No-reference IPTV Video Quality Modeling Based on Contextual Visual Distortion Estimation

Ning Liao* and Zhibo Chen*
*University of Science and Technology China, China
E-mail: lyne.liao@gmail.com, chenzb@ieee.org

***Abstract*—No-reference IPTV H.264 video quality modeling at bitstream level has just been standardized in ITU-T SG12/Q14 P.NBAMS work group as P.1202.2. Compression artifacts, channel artifacts, and their mutual influence are considered in the database design to reflect the realistic situations. For P.NBAMS, we contributed a no-reference slicing channel artifact measurement method based on contextual visual distortion estimation, shortly named CVD in this paper, which has been accepted into the final P.1202.2 Recommendation due to its best performance in standard competition. In CVD scheme, first we predicted the initial visibility of channel artifacts in an individual frame where packet loss occurs and detected the scene cut artifacts at bitstream level. Second, we applied a low-complexity zero-motion-based visible artifact propagation procedure, which emphasizes the most significant visual distortion rather than equally weights the propagated distortion and the initial distortion. Finally, we modeled the visibility of temporal artifacts by extracting two new features from the contextual distortions. The proposed CVD scheme outperforms or emulates the full-reference metric MSE on the five training databases of P.NBAMS, with an average correlation of 0.838 and an average RMSE of 0.42.**

I. INTRODUCTION

Video quality assessment model in IPTV scenario has recently been finalized in ITU-T P.NBAMS [1] (**N**on-intrusive **B**itstream model for the **A**ssessment of performance of **M**ultimedia **S**treaming) work group as ITU standard P.1202.2 [2]. It is no-reference bitstream-level model, with input being H.264 compressed video bitstream and all transmission packet headers, e.g., UDP/IP/RTP/TS. The model has two modes named parsing mode and decoding mode [1]. In parsing mode, the model operates by analyzing information in the video bitstream without fully decoding the bitstream (i.e. no pixel information is used) for Mean Opinion Score (MOS) estimation. In decoding mode, in addition to the bitstream information which parsing mode uses, the model can also decode parts or all of the video bitstream (i.e. pixel information is used) for MOS estimation. Decoding-mode model should improve the prediction accuracy but at a cost of increased computational complexity.

Online video quality monitoring of IPTV service at gateway or setup box is a major target application of P.NBAMS models. Thus, both prediction accuracy and algorithm complexity are important aspects to evaluate the model.

Test conditions of P.NBAMS databases [3][4] are designed to reflect the realistic application situations, which are more challenging to modeling. In P.NBAMS IPTV scenario, combinations of various coding bitrates and various packet loss rates exist. For example, the coding bit rate varies from 1Mbps to 30Mbps to reflect a wide range of video quality; packet loss rate varies from 0 ~ 2%, and both random and burst loss patterns are included. Furthermore, different packet loss concealment (PLC) strategies, which are employed at decoder, also significantly influences the video quality perceived by end-user. Two types of PLC strategies shortly named freezing PLC and slicing PLC, are employed in P.NBAMS test conditions. Freezing PLC represents the worst PLC strategy, where the pictures affected by packet loss are frozen on the previously correctly decoded picture until the next correctly reconstructed I-frame. The slicing PLC represents the average PLC strategy, where lost macro blocks (MBs) are filled with the pixels of the collocated MBs in certain previously decoded frame. Combinations of coding and channel artifacts (slicing artifacts or freezing artifacts) exist in one database, which requires an integrated framework to handle the joint influence of different types of artifacts on overall visual quality.

We developed a complete solution for parsing mode P.NBAMS in IPTV scenario, where the visual distortion of three types of artifacts (coding, slicing, freezing) are modeled separately and trained on the distinct subset of samples whose visual quality are mainly influenced by the corresponding artifact type. We then modeled the mutual influence of perceptible compression artifacts and channel artifacts based on the output of the individual artifact models with respect to the annoyance levels of the individual artifact types co-existing in a video clip. One idea behind this framework is that the annoyance of different artifact types should be modeled differently since they are caused by different technical features. Moreover, human perception of a naïve viewer is more influenced by the most significant then the secondary artifacts, and is not very sensitive to the difference between artifact types. This framework demonstrated good performance in practice with consistence to human perception property.

* This work was done when authors were working in Research & Innovation, Technicolor.

In this paper, we only deal with our proposed visual distortion estimation model for slicing artifacts, which showed best performance in the international competition among all candidate P.NBAMS models and was adopted into ITU-T P.1202.2 [2].

Visual quality modeling of slicing artifact is quite challenging due to the facts that the error concealment performance is highly dependent on the characteristics of video content affected by loss, and that the temporal extent of the visual quality degradation caused by a lost packet is highly dependent on the variety of H.264 encoding configuration, e.g. prediction modes, multiple reference frames, etc.

Bitstream-level model can utilize the bitstream-level information parsed from compressed video bitstream together with the packet-layer information from all transmission packet headers. In literatures, simple packet-layer features like Packet Loss Ratio (PLR) [5], Burst Loss Frequency (BLF) [6] or frame-layer feature like Invalid Frame Ratio (IFR) [7] have been used in combination with coding bitrate or content classification to predict the video quality. We demonstrated in [8] that estimating the visibility level of loss-impaired frame based on video complexity can significantly improve the prediction accuracy of over-all video quality, as compared with PLR/BLF/IFR solutions because the dependency of error concealment effectiveness on video characteristics is explored to an extent.

With bitstream-level information, the error concealment effects and the spatio-temporal error propagation effects can be more accurately considered into quality assessment model. Authors in [9] proposed a no-reference metric by counting the Error-Concealed Macro Blocks (EC_MBs) for which PLC is ineffective. Authors in [10][11] studied the contributing factors to the visibility of packet loss in video transmission. It is shown that, the initial Mean Square Error (MSE) of the loss-impaired frame is an important factor to the artifact's visibility and varies significantly with different PLC strategies and video content types. Authors of [12][13] applied a no-reference end-to-end MSE estimation model to video quality assessment, which tried to accurately take into consideration the effects of error concealment, content characteristics, and error propagation. However, full-reference MSE does not perfectly match with perceptual quality [14], let alone the no-reference estimated MSE. In [15], visibility of packet loss was estimated at first using support vector regression with many features extracted from bitstream, and then was used to weight the no-reference MSE estimation to improve the prediction accuracy of perceived quality.

In comparison of our proposed model with the previous literatures, first, we estimate the visibility level of each EC_MB rather than that of a packet loss and we treat the scene cut artifacts differently. We detect scene cut artifact location at bitstream level, where the decoded pixel signal information is not available and the scene cut frame in the original video may be totally lost after transmission. Second, the estimated visibility level rather than the estimated MSE of the initial artifact in an EC_MB is involved into a low-complexity visual distortion propagation procedure, in order to achieve better correlation with perceptual quality. In the proposed visible artifact propagation procedure, we emphasize the influence of the most significant visual distortion rather than equally weight the propagated distortion and the initial distortion. Besides, we verified that the low-complexity zero-motion-based visible artifact propagation procedure can reach a good tradeoff between complexity and quality prediction accuracy. Third, we modeled the visibility of temporal artifacts, which may be perceived differently when individual frames are displayed continuously, by utilizing the contextual information around the individual frame's artifacts. Hereafter, our proposal is abbreviated as CVD (Contextual Visual Distortion) model.

The structure of the paper is as follows. In section 2, the proposed model is described in details. In section 3, the proposal's performance is compared with the full-reference metric MSE. Conclusions are given in section 4.

II. PROPOSED MODEL

Generally, the goal of error concealment is to estimate lost MBs in order to minimize perceptual quality degradation caused by packet loss. However, visual artifacts may still be perceived after error concealment, because error concealment may be not effective therein. Such visual artifacts caused by EC_MBs are denoted as initial visible artifacts. If a MB having initial visible artifacts is used as a reference, for example, for inter prediction, the initial visible artifacts may propagate temporally to the MBs in other frames through prediction. Such propagated artifacts are denoted as propagated visible artifacts. The overall artifacts, caused by initial and/or propagated visible artifacts, are denoted as overall visible artifacts.

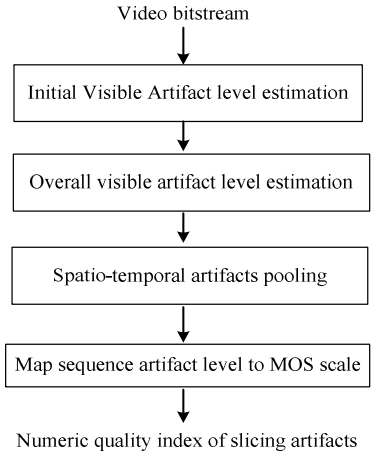

Fig. 1: Flow diagram depicting the proposed CVD model.

The algorithm procedure of the CVD model is shown in Figure 1. The initial visible artifact level, $LoVA_{init}(n,i,j)$, is determined by using the estimated features of the lost content. (n,i,j) indexes a MB in frame n with the coordinate of (i,j). Further, $LoVA_{init}(n,i,j)$ will be propagated temporally to the areas that use the concealed MB (n,i,j) as reference. All

calculation is performed at MB level. That is, for each MB in a frame, a numeric artifact level, $LoVA(n,i,j)$, is obtained. In the spatiotemporal artifact pooling stage, the artifacts level of each MB in a frame is weighted according to its distance to the center of the frame and summed up to obtain the spatial artifact level of the frame, $LoVA(n)$. Then the context-based visible temporal artifact level of a frame n, $CLoVA(n)$, is estimated on the basis of the spatial artifact of all frames in a video clip, and the $CLoVA$ of all frames are aggregated into a sequence-level visible artifact level, $LoVA\_seq$. Finally, map $LoVA\_seq$ to a numeric video quality index.

A. Initial Visible Artifact Estimation

The perceived strength of artifacts produced by transmission errors depends heavily on the employed error concealment techniques. For example, if a frame far away from a current frame is used to conceal a current MB, the concealed MB is more likely to have visible artifacts. In addition, the artifact strength is also related to the video content. For example, a slow moving video is easier to be concealed when temporal error concealment method is used. Thus, two features of an EC_MB, namely, motion magnitude and its distance to the reference frame from which the collocated MB is copied, are used to assess the error concealment effectiveness and the quality of concealed video.

The motion magnitude of the EC_MBs should be estimated, because EC_MBs are MBs lost or not decodable due to loss of previous MBs in the same H.264 slice.

In order to calculate motion magnitude of an EC_MB, it is necessary to differentiate the frame type of the MB at first. If its frame type is I frame, the lost MB inherits the motion magnitude of the collocated MB in the nearest decoded reference frame (i.e. P frame, reference B frame). If its frame type is non-I frame, (i.e., B or P frame), the median value of the available motion vectors (MVs) of its four nearest adjacent neighbors is used as an estimate of the motion vector of the lost MB to deduce the motion magnitude. The term "available motion vector" means that, the corresponding MB is correctly decoded and is an inter-predicted MB. For a correctly received and decoded MB, its motion magnitude, which may be inherited by the collocated lost MB of a later frame, is calculated based on the median MV of the available MVs of its four adjacent MBs and itself. The median operation is used to avoid any anomaly MV, because the MV in the compressed video stream is not true motion. Note that multiple reference frames are used in H.264 predictive coding; thus it is necessary to normalize the parsed MV with a proper reference direction and its distance to reference frame. Taken a 4x4 block of an inter-predicted MB for an example, its MV is normalized as

$$MV_{norm} = \begin{cases} \frac{MV_0}{dist_0}, & forward\ prediction \\ \frac{-MV_1}{dist_1}, & backward\ prediction \\ \frac{MV_0}{dist_0} - \frac{MV_1}{dist_1}, & bidirectional\ prediction \end{cases} \quad (1)$$

Where MV is a vector with two components (x, y); the subscripts (0 or 1) denote forward prediction or backward prediction respectively. $dist_0$ is the distance between the frame to which the MB belongs and its forward reference frame, while $dist_1$ is the distance to its backward reference frame.

The feature $ecdist$ of an EC MB denotes the distance in video display order between the frame in which the EC_MB locates and the reference frame from which the collocated MB is copied. The product of $MV_{norm} * ecdist$ reflects the displacement of an object in a time duration from the copied frame to the current frame. The larger the value is, the more likely the concealed artifacts are visible.

The initial artifacts visibility level for an EC_MB indexed by (i ,j) of frame n is given by

$$LoVA_{init}(n,i,j) = f\left(\left\|MV_{norm_{ij}}\right\| * ecdist/4.0\right) \quad (2)$$

where $$f(x) = \begin{cases} v_1, & ,x < T_1 \\ \frac{(v_2 - v_1)}{thrd_2 - thrd_1} * (x - T_1), & T_1 \le x \le T_2 \\ v_2 & ,x > T_2 \end{cases}$$

The reason for the denominator 4.0 in the above equation is that the motion estimation accuracy in H.264 is quarter sub-pixel. The constants take following values: $v_1 = 0$, $v_2 = 100$; $T_1 = 1$ and $T_2 = 8$ in the unit of pixel.

Note that the feature $ecdist$ not only helps to estimate the error concealment effectiveness more accurately when B-frames or adaptive B-frames are used in H.264 encoding, but also helps to count in the different impacts of random packet loss and burst packet losses (resulting in consecutive frames lost) on visual quality.

B. Scene Cut Artifact Detection

The above visibility estimation of concealed artifacts is based on the assumption that the current picture and the previous pictures are highly correlated, thus the motion and residual in these frames are similar. When scenes change significantly, this assumption usually does not hold. When there is a significant scene change between two adjacent frames i and i+1 and packet loss occurs in the second frame i+1 of the two adjacent frames as shown in Fig. 2(a), the concealed second frame i+1 will have very strong visible artifacts as shown in Fig. 3(a) because temporal error concealment method is used. These artifacts are defined as scene cut artifacts. Scene cut artifacts may also be detected in the first received frame (e.g., frame i+3 in Fig. 2(b)) after one or more subsequent frames (e.g., frames i+1, i+2) have been lost completely and the first received frames (e.g., i+3) is compressed using a lost scene cut frame (e.g., frame i+2) as reference frame. The effect is shown in Fig. 3(b).

Since the concealed artifact occurring at a scene cut frame is very strong, the corresponding MB's visible artifact level is set to a larger value i.e. $LoVA_{init}(n,i,j) = v_2$, when a frame is detected as having scene cut artifact.

The basic ideas to detect a frame with scene cut artifact at bitstream level without reconstructing the pixel information of a video are shown in Fig. 4. One observation is that, a scene cut frame is usually encoded as an I-frame or a P-frame having higher ratio of Intra-MBs, and the prediction residual energies of two different scenes are more probably quite different. This is shown in Fig. 4(a) where scene cuts occur at the 99th frame and the 221st frame. The difference of

prediction residual energies of the 99th frame (I-frame) from the previous I-frame (i.e. the 75th frame) is quite large. The same thing happens to the 221st frame (I-frame) and its previous I-frame, the 199th frame.

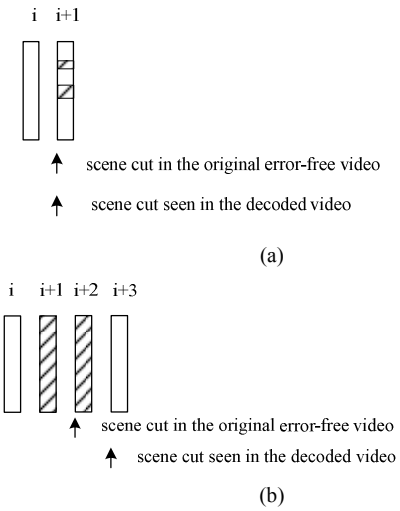

Fig. 2: illustration on how scene cut artifacts relate to scene cut

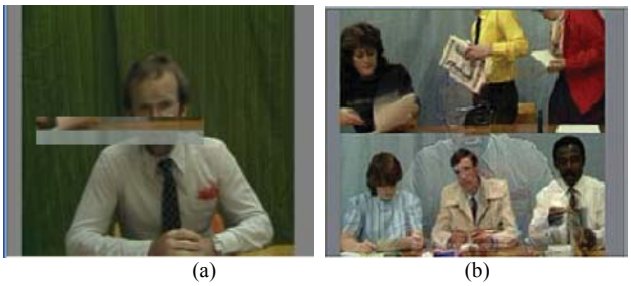

Fig. 3: (a) a pictorial example depicting a frame with scene cut artifacts at a scene cut frame; (b) a pictorial example depicting a frame with scene cut artifacts at a frame which is not a scene cut frame originally. The examples are made from a public video sequence named *trevor*.

Thus, when a partially lost frame is an I-frame or a P-frame with higher intra MB ratio, compare the prediction residual energy difference between the current frame and a preceding I-frame or P-frame with higher intra MB ratio. If the difference between the energy factors is T_3 times larger than the larger energy factor, the impaired frame is detected as a scene cut frame having scene cut artifacts in the decoded video. T_3 is set to 0.45 in the recommendation implementation.

The prediction residual energy of frame n, $E(n)$, is calculated as the average variance (i.e. the energy of AC components) of all correctly received MBs. The variance of a correctly received MB is calculated by following equation:

$$Var = \sum_{k,u,v}\left(X_k(u,v)\right)^2 \times Qstep^2(QP) - DC^2 \qquad (4)$$

where $X_k(u,v)$ are the DCT coefficients. k is the index of the sub-block in the MB and takes value ranging from 0-3 for 8x8 DCT and value ranging from 0-15 for 4x4 DCT. (u,v) is the index of frequency components with value (0,0) representing the DC component. $X_k(0,0)$, $k = 0, \ldots 15$, are not explicitly available after parsing video bitstream in the case that the MB mode is INTRA_16X16. Instead, its Hadamard transform coefficient $Y(u,v)$ is available after parsing syntax. Since both DCT and Hadamard transforms are normalized orthogonal, the total energy of residual is equal to the sum of transform coefficient energy, for example, $\sum_{k=0}^{15} X_k(0,0) = \sum_{u,v} Y(u,v)$ when the MB mode is INTRA_16x16. $Qstep^2(QP)$ is the square of quantization step. QP is the quantization parameter of the MB, which is extracted from syntax parsing with a value from 0 to 51. The quantization step is calculated as:

$$Qstep(QP) = 0.625 \cdot \left(\sqrt[6]{2}\right)^{QP\%6} \cdot 2^{floor(QP/6)}$$

DC energy of the MB is calculated as

$$DC^2 = \begin{cases} \frac{1}{256}\left(Y(0,0)\right)^2 \times Qstep^2, & \text{INTRA\_16x16} \\ \frac{1}{64}\left[\frac{1}{4}\sum_{k=0}^{3} X_k(0,0)\right]^2 \times Qstep^2, & \text{8x8\_TRANSFORM} \\ \frac{1}{16}\left[\frac{1}{16}\sum_{k=0}^{15} X_k(0,0)\right]^2 \times Qstep^2, & otherwise \end{cases} \qquad (5)$$

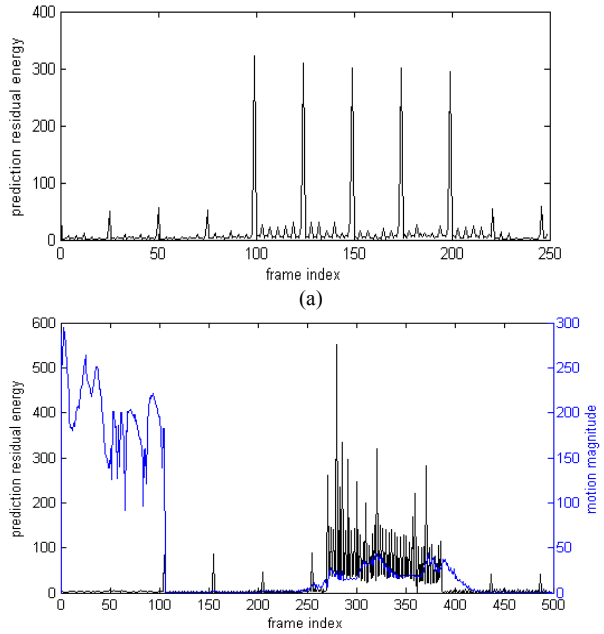

Fig. 4: illustration on how to detect a frame with scene cut artifact

Another observation is that the prediction residual energy change or motion change around a scene change is often greater, as shown in Fig. 4(b) where scene cuts occur at the 105th frame, 271st frame and 387th frame. The motion difference around the 105th frame is significant, and the prediction residual energy differences around the 271st frame and the 387th frame are significant.

Thus, if the residual energy difference (or motion difference) around a candidate scene cut artifact location is T_4 times larger than the larger energy factor (or motion factor), then the candidate frame is detected as having scene cut artifacts. This method is applicable no matter the scene cut frame in the original video is partially impaired or completely lost during transmission as shown in Fig. 2.

Specifically, the candidate frame n is judged as a frame with scene cut artifact if the difference of energy factors in the windows of L_{win} frames before and after current frame n satisfies the following condition:

$$\frac{\left|\sum_{i=1}^{L_{win}} E(n-i)-\sum_{i=1}^{L_{win}} E(n+i)\right|}{max\left\{\sum_{i=1}^{L_{win}} E(n-i),\sum_{i=1}^{L_{win}} E(n+i)\right\}} > T_4 \quad (6)$$

or if the difference of motion factors satisfies the similar condition:

$$\frac{\left|\sum_{i=1}^{L_{win}} M(n-i)-\sum_{i=1}^{L_{win}} M(n+i)\right|}{max\left\{\sum_{i=1}^{L_{win}} M(n-i),\sum_{i=1}^{L_{win}} M(n+i)\right\}} > T_4 \quad (7)$$

$M(n)$ is the average motion magnitude over all correctly received MBs in frame n. T_4 is set to 0.7. Generally, the prediction residual energy of P-frame and B-frame is not at the same order of magnitude, and the prediction residual energy of B-frame is less reliable to indicate video content information than that of P-frame. Thus, we prefer using the residual energy of P-frames in the above Eq. (6). Similarly, motion factors of P-frames are preferred in the above Eq. (7).

Since scene cut artifacts occur at partially received scene cut frames or at frames referring to lost scene cut frames, the frames with or surrounding packet losses may be regarded as potential scene cut artifact locations. Note that when one scene changes gradually to another scene, and if packet loss occurs in a gradual transition frame, the artifacts in the error concealed frame are less visible. This is quite contrary to the scene cut artifacts. However, when scenes change gradually, the energy or motion difference around the gradual transiting frames may still highly probably be large. Thus, it is better to detect and exclude the gradual transiting frames from the candidate scene cut artifact location. We used a preliminary method to detect gradual transition frames. The basic idea of the detection method is that the consecutive frames in a shorter period having larger intra MB ratio than their surrounding frames in a longer period are highly probably the gradual transition frames.

C. *Visual Distortion Propagation*

The initial visible artifacts are propagated temporally when they are used as reference by the inter-predicted frames. How the initial visible artifact propagates can be traced through motion vectors. A less-complex practice is to use zero motion vector instead of pixel-accurate motion vector to roughly track the temporal propagation of visible artifact, i.e., to calculate simply at the MB level instead of the pixel level. We verified on P.NBAMS databases that using the zero motion vectors to roughly track visible artifact propagation can achieve a good tradeoff between complexity and prediction accuracy in quality modeling.

The overall artifact level in a MB, $L\,oVA(n,i,j)$, should consider both initial artifacts and propagated artifacts. Specifically,

$$\mathrm{LoVA(n,i,j)} = \max\left(\mathrm{LoVA(n-k,i,j)}, \mathrm{LoVA_{init}(n,i,j)}\right) \quad (8)$$

where $\mathrm{LoVA(n-k,i,j)}$ is the propagated visible artifact for MB (n,i,j), and frame n-k is the reference frame of MB (n,i,j). The idea behind above equation is that the annoyance level of the overall artifact is determined mainly by the most significant artifact. Note that this is different from the traditional practice of summing up the initial artifact and the propagated artifacts equally. In our P.NBAMS model implementation, it is verified that giving the most significant artifact a 100% weighting factor can achieve much better performance than summing up all artifacts equally.

D. *Context-Based Temporal Artifact Pooling*

First, the center-region-of-interest weighting is applied to obtain each frame's visible spatial artifact level, i.e.,

$$LoVA(n) = \sum_{i=0}^{M}\sum_{j=0}^{N} LoVA(n,i,j) \times \left(1 - \frac{\sqrt{\left(i-\frac{M}{2}\right)^2+\left(j-\frac{N}{2}\right)^2}}{\sqrt{\left(\frac{M}{2}\right)^2+\left(\frac{N}{2}\right)^2}}\right) \quad (9)$$

where M and N are the width and height of a frame in MB unit. The artifacts located near the center of a frame are generally more annoying than those located near the edge of the frame, thus are weighted with larger factors. In Eq. (9), the weighting factor is inversely proportional to the distance of the corresponding MB to the center of the frame.

We use "spatial artifact" here to denote artifact perceived in a frame in a video sequence when the frame is viewed as a still image, and use "temporal artifact" to denote artifact that is perceived in a frame of a video sequence when frames in the video sequence are continuously displayed. Spatial artifact in frames needs to last for a period of time for human eyes to focus and recognize it as artifact. When the frames are part of a video sequence and each is displayed only for a very short period of time (for example, a period of 1/30 seconds when the video is played in a frame rate of 30 fps), the perceived video distortion at the time instant of frame n, i.e., temporal distortion at frame n, CLoVA(n), can be quite different from spatial distortion of frame n, LoVA(n). The perceived temporal distortion at time instant of frame n depends on the context of its neighboring frames. The context includes the duration and the pattern of the distortion.

The flow chart to derive CLoVA(n) at the time instant of frame n is shown in Fig. 5. Two context features are proposed and used to weight the visible spatial artifact LoVA(n). In a neighborhood around current frame n, find a sliding window of $\mathrm{L_0}$ duration, which the current frame belongs to and has a highest density of *large distortion* (i.e., the visible spatial distortion level exceeds a certain threshold, T_5). The *large distortion density* of the identified sliding window is one context feature.

Specifically, as shown in the first step in Fig. 5, a sliding window (denoted as $\mathrm{S_{i,n}}$) of L_0 frames that includes frame n starts at frame $(\mathrm{n-i})$ and ends at frame $(\mathrm{n-i+L_0-1})$, $0 \le \mathrm{i} < \mathrm{L_0}$. For each sliding window $\mathrm{S_{i,n}}$, obtain the ratio (denoted as $\mathrm{R_{i,n}}$) between the number of frames with large distortion in $\mathrm{S_{i,n}}$ and the total number of frames in $\mathrm{S_{i,n}}$, that is,

$$\mathrm{R_{i,n}} = \frac{\sum_{\mathrm{j}} \mu(\mathrm{LoVA(j)})}{\mathrm{L_0}}, \quad \text{frame j} \in \mathrm{S_{i,n}} \quad (10)$$

where $\mu(\mathrm{x}) = \begin{cases} 1, & \mathrm{x} \ge \mathrm{T_5} \\ 0, & \text{otherwise} \end{cases}$, T_5 is the lower-bound threshold of visible spatial artifacts and set to $M \times N/100$ in this implementation. Essentially, the ratio $\mathrm{R_{i,n}}$ is the *large distortion density* for the sliding window $\mathrm{S_{i,n}}$.

Further, identify the window having the highest *large distortion density* among all the sliding windows that include current frame n. The *large distortion density* of the identified

window is used as a context feature to weight current frame's spatial distortion later. The context feature is calculated as:

$$w_n = \max\{R_{i,n},\ 0 \le i < L_0\} \qquad (11)$$

Another context feature is the distance (denoted as dist(n)) between current frame n and the closest frame with large distortion in the sliding window corresponding to the highest *large distortion density*, W_n. If there is no other frame with large distortion in the corresponding sliding window, leave dist(n) to the default value, a very big value of 1000. The idea is that, when two frames with large distortion are closer, the distortion becomes more visible to human eyes.

The perceived video distortion at frame n is calculated as the weighted distortion:

$$\mathrm{CLoVA}(n) = w_n \times \mathrm{LoVA}(n)/\mathrm{dist}(n) \qquad (12)$$

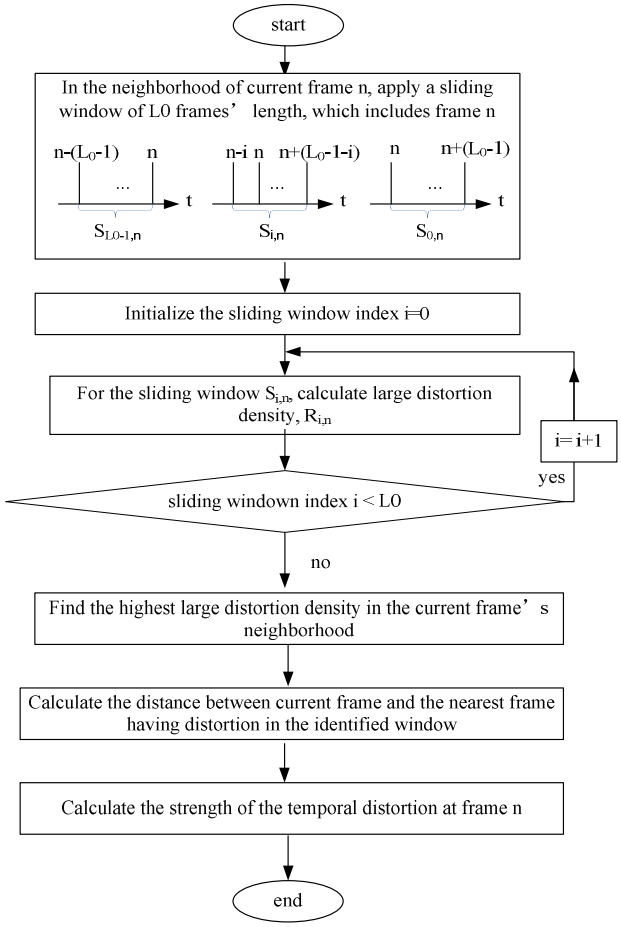

Fig. 5: Flow diagram depicting the method for modeling temporal distortion at frame n

competition (i.e., in ITU-T P.NBAMS work group). Generally, full-reference metric is supposed to be more accurate than no-reference metric. Mean square error (MSE) is a full-reference metric widely used e.g., in [12][13]. We will show that the proposal outperforms or emulates MSE metric in this section.

A. Databases' Description

Five training databases from ITU-T P.NBAMS standard working group for IPTV scenario are used to test our proposed model in this section. Each database includes 240 processed video sequences (PVSs). 24 valid subjects are recruited to rate the PVSs of a database. Five-scale ACR-HR (Absolute Category Rating with Hidden Reference) [16] is used in the subjective video quality assessment test. For each PVS, a mean opinion score (MOS) is obtained by averaging the rates of 24 valid subjects.

The PVSs dominated by slicing artifacts on each database (i.e., their corresponding PVSs without packet loss have MOS larger than 3.5) are used to test the above proposal, and the number of such PVSs is denoted as num_slc in Table 1. Actually, in realistic IPTV services, majority of the transmitted videos are compressed with a quality without annoying coding artifacts. This fact is also reflected in the P.NBAMS IPTV databases' design, where only two of the five databases have PVSs with annoying coding artifacts (MOS lower than 3.5 points) and slicing artifacts as well with a ratio of about 10% and 25%.

Table 1. General configuration of five databases

	db1	db2	db3	db4	db5
resolution	1080p	1080p	720p	1080i	PAL
slicing mode	frame	MBrow	frame	frame	MBrow
Framerate	30fps	25fps	50fps	30fps	25fps
num_slc	79	98	96	67	97
num_scut	3	31	8	6	8
bitrate(Mbps)	2,5,6,15	5, 15	5,15	3,7,15	2,4,6
max PLR	0.8%	1.5%	1%	1%	2%

The major configuration parameters of the available databases till now are shown in Table 1. Slicing mode is an H.264 encoding option. Slice is an independent coding unit from other slice. In one frame per slice mode, if one MB is lost due to packet loss, then the following MBs in the frame in scanning order will be discarded even if their data are received correctly. In one MB row per slice mode, the loss in one MB row will not influence the decoding of the MBs in other correctly received MB rows. Obviously, by employing MB-row-per-slice mode, a compressed video can tolerate more packet loss than that using one-frame-per-slice mode while achieving similar level of video quality. num_scut denotes the number of PVSs having packet loss occurred in scene cut frame.

B. Comparison with MSE

MSE of a PVS is calculated as the average MSE of all frames in it. The MSE of a degraded frame $\{s_k\}$ and its reference frame $\{r_k\}$ is defined as:

$$\mathrm{MSE} = \frac{1}{P}\sum_{k=0}^{P}(s_k - r_k)^2 \qquad (15)$$

where P is the total number of pixels of a frame. For a PVS having packet loss, its mse_all is calculated using the original sequence without compression artifacts as reference; whereas

mse_slc is calculated using the PVS only with compression artifacts as reference, from which the evaluated PVS is derived by discarding some packets.

In Table 2, the Spearman Correlation (SC) between MSE and subjective MOS, the SC between the proposed slicing artifact evaluation metric CVD and the subjective MOS, and the rooted mean square error (RMSE) of CVD are given for the samples without scene cut artifacts. In Table 3, the SC and RMSE are given based on all the samples with slicing artifacts (i.e., including samples with scene cut artifacts). Note that, Spearman Correlation, instead of Pearson Correlation, is used for performance evaluation of MSE, because the mapping between MSE and MOS value is nonlinear.

Table 2. Spearman correlation between metrics and MOS

	db1	db2	db3	db4	db5
num_slc – num_scuts	76	67	88	61	89
mse_all vs MOS	-0.495	-0.593	-0.856	-0.719	-0.152
mse_slc vs MOS	**-0.813**	**-0.844**	**-0.876**	**-0.834**	**-0.729**
CVD vs MOS	**0.882**	**0.802**	**0.881**	**0.857**	**0.741**
RMSE_CVD	0.477	0.407	0.368	0.365	0.454

Table 3. Spearman correlation between metrics and MOS

	db1	db2	db3	db4	db5
num_slc	79	98	96	67	97
mse_allvs MOS	-0.546	-0.586	-0.868	-0.769	-0.302
mse_slcvs MOS	**-0.830**	**-0.888**	**-0.884**	**-0.863**	**-0.768**
CVD vs MOS	**0.883**	**0.777**	**0.892**	**0.873**	**0.765**
RMSE_CVD	0.486	0.432	0.368	0.370	0.450

Interestingly, as shown in Table 2 and Table 3, the mse_slc that only counts in channel artifacts outperforms the mse_all that equally counts in the compression artifact and the channel artifact by large margin on db1, db2, and db5. This is consistent with our idea that, the overall quality is determined by the dominant visual distortion, and that the discriminative treatment of signal difference caused by different types of artifacts may bring significant performance gain. For fair comparison, mse_slc is referred to as benchmark hereafter.

In Table 2, the proposed metric CVD emulates the full-reference MSE metric, mse_slc, and even outperforms on most databases, e.g., db1, db3-db5. One merit of our model over MSE is that the visual distortion other than the signal difference is measured. In the visual distortion calculation, we considered the relationship of the visible distortion and the video content characteristics. Further, we modeled the error propagation differently from the traditional practice of averagely summing up all distortion. The most significant visible artifact among the propagated artifact and the initial artifact is given the largest weighting.

In Table 3, the proposed metric CVD also emulates the full-reference MSE metric, mse_slc, on most databases, when the samples having scene cut artifacts are also considered. However, the performance gap between CVD and MSE on db2 is much larger. Notice that, on db2, 31 PVSs out of 98 PVSs have scene cut artifacts. As shown in Fig. 3, the signal differences at scene cut artifact location are significant. MSE, which measures pixel signal difference of a distorted frame with its reference frame, can more accurately detect the scene cut location. Our proposed bitstream-level model works without pixel signal information and reference frame information and can only roughly estimate the scene cut location. The estimation is even difficult when a scene cut frame is lost totally, or the adjacent scenes don't show significant changes in overall motion or texture energy. Therefore, the proposed bitstream-level model detects the scene cut artifact location with less accuracy than MSE, resulting in a larger performance gap in db2 in Table 3.

In summary, the proposed model emulates the full-reference MSE metric as shown in Table 2 and 3. The average RMSE of the predicted quality score on five databases is 0.421. This prediction error level is sufficient for practical usage.

C. Role of Context-Based Temporal Artifact Pooling

A simple traditional method to aggregate frames' spatial artifact to a sequence's artifact value is to average the spatial visual artifact over the total number of frames of the sequence. This is denoted as AVD (Average Visual Distortion) solution in Table 4, which gives the Spearman Correlations of two metrics (AVD and CVD) and subjective MOS. It can be seen that, compared with AVD, our proposed context-based temporal artifact pooling method can more accurately reflect the visual distortion when the distorted frames of a video are played continuously in real time.

Table 4. Spearman correlation between metrics and MOS

	db1	db2	db3	db4	db5
AVD vs MOS	-0.844	-0.765	-0.903	-0.856	-0.742
CVD vs MOS	-0.885	-0.768	-0.894	-0.885	-0.766

IV. CONCLUSIONS AND FUTURE WORK

We proposed a no-reference bitstream-level video quality assessment model, which involves the prediction of initial visibility of artifacts, an error propagation procedure, and a context based temporal pooling. The proposal outperforms or emulates the full-reference metric MSE on the five P.NBAMS databases for IPTV scenario, with an average Spearman correlation of 0.838 and an average RMSE of 0.421. For future work, texture and luminance masking effects can be further analyzed to improve model performance.

ACKNOWLEDGMENT

Our thanks to Deutsche Telecom, NTT, YONSEI university, and Netscourt for allowing us to use the databases that they have contributed to ITU-T P.NBAMS standardization work.

REFERENCES

[1] ITU-T Q14/12, "P.NBAMS Terms of Reference," Jan. 2011.

[2] ITU-T P.1202.2, “Parametric non-intrusive bitstream assessment of video media streaming quality – higher resolution application area”, March 2013
[3] ITU-T Q14/12, “P.NAMS and P.NBAMS Test Plan,” Sept. 2011.
[4] ITU-T Q14/12, “Processing Chain for P.NAMS/P.NBAMS Hypothetic Reference Circuits”, Sept. 2011.
[5] K. Yamagishi and T. Hayashi, “Video-quality planning model for videophone services,” Information and Media Technologies, vol. 4 no.1,pp. 1-9, 2009
[6] K. Yamagishi and T. Hayashi, “Parametric Packet-Layer Model for MonitoringVideo Quality of IPTV Services,”IEEE International Conference on Communications, pp. 110-114, 2008.
[7] T. Hayashi†, M. Masuda,T. Tominaga, and K. Yamagishi, “Non-intrusive QoS monitoring methodfor realtime telecommunication services,” NTT technical review, vol. 4 no. 4, pp. 35-40, 2006
[8] N. Liao, ZB Chen, “A Packet-Layer Video Quality Assessment Model with Spatiotemporal Complexity Estimation,” EURASIP Journal on Image and Video Processing, 2011
[9] T. Yamada, Y. Miyamoto, and M. Serizawa, “No-reference video quality estimation based on error-concealment effectiveness,” Packet Video, pp. 288-293,2007
[10] S. Kanumuri, S. G. Subramanian, P. C. Cosman, and A. R. Reibman, “Predicting H.264 packet loss visibility using a generalized linear model,” Proc. ICIP, pp. 2245-2248, 2006.
[11] A. R. Reibman and D. Poole, “Predicting packet-loss visibility using scene characteristics,” Packet Video, pp. 308-317, 2007
[12] AR Reibman, VA Vaishampayan, Y Sermadevi, “Quality monitoring of video over a packet network,” IEEE Trans. Multimedia, vol. 6, no. 2, pp. 327–334, 2004.
[13] M. Naccari, M. Tagliasacchi, and S. Tubaro, “No-reference video qualitymonitoring for H.264/AVC coded video,” IEEE Trans. On Multimedia, vol. 11, no. 5, pp. 932-946, Aug. 2009.
[14] Z. Wang and A.C. Bovik, “Mean Squared Error: Love it or Leave it?,”, IEEE Signal Processing Magazine, vol. 1, no. 1, pp. 98-117, Jan. 2009.
[15] S. Argyropoulos, A. Raake, M. Garcia, P. List, “No-reference video quality assessment for SD and HD H.264/AVC Sequences based on continuous estimates of packet loss visibility,”QoMEX, 2011
[16] ITU-R Rec. BT.500-10, “Methodology for the subjective assessment of the quality of the television pictures,” Mar. 2000