Compressed-domain-based Transmission Distortion Modeling for Precoded H.264/AVC Video

Fan Li, Student Member, IEEE, and Guizhong Liu, Member, IEEE

Abstract—Transmission distortion analysis for video streams is a considerably challenging task. In this paper, a compressed-domain-based (CDB) transmission distortion model for precoded H.264/AVC video streams is developed. Unlike the earlier schemes, which were based on pixel domain and required a complete decoding of the compressed video streams, the CDB model only requires some information on the video features, which can be directly extracted from the compressed video streams. Therefore, the complexity of the calculations is substantially reduced, which is well suited for real-time applications. More specifically, the model is applicable to the real-time transmission for precoded video streams, such as video on demand and mobile video. The experimental results demonstrate high accuracy of the model. Furthermore, an application example using the CDB model in resource allocation in real-time multi-user video communication reveals the applicability and effectiveness of the model.

Index Terms—Video; wireless, H.264, distortion modeling, compressed domain

I. INTRODUCTION

DISTORTION analysis has recently gained much attention in the research community. Accurate transmission distortion estimation has important applications on multiuser scheduling and wireless resource allocation at the sender [1]. However, it is still a considerably challenging task for transmission distortion analysis. When the video sender drops packets due to congestion, or when packets are lost in the channel, transmission error occurs and would further propagate to its subsequent frames along the motion prediction path in the video streams. As a result, accurate prediction of the transmission distortion is difficult. Many researchers have carried out significant amount of studies in this area [2]–[7]. The well-known recursive optimal per-pixel estimation (ROPE) method [2], [3] recursively calculates the first- and second-order moments of the decoded value for each pixel, from which one can determine the total distortion of each pixel. In [5], a Group of Picture (GOP) level distortion model is developed. The expected distortion is estimated by explicitly accounting for both the loss pattern probability and the resulting distortion. However, the models in earlier [2]–[7] are pixel-based and the distortion estimation requires a complete decoding of the compressed video streams, which is obviously computationally inefficient. Compressed-domain-based methods develop novel approaches, which are characterized by a low computational complexity. They have been widely utilized in transcoding [8], scene detection [9] and so on. In our opinion, the compressed-domain-based method is also well applicable in the transmission distortion analysis.

In this paper, we propose a compressed domain approach to the transmission distortion modeling for precoded H.264/AVC video. The model explicitly accounts for both the packet losses in the networks and the content of the video streams. An estimation of the characteristics of H.264/AVC video content is made directly from the compressed domain, instead of the pixel domain which has to fully decode the video streams. The approach only needs some information about the video features, which can be directly extracted from the compressed video streams. As a result, the calculation complexity is substantially reduced. More importantly, since it is a predictive model, the sender can predict the distortion caused by the video transmission under the current network condition. Thus, the CDB model has important applications in resource allocation and performance optimization in real-time video communications because of the low computational complexity and predictive property.

The rest of the paper is organized as follows. Section II proposes a compressed domain approach to the transmission distortion modeling. In Section III, experimental results are presented to justify the CDB model and an application example using the CDB model for the multi-user video transmission is provided to verify validity of the model. Finally, a conclusion is drawn in Section IV.

II. TRANSMISSION DISTORTION MODELING

Let us consider a scenario in which the raw video is compressed offline without the knowledge of the eventual network conditions. The objective of the model is to accurately predict the transmission distortion before the video is transmitted. It is assumed that the raw video is pre-encoded by H.264/AVC encoder with motion compensation using a single reference frame. Each frame can be partitioned into one or more slices, each of which consists of a row of macroblocks (MBs). The slices are packetized into several packets with fixed length. Since the decoding resynchronization is done at the slice header for the H.264 video, the loss of any packet in one slice will cause unsuccessful decoding of the whole slice. Thus,
the transmission distortion of the $f$th frame caused by the transmission errors can be calculated as follows:

$$D = \sum_{n=1}^{NS} D(f, n),$$  \hspace{1cm} (1)

where $NS$ is the number of slices in one frame, and $D(f, n)$ is the expected distortion of the $n$th slice in the $f$th frame. From the expected loss probability of the slice, we obtain

$$D(f, n) = (1 - P_L(n)) * D_R(f, n) + P_L(n) * D_L(f, n),$$  \hspace{1cm} (2)

where $P_L(n)$ is the loss probability of the $n$th slice. Here, $D_R(f, n)$ and $D_L(f, n)$ are the distortion when the $n$th slice is received correctly and lost, respectively.

If a slice is lost during delivery, then both the motion vectors and the texture information are lost. Thus, the lost slice will be retrieved by the error concealment scheme at the decoder on the receiver side. We assume a simple error concealment strategy, called Temporal Replacement (TR), available in the decoder. The TR scheme conceals the loss by copying the information of the entire slice at the corresponding location of the latest decoded frame. Owing to its simplicity and efficiency, it is commonly used in most decoders.

If a slice is correctly received, the motion vectors and the texture information can be successfully decoded. However, the information of the MBs in the slice may be dependent on that of the slices in the previous frame through motion prediction. To reconstruct the $n$th slice, the data from the reference frame must be added. As a result, the transmission error in the $(f-1)$th frame may propagate into the $f$th frame.

A. Estimation of $D_L(f, n)$

Let $\hat{F}(f, n, i)$ and $\hat{F}(f, n, i)$ be the $i$th reconstructed pixel of the $n$th slice in the $f$th frame at the encoder and decoder, respectively. If the $n$th slice is lost, then the error concealment scheme is employed to conceal the error at the decoder side. Thus, $\hat{F}(f, n, i)$ will be replaced by $\hat{F}(f-1, n, i)$. Hence, $D_L(f, n)$ is given by

$$D_L(f, n) = E[(\hat{F}(f, n, i) - \hat{F}(f-1, n, i))^2] - E[(\hat{F}(f, n, i) - \hat{F}(f-1, n, i))^2] + E[(\hat{F}(f-1, n, i) - \hat{F}(f-1, n, i))^2]$$

$$= RFD(f, f-1, n) + D(f-1, n),$$  \hspace{1cm} (3)

where $D(f-1, n)$ is the expected distortion of the $n$th slice in the $(f-1)$th frame and $RFD(f, f-1, n)$ is the mean squared error (MSE) of the $n$th slice between the $(f-1)$th and $f$th reconstructed frames at the encoder. It should be noted that the second identity in Eq.(3) is based on the assumption that the frame difference and the channel distortion are uncorrelated [11]. Traditionally, the calculation for $RFD(f, f-1, n)$ employs a per-pixel estimation. However, the direct calculation in the pixel domain requires a complete decoding of the compressed video streams, and hence does not satisfy the real-time requirement owing to its complexity. For calculation efficiency, we estimate $RFD(f, f-1, n)$ using the coding features, which can be directly extracted from the compressed domain of the video streams. The H.264 coding standard employs various intra- and inter-coded prediction modes. The mode selection for coding a slice in accordance with the motion information can indicate the similarity of the slices in the same position of the neighboring frames. We divide every MB into 16 blocks of $4*4$ size. The $RFD(f, f-1, n)$ can estimated as

$$RFD(f, f-1, n) = \sum_{i=1}^{intra} W_i + \sum_{i=1}^{inter} (W_i * \sum_{j=1}^{blk} Q_{i,j}),$$  \hspace{1cm} (4)

where $intra$ and $inter$ are the numbers of the intra-coded and inter-coded MBs in the $n$th slice of the $f$th frame, respectively, and $blk$ is the number of the blocks in an inter-coded MB, which is equal to 16. $W_i$ and $Q_{i,j}$ are stated as follows:

$W_i$ represents the prediction mode weight of the $i$th MB. The H.264 coding standard supports the intra- and inter-coded prediction modes, which also provide abundant MB partition modes. Intra-coded prediction has $16*16$ and $4*4$ partition modes, whereas the inter-coded prediction mode possesses seven MB partition modes, namely $16*16$, $16*8$, $8*16$, $8*8$, $4*4$, $4*8$, and $4*4$ [10]. In a statistical sense, the selection of the prediction modes for an MB together with the motion information represents the degree of similarity between the current MB and those in the reference frame. In the P frames, an MB would use the inter-coded prediction mode if there is a reference MB well matching it and hence, a smaller $W_i$ is allocated for it. Otherwise, the MB uses the intra-coded prediction mode, and should be given a larger $W_i$. Furthermore, for intra-coded MB, the $16*16$ partition mode works best in the homogeneous areas of an image, and the MB using that mode should be given a lower $W_i$ than the one using the $4*4$ partition mode. In the I frames, $W_i$ of the $16*16$ or the $4*4$ intra-coded prediction mode is the same as that in the P frames. The selection of the values of $W_i$ is discussed in Section III.B in detail.

The quantity $Q_{i,j}$ represents the relative motion intensity of the $j$th block in the $i$th inter-coded MB. The use of $Q_{i,j}$ relies on the fact that an MB with significant motion has large dissimilarity with the one at the same location of the reference frame, and its loss would incur serious degradation of the video quality. Therefore, we employed the motion vectors (MVs) to estimate $Q_{i,j}$. Basically, MVs indicate the motion between the current frame and the reference frame. Assuming that the concealment technique used does not rely on MV estimation, spatial correlation between neighboring MBs is lower for high MV values. Note that the MVs of each $4*4$ block are identical with those of the corresponding MB. In H.264, the inter-coded prediction mode has multiple MB partition modes (including $16*16$, $16*8$, $8*16$, $8*8$, $4*8$, $8*4$ and $4*4$; see details in [10]). The $M * N$ is employed to represent the partition mode (such as $8*4$ with $M = 8$ and $N = 4$). Thus, $Q_{i,j}$ is given by

$$Q_{i,j} = \frac{1}{2} (Q_{i,j}^x + Q_{i,j}^y),$$  \hspace{1cm} (5)

$$Q_{i,j}^x = \frac{|MV_{x,j}^x|}{M}, \hspace{1cm} Q_{i,j}^y = \frac{|MV_{y,j}^y|}{N},$$  \hspace{1cm} (6)

where $(MV_{x,j}^x, MV_{y,j}^y)$ is the MV of the $j$th block in the $i$th inter-coded MB. $Q_{i,j}^x$ and $Q_{i,j}^y$ represent the relative motion
The distortion estimation for an inter-coded MB.

Fig. 1. The distortion estimation for an inter-coded MB.

intensity in the X and Y axes. The normalization by the size of the partitioned subblock is to fairly estimate the relative motion degree by removing the influence of multiple MB partition modes. The \( Q_{i,j} \) is calculated as the arithmetic average of \( Q_{i,j}^1 \) and \( Q_{i,j}^2 \) in consideration of computational complexity.

B. Estimation of \( D_R(f,n) \)

When the data of the \( n \)th slice is correctly received, \( D_R(f,n) \) is estimated by

\[
D_R(f,n) = \sum_{m=1}^{M_R} D_{RB}^{MB}(f,n,m),
\]

where \( M_R \) is the number of the MBs in one slice and \( D_{RB}^{MB}(f,n,m) \) is the distortion of the \( n \)th MB of the \( n \)th slice in the \( f \)th frame when the \( n \)th slice is received correctly.

If the \( m \)th MB is intra-coded, it can be accurately decoded, and hence

\[
D_{RB}^{MB}(f,n,m) = 0.
\]

If the \( m \)th MB is inter-coded, even though the motion vector and the texture information can be successfully decoded, the data from the reference MBs in the previous frame must be added. Therefore, we used the distortion of the reference MBs in the previous frame to estimate \( D_{RB}^{MB}(f,n,m) \). As shown in Fig. 1, \( m \) is an inter-coded MB of the \( n \)th slice in the \( f \)th frame. It has the motion vector \( \Delta x, \Delta y \), which is assumed to have been correctly received. Subsequently, the corresponding reference MBs can be specified in the \((f-1)\)th frame, and labeled as \( m_1, m_2, m_3, m_4 \). The areas of the overlapped regions between the MBs used for prediction and the reference MB are marked as \( S_1, S_2, S_3, S_4 \), respectively. Note that \( S_1, S_2, S_3, S_4 \) can be calculated in terms of the motion vector \( \Delta x, \Delta y \) of the \( m \)th MB. Thus,

\[
D_{RB}^{MB}(f,n,m) = \frac{1}{S_{slice}} \left( (S_1 + S_2) * D(f-1,n') + (S_3 + S_4) * D(f-1,n'') \right),
\]

where \( S_{slice} \) is the area of a slice, and \( n' \) and \( n'' \) are the slices in which the reference MBs are located, as shown in Fig. 1.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. Comparison of the results by the actual distortion with those by transmission distortion model

For the simulations, video sequences foreman and container are used. Using H.264/AVC reference software JM10.6 [14], the common intermediate format (CIF, 352*288) and the quarter common intermediate format (QCIF, 176*144) video sequences are coded at 25 fps. All the frames except the first one were encoded as P frames. To reduce error propagation due to packet losses, 15 random I MBs were inserted into each frame. The data packet size is 54 bytes. The prediction mode weight \( W_i \) of the inter-coded prediction is set to 1, and that of the intra-coded prediction for 16*16 and 4*4 modes are set to 1.2 and 1.4, respectively. Every simulation is run 100 times to obtain the average transmission distortion. The actual transmission distortion and the estimated one with the CDB model, in the transmission packet error rate (PER) of 10%, are shown in Fig. 2. To validate the accuracy of the CDB model in the wireless networks, we simulate a multihop wireless network transmission with NS-2 and collect the PER as shown in Fig. 3. It can be observed from the figure that the channel is time-varying with a burst packet error ratio. Simulation results of the selected values of the PER (Fig. 3) are shown in Fig. 4, and both are normalized to 1. From Fig. 2 and 4, we can see that the CDB model is quite accurate and robust for the test videos in CIF and QCIF in the fixed PER channel and the time-varying wireless channel. To numerically evaluate the estimation accuracy, the following formula is used to measure the average relative error (ARE):

\[
e = \frac{\sum_{f=1}^{F} |\bar{D}_e(f) - \bar{D}_t(f)|}{F} \times 100\%,
\]

where \( F \) is the number of video frames. Here, \( \bar{D}_e(f) \) and \( \bar{D}_t(f) \) are the normalized estimated and actual transmission distortions, respectively. We compare our CDB model with the well-known ROPE approach presented in [2] and the Loss Pattern Probability (LPP) approach presented in [5]. Table I shows the ARE for the test videos. It can be seen that the ARE of the CDB model is less than 10% for all cases. They imply that the CDB model is accurate and robust. Moreover, the performance of the CDB model is substantially better than the LPP approach, but worse than the ROPE approach. Specifically, the ROPE approach significantly outperforms the CDB model for the high bit rate video. It is because the residual has more impact with the increasing of bit rate. The ROPE approach well considers the residual information at the cost the high computational complexity. The influence of the bit rate to the accuracy of the CBD model is discussed in Subsection C in details. Furthermore, the CDB model has a limitation - it cannot predict the real values of the distortion but can only estimate the relative values. The reason for this

Fig. 2. Estimation of the transmission distortion of "foreman" for PER=10%.
(a) CIF 600kbps, (b) QCIF 300kbps.
drawback is that the CDB model follows a compressed domain approach and the information extracted from the compressed domain can express the features of the video stream, but cannot be directly used to calculate the different values of the pixels. However, the superiority of the CDB model is obviously demonstrated in the following aspects: (a) it has lower computational complexity, which is favorable for the real-time applications; (b) it has important applications in resource allocation in the real-time multi-user video transmission, where the relative distortion values are sufficient for the control purposes and the accuracy can satisfy the requirement. Subsection D and E verify these advantages by simulations.

B. Discussion of the selection of the model parameters

By definition, parameter $W_i$ in Eq.(4) represents the prediction mode weight of the $i$th MB, which influences the accuracy of the CDB model. Deriving the model parameter theoretically is, however, quite difficult. Here detailed discussions are given in terms of the effect of the $W_i$ on the accuracy of the CDB model. As described in Subsection A, the relative value of the distortion is estimated by the CDB model. Therefore, we only focused on the proportion between $W_i$ of the inter-coded prediction to that of the intra-coded prediction for both the 16*16 and 4*4 modes. We compare the accuracy of the CDB model at $W_i = (\beta,1.1\beta,1.2\beta)$, $(\beta,1.2\beta,1.3\beta)$, $(\beta,1.3\beta,1.4\beta)$, and $(\beta,1.4\beta,1.6\beta)$ for both the inter-coded prediction and the intra-coded prediction of the 16*16 mode, and 4*4 mode, respectively. $\beta$ is the common factor, and can be set to arbitrary real number. Fig.5 illustrates a normalized trace of Eq.(4) and the actual dissimilarity of the neighboring frames for the video sequences with 300kbps. All the traces are normalized to 1. In Table II, the deviation values are provided. For clear presentation, the four sets of values of $W_i$ are noted as V1, V2, V3 and V4, respectively. It can be seen that the curve with $W_i = (\beta,1.2\beta,1.4\beta)$ fits best with the actual RFD and give the smallest deviation values. Table III shows the ARE of video sequences in the PER of 10%. We can also see that $W_i = (\beta,1.2\beta,1.4\beta)$ gives more accurate results for all the test videos at different bit rates.

C. Discussion on the influence of the bit rate to the accuracy of the CDB model

An important factor affecting the accuracy of the CDB model is the bit rate of videos. Fig. 6 shows the ARE of CIF video sequences at the bit rate of 150kbps, 300kbps,
Fig. 5. Normalized trace of Eq.(4) and the actual RFD for different values of $W_i$.

Fig. 6. ARE at different bit rates. (a) three test videos at PER=10%, (b) average of all test videos for different channels.

450kbps and 600kbps, respectively. Fig. 6(a) demonstrates the results for “foreman”, ”coastguard” and ”container” in PER of 10% and Fig. 6(b) shows the ARE of the average of all test videos for different channels. We can see that as the bit rate goes up, the accuracy of the model decreases. This phenomenon is due to the influence of the coding residuals. Considering of computational complexity, the coded residuals are not precisely calculated. At the low bit rate, the influence of the coded residuals is so small that it is acceptable to ignore. When the bit rate goes up, residuals have more impact on the accuracy of the model. However, it is difficult to retrieve the residuals in compressed domain for H.264 video streams, because of the utilization of the new compression techniques, such as intra prediction coding and 4*4 DCT [8]. A compressed-domain approach for computing residuals is newly reported in [13]. However, it is still with high computational complexity.

Although the accuracy decreases, the CDB model is applicable and effective at the high bit rate for our objective applications. Subsection E provides an application example using the model for the multi-user transmission of test videos at the bit rate of 600 kbps.

D. Complexity analysis

A significant superiority of the CDB model is the low computational complexity. The procedure of the modeling consists of two steps: feature extraction and distortion estimation, both of which consume shorter time and fewer operations compared with the pixel-based models.

In the step of the feature extraction, only the information of MB prediction modes and motion vectors are required in the CDB modelling. The information can be directly extracted from the compressed domain. On the contrary, the pixel-based models, such as ROPE and LPP, have to decode the pre-coded videos and reconstruct frames completely, which is obviously time-consuming. To study the time complexity, we measure the CPU time of the feature extraction for six CIF video streams by the H.264/AVC reference software JM10.6 on a P4 2.8-G PC with 1-G RAM. The results show that the processing time of the feature extraction in the CDB model is only 42.8% of that in the ROPE and LPP approaches.

In the step of the distortion estimation, the CDB model is based on the MB level estimation. On the contrary, the ROPE and LPP approaches are based on the pixel level estimation, and computation is operated per pixel. For CIF videos, 28422 addition/multiplication operations are needed for one frame in the CDB model. However, since the ROPE approach needs to track two moments at every pixel, 11 or 16 addition/multiplication operations are required for a pixel belonging to the intra-coded or inter-coded MB. Therefore, the ROPE approach needs 1622016-2128896 addition/multiplication operations for one frame. The computation complexity of the LPP approach exponentially increases with the length of the GOP, e.g. it needs 1622016 addition/multiplication operations when the length of the GOP $N$ is set to 5. Therefore, the number of operations in the CDB model is approximately 1.34%-1.75% of the number in the ROPE and LPP approaches. Note that the temporal error concealments used by all the three models require negligible additional complexity.

E. An application example of multi-user video transmission using the CDB model

Let us consider an application scenario where a base station delivers the video streams to three mobile users located in its cell. All the users are assumed to be sharing the same network resources but requesting different video contents (foreman, coastguard, and mother&daughter, at CIF with 600 kbps). The total transmission capacity of the system is assumed to be 900 ksymbols/s. Multiple transmission modes are available to each user, with each mode representing a set of a specific modulation scheme, a Convolutional Code, and a Reed-Solomon Code (see Table IV), corresponding to a specific coding rate. It is further assumed that there are 10 cases of resource allocation for time arrangement in the TDMA-based multi-user scheduling as shown in Table V. For instance, when a user is...
TABLE IV
TRANSMISSION MODES

<table>
<thead>
<tr>
<th>Mode</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modulation</td>
<td>QPSK</td>
<td>QPSK</td>
<td>16QAM</td>
<td>64QAM</td>
<td>64QAM</td>
</tr>
<tr>
<td>RS Code</td>
<td>(32,23,7)</td>
<td>(32,30,2)</td>
<td>(64,48,16)</td>
<td>(80,70,10)</td>
<td>(108,96,12)</td>
</tr>
<tr>
<td>CC code</td>
<td>2/3</td>
<td>4/5</td>
<td>2/3</td>
<td>4/7</td>
<td>3/4</td>
</tr>
<tr>
<td>Coding Rate</td>
<td>1.0</td>
<td>1.5</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>

TABLE V
TEN CASES OF TIME ARRANGEMENT

<table>
<thead>
<tr>
<th>User</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>User1</td>
<td>1/3</td>
<td>4/9</td>
<td>4/9</td>
<td>3/9</td>
<td>3/9</td>
<td>2/9</td>
<td>2/9</td>
<td>2/9</td>
<td>1/3</td>
<td>1/6</td>
</tr>
</tbody>
</table>

assigned 2/9 of the total transmission time, the transmission mode 4 (shown in Table IV) should be chosen to transmit the video with the bit rate of 600 kbps. Based on the CDB transmission distortion model, the decision function is defined as

$$F(x) = \sum_{k=1}^{3} D_k(x),$$  \hspace{1cm} (11)

where $D_k(x)$ is the estimated distortion of the $k$th mobile user when the $x$th resource allocation case is employed, which is estimated by Eq.(1). The optimum decision of the resource allocation $x_{opt}$ can be chosen by

$$x_{opt} = \arg \min_{x \in X} F(x),$$  \hspace{1cm} (12)

We compare three resource allocation schemes. They are: 1) CBDRA scheme: using the CBD model as the decision function. 2) CLD scheme: using the LPP model as the decision function proposed in [7]. 3) No optimization scheme: assigning the time slots equally to all the users. In other words, each user is assigned 1/3 time slots and the transmission mode 4 in the Table IV is employed for each user. Fig. 7 shows the average quality of the received video streams. In Fig. 7(a), the results are averaged over each user as well as 200 channel realizations. We use the widely accepted Peak Signal-to-Noise Ratio (PSNR) as a measure of video quality. The CBDRA scheme outperforms the CLD scheme by 1.46 dB, and the no optimization scheme by 2.82 dB. The performance of the CBDRA scheme is better than those two other schemes, because the objective function using the CDB model more effective. Fig. 7(b) shows the cumulative density function (CDF) of the average PSNR. It can also be seen that the CBDRA scheme significantly outperforms the other two schemes with respect to average received quality. Furthermore, the efficiency of the CBD model can be noted.

IV. CONCLUSION

In this paper, a compressed domain approach to the transmission distortion modeling has been proposed. The approach has a much lower computational complexity when compared with that in the conventional pixel-domain-based methods. The experimental results demonstrate the accuracy and robustness of the model. An application example of the resource allocation in real-time multi-user video communication using the proposed model validates the availability and effectiveness of the proposed model. This model can be widely used in multi-user video scheduling, cross-layer design and so on.

In our next step of this research, we shall develop more robust transmission model for advanced error concealment algorithms, such as Motion Vector Copy algorithm. In addition, more accurate approaches to adaptively choose the model parameters are also worth further study.

REFERENCES