doi:10.3772/j.issn.1006-6748.2019.03.014

# Error concealment in pixel based Wyner-Ziv video coding[①]

Yuan Chengzong(袁承宗)[②*], Han Chuanzhao[**], Wang Yan[***], Zhang Ning[***],
Zhang Zhen[***], Zhu Xinzhong[**], Li Xian[**], Qu Di[*]

([*] Key Laboratory of Ministry of Education and Ministry of Health Shanghai Medical College,
Fudan University, Shanghai 200032, P. R. China)
([**] China Academy of Space Technology, Beijing 100094, P. R. China)
([***] Shanghai Aerospace Elecetronic Technology Institute, Shanghai 201109, P. R. China)

**Abstract**

In the Wyner-Ziv (WZ) video coding paradigm, a virtual correlation channel is assumed between the quantized source and the side information (SI) at the decoder, and channel coding is applied to achieve compression. In this paper, errors caused by the virtual correlation channel are addressed and an error concealment approach is proposed for pixel-based WZ video coding. In the approach, errors after decoding are classified into two types. Type 1 errors are caused by residual bit errors after channel decoding, while type 2 errors are due to low quality of SI in part of a frame which causes SI not lying within the quantization bin of a decoded quantized pixel value. Two separate strategies are respectively designed to detect and conceal the two types of errors. Simulations are carried out and results are presented to demonstrate the effectiveness of the proposed approach.

**Key words**: Wyner-Ziv (WZ) video coding, error detection, error concealment, side information (SI)

## 0 Introduction

Based on the Slepian-Wolf (SW)[1] and the Wyner-Ziv (WZ)[2] theorems, WZ video coding, or lossy coding of video with decoder side information (SI), provides potential of flexibly shifting computational complexity between encoder and decoder while still maintaining a comparable performance as conventional video coding techniques, such as H. 26x, and MPEG-x, which makes WZ video coding very promising for applications in which low complexity encoder is a must, for example wireless video surveillance, mobile video phones, and so on. As a result, WZ video coding has attracted more and more research efforts in recent years. For the background and current development of WZ video coding, the readers are referred to Refs[3-5]. The readers can also be referred to Refs[13-15] for the current research efforts in this field.

Different from conventional video coding techniques, compression of quantized source in WZ video coding is generally realized through channel coding. SI generated at the decoder can be considered as the output of a virtual correlation channel where the quantized source is the input, and WZ coding of video becomes a channel coding problem. So channel capacity approaching error correcting codes, such as turbo codes or low density parity check (LDPC) codes, can be applied to correct the ‘errors’ in SI to restore the source by transmitting parity bits to the decoder.

According to the channel codes used, WZ video coding architectures can be divided into several types[6-8], among which turbo code based WZ video coding is very popular thanks to the good performance of turbo codes for short length sequences. In the turbo code based architecture, two variants exist, namely pixel based and transform based schemes. In this paper, only a turbo code based codec is considered which works in pixel domain. In this scenario, turbo code works on bit planes of the quantized pixel values of a video frame, and reconstruction of the frame at the decoder is carried out taking into consideration both SI and the decoded quantized pixel values.

In pixel based WZ video coding, even after bit planes of the quantized pixel values are successfully decoded, errors will still remain in the decoder reconstruction generally. Here, errors only refer to those caused by the virtual correlation channel, not the real

[①] Supported by the National Science and Technology Major Project of China (No. 2018ZX10734401-004).

[②] To whom correspondence should be addressed. E-mail: arm _ yuan@163. com

Received on July 14, 2018

transmission errors. The errors can be classified into two types. The error of type 1 comes from the residual bit errors in the decoded bit planes after turbo decoding. Due to the temporal and spatial variational characteristics of the virtual correlation channel, in WZ video coding, the residual bit error rate (BER) threshold can not be set to 0. As a result, after turbo decoding, there may still be many errors left in the decoded bits. Take a QCIF-sized frame as an example, there are 25 344 bits in a bit plane, with a residual BER threshold of order $10^{-3}$ (which is usually the case), there may be as many as 25 errors still remaining in the bit plane after turbo decoding. The more bits a pixel value is quantized in, the more residual bit errors may result. If these errors are not corrected, they will exhibit themselves as isolated blinking pixels in the reconstructed frame and are very annoying for human eyes. The error of type 2 is due to SI not lying within the quantization bin of a quantized pixel value. The main cause is the poor quality of SI. Due to occlusion or complex motion, some parts of SI may have very poor quality, so even though the quantized pixel values are correctly decoded, in these parts, SI may still lie out of the quantization bin of the quantized pixel values. Isolated type 2 errors can cause objective quality degradation of the decoder reconstruction, while clusters of pixels with type 2 errors will lead to degradation of both objective and subjective qualitied of the decoder reconstruction. Note that type 1 errors can also make SI lie out of the quantization bin of quantized pixel values even though SI is good. Till now, there is still little effort to handle these errors. In Ref. [7], to remedy reconstruction errors due to SI not lying within the quantization bin, subtract dithering was performed by shifting the quantizer partitions for every pixel using a pseudo-random pattern. To our knowledge, this is the only paper which takes into consideration errors in decoder reconstruction, but they did not handle residual bit errors after channel decoding and strictly speaking, their approach for type 2 errors is a very basic one.

To combat these errors, in this paper, an error concealment approach is proposed for pixel-based WZ video coding. In the approach, part of the type 1 errors are detected by examining the confidence measures of the decoded bits, while the remaining type 1 errors and type 2 errors are detected by examining whether SI lies within the quantization bin of a quantized pixel value. Two separate strategies are then designed to deal with these two types of errors. Simulation results on several test video sequences verify the effectiveness of the proposed approach.

The rest of this paper is organized as follows. In Section 1, a structure of the WZ video codec considered in this paper is presented. In Section 2, the detection of the two types of errors is proposed in detail. In Section 3, an approach to conceal the two types of errors is given. Simulation results are presented in Section 4. And conclusion is drawn in Section 5. Now this technology has been applied to the video surveillance system in BSL-3 laboratory.

## 1 WZ video codec structure

The codec structure considered in this paper is depicted in Fig. 1. The input video sequence is divided into key frames and WZ frames. The key frames are intra encoded and decoded by an H. 264 intra codec. Each WZ frame is encoded independent of the other frames. For a particular WZ frame, pixels are quantized with a uniform quantizer, and then each bit plane of the quantized pixel values in the frame is fed to the turbo encoder. The systematic bits are discarded and only the parity bits of the turbo coder output are stored in a buffer. The parity bits are sent to the decoder in chunks upon decoder requests.

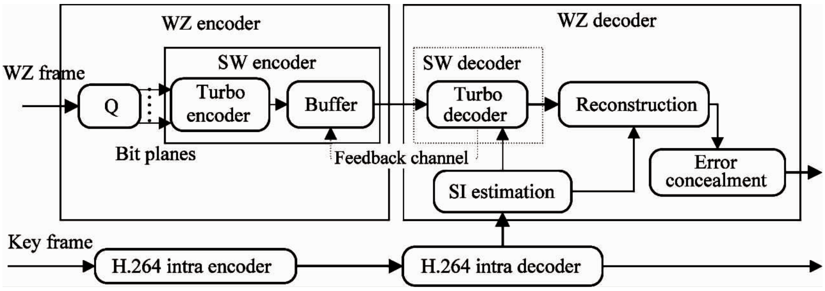


**Fig. 1** The structure of the WZ video codec.

At the decoder, SI is first estimated through motion compensated temporal interpolation (MCTI) using the key frames. A Laplacian distribution is adopted for the virtual correlation channel. For each bit plane, upon reception of a chunk of parity bits, the turbo decoder performs maximum posterior (MAP) turbo decoding using the SI and the received parity bits. The residual BER is then estimated. If the residual BER is not bellow a threshold, the decoder requests for more chunks of parity bits through the feedback channel. After turbo decoding, reconstruction of the frame is carried out using the decoded quantized pixel values and SI. Before output, the decoder reconstruction further undergoes an error detection and concealment process as detailed in the sub-sections that follows.

## 2 Error detection

Let $X_i$ be the $i$th pixel of the current WZ frame $X$, and $X_i$ is quantized to $QX_i$ by a uniform quantizer of $M$ bits, the bits of $QX_i$ are numbered by $b$, $b = 0, \cdots, M-1$ with $b = 0$ denoting the most significant bit. It is further assumed that SI for $X_i$ is $Y_i$. At the WZ video decoder, the value of the $b$th bit of $QX_i$, denoted by $u_i^{(b)}$, is determined by computing the logarithm of the posteriori probability ratio (LAPPR) through iterative turbo decoding as specified in Eq. (1):

$$LAPPR_i = \log\left(\frac{\Pr(QX_i^{(b)} = 1 \mid Y_i,\ QX_i^{(b-1)},\ QX_i^{(b-2)}, \cdots, QX_i^{(0)})}{\Pr(QX_i^{(b)} = 0 \mid Y_i,\ QX_i^{(b-1)},\ QX_i^{(b-2)}, \cdots, QX_i^{(0)})}\right) \quad (1)$$

where $QX_i^{(b-1)}, \cdots, QX_i^{(0)}$ denote the already decoded bits of $QX_i$. From Eq. (1), it is known that decoding of $QX_i^{(b)}$ is conditioned on both SI and the already decoded more significant bits of $QX_i$. $u_i^{(b)}$ is then determined by

$$u_i^{(b)} = \begin{cases} 1 & \text{if } LAPPR_i > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

In the turbo decoding process, statistics on LAPPR is a good measure to determine whether a bit plane is successfully decoded[9-11], and the value of $LAPPR_i$ can be used as a confidence measure for the $i$th bit of the bit plane. More specifically, if $|LAPPR_i|$ is large enough, the bit is more likely correctly decoded, while a small $|LAPPR_i|$ means that the decoded bit is unreliable. But a small $|LAPPR_i|$ does not always correspond to an error decoded bit, while a large $|LAPPR_i|$ does not always mean a correctly decoded bit either. But by checking bits with a small $|LAPPR_i|$, more chances to are got find bits with residual bit errors. So part of the type 1 errors are detected by finding pixels containing at least one bit which has a small LAPPR value. The remaining type 1 errors will be detected with other method which is detailed later. To do this, after a bit plane is successfully decoded, the turbo decoder is asked to return the LAPPR for each bit in the bit plane, and record those bits whose absolute LAPPR values are below a predefined threshold $T_1$ in map $C$. $C$ has the same number of elements as the WZ frame. All the elements of $C$ are initially set to 0. For the $i$th bit of the bit plane, if $|LAPPR_i| < T_1$, it is recorded in the map by $C_i = C_i + 1$. When all the bit planes are turbo decoded, and the decoder reconstruction frame has been constructed, the following process is carried out to detect this part of type 1 errors.

```
1)  For each pixel $\hat{X}_i$ in the decoded frame $\hat{X}$, do
2)      if $C_i == 0$, then
3)  continue
4)      else
5)  Find the $3 \times 3$ neighbourhood for pixel $\hat{X}_i$
6)  Sort the pixel values in the neighbourhood, and denote the minimum and maximum by $\min_{3\times3}$ and $\max_{3\times3}$.
7)  if the value of $\hat{X}_i$, denoted by $\hat{x}_i$, satisfy $\hat{x}_i \neq \min_{3\times3}$, and $\hat{x}_i \neq \max_{3\times3}$, then
8)  set $C_i$ to $C_i = 0$.
9)  end if
10)     end if
11)  end for
```

After the above process, a pixel $\hat{X}_i$ with $C_i \neq 0$ is considered as having a type 1 error. The basic idea of the above procedure is to find in the $3 \times 3$ neighbourhood of $\hat{X}_i$, and whether $\hat{X}_i$ is a local peak or a valley. As a type 1 error exhibits as an extreme pixel value, so by checking whether $\hat{X}_i$ is peak or valley at its neighbourhood, a type 1 error can be detected.

Type 2 error can generally be detected more easily. Check whether SI $Y_i$ for pixel $X_i$ lies within the quantization bin of $QX_i$, which is carried out by quantizing $Y_i$ with the same uniform quantizer to $QY_i$ and checking whether they have the same value. Like type 1 errors, type 2 errors are recorded in another map $D$, this map is built as follows:

```
1)  Initialize all elements of $D$ to 0.
2)  For each turbo decoded quantized pixel $QX_i$, do
3)  if $QX_i$ and $QY_i$ are different, then
4)  set $D_i$ to $D_i = 1$
5)  end if
6)  end for
```

Typically, the above type 2 error detection process will locate a lot of pixels. Some of them are isolated ones, while others are clusters. In this paper, only those clustered type 2 errors with larger size are. The handling of the remaining type 2 errors will be considered in the future work. Morphological opening of the map $D$ with a $2 \times 2$ square structuring element is carried out to remove those isolated pixels and regions of small size.

As mentioned above, error of type 1 can also cause $Y_i$ not lying within the quantization bin of $QX_i$. But type 1 errors are isolated ones and for them the value of $QY_i$ and $QX_i$ are very different. So by computing the absolute difference of the value of $QY_i$ and $QX_i$ and then comparing against threshold $T_2$, pixels with possible type 1 errors can be located. By further checking whether a located pixel is a local peak/valley in the reconstructed frame, the remaining type 1 errors which are not detected using confidence measure can be located. Of course, pixels with type 1 errors should be removed from map $D$. This is simply carried out by another scanning of map $D$, and reset those elements to 0.

## 3 Error concealment

After type 1 errors are detected, median filtering is used to conceal them. For each pixel $\hat{X}_i$ with $C_i \neq 0$, its 3 × 3 neighbourhood is found as in the error detection process. then the median of the pixel values in the 3 × 3 neighbourhood is found and the median is used to replace the value of $\hat{X}_i$.

Unlike type 1 errors, type 2 errors are more difficult to conceal. There are several approaches that can be considered here, namely spatial concealment, temporal concealment, or combined concealment. It is regarded type 2 errors are caused by non-accurate motion estimation during the SI estimation process due to the lack of information of the original WZ frame at the decoder. So with the turbo code decoded quantized pixel values, it is now possible that more accurate motion to be estimated, and type 2 errors concealed. Based on this assumption, a temporal concealment approach is proposed, in which modified boundary block matching (MBBM) is utilized to estimate the motion for blocks containing type 2 errors.

The approach is composed of several major steps: boundary block formation for MBBM, matching criterion determination, motion estimation and error concealment.

Before boundary blocks can be formed, regions with type 2 errors in the current decoded frame should be located first. To make things easier, as in the SI estimation process, map $D$ and the current decoded frame are divided into 8 × 8 non-overlapping regular blocks. The blocks in the current decoded frame corresponding to those in map $D$ with non-zero entries are the ones with type 2 errors and located for further processing. As can be seen, using block of size 8 × 8 will make the search of the motion of a block easier.

After all 8 × 8 blocks with type 2 errors in the current decoded frame are found, boundary block formation for MBBM motion estimation is conducted for each block one by one. Assume $B$ is such a block, it is further divided into 4 × 4 sub-blocks. According to the pixels and sub-blocks containing type 2 errors, several boundary block formation strategies are separately used. In the first case, when more than 48 pixels are in the block containing type 2 errors, a rectangular block will be formed which contains $B$ and has an extra boundary of 2 pixels on each side. If some of the borders have less than 2 extra pixels, they will be ignored. If all borders have less than 2 pixels, concealment of $B$ will be delayed. In the second case, when three or four sub-blocks contain type 2 errors, $B$ itself will be taken as the boundary block. In the third case, if 2 sub-blocks contain type 2 errors, and the two sub-blocks with type 2 errors are located in the two diagonal directions, $B$ will be taken as boundary block again. In the fourth case, when 2 sub-blocks contain type 2 errors and are located along one border of $B$, the minimum bounding rectangle of the errors is first found, and boundary block is then formed by extending three borders of the bounding rectangle to the three corresponding borders of $B$, and extending the fourth border of the bounding rectangle 2 pixels to the inner part of $B$. Fig. 2 illustrates the case in which the two sub-blocks with type 2 errors are located along the top border of $B$. In the figure, the errors and added borders are respectively represented by dark and light grey shields. The dashed lines in the figure indicate the borders of the four sub-blocks. In this case, the left, top and right borders of the minimum bounding rectangle are extended to the corresponding borders of $B$, while the bottom border of the minimum bounding rectangle is extended 2 pixels downward to form the fourth border of the boundary block. Finally, in the fifth case, when only one sub-block contains type 2 errors, the minimum bounding rectangle of the errors are found again, and boundary block is formed by extending two

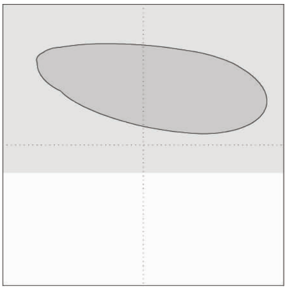


**Fig. 2** Illustration of boundary block formation when two 4 × 4 sub-blocks contain type 2 errors

borders of the minimum bounding rectangle to the corresponding borders of $B$, and extending the other two borders 2 pixels respectively to the inner part of $B$. The case in which only the top left sub-block containing type 2 errors is illustrated in Fig. 3.

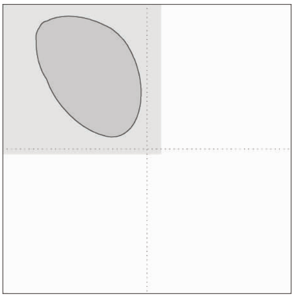


**Fig. 3** Illustration of boundary block formation when only a 4 × 4 sub-block contains type 2 errors

After boundary block is formed, A block will contain both reconstructed pixels and pixels with type 2 errors correctly. For the pixels with type 2 errors, in MBBM, their decoder reconstruction should be replaced by the decoded quantized pixel values in the search of motion, which is different from conventional BBM, where only boundary pixels are used. As a result, a different matching criterion is needed. Assume $B'$ is such a boundary block, $S_{B1}$ and $S_{B2}$ are the two sets for $B'$, which contain boundary pixels and pixels with type 2 errors, respectively. In addition, assume $k$ is a pixel in $B'$, and if $k \in S_{B1}$, its reconstruction from both SI and quantized pixel value will be $\hat{x}(k)$, if $k \in S_{B2}$, its quantized pixel value will be $\hat{q}(k)$. At a certain search location, the following matching criterion will be used:

$$SAD' = \sum_{k \in S_{B1}} | \hat{x}(k) - \hat{x}'(k) | + \lambda \sum_{k \in S_{B2}} | \hat{q}(k) - \hat{q}'(k) | \quad (3)$$

where $\hat{x}'(k)$ is the decoded pixel value for the corresponding pixel $k$ in the reference frame, $\hat{q}'(k)$ is the quantized pixel value for $\hat{x}'(k)$ using the same *M-bit* uniform quantizer, and $\lambda$ is a parameter to adjust the relative importance of the two types of pixels in $B'$. Obviously, $\lambda$ can be written as

$$\lambda = c_M \lambda' \quad (4)$$

where $c_M$ is a constant varying with the number of bits of the quantized pixel values $M$, when $M = 1$, $c_M = 255$, and $M = 2$, $c_M = 85$, and so on. $\lambda'$ is another constant.

Two strategies can be taken to estimate the motion for a boundary block, one is to search its motion and ignore that motion has been estimated for all blocks in the WZ frame when SI is estimated. The other is to use the motion already estimated as a starting point and refine it in a small search region. And the latter strategy is taken to reduce the computational load. As boundary block $B'$ is mostly formed with pixels from block $B$, so the motion vectors for block $B$ determined in the SI estimation process is taken as the starting search point for the boundary block $B'$ associated with it.

Two motion vectors will be searched for boundary block $B'$, one with reference to the previous decoded key frame $\hat{X}'$ (forward motion estimation), and the other with reference to the next decoded key frame $\hat{X}''$. So the determination of starting search point will be carried out in $\hat{X}'$ and $\hat{X}''$, respectively. Refinement of the motion in $\hat{X}'$ and $\hat{X}''$ is carried out in a small search region with $SAD'$ as the matching criterion.

From the two estimated motion vectors for a boundary block $B'$, and forward prediction, backward prediction, and the average of the two are chosen and the one which leads to the minimum $SAD'$ is taken as the new SI for the block. Pixels in $B'$ with type 2 errors are concealed by reconstructing these pixels using the new SI and the decoded quantized pixel values again as in the decoder reconstruction process.

Several runs of the above error concealment process may be needed to conceal blocks not handled in the previous runs.

## 4 Simulation results

The proposed error concealment approach was implemented and was integrated into a baseline pixel based WZ video codec[12]. Different from the baseline codec in Ref. [12], in which ideal bit error detection was assumed, in the codec used in this paper, the residual BER is estimated by statistics on *LAPPR* as in Refs[9-11]. The impact of error concealment on the performance of the codec is assessed using the first 101 frames of four QCIF video sequences: Foreman, Hall monitor, Carphone and Stefan. In the simulations, all even number frames of a sequence are H. 264 intra coded, while all odd number ones are WZ coded.

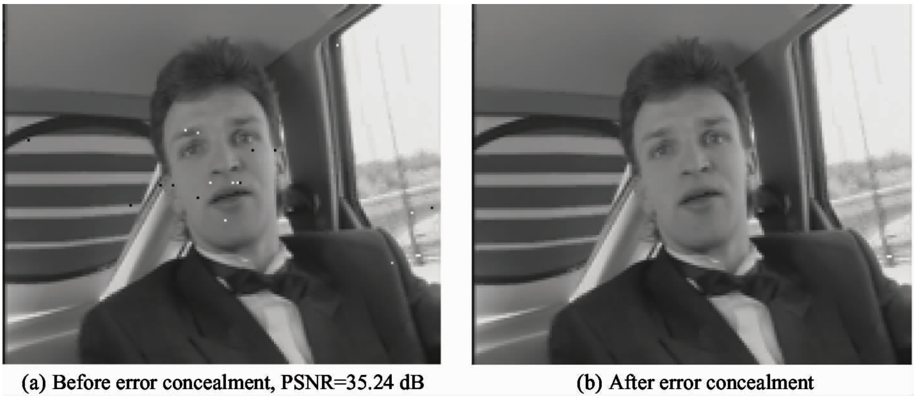
The residual BER threshold for a bit plane was set to 0.0025. The threshold $T_1$ was set to log99. $T_2$ was set to $(1 << M)/3$, as before, $M$ denotes the number of bits a pixel is quantized in, and $<<$ denotes left shift. $\lambda'$ in equation 4 was set to 1.3. As already mentioned in the previous section, in the SI estimation process, 8 × 8 blocks were used for motion estimation, and the search range was set to $[-7, +7]$, in the MBBM, the search range for motion refinement was set to $[-3, +3]$.

Table 1 shows the simulation results for the test video sequences when a 16-level uniform quantizer is used ($M = 4$). In the table, the first row shows the sequences tested. The other rows give the average PSNR for the test video sequences with and without the error concealment respectively. In the table, the PSNR is only collected from the luminance component of the WZ frames. From the table, it can be seen, with error concealment, improvement in PSNR ranging from 0.14 dB to 0.44 dB has been obtained. For example, for Carphone sequence, without error concealment, average PSNR is 39.07 dB, with error concealment, the average PSNR is 39.51 dB, and a gain of about 0.44 dB is obtained.

Table 1 Simulation results

| Video sequence | Foreman | Carphone | Hall | Stefan |
|---|---|---|---|---|
| With error concealment | 39.43 | 39.51 | 42.98 | 36.18 |
| Without error concealment | 39.17 | 39.07 | 42.84 | 35.81 |

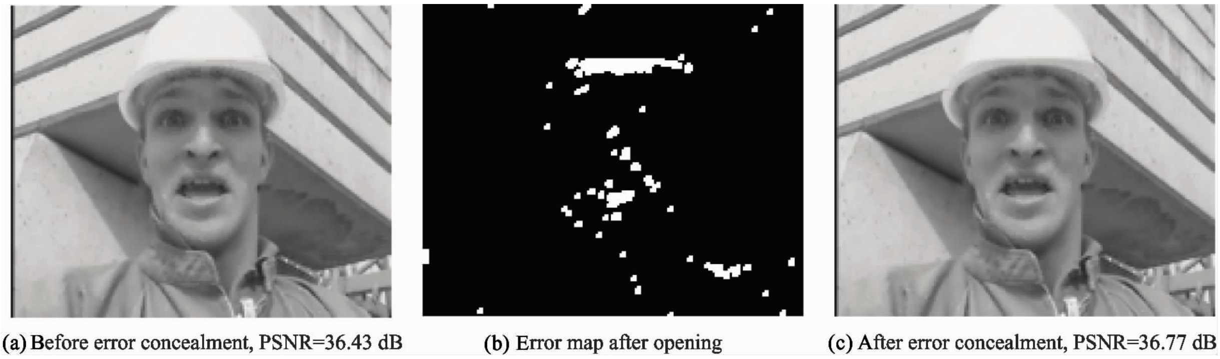
Fig. 4 shows the result for the 21th frame of Carphone sequence. Fig. 4(a) shows the decoded frame without error concealment. Fig. 4(b) shows the same decoded frame after error concealment.



(a) Before error concealment, PSNR=35.24 dB (b) After error concealment

**Fig. 4** The decoded 21th frame of Carphone

Fig. 5 shows the result for the 63th frame of Foreman sequence. Fig. 5(a) shows the decoded frame without error concealment, Fig. 5(b) shows map $D$ after opening, and Fig. 5(c) shows the decoded frame after error concealment.



(a) Before error concealment, PSNR=36.43 dB (b) Error map after opening (c) After error concealment, PSNR=36.77 dB

**Fig. 5** The decoded 63th frame of Foreman

Fig. 6 shows the frame to frame PSNR for Carphone sequence. In the figure, results obtained before error concealment and after error concealment are denoted by ‘without error concealment’, and ‘with error concealment’, respectively. It can be seen from the figure, with error concealment, PSNR for all WZ frames is improved.

From the figures, it can be seen that, with error concealment, the quality of the decoded video is obviously improved.

Regarding to the computational overhead caused by the proposed approached, the most time consuming part is the MBBM in type 2 error concealment. It performs motion estimation in [ −3, +3 ] with boundary blocks. Compared with the computational cost caused by the [ −7, +7] motion search in SI estimation, the overhead of MBBM is very low. As in a frame there is only part of the blocks containing type 2 errors. For example, in a frame with 30% of blocks containing type 2 errors, the computational overhead of MBBM is about 15% of the motion estimation process for SI estimation.

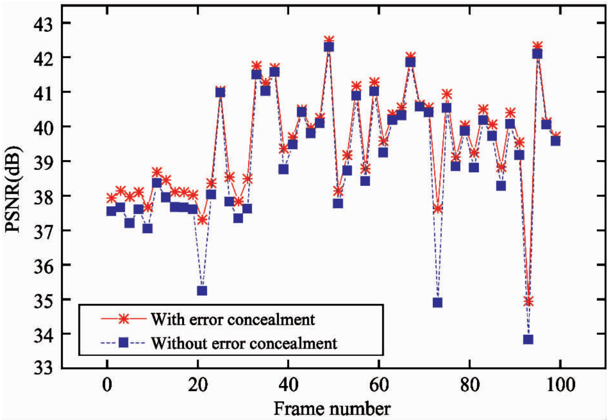


**Fig. 6** Frame to frame PSNR, Carphone sequence

## 5 Conclusion

An error concealment approach for pixel-based WZ video coding is proposed. In the approach, the errors in the decoded frame are classified into two types, namely errors caused by residual bit errors after turbo decoding, and those caused by poor quality of SI. Separate strategies are then designed to detect and conceal the errors. Simulations are carried out to assess the approach, and the results show the improvement in both objective and subjective quality of the decoded video.

More work is under consideration, for example, the concealment of type 2 errors for isolated pixels and regions of small size.

### References

[ 1] Slepian D, Wolf J K. Noiseless coding of correlated information sources[J]. *Transactions on Information Theory*, 1973,19(4): 471-480

[ 2] Wyner A, Ziv J. The rate-distortion function for source coding with side information at thedecoder[J]. *Transactions on Information Theory*, 1976, 22(1): 1-10

[ 3] Girod B, Aaron A, Rane S, et al. Distributed video coding[J]. *Proceedings of IEEE*, 2005, 93(1): 71-83

[ 4] Xiong Z, Liveris A D, Cheng S. Distributed source coding for sensor networks[J]. *IEEE Signal Processing Magazine*, 2004, 21(5): 80-94

[ 5] Puri R, Majumdar A, Ishwar P, et al. Distributed video coding in wireless sensor networks[J]. *IEEE Signal Processing Magazine*, 2006, 23(7): 94-106

[ 6] Pradhan S S, Ramchandran K. Distributed source coding using syndromes (DISCUS): design and construction [J]. *IEEE Transactions on Information Theory*, 2003, 49 (3): 626-643

[ 7] Aaron A, Zhang R, Girod B. Wyner-Ziv coding of motion video[C]. In: Proceedings of the 36th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, USA, 2002. 240-244

[ 8] Xu Q, Xiong Z. Layered Wyner-Ziv video coding[C]. In: Proceedings of the Electronic Imaging 2004, San Jose, USA, 2004. 3791-3803

[ 9] Morbee M, Parades-Nebot J, Pizurica A, et al. Rate allocation algorithm for pixel-domain distributed video coding without feedback channel[C]. In: Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal and Signal Processing ICASSP, Honolulu, USA, 2007. 521-524

[10] Tagliasacchi M, Pedro J, Pereira F, et al, An efficient request stopping method at the turbo decoder in distributed video coding[C]. In: Proceeding of the 2007 15th European Signal Processing Conference, Poznan, Poland, 2007. 1427-1431

[11] Kubasov D, Lajnef K, Guillemot C. A hybrid encoder/decoder rate control for Wyner-Ziv video coding with a feedback channel[C]. In: Proceedings of the 2007 IEEE 9th Workshop on Multimedia Signal Processing, Crete, Greece, 2007. 251-254

[12] Wang Y, Jeong J, Wu C, et al. Wyner-Ziv video coding with spatio-temporal side information[C]. In: Proceedings of the 2007 IEEE International Conference on Multimedia an Expo, Beijing, China, 2007. 132-135

[13] Taheri Y M, Ahmad M O, Swamy M N S. A joint correlation noise estimation and decoding algorithm for distributed video coding[J]. *Multimedia Tools and Applications*, 2018,77(6): 7327-7355

[14] Jia Y, Wang Y, Song R, et al. Decoder side information generation techniques in Wyner-Ziv video coding: a review[J]. *Multimedia Tools and Applications*, 2015, 74 (6):1777-1803

[15] Qing L, He X, Ou X, et al. Distributed video coding based on multi-source correlation mode [J]. *Applied Mathmetics and Information Sciences*, 2013,7(4):1609-1614

**Yuan Chengzong**, born in 1985. Now he is studying for Doctor of Engineering (D. Eng) at Key Laboratory of Ministry of Education and Ministry of Health Shanghai Medical College of Fudan University. He received his M. S. degree in School of Communication and Information Engineering of Shanghai University in 2010. He also received his B. S. from Shanghai University in 2007. His research interests include the design of embedded system, digital image processing, and video codec technology.