rate. That is because, actually, the intra-coded MBs in inter frames may cause error propagation if it is not received. Without considering it, the end-to-end distortion estimation may underestimate the impact of error propagation from intra-coded MBs, especially for high-motion equences and high packet loss rate, since they usually reveals in high intra rate. Figure 6.4 shows the average PSNR frame by frame with packet loss rate 10% and $QP = 28$ for different sequences *Foreman*, *Football* and *News*. Since the intra-coded MB rate has a big gap (27%) between the methods with and without consideration of intra-coded MB in end-to-end distortion estimation for sequence *Football* with packet loss rate 10%, the degradation of average PSNR is high.

6.3 Performance of Fast Motion Estimation

The proposed fast motion estimation algorithm (FME) involves AFDVS to predict MVs on DIF rather than the way using co-located MBs. Moreover, two key frames (the DIF and the immediately previous frame) are used to determine the trend of prediction. Table 6.3 shows the reference ratios referencing to DIF and the four previous frames. From the table, we can see that, by involving fast motion estimation algorithm, the percentage of reference to the DIF increases about 4.2%, 1.2% and 0.3% at low packet loss rate for *Foreman*, *Football* and *News*, respectively. At high packet loss rate, the difference of reference ratio is about 6.1%, 1.8% and 0.6% for *Foreman*, *Football* and *News*, respectively. Those blocks which change reference frame to DIF may provide more resilience against error propagation. However, referencing to the DIF may require more coding bits. From the performance comparison, as shown in figure 6.5, the performance of proposed method with FME is better than the one without FME for *News* and, especially, for *Foreman*, however, it's worse for *Football*. The main reason is that, for low intra rate sequences such as *Foreman* and *News*, the gain of referencing to the DIF is higher enough to counterbalance against the increase of coding bits. The effect is obvious at high packet loss rate. However, for high-motion sequences such as *Football*, the gain of referencing to the DIF is quite small due to high intra rate. Futhermore, the coding bits by referencing to the DIF is much higher for high-motion sequences than the one for low-motion sequences. That is the reason that, for *Football*, the proposed method with FME gains lower than the method without FME. Even though fast motion estimation causes lower gain for high-motion sequences, it still costs low computational complexity, which is described in chapter 5.

Sequence	Method	Reference Frame	Percentage of Reference		
			$p = 1\%$	$p = 5\%$	$p = 10\%$
Foreman	Proposed (5 Ref)	DIF	14.2%	21.4%	28.1%
		Others	85.8%	78.6%	71.9%
	Proposed w/o Fast ME (5 Ref)	DIF	10.0%	16.4%	22.0%
		Others	90.0%	83.6%	78.0%
Football	Proposed (5 Ref)	DIF	10.3%	19.2%	27.0%
		Others	89.7%	80.8%	73.0%
	Proposed w/o Fast ME (5 Ref)	DIF	9.1%	16.4%	25.2%
		Others	90.8%	83.6%	74.8%
News	Proposed (5 Ref)	DIF	5.3%	5.7%	6.2%
		Others	94.7%	94.3%	93.8%
	Proposed w/o Fast ME (5 Ref)	DIF	5.0%	5.2%	5.6%
		Others	95.0%	94.8%	94.4%

Table 6.3 Reference Ratios with different packet loss rates $p = 0.01, 0.05, 0.10$ and $QP = 28$ for sequences *Foreman*, *Football* and *News*

6.4 Performance of DIF Replacement Mechanism

In chapter 4, we propose a DIF replacement mechanism which adaptively changes the dominant intra frame when scene change happened. In the experiment, we cascade sequences *Foreman* and *Stefan* as a composite sequence of 100 frames with scene changes at frame 34 and frame 63. Figure 6.6 shows the performance comparison of the proposed methods with and without DIF replacement mechanism. According to the results, we can see that, by involving DIF replacement mechanism, the performance gain is about 0.1 dB and 0.3 dB for packet loss rates low and high, respectively.

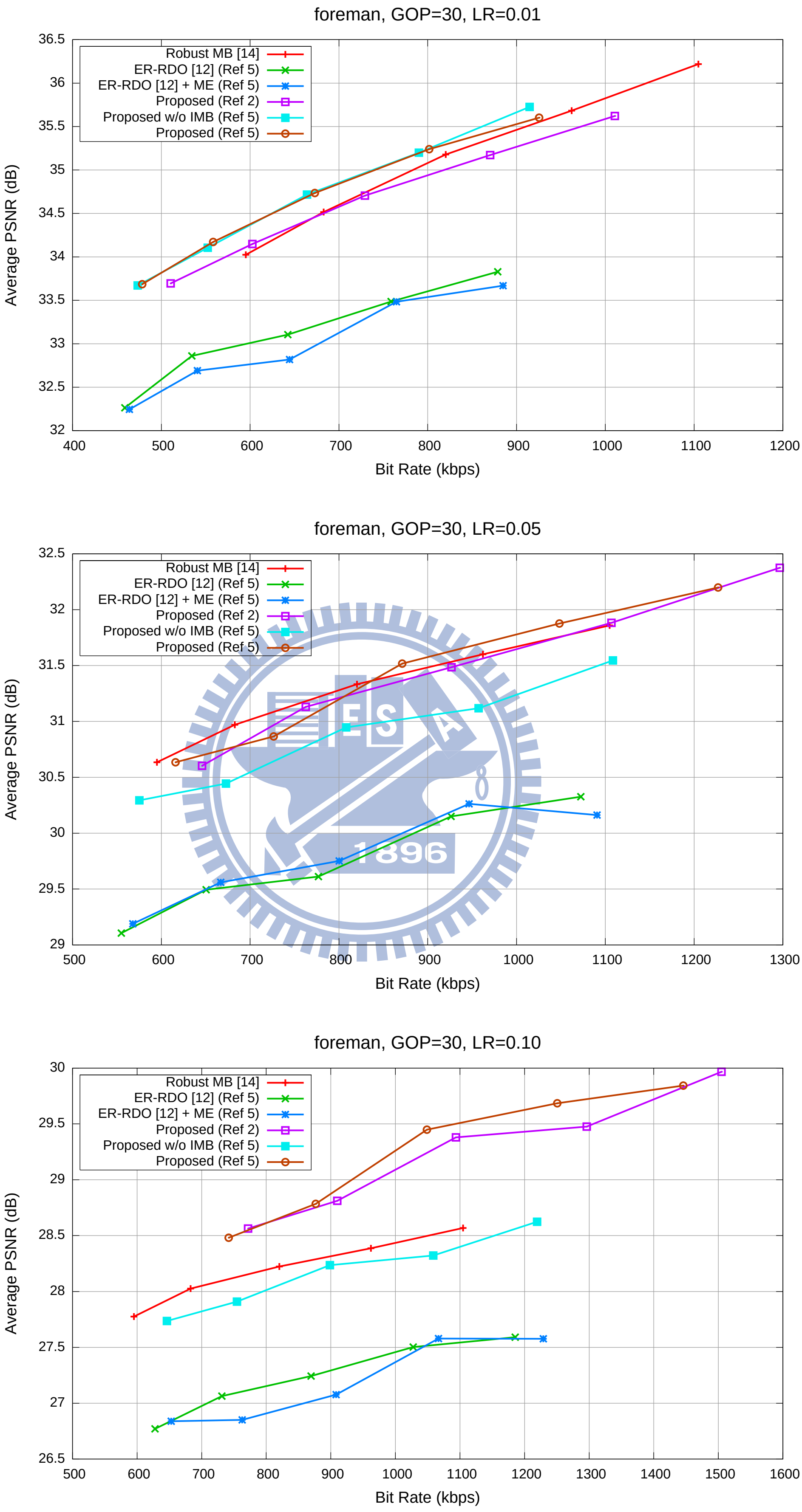

Figure 6.1 *Foreman* - Bit Rate v.s. Average PSNR with $p = 0.01, 0.05, 0.10$

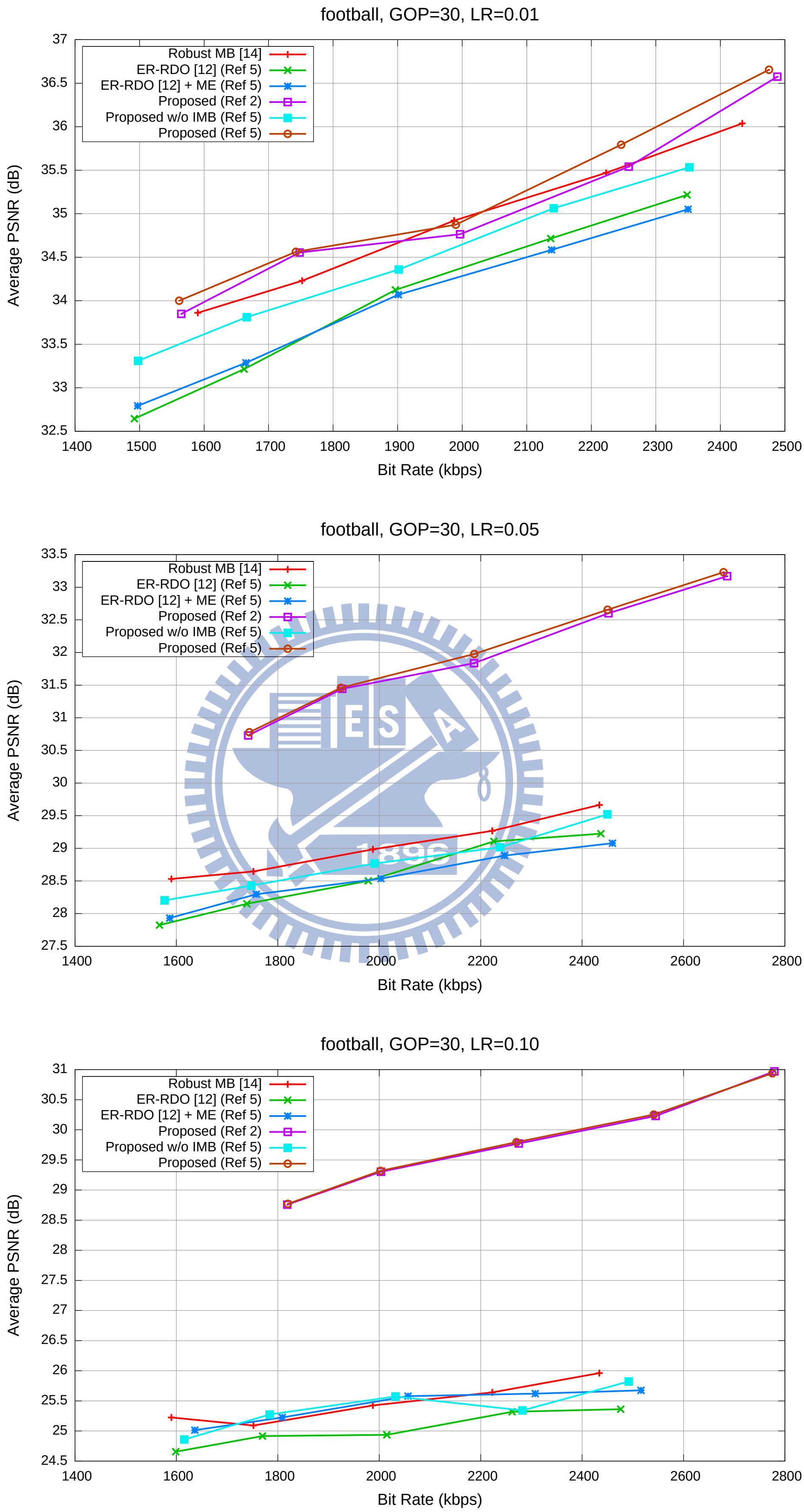

Figure 6.2 *Football* - Bit Rate v.s. Average PSNR with $p = 0.01, 0.05, 0.10$

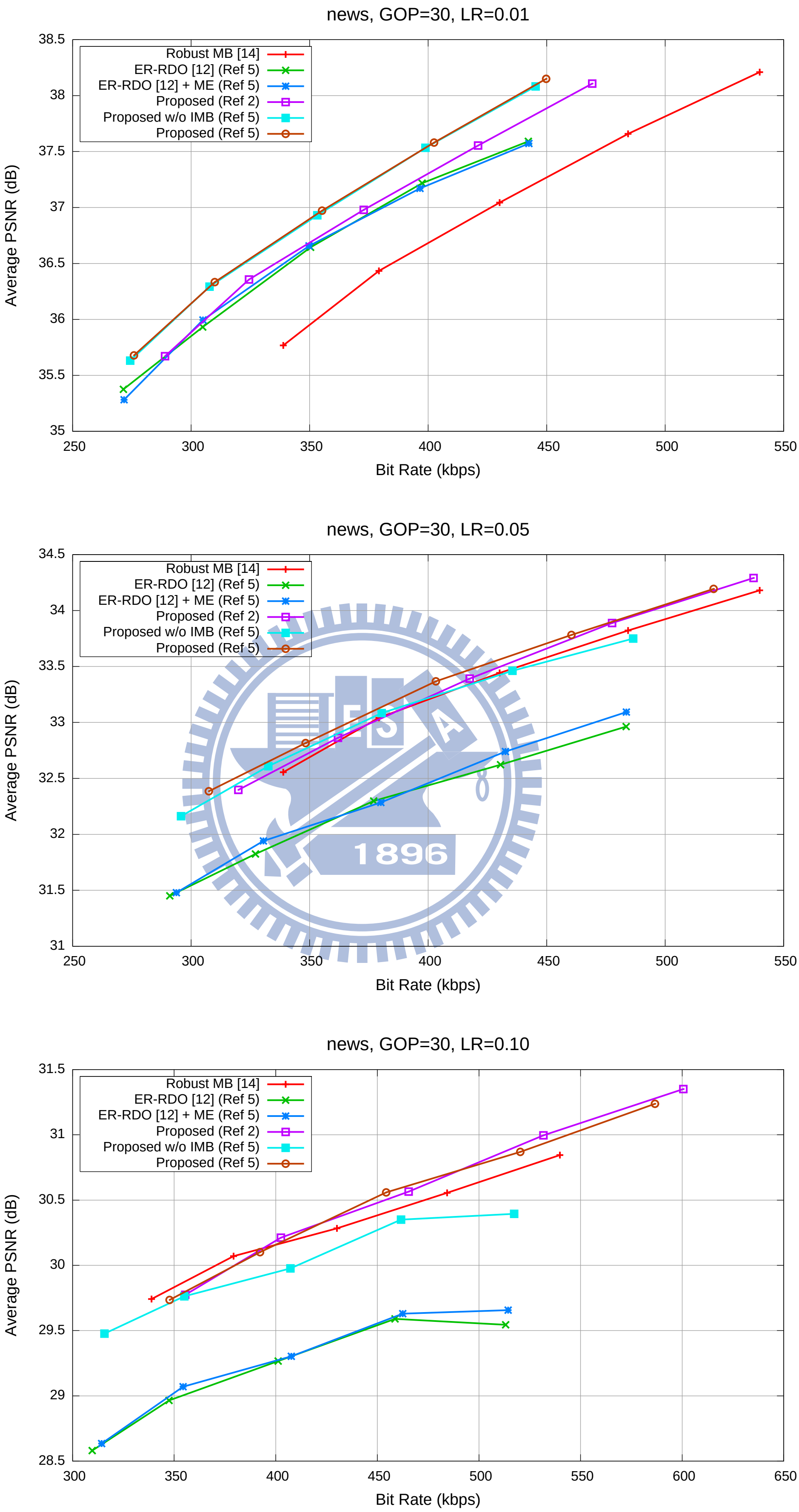

Figure 6.3 *News* - Bit Rate v.s. Average PSNR with $p = 0.01, 0.05, 0.10$

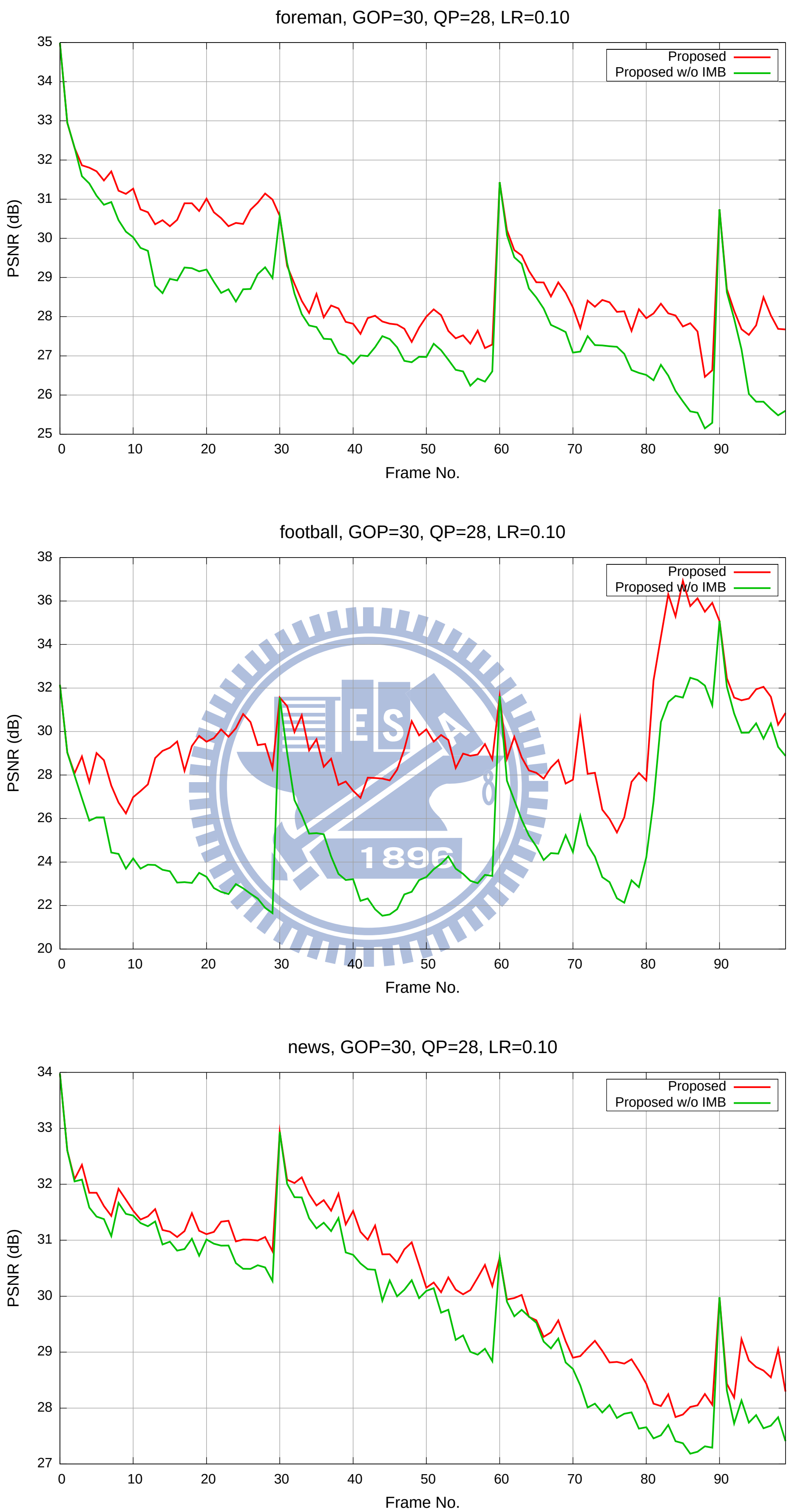

Figure 6.4 Frame v.s. PSNR with $p = 0.10$ for sequences (a) *Foreman* (b) *Football* (c) *News*

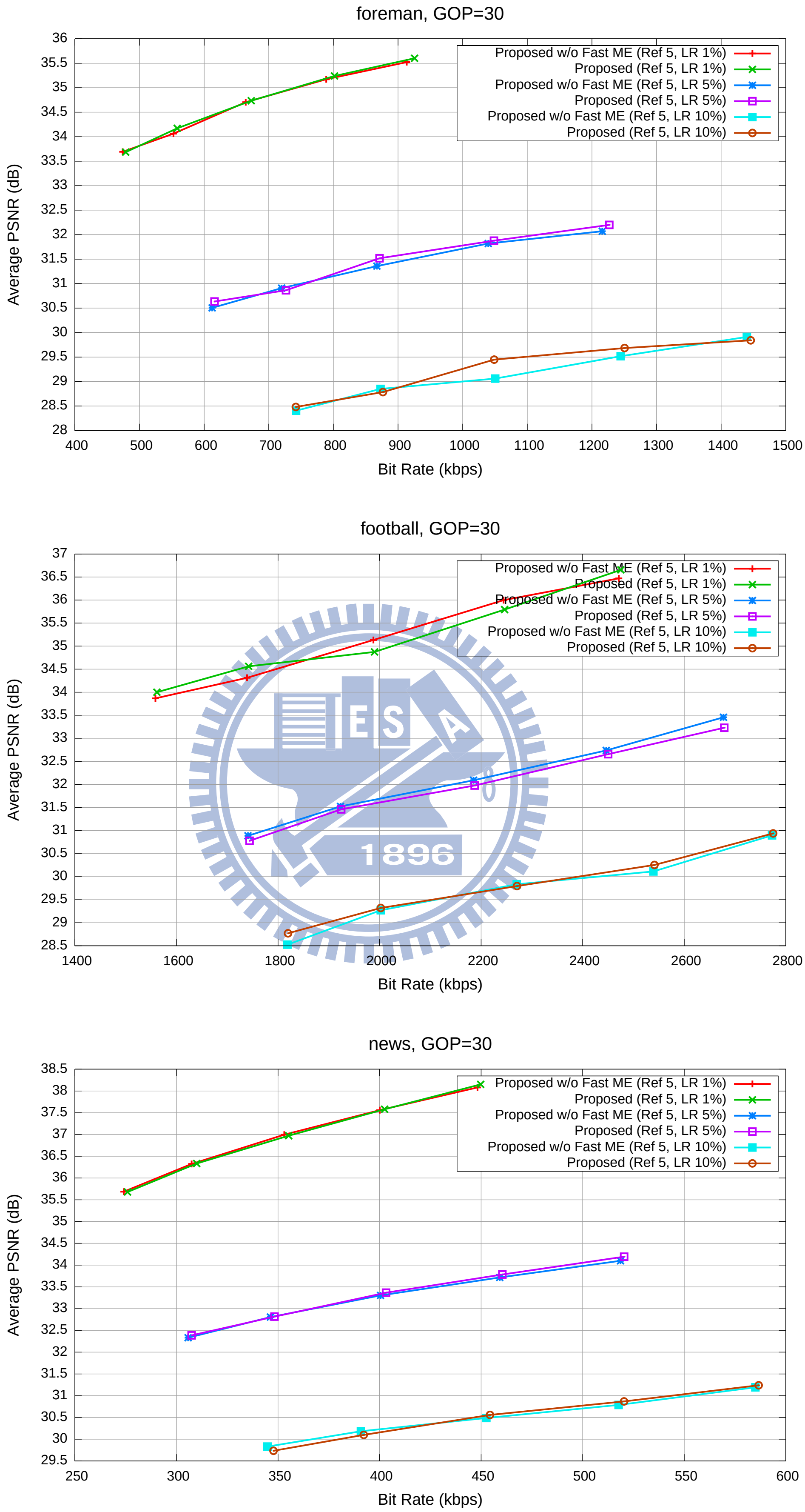

Figure 6.5 Performance comparison between the proposed methods with and without FME for sequences (a) *Foreman* (b) *Football* (c) *News*

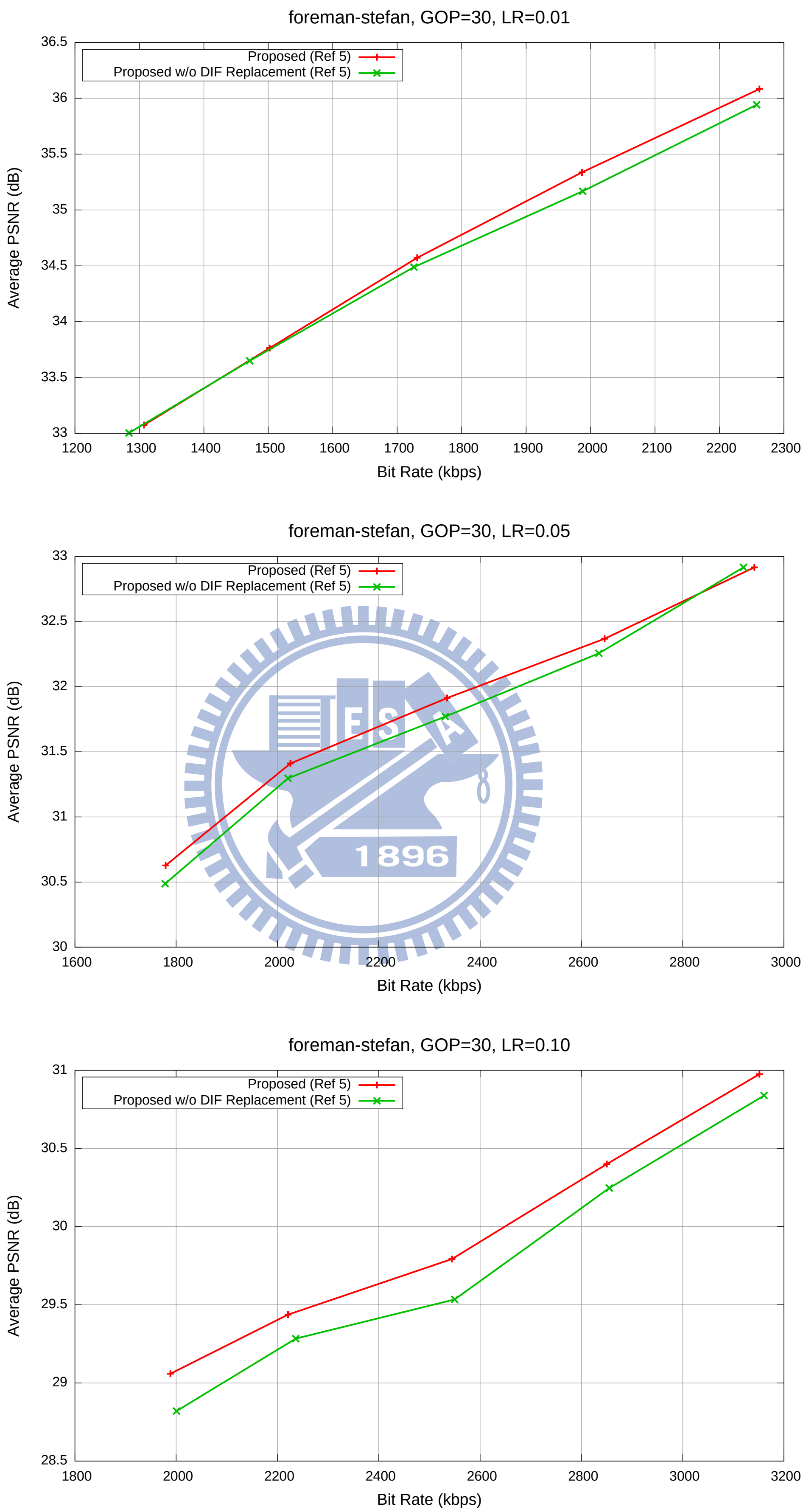

Figure 6.6 *Foreman-Stefan* - Bit Rate v.s. Average PSNR with $p = 0.01, 0.05, 0.10$ when scene change happened at frame 34 and frame 63

Chapter 7

Conclusion

In this thesis, an RDO-based error resilient scheme using MRF has been presented. We propose a candidate reference frame set with inclusion of the dominant intra frame (called DIF) to suppress the error propagation and an error resilient RDO model applied to both stages Motion Estimation and Mode Decision for better coding efficiency in different network conditions and different contents. Futhermore, we propose a fast motion estimation algorithm to reduce the computational complexity. To fit in the situation with scene changes, we also propose a replacement mechanism for DIF. The experimental results show that, our proposed scheme improves coding efficiency with different content and network conditions but without much increase of computational complexity in the MRF environment.

References

[1] J. Mochnac and S. Marchevsky, "Error resilience tools in the mpeg-4 and h.264 video coding standards," *Radioelektronika, 2008 18th International Conference*, pp. 1 – 4, Mar 2008.

[2] A. L. T Wiegand, G Sullivan, "Draft itu-t recommendation and final draft international standard of joint video specification (itu-t rec. h.264/iso/iec 14 496-10 avc)," *JVT-G050r1*, 2003.

[3] N. Thomos, S. Argyropoulos, and N. Boulgouris, "Robust transmission of h. 264/avc video using adaptive slice grouping and unequal error protection," *IEEE Int. Conf. on Multimedia and Expo*, pp. 593 – 596, Jan 2006.

[4] M. Luby, L. Vicisano, J. Gemmell, L. Rizzo, M. Handley, and J. Crowcroft, "The use of forward error correction (fec) in reliable multicast," 2002.

[5] S. B. Wicker, *Reed-Solomon Codes and Their Applications.* Piscataway, NJ, USA: IEEE Press, 1994.

[6] T. Stockhammer, D. Kontopodis, and T. Wieg, "Rate-distortion optimization for jvt/h.26l video coding in packet loss environment," *Proc. Packet Video Workshop*, Jan 2002.

[7] T. Stockhammer and S. Wenger, "Standard-compliant enhancement of jvt coded video for transmission over fixed and wireless ip," *Proc. IWDC 2002*, 2002.

[8] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Magazine*, Jan 1998.

[9] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h. 264/avc video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 560 – 576, Jan 2003.

[10] R. Zhang, S. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *Selected Areas in Communications, IEEE Journal on*, vol. 18, pp. 966 – 976, Jun 2000.

[11] H. Yang and K. Rose, "Rate-distortion optimized motion estimation for error resilient video coding," *IEEE International Conference on Acoustics*, vol. 2, pp. 173 – 176, Jan 2005.

[12] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao;, "Joint source-channel rate-distortion optimization for h.264 video coding over error-prone networks," *Multimedia, IEEE Transactions on*, vol. 9, pp. 445 – 454, Apr 2007.

[13] J. Zheng and L.-P. Chau;, "Error-resilient coding of h.264 based on periodic macroblock," *Broadcasting, IEEE Transactions on*, vol. 52, pp. 223 – 229, Jun 2006.

[14] Q. Zhang and G. Liu, "Error resilient coding of h.264 using intact long-term reference frames," *Visual Information Engineering, 2008. VIE 2008. 5th International Conference on*, pp. 62 – 66, Jan 2008.

[15] S. Pejhan, M. Schwartz, and D. Anastassiou, “Error control using retransmission schemes in multicast transport protocols for real-time media,” *IEEE/ACM Transactions on Networking (TON)*, vol. 4, pp. 413 – 417, Jan 1996.

[16] W. Tu and E. Steinbach, “Proxy-based reference picture selection for error resilient conversational video in mobile networks,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 151 – 164, Jan 2009.

[17] J. Youn and M.-T. Sun;, “A fast motion vector composition method for temporal transcoding,” *Circuits and Systems, 1999. ISCAS ’99. Proceedings of the 1999 IEEE International Symposium on*, vol. 4, pp. 243 – 246, Jan 1999.

[18] S. Yang, D. Kim, Y. Jeon, and J. Jeong, “An efficient motion re-estimation algorithm for frame-skipping video transcoding,” *IEEE International Conference on Image Processing*, vol. 3, pp. 68 – 71, Jan 2005.

[19] A. M. Tourapis and A. Leontaris, “H.264/mpeg-4 avc reference software,” *http://iphome.hhi.de/suehring/tml/*, 2009.