國立交通大學

電子工程學系電子研究所碩士班

碩士論文

以運動向量修正爲基礎的抗錯性編碼技術與

以損失畫面修復爲基礎的錯誤修補技術

Error Resilience Based on Motion Vector Correction and

Error Concealment Based on Missing Frame Prediction

指導教授： 王聖智 教授

研究生： 曾禎宇

以運動向量修正爲基礎的抗錯性編碼技術與

以損失畫面修復爲基礎的錯誤修補技術

研究生： 曾禎宇　　指導教授：王聖智 教授

國立交通大學

電子工程學系 電子研究所碩士班

摘要

壓縮視訊經過無線網路傳輸時，由於傳輸品質不穩定可能會造成傳送的封包丟失，當所傳送的視訊爲低位元率時，封包丟失可能會造成全畫面的損失，而產生嚴重的錯誤傳遞問題。爲了克服這樣的問題，我們在解碼端提出了一種新的錯誤修補技術，以運動連續性爲基礎，利用後向運動投射來估測損失時的畫面運動，讓原本以損失畫面爲參考的畫面得以略過損失的畫面，而改以前一張正確的畫面來進行解碼。這樣的做法會比傳統修正損失畫面的做法簡單許多。

另一方面，由於我們所提出的修補方式是基於畫面運動的連續性特質，若是原本編碼時的運動向量與真實運動軌跡不相符時，將造成錯誤的修補效果。爲了增進修補效果，我們也提出了在編碼端的運動向量修正技術，利用模糊邏輯來判斷運動向量的可靠度，讓修正之後的運動向量能比較接近真實運動軌跡。而由於運動向量的修正可能會造成編碼效率降低的問題，所以我們也進一步討論運動向量修正在編碼效率與解碼端錯誤修補效果之間的權衡問題，並提出一種運動向量部分修正的技術，能夠在不過度增加編碼負荷的情況下，達到比較好的錯誤修補效果，以降低畫面損失對解碼視訊品質的破壞。

Error Resilience Based on Motion Vector Correction and Error Concealment Based on Missing Frame Prediction

Student: Chen-Yu Tseng Advisor: Prof. Sheng-Jyh Wang

Department of Electronics Engineering
Institute of Electronics
National Chaio Tung University

Abstract

When low bit-rate videos are transmitted over error-prone wireless networks, a packet loss may correspond to the loss of an entire frame. Moreover, with error propagation, the quality of subsequent frames will also be seriously degraded. To overcome this video degradation problem, we propose a new error concealment technique that is based on motion consistency. Whenever a reference frame is lost, we recover the corrupted motion information based on backward motion projection. Here, we skip the lost frame and use a previously decoded frame as a substitute of reference frame. If compared with some existing frame-based error concealment methods, the proposed algorithm offers a much simpler solution.

Since the proposed error concealment is based on motion consistency, the performance of error concealment may get deteriorated if the encoded motion vectors do not reflect the actual motion trajectories. Hence, at the encoder site, we also propose a new motion vector correction algorithm based on fuzzy logic to provide motion vectors closer to the true motion trajectories. Since the correction of motion vectors may increase the bit-rate of the encoded videos, we also discuss the trade-off between encoding efficiency and concealment performance. A partial motion vector correction algorithm is then proposed to provide improved concealment performance while maintaining reasonable coding efficiency.

Contents

Tables

Figures

Chapter 1 Introduction

Recent developments of video compression technology, like H.264/AVC, have greatly improved the efficiency of video coding and video transmission. On the other hand, the progress of wireless transmission technology has also promoted the development of wireless video transmission, such as video conference or video telephone over wireless environments. However, when the video transmission condition is unstable, transmitted packets may easily get lost. The loss of information contained in the lost packet causes errors in decoding processing. The decoding errors not only damage the present decoding frame, but also affect the frames which use the damaged frame as a reference. The errors further affect the subsequent frames and cause the error propagation phenomenon.

The error propagation problem will reduce the quality of decoded videos. However, in real-time applications, such as video telephone, the retransmission of lost packets is not allowable. To overcome the video degradation problems, we may apply the error concealment technology in the decoding process. The lost parts can be concealed based on the information from some other correctly received neighboring frames. On the other hand, we can also apply error resilience in the encoding process. By adding some extra information during the encoding process, we can increase the error robustness of the decoding process.

Most traditional error concealment techniques are based on Macroblock (MB). The lost block is concealed with the information of neighboring blocks. However, in low bit-rate video transmission, a lost packet contains several MBs. The neighbors of lost MB may also get lost. Moreover, the loss of a whole frame may probably happen. In this condition, these traditional methods are not applicable and we can only conceal the error parts in the frame level. Due to the temporal consistency of the video frames, we can predict a lost frame from its previous frames with the motion information. Generally, most existing frame-based concealment techniques use the motion information of previous frames to forward project to the current missing frame to recover the lost motion information. With the recovered motion vectors, the missing frame could be reconstructed by motion compensation.

Because the error concealment is based on the motion information, motion vectors (MVs) are the already available information from the motion estimation operation of the encoder. However, the general motion estimation is based on blocking matching. Sometimes the estimated MVs may not correspond to the true motion trajectory. These wrong MVs will reduce the performance of concealment results. To make the MVs approach the true motion trajectory, we proposed a motion

vector correction based on fuzzy logic. In the encoding process, the motion vector correction is performed to correct the MVs depending on the fuzzy reliability of MVs. This technique could be considered as an error resilience method that improves the performance of error concealment.

On the other hand, in the decoding process, we proposed a new error concealment technology that is based on backward motion projection. If compared with existing frame-based concealment techniques, the proposed method offers a much simpler solution to release the error propagation problem.

After motion vector correction in the decoding process, the performance of error concealment can be improved. However, the residual of motion compensation is increased, and the coding efficiency is lowered. Hence, we will also discuss the relation between the error resilience in the encoder and the error concealment in the decoder. Furthermore, we proposed a partial motion vector correction algorithm that offers a trade-off between concealment performance and coding efficiency.

Moreover, B slices are an additional option provided by the Main Profile of H.264/AVC to enhance the coding efficiency. The prediction of B slices is bidirectional. Hence, B slices may be predicted from one or two reference pictures, before or after the current picture. If compared with the IPPP… GOP type, the error problems of GOP with B slices is somewhat different. Hence, the corresponding error concealment issue is also different. For example, the distance between two P frames is increased. Therefore the difficulty of concealment is increased. In this thesis, we propose a new GOP structure and a corresponding concealment technique to overcome the error problems when B slices are present in the coding process.

This thesis is organized as follows. In Chapter 2, we describe the video compression codec, transmission, and the technology of error concealment and error resilience. In Chapter 3, we first introduce the prior arts of error concealment for a whole-frame loss based on optical flow. Then we will introduce the proposed error resilience and error concealment algorithms. The experiments and results will be shown in Chapter 4. Finally, we will give conclusions in Chapter 5.

Chapter 2 Background

In this chapter, we first introduce video compression. The H.264/AVC codec is briefly introduced is Section 2.1. After the introduction of video compression, compressed video transmission and the error problem during transmission are described in Section 2.2. Furthermore, some existing error concealment and error resilience techniques are described in Sections 2.3 and 2.4.

2.1 Introduction to Video Compression

Video compression technology is a very important technology for promoting the efficiency of video storage or video transmission. Generally, the video compression can be performed in temporal domain or spatial domain. In the temporal domain, there is a high correlation between neighboring frames. In other words, the difference between neighboring frames is low. Therefore, we can predict the current frame based on the previous ones to remove the temporal redundancy. On the other hand, there is also a high correlation between neighboring parts within a frame. Moreover, high frequency components in a video frame is less important to human vision. Hence, video compression is usually performed based on the above observations.

In general we can divide existing video compression techniques into the following processes:

1. **Prediction Coding**

   As described before, there are high correlations between neighboring video components in both temporal and spatial domains. The prediction performed in spatial domain is called **Intra Prediction**. On the other hand, the prediction performed in the temporal domain is called **Inter Prediction**, which is based on motion estimation and motion compensation to remove temporal redundancy.

2. **Transform Coding**

   With a suitable data transformation from the original data space to another space, the transformed data become more compact and the compression efficiency can be enhanced. For example, the Discrete Cosine Transform (DCT) and Wavelet Transform are two widely used transformations.

3. **Quantization**

   Because of the low sensitivity of human vision to picture detail, a lossy

coding is acceptable in order to reduce the amount of coded data. With quantization, the fine detail of picture is dropped to increase the coding efficiency.

4. **Entropy Coding**

   Entropy coding is based on the well-known Shannon's information theory. Currently, variable length coding techniques, such as Huffman Coding and Arithmetic Coding, are widely used to reduce the bit-rate of coded.

In the subsequent subsections, we focus on the H.264/AVC standard and give a brief introduction to the H.264/AVC codec.

2.1.1 Introduction to H.264/AVC

H.264/AVC is a standard proposed by the Joint Video Team of ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Pictures Experts Group (MPEG) for video compression. It is the newest video coding standard that provides high coding efficiency and network-friendly video representation. As shown in Fig. 2.1, the scope of the H.264/AVC standard only includes the decoder part. In the following paragraphs, we will further introduce the details of H.264 encoder and decoder.

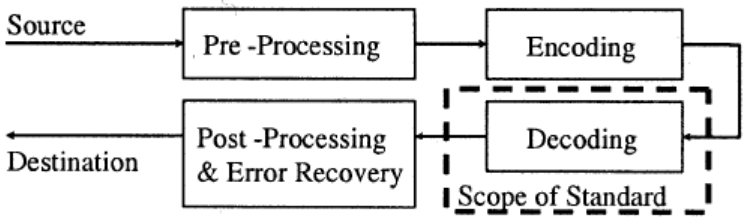


Fig. 2.1 Scope of H.264/AVC [1]

2.1.1.1 H.264/AVC Encoder

A schematic H.264/AVC encoder is shown in Fig. 2.2, where F_n is the current encoding picture. First, the encoder divides F_n into one or several slices. Each slice contains several Macroblocks (MBs), which are the basic coding units. Each MB is encoded in the intra or inter mode depending on the corresponding slice mode (see Table 2-1). The residual between the predicted MB (marked 'P' in Fig. 2.2) and the encoded MB is further encoded with the transform coding, quantization, reordering and finally the entropy coding. Then, the compressed bit-stream is passed to the Network Abstraction Layer (NAL) for transmission or storage.

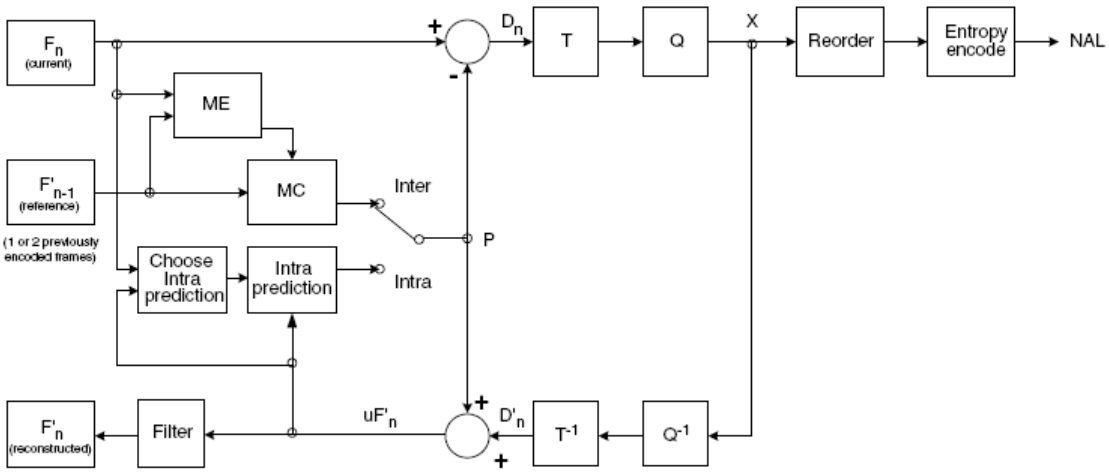


Fig. 2.2 H.264/AVC Encoder [2]

Table 2-1 H.264 Slice modes

Slice type	Description	Profile(s)
I (Intra)	Contains only I macroblocks (each block or macroblock is predicted from previously coded data within the same slice).	All
P (Predicted)	Contains P macroblocks (each macroblock or macroblock partition is predicted from one list 0 reference picture) and/or I macroblocks.	All
B (Bi-predictive)	Contains B macroblocks (each macroblock or macroblock partition is predicted from a list 0 and/or a list 1 reference picture) and/or I macroblocks.	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

1. Intra Prediction

When intra mode is applied, P is predicted based on the spatial information. For the luma samples, P is formed for each 4x4 block or 16x16 MB. There are nine optional prediction modes for each 4x4 luma block (Fig. 2.3) and four optional modes for each 16x16 luma block. Fig. 2.4 shows the corresponding prediction blocks of a 4x4 luma block.

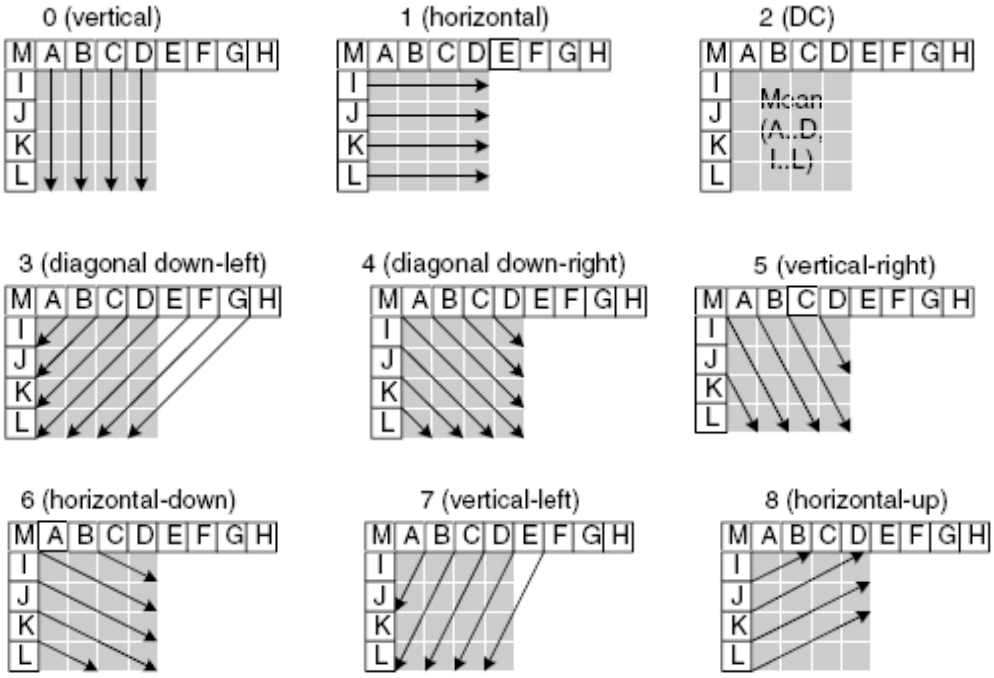


Fig. 2.3 4x4luma prediction modes[2]

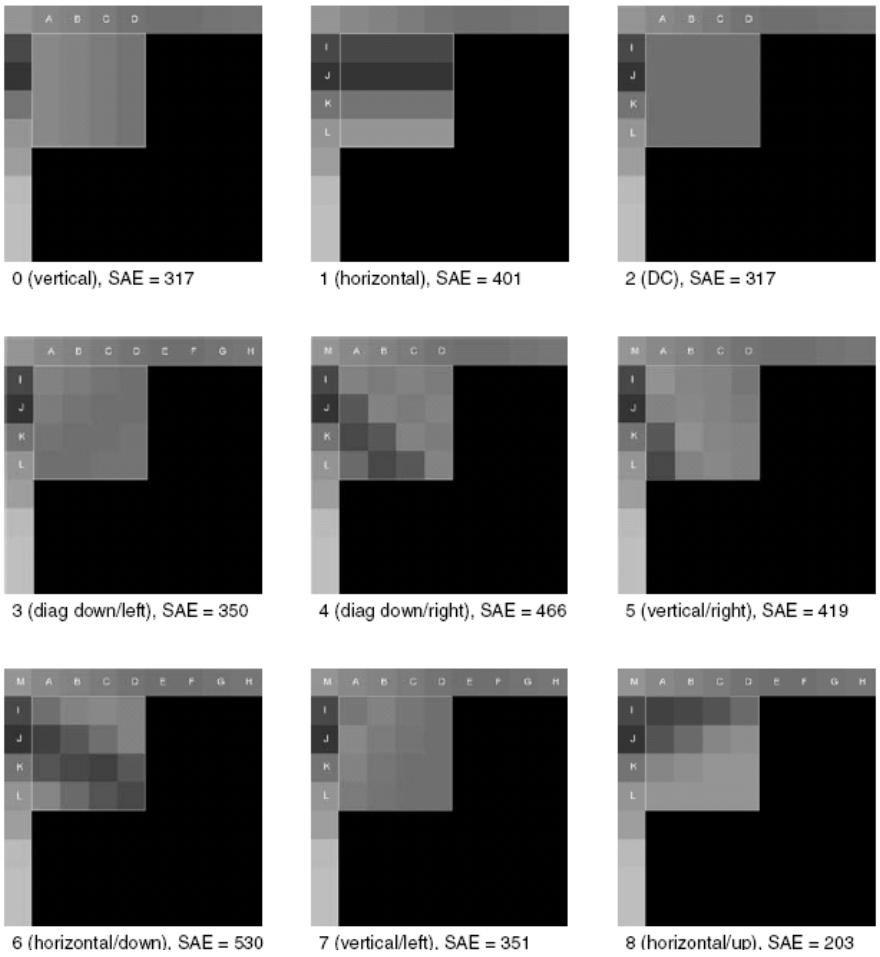


Fig. 2.4 Prediction blocks (luma 4x4).[2]

2. Inter Prediction

Compared with intra prediction, inter prediction is based on temporal information and generally has higher predicting accuracy. Hence, inter prediction plays a very important role in video compression.

A predicted block P is searched from the reference picture F_{n-1} by motion estimation. The displacement from the current block to the predicted block P is called Motion Vector (MV). With the encoded information of MVs and residual, the current picture could be reconstructed from the reference picture F_{n-1} by motion compensation. In other words, only the MVs and residual need to be encoded. Hence, the amount of coded data can be efficiently reduced. Next, we will describe the features of H.264 inter prediction.

A. Tree-structure motion compensation

One of the main differences from previous coding standards is that H.264 supports variable block sizes from 16x16 to 4x4 in motion compensation. Fig. 2.5 shows the segmentation of a macroblock into one 16x16 partitions, two 8x16 partitions, two 16x8 partitions, or four 8x8 partitions. If the 8x8 partitions is chosen, each of the 8x8 blocks may be further split in four different ways, as shown in Fig. 2.6. In general, if the block size is smaller, the residual is smaller but more bits of MVs are required. The allowance of variable block sizes provides a flexible choice of block size to increase the coding efficiency.

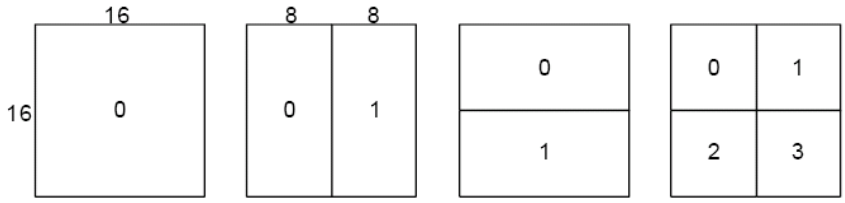


Fig. 2.5 Macroblock partitions: 16x16, 8x16, 16x8, 8x8. [1]

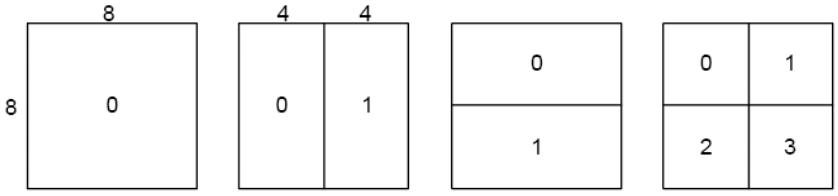
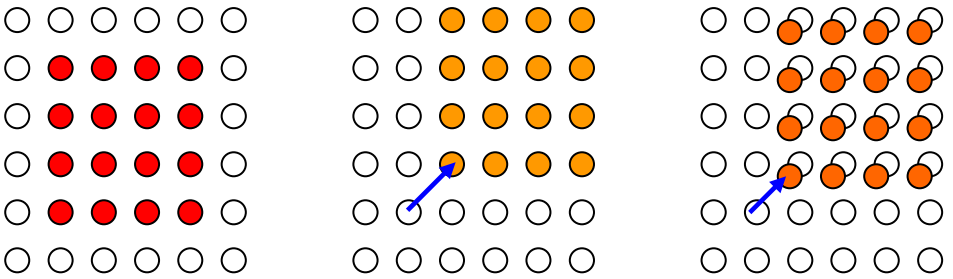


Fig. 2.6 Sub-macroblock partitions: 8x8, 4x8, 8x4, 4x4. [1]

B. Sub-pixel motion vectors

In order to increase the accuracy of motion compensation, H.264 supports sub-pixel motion compensation that provides quarter-pixel accuracy. Fig. 2.7 shows an example of sub-pixel prediction. If the horizontal and vertical components of motion vector are all integers, the relevant samples in the

reference block actually exist as shown in Fig. 2.7 (b). Relatively in (c), the motion vector is fractional and the prediction samples are generated by interpolation between adjacent samples in the reference picture.



(a) 4x4 block in current frame

(b) Reference block: vector (1, -1)

(c) Reference block: vector (0.75, -0.5)

Fig. 2.7 Example of integer and sub-sample prediction [2]

2.1.1.2 H.264/AVC Decoder

After receiving the compressed data from the NAL layer, the decoder performs the decoding process which can be seen as a backward operation of the encoding process. A schematic presentation of the decoder is illustrated in Fig. 2.8. After entropy decoding, reordering, inverse quantization and inverse transform, the residual D'_n is reconstructed. With the addition of predicted block P by intra or inter prediction, we can reconstruct the pictures.

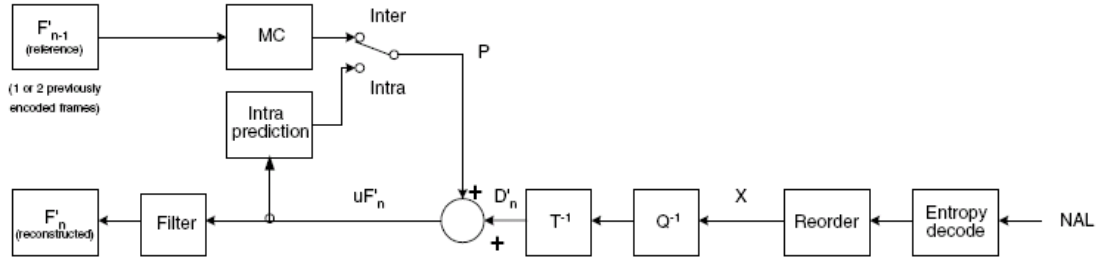


Fig. 2.8 H.264/AVC Decoder. [2]

2.1.1.3 Highlighted Features of H.264/AVC

In this section, the highlighted features of H.264/AVC are described below which can enhance the coding efficiency and are somewhat different from previous coding standards.

1. **Improvement of prediction**
   - Variable block-size motion compensation with small block sizes

- Quarter-sample-accurate motion compensation
- Motion vectors over picture boundaries
- Multiple reference picture motion compensation
- Decoupling of referencing order from display order
- Decoupling of picture representation methods from picture referencing capability
- Weighted prediction
- Improved “skipped” and “direct” motion inference
- Directional spatial prediction for intra coding
- In-the-loop deblocking filtering

2. Improvement of coding efficiency

- Small block-size transform
- Hierarchical block transform
- Short word-length transform
- Exact-match inverse transform
- Arithmetic entropy coding
- Context-adaptive entropy coding

3. Robustness of network transmission

- Parameter set structure
- NAL unit syntax structure
- Flexible slice size
- Flexible macroblock ordering (FMO)
- Arbitrary slice ordering (ASO)
- Redundant pictures
- Data Partitioning
- SP/SI synchronization/switching pictures

2.2 Compressed Video in Wireless Networks

After the introduction of the video compression about H.264 Video Coding Layer (VCL), we will discuss the video transmission over wireless networks in this section.

In general, we can divide the video transmission process into three steps: encoding, transporting, and decoding, as illustrated in Fig. 2.9. In different applications, the corresponding categories are different. For example, in real-time applications such as video conference or video telephone, encoding, transporting, and decoding are performed simultaneously. The PCS (Packet-switched conversational service) is a suitable service category in this case. So far, there are three major service

categories in wireless networks [3]:

1. Packet-switched conversational services (PCS) for video telephony and conference;
2. Live or pre-recorded video packet-switched streaming services (PSS); and
3. Video in multimedia messaging services (MMS).

As mentioned before, PCS is applied for real-time video communications. Different from PCS, the video source data of PSS is encoded beforehand. In PSS applications, the encoding is separated from transporting, but the transporting and decoding are usually performed simultaneously in order to minimize the initial delay and memory usage in receivers. In contrast with PCS and PSS, MMS does not include any real-time constraints. Therefore, all the encoding, transporting, and decoding are processed separately.

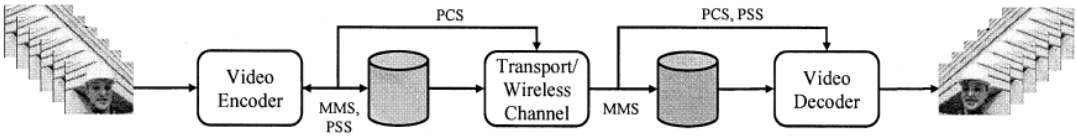


Fig. 2.9 Wireless video applications differentiated by real-time or offline processing: MMS, PSS, and PCS: [3]

Although the service categories are different depending on the applications as mentioned before, video streaming in wireless networks still possess some common characteristics. In general, video streaming is transferred in packet-based wireless networks. Fig. 2.10 demonstrates an example of H.264/AVC standard, which shows that after the encoding through VLC (video coding layer) and NAL (network abstraction layer), different transport protocols can be used for different purposes. For example, if we want to transfer video data as a file (as MMS), TCP/IP is an appropriate transport protocol. On the other hand, in real-time conversational applications, such as video conference, RTP/UDP/IP would be a preferred choice. In the next subsection, we will further introduce the H.264/AVC Network Abstraction Layer (NAL).

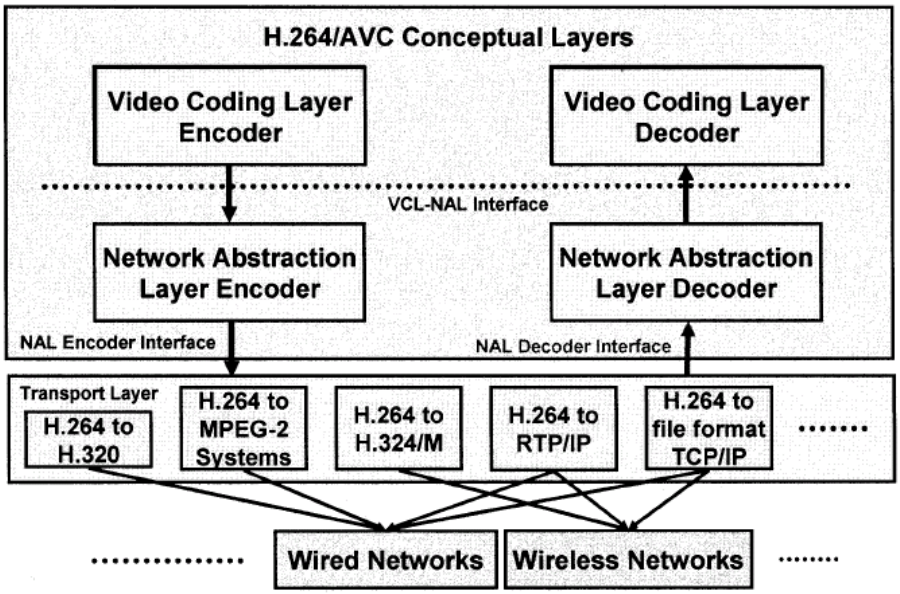


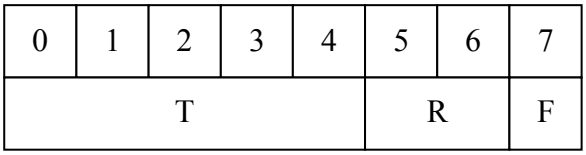
Fig. 2.10 H.264/AVC standard in transport environment [3]

2.2.1 H.264/AVC Network Abstraction Layer

In this section we will briefly introduce the H.264/AVC Network Abstraction layer (NAL) designed to make the H.264/AVC code "network friendly". It facilitates the H.264/AVC VLC code transport layer as shown in Fig. 2.10. For example, in real-time video communication applications, RTP/UDP/IP is a suitable protocol. There is exactly one NAL unit in one RTP packet for simple packetization. Next, we further introduce the NAL units.

2.2.1.1 NAL Units

After choosing a suitable transporting protocol, the coded video is organized in terms of NAL units (NALUs). Each NALU is a video packet containing an integer number of bytes. The first byte is the header byte of NALU, as shown in Fig. 2.11. The header byte contains the NAL unite type (T), the nal_reference_idc (R) that indicates the importance of an NALU for the reconstruction process, and the forbidden_bit (F) which is set to '0' in H.264 encoding.



T: NAL unit type
R: NAL_reference_idc
F: forbidden_bit

Fig. 2.11 Header byte of NAL unit [2]

2.2.2 Video Transmission over Networks

As mentioned before, the coded video data would be organized into NALUs as video packets. Each packet contains the information of a number of macroblocks. After transferring thought a wireless network, transferred packets may be lost due to the unstable transmission. The packet loss not only damages the current picture, but also affects the following pictures due to the use of temporal information for compression. Take inter prediction as an example, the following picture would use the previous picture as a reference. If there is some error blocks in the current picture caused by packet loss, these parts of the next picture that refer to the erroneous blocks cannot be decoded correctly and further affect the subsequent pictures. This is known as error propagation and is illustrated in Fig. 2.12. In order to minimize the quality degradation caused by error propagation, several error concealment and error resilience techniques have been proposed.

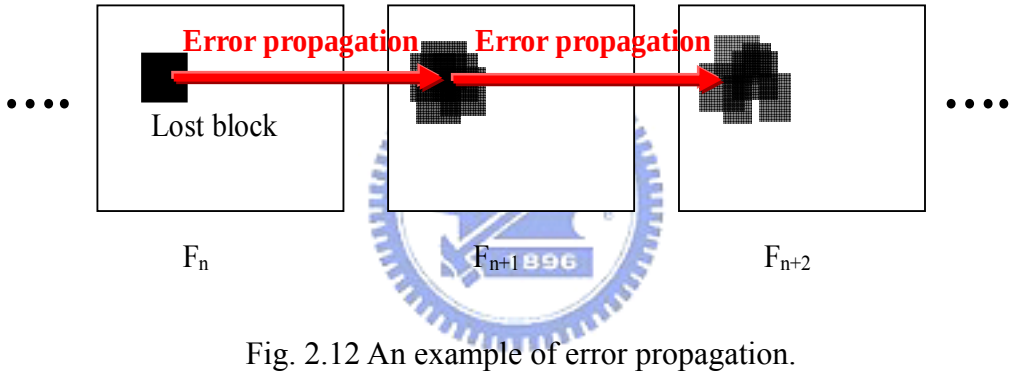


Fig. 2.12 An example of error propagation.

2.2.2.1 Simulation Tools for Video Transmission

In order to realize how video quality gets degraded by packet losses, we may apply some simulation tools to simulate the video transmission over a packet-lossy network. There have been several research works discussing the affection of packet losses in video decoding [4] [5]. Some simulation tools have also been developed to simulate video transmission over real networks, as shown in Fig. 2.13. For example, Fig. 2.14 shows a schematic representation of an evaluation framework [7] with NS-2[8]. These simulation tools may help us to evaluate the degradation of received videos.

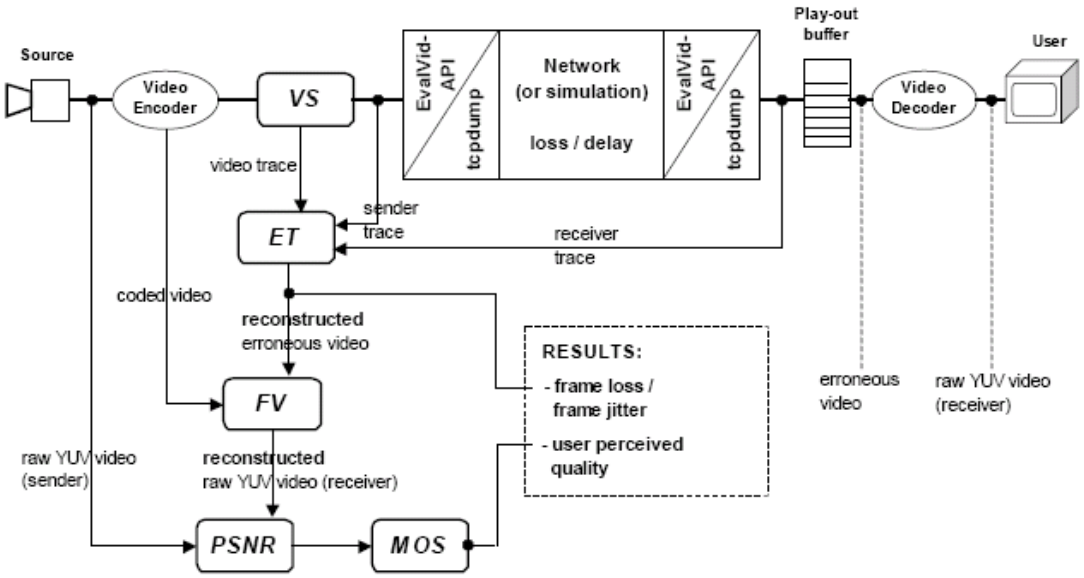


Fig. 2.13 Schematic representation of an evaluation framework. [6]

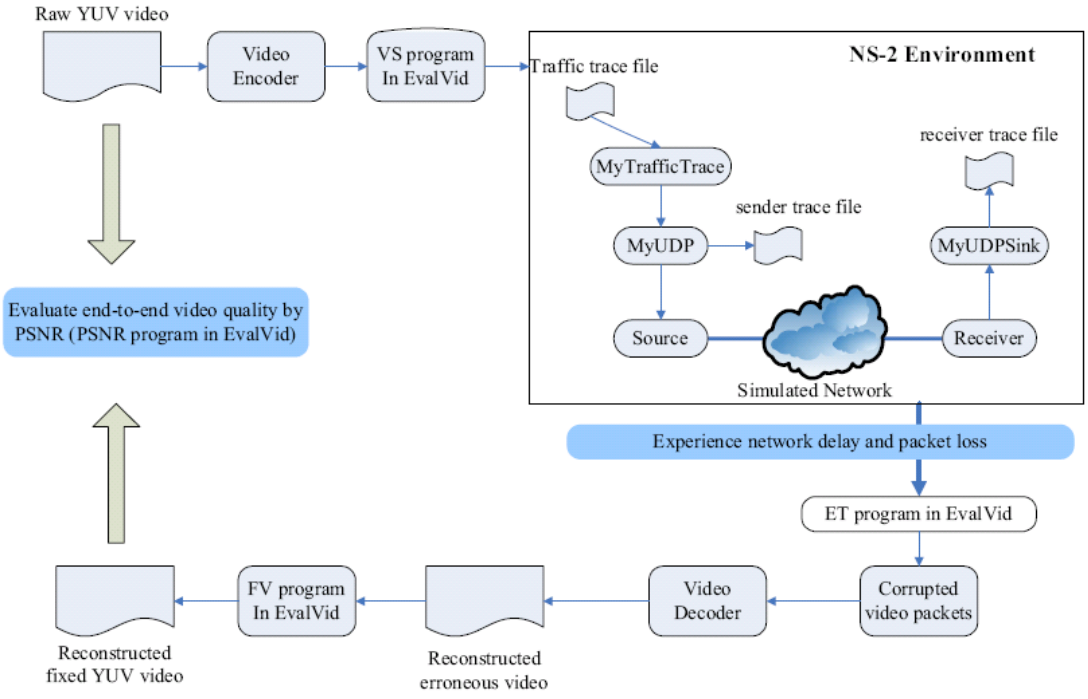


Fig. 2.14 Schematic representation of an evaluation framework with NS-2. [7]

2.3 Error Concealment

In last section, we introduced the video transmission and the packet loss problems. Due to error propagation, packet loss degrades the subsequent pictures until presence of the next new intra frame. Therefore, when a video packet is lost, video quality will be degraded unless the lost packet is retransmitted. However, in some real-time applications, such as PCS or PSS, the transporting and decoding processes are performed simultaneously. The packet retransmission may cause decoding delay in the receiver. In these cases, packet retransmission is not allowable. To release the video degradation problems caused by to packet loss, error concealment is widely used.

In this section, we will introduce some existing error concealment techniques, which can be categorized into Block-level and Frame-level techniques. In a Block-level technique, error concealment is performed based on macroblock units. If assuming the neighboring MBs of the lost MB is correctly received, we could reconstruct the lost MB with the information of neighbors. However, in general, a packet contains more than one MB and the neighboring MBs of the lost MB are also lost. Furthermore, in low bit-rate cases, one packet may contain a whole frame and thus packet loss possibly causes the loss of an entire frame. In these cases, block-level concealment is not appropriate and frame-level concealment is required instead. The study of frame-level concealment is a main subject of this thesis.

2.3.1 Block-Level Concealment

If the neighboring MBs of a missing MB in the lost packet are received correctly, the lost MB can be concealed with the neighboring information, as illustrated in Fig. 2.15. Here, (a) is the frame with the missing MB and (b) illustrates the concealment based on spatial interpolation, which will be described in 2.3.1.1. Fig. 2.15 (c) illustrates the temporal concealment that will be further described in 2.3.1.2 and 2.3.1.3 .

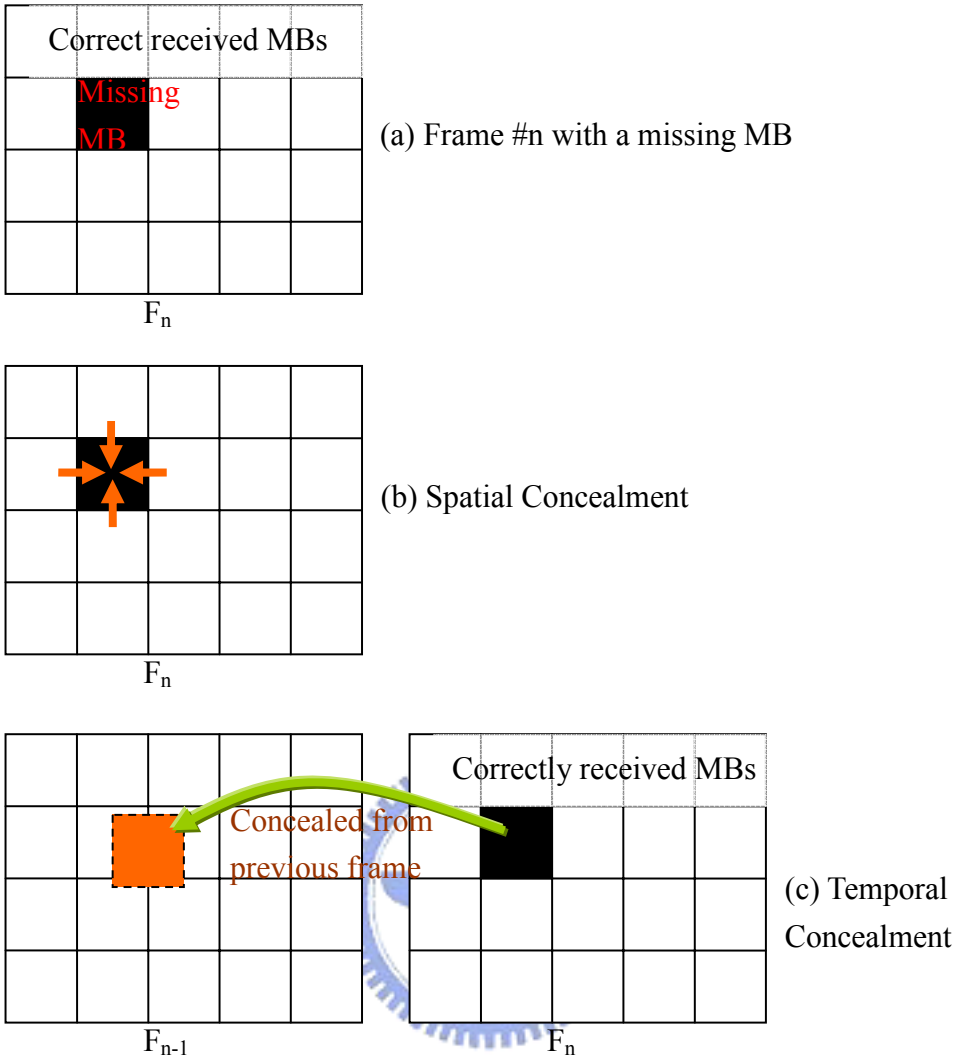


Fig. 2.15 Block-level concealment technique

2.3.1.1 Concealment Based on Spatial Interpolation

In this section, we will describe error concealment based on spatial interpolation as shown in Fig. 2.15(b). Depending on the consistency in the spatial domain, the missing parts can be predicted from the correctly received neighbors. However, with the simple linear interpolation, the predicted block may be over-smoothed. It is hard to predict the edge components in the missing block. Therefore, some directional interpolation techniques are proposed in [9-11], which attempt to predict the edge information in the missing block based on the edge detection of the neighboring parts as shown in Fig. 2.15. Although these methods preserve more edge information as shown in Fig. 2.16, it is still hard to reconstruct more complicated MBs by spatial interpolation. For complicated case, temporal based concealment techniques usually have better performance, which will be introduced in the next subsection.

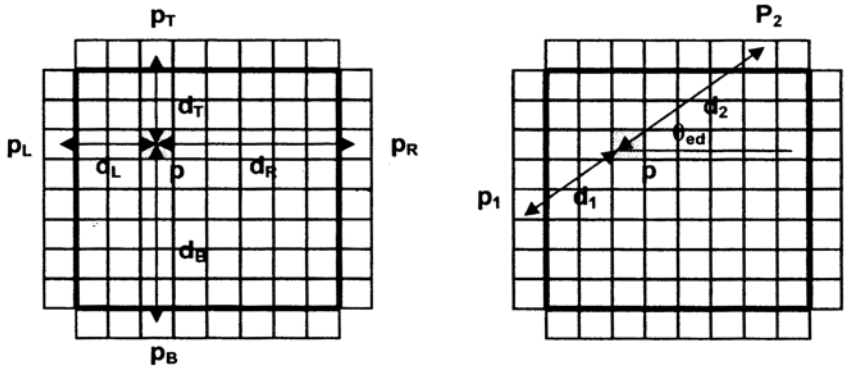


Fig. 2.15 Spatial interpolation: weighted averaging (left), main direction smoothing (right) [10].

Original sequence | Edge preserving | Bilinear interpolation

Fig. 2.16 Example of spatial interpolation [11].

2.3.1.2 Concealment Based on Boundary Matching

Depending on the temporal continuity of video frames, there may be some blocks in the reference frame that are similar to the missing block. Hence, we may find a "suitable" block in the reference frame to substitute the missing block. The question is how to find the "suitable" block. One of the well known methods is the Boundary Matching Algorithm (BMA) proposed by W. M. Lam *et al.* in [12]. With the boundary information of the missing MB, the substituted block is found when the boundary of it is most matched to the boundary of the missing block, as illustrated in Fig. 2.17. Thereafter, based on BMA, such as EBMA (Extended Boundary Matching Algorithm [12]) and WEDMA (Weighted Extended Boundary Matching Algorithm [13]), there are some extended algorithms that improve the matching criteria to find better substituted blocks. Moreover, there are some algorithms that combine the temporal concealment and spatial concealment [14][15] to improve the performance.

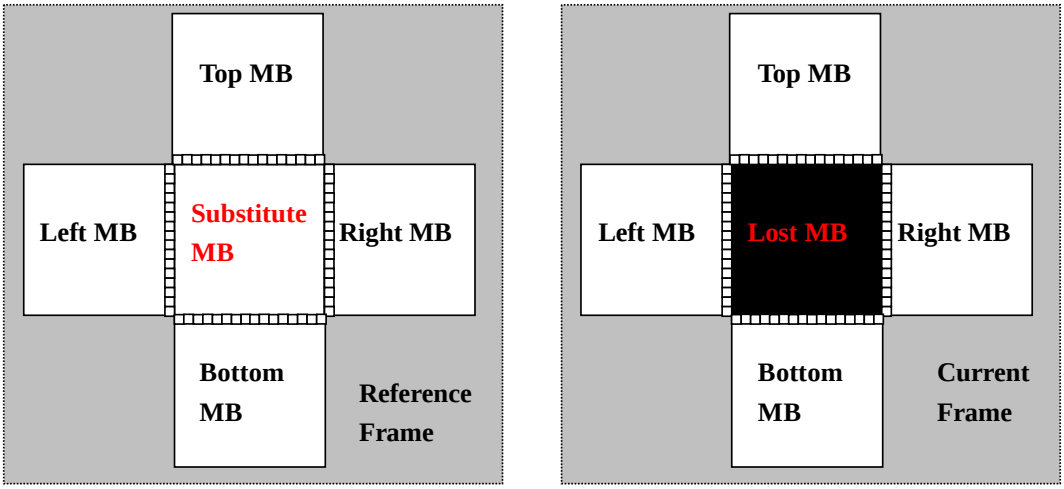


Fig. 2.17 Boundary Matching Algorithm (BMA)

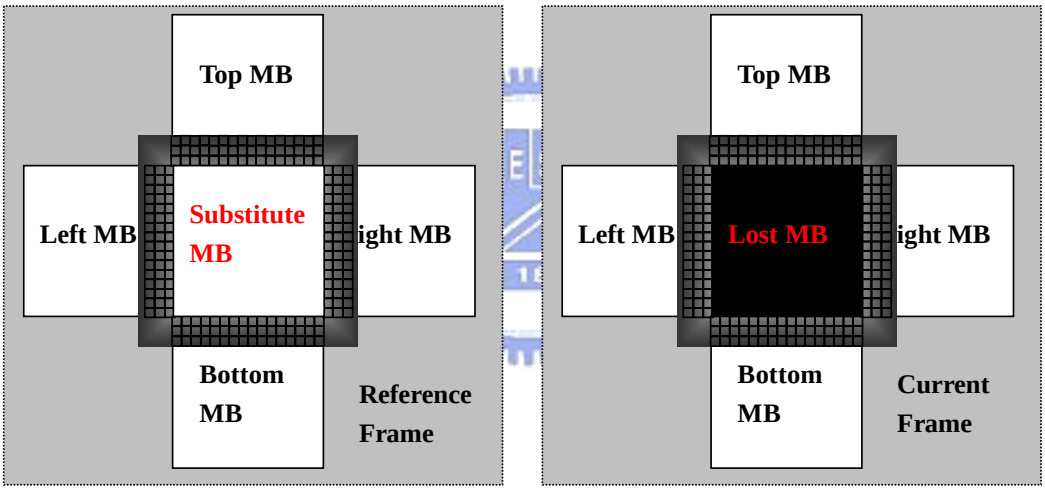


Fig. 2.18 Weighted Boundary Matching Algorithm (WBMA)

2.3.1.3 Concealment Based on Motion Vector Field Recovery

The BMA-based concealment introduced in the previous section can be seen as a process of motion vector recovery, as illustrated in Fig. 2.19. With the recovered MV, we can find a block in the reference frame to substitute the missing block. BMA is one way to find the recovered MV. Besides, the recovery of motion vector field is another commonly used concealment method.

Fig. 2.20 is an example of motion vector field recovery, where the left shows the original MV field. The missing MV can be predicted with the information of the surrounding MVs depending on the motion consistency in the spatial domain. For

example, if one car is moving, the whole part of the car is generally moving in the same direction. Therefore, spatial consistency exists in the motion vector field and the missing MV can be recovered by some method like the Motion Field Interpolation method proposed by M. E. Al-Mualla *et al.* in [16] and the Lagrange interpolation [17].

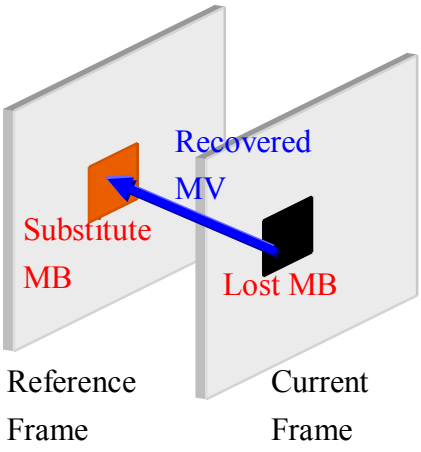


Fig. 2.19 Compensation based concealment with recovered motion vector

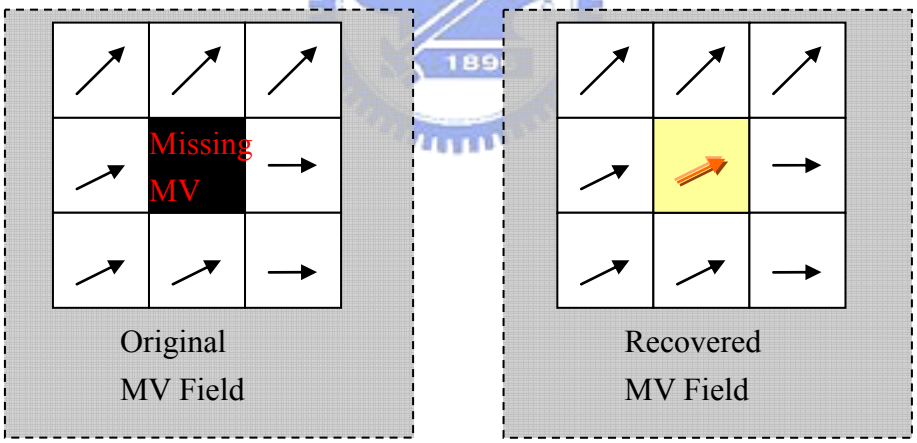


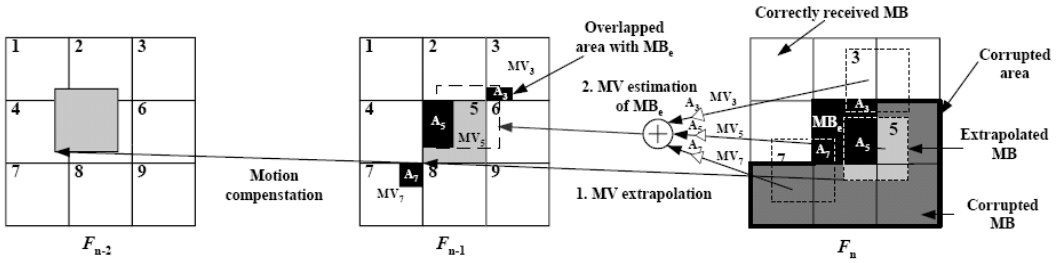
Fig. 2.20 Motion vector field recovery.

In addition to the spatial information in the MV field, the temporal information is useful in the prediction of the missing MV. Jae-Young Pyun *er al.* proposed the Bidirectional Motion Tracking algorithm to predict the missing MV from temporally adjacent MVs, as shown in Fig. 2.21 [18]. The prediction can be divided into forward direction and backward direction. The former is motion vector extrapolation from the successive MVs, as illustrated in Fig. 2.21(a), and the latter is backward motion tracking, as illustrated in Fig. 2.21(b). The recovered MV is the weighted sum of the

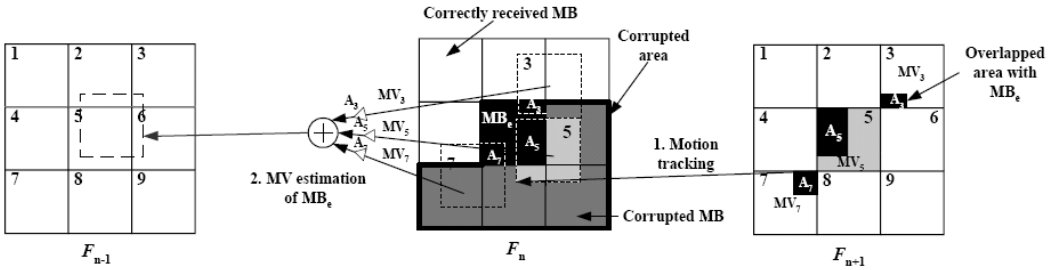
projected MVs, and the weight of the projected MV depends on the projected area on the target block. We will further discuss the temporal MV recovery in Chapter 3.

$$MV_E^n = \frac{\sum_{i=1}^{M}[\alpha_i \cdot MV_i^{n-1}]}{\sum_{i=1}^{M}\alpha_i}, \quad (2.1)$$

$$MV_B^n = \frac{\sum_{i=1}^{M}[\alpha_i \cdot MV_i^{n+1}]}{\sum_{i=1}^{M}\alpha_i}, \quad (2.2)$$



(a)



(b)

Fig. 2.21 Temporal motion vector recovery [18]：
(a) Motion vector extrapolation (FMT) and (b) Backward motion tracking (BMT).

2.3.2 Frame-Level Concealment

2.3.2.1 Motivation

The block-level concealment algorithms, no matter based on spatial interpolation or temporal compensation, are feasible only when the adjacent blocks are received correctly. However, a lost packet contains a large number of MBs. If successive MBs in the same packet are lost, the adjacent MBs of a lost one may probably be corrupted

too. In this case, these block-level concealment methods become impractical. To overcome this problem, FMO (Flexible Macroblock Ordering) is provided in H.264 to partition MBs into different packets, as shown in Fig. 2.26. However, in real cases, when errors happen in a burst way, as modeled by the Gilbert-Elliot loss model [20] [21] in a harsh channel condition, succcesive packet losses may happen. In this case, adjacent MBs may probably get corrupted at the same time due to bursting errors.

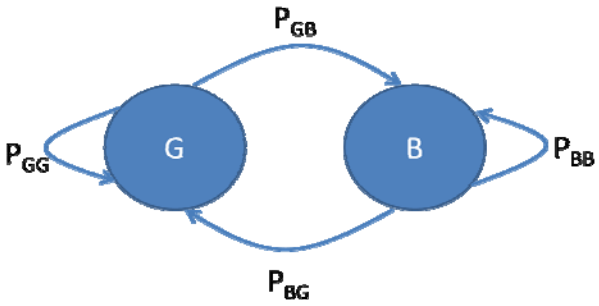


Fig. 2.22 Gilbert-Elliot loss model.

The situation gets even worse in the low bit-rate video transmission. In low bit-rate video transmission, a packet loss may cause the loss of an entire frame. Take the 3G application as an example [22]. If one QCIF video is transferred at 10fps on a 64kbit/s channel, the coded frame is 800 bytes on average. In general, the packet size is set to be around 1k bytes. Hence, it is very likely that one packet may contain a whole frame. Even if we make the packet size smaller, the problem may still exist. This is because bursting errors may cause successive packet losses, as shown in Fig. 2.23 [19].

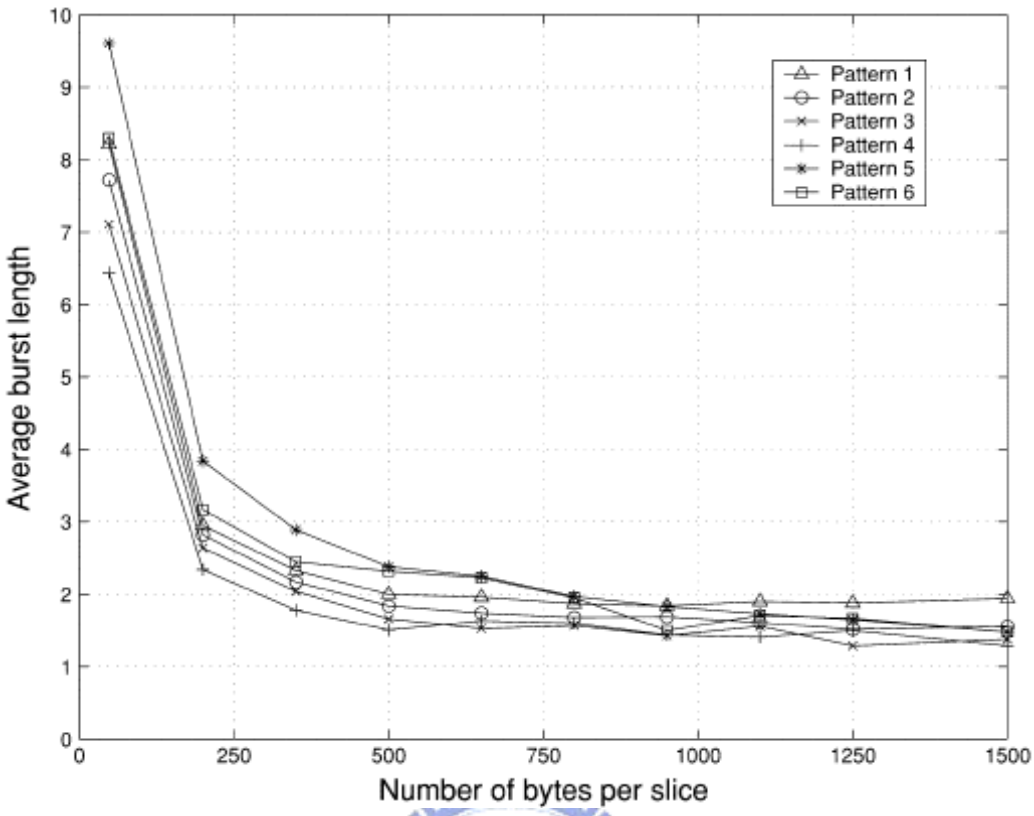


Fig. 2.23 Average burst length versus packet size [19]

2.3.2.2 Concealment of Whole-Frame Loss

If compared with block-level concealment, there is no spatial information for frame-level concealment since the whole frame is lost. In this case, only temporal information is available. Depending on the temporal consistency of video sequence, the lost frame can be reconstructed with subsequent frames. For example, in Fig. 2.24, the ball moves from the middle of F_{n-2} to the lower left side of F_{n-1}. Hence, we may presume that the ball moves continuously in the same direction to F_n.

Fig. 2.25 illustrates how to reconstruct the missing frame from the information of successive frames. In Fig. 2.25(a), F_n is the missing frame and F_{n-1} is the reference picture. Based on the motion consistency, we can inverse the MV of the reference picture to form a forward MV and project it to the missing frame F_n, as shown in Fig. 2.25(b). Recoding the reference MV in the projected block, we can obtain the recovered MV as shown in Fig. 2.25(c). After recovering all the MVs in the missing frame, we can reconstruct the missing frame based on the recovered MVs and the reference frame via motion compensation. The temporal MV recovery appraoach is the main subject of this thesis. We will have further discussion in Chapter 3.

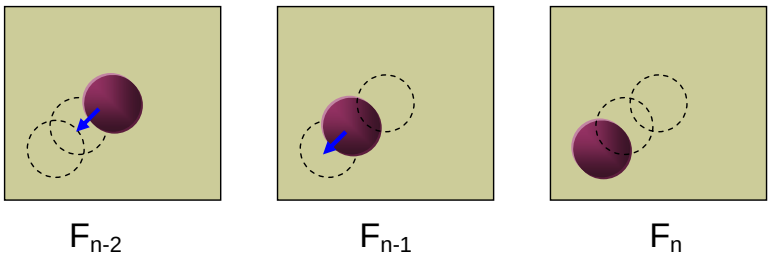


Fig. 2.24 An example of motion consistency.

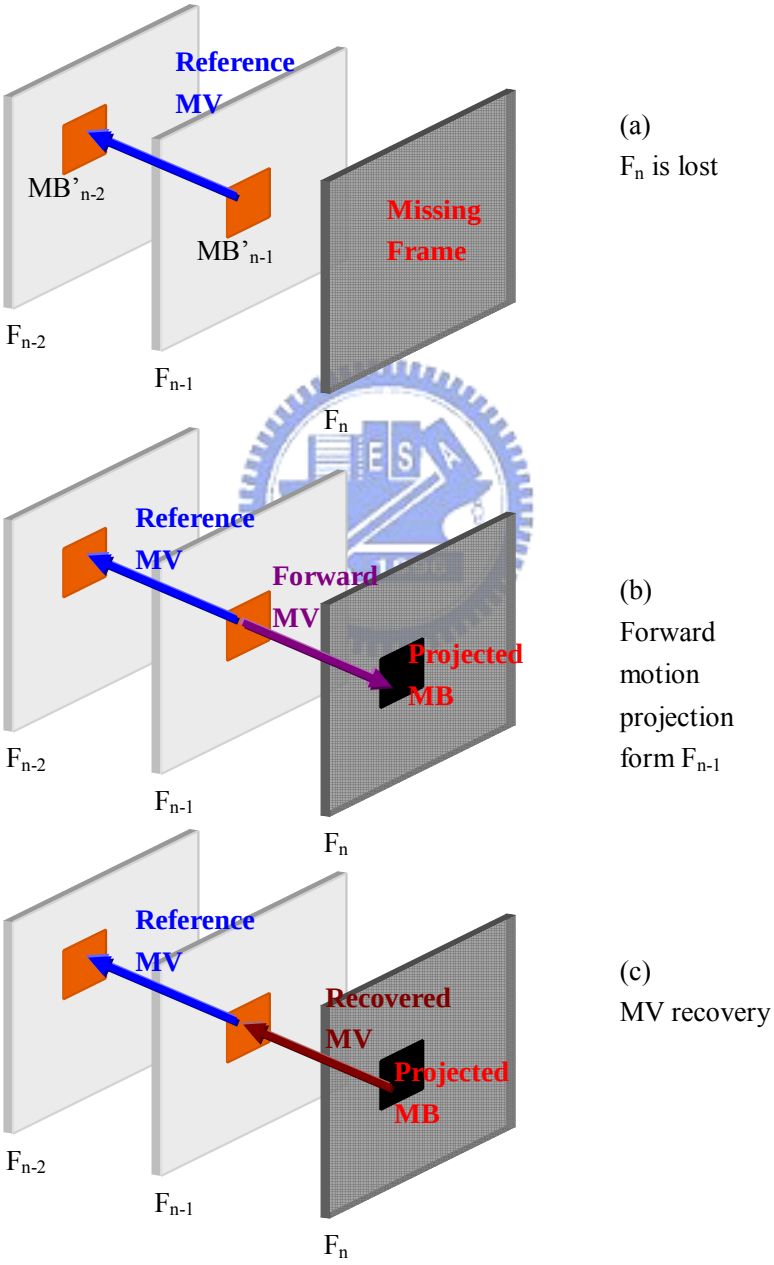


Fig. 2.25 Motion vector field recovery.

2.4 Error Resilient Tools in H.264

In order to minimize the video degradation caused by packet loss, the aforementioned error concealment technology is commonly used. Here we further introduce another technology, known as error resilience, to improve the encoding process or adds more information to enhance the error robustness of the coded videos. So far, there are several error resilient tools in H.264, as to be described below. [4]

A. Intra placement

As mentioned before, the error propagation is drifting till the next new intra slice appears. H.264 allows that the MBs in a slice to be intra coded independently. It means an intra MB can be placed at any position to suppress error propagation.

B. Picture segmentation

Picture segmentation provides the flexibility that the number of MBs in a slice can be decided without constraints. In general, one packet contains one slice. If the size of slice is flexible, the packet size can be made smaller and the packet loss may probably become lower.

C. Reference picture selection

H.264 allows at most five reference pictures. In a feedback-based system, the encoder receives the information about packet loss from the decoder. Then the subsequent pictures may avoid the reference of the corrupted areas.

D. Data partitioning

The H.264 coded data can be partitioned into three parts:

1. **Header information**
2. **Intra partition**
3. **Inter partition**

Depending on their affections to video decoding, different partitions have different degrees of importance. The header information is the most important part and contains some crucial information such as MB types, QP values, and motion vectors. For these important data, higher level protection is usually applied.

E. Parameter sets

There are two parameter sets in H.264: sequence parameter set and picture parameter set. The former contains the information related to a sequence of pictures, while the latter contains the information related to the slices belonging to a single picture. Both sets are packetized in a packet and are sent to the receiver in a more reliable way to guarantee the arrival of the vital information.

F. Flexible marcoblock ordering (FMO)

When a video packet is lost, the contained slice is lost. To avoid neighboring

MBs from getting corrupted together, H.264 provides the FMO option that disperses MBs into different slices, as shown in Fig. 2.26. This arrangement helps the decoder to conceal missing MBs from their adjacent MBs.

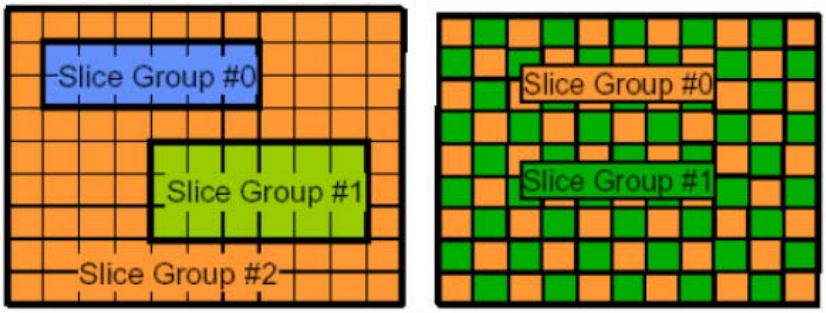


Fig. 2.26 Possible subdivisions of a picture into slices with FMO

G. Redundant slices (RS's)

For one MB, H.264 allows one or more redundant representations, called redundant slices (RS's). For example, beside the primary coding data, we can code low QP representations as RS's. The RS's are packetized in a different packet. Therefore, when the primary data are lost, the RS's can be used to reconstruct a video of lower but acceptable quality.

Chapter 3 Error Concealment and Error Resilience for Compressed Video

In this chapter, we focus on the problem that an entire frame is missed in the decoding process. In Section 3.1, we will introduce some existing error concealment techniques for whole-frame loss. In these techniques, the recovery of corrupted MVs is the key factor based on motion consistency. Hence, the accuracy of the MVs directly affects the performance of the concealment. Therefore, in Section 3.2, we will introduce the proposed MV correction as an error resilient technique. Then, the proposed concealment algorithm will be introduced in Section 3.3. Instead of reconstructing the lost frame, we proposed a method that skips the erroneous frames. In Section 3.4 we will discuss the relation between the proposed concealment and error resilience. The proposed partial MV correction will be introduced to avoid the raise of bit rate due to the use of MV correction. Moreover, if taking into account the B slices, the corresponding concealment technique will be somewhat different. In Section 3.5, we will introduce the proposed error resilient GOP structure that supports the inclusion of B slices, together with the corresponding concealment technique.

3.1 Previous Methods of Missing Frame Concealment

In low bit-rate video transmission, such as mobile video communication, a whole frame may be packetized within one single packet. Hence, a packet loss may cause the loss of an entire frame. In this case, these block-level concealment techniques introduced in Section 2.3.1 are not practicable since the spatial information surrounding the missing MBs is also corrupted. Comparatively, frame-level concealment based on temporal motion recovery is practicable for this case. Some existing frame-level concealment techniques will be first introduced in the following paragraph.

The algorithm of missing frame concealment was first proposed by S. Belfiore *et al.* in [22]. It features the capability of estimating entire missing frames of a video sequence. This is associated with the development of H.264, which provides high compression rate and low bit-rate video communication. After that, there have been several improved algorithms based on temporal motion recovery to restore the missing frame. We will further describe these algorithms in the following sections.

3.1.1 Motion Consistency and Optical Flow

As mentioned before, only temporal information is available for us to estimate the missing frame. If treating a video sequence as a three-dimensional structure as shown in Fig. 3.1, the pictures at different time instants correspond to different samples in time. For example, the ball moves continuously from the time instant n-2 to the time instant n. In the spatio-temporal domain, the motion of the moving ball can be described by optical flow (OF) with motion consistency. Moreover, each picture of the video sequence a temporal sample of the OF space.

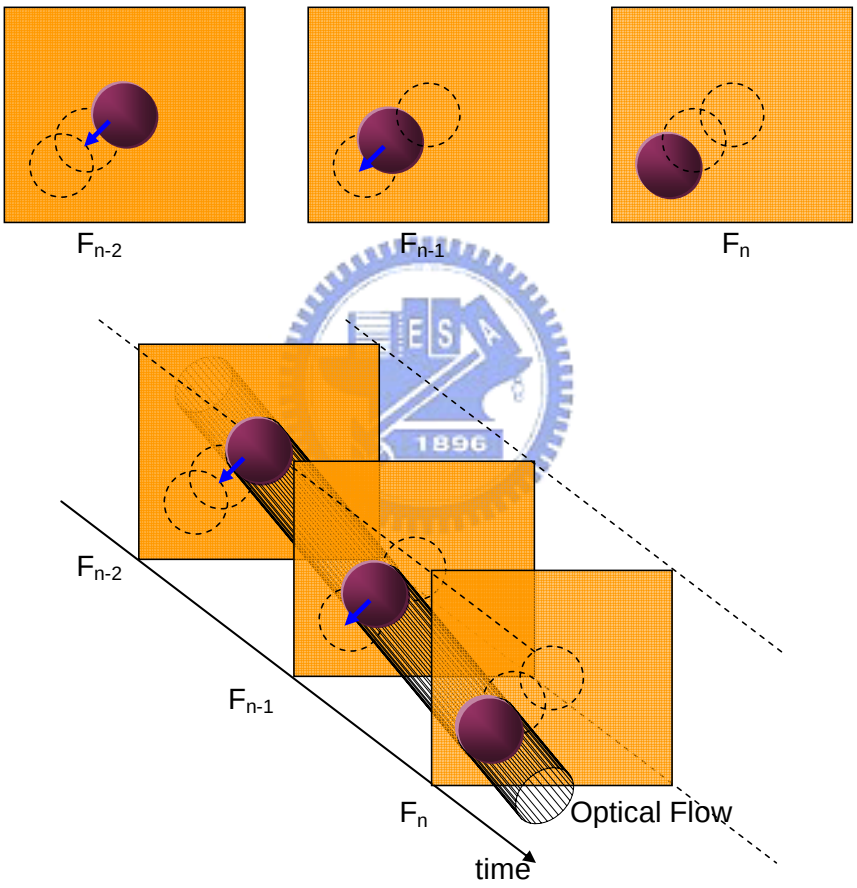


Fig. 3.1 An example of motion consistency.

A packet loss corresponds to a corruption of the optical flow, as shown in Fig. 3.2.When a video packet is lost, the contained information such as MVs is missing. The decoded picture is lost due to the corruption of the optical flow, as shown in Fig. 3.3. Correspondingly, if the corrupted OF can be recovered, the relation between the missing frame and the reference frame can be reconstructed, and we can estimate the

missing frame by using motion compensation.

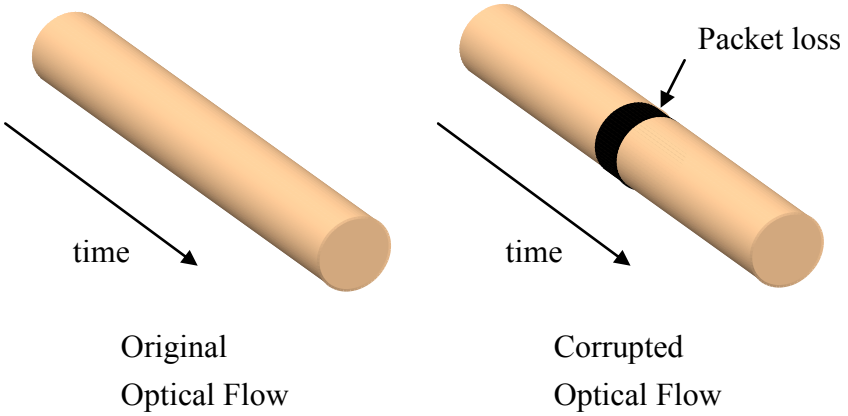


Fig. 3.2 A packet loss and the corruption of optical flow.

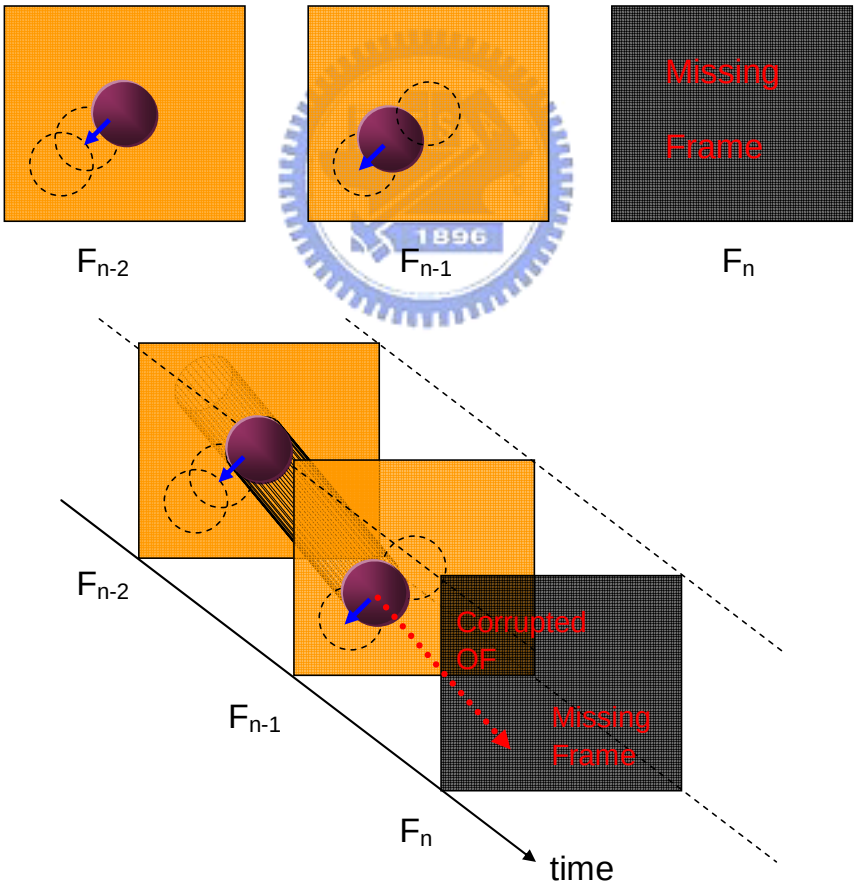


Fig. 3.3 Frame loss and the corruption of optical flow.

3.1.2 Optical Flow Estimation

In last subsection, we mentioned that the temporal motion information of a video sequence can be described by optical flow. A lost frame can be reconstructed based on the recovered OF field. In this section we will introduce the estimation of optical flow.

In optical flow, it is assumed that the horizontal and vertical spatial coordinates of a point may changes over time in the video sequence, but its intensity value does not. A well-known model of optical flow is expressed in Equation (3.1) [27], where s_H and s_V correspond to the horizontal and vertical coordinates in time t, and $x(s_H, s_V, t)$ corresponds to the intensity distribution of the video sequence. Equation (3.2) is the differentiation of (3.1), where $v_H(s_H, s_V, t) = ds_H/dt$ and $v_V(s_H, s_V, t) = ds_V/dt$ correspond to the horizontal and vertical velocity. It means that optical flow is the representation such that the intensity $x(s_H, s_V, t)$ remains constant along a motion trajectory.

$$\frac{dx(s_H, s_V, t)}{dt} = 0 \tag{3.1}$$

$$\frac{\partial x(s_H, s_V, t)}{ds_H} v_H(s_H, s_V, t) + \frac{\partial x(s_H, s_V, t)}{ds_V} v_V(s_H, s_V, t) + \frac{\partial x(s_H, s_V, t)}{dt} = 0 \tag{3.2}$$

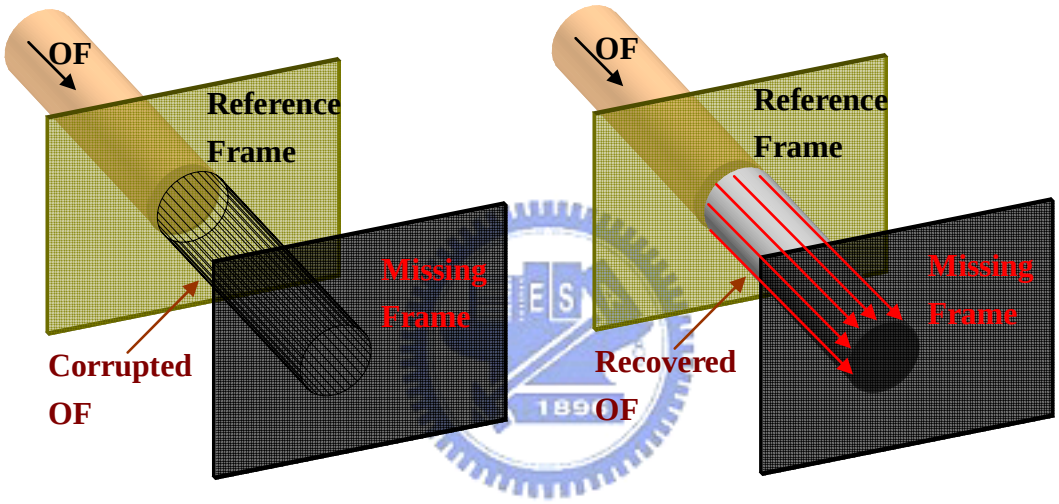
Hence, the optical flow of a vido sequence can be estimated by the equation described above. However, the computational load is too high for concealment in real-time applications. In addition, the coded motion vectors are the available information about the video motion. Therefore, in practice, we usually use MVs to substitute OF.

Moreover, the motion vector estimation in an encoding process is generally based on block matching. The estimated MVs represent the displacement between the coded block and the most matching reference block, instead of the actual motion trajectory. Especially in some cases, such as featureless areas or repeated patterns, the estimated MVs may be quite different from the actual motion trajectory. Hence, once if we estimate the missing frame based on incorrect motion information, the performance of concealment will be seriously degraded. In the following section, we will introduce the proposed MV correction algorithm to overcome this problem.

3.1.3 Motion Recovery of Missing Frame

Depending on motion consistency, the missing motion information can be recovered from previous frames. In this section, we will introduce the motion

recovery of missing frame based on optical flow. Fig. 3.4 shows that when the optical flow of the reference is available, we can extend it to the missing frame to recover the corrupted OF. The motion recovery in [22][23] is based on the temporal consistency of MV. In [23], S. Belfiore *et al.* proposed the generation of MV history (MVH) as shown in Fig. 3.5, which recodes the MV of each pixel from Frame n-1 to Frame n-L (the termination) along the MV trajectory. After recording the MVH, we can further estimate the optical flow with some MVH processing, such as mean average, median filter, or weighted averages with the weights linearly and exponentially decaying over time. In [23], they found that the mean value consistently yields the best result and thus they adopted the linear velocity model.



(a) Original **(b) Motion Recovery**

Fig. 3.4 An illustration of motion recovery.

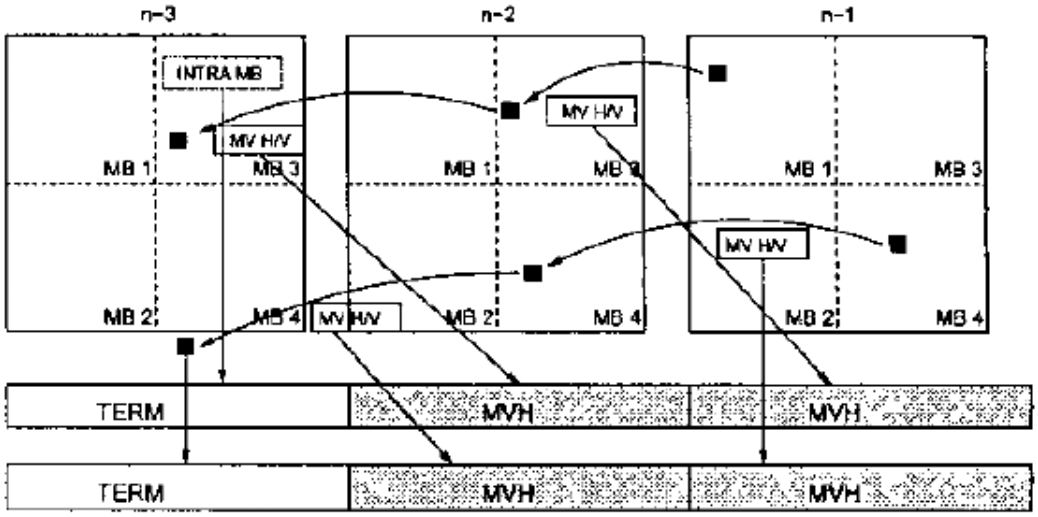
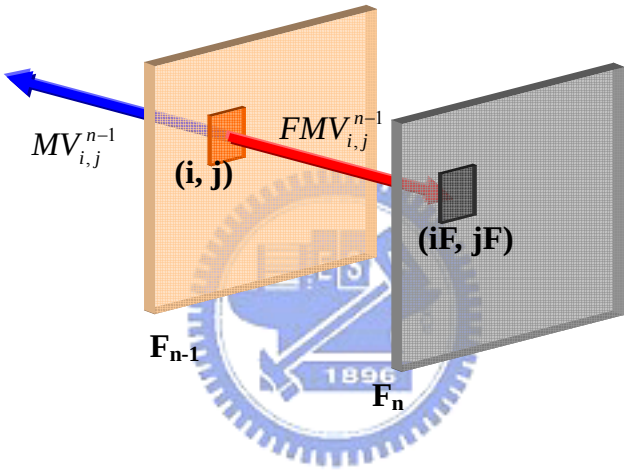


Fig. 3.5 Generation of MV history[23]

Based on the linear velocity model, a lost frame can be reconstructed by forward motion projection. For example, in Fig. 3.6, F_{n-1} is the reference frame of the missing frame F_n, and $MV_{i,j}^{n-1}$ is the motion vector corresponding to the spatial coordinate (i, j) in Frame n-1. Depending on motion consistency, we can inverse the $MV_{i,j}^{n-1}$ and project the inversed MV to (iF, jF) in Frame F_n to replace the projected motion vector $MV_{iF,jF}^{n}$ with $MV_{i,j}^{n-1}$. This is the main concept of motion vector recovery based on forward motion vector projection and the MVs of the lost frame can be estimated in this way. After that, we can reconstruct the lost frame with the recovered MVs by motion compensation.



$$FMV_{i,j}^{n-1} = -MV_{i,j}^{n-1}$$

$$MV_{iF,jF}^{n} = MV_{i,j}^{n-1}$$

$$\text{where} \begin{cases} iF = i + FMV_{i,j,H}^{n-1} \\ jF = j + FMV_{i,j,V}^{n-1} \end{cases}$$

Fig. 3.6 Forward MV projection and MV recovery.

3.1.4 Overall System

After introducing the motion consistency and the motion vector recovery, we will describe the overall system of the error concealment for whole-frame loss. At first, we will introduce the pixel-based concealment algorithm proposed by S. Belfiore *et al.* in [22]. In order to improve the computational complexity, P. Baccicht *et al.* proposed a block-based concealment scheme in [24][25] whose performance is close to that of

pixel-based concealment.

3.1.4.1 Pixel-Based Concealment

The scheme of Belfiore's algorithm is illustrated in Fig. 3.7, where the temporal regularization estimates the OF corresponding to FMV, which the mean value of the motion vector history (MVH) as described in Section 3.1.3. In order to avoid the bad affection of incorrect MVs, a spatial regularization based on the spatial consistency of MV field is applied with a median filter in the FMV field. We will further describe the processing of MV correction in Section 3.2 .

After the temporal regularization and spatial regularization, the reconstruction of the missing frame is performed at half-pixel resolution, as shown in Fig. 3.8. Every pixel of Frame *t-1* contributes to a 2x2 block in Frame *t* with the projection according to the FMV. After the projection, some pixels of the missing frame may not be filled. These missing pixels are recursively interpolated by the output mean of a 9x9 median filter. Finally, a downsampling filter is used to reduce the resolution of the estimated picture by a factor two.

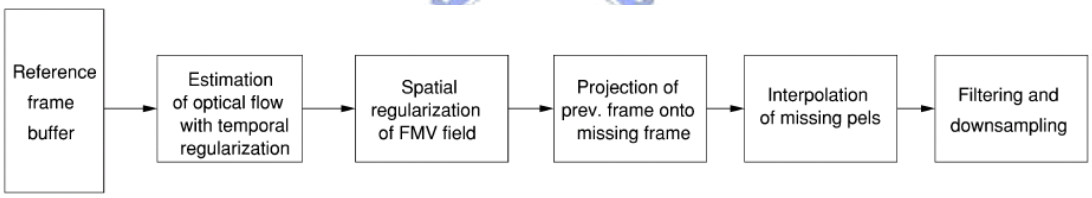


Fig. 3.7 Scheme of Belfiore's algorithm. [22]

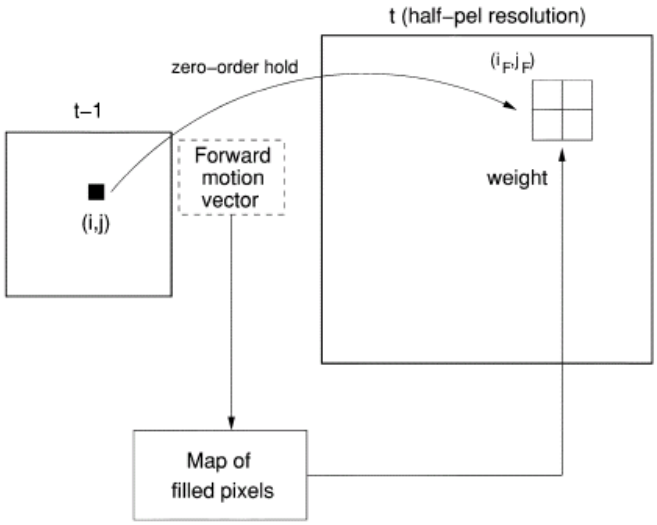


Fig. 3.8 Reconstruction of frame *t* at half-pixel resolution. [22]

3.1.4.2 Block-Based Concealment

The drawback of the pixel-based concealment is the computational complexity. To overcome this problem, block-based concealment offers a good trade-off between the performance and computational cost.

The scheme of Baccicht's algorithm is illustrated in Fig. 3.9. The feature of block-based concealment is the process of motion vector recovery. After forward motion vector projection from Frame *t-1* to Frame *t*, we may estimate the MV field of the missing frame. However, the projected MVs may not fit the block in the missing frame. The recovered MVs in the same block may not be identical. Hence, some additional processing of MV field is applied. After motion vector projection, the statistics for each 16x16 macroblock and each 4x4 block of the Frame *t* is computed as follows:

- M_{16x16} Mean value of projected MVs into a 16x16 MB
- M_{4x4} Mean value of projected MVs into a 4x4 block
- σ_{16x16} Variance value of projected MVs into a 16x16 MB
- σ_{4x4} Variance of projected MVs into a 4x4 block
- N_{16x16} Number of MVs falling into a 16x16 macroblock
- N_{4x4} Number of MVs falling into a 4x4 macroblock

Basing on the collected statistics, the 16x16 (macroblock) level or 4x4 block level motion vector field estimation is performed as follows:

MB-Level MV field estimation:

$$\left(\sigma_{16\times16} < THR_{V_{16\times16}}\right) AND \left(N_{16\times16} < THR_{N_{16\times16}}\right)$$
$$\Rightarrow MV^t = M_{16\times16}$$

Block-Level MV field estimation:

$$\left(\sigma_{4\times4} < THR_{V_{4\times4}}\right) AND \left(N_{4\times4} < THR_{N_{4\times4}}\right)$$
$$\Rightarrow MV^t = M_{4\times4}$$

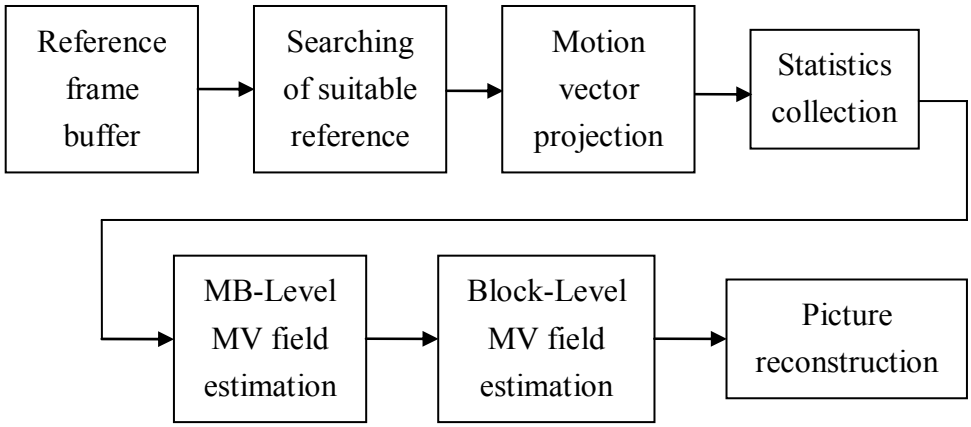
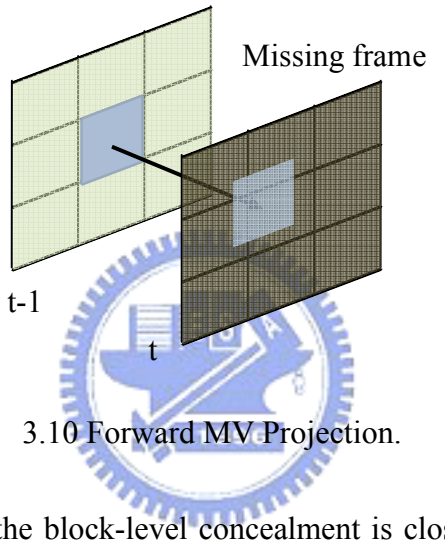


Fig. 3.9 Scheme of Baccicht's algorithm. [25]



3.10 Forward MV Projection.

The performance of the block-level concealment is close to the performance of the pixel-level concealment, as shown in Table 3-1. The simplicity of the block-level concealment makes it preferable for real-time applications.

Table 3-1 Performance of concealment algorithms. [25]

Sequence encoded with the full search motion estimator and one reference frame.

Sequence	QP	QCIF					CIF				
		PSNR			GAIN		PSNR			GAIN	
		Copy	CA_P	CA_B	CA_P	CA_B	COPY	CA_P	CA_B	CA_P	CA_B
Akiyo	23	36.879	37.842	37.630	0.963	0.751	35.53	36.879	36.664	1.349	1.134
	31	34.169	34.719	34.581	0.550	0.412	34.105	35.028	34.896	0.923	0.791
	39	30.093	30.252	30.206	0.159	0.113	31.466	31.844	31.78	0.378	0.314
Coastguard	23	23.531	28.675	28.206	5.144	4.675	21.578	26.010	25.145	4.432	3.567
	31	23.699	27.572	27.201	3.873	3.502	21.767	25.707	25.157	3.940	3.390
	39	23.317	25.036	24.963	1.719	1.646	21.889	24.339	24.120	2.450	2.231
Foreman	23	23.520	25.632	25.397	2.112	1.877	22.136	24.261	23.949	2.125	1.813
	31	23.355	25.261	25.065	1.906	1.710	22.224	24.176	23.930	1.952	1.706
	39	22.776	24.198	24.038	1.422	1.262	22.173	23.772	23.597	1.599	1.424
Table Tennis	23	24.133	27.232	26.233	3.099	2.100	23.242	26.033	24.947	2.791	1.705
	31	23.820	26.655	25.854	2.835	2.034	22.966	25.639	24.737	2.673	1.771
	39	22.970	25.089	24.689	2.119	1.719	22.342	24.526	24.009	2.184	1.667

3.1.5 Problems of Forward Motion Projection

In the frame concealment algorithms described above, the reconstructed frame is under projection. It means that the patches of the missing frame are projected from the reference picture based on FMVs. However, there are two main problems regarding forward motion projection:

1. Non-covered case：

   After motion vector projection, some parts in the motion vector field of the missing frame may not be projected. This causes some unfilled “holes” in the reconstructed picture. Therefore, Baccicht *et al.* investigated two possible solutions to solve this problem in [25]. One is the interpolation performed in the motion vector field to recover unfilled MVs, and the other is the interpolation performed in the estimated picture to fill the holes. They found that the first solution provides better performance

2. Conflict case：

   When performing motion vector projection, two different MVs in the reference frame may project to the same area and cause the conflict problem as shown in Fig. 3.11. For this problem, Zhenyu Wu *et al.* proposed a solution in [26]. They record the covering area of each projected MV. When a conflict happens, they compare the covering area and preserve the MV which owns a larger area, as shown in Fig. 3.12.

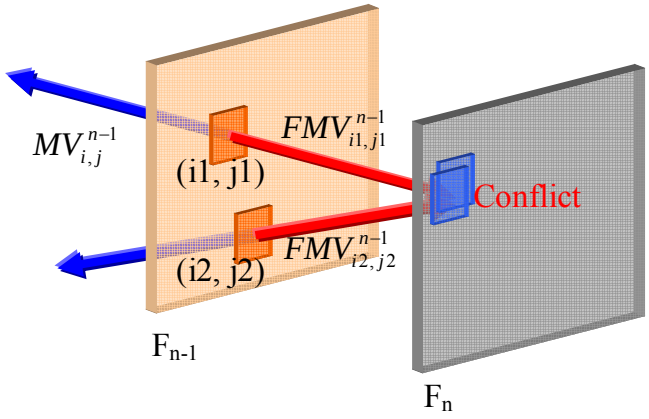


Fig. 3.11 Conflict problem of motion vector projection.

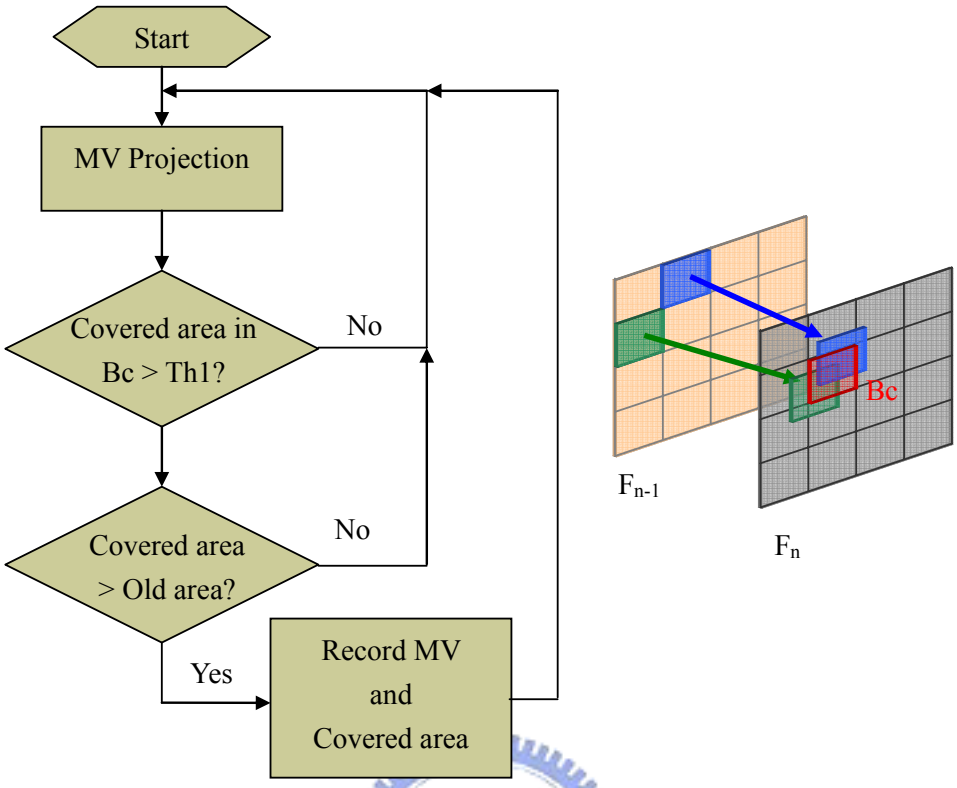


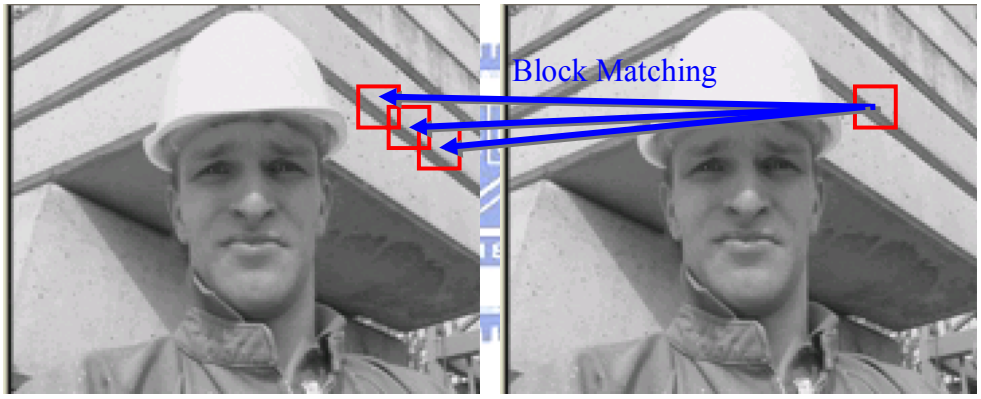
Fig. 3.12 Conflict handling of motion vector projection.

We have described some possible solutions to solve the no-covered and conflict problems. However, a main factor of these problems has not been mentioned yet. We found that the difference between the coded MVs and the actual motion trajectory is a key factor. A wrong MV may project to a wrong position and cause conflict with other projection. Even taking the solution proposed in [26], the larger area covering may not correspond to the correct projection, and thus an incorrect MV may substitute the correct one and cause negative impact over the performance of concealment. Therefore, the accuracy of motion vector greatly influence the performance of concealment. In the next section, we will introduce our motion correction algorithm.

3.2 Proposed Motion Vector Correction Algorithm

3.2.1 Motivation

In last section, we described some existing frame concealment algorithms, which are based on forward motion projection to recover the corrupted motion information. To avoid high computational complexity, the coded MVs are usually used to substitute the OF for concealment. However, as mentioned before, the estimated MV may not correspond to the actual motion trajectory, especially in featureless areas or repeated patterns, as shown in Fig. 3.13. Here, there are more than one suitable block in the reference picture. In this case, the estimated MV may be influenced by some factors, such as luminance change or noise. These factors may cause the deviation of MV from the actual motion trajectory.



Reference Frame
Foreman #4

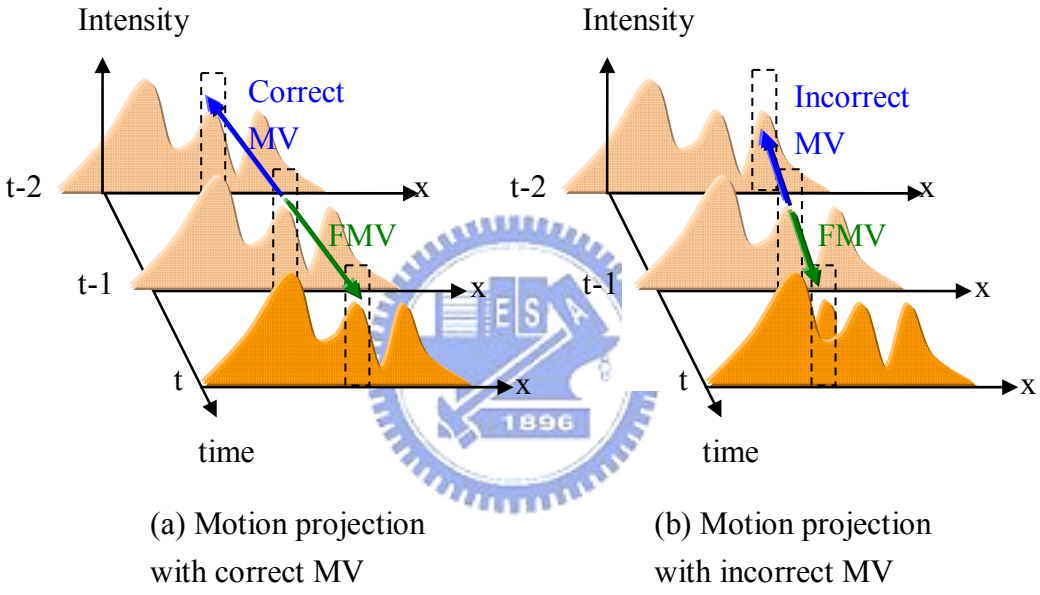
Current Frame
Foreman #5

Fig. 3.13 Motion estimation based on block matching.

For basic encoders and decoders, the difference between MV and the actual motion trajectory does not cause any problem since the purpose of motion estimation is to reduce temporal redundancy. However, when we use MVs to substitute OFs for concealment, incorrect motion information may negatively impact the performance of concealment and cause some aliasing defects in the reconstructed picture. For example, in Fig. 3.14, (a) is the case of motion vector projection with the correct MV to reconstruct the frame at *t*. On the contrary, (b) is the case when the projected MV is deviated from the actual motion trajectory. It can be seen that the estimated picture is filled with incorrect patches. This shows that incorrect MVs may seriously reduce the performance of error concealment.

To solve this problem, existing concealment techniques provide a few solutions. In Section 3.1.4.1, some pixel-based concealment techniques use the median filter to remove the outliers in the FMV field. In fact, the median filtering operation for each pixel increases the computational complexity and makes it unsuitable for real-time applications. In Section 3.1.4.2, each block of the reconstructed MV field is estimated based on statistics. However, the reconstructed MV field is projected from uncorrected MVs of the reference frame. Hence, incorrect MVs may be recorded at incorrectly projected positions. This causes difficulty in the following MV reconstruction.

In the following subsection, we will introduce our MV correction algorithm, which is performed in the original MV field based on fuzzy reliability.



(a) Motion projection with correct MV

(b) Motion projection with incorrect MV

Fig. 3.14 Motion vector projection with (a) a correct MV, and (b) an incorrect MV.

3.2.2 Proposed Fuzzy MV Correction

As mentioned before, moving objects in a video sequence usually obey spatial consistency and temporal consistency. Fig. 3.15 shows an example that one car moves from left to right in the sequence. In a short period, the velocity of the car can be assumed to be constant and we call this fact as temporal consistency. On the other hand, the neighboring parts of the car also move at a similar velocity. We call this fact spatial consistency. Depending on spatial consistency and temporal consistency, we can decide the reliability of the coded MVs. If one MV is inconsistent with the temporally adjacent MVs, the reliability of this MV is lower. Similarly, if one MV is

different from the spatially adjacent MVs, we also give this MV a lower reliability. Hence, we can decide the reliability of each MV based on fuzzy logic and correct these MVs with lower reliability.

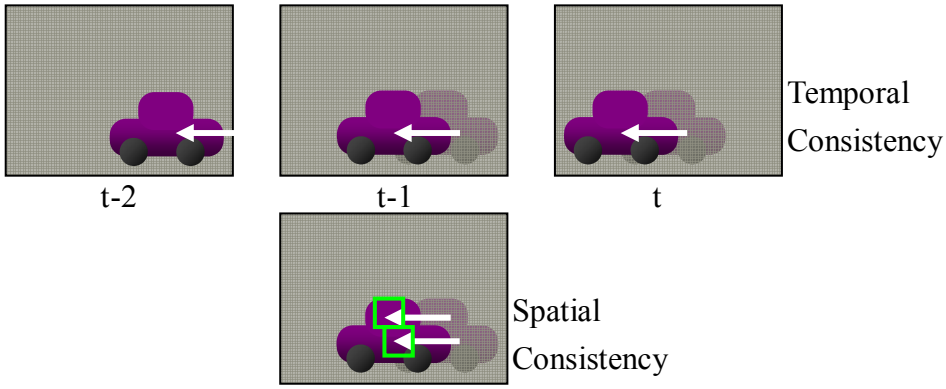
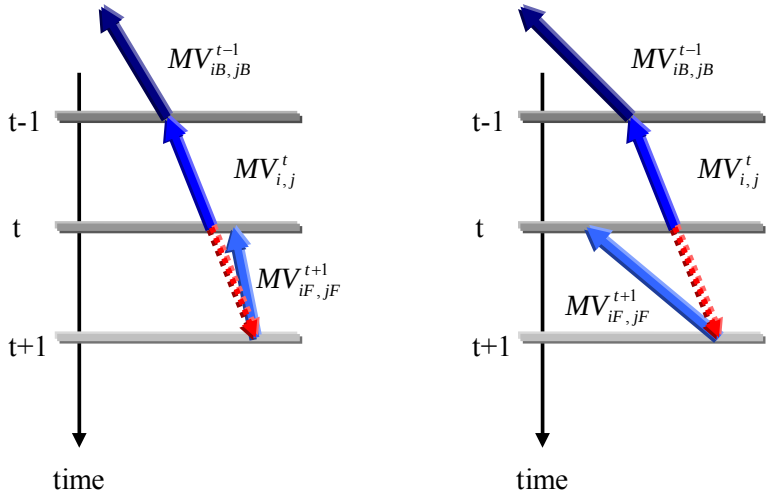


Fig. 3.15 Examples of temporal consistency and spatial consistency.

3.2.2.1 Temporal MV Reliability

The temporal MV reliability can be decided based on the temporal consistency of MVs. We can estimate the temporal difference (TD) as defined in Equation (3.3), which corresponds to temporal inconsistency. Fig. 3.16 illustrates two examples. In Fig. 3.16 (a), the target $MV_{i,j}^{t}$ is close to the mean of the backward projected motion vector $MV_{iB,jB}^{t-1}$ and the forward projected motion vector $MV_{iF,jF}^{t+1}$. Hence, the corresponding reliability is higher. On the contrary, in Fig. 3.16 (b), the temporal difference of $MV_{i,j}^{t}$ is higher and thus the corresponding reliability of $MV_{i,j}^{t}$ is lower.

$$TD_{i,j}^{t} = \left\| 2 \cdot MV_{i,j}^{t} - MV_{iB,jB}^{t-1} - MV_{iF,jF}^{t+1} \right\|$$
$$\text{where} \begin{cases} [iB, jB] = [i, j] + MV_{i,j}^{t} \\ [iF, jF] = [i, j] - MV_{i,j}^{t} \end{cases} \qquad (3.3)$$



(a) $MV_{i,j}^{t}$ → high reliability　　(b) $MV_{i,j}^{t}$ → low reliability

Fig. 3.16 Temporal MV reliability：(a) High reliability；(b) Low reliability.

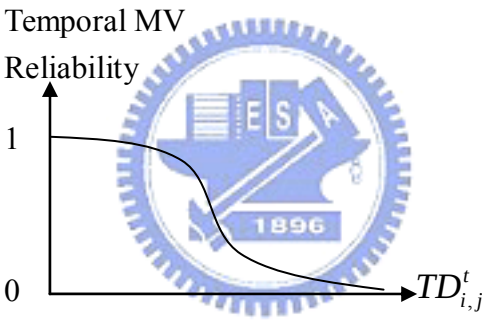


Fig. 3.17 The relation between temporal difference and temporal MV reliability.

3.2.2.2 Spatial MV Reliability

There is generally some degree of spatial consistency among adjacent MVs in the motion vector field. If one MV is quite different from other adjacent MVs, it may possibly be an outlier, as illustrated in Fig. 3.18, and the reliability of this MV is relatively lower.

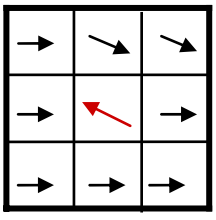


Fig. 3.18 An example of MV field with an outlier MV

In order to measure the difference between a target MV and its adjacent MVs, we estimate the mean value of the surrounding MVs. If the target MV is quite different from the mean MV, it is probably an outlier. On the other hand, if the target MV is between two objects with different motion vectors, the averaged MV cannot well represent the main MV of the surrounding MVs. Hence, we define the measurement of spatial difference to be

$$SD_{i,j}^{t} = \left\| MV_{i,j}^{t} - \overline{MV_{i,j}^{t}} \right\|,$$

$$\overline{MV_{i,j}^{t}} = \begin{cases} \dfrac{\left(MV_{i-1,j}^{t} + MV_{i+1,j}^{t}\right)}{2}, & \text{if } \left\| MV_{i-1,j}^{t} - MV_{i-1,j}^{t} \right\| \le \left\| MV_{i,j-1}^{t} - MV_{i,j+1}^{t} \right\| \\ \dfrac{\left(MV_{i,j-1}^{t} + MV_{i,j+1}^{t}\right)}{2}, & \text{if } \left\| MV_{i-1,j}^{t} - MV_{i-1,j}^{t} \right\| > \left\| MV_{i,j-1}^{t} - MV_{i,j+1}^{t} \right\| \end{cases} \quad (3.4)$$

Here $SD_{i,j}^{t}$ corresponds to the spatial difference between the target $MV_{i,j}^{t}$ and the surroundings and we use $\overline{MV_{i,j}^{t}}$ to represent the mean value of the surrounding MVs. First, we measure both vertical MV difference and horizontal MV difference, and then average the surrounding MVs along the direction with smaller variance. For example, the vertical variance of the left figure of Fig. 3.19 is higher than the horizontal one due to the blue outlier. In this case, the mean value of the left MV and the right MV is defined to be $\overline{MV_{i,j}^{t}}$. With this approach, we can better avoid the influence of other motions in different directions.

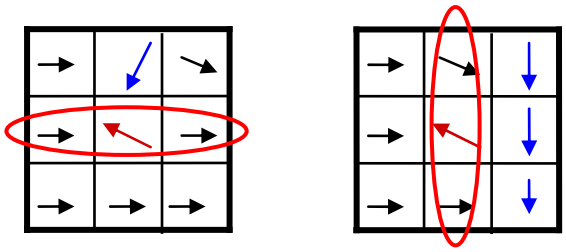


Fig. 3.19 Example of $\overline{MV_{i,j}^{t}}$ measurement.

3.2.2.3 Spatio-Temporal Fuzzy Reliability

After measuring $TD_{i,j}^{t}$ and $SD_{i,j}^{t}$, we can further determine the reliability of the target MV based on fuzzy logic. As shown in Fig. 3.20, when the value of $TD_{i,j}^{t}$ and

$SD_{i,j}^t$ are higher, we give the target MV a lower reliability. On the contrary, the reliability gets higher when both $TD_{i,j}^t$ and $SD_{i,j}^t$ have lower values.

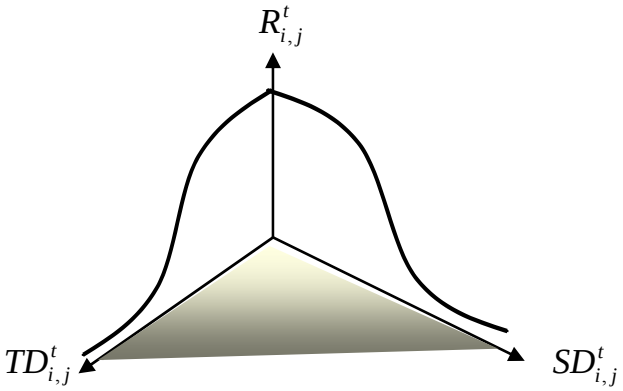


Fig. 3.20 Spatio-Temporal fuzzy reliability.

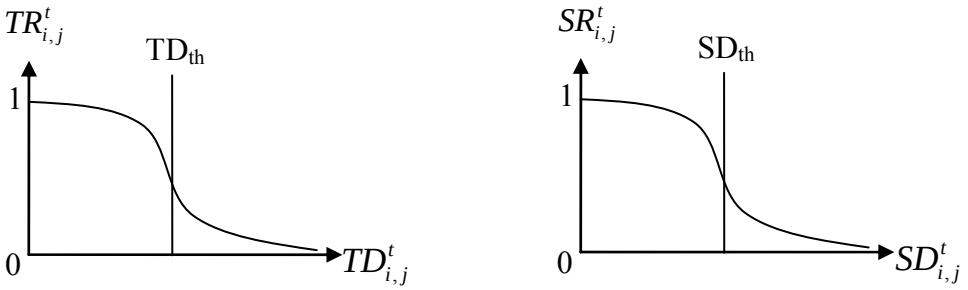
The membership functions for the temporal and spatial reliabilities are defined respectively as below:

$$TR_{i,j}^t = \frac{1}{1+\exp\left(\alpha \cdot (TD_{i,j}^t - TD_{Th})\right)}, \text{ and} \quad (3.5)$$

$$SR_{i,j}^t = \frac{1}{1+\exp\left(\beta \cdot (SD_{i,j}^t - SD_{Th})\right)}, \quad (3.6)$$

where $TR_{i,j}^t$ represents the temporal reliability and $SR_{i,j}^t$ represents the spatial reliability. Depending on these two reliabilities, we can correct MVs of lower reliability by adjacent MVs of higher reliability as described in Equation (3.7), where N represents the neighbors of (i, j) and $MVC_{i,j}^t$ represents the corrected MV.

$$MVC_{i,j}^t = \sum_{i,j \in N} R_{i',j'}^t \cdot MV_{i',j'}^t$$
$$R_{i',j'}^t = \frac{SR_{i',j'}^t \cdot TR_{i',j'}^t}{\sum_{i,j \in N} SR_{i,j}^t \cdot TR_{i,j}^t}, \quad (3.7)$$



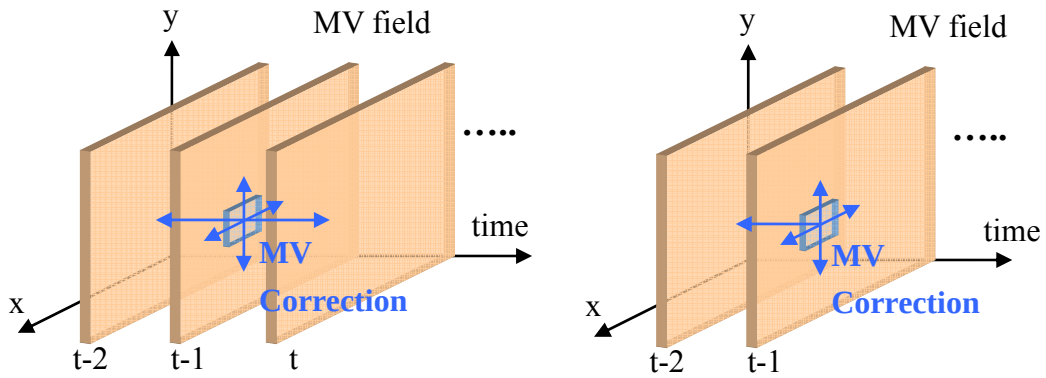
Membership Function of Temporal Reliability

Membership Function of Spatial Reliability

Fig. 3.21 Membership function of temporal reliability and spatial reliability.

3.2.3 Encoder with MV Correction

After introducing the MV correction based on fuzzy reliability, we further describe the encoder scheme with MV correction. The adoption of MV correction increases the performance and the efficiency of concealment in the decoding process. The proposed MV correction can be seen as a spatio-temporal, three dimensional correction, as illustrated in Fig. 3.22(a), where the MV at time *t-1* is under correction with the spatial information and temporal information of the backward and forward reference pictures at time *t-2* and *t*.



(a) Bidirectional MV Correction (b) Backward MV Correction

Fig. 3.22 MV correction in the spatio-temporal MV field.

Due to the use of temporally adjacent MV field, when we correct the MV at time *t-1*, the correctness of the reference MV fields at time *t-2* and *t* may affect the

correction result. Hence, we require that the reference MV fields are also corrected. To achieve that, we propose a two-steps MV correction algorithm, which performs backward MV correction and bidirectional MV correction interactively. The backward MV correction is shown in Fig. 3.22(b), where only the preceding MV field is used for temporal reference. On the other hand, the bidirectional MV correction use both preceding and successive MV fields as temporal references.

The process of the proposed MV correction algorithm is shown in Fig. 3.23, where the horizontal axis corresponds to the time and the vertical axis represents the iterations of correction. For example, in Iteration *k*, the backward MV correction is performed over MV_n with the reference MV_{n-1}. In this figure, the uncorrected MV field is represented in white color while the corrected MV field is colored in orange. In the next iteration *k+1*, the bidirectional MV correction is performed over MV_{n-1} with the reference pictures being the corrected MV_{n-2} and MV_n. With this arrangement, the reference pictures are more reliable for the subsequent corrections.

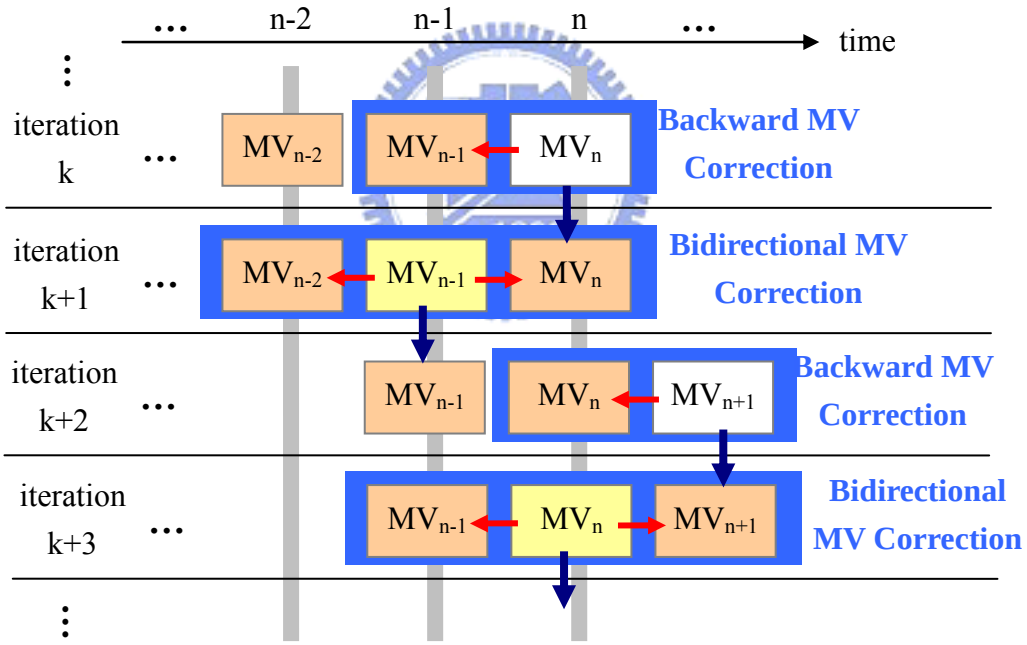
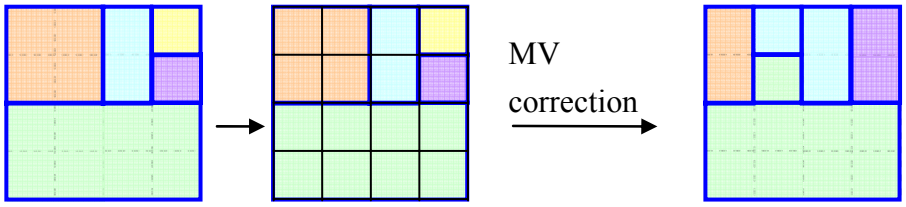


Fig. 3.23 The process of MV correction.

3.2.3.1 Variable Block Size

As H.264 allows the motion estimation to be performed in variable block sizes, our MV correction also supports coding in variable block sizes. Fig. 3.24 shows an example of the MV correction performed in variable sizes. First, the original MV field is transformed to the field organized with 4x4 blocks. After that, the correction is

performed to each basic 4x4 block. After the quantization of the corrected MV field, the adjacent and similar blocks are combined together to form the new corrected MV field with variable block size.



(a) Original MV Field of a MB

(b) 4x4 based MV Field of a MB

(c) Corrected MV Field of a MB

Fig. 3.24 MV correction in variable block size.

3.3 Proposed Error Concealment Algorithm

Back in Section 3.1, we have introduced some existing concealment techniques for whole-frame loss based on motion consistency. In order to make the coded MV close to the actual motion trajectory, we have proposed our MV correction algorithm in Section 3.2. In this section, we will introduce our new error concealment algorithm.

3.3.1 Backward Motion Projection

So far, we have introduced many algorithms for block-level and frame-level concealment. The strategy of these algorithms relies on the reconstruction of the missing parts. However, a major purpose of concealment is actually to relieve the error propagation problem, but not to reconstruct the missing parts only. Hence, in this thesis, we try to skip the corrupted picture and focus on the decoding of subsequent frames, as shown in Fig. 3.25.

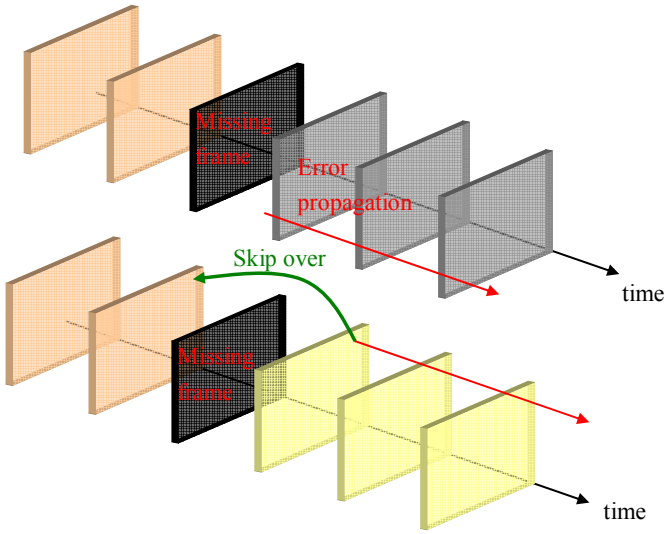
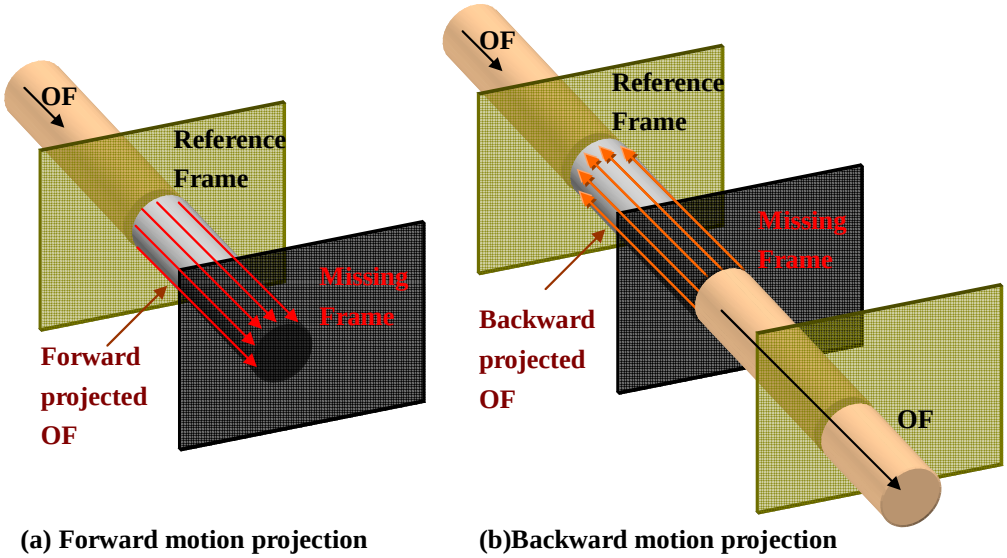


Fig. 3.25 Decoding process by skipping the missing frame.

Based on this concept, we developed a new concealment algorithm that avoids the usage of the erroneous frames as the reference frames. The main innovation of the proposed algorithm is the backward MV projection, as shown in Fig. 3.26. Theoretically, the basic concepts of forward MV correction and backward MV correction are similar. Both are based on the temporal consistency of video motion, and both extend the available MVs to recover the corrupted motion of the lost frame. Hence, the performance of both approaches is nearly equal. However, if considering the computational complexity, the proposed backward projection-based concealment is much simpler.



(a) Forward motion projection **(b)Backward motion projection**

Fig. 3.26 MV projection: (a) forward; (b) backward.

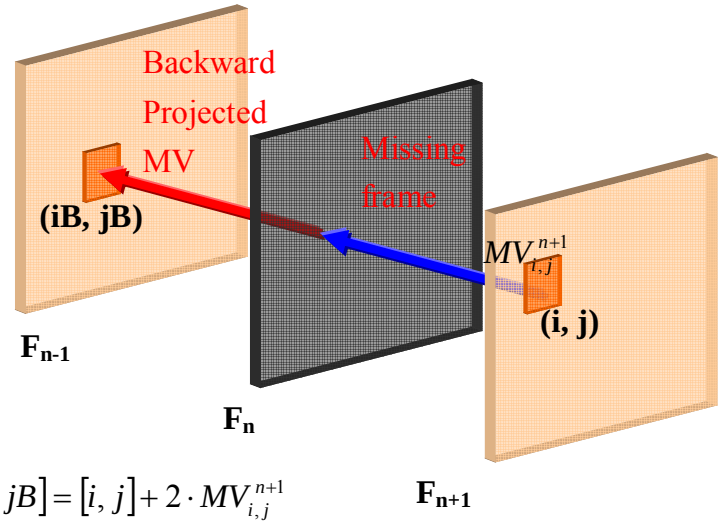


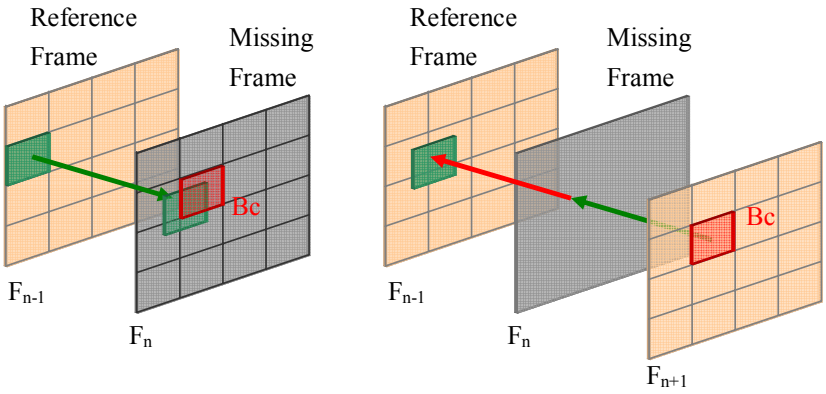
Fig. 3.27 The concealment based on backward MV projection.

The proposed concealment algorithm is illustrated in Fig. 3.27, where F_n is the missing frame and F_{n+1} is the subsequent frame. F_{n+1} is decoded with the reference of F_n. In order to reduce the bad influence of the erroneous frame, traditional methods attend to reconstruct the lost picture. In our approach, however, we isolate the erroneous picture and decodes the F_{n+1} with the reference of F_{n-1} by using backward MV projection. Hence, as illustrated in Fig. 3.27, we only need to extend the MV field of F_{n+1} by a factor of two, and then perform the motion compensation operation to decode the subsequent frame.

3.3.2 Comparison between Forward MV Projection and Backward MV Projection

In forward MV projection, the recovered MV field is “passive” and is projected from the MV field of the reference frame. As shown in Fig. 3.28(a), most projection blocks do not exactly coincide the blocks in the recovered MV field. Hence, additional processes, as described in 3.1.4.2, are needed in forward MV projection. Moreover, the aforementioned non-covered and conflict problems also affect the performance of concealment.

If considering backward MV projection, we simply extend the MV field of the frame right after the corrupted one, as shown in Fig. 3.28(b). We do not need these additional processes in forward MV projection and we can greatly reduce the computational cost.



(a) Forward MV Projection (b) Backward MV Projection

Fig. 3.28 Comparison between forward MV projection and backward MV projection.

The greatest benefit of our proposed concealment is its simplicity in implementation. When a frame loss occurs, we only focus on the subsequent frame and extend its MV field by a normal factor. Moreover, our proposed algorithm is independent of the use of intra frame. If the concealment is base on forward MV projection, the corrupted frame is reconstructed from the extended MV of the preceding frame, as shown in Fig. 3.29. When the preceding frame is an intra frame, there is no available MV for MV projection and some exceptional handling is required to handle the missing of MVs. On the contrary, since the reconstruction is based on the subsequent frame, there is no such problem in backward MV projection. In other words, no matter the subsequent frame is an inter frame or intra frame, the concealment process can be performed correctly without any exceptional handling.

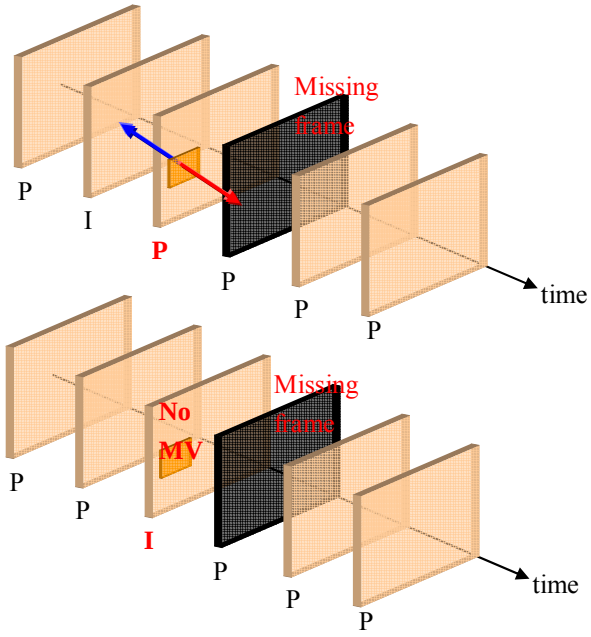


Fig. 3.29 The intra frame problem in forward MV projection.

3.4 CODEC with Resilience and Concealment

We have described the proposed error resilience method based on MV correction in Section 3.2. We also described the proposed concealment algorithm in Section 3.3. On one hand, error resilience affects the performance of concealment; while on the other hand, error resilience may reduce the coding efficiency. Hence, the interaction between error concealment and error resilience, as illustrated in Fig. 3.30, should be taken into consideration. In this section, we will discuss the interaction between the encoder with the proposed resilience method and the decoder with the proposed concealment method.

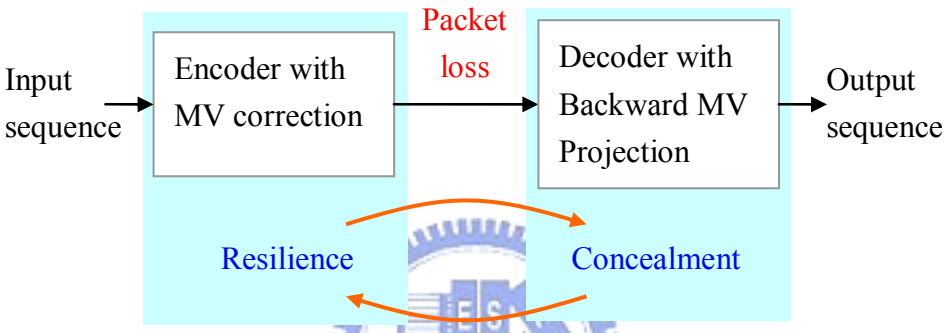
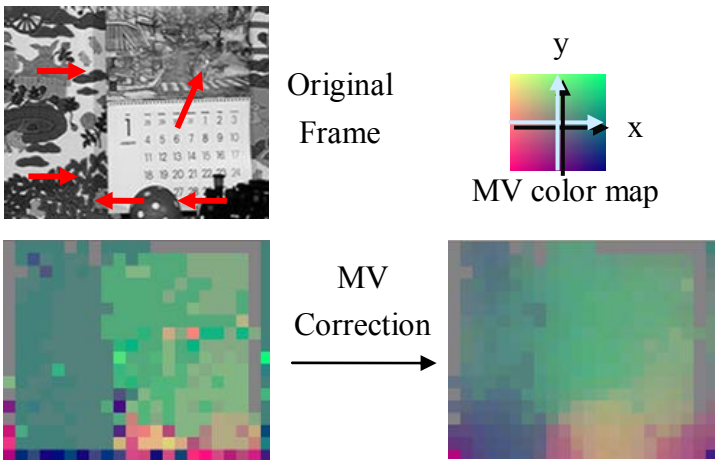


Fig. 3.30 The interaction between the proposed resilience method at the encoder with and the proposed concealment method at the decoder.

3.4.1 The Relation between MV Correction and MV Projection

First, we discuss the performance improvement of error concealment with respect to the use of error resilience. Fig. 3.31 shows the difference between the MV projection with and without MV correction. We can find that the MV field is sprawling before MV correction and may cause incorrect MV projections in the concealment process. On the contrary, the corrected MV field is smoother and more coherent for projection. There are two reconstructed frames in Fig. 3.32, where the left one is reconstructed based on the original MVs. We can find that there are many obvious artifacts in the picture. Compared with the right one, which is based on the corrected MVs, the artifacts are efficiently reduced.



Resilience

Concealment

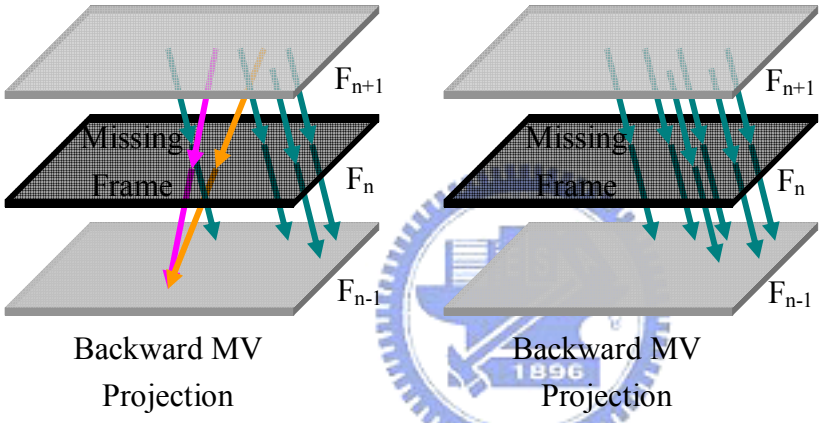


Fig. 3.31 The MV field and MV projection:
(left) without MV correction; (right) with MV correction.

Fig. 3.32 The reconstructed frame:
(left) without MV correction; (right) with MV correction.

3.4.2 Coding Efficiency with MV Correction

The purpose of the proposed MV correction is to make the coded MVs approaching the actual motion trajectory. However, since the actual MVs may not be the MVs that minimize the residual of motion compensation, the coding efficiency may be reduced after MV correction. Fig. 3.33 shows the PSNR of motion compensation of the foreman sequence. The blue line is the PSNR corresponding to the original MVs, while the red one corresponds to the PSNR based on the corrected MVs.

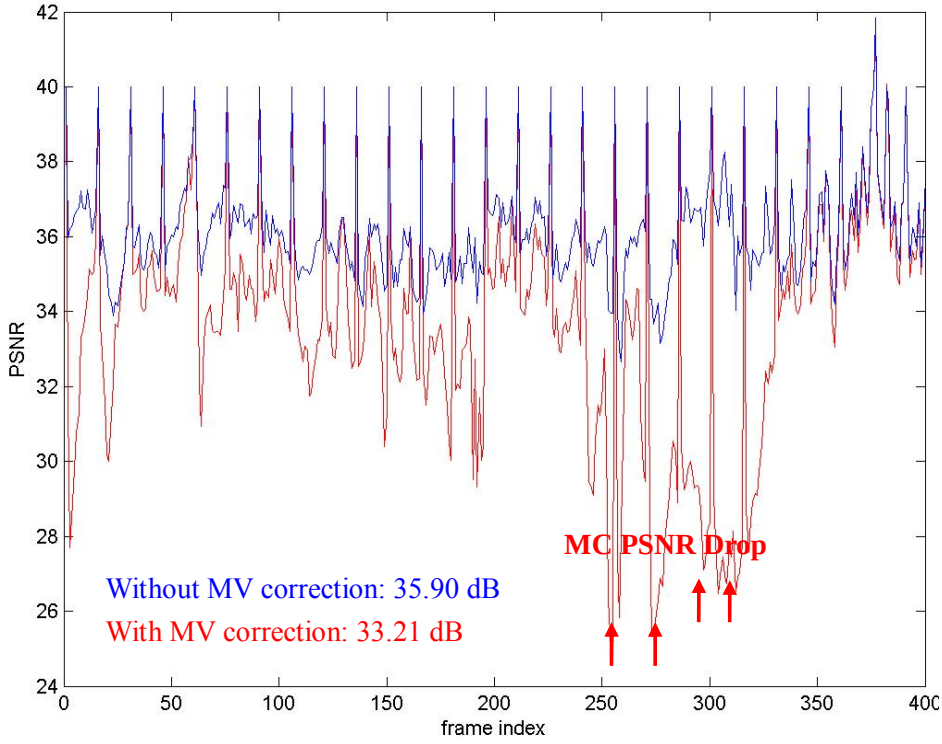


Fig. 3.33 PSNR of motion compensation. (foreman sequence; intra period 15; QP 28)

As we can predict, the PSNR may be brought lower after MV correction. The main reason is that the video motion does not always obey the the linear motion model. Sometimes, the video objects may be under warping, occlusion, or zooming. In these cases, the reference block corresponding to the corrected MVs may not match the coded block. Moreover, because the proposed MV correction is based on motion consistency, if the actual motion deviates from this assumption, such as a violent acceleration or scene change, the corrected MVs may be very different from the actual motion. In this case, the MV correction process may cause serious PSNR drops, as shown in Fig. 3.33, and the coding efficiency is seriously degraded.

3.4.3 Partial MV Correction

As mentioned above, coding efficiency may dramatically drop due to incorrect MV correction for non-linear video motions. The concealment process also suffers from the similar problem, since concealment also relies on the assumption of motion consistency. When the video motion is non-linear, the proposed MV correction process not only cannot work well but only reduces the coding efficiency. Hence, we need to stop MV correction at these cases.

To overcome this problem, we proposed a partial MV correction algorithm, as illustrated in Fig. 3.34. We compare the residual of each block before and after MV correction. If the sum of the absolute value of the residual after correction is greater than that before correction by a factor α, it means the target motion may not obey the linear velocity assumption. In this case, the MV correction process is stopped. Here, the controlling factor α decides the amount of the MV correction. If we have a smaller α, the correction is with more restriction.

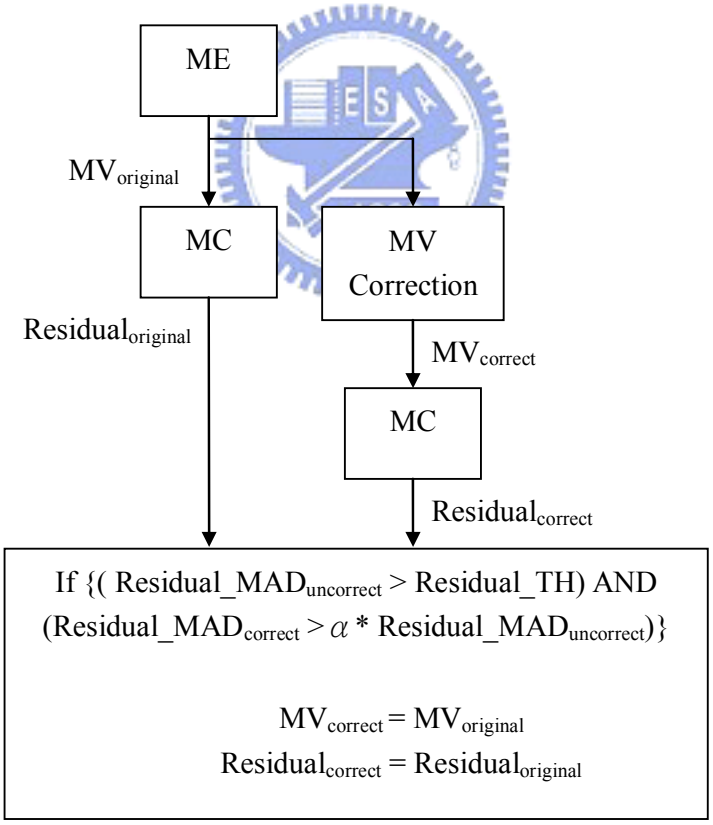


Fig. 3.34 The scheme of the proposed partial MV correction algorithm.

The decision of the α value is a trade-off between coding efficiency and concealment performance. Here we propose a decision rule for the selection of α. Fig.

3.35 shows a normal GOP structure containing IPPP…frames. We know that when a frame loss occurs, the error effects drift till the next new intra frame. Therefore, when a error occurs more far away from the next intra frame, the error propagation problem gets more serious. In other words, the importance of the concealment is higher for the P frame with a larger distance from the next I frame. Hence, the corresponding α is given a higher value, as shown in Fig. 3.35. In one GOP, we have an α varying from high to low according to the frame-order to obtain a better trade-off.

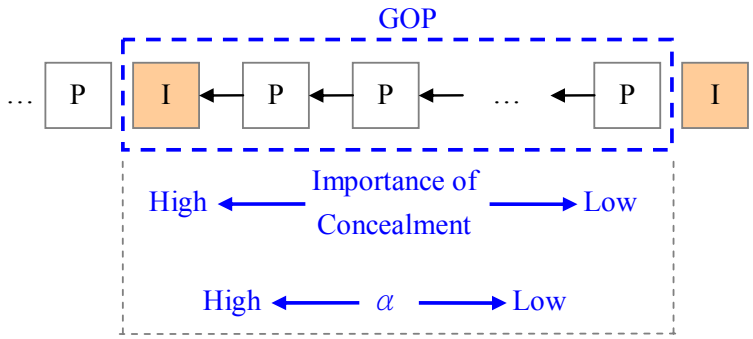


Fig. 3.35 The decision of the value of α in a GOP.

3.5 Resilience and Concealment for GOP with B Frames

In previous discussion of this thesis, a GOP follows the normal IPPP… structure. However, if considering a GOP containing B frames, the problem of frame loss will be different from that we have discussed before. Hence, we also propose an error resilience and error concealment algorithm for the GOP with B frames.

3.5.1 Frame Loss of the GOP with B slices

The normal P-frame inter prediction is to take one preceding frame as the reference. Instead of using one reference frame only, the B frame may be predicted from one or two reference pictures, which are before or after the current frame in the temporal order. Fig. 3.36 shows three examples: (a) one previous and one future reference, (b) two previous references and (c) two future references. Therefore, when the B frame is taken into consideration in the GOP, the order pf the coded frames is different from that of the normal IPPP… structure. In addition, the corresponding error propagation problem is also different from the previous discussion.

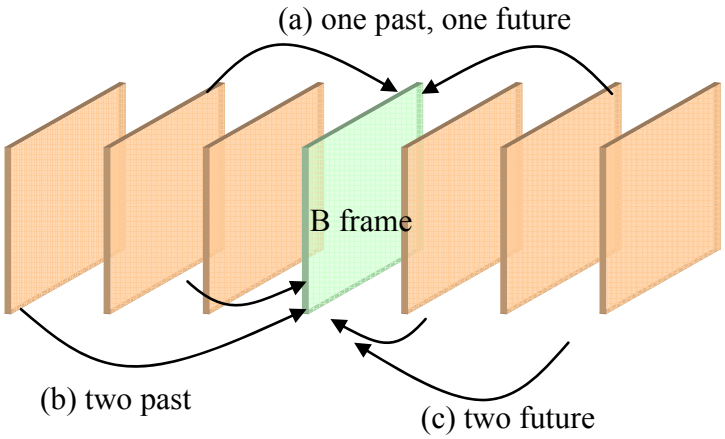
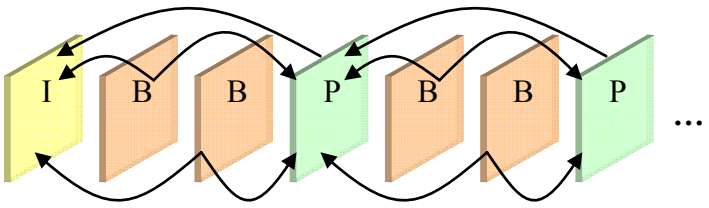
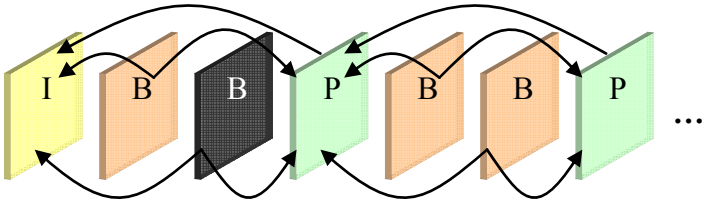


Fig. 3.36 The prediction examples of B frame: (a) past/future, (b) past, (c) future

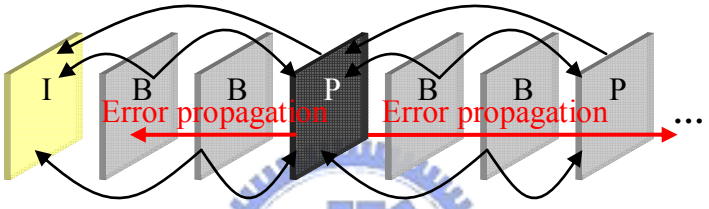
We have the examples of error propagation occurring in the GOP with B frames in Fig. 3.37, where (b) is the case of one lost B frame. Because the sunsequent pictures will not take the erroneous frame as the reference picture, the lost B frame does not cause the error propagation problem. However, if the lost frame is a P frame, as shown in (c), it not only influences the subsequent B frames predicted from the lost frame, but also the successive P frames. The error propagation problem gets more serious in this case.



(a) Original sequence



(b) one B slice is lost



(c) one P slice is lost

Fig. 3.37 Error propagation due to frame loss: (a) original sequence, (b) B frame loss, (c) P frame loss.

3.5.2 Concealment Based on Forward Motion Projection

First we discuss the application of the concealment based on forward MV projection. The influence of B frame loss is slight for the decoded sequence. Hence, we first focus on the case of P frame loss.

When the decoding frame is lost, in order to relieve error propagation, we usually attempt to recover the lost picture based on the information of previously decoded frames. In the case of a P frame loss, as shown in Fig. 3.38(a), the last available information is the last decoded P frame. Hence, when applying the forward MV projection, we project the MV of the preceding P frame to the current frame as shown in Fig. 3.38(b).

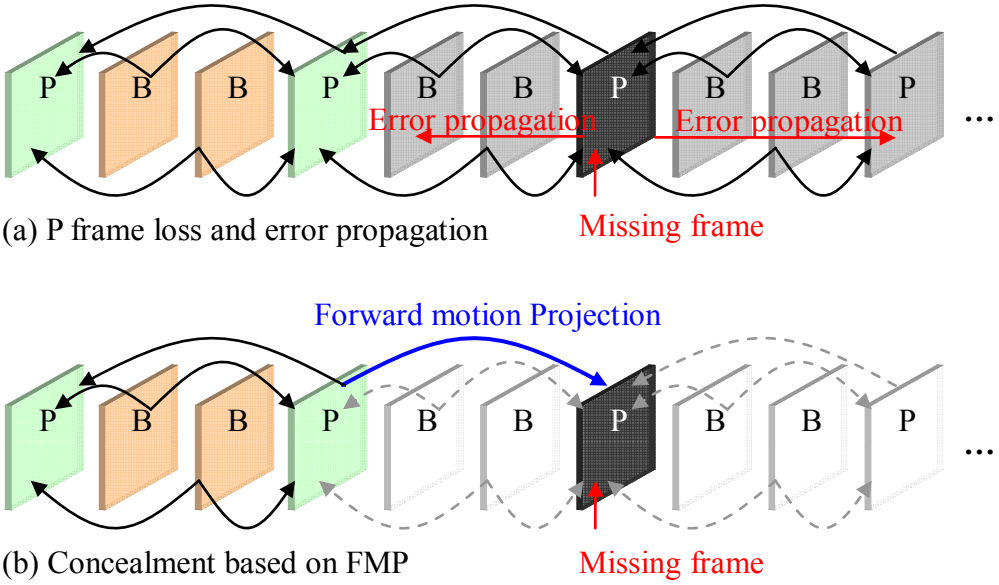


Fig. 3.38 P frame loss and the concealment based on forward MV projection.

This method is similar to which described in Section 3.1. However, the major difference is that the projection will cross through some B frames between the two P frames, as shown in Fig. 3.38(b). In other words, the distance of the projection is larger than that we have described before. The assumption of linear velocity is established within a short interval. The assumption is no longer true as the interval gets longer. We have an example in Fig. 3.39, where (a) is the case of a short projection, and (b) corresponds to a long projection. We can find that it is hard to work for long time-interval. In addition, the performance of concealment based on MV projection is also restricted by the distance of the motion projection.

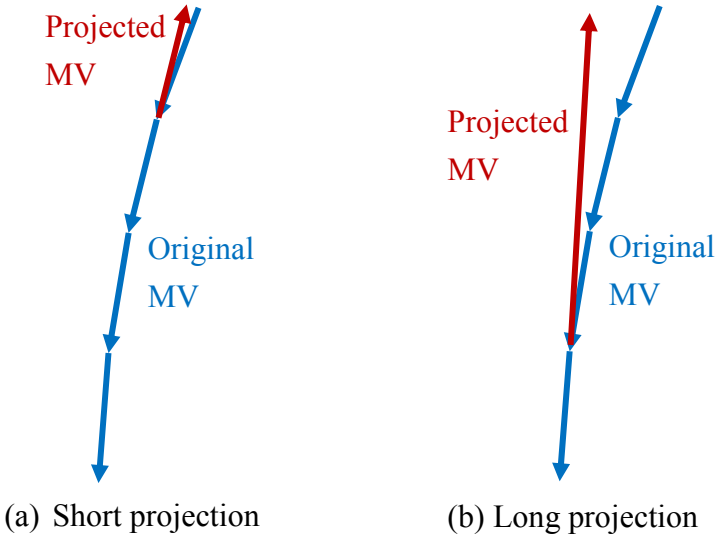


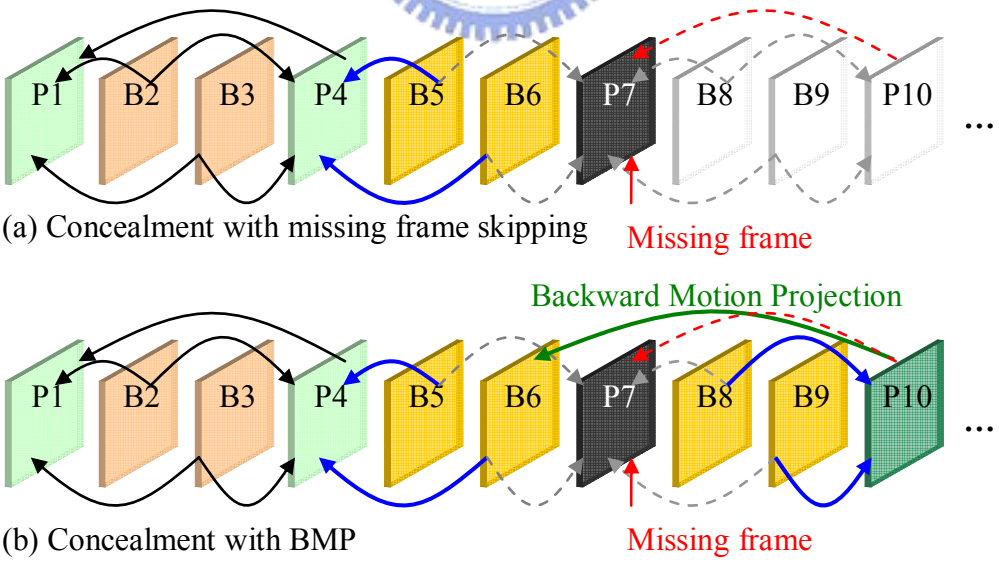
Fig. 3.39 The MV projection of (a) short interval and (b) long interval.

On the other hand, our proposed MV correction algorithm is also suffering from the problem that the MV is out of the liner motion trajectory due to the long interval. Hence, the correction may not help to promote the performance of the concealment.

As described above, it is really a tough problem to conceal the lost P frame when the GOP contains B frames. Moreover, the P frame loss may cause serious error propagation. In order to overcome this problem, we proposed a new GOP structure and a corresponding concealment algorithm.

3.5.3 Proposed Concealment Algorithm

In the last section, we have mentioned that the P frame loss may seriously degrade the video quality and cannot be easily concealed. Again, similar to the concept of the backward motion projection, we wonder if we can avoid the erroneous picture and decode the subsequent pictures by using other correctly received pictures. Based on this concept, we examine the problem again. We give an example in Fig. 3.40, where P7 is the missing frame. Though P7 is lost, B5 and B6 are the B frames with bidirectional MVs. We can disuse the MV that points to the lost P7, as shown in Fig. 3.40(a), and takes only the P4 as the reference picture. However, the next P frame, P10, also requires the reference P7. Hence, we apply the backward motion projection to skip over P7 and use B6 as the substitute, as shown in Fig. 3.40(b). After that, the following B8 and B9 also only take P10 as reference, instead of using P7 as reference.



(a) Concealment with missing frame skipping

(b) Concealment with BMP

Fig. 3.40 The proposed concealment for the GOP with B frames.

However, there is still the same problem about the long projection interval in the

proposed concealment technique. The original MV of P10 points to P7 through the interval of the B frames, and the long projected MV may probably be out of the actual motion trajectory. If the projection is performed with this incorrect MV, the predicted picture may be not what we wish. To solve this problem, we further propose a new GOP structure described in the next subsection.

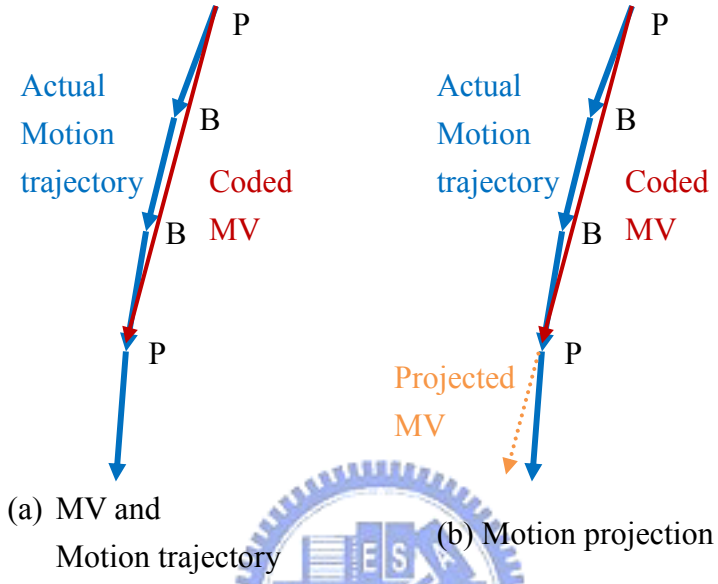


Fig. 3.41 Backward motion projection with the MV of long time-interval.

3.5.3.1 Proposed GOP Structure

The proposed GOP structure is illustrated in Fig. 3.42(a), where we separate the connection between the successive P frames. We know that the P frames always be predicted from the last P frame or I frame, as shown in Fig. 3.38. This is the reason of the serious error propagation when a P frame is lost. In our proposed GOP structure, instead of taking the last P frame as reference, the current P frame is predicted from the B frame before the last P frame. For example, in Fig. 3.42(a), P7 is predicted from B3, instead of P4. Hence, if P4 is lost, B2 and B3 can avoid P4 and can be decoded only with Y1. After that, P7 is decoded from B3, and the following B5 and B6 are also decoded from P7 only, as shown in Fig. 3.42(b). Furthermore, we can find that the lost P4 is isolated, and the error propagation can be relieved.

The feature of the proposed GOP and the corresponding concealment is the isolation of the lost P frame. This is because the concealment of P frame is difficult. On the contrary, since the B frame contains bidirectional MV, the predicted result is acceptable with only one-side MV.

On the other hand, the P frame of the proposed GOP structure is predicted from the preceding B frame. Hence, if the reference B frame is lost, the decoding of the following P frame will be influenced. Therefore, we further discuss the problem of B frame loss. First, we divide the coded B frame into two types: *Significant B frame* and *Insignificant B frame*. The former is the B frame which is used to predict the next P frame, as B0, B3, and B6 in Fig. 3.42. If these frames are lost, the following P frame will not be decoded correctly. This is why we call them Significant B frames. The other B frames, which are not used to perform the inter prediction, are called *Insignificant B frame.* The loss of Insignificant B frame does not cause the error propagation problem. For the loss of Insignificant B frame, as B5 in Fig. 3.42(c), we simply copy the preceding picture P4 to substitute the lost frame B5. However, when a Significant B frame is lost, as shown in Fig. 3.42(d), the following P7 cannot be decoded if B3 is lost. In this case, we need to conceal B3 by the forward MV projection with the information of B2. Since B3 is adjacent to B2, the performance of the concealment is better and the error propagation problem can be better relieved.

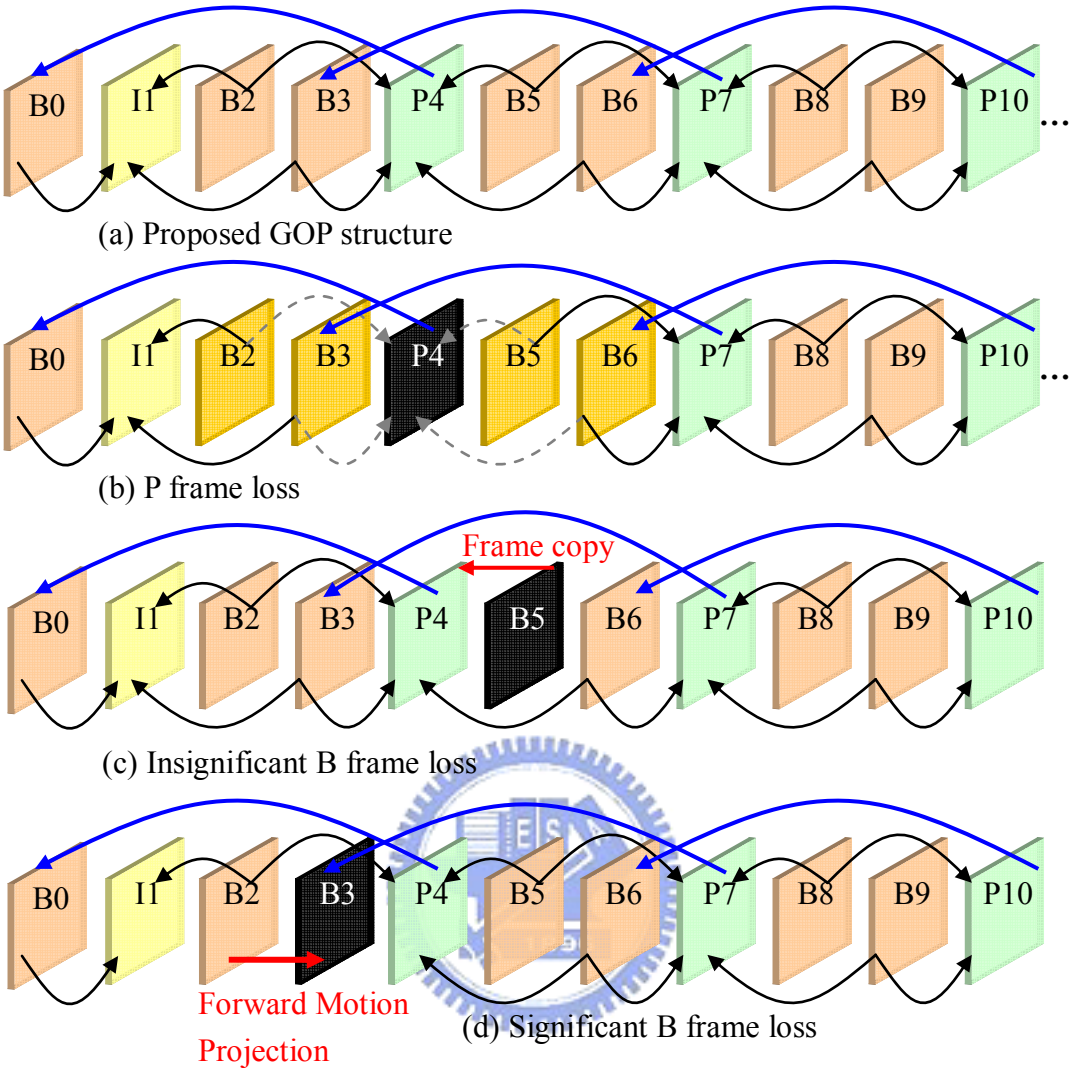


Fig. 3.42 The proposed GOP structure and the corresponding concealment algorithm.

Chapter 4 Experiments & Results

In this chapter we discuss the influence of packet loss upon the quality of received H.264 videos, which are transmitted over wireless networks. The schematic diagram of our experiment is illustrated in Fig. 4.1, where the test sequence is encoded following the H.264 standard. The encoded video data are transmitted over the network simulator as described in 2.2.2.1 with different packet loss rates. After the simulation of video transmission, we obtain received video data, which are suffered from packet loss. Then the decoding process is performed with error concealment. Finally, we will compare the reconstructed sequence with the original sequence to evaluate the performance of the concealment and resilience algorithms.

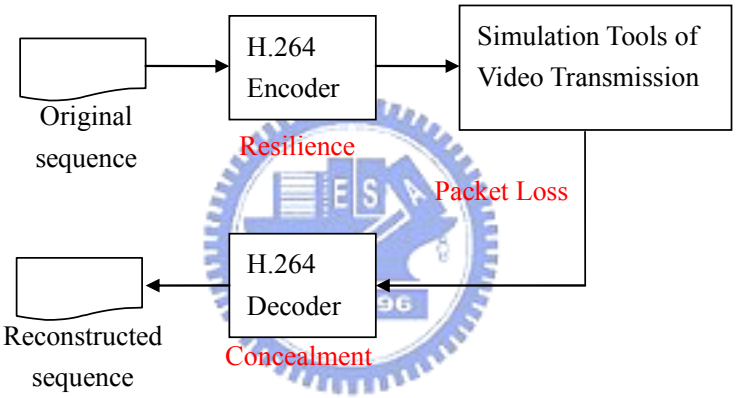


Fig. 4.1 The schematic diagram of our experiment.

In Section 4.1, we will apply the proposed concealment based on backward MV projection to reconstruct missing frames. The performance of the proposed concealment will be compared with the performance of the existing concealment based on forward MV projection. Furthermore, the MVs used in Section 4.1 do not go through MV correction. In Section 4.2, the MV correction process will be performed and we will discuss the enhancement based on error concealment with corrected MV. However, since the coding efficiency may be lowered due to the MV correction process, we will discuss the residuals of motion compensation with and without MV correction in Section 4.3. Besides, the performance of the proposed partial MV correction will also be discussed. Finally in Section 4.4 we will discuss the GOP with B frames and compare the performance between some existing concealment algorithms and the proposed algorithm.

4.1 Concealment without MV Correction

In this section, we perform the proposed concealment algorithm based on backward MV projection to reconstruct the received sequence. We'll do comparison with respect to the error concealment based on forward MV projection. In **Experiment 1**, we transmit the coded Forman sequence in different packet loss rates. Table 4-1 lists the corresponding number of lost frames. In addition, the error patterns are generated based on the Gilbert-Elliot loss model. In the case of a high packet loss rate, there may be successive frame losses. This case may even increase the difficulty of error concealment.

Experiment 1:
Foreman QCIF, QP 28, Baseline Profile, fps 15 Hz, Intra period 15, Total 400 frames.
Error Model: Gilbert-Elliot loss model.

Table 4-1 The number of lost frames, Foreman sequence.

PLR 2.8%	{79, 115, 317, 318, 319, 325, 327, 331, 360, 369, 376, 378, 384}
PLR 5.5%	{78, 79, 115, 116, 198, 310, 317, 318, 319, 321, 325, 327, 331, 355, 360, 361, 366, 367, 369, 376, 378, 384}
PLR 7.4%	{78, 79, 105, 106, 107, 108, 115, 116, 198, 310, 317, 318, 319, 321, 325, 326, 327, 329, 331, 355, 356, 360, 361, 366, 367, 369, 376, 378, 381, 384}

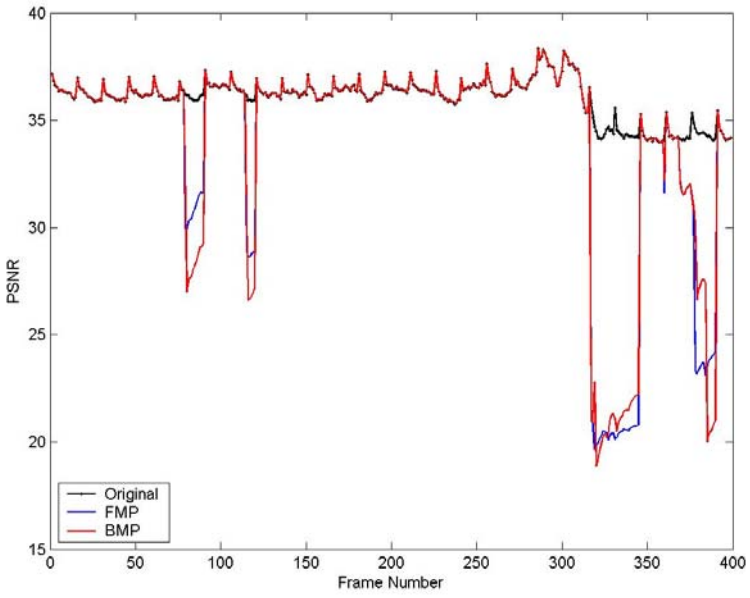


Fig. 4.2 Performance comparison on Foreman sequence, (Packet loss rate 2.8%)

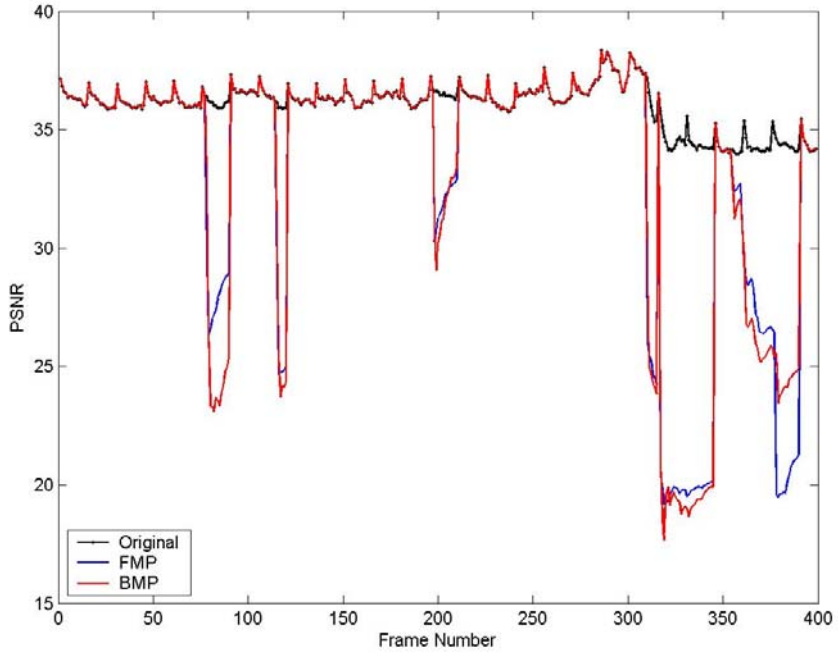


Fig. 4.3 Performance comparison on Foreman sequence, (Packet loss rate 5.5%)

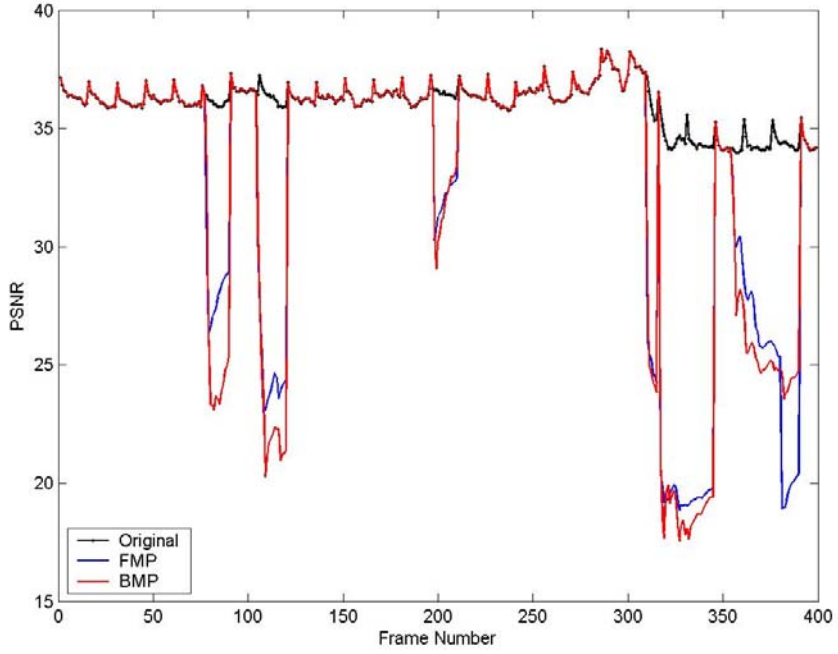


Fig. 4.4 Performance comparison on Foreman sequence, (Packet loss rate 7.4%)

Frame No. 81

Frame No. 120

Fig. 4.5 The reconstructed frames of the Foreman sequence based on forward MV projection (left) and backward MV projection (right), (Packet loss rate 5.47%)

Considering the concealed frames, as shown in Fig. 4.5, we can find that, due to incorrect MVs, there is serious aliasing in the concealed frame, especially in some featureless areas, such as the white hat, and some repeating patterns, such as the oblique edges on the wall. In these cases, the coded MV may probably be deviated from the actual motion trajectory and these incorrect MVs cause the quality reduction of the reconstructed frames.

Experiment 2:

Flower QCIF, QP 28, Baseline Profile, fps 15 Hz, Intra period 15, Total 250 frames

Error Model: Gilbert-Elliot loss model

Table 4-2 The number of lost frames, Flower sequence.

PLR 3.6%	{12, 19, 24, 38, 42, 47, 49, 54, 60, 121, 125, 126, 244}
PLR 6.2%	{12, 17, 19, 20, 24, 31, 38, 42, 47, 49, 54, 60, 61, 75, 76, 87, 121, 125, 126, 128, 244}
PLR 10.1%	{12, 17, 19, 20, 23, 24, 31, 38, 42, 47, 49, 54, 60, 61, 69, 75, 76, 78, 81, 87, 121, 123, 124, 125, 126, 128, 181, 203, 211, 244, 250}

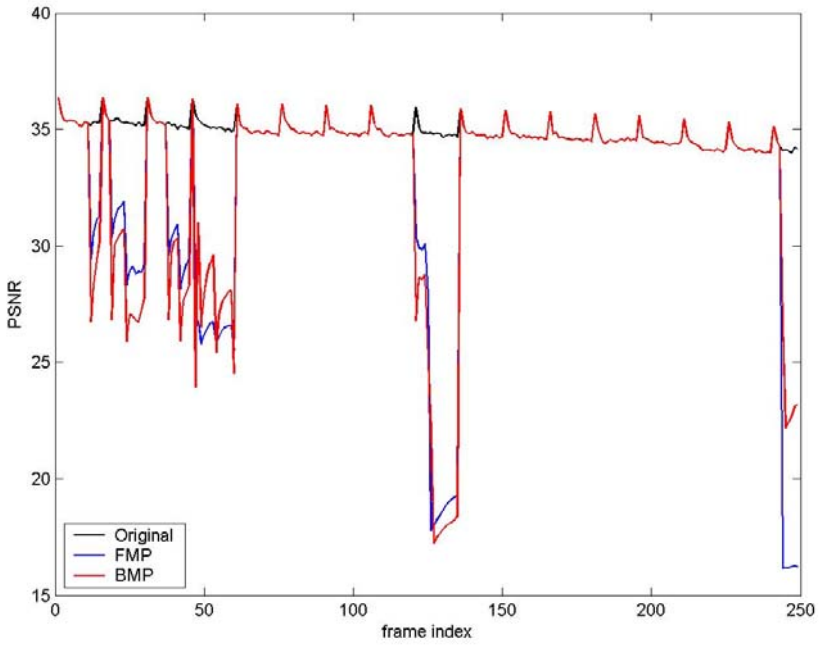


Fig. 4.6 Performance comparison on Flower sequence, (Packet loss rate 3.6%)

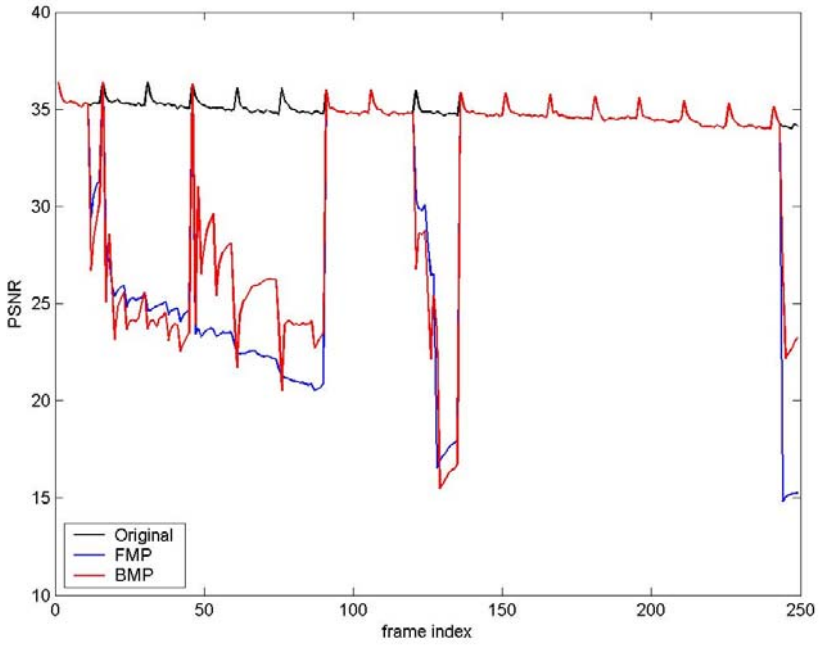


Fig. 4.7 Performance comparison on Flower sequence, (Packet loss rate 6.2%)

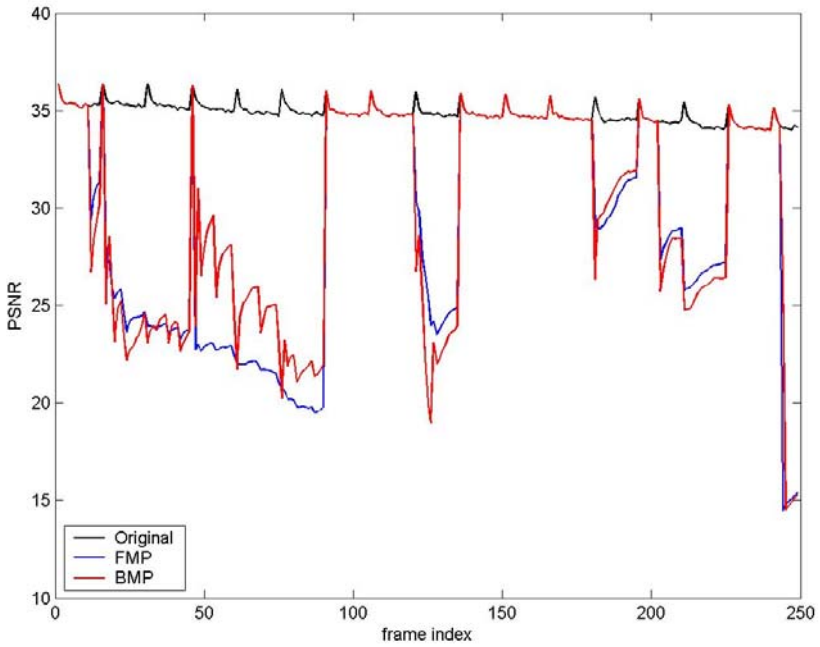


Fig. 4.8 Performance comparison on Flower sequence, (Packet loss rate 10.1%)

Frame No. 22

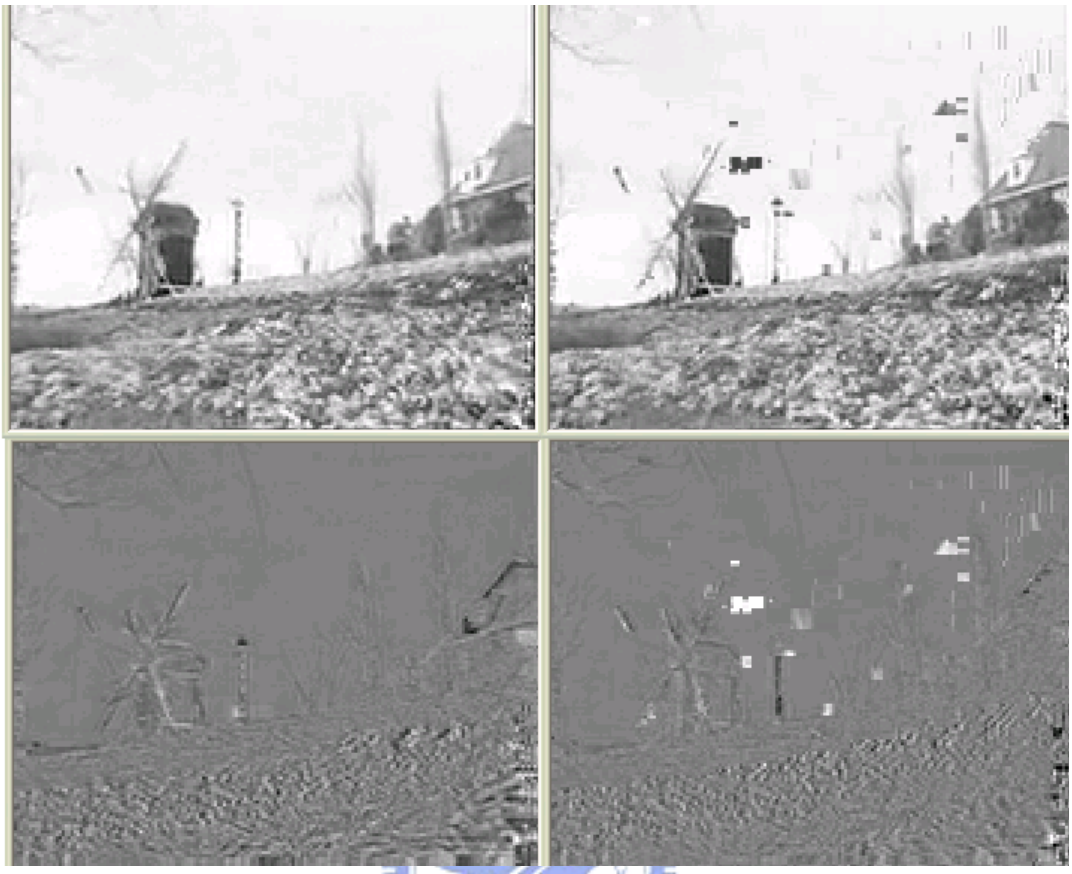


Fig. 4.9 Reconstructed frames and the residual of the Flower sequence. (top left) the reconstructed frame based on FMP; (top right) the reconstructed frame based on BMP; (bottom left) the residual based on FMP; (bottom right) the residual based on BMP. (Packet loss rate 10.1%)

Considering the flower sequence, we can also find that there are still some artifacts in the featureless areas due to the incorrect MVs, as shown in Fig. 4.9 (top right). In Experiment 2, there are some lost frames which are right after an intra frame. For these cases, we cannot perform the normal FMP process since there are no reference MVs. For example, in Table 4-2, the lost 17th and 47th frames of the flower sequence are right after an intra frame and we can only take some earlier frames as the reference frames to predict the lost frames. On the contrary, error concealment based on BMP does not suffer from this problem and may provide better performance, as shown in Fig. 4.9.

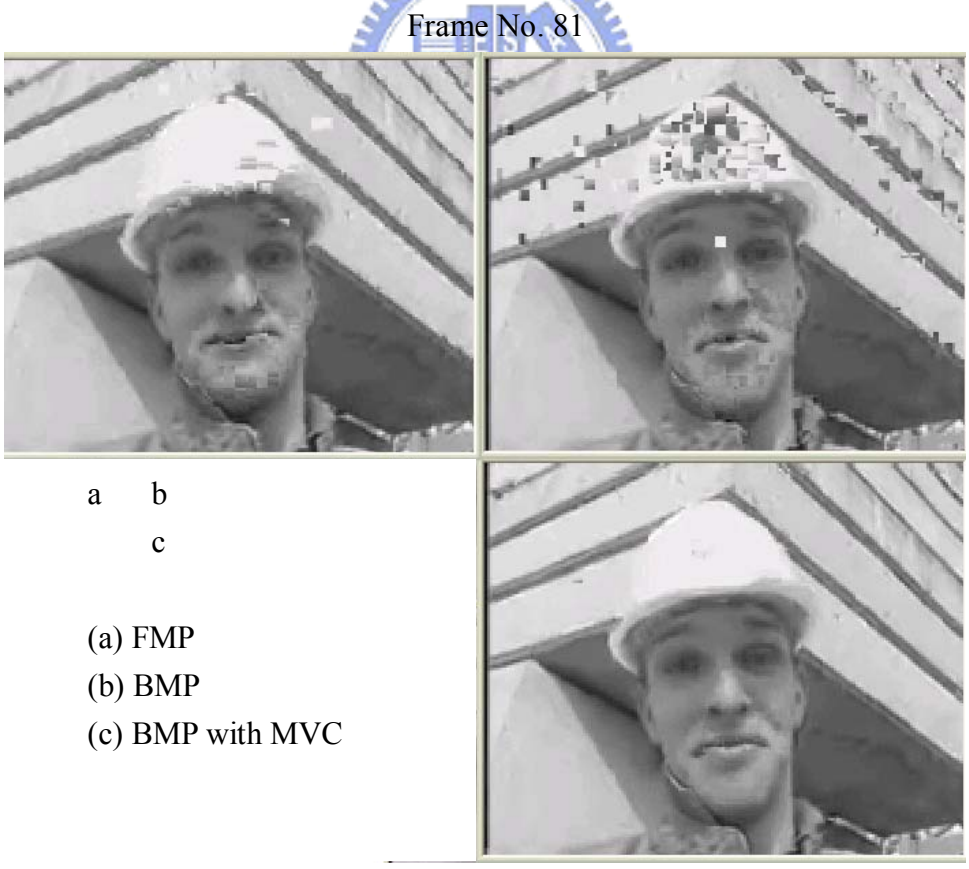
Table 4-3 Performance comparison of the PSNR at different packet loss rates.

Sequence	pB	PLR	FMP	BMP	BMP-FMP
foreman	0.1	2.84%	34.32	34.30	-0.02
	0.2	5.47%	33.43	33.37	-0.06
	0.3	7.44%	33.07	32.87	-0.19
flower	0.1	3.64%	32.66	32.68	0.03
	0.2	6.16%	30.14	30.70	0.57
	0.3	10.08%	29.18	29.51	0.33

4.2 Concealment with MV Correction

The MVs used in Section 4.1 are uncorrected and there are many artifacts in the reconstructed frames. In this section, we apply MV correction and perform comparison over the performance of error concealment.

Fig. 4.10 shows the reconstructed frames based on FMP, BMP, and BMP with MV correction. After MV correction, serious artifacts in (b) can be effectively removed, as shown in Fig. 4.11.



(a) FMP
(b) BMP
(c) BMP with MVC

Fig. 4.10 The reconstructed frames of the Foreman sequence, (Packet loss rate 5.47%)

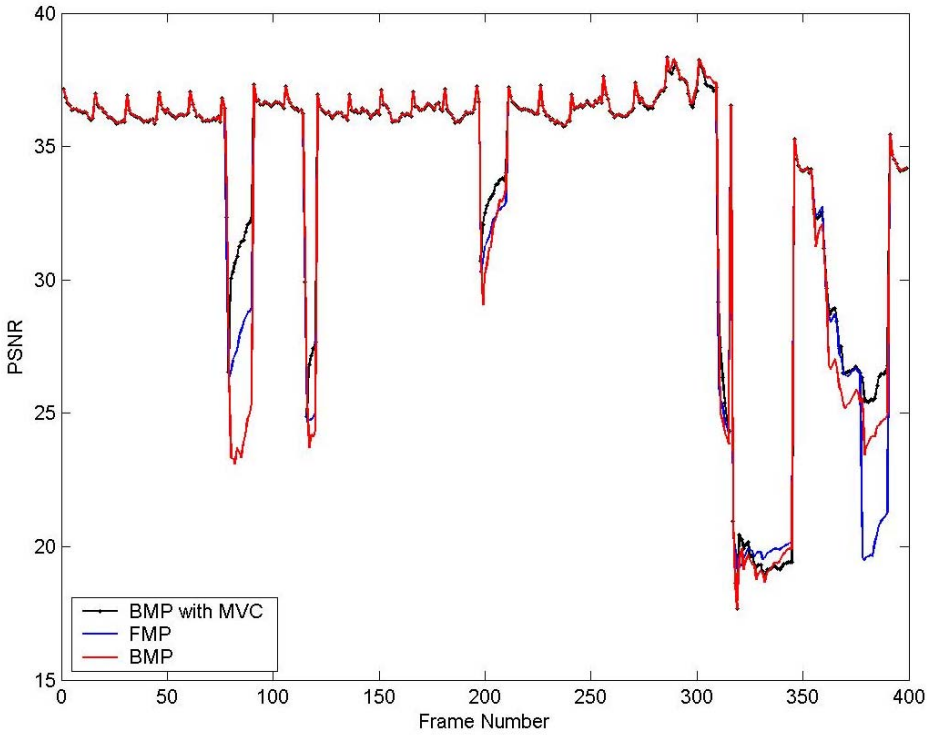


Fig. 4.11 Performance comparison on the Foreman sequence, (Packet loss rate 5.5%)

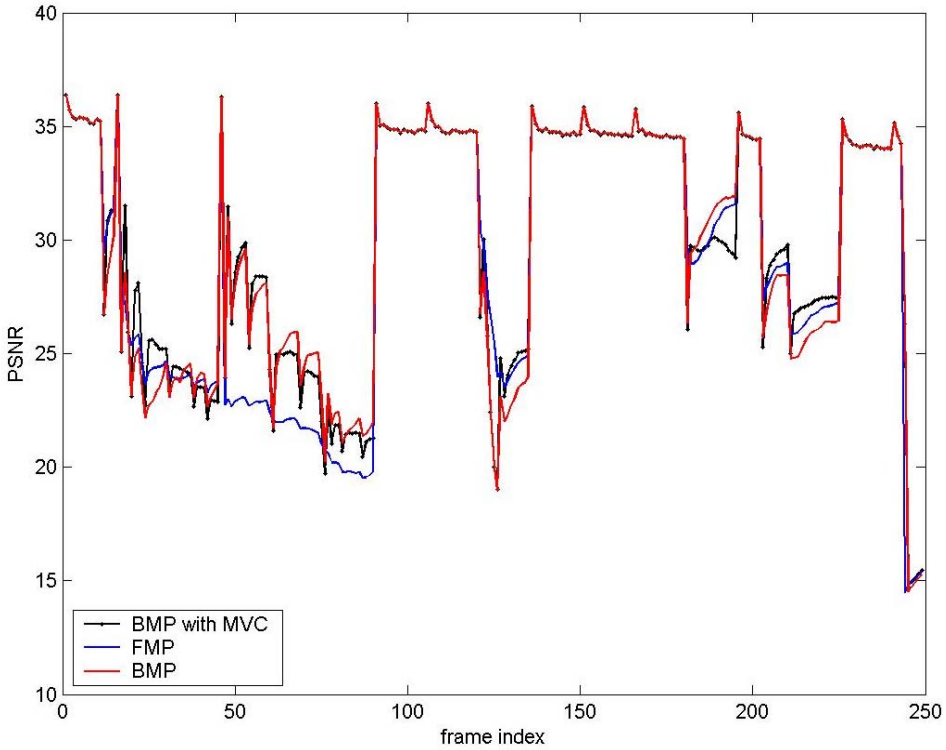


Fig. 4.12 Performance comparison on the Flower sequence, (Packet loss rate 10.1%)

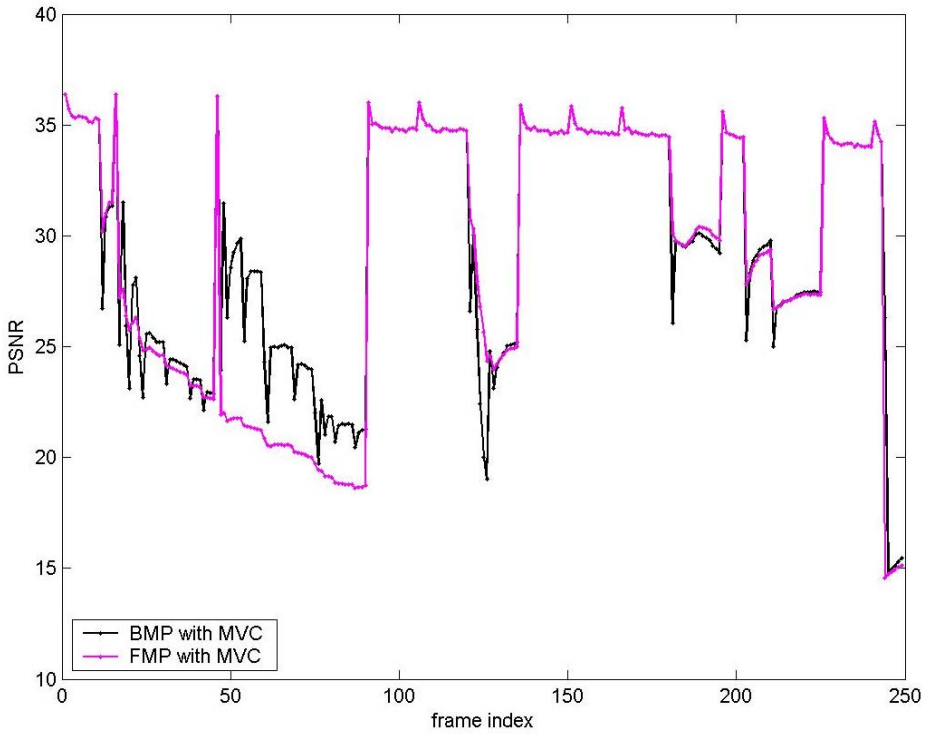
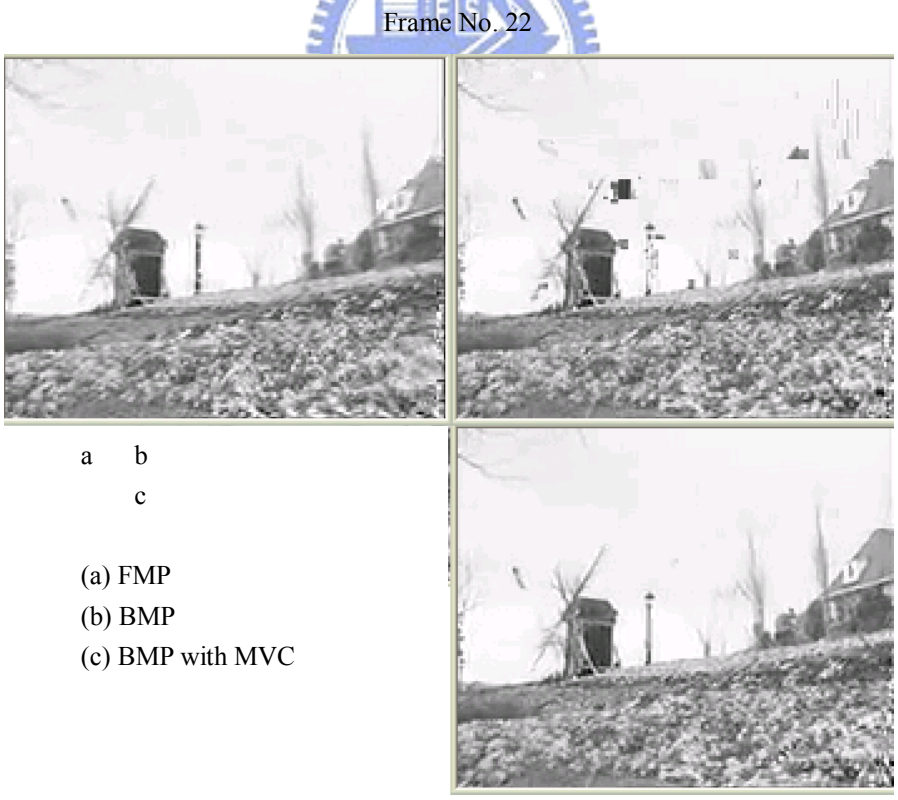


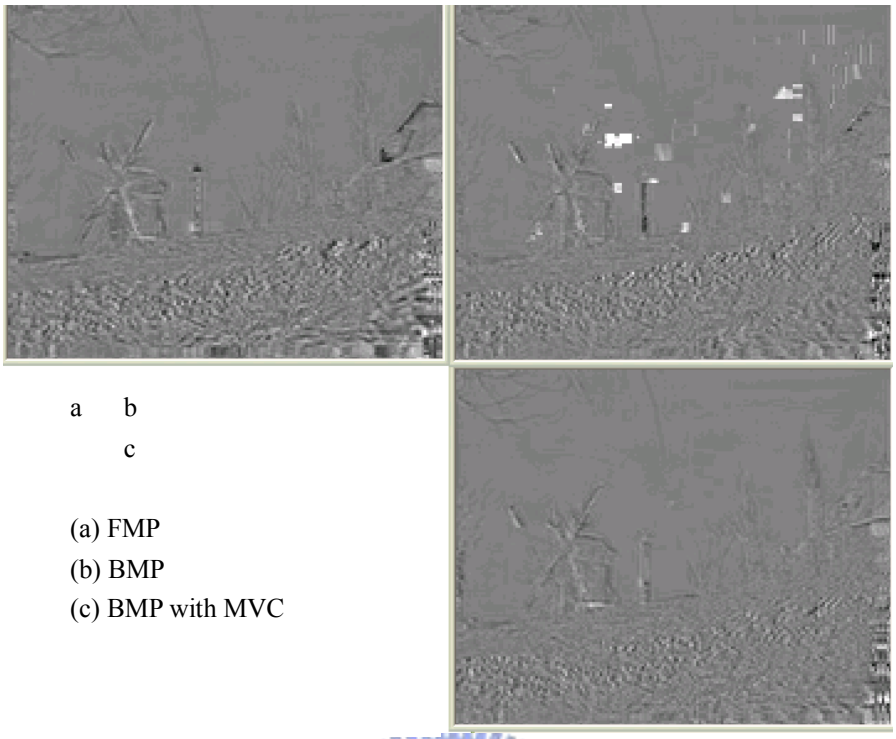
Fig. 4.13 Performance comparison on the Flower sequence, (Packet loss rate 10.1%)



(a) FMP
(b) BMP
(c) BMP with MVC

Fig. 4.14 The reconstructed frames of Flower sequence, (Packet loss rate 10.1%)

Frame No. 22



a b

c

(a) FMP
(b) BMP
(c) BMP with MVC

Fig. 4.15 The residual of reconstruction on the Flower sequence, (Packet loss rate 10.1%)

Table 4-4 Performance comparison of the PSNR at different packet loss rates.

Sequence	pB	PLR	Without MVC		With MVC	
			FMP	BMP	FMP	BMP
foreman	0.1	2.84%	34.32	34.30	34.42	34.55
	0.2	5.47%	33.43	33.37	33.62	33.77
	0.3	7.44%	33.07	32.87	33.25	33.35
flower	0.1	3.64%	32.66	32.68	32.76	32.85
	0.2	6.16%	30.14	30.70	30.06	30.77
	0.3	10.08%	29.18	29.51	29.00	29.66

4.3 Partial MV Correction

In last section we have discussed the enhancement of the concealment with MV correction. However, the adoption of MV correction may reduce the coding efficiency at the encoder site. Here, we further compare the coding efficiency corresponding to MV correction and partial MV correction.

First, for the foreman sequence and flower sequence, we estimate the PSNR of the motion compensation residual with and without MV correction, as shown in Fig. 4.16 and Fig. 4.17, We find that, in Fig. 4.16, there are serious PSNR drops when the motion is away from the expected linear velocity. The motion of the flower sequence is relatively stable. Hence, it suffers less PSNR drop, as shown in Fig. 4.17.

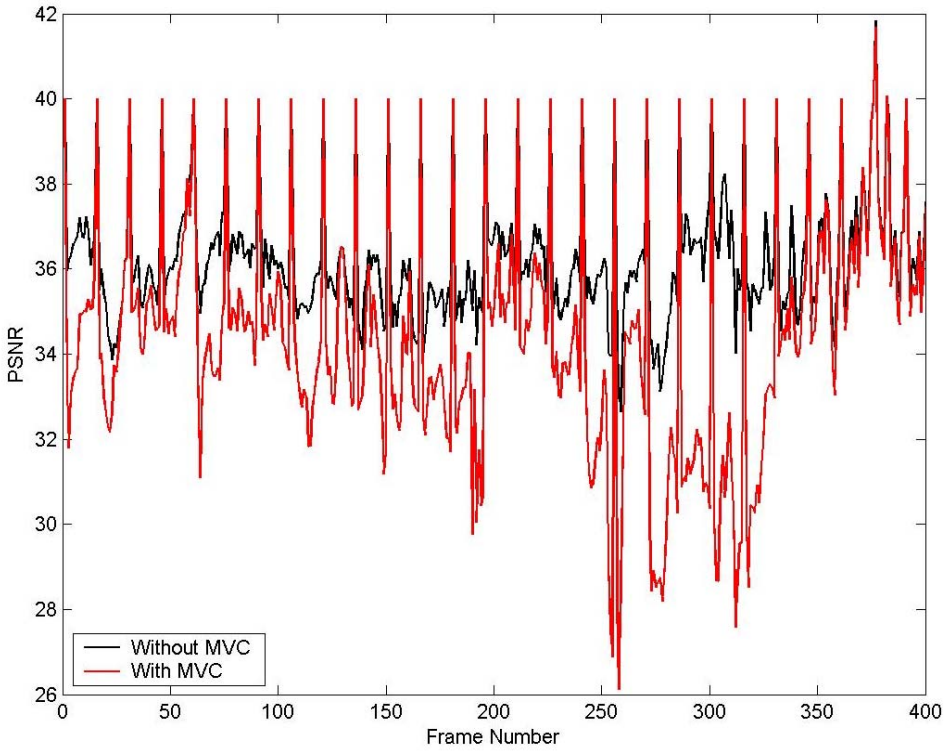


Fig. 4.16 The PSNR of motion compensation,(Foreman sequence QP28)

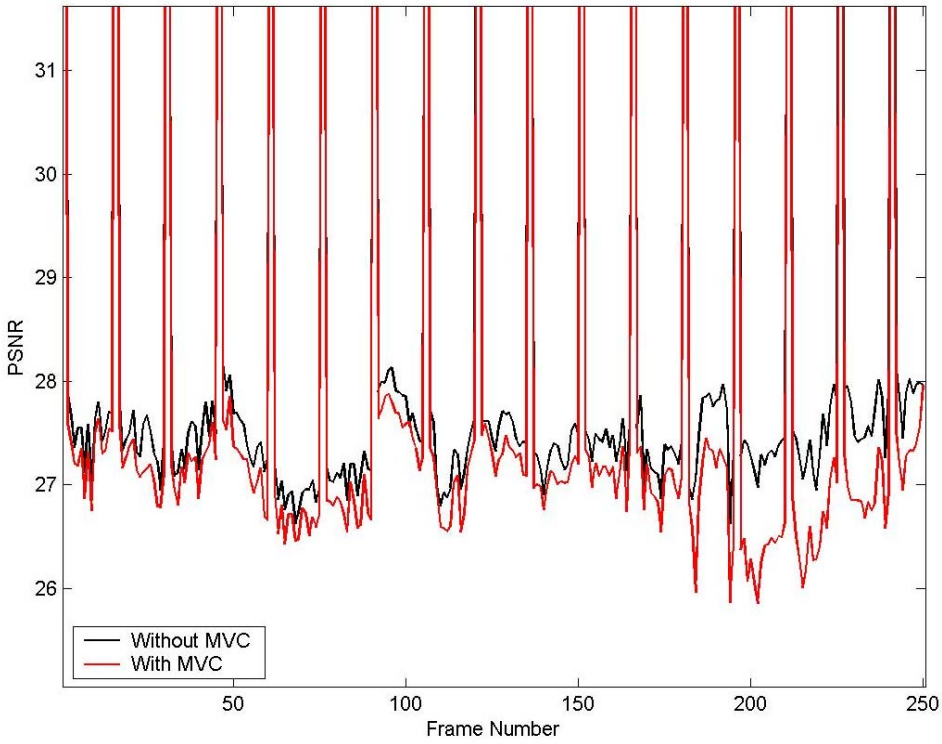


Fig. 4.17 The PSNR of motion compensation, (Flower sequence QP28)

Table 4-5 shows the PSNR after motion compensation and the PSNR after decoding with the concealment based on BMP. As described in Section 3.4.3 the controlling factor α decides the restriction of the MV correction. The zero α corresponds to "no correction", and infinite α corresponds to "full correction". Furthermore, "partial" corresponds to the proposed partial MV correction. In each GOP, the α decays 0.1 per frame from the initial value 2.

We plot the corresponding value of MC_PSNR and EC_PSNR with different α in Fig. 4.18 and Fig. 4.19. It can be seen that even though full correction may reduce the coding efficiency, it does not necessarily provide better performance than partial correction.

Table 4-5 Performance comparison between different levels of MV correction, (Flower sequence QP28)

MC_PSNR: Motion Compensation PSNR；

EC_PSNR: PSNR of the concealment based on BMP

		PLR= 0.036	PLR= 0.07
α	MC_PSNR	EC_PSNR	EC_PSNR
0	35.890	34.300	33.368
0.5	35.951	34.531	33.730
1	36.034	34.540	33.736
1.5	35.769	34.546	33.763
2	35.408	34.568	33.797
2.5	35.102	34.565	33.796
3	34.851	34.578	33.815
3.5	34.647	34.581	33.823
4	34.487	34.584	33.818
4.5	34.362	34.577	33.808
5	34.264	34.573	33.806
inf	33.759	34.546	33.771
partial	35.629	34.544	33.755

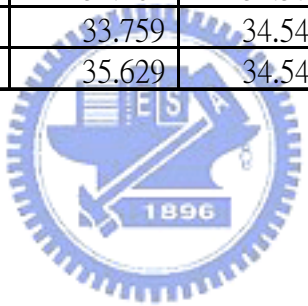


Fig. 4.18 The relation between MC_PSNR and EC_PSNR corresponding to different levels of MV correction, (Flower sequence PLR 2.84%)

Fig. 4.19 The relation between MC_PSNR and EC_PSNR corresponding to different levels of MV correction, (Flower sequence PLR 5.47%)

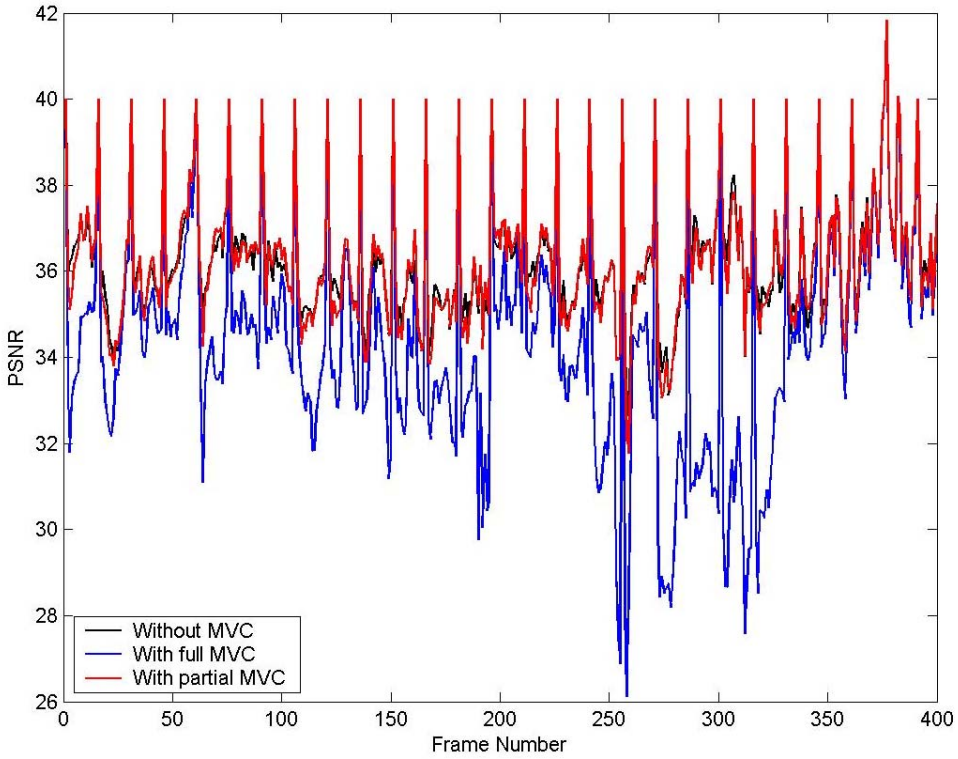


Fig. 4.20 Motion compensation PSNR, (Foreman sequence QP28)

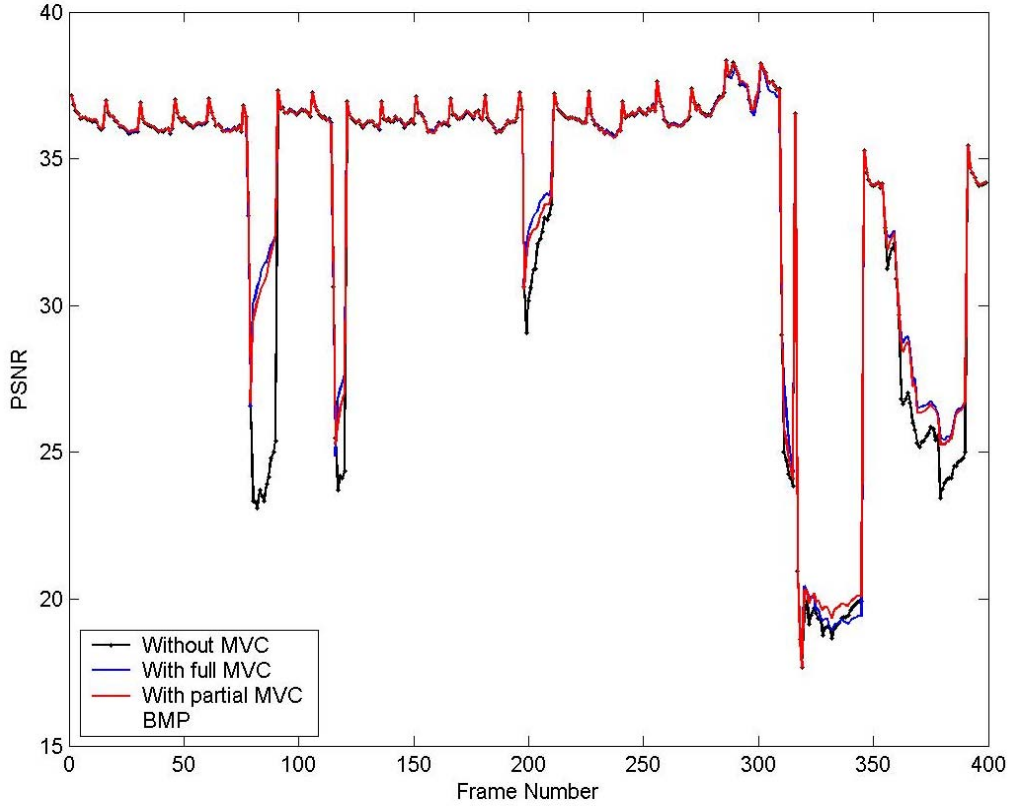


Fig. 4.21 Error concealment PSNR of the Foreman sequence, (Packet loss rate 5.47%)

Fig. 4.20 shows the PSNR of motion compensation with MV correction, without MV correction, and with partial MV correction. We can find that the PSNR with partial MV correction can avoid serious PSNR drops. On the other hand, the performance of error concealment with partial MV correction is close to that with full MV correction. Hence, with partial MV correction, we may enhance the performance of concealment but without sacrificing too much coding efficiency.

Frame No. 88

Fig. 4.22 Reconstructed frames based on FMP (top left), based on BMP (top right), based on BMP with full MV correction (bottom left), and based on BMP with partial MV correction (bottom right), (Foreman sequence, PLR 5.47%)

4.4 GOP with B frames

The previous experiments are works under the Baseline Profile of H.264, and the GOP structure follows IPPP…. Now, we take B frames into consideration under the Main Profile and set the GOP structure as IBBP. Here, we evaluate the performance of our proposed algorithm.

Experimental setting：
Foreman QCIF, QP 28,
Main Profile, fps 15 Hz, Intra period 5 Total 400 frames GOP: IBBPBB…
Error Model: Random Error

Table 4-6 The lost frame-numbers of foreman sequence

PLR 2%：{11, 26, 31, 60, 62, 155, 166, 178, 215, 352, 373}
PLR 5%：{11, 26, 31, 60, 62, 65, 69, 91, 126, 151, 155, 166, 178, 209, 215, 224, 230, 238, 263, 275, 318, 328, 352, 373, 374, 375, 394, 398}

According to Table 4-6, we analysis the lost frame-types at PLR 2% below:

I：31、166

P：178、352、373

B：11、26、60、62、155、215

As an I frame or P frame is lost, it is hard to conceal the lost frame based on the FMP approach. This is due to the long temporal period. In Fig. 4.23, we can see serious error propagation. If we apply our proposed method, the error propagation problem can be relieved. Fig. 4.24 shows the reconstructed frames, and we can find that the proposed method demonstrates much better performance.

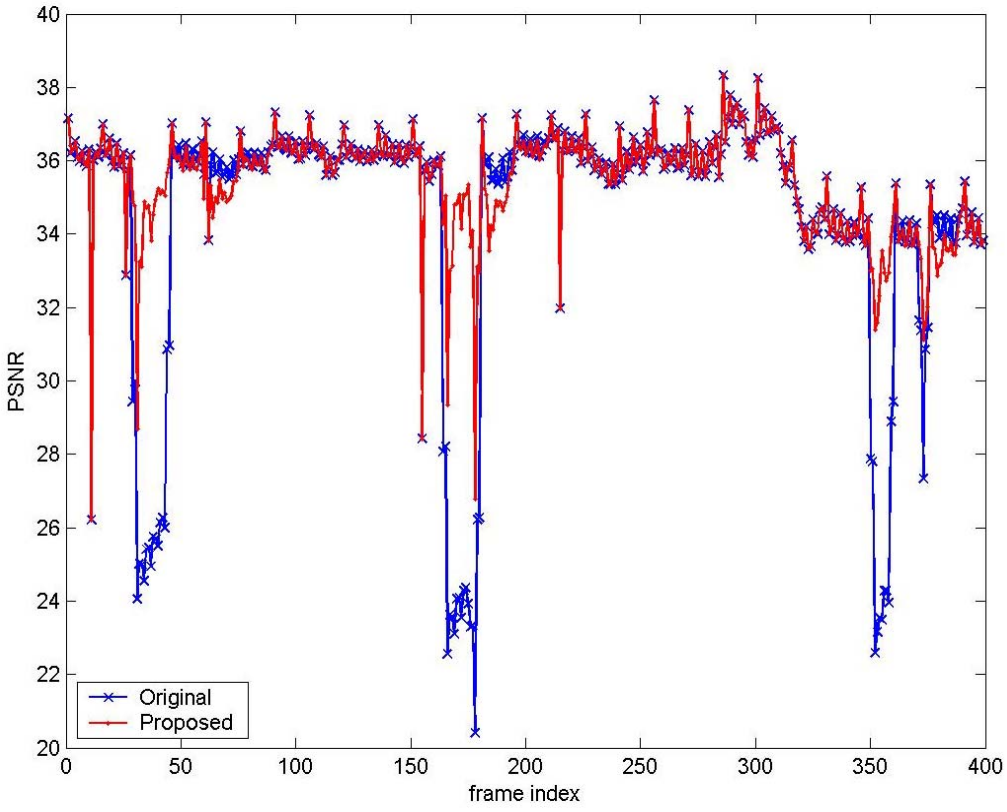
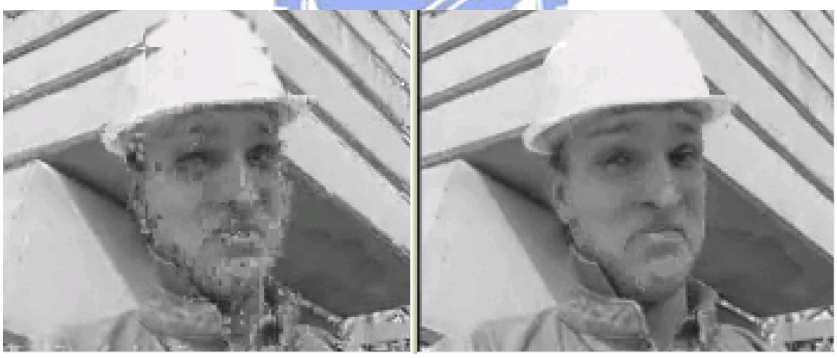


Fig. 4.23 Performance comparison on the foreman sequence, (Packet loss rate 2%)

Frame No. 169

Frame No. 170

Frame No. 172



Frame No. 179

Fig. 4.24 The reconstructed frames. (left) FMP based; (right) proposed.

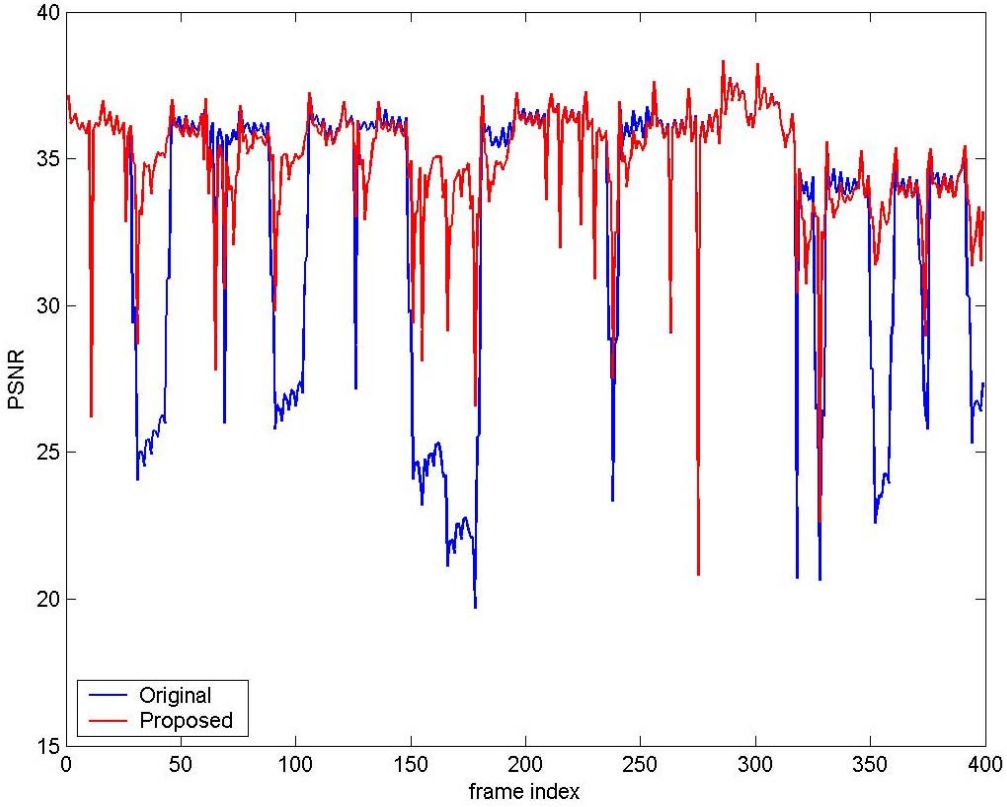


Fig. 4.25 Performance comparison on the foreman sequence, (Packet loss rate 5%)

Table 4-7 The corresponding MC_PSNR and EC_PSNR

Foreman	MC_PSNR	EC_PSNR	
		PLR 2%	PLR 5%
Original	35.70	34.54	34.88
Propoesd	35.56	35.41	33.17

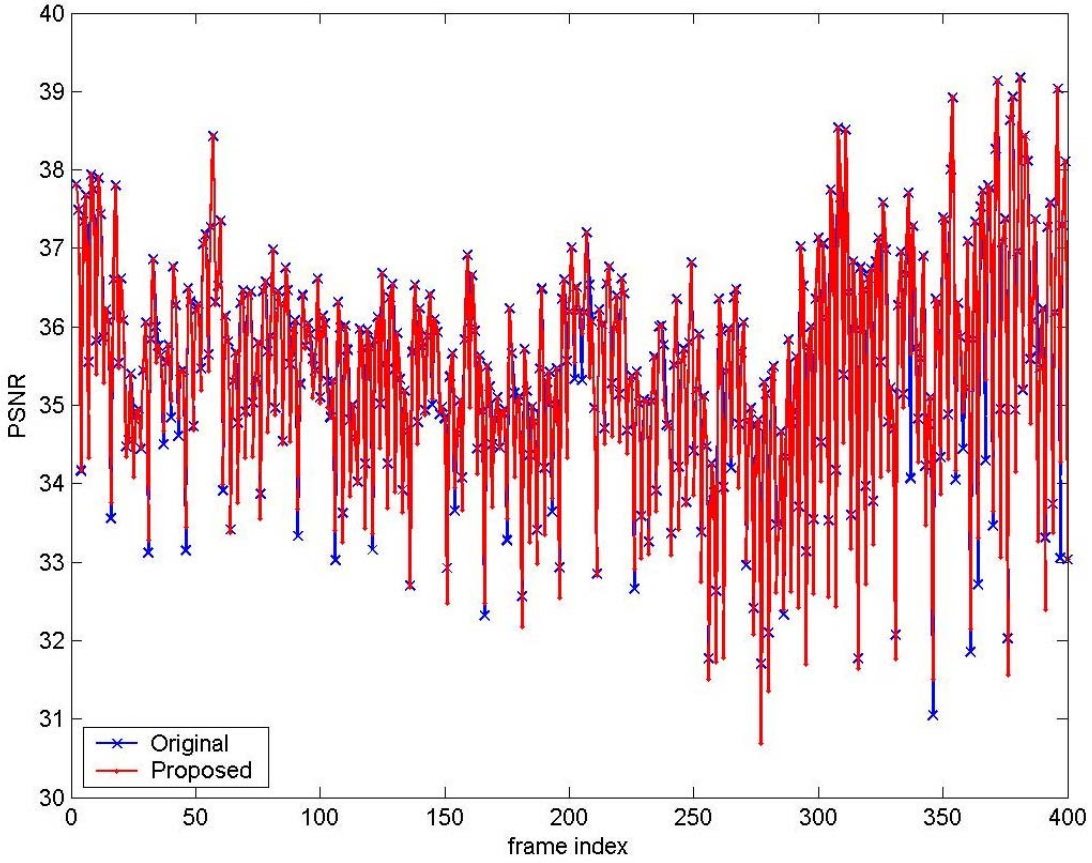


Fig. 4.26 The motion compensation PSNR of the proposed GOP and the traditional GOP, (Forman sequence QP28)

Chapter 5 Conclusion

In this thesis, we propose a novel error concealment technique to deal with the case of entire-frame loss. We also propose an error resilience technique based on MV correction. We conclude our accomplishment as below:

1. We propose a novel MV correction algorithm based on the fuzzy reliability that depends on the spatial and temporal consistency. This algorithm makes the corrected MV more close to the actual motion trajectory and may increase the performance of error concealment based on motion projection.
2. We propose a novel error concealment algorithm based on backward MV projection. This algorithm avoids using the corrupted picture and can decode the subsequent frames with the reference of the last correctly received picture. If compared with existing concealment based on forward MV projection, the proposed technique is much simpler for implementation.
3. In order to find a trade-off between the coding efficiency and the performance of concealment, we propose a partial MV correction algorithm. With the adjustment of the controlling factor, we can not only avoid the increment of motion compensation residual due to MV correction, but also preserve the improvement of error concealment.
4. Finally we also consider the case when B frames are present. We propose a new GOP structure to disconnect successive P frames in order to release the error propagation problem when some P frames are lost. With the corresponding concealment, we indeed improve the quality of the received video sequences.

Bibliography

[1] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. on Circuits Syst. Video Technol.*, Vol. 13, No. 7, pp.560 – 576, July 2003

[2] Iain E G Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley, 2003

[3] Thomas Stockhammer, Miska M. Hannuksela, and Thomas Wiegand, "H.264/AVC in Wireless Environment," *IEEE Trans. on Circuits Syst. Video Technol.*, Vol. 13, No. 7, pp. 657-671, July 2003

[4] Stephan Wenger, "H.264/AVC Over IP," *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 13, No. 7, pp. 645-656, July 2003

[5] Thomas Stockhammer, Thomas Wiegand, Tobias Oelbaum, and Florian Obermeier, "Video coding and transport layer techniques for H.264/AVC-based transmission over packet-lossy networks," *Proceedings of IEEE International Conference on Image Processing (ICIP 2003)*, Vol. 3, pp. 481-484, September 2003.

[6] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid – A Framework for Video Transmission and Quality Evaluation", *Proceedins of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Urbana*, Illinois, USA, pp. 255-272, September 2003.

[7] Chih-Heng Ke, Cheng-Han Lin, Ce-Kuen Shieh, Wen-Shyang Hwang, "A Novel Realistic Simulation Tool for Video Transmission over Wireless Network", *Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC2006)*, pp. 275-283, June 5-7, 2006

[8] Network Simulator, http://www.isi.edu/nsnam/ns/

[9] Zhang Rongfu, Zhou Yuanhua, and Huang Xiaodong, "Content-adaptive spatial error concealment for video communication," *IEEE Trans. on Consumer Electronics,* vol. 50, pp. 335-341, Feb. 2004.

[10] Olivia Nemethova, Ameen Al-Moghrabi and Markus Rupp, "Flexible Error Concealment for H.264 Based on Directional Interpolation," *Proceedins of International Conference on Wireless Networks, Communications and Mobile Computin,* vol. 2, pp. 1255-1260, June 2005.

[11] Steven Beesley, Andrew Armstrong, Christos Grecos, "An Edge Preserving Spatial Error Concealment Technique for the H.264 Video Coding Standard,"

Research in Microelectronics and Electronics 2006, Ph. D., pp 113-116, June 2006.

[12] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," *Proceedins of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 93)*, pp. 417-420, Apr. 1993.

[13] E. T. Kim, S.-J. Choi, and H.-M. Kim, "Weighted boundary matching algorithm for error concealment in the MPEG-2 video bit stream," *Signal Processing.*, vol. 73, pp. 291–295, Mar. 1999.

[14] Yan Chen, Au, O., Chiwang Ho, and Jiantao Zhou, "Spatio-temporal boundary matching algorithm for temporal error concealment," *Proceeedings of IEEE International Symposium on Circuits and Systems*, pp. 686-689, May 2006.

[15] Agrafiotis, D., Bull, D.R., Canagarajah, C.N., "Enhanced Error Concealment With Mode Selection," *IEEE Trans. on Circuits and Systems for Video Technology,* vol. 16, pp. 960-973, August 2006.

[16] M. E. Al-Mualla, C. N. Canagarajah, and D. R. Bull, "Error concealment using motion field interpolation," *Proceedings of IEEE International Conference on Image Processing,* Vol. 2, pp. 512—516, October 1998.

[17] Jinghong Zheng, Lap-Pui Chau, "A temporal error concealment algorithm for H.264 using Lagrange interpolation," *Proceeedings of IEEE International Symposium on Circuits and Systems,* vol. 2, May 2004.

[18] Jae-Young Pyun, Jun-Suk Lee, Jin-Woo Jeong, Jae-Hwan Jeong, and Sung-Jea Ko, "Robust error concealment for visual communications in burst-packet-loss networks," *IEEE Trans. on Consumer Electronics,* vol. 49, pp. 1013-1019, November 2003

[19] S. Gnavi, M. Grangetto, E. Magli, and G. Olmo, "Comparison of rate allocation strategies for H.264 video transmission over wireless lossy correlated networks," *Proceedings of IEEE International Confereence on Multimedia and Expo*, 2003, pp. 517-520, July 2003.

[20] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253-1265, Sept. 1960.

[21] E. 0. Elliott, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, pp. 1977-1997, Sept. 1963.

[22] S. Belfiore, M. Grangetto, E. Magli, G. Olmo, "Concealment of whole-frame loss for wireless low bit-rate video based on multiframe optical flow estimation," *IEEE Transactions on Multimedia*, vol. 7, no. 2, pp. 316-329, Apr. 2005.

[23] S. Belfiore, M. Grangetto, E. Magli, G. Olmo, "An error concealment algorithm for streaming video," *Proceedings of IEEE International Conference on Image Processing*, pp. 649-652, 2003.

[24] Baccichet, P.;and Chimienti, A., **"**A low complexity concealment algorithm for the whole-frame loss in H.264/AVC,**"** *Proceedings of IEEE 6th Workshop on Multimedia Signal Processing*, pp. 279-282, October 2004.

[25] P. Baccicht, D. Bagni, A. Chimienti, L.Pezzoni, and F. Rovati, "Frame concealment for H.264/AVC decoders," *IEEE Transactions on Consumer Electronics,* vol. 51, no. 1, pp. 227-233, Feb. 2005.

[26] Zhenyu Wu and J. M. Boyce, "An error concealment scheme for entire frame losses based on H.264/AVC," *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2006),* pp. 4463-4466, May 2006.

[27] A. M. Tekalp, "Digital Video Processing." Englewood Cliffs, NJ: Prentice- Hall, 1995.