# EVALUATION AND MONITORING OF VIDEO QUALITY FOR UMA ENABLED VIDEO STREAMING SYSTEMS

*R. Venkatesh Babu** *Andrew Perkis* *Odd Inge Hillestad*

*School of Electrical and Electronics Engineering, NTU, Singapore
Centre for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
Trondheim, Norway
e-mail: venkatesh.babu@gmail.com, {andrew, hillesta}@Q2S.ntnu.no

**ABSTRACT**

This paper[1] deals with monitoring user perception of multimedia presentations in a Universal Multimedia Access (UMA) enabled system using objective no-reference (NR) metrics. These NR metrics are designed for an UMA-enabled system, in a novel architecture, for a multimedia viewer. The first metric measures block-edge impairments in a video frame at the receiver end, based on the observation that they occur in regions with low spatial activity. The second metric evaluates the quality of the reconstructed video frame in the event of packet loss. Here, the structure of the artifact is itself exploited for the evaluation. Both the metrics involve low computational complexity and are feasible for real-time monitoring of streaming video in a multimedia communication scenario. Further, in rate-adaptive streaming of video, these metrics could serve as feedback parameters to dynamically adapt the video rates based on network congestion.

## 1. INTRODUCTION

An increasing demand for ubiquitous access to the continuously expanding variety and amount of multimedia content being produced, as well as for end-user terminals and networking facilities, calls for a

[1]This work was supported by the Centre for Quantifiable Quality of Service in Communication Systems, Centre of Excellence" appointed by The Research Council of Norway. http://www.ntnu.no/Q2S/

solution which can facilitate good user experience of media consumption. Some essential aspects of this problem are being addressed through the concept of Universal Multimedia Access (UMA) [1] which deals with the delivery of images, video, audio and multimedia content in general under various network access and resource conditions, communication device capabilities and end- user preferences. To achieve this goal, multimedia contents have to be adapted to meet the limitations of the User's terminal and network characteristics. Such multimedia adaptation could be, for instance, transcoding from one video format to another, or scaling a video in spatial domain so as to fit on the terminal's screen. Furthermore, UMA-enabled systems are required to provide the user with the best possible variation of a multimedia resource that the User is capable of receiving. In this sense, the concept of UMA deals with the quality of the content that is delivered. The quality is treated as an end-to-end Quality of Service aggregate which we choose to view as *Quality of Experience* (QoE). Increasingly, this idea is evolving to include the User and the perception of the media being delivered. In this premise, known as the Universal Multimedia Experience (UME)[2], the network and the terminal are considered purely as means to deliver the content. The aim of this paradigm shift is to enable adaptation of the media content presented to the end User based on the way it is perceived in a specific environment and context. For example, UME might include insertion of subtitles into a video to allow deaf users to follow the spoken content in a video. In other words, UME emphasises the end User, whose ultimate goal is to provide the end user with meaningful content that maximizes the User's QoE.

We call a generic UMA-enabled communication device used to consume multimedia presentation, a UMA viewer. This viewer is central to the notion of QoE. In addition to being a media player, it is required that the UMA viewer incorporates an awareness of UME, resulting in an intelligent behavior regarding the way the content is presented, delivered, and ultimately, its perception by the end user. The latter is a subjective attribute that depends on several sensory factors that are not completely understood and are difficult to evaluate. Nonetheless, there is a clear need for automated evaluation of perceived quality of the rendered multimedia presentation. This calls for devising a metric that would measure the quality of the rendered content, which would be required to be strongly correlated with the way it is perceived by the Users. The metric would be practically useful only if the computational cost involved is low. It would also be required to perform consistently over a wide range of content types. Further, one would require a metric to evaluate the perceptual quality of the content with either limited or no access to the reference, in situations such as video streaming. Such metrics are called reduced-reference (RR) and no-reference (NR) metrics, respectively [3, 4]. Metrics that estimate the perceived quality using the uncompressed original as reference, are called full-reference (FR) metrics.

In the process of coding, distortions introduced by the compression algorithms are responsible for loss of perceptual quality. In particular, for block-based video compression schemes such as ISO/IEC

and ITU video coding standards (e.g. MPEG-1/2/4, H-261/3/4) the main forms of distortion include block-edge impairment, blurring, ringing and the DCT basis image effect [5, 6]. A recent work on NR metric proposed by Yang et al., [7] uses the difference between the corresponding translational regions of adjacent frames for measuring spatial distortion. The region correspondence is obtained by estimating the motion vectors, which is computationally expensive. The work by Farias et al., [8] indirectly estimates video quality by measuring degradation of the embedded mark at the receiver end. This method, however, needs the reference video to embed the mark. The reported NR quality metrics have attempted to quantify the effects of these distortions [9, 10], but the emphasis of research has been predominantly on the aspect of block-edge impairment artifacts [11, 4, 12, 13]. This is because, block-edge artifacts tend to be perceptually the most significant of all coding artifacts[11]. With the Video Quality Experts Group (VQEG) working towards their standardization [14], NR metrics remain a topic of great research interest.

For an IP-based network, the deterioration in perceived quality is typically due to packet loss [15, 16], which is a major issue for multimedia over IP. Packet loss typically occurs because packets are dropped at congested routers in a network. In the current best-effort internet scenario, there are no guarantees on network packet loss, delay jitter and the available bandwidth. Indeed, even delay jitter manifests as packet loss, since the packets received after the playout time are not useful. Several methods of concealing packet loss effects in video have been reported [17], but have not been quantified with respect to video quality.

Boyce et al., [16] studied the effect of how MPEG transmission over IP affected the received video quality. Though they discuss the effects of packet loss over various frame types, no quantitative measures have been proposed for measuring the perceptual quality of video. Verscheure et al., in [18] analyze the relation between the perceived quality and the encoding bit-rate for MPEG-2 video. Further, they show that PSNR is not a reliable measure of video quality. Their final conclusion indicates that the image quality cannot be improved by acting on the coding bit rate alone. Kimura et al., [19] addressed the issue of creating layers to maximize the perceived quality of video over a given range of network conditions. They used the perceptual distortion metric (PDM) proposed by Winkler [20], which is a general HVS based distortion metric, for evaluating video quality. However, this is an FR metric since it needs the reference video frames for evaluation and hence is not be suitable for measuring the effect of packet loss for video over IP. Feamster et al., [15] have analyzed the effect of packet loss on the quality of MPEG-4 video and proposed a model to explain them. They have used re-transmission of data providing extra buffering to reduce the effect of packet loss. The re-transmission based error resilience is not suitable for streaming video over IP since lost data takes at least one round-trip time.

Algorithms to measure blockiness have used a variety of methods to do so. Wang and Bovik proposed

an algorithm based on taking the FFT along the rows and columns to estimate the strength of the block-edges [21], while Vlachos used cross-correlation of subsampled images to compute a blockiness metric [22]. Wu and Yuen proposed a metric based on computing gradients along block boundaries while tempering the result with a weighing function based on the human visual system (HVS) [11]. The computations yielded a number for each frame proportional to the block edge strength for that frame. Similar ideas about the HVS were utilized by Suthaharan [23] and Gao et. al.[13]. The general idea behind these metrics was to temper the block-edge gradient with the masking activity measured around it. This approach utilizes the fact that the gradient at a block-edge can be masked by more spatially active areas around it or, in very dark or bright regions [6]. Several of these approaches have proven to be effective but can be computationally quite complex for real-time implementation.

In this paper, we propose an architecture of UMA viewer for video streaming in the framework of UMA, along with two novel NR metrics. The first metric evaluates the block-edge impairment (or blockiness), while the other measures the effects of packet loss on the overall video frame quality. The proposed blockiness metric is based on measuring the activity around the block edges, to determine its contribution to the overall perception of blockiness in the video frame. The effect of packet loss is measured by exploiting the structural pattern of the artifacts.

These metrics, designed for an UMA viewer, are intended to monitor the quality of video streaming, along with, being a part of the feedback mechanism to assist in adapting the media to network conditions. Unlike the metric reported in [12], the proposed ones do not require any training set for tuning the parameters.

The paper is organized as follows. Section 2 describes the QoS-based UMA viewer and its requirements. Section 3 presents the basic ideas underlying the block-edge impairment metric and describes the way it computes. Section 4 describes the measurement of effect of packet loss. Results and discussion are presented in Section 5. Finally Section 6 concludes the paper.

## 2. QOS-BASED UMA VIEWER

The rapidly growing multimedia content on one hand, and application devices on the other, calls for a concept that can enable users to experience any chosen media content on any available device, under different environments. The above issue is addressed in the concept of Universal Multimedia Access (UMA) [1] which deals with delivery of images, video, audio and multimedia content under various network conditions, user preferences and capabilities of terminal devices. The objective of an UMA enabled system is to provide different presentations of the same information from a single source base using appropriately chosen media conversions. Fig. 1 illustrates the concept of UMA. It shows how

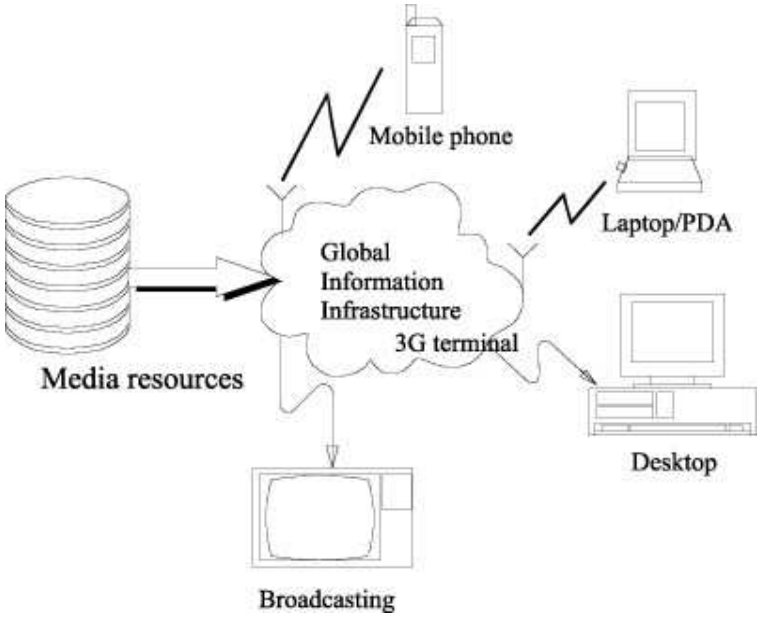



**Fig. 1**. Concept of UMA

users can access the same media resources with different terminal equipment and preferences.

### 2.1. Requirements for the UMA Viewer

The UMA concept places certain requirements on the viewers used for consumption of the presentation. Some of these can be summarized for traditional multimedia consumption as follows:

- Being able to buffer a given amount of data to prevent frame delays during small network traffic variations when the channel characteristics change dynamically. Commercial viewers use this today; the problem is how to take control of the buffer based on dynamic channel and network feedback.

- Being able to use a media description annotation to automatically extract media conversions from an original sequence. This should be done instantaneously and continuously, or alternatively the media descriptor can be as simple as a pointer to the correct conversion of the content on the server.

- Being able to provide adequate support for configuration based on negotiation procedures during time variations in channel conditions and access schemes (also for initial setup).

- Being able to provide intelligent QoS control of the streaming of the content, including fast response to changes in channel bandwidth and automated presentation of changing frame rates.

- Being able to support mechanisms and metrics for quality evaluation and monitoring in order to provide maximum QoE.

Further the viewer should be able to understand the content being streamed so that it can decode the incoming bit-stream. Nowadays there exists a lot of proprietary media content which can be encoded/decoded only by the corresponding proprietary software/hardware. In such a situation, the UMA viewer has no control over those specific media files. This poses problems for providing Quality of Service (QoS). To some extent this problem can be overcome if the media descriptor describes the nature of the content, which can be used to take appropriate actions for viewing the content. For example the media description can be used to invoke the correct decoder. If the media content belongs to some open standards such as ISO/IEC standards, the viewer can take control over the media content for adapting it for the present network condition, channel variations and user limitations. Further it can provide intelligent QoS control over the media.

Packet switched communication using the Internet protocol (IP) is becoming the common denominator for rapidly growing areas of multimedia services and wireless access. Multimedia over IP and wireless networks face many challenges due to the intrinsic natures of these networks such as unknown and dynamic bandwidth, delay jitter and packet-loss. This imposes some necessary trade-offs between QoS guarantee and resource utilization efficiency. These problems need to be tackled intelligently for efficient delivery of multimedia content for various Users.

A media streaming architecture involving UMA viewer is shown in Fig. 2. For better understanding of the functionalities of UMA viewer, only the transactions between the Server and UMA Viewer are briefly explained below.

- *Transaction 1:* The client requests for service to the Content Provider describing his Viewer limitations.

- *Transaction 4:* The available services that satisfy the capability of the user are presented through a menu.

- *Transaction 5:* The User makes the choice of service and requests for the service.

- *Transaction 8:* The service provider adapts the content for the User and sends it to the client for viewing.

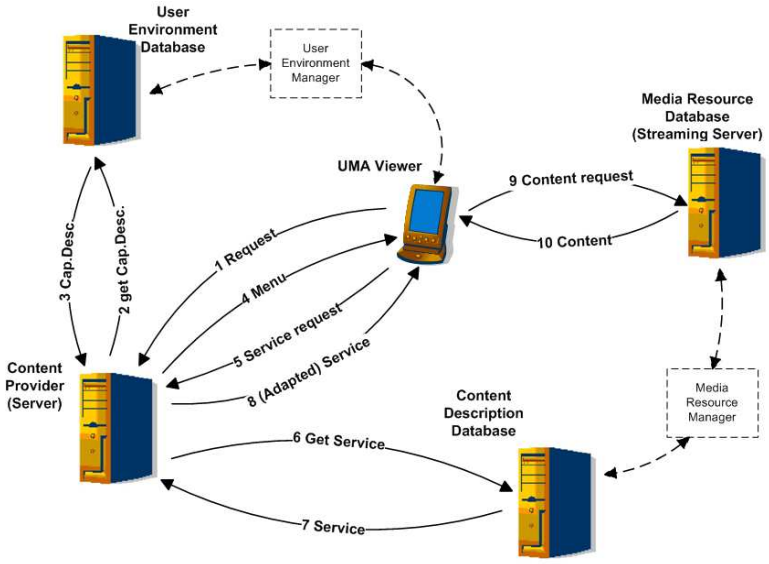



**Fig. 2**. A typical streaming architecture with UMA viewer.

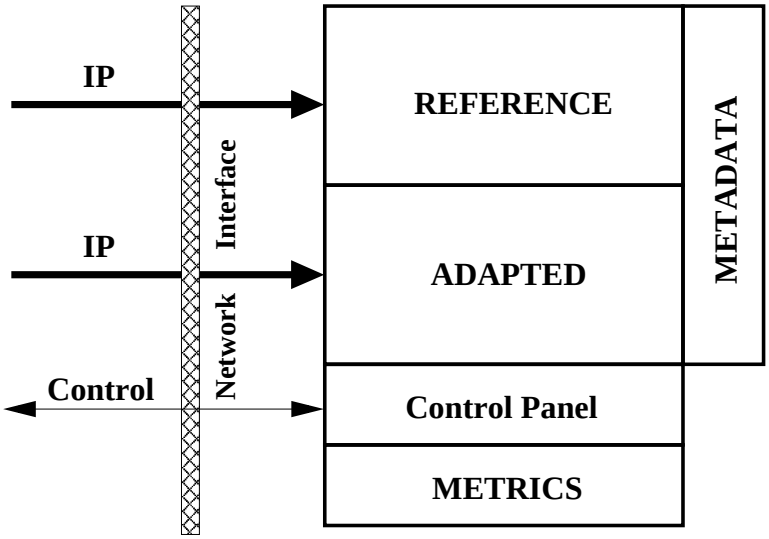



**Fig. 3**. The UMA Viewer.

At present there is no viewer which can provide all the above mentioned basic functionalities. It is therefore necessary to design a viewer incorporating all these functionalities. As a first attempt, we consider the above requirements and propose the architecture shown in Fig. 3. This architecture is meant for laboratory evaluation of an UMA viewer. Here the 'reference' is the original streamed media and the 'adapted' is the stream adapted to suit the detected IP impairments such as network conditions, channel variations and user capabilities. The 'reference' is used in computing 'Full-reference' (FR) and 'Reduced-reference' (RR) metrics. The adapted presentation makes use of the NR metrics.

The control panel in Fig. 3 is used to exploit the feedback possibilities we foresee in UMA enabled systems. For real systems these will be agent controlled as well as user driven. The control signal being delivered to the server in a streaming environment are control packets containing parameterized information extracted from the available metadata and metrics in the viewer. These control packets are fundamental in the system aspects of streaming media controlling the adaptation mechanism in the UMA system such as scalable video coding. Some issues of this system, such as, how to extract control parameters from the available metrics and how to use available metadata, are not covered in this paper.

The 'metrics' window provides the information regarding the quality of the adapted content. As discussed above, the metrics are also envisaged to be included in the control signal to facilitate long term adaptation of media, based on measured quality and network resources. Finally the 'metadata' window lets the viewer show and use available content and descriptors.

In the following section we propose a novel NR metric for measuring the quality of compressed video.

## 3. NR BLOCKINESS METRIC FOR VIDEO

The metric proposed in this work is based on the idea that a block-edge gradient can be masked by a region of high spatial activity around it. It can be observed that blockiness is perceived in a frame due to blocks whose edges exhibit low activity. Let $B_{ij}$ represent an $8 \times 8$ starting at location $(i, j)$ in a given frame. $I_k$, $k = 1, \ldots, 4$ represents the edges of the block as shown in Figure 4. The blockiness is perceptually visible when the edges have low activity for a certain minimum length. It is observed that when a blocky edge length is below 5 pixels, its contribution to the perceived blockiness is negligible. Hence in our experiments, the minimum blocky edge length chosen is set at 6 pixels.

To measure the activity along a given edge $I_k$ we first divide it into three segments of length $6$,

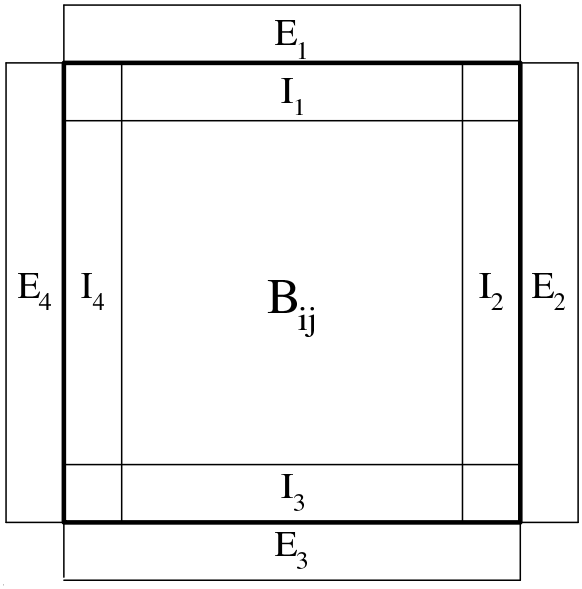



**Fig. 4**. An $8 \times 8$ block and its edges.

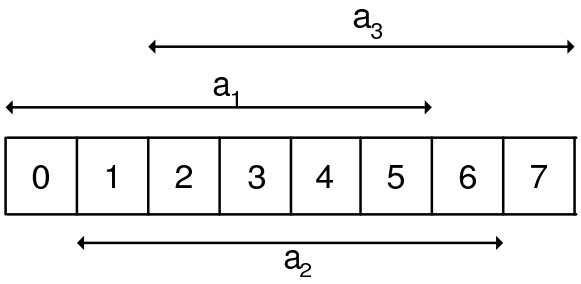



**Fig. 5**. One edge of $8 \times 8$ block.

namely, $a_{k1}$, $a_{k2}$ and $a_{k3}$.

$$\begin{aligned} a_{k1} &= I_k(n) : n = 0 \ldots, 5 \\ a_{k2} &= I_k(n) : n = 1 \ldots, 6 \\ a_{k3} &= I_k(n) : n = 2 \ldots, 7 \end{aligned} \tag{1}$$

This is shown in Figure 5. We quantify activity as the standard deviation, $\sigma_{kl}$ for each $a_{kl}$, and $l = 1, \ldots, 3$. For a given edge $I_k$, $k = 1, \ldots, 4$ activity is defined to be low if at least one of $\sigma_{kl}$, $l = 1, \ldots, 3$ is below a chosen threshold $\varepsilon$. In other words, if there is at least one segment of the edge which has low activity (standard deviation) then the edge and thus the block it belongs to can contribute to the overall perception of blockiness of the frame.

The metric is then computed as follows. For each frame:

1. initialize the block counter $C_B = 0$.

2. In each block $B_{ij}$ along each edge $I_k$, for each $a_{kl}$, $k = 1, \ldots, 4$ and $l = 1, \ldots, 3$ compute the standard deviation, $\sigma_{kl}$.Thus we obtain three activity measures per edge giving us a total of twelve activity measures.

3. Now compute the gradient corresponding to each $a_{kl}$

$$\begin{aligned} \Delta_{k1} &= \text{mean}|I_k(n) - E_k(n)| : n = 0 \ldots, 5 \\ \Delta_{k2} &= \text{mean}|I_k(n) - E_k(n)| : n = 1 \ldots, 6 \\ \Delta_{k3} &= \text{mean}|I_k(n) - E_k(n)| : n = 2 \ldots, 7 \end{aligned} \tag{2}$$

   where $E_k$, $k = 1, \ldots, 4$ are the edges adjacent to the corresponding block edges, $E_k$, $k = 1, \ldots, 4$, as shown in Figure 4.

4. If at least one segment satisfies

$$\begin{aligned} \sigma_{kl} &< \varepsilon \\ \Delta_{kl} &> \tau \end{aligned} \tag{3}$$

   $k = 1, \ldots, 4$ and $l = 1, \ldots, 3$, increment $C_B$ by 1. That is, we count $B_{ij}$ as contributing towards the overall perception of blockiness of the frame.

The overall blockiness measure $\mathcal{B}_F$for the present frame, is then

$$\mathcal{B}_F = \frac{C_B}{\text{Total number of blocks in the frame}}. \tag{4}$$

Clearly, the range of the metric is $[0, 1]$ where 0 corresponds to no blockiness and 1 to the scenario where all the blocks in a frame are visible. The bit depth for the video sequence is assumed to be 8 bits or 255 gray-scale levels. The value of $\varepsilon$ is chosen as a threshold to isolate edges with low activity. To this end, we chose $\varepsilon = 0.1$. This corresponds to the situation when there is a minimal deviation from the mean of the segment. Increasing the value of $\varepsilon$ would result in edges with greater standard deviation being picked. This would mean picking undesired blocks with segments that might have enough spatial activity to mask the block-edge gradient for that edge.

The value of $\tau$ can be chosen so that given low activity, the largest perceivable block-impaired edges will be counted in the metric. Increasing the value of $\tau$ would mean undesired rejection of segments with low spatial activity which also have a block-edge gradient that can be perceived. On the other hand, choosing a very small value of $\tau$ would result in an undesired situation, where an imperceptible edge might result in a block being counted. For our simulations we chose a value of $\tau = 2.0$, because we found that it performed better for a wide range of video sequences.

## 4. NR PACKET LOSS METRIC FOR VIDEO

The loss of video packets often results in loss of slice information, which in turn results in corruption of visual information along macroblock edges. Further, a feature of the MPEG-2 system we consider is that during reconstruction of an affected video frame, a damaged/lost macroblock is simply replaced by the corresponding macroblock from the previous frame. This causes a visible discontinuity when the damage falls in region with motion as shown in Figs. 13 and 14. These structural distortions can be captured by checking edges of the macroblock rows. The length of the artifact along a macroblock is proportional to the damage incurred by a slice. The metric proposed in this section measures the length of the artifact to estimate the damage done to the given video frame because of packet loss. At the same time, it can estimate the efficacy of concealment strategy being used by the decoder. We note that although the concealment strategy considered is simple, this metric can also be applied to gauge the performance of more complex strategies which attempt a smoother concealment of packet-loss artifacts. Indeed, it could be used to further refine concealment algorithms to mitigate packet loss related artifacts. In reported literature thus far, no metric is capable of this evaluation.

### 4.1. MPEG-2 Transport Stream Over IP

In our experiments, for measuring the effect of packet-loss in video over IP scenario, we use MPEG-2 video streams. MPEG-2 specifies the following two systems: i) Program stream (PS): Multiplexed video, audio and data together form a PS which is meant for recording applications such as DVD ii) Transport stream (TS): A packet-based protocol for transmission applications such as cable TV, video on demand and interactive games. PS and TS are created from the 18,800-byte segments of packetized elementary stream (PES) obtained from the source encoder.

The MPEG-2 video stream hierarchy consists of a sequence composed of three types of pictures namely, i) intra-coded (I) ii) predictive-coded (P) iii) bidirectional (B) . Each picture is composed of slices, which are comprised of one or more contiguous macroblocks. Macroblock is the basic coding unit in the MPEG algorithm. It is a $16 \times 16$ pixel segment in a frame. The macroblocks within a slice are ordered from left-to-right and top-to-bottom. Slices are important in the handling of errors. If the bitstream contains an error, the decoder can skip to the start of the next slice. Larger number of slices in the bitstream allows better error concealment, but on the flip side, it uses bits that could otherwise be used to improve picture quality.

Entropy and differential coding lead to the spread of data loss within the frame till the next resynchronization point (*i.e.*, next picture or slice header). In the event of loss in a reference frame (I or P), error propagates temporally due to motion compensation. Although, error concealment techniques reduce the

sensitivity of data loss to an extent, no perfect error concealment technique is known to exist.

### 4.2. Proposed Approach

Consider an image frame of size $m \times n$ (height $\times$ width) and let $R_i$ indicate the $i$th row of the frame. We compute,

$$\begin{aligned} \hat{E}_j &= |(R_{(i-1)} - R_{(i+1)}) * P|, \\ \hat{E'}_j &= |(R_{(i-2)} - R_i) * P|, \quad \text{for } i \in \{16, 32, \ldots, m-16\} \end{aligned} \tag{5}$$

where, $*$ stands for convolution operation and $j = i/16$. $\hat{E}_j$, a row vector of length $n$, gives the edge strength across the macroblock rows $j$ and $j+1$ while $\hat{E'}_j$ gives the edge strength very close to $\hat{E}_j$ within $j$th macroblock. $P = [1,\ 1,\ 1]/3$ is a simple low-pass filter. Each row vector $\hat{E}_j$ and $\hat{E'}_j$ is then subjected to a threshold $\tau$ to give the final binary edges $E_j$ and $E'_j$. In order to avoid the noisy edges and to pick-up the visible horizontal edges, the value of $\tau$ is set at 15 for all our experiments.

$$\begin{aligned} E_j(k) &= \begin{cases} 1 & : \text{ if } \quad \hat{E}_j(k) > \tau; k = 1, 2, \ldots, m \\ 0 & : \text{ otherwise} \end{cases} \\ E'_j(k) &= \begin{cases} 1 & : \text{ if } \quad \hat{E'}_j(k) > \tau; k = 1, 2, \ldots, m \\ 0 & : \text{ otherwise} \end{cases} \end{aligned} \tag{6}$$

Note that it is important to avoid edges belonging to video content in the frame while estimating the effect of packet loss. For this purpose, the $E'_j$ corresponding to the $E_j$ under consideration is used (see Fig. 6). Since $E_j$ and $E'_j$ represent edge maps of adjacent rows, they also have similar edges in a regular undistorted frame. This happens since most images exhibit smooth spatial edge continuity; A sudden horizontal edge across the macroblock boundary of a normal frame is a very rare phenomenon. Hence the effect of packet loss artifact along the macroblock row $j$ is computed as the difference between the edge maps $E_j$ and $E'_j$. A small threshold $\zeta$ is used to avoid the effect of noisy edges. In our experiments, the value of $\zeta$ is set as 10% of the maximum possible error (*i.e.*, 10% of frame width ($n$)).

The effect of packet loss for the $j$th macroblock row is obtained as:

$$H_j = \begin{cases} \sum_i |E_j(i) - E'_j(i)| & : \text{ if } \quad \sum_i |E_j(i) - E'_j(i)| > \zeta \\ 0 & : \text{ otherwise}, \quad j = 1, 2 \ldots \frac{m}{16} - 1 \end{cases} \tag{7}$$

$H_j$ is thus a measure of the extent to which the edge of the slice is corrupted.

The cumulative effect of $H_j$ gives a packet loss metric for the entire frame. In our experiments, the value of $H_j$ is normalized to the range 0 to 1. The cumulative effect of packet-loss for the current frame

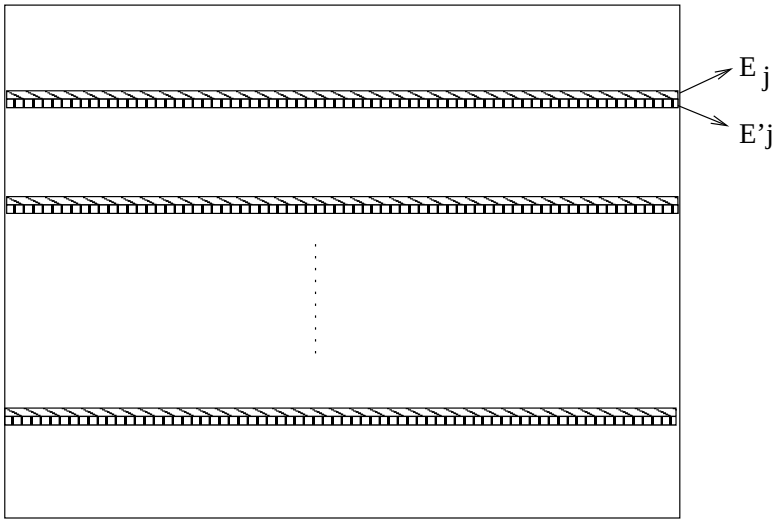



**Fig. 6**. Computation of pocket loss metric. Every 8th row of a frame is considered for measuring packet loss.

is computed as,

$$F = \sum_{j} H_j^2 \tag{8}$$

Since $H_j$ lies between 0 and 1, $H_j^2$ lends more weightage to the higher values of $H_j$ than the lower ones. In other words, lengthier artifacts are given more importance. Since this method exploits the structure of the artifact across the macroblock boundaries, the algorithm is computationally very economical.

We note that the majority of computations involved in both the proposed metrics are the basic arithmetic operations, carried out only on the block-edge pixels, which form a small fraction of the entire image pixels. Hence, these metrics are computationally inexpensive and are suitable for real-time applications.

## 5. SIMULATIONS AND RESULTS

In the following two subsections we describe the experimental setup and results obtained for the proposed NR metrics.

### 5.1. NR blockiness metric

For our simulations we considered video sequences of time span 10sec., in CIF resolution (frame size of $352 \times 288$), 30 frames/sec and YUV (4:2:0) format. For results presented here, we only consider the $Y$ (luminance) channel. The original video sequence was encoded using the XviD MPEG-4 ASP

codec [24] with a GOP size of 30 frames. The NR metric was computed for each frame of the original and the encoded sequence. Here we present results obtained for the "Mother-Daughter" and "Paris" sequences. We compare the performance of the proposed metric with the Wang, Sheik and Bovik (WSB) quality assessment model [12]. MATLAB code for the model was obtained from [25]. Since the WSB metric increases with image quality, and typically ranges from 0 (best) to 10 (worst). For feasibility of comparison with the proposed metric, we normalize it to the range 0 to 1, followed by subtraction from 1, to denote 0 (worst) and 1 (best). Figure 7 shows the result of applying the proposed metric to the first two GOPs (frames 1-60) of the "Paris sequence, while Figure 8 shows the corresponding result for the WSB metric. We note that the proposed metric is nearly zero for the original (uncompressed) sequence. In other words, it measures no blockiness in the uncompressed original video as expected. Further, we observe that the blockiness measure increases with compression; equivalently, it increases as the bit rate decreases. This is consistent with the fact that higher compression implies coarser quantization and consequently increased perceived blockiness. The peaks in the figures 7 and 10 indicate the *I* (intracoded) frame. The peak indicates that the blockiness perceived in I-frame is the highest in a GOP at all bit rates. This behavior is due to the fact that in I frames, DCT is obtained for every block of image pixels (similar to JPEG coding) unlike P or B frames. Blockiness becomes perceptually visible when DCT is applied to image pixels, whose spatial frequency is less than the high frequency error signal. Since P and B frames deal with DCT of the error signal, blockiness of P and B frames is less than that of the I frame. Further, the extent of blockiness in P and B frames decreases as we move away temporally from the I frame. This is attributed to the waning influence of I frame on the subsequent P and B frames in a GOP. The above mentioned behavior is captured more convicingly by the proposed metric (Figs. 7,10), as against the WSB metric (Fig. 8).

Figure 9 plots changes in blockiness with bit rates, for both the metrics, for a chosen frame. Here, we happen to choose frame number 31 which is an *I* (intracoded) frame encoded at various rates, namely, 1234 Mbps, 699 kbps, 489 kbps, 346 kbps, 233 kbps, 186 kbps 147 kbps and 128 kbps. It can be seen that both curves show a graceful degradation indicating consistency in behaviour.

Figure 10 shows the performance of the proposed metric for first 60 frames of the "Mother-Daughter sequence. Here again, we note that the metric results in a blockiness measure of nearly zero for the original uncompressed video frame. We also note that the metric measures maximum blockiness for frame number 31 which happens to be the I-frame.

Figure 11 shows frames 31 (I-frame), 40 and 55 of the "Mother-Daughter" sequence, coded at 88.5 kbps. In consistence with the decrease in visually perceived blockiness, we note that the metric too results in decreasing measure of blockiness from 0.62 for 11(a)(frame 31) to 0.28 for 11(c)(frame 55) . In addition, we also note impairments like blurriness also contributes to the overall perception of the

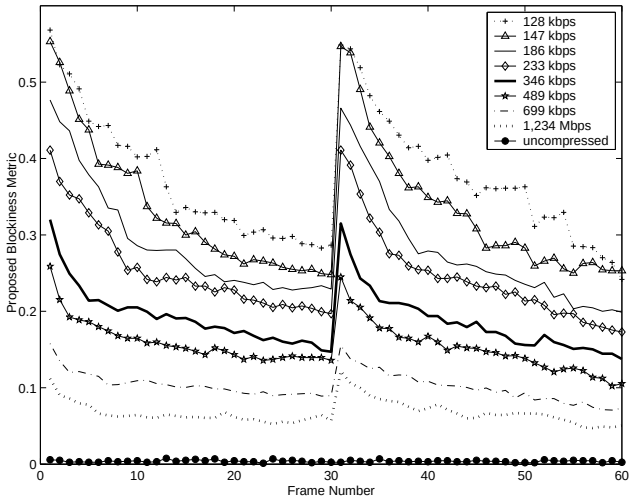



**Fig. 7**. Proposed blockiness metric for the first 60 frames of the ”Paris” sequence coded at different bitrates.

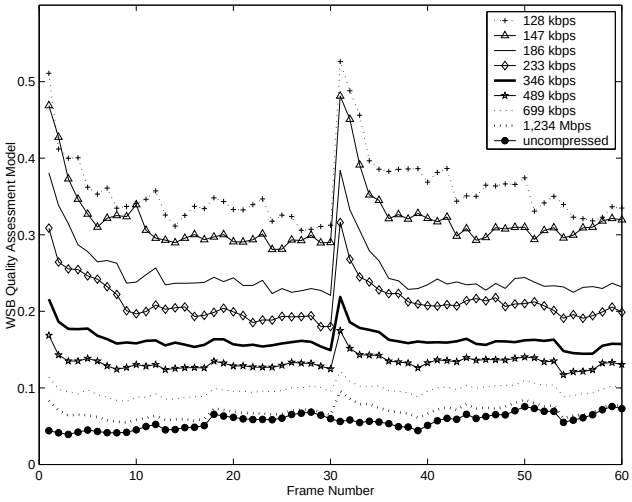



**Fig. 8**. WSB metric for the first 60 frames of the ”Paris” sequence coded at different bitrates.

frame.

Figure 12 shows frame number 31 which is an $I$ (intracoded) frame, encoded at three different rates, namely, 362.1 kbps, 141 kbps and 88.5 kbps, along with the original. The visually perceived blockiness increases from Figure 12(a) through Figure 12(d), as does the value of blockiness measured by the proposed metric. In the original image Figure 12(a), the blockiness measure stands at a mere 0.001, while Figure 12(d) which exhibits visibly substantial blockiness results in the value 0.62. We note the strong correlation between visually captured blockiness and the resulting measures of the proposed metric.

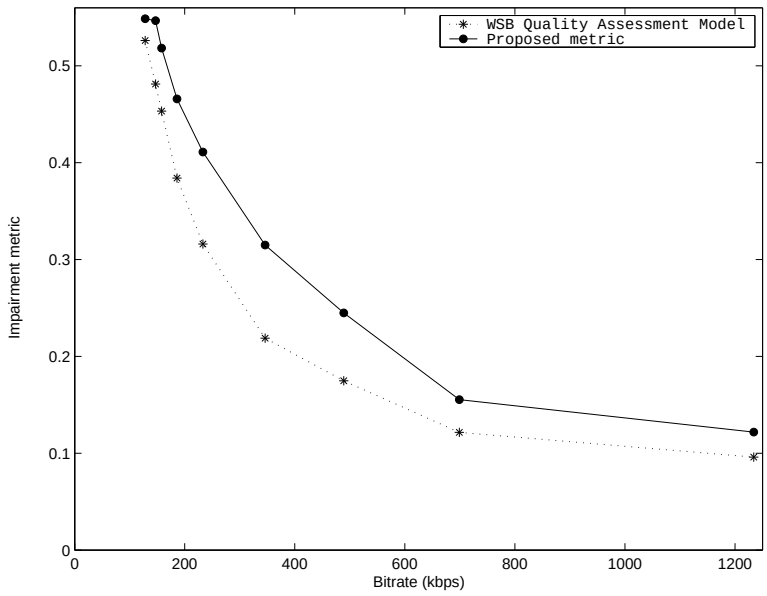



**Fig. 9**. Comparison of the proposed metric and the WSB metric for frame 31 of the "Paris" sequence at different bit-rates.

### 5.2. NR Packet-loss metric

For our experiments we have used the elementary streams provided by Tektronix [26]. All clips were coded to the following specifications: Bit-rate=1.5 Mbps, Frame Rate=30 fps, Frame size = $352 \times 240$, and duration=15 secs. In our simulations, we have used the random packet loss generation program developed by NTT Mobile Communications Network, Inc (DoCoMo) [27] for simulating packet-loss in the MPEG-2 transport stream for various packet loss ratios (PLR). A snap shot of the affected video for PLR=1% and 5% for the "susi sequence and the corresponding values of $H_j$ are shown in Fig. 13, These figures show how $H_j$ is proportional to the length of the artifacts along the macroblock edges. Fig. 15 shows the cumulative packet-loss effect ($F$) for each frame of "susi, and "table tennis sequence for the no packet-loss, PLR=1%, and PLR=5% scenarios. We note that for both sequences the packet loss metric stays very close to zero for the original (no packet-loss case), and increases in proportion to the artifacts in each frame as described above.

## 6. CONCLUSION

In this paper, we have presented an architecture and proposed two novel NR metrics for evaluation and monitoring of an Universal Multimedia Access (UMA) enabled video streaming system. The UMA-enabled viewer serves as the decoder. The UMA viewer receives IP packets from a streaming server and extracts the associated metadata and bit stream for decoding. After decoding, the NR metrics are com-

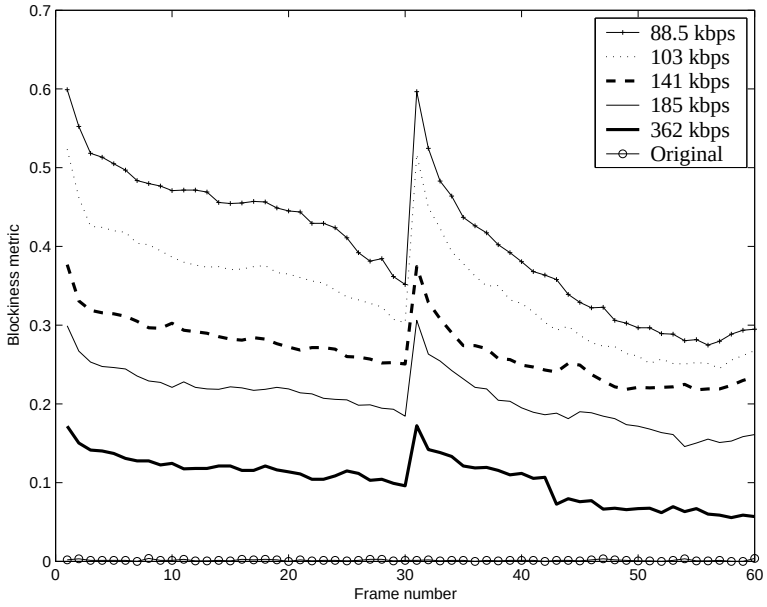



**Fig. 10**. Blockiness measurements for the first 30 frames of the "Mother-Daughter" sequence coded at different bit-rates.

puted. The first proposed NR metric is used for monitoring block-impairment artifacts in a reconstructed video frame, while the other is to evaluate the quality of the reconstructed video frame in event of packet loss. The proposed NR metrics rely solely on the received bit stream at the decoder end. As the experiments illustrate, both the metrics strongly correlate with perceptual observations. In particular, both the proposed metrics monotonically increase with deteriorating video quality (increased compression) and are nearly zero for the original, lossless video stream. Comparisons with the existing metric to measure blockiness are presented. It is illustrated that the proposed metrics capture the behavior of the waning influence of $I$ frame over the blockiness measure of $P$ and $B$ frames, within a $GOP$ more convincingly. The metrics and metadata can be used as an integral part of the control panel providing a system solution to adapting multimedia content for streaming media. The metrics could be improved by considering various other compression artifacts such as blurriness, ringing, etc., for more accurate estimate of video quality. Additionally, inter-frame correlations could be explored as against the currently used intra-frame based measurements of video quality.

## 7. REFERENCES

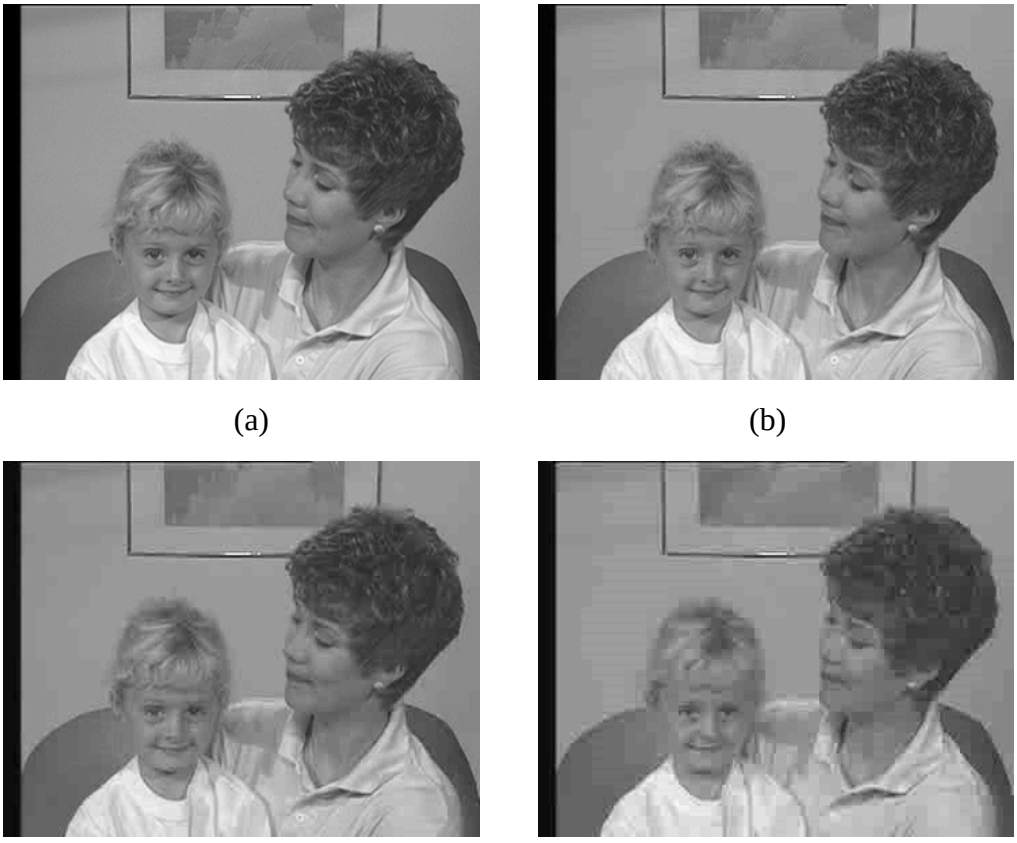
[1] A. Perkis, Y. Abdejaoued, C. Christopoulos, T. Ebrahimi, and J. F. Chicharo, "Universal multimedia access from wired and wireless systems," *Circuits, Systems and Signal Processing; Special issue on Multimedia Communications*, vol. 20, no. 3, pp. 387–402, 2001.

(a)

(b)

(c)

**Fig. 11**. Frame nos: (a) 31 ($\mathcal{B}_F = 0.62$), (b) 40 ($\mathcal{B}_F = 0.38$), and (c) 55 ($\mathcal{B}_F = 0.28$), from the video stream coded at 88.5kbps.



(a) (b)

(c) (d)

**Fig. 12**. Frame no. 31 from the "Mother-Daughter" sequence coded at different rates: (a) original ($\mathcal{B}_F = 0.001$), (b) 362.1 kbps ($\mathcal{B}_F = 0.17$), (c) 141 kbps ($\mathcal{B}_F = 0.37$), (d) 88.5 kbps ($\mathcal{B}_F = 0.62$).

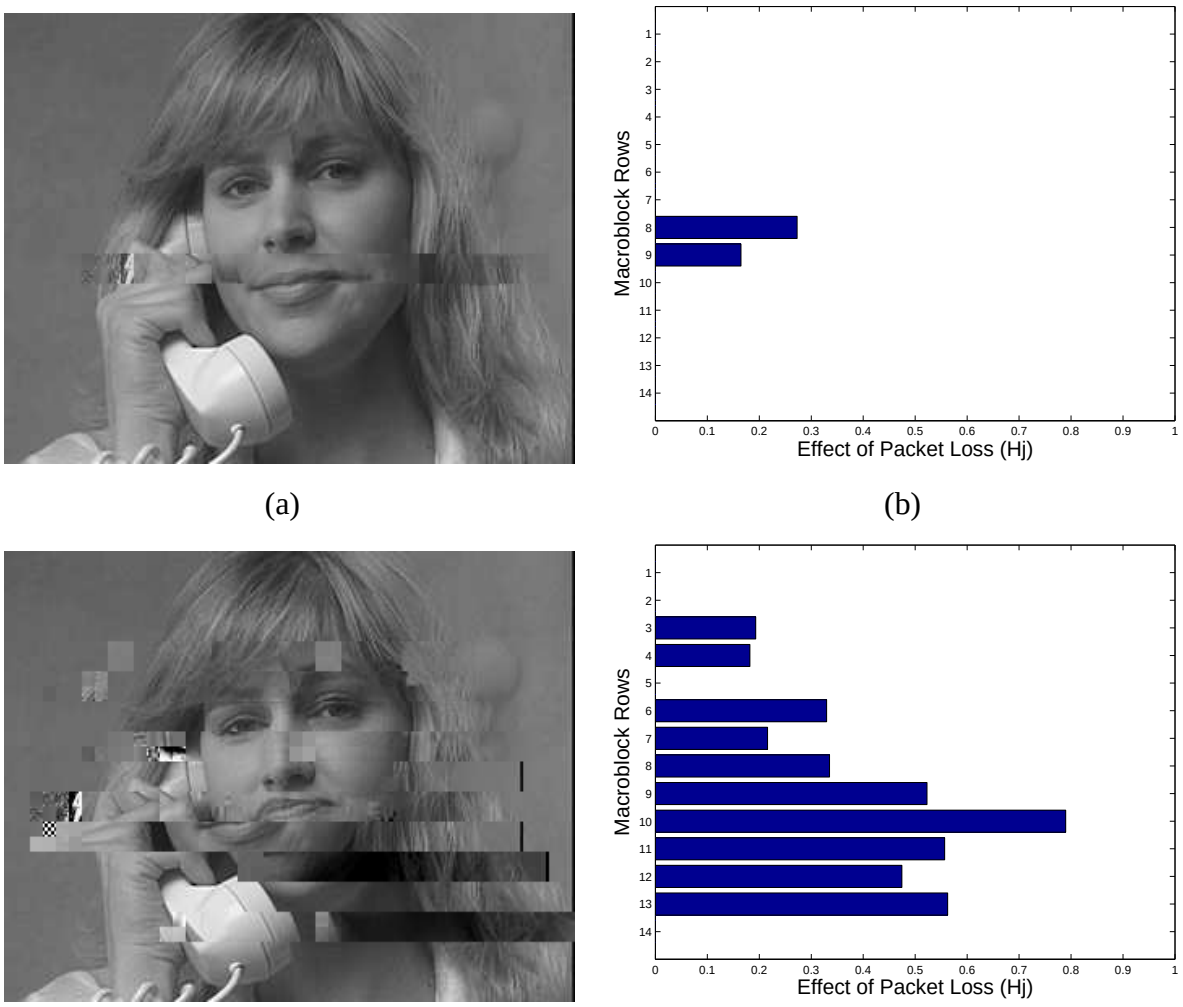



**Fig. 13**. A corrupted frame (frame no. 83) of susi sequence with (a) PLR=1%, (c) PLR=5% and (b,d) the corresponding measure $H_j$ along the macroblock row edges


[2] F. Pereira and I. Burnett, "Universal multimedia experiences for tomorrow," *IEEE Signal Processing Magazine*, vol. 20, no. 2, pp. 63–73, March 2003.

[3] L. Lu, Z. Wang, A. C. Bovik, and J. Kouloheris, "Full-reference video quality assessment considering structural distortion and no-reference quality evaluation of mpeg video," in *IEEE International Conference on Multimedia and Expo*, Yorktown Heights, NY, US, 2002, pp. 61– 64.

[4] S. Winkler and A. Sharma and D. McNally, "Perceptual video quality and blockiness metrics for multimedia streaming applications," in *Proc. 4th International Symposium on Wireless Personal Multimedia Communications*, Aalborg, Denmark, September 2001, pp. 553–556.

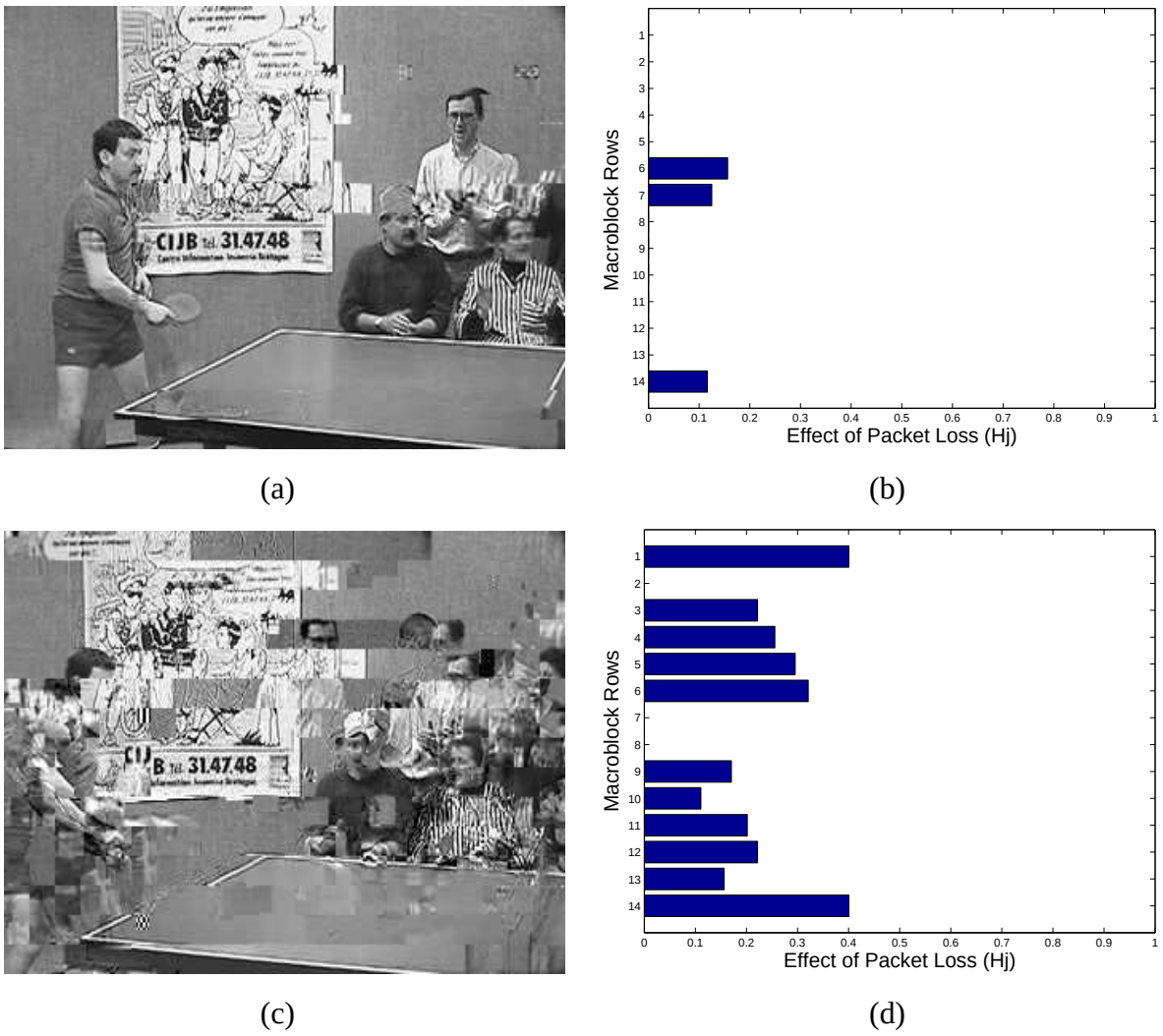



**Fig. 14**. A corrupted frame (frame no. 308) of table tennis sequence with (a) PLR=1%, (c) PLR=5% and (b,d) the corresponding measure $H_j$ along the macroblock row edges

[5] H. R. Wu, M. Yuen, and B. Qiu, "Video coding distortion classification and quantitative impairment metrics," in *International Conference on Signal Processing*, October 1996, vol. 2, pp. 962–965.

[6] M. Yuen and H. R. Wu, "A survey of hybrid MC/DPCM/DCT video coding distortions," *Signal Processing*, vol. 4, no. 11, pp. 317–320, November 1997.

[7] Fuzheng Yang, Shuai Wan, Yilin Chang, and Hong Ren Wu, "A novel objective no-reference metric for digital video quality assessment," *IEEE Signal Processing Letters*, vol. 12, no. 10, pp. 685–688, October 2005.

[8] Myléne C.Q. Farias, Marco Carli, and Sanjit K. Mitra, "Objective video quality metric based on

data hiding," *IEEE Trans. Consumer Electronics*, vol. 51, no. 3, pp. 983–992, August 2005.

[9] J. Caviedes and S. Gurbuz, "No-reference sharpness metric based on local edge kurtosis," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, September 22–25, 2002, vol. 3, pp. 53–56.

[10] Pina Marziliano, Frédéric Dufaux, Stefan Winkler, and Touradj Ebrahimi, "A no-reference percep-

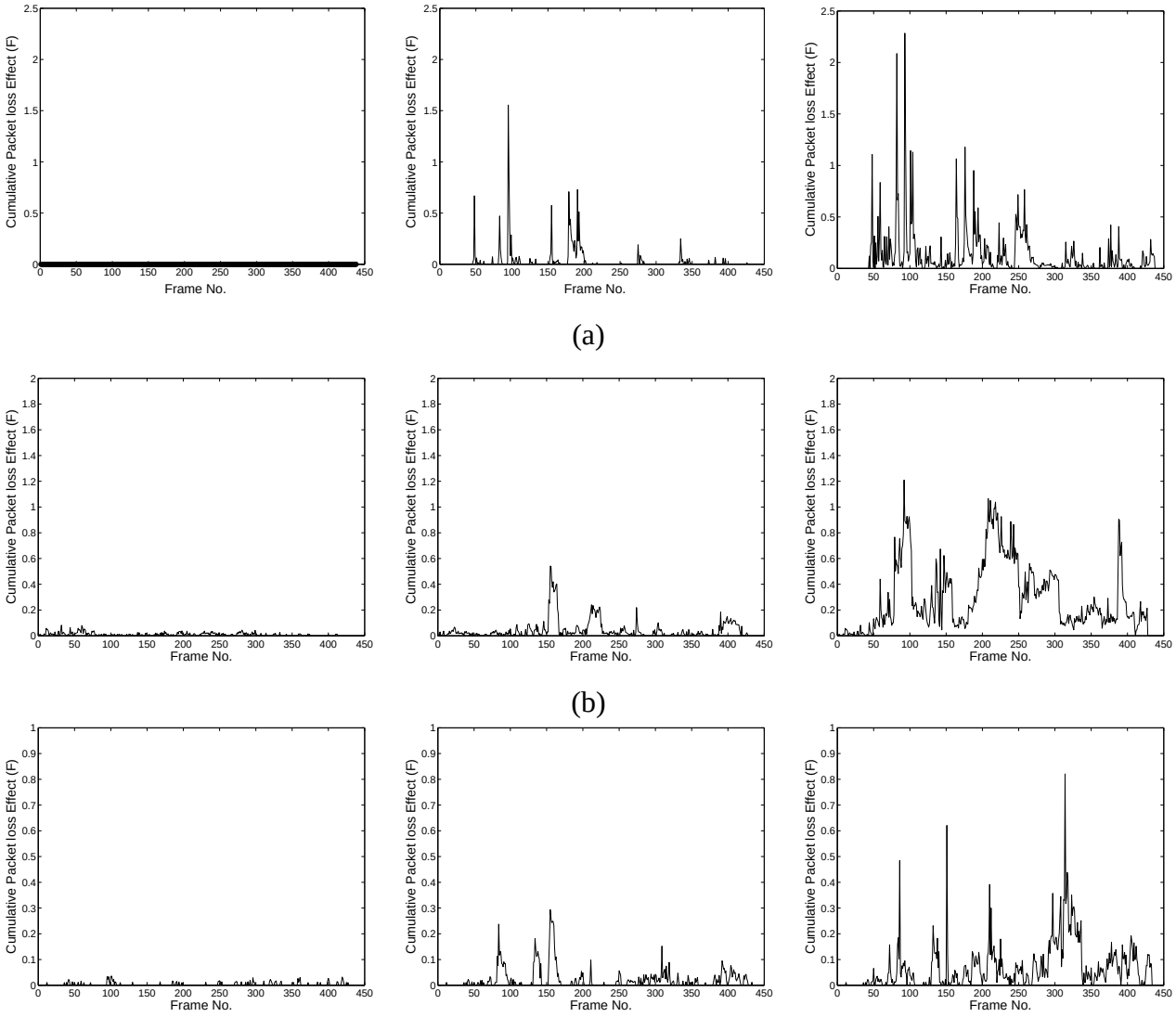



**Fig. 15**. The cumulative effect of packet loss ($F$) for original, PLR=1% and PLR=5% of (a) 'susi', (b) 'mobile-calendar' and (c) 'flower garden' sequences.

tual blur metric," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, September 22–25, 2002, vol. 3, pp. 57–60.

[11] H. R. Wu and M. Yuen, "A generalized block-edge impairment metric for video coding," *IEEE Signal Processing Letters*, vol. 70, no. 3, pp. 247–278, November 1998.

[12] Z. Wang and H. R. Sheikh and A. C. Bovik, "No-reference perceptual quality assessment of JPEG compressed images," in *Proc. ICIP'02*, September 2002, vol. 1, pp. 477–480.

[13] W. Gao, C. Mermer, and Y. Kim, "A de-blocking algorithm and a blockiness metric for highly compressed images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1150–1159, December 2002.

[14] Video Quality Experts Group (VQEG), "website: http://www.vqeg.org," .

[15] N. Feamster and H. Balakrishnan, "Packet loss recovery for streaming video.," in *International Packet Video Workshop*, April 2002.

[16] J. Boyce and R. Galianello, "Packet loss effects on MPEG video sent over public internet.," in *ACM International Multimedia Conference*, 1998, pp. 181–190.

[17] Y. Wang and Q. Zhu, "Error control and concealment for video communication: A review," *Proceedings of the IEEE*, vol. 86, no. 5, May 1998.

[18] O. Verscheure, P. Frossard, and M. Hamdi, "User-oriented QoS analysis in MPEG-2 video delivery," *Real-time Imaging*, vol. 5, pp. 305–314, 1999.

[19] J. Kimura, F. A. Tobagi, J. M Pulido, and P. J. Emstad, "Perceived quality and bandwidth characterization of layered MPEG-2 video encoding," in *SPIE Intl. Sym. on Voice, Video and Data Communications*, September 1999.

[20] S. Winkler, "A perceptual distortion metric for digital color video," in *Proc. SPIE Human Vision and Electronic Imaging*, San Jose, California, USA, January 1999, vol. 3644, pp. 175–184.

[21] Z. Wang and A. C. Bovik and B. L. Evans , "Blind measurement of blocking artifacts in images," in *Proc. ICIP'00*, September 2000, vol. 3, pp. 981–984.

[22] T. Vlachos, "Detection of blocking artifacts in compressed video," *Electronics Letters*, vol. 36, no. 13, pp. 1106–1108, 2000.

[23] S. Suthaharan, "A perceptually significant block-edge impairment metric for digital video coding," in *Proc. ICASSP'2003*, Hong Kong, 2003, vol. 3, pp. 681–684.

[24] "Website: http://www.xvid.org," .

[25] Z. Wang webpage, "http://www.cns.nyu.edu/ zwang/," .

[26] "Tektronix test sequences," ftp://ftp.tek.com/tv/test/streams/Element/index.html.

[27] "NTT DoCoMo: Error Generating Software," provided to the MPEG Resilience Ad Hoc Group Reflector on 31st October 1996.