Predicting packet-loss visibility using scene characteristics

Amy R. Reibman and David Poole
AT&T Labs – Research
{amy,poole}@research.att.com

Abstract— **We examine the influence of scene-level content on the visibility of packet loss impairments in MPEG-2 and H.264 compressed video. We consider both global camera motion and proximity to a scene cut. We use Patient Rule Induction Method (PRIM) to pick out both highly visible and very invisible packet losses. We show that global camera motion significantly increases visibility relative to a still camera. Further, while packet losses that are concealed using a prior scene's image are strongly visible, all other packet losses near a scene change are much less likely to be visible, all other factors being equal.**

I. Introduction

The growing popularity of transmitting compressed video over the Internet increases the need for quality assessment methods that can accurately characterize how the network is affecting the video quality seen by the end-user. Accurate quality assessment is essential when specifying requirements for, designing, and testing systems that transport video over networks.

Significant progress has been made on methods that assess video quality for applications other than video transmission. Image quality metrics like the Structural SIMilarity index (SSIM) [1] and those evaluated in [2] are becoming more accurate at predicting quality of individual frames of video. Full-reference (FR) video quality metrics like the Continuous Video Quality Evaluation (CVQE) metric [3] incorporate the temporal aspects of human perception. Reduced-reference (RR) metrics like [4] extract low-bandwidth information at the sender to be sent reliably to the receiver to estimate resulting video quality. No-Reference (NR) methods such as the blurring metric in [5] only use information available in the bitstream or at the decoder.

Recent work considers the visual quality produced by network impairments. A random-neural-network model is developed in [6] to assess quality given different bandwidth, frame-rate, packet loss rate, and I-block refresh rate. However, quality is evaluated only in an average sense without considering the impact of source content. Average performance across an entire video sequence is also the focus in [7], which uses MSE to assess quality for different compression standards and different concealment techniques, using a specific model for each compression standard and concealment technique. MSE of a packet loss impairment (PLI) is estimated using a NR metric in [8]. Fluidity impairments associated with temporal downsampling or freeze frames resulting from packet loss are assessed by an NR metric in [9]. Two recent papers [10], [11] present NR metrics that directly measure the length and strength of PLI from each decoded image.

Our recent work focuses on predicting the visibility of PLI for both MPEG-2 [12] and H.264 [13] separately. In these papers, we focus on exploring what can be extracted from a compressed bitstream to predict when PLI are visible to human viewers. In [14], we take a different approach and focus on exploring features of the video frames themselves: encoded signal, decoded signal, and the error between them. In addition, in [14], we obtain a generic model that predicts the visibility of PLI for two compression standards and three decoder concealment techniques without prior specification of either standard or concealment technique.

However, all our previous work focused on information that could be extracted about the video content precisely at the point of packet loss. In the current paper, we consider extracting information about the *scene* at the time of packet loss. Specifically, we consider how both camera motion and proximity to scene changes affects PLI visibility.

The impact of scene changes on visibility of impairments has been explored for video compression in the context of both blurry impairments [15] and block-based compression artifacts [16]. Seyler and Budrikis [15] explore whether spatial detail can be reduced immediately following a scene change without the viewers noticing. They show that the spatial resolution of the new scene can be substantially degraded for over half a second, provided the spatial detail is gradually increased (and does not appear suddenly). Also, in their experiments the length of time after the scene change during which spatial degradation can be invisibly injected increased when there is motion in the subsequent scene.

Tam et al. [16] explore temporal masking of compression artifacts immediately following a scene change. Their goal is to explore whether it is possible to reduce the bitrate immediately following a scene change without having humans notice the reduced quality. In their experiments, visual masking is present only in the first or second frame following the scene change.

In this paper, we extend our work in [14] in the following ways. First, we provide more detail about the generic model for predicting packet loss visibility presented in [14] where space constraints prevented more detail from being presented. Second, we use Patient Rule Induction Method (PRIM) [17] to gain a deeper understanding of

the very-visible and very-invisible PLI. Third, we show that packet loss visibility is affected by proximity to a scene change. And finally, we show that camera motion (for example, pans and zooms) alter packet loss visibility.

Section II describes the subjective test datasets we use, emphasizing the diversity of the parameters underlying the data. Section III describes the attributes of PLI and measurements that can be extracted from the encoded signal, the decoded signal, and the error between them, to predict PLI visibility. Section IV presents our model in [14] which provides the starting point for the analysis in the current paper. We present exploratory data analysis in Section V to better understand how scene-level characteristics affect PLI visibility. The exploratory data analysis provides useful insights into how specific factors affect PLI visibility. These allow us to define new factors that can improve our visibility models. Section VI presents these improved models that incorporate scene-level characteristics. Section VII concludes with a discussion of potential future improvements.

II. Subjective datasets

The primary application of this work is for high-quality video transmission over mostly reliable networks. In this application, there are few, if any, visible compression artifacts and only isolated packet-loss events. Our subjective tests are designed to measure only visibility of PLI, and do not address the more difficult question of assessing quality given rare packet loss events.

Table I summarizes the three datasets obtained with subjective testing that we use in this study. All three subjective tests are based on the same methodology: a single-stimulus test in which the viewers' task is to indicate by pressing the space-bar when they saw an artifact. Twelve viewers are used to label each packet loss in all three tests. For each of the three tests, packet losses are injected into the video such that every non-overlapping four-second interval contains a single packet loss. More detail on the methodology can be found in [12] and [13].

The viewers sat a distance of approximately six picture heights away from a CRT display in a typical office environment. No more than one viewer for each PLI is an expert viewer. Based on comments from viewers after the tests, the full-color full-motion video was sufficiently compelling that they were immersed in the viewing process rather than searching for every artifact.

Datasets 1 and 2 use video compressed by MPEG-2 at spatial resolution 720 by 480 with an adaptive GOP structure in which an I-frame is inserted at each scene change. In these videos, there are usually 2 B-frames between each reference frame, and the typical GOP length is 13 frames. However, each GOP ends with a reference frame, and so there are no B-frames between the final P-frame of one GOP and the first I-frame of the next GOP. Dataset 3 uses H.264 video at spatial resolution 352 by 240 with a fixed IBPBP-type GOP structure of 20 frames. The encoder in this case uses each I-frame as a long-term prediction frame, but does not use Flexible Macroblock Ordering (FMO). All encoders have one slice (or Network Adaptation Layer Unit (NALU)) per row of macroblocks.

The video content used in each test is highly varied in its motion and spatial texture. The signal attributes of per-frame mean, variance, mean motion-vector length, and residual energy after motion compensation are all statistically identical across the three tests. The video content in Dataset 3 is identical to half the video content in Dataset 2, while the content in Dataset 1 is unique and includes some content from film encoded at 24 fps.

We assume the decoder, in all cases, discards partially received slices. With this assumption, regardless of the packetization strategy, each packet loss is equivalent to the loss of one or more consecutive slices. In Dataset 1, while randomly injecting one packet loss in each four-second interval, we also force roughly 1/7th of all losses to be in B-frames, 1/7th in I-frames, and 5/7th in P-frames, and we also force roughly 20% of losses to cause an entire frame to be lost. In Dataset 2, we have a similar ratio of losses in I/B/P frames, and roughly 30% of losses cause an entire frame to be lost. In Dataset 3, roughly half of the losses are in B-frames and about 5% of losses in I-frames. All losses in Dataset 3 consist of a single slice.

One important difference among the tests is the decoder concealment strategy. Dataset 3 uses Motion-compensated Error Concealment (MCEC) as detailed in [13]. Dataset 2 uses zero-motion error concealment (ZMEC), while Dataset 1 uses a naive error concealment that is typical of software decoders[1]. One common feature of all the decoder error-handling strategies is that the video decoder only processes slices that are completely received. Table I shows both that significantly more viewers saw each loss in Dataset 1 and Dataset 2 than in Dataset 3, and that the initial MSE (IMSE) for those pixels whose packet was lost is very different due to the different concealment strategies.

III. Attributes of packet-loss impairments

We begin by describing attributes of packet loss impairments with the goal of understanding aspects of their features that affect their visibility. Next, we explore how these attributes can be measured in a real system.

A. Description of attributes

In general, the impairment caused by a packet loss depends on

1) the encoding algorithm and its parameters,
2) the packetization strategy,
3) how the decoder handles errors, and
4) the video content.

Rather than trying to evaluate the visible impact of the resulting impairment as a function of any one of these (for example, the frequency of I-frames or the packetization

[1]To improve speed, many software decoders merely swap pointers between forward and backward reference frames. In this "default concealment", missing macroblocks are never overwritten. So a reference frame is "concealed" using data from *two* reference frames ago, while B-frames are "concealed" using data from the most recent *B-frame*.

	Data 1 [18]		Data 2 [12]	Data 3 [13]
Spatial resolution	720x480		720x480	352x240
Frame rate (fps)	30	24	30	30
Duration (minutes)	7.3	8.9	72	36
Compression standard	MPEG-2		MPEG-2	H.264
GOP structure	I-B-B-P-		I-B-B-P-	I-B-P-
I-frame insertion	scene adaptive		scene adaptive	fixed
GOP length	≤ 13	≤ 15	≤ 13	20
concealment	default[1]		ZMEC	MCEC
Losses	108	107	1080	2160
Losses in B-frames	14%		14%	50%
Full-frame losses	20%		30%	0%
Mean num. viewers who saw each loss	4.56	5.13	3.11	1.32
Null Pred. error	0.14599		0.12236	0.041571
Initial mean sq. pix. error	5.245		3.919	1.708

TABLE I
SUMMARY OF SUBJECTIVE TEST DATASETS

strategy), we instead consider the impact of a generic packet loss impairment directly on the video content.

Let the original uncompressed video frame at time t be $f(t)$, let the compressed video frames be $\hat{f}(t)$, and let the decoded video frames be $\tilde{f}(t)$. The error is $e(t) = \hat{f}(t) - \tilde{f}(t)$. We focus here on describing the PLI according to attributes of

1) the error $e(t)$,
2) the decoded signal including error, $\tilde{f}(t)$,
3) the encoded signal (without impairment), $\hat{f}(t)$, *at the location of the PLI*, and
4) scene-level parameters (e.g. global camera motion and proximity to a scene change) at the time of the PLI.

The error caused by the impairment, $e(t)$, is completely characterized by its support and its amplitude. The error support is characterized by size, spatial pattern, duration, and location. The size is controlled by the packet size as well as the frequency of synchronization codewords like slice start codes. The FMO option of H.264 governs the spatial pattern of the error.

Error duration is governed by the frequency of I-frame or I-block information. However, error amplitude may decrease as a function of time even when no I-blocks are present due to the motion-compensation prediction process [19]. In addition, using long-term prediction in H.264 can improve error attenuation [20]. However, it is important to note that the initial amplitude of the error at the time of the loss depends more heavily on the underlying video content and the decoder concealment strategy than on the compression standard itself. Error concealment clearly affects the initial error of the macroblocks. The effectiveness of error concealment strategies depends heavily on the content itself, since some content is more easily concealed than others; however, it can be improved with a careful selection of encoding parameters as well. For example, concealment motion vectors in MPEG-2 I-frames are very helpful.

The decoded signal, $\tilde{f}(t)$, at a PLI has several attributes that are likely to affect packet-loss visibility. A lost frame is likely to introduce temporal edges and a lost slice to introduce both temporal and horizontal edges into the decoded signal. Vertical edges may also be introduced with FMO, or when the impairment propagates into subsequent frames. Moving vertical edges that are continuous in the encoded signal may also become disjointed in the decoded signal due to the impairment. All of these edge artifacts are likely to increase the visibility of the impairment.

Attributes of the encoded signal without PLI, $\hat{f}(t)$ at the location of the PLI can also affect visibility. Texture masking, luminance masking, and motion masking may each reduce visibility of the PLI. Motion tracking may enhance visibility of PLI in smoothly moving regions, yet local signal variance and motion variability may hide the PLI. In a high-quality encoding, these features of the encoded signal are essentially equal to those of the original uncompressed signal. Furthermore, these signal attributes do not depend on the compression standard.

Finally, attributes of the encoded signal *surrounding* the location of the PLI can also affect visibility. A scene change may create forward masking (which decreases visibility of PLI *after* a scene change), as well as backward masking (which decreases visibility of PLI *before* a scene change). When the scene cuts to a new still scene, and a packet is lost immediately at the scene cut, the MSE may be very large but the impairment may be invisible until several frames have passed, as pointed out in [21]. Also, camera motion may increase visibility of impairments because viewers are likely to follow, or track, consistent camera motion, which will enhance the visibility of temporal glitches.

B. Measuring attributes

To obtain an accurate quality metric for PLI, we must measure the effects described above. In this section, we describe measurements of these attributes that can be extracted with a FR, a RR, or NR video quality metric and discuss limitations of these measurements.

1) Full-Reference measurements: In this work, we focus on measuring the impact of packet loss; we are not interested in the quality degradation introduced by the encoding process. Therefore, here we consider full-reference measurements to use both the encoded signal and the decoded signal after packet loss.

MSE, a FR metric, measures the error, $\hat{f}(t) - \tilde{f}(t) = e(t)$, directly. It characterizes the error amplitude in part, but cannot quantify the spatio-temporal frequency characteristics of the error. MSE only indirectly measures attributes like error size and duration, but cannot capture any information about error location or pattern. Finally, MSE only characterizes the error between $\hat{f}(t)$ and $\tilde{f}(t)$ and nothing about the encoded signal and the decoded signal individually.

SSIM, also a FR-metric, characterizes the error, $e(t)$, to the same degree and accuracy as MSE; namely, it measures error amplitude, but not error size or duration. However, SSIM also incorporates some information about the signal at the location of the impairment. It captures statistical information about $\tilde{f}(t)$ through its mean and

variance, but does not directly measure the decoded impairment attributes (like horizontal and temporal edges) listed above.

For both MSE and SSIM, the averaging (or pooling) interval is an important consideration. In this paper, we consider MSE-1 and SSIM-1, which are averages across all pixels in a one-second interval that contains the PLI. We also consider macroblock (MB) pooling below.

2) Reduced-Reference measurements: For RR measurements, we envision a video encoder or video server reliably providing per-MB information about $e(t)$ and $\hat{f}(t)$ for video quality assessment and having the video quality monitor compute similar information about $\tilde{f}(t)$. The measurements of $e(t)$ at the encoder or server assume knowledge of the decoder concealment strategy and may have reduced accuracy when using motion-compensated error concealment (MCEC) since its estimate of the missing motion depends on the received data.

While in theory, MSE-1 and SSIM-1 can be computed in a similar manner at the server and sent reliably to a monitoring location, this is prohibitive as it would be necessary to run a complete decoding for each possible loss. Therefore, we consider it practical only to classify *initial* MSE (or IMSE) and *initial* SSIM (or ISSIM) as RR metrics.

In this paper, we consider two ways to pool the initial MSE and initial SSIM. The first is IMSE (or ISSIM), the MSE (or SSIM) averaged over the entire frame that is initially impacted by the loss. This definition was also considered in [22]. Another pooling strategy for the initial MSE or initial SSIM is to consider extrema over a small spatial window; this can be effective because there is evidence that our attention may be drawn to "worst-case" errors. Hence, we consider here Max-IMSE, defined as the maximum per-MB MSE over all MBs in the initial impairment, and Min-ISSIM, defined as the minimum per-MB SSIM over all MBs in the initial impairment. Max-IMSE was shown to be a useful quantity in [13]. An equation to compute a per-MB *initial* SSIM in an RR framework is presented in [14] using the local means and variances of the encoded and decoded signal, as well as their MSE.

We also consider the following RR signal descriptors: SigMean, SigVar, MotionMean, MotionVar, and ResidEng (the residual energy after motion compensation). MotionMean, MotionVar, and ResidEng were also considered in [12]. Because our goal is a standards-independent PLI metric, we measure these directly from the underlying signal, independent of the compression algorithm, using 16x16 motion blocks. We store these using one value each for a complete row of MBs. One additional signal measure we do *not* consider here is AvgInterParts, which was shown to be useful in [13]. AvgInterParts indicates when the video content cannot be represented accurately by a single translational motion vector.

3) No-Reference measurements: NR measurements may be based on pixels only (NR-P), the lossy bitstream only (NR-B), or both bitstream and pixels (NR-BP). A NR-B method can use the lossy bitstream to exactly measure error size, pattern, location, and duration. However, a NR-P method, with access only to the decoded pixels, can exactly measure information about the decoded signal $\tilde{f}(t)$ and can only estimate information about the error $e(t)$. Estimates of the signal $\hat{f}(t)$ at the location of the impairment can be estimated using either NR-P and NR-B using information from neighboring unimpaired frames.

As in our past work, we consider here the NR-B measurements of initial spatial extent, temporal duration, and location. We also use the NR-BP Slice-Boundary Mismatch (SBM) measure presented in [14] which is a modification of a similar NR-P metric presented in [11].

The NR-PB SBM measure consists of a NR-B detection process and a NR-P estimation process. The NR-B detection process exactly pinpoints impairment location in the decoded frame. Our NR-P estimation process differs from that in [11] in minor details only. We define the SBM metric based on the impact of the impairment on slice boundaries, and apply the metric only on the boundaries between slices that contain non-zero errors.

$$\mathrm{SBM} = \frac{1}{r-1}\sum_{i=1}^{r-1} \mathrm{SBM}(i) * I(i)$$

where there are r rows of MBs, $\mathrm{SBM}(i)$ is the Slice-Boundary Mismatch along the i-th MB boundary, and $I(i)$ indicates we detected an impairment in one of the slices on either side of the boundary. Let $S_u(i)$ and $S_d(i)$ be the sum of absolute row difference up above and down below the MB row-boundary i, respectively, and let $S_m(i)$ be their mean. Then,

$$\mathrm{SBM}(i) = \max\{(S_b(i) - S_m(i))/S_m(i), 0\}$$

when both $S_b(i) > T$ and $S_m(i) > T$, where $S_b(i)$ is the summed absolute row difference across MB row-boundary i, and T is a noise threshold set to 10 here. In our studies below, we consider Max-SBM, the maximum value across all impaired frames.

Finally, we extract information about scene-level parameters from the video signal $\hat{f}(t)$. Many techniques exist to detect scene boundaries, including those in [23] and [24]. We are interested here primarily in scene cuts, in which the scene changes quickly in one frame. We label each PLI by the distance in time between the first frame affected by th ePLI and the nearest scene cut, either before or after. We also consider scene or camera motion, which can again be extracted using a number of techniques, including those in [25]. In this paper, we classify scenes based on four camera-motion types: still, panning, zooming, or complex camera motions.

Table II summarizes the factors.

IV. Summary of PLI visibility model from [14]

We use logistic regression [26] to predict packet loss visibility, as in [12], [13]. Logistic regression is a special case of regression using a Generalized Linear Model (GLM) where the link function is the logit function. Our

Error (FR): 1-second error pooling	MSE-1, SSIM-1
Error (RR): Per-frame pooling of initial error	IMSE, ISSIM
Error (RR): Max-over-macroblock pooling of initial error	MaxIMSE, MinISSIM
Error (NR)	InitialSpatialExtent, Duration, Height
Signal (RR)	SigMean, SigVar, MotionMean, MotionVar, ResidEng
Signal + error (error across all frames)	Max-SBM
Signal + error (initial error only)	ISBM
Scene	DistFromCut, CameraMotion

TABLE II
FACTORS FOR PREDICTING VISIBILITY FROM ERROR, SIGNAL, SIGNAL+ERROR, AND SCENE

Model 1: 1-sec pooling	Model 2: Init. err.	Model 3: Max-MB error
$\log(\text{MSE1})$	$\log(\text{IMSE})$	$\log(\text{MaxIMSE})$
$\log(1-\text{SSIM1})$	$\log(1-\text{ISSIM})$	$\log(\text{MinISSIM})$
$\log(\text{MaxSBM})$	–	–
Duration	(Duration<0.05)	(Duration<0.05)
InitialSpatialExtent	InitialSpatialExtent	—
$\lvert\text{SigMean}-128\rvert$	$\lvert\text{SigMean}-128\rvert$	$\lvert\text{SigMean}-128\rvert$
$\log(\text{SigVar})$	$\log(\text{SigVar})$	$\log(\text{SigVar})$
$\text{MotMean} > 1/\sqrt{2}$	—	$\text{MotMean} > 1/\sqrt{2}$
MotionVar	—	—
$\log(\text{ResidEng})$	$\log(\text{ResidEng})$	$\log(\text{ResidEng})$

TABLE III
FACTORS IN EACH MODEL IN [14]

goal is to estimate p_i, the fraction of viewers who saw error i.

One important challenge of fitting models using the datasets shown in Table I is that the distribution of samples across a given factor may not be representative of what is expected in practice. The most obvious example of this is that the Dataset with the default concealment (Dataset 1) is much smaller than the Dataset with motion-compensated concealment (Dataset 3). If this lack of balance is not taken into consideration, then our model may be biased toward the dataset with more data. To overcome this challenge, we define the following procedure to fit, test, and evaluate the resulting models.

We would like to fit models using an equal number of samples from each of the datasets, and then use cross-validation to evaluate the goodness of fit and select the best model. Cross-validation [27] is commonly used for model evaluation and to prevent over-fitting when data are sparse. A model is trained on a fraction of the data and then evaluated using the remaining data points (test set). This is repeated using different non-overlapping training and test partitions of the data and the overall test error is computed. In this case we select our training and test sets based on the fact that we should achieve representative sampling from Dataset 1, which has the fewest samples. We choose four-fold cross-validation on Dataset 1.

Specifically, to form a first test set, we randomly choose 159 samples from each Dataset to fit a first model. We create a testing set containing the remaining 56 samples from Dataset 1, the remaining 921 samples from Dataset 2, and the remaining 2001 samples from Dataset 3. We evaluate the performance of the first fitted model by computing

$$\frac{1}{3}\sum_{k=1}^{3}\left[\frac{1}{N_k}\sum_{i \text{ in Dataset } k}(p_i-\tilde{p}_i)^2)\right], \qquad (1)$$

where $\tilde{p}_i$ is the predicted fraction of viewers who saw error i and N_k is the number of samples in Dataset k. We then repeat the fitting process for another non-overlapping set of training data from each Dataset, for a total of four fitted models. We average the performance from (1) across the four fitted models. We then repeat this procedure for a total of four different random seeds. We define the average performance for each of these sixteen models as Q. In what follows, we will refer to this procedure as our validation procedure.

We include a specific factor in our model only if the model with that factor included has smaller Q than the model without that factor. In [14], we present three classes of models according to the type of pooling used for measuring the error: one-second pooling across all frames affected by the PLI, per-frame averaging across the initial frame affected by the PLI, and maximum-over-macroblock pooling across the initial frame affected by the PLI. We also explore some nonlinear mappings of each factor to see if improved fitting performance is possible. Table III summarizes the resulting models for each class.

V. EXPLORATORY ANALYSIS OF SCENE-LEVEL PARAMETERS

In this section, we explore the scene-level parameters in the combined Datasets 1–3. We begin with exploring the scene cuts, and then explore the camera motion. Exploratory data analysis (EDA) is an important step in statistical analysis [28], as the process facilitates designing appropriate models.

In the video shown to viewers, there were 704 scene cuts, with an average time between scene cuts of 221 frames. Of the 3455 losses in our datasets, 935 are within one second of a scene cut, either before or after. Figure 1 shows the histogram of the number of losses as a function of the distance from the loss to the closest scene cut, for those losses within 4 seconds of the closest scene cut. With such a large fraction of losses near a scene change, it is important to understand its impact.

Figure 2 shows the boxplot of the number of viewers who saw each loss, as a function of the distance of the loss from the closest scene cut. The solid red line indicates the average number of viewers who saw an error. The boxplot is configured such that the x-axis is partitioned into large intervals at either end and single-frame intervals in the middle. This plot indicates that relative to the average

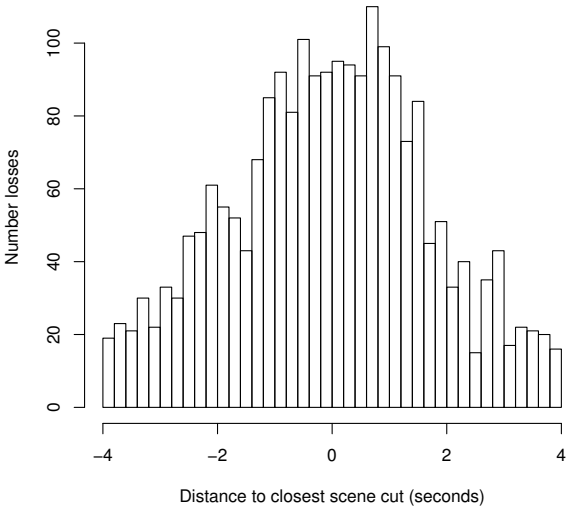

Fig. 1. Distribution of losses within 4 seconds of a scene cut

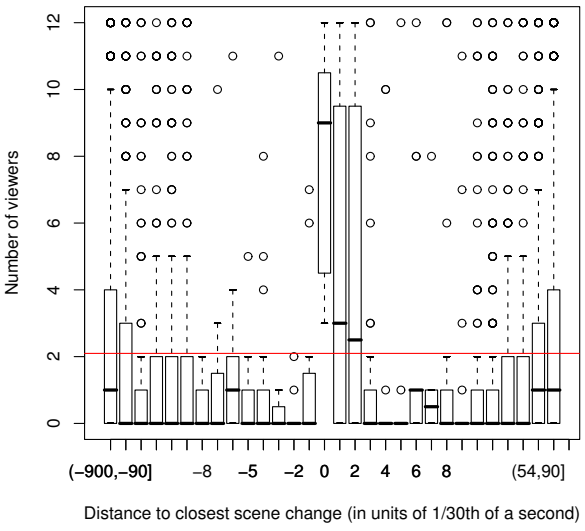

Fig. 2. Number of viewers as a function of proximity from scene cut

number of viewers across all losses, many more viewers saw impairments within one- or two-thirtieths of a second after a scene change, while relatively fewer viewers saw the losses up to one-third of a second before or after a scene change.

As a next step, we examine the prediction error of our Model 2 from [14], and plot it as a function of scene-cut distance in Figure 3. The y-axis is the difference between the actual and predicted number of viewers who saw a PLI; each line represents the results for a different one of the four test sets given one random seed in our validation procedure. The x-axis here uses the same partitions as in Figure 2. As can be seen, the model predicts fewer people will see errors immediately at the scene cut than actually do. Also, it predicts that more people will see errors in

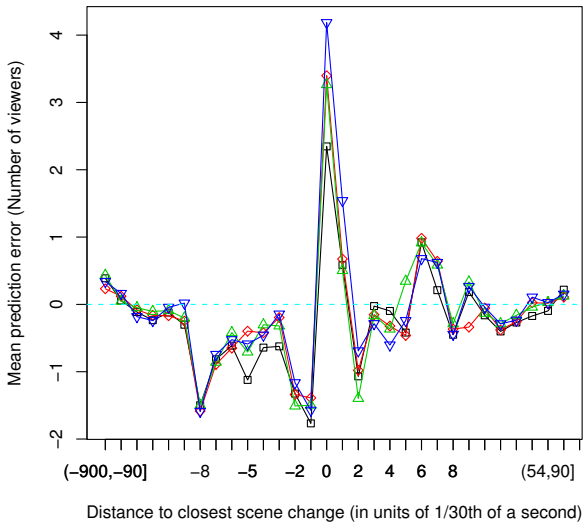

Fig. 3. Prediction error as a function of proximity to scene cut; model 2 from [14]

the one-third of a second before a scene cut than actually do, and that more people will see errors in the one-sixth of a second after a scene cut than actually do.

Further exploration indicates that the enhanced visibility immediately at the scene change occurs when error concealment takes place across the scene boundary. We define a new variable, **DistToRef**, which describes the distance between the current frame (with the packet loss) and the reference frame used for concealment. We also define a Boolean variable, **AtScene**, which is TRUE if $0 \leq$DistFromCut$<$DistToRef.

To account for the depressed visibility immediately *before* the scene cut, we define a Boolean variable **BeforeScene**, which is TRUE if $-0.4sec <$ DistFromCut $< 0sec$. Depressed visibility *after* a scene cut requires that the PLI not only appear close to the scene cut, but also *disappear* near the scene cut. Therefore, to account for the depressed visibility immediately after a scene cut, we define the Boolean variable **AfterScene**, which is TRUE if $0sec <$ (DistFromCut + Duration) $< 0.25sec$ and AtScene is not true.

Next, we examine the influence of the camera motion on visibility. We consider four types of camera motion: still, panning, zooming, and "complex" camera motion. Table IV indicates the distribution of camera motion both in the complete video shown to viewers, as well as the fraction of losses which occurred in each type of camera motion. Significantly fewer viewers saw PLI in still scenes than in panning or zooming scenes. PLI in scenes with complex camera motion have visibility somewhere between these.

A. PRIM

Next, we experiment with PRIM [17], an information mining algorithm whose goal is to find "bumps", which are sub-regions of the multi-dimensional space of factors

Camera type	% frames	# Losses	% losses	Mean viewers
Still	63.7	2380	68.9	1.31
Panning	23.6	814	23.6	3.95
Zooming	6.7	169	4.9	3.99
Complex	1.8	92	2.7	2.62

TABLE IV
DISTRIBUTION OF CAMERA MOTION IN ORIGINAL CONTENT AND IN LOSSES

where the average response (number of viewers who saw the error) is significantly larger than its average across the entire dataset. PRIM essentially focuses only on characterizing these high-response regions, and does not attempt to characterize mid-range responses. By applying PRIM to the inverse of the response (namely, the number of viewers who did *not* see an error), we can also characterize low-response regions. One advantage of PRIM is that it can find "bumps" in datasets that have a large number of factors and are highly non-linear and possibly non-monotonic. PRIM is also especially suited to find interactions among variables.

The output of the PRIM algorithm is a series of "boxes" that select which factors, and their corresponding range of values, will lead to expected high response. Our goal in applying PRIM to our subjective data is to (a) gain a better understanding of the nonlinearities present in the data, and (b) find quick and easy ways to classify a packet loss as leading to a clearly visible or clearly invisible impairment.

We apply PRIM to the same sets of factors in the three models presented in Table III, in addition to the scene-related factors described above, omitting the SSIM factors. (The SSIM factors are highly correlated with the MSE factors; the PRIM boxes are more interpretable when only one of these factors is included in the analysis.) We report the results only for Model 2, although similar conclusions apply for the other models as well. The resulting "boxes" and the resulting mean number of viewers and percent of losses included in each box are shown in Tables V and VI for high responses and low responses, respectively. These boxes are sequential, in that the data in the first box are removed from the dataset prior to fitting the second box, etc. Because PRIM does not define a model as such, we present only the first few boxes as they provide the most insight and understanding.

A quick glance at the table yields the following observations. First, the error descriptor IMSE appears in all boxes except the first maximum. Given the long-running success of MSE for analyzing video compression and transport, this is not surprising. Second, the scene variables defined above play a large role in describing high and low responses. Camera motion is present in all boxes but the first and third maximum boxes; high response occurs when the camera motion is panning or zooming, and low response occurs when the camera motion is still or (sometimes) complex. Also, losses immediately at a scene cut are picked out as being highly visible, and losses

Box	Description	Box mean (Num. viewers)	Box size
1	AtScene=TRUE	8.92	0.75 %
2	ResidEng$\leq$2198870 0.998 $\leq$ MotionMean $\leq$ 13.303 CameraMotion ! = Still DistToRef $\geq$ 2 frames IMSE$\geq$ 6.935	8.33	8.16 %
3	SigMean $\geq$ 55.46 SigVar $\leq$ 709.70 MotionVar $\geq$ 1.93 4 $\leq$ Height $\leq$ 19.5 SceneAfter = FALSE SceneBefore = FALSE DistToRef $\geq$ 3 frames IMSE$\geq$ 5.23	6.17	6.45 %

TABLE V
PRIM BOXES (MAXIMUM, HIGH VISIBILITY). MODEL 2 PARAMETERS.

Box	Description	Box mean (Num. viewers)	Box size
1	ResidEng$\leq$503101 CameraMotion = Still DistToRef $\geq$ 3 frames IMSE$\leq$ 0.718	0.13	6.60 %
2	Duration$\leq$0.6333 sec MotionMean $\geq$ 0.06 CameraMotion = Still, Complex DistToRef = {1,2,6} frames IMSE$\leq$ 1.932	0.45	32.8 %
3	CameraMotion = Still, Complex SceneAt = FALSE DistToRef = {1,3} frames IMSE$\leq$ 10.85	0.80	12.9 %

TABLE VI
PRIM BOXES (MINIMUM, LOW VISIBILITY). MODEL 2 PARAMETERS.

shortly before or shortly after a scene cut are picked out as being nearly invisible.

An additional factor that appears several times in the PRIM results is DistToRef, the parameter defined above to characterize how far away the frame used for concealment is from the frame in which the error occurred. Recall that this is an important difference among the different datasets due to their different concealment strategies. One interesting aspect is that both high and low responses occur for larger values of DistToRef. For example, both the third maximum box and the first minimum box hold when DistToRef $\geq$ 3 frames.

This interaction can be better understood by examining Figure 4. Here, we plot the mean number of viewers as a function of both DistToRef (x-axis) and IMSE. Different curves on the plot correspond to the different quartiles of IMSE. We can see that when IMSE is below the median (lowest two quartiles) DistToRef plays very little role in predicting the PLI visibility. However, when IMSE is above the median, increasing DistToRef increases the PLI visibility.

Pursuing the relationship between visibility, IMSE, and DistToRef further, we also consider the impact of Dist-

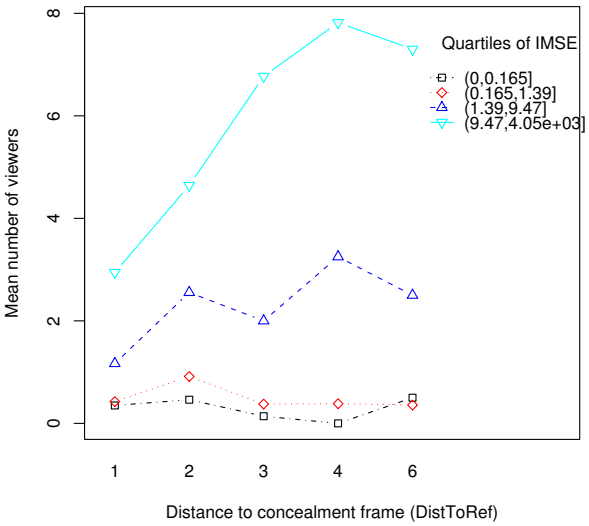

Fig. 4. Number of viewers as a function of DistToRef for the quartiles of IMSE.

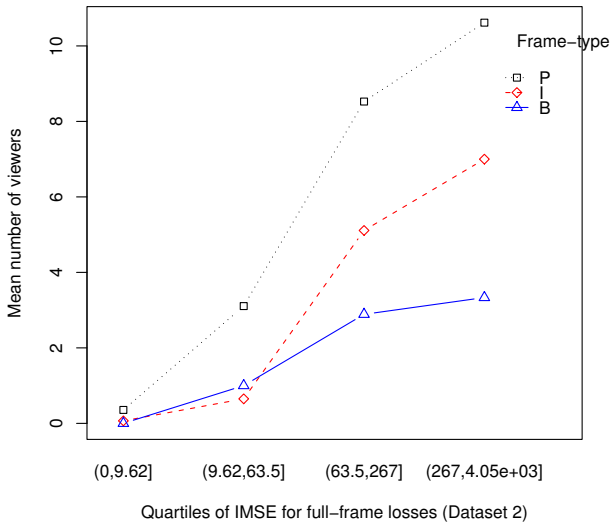

Fig. 5. Number of viewers as a function of IMSE for full-frame losses in B-, P- and I-frames in Dataset 2.

ToRef by examining Dataset 2 specifically. Based on the GOP structure of the video in that Dataset, P-frames are concealed using images three frames ago (DistToRef=3), while both I-frames and B-frames are concealed using images only one frame ago (DistToRef=1). Thus, it is useful to examine the visibility of losses that affect an entire I-, P-, and B-frame in Dataset 2 to better understand the role of Dist2Ref.

Figure 5 shows the number of viewers as a function of the quartiles of IMSE for these losses, decomposed into B-, P- and I-frames. In this figure, we see that for the same IMSE, visibility is substantially reduced when the concealment image is closer to the current image. This is not surprising for B-frames; however, interestingly, this conclusion also holds for I-frames, even though impairments in I-frames last longer on average than those in P-frames. This reduced visibility occurs because there are fewer temporal artifacts when the concealment image is temporally closer to the current image, particularly in areas with motion. Therefore, DistToRef appears to be an important factor to consider when modeling packet loss visibility.

VI. Logistic Regression Results Using New Variables

In [14], we present a logistic regression model that predicts packet loss visibility using both error characteristics (Duration, InitialSpatialExtent, MSE, SSIM, and SBM) as well as signal characteristics (SigMean, SigVar, MotMean, MotVar, ResidEng). These factors only characterize what is happening in the image at the exact location of the error. Here, we extend this model to include the new factors found to be important in our exploratory data analysis above. These factors are summarized in Table VII. With the exception of the first (DistToRef), these new factors are all Boolean variables.

DistToRef	Distance to concealment reference (frames)
AtScene	$0 \leq \text{DistFromCut} < \text{DistToRef}$
SceneBefore	$-0.4sec < \text{DistFromCut} < 0sec$
SceneAfter	$0sec < (\text{DistFromCut} + \text{Duration}) < 0.2sec$
StillCamera	CameraMotion = Still
FarConceal	$\text{DistToRef} \geq 3$

TABLE VII
New Factors for Predicting Visibility

We use the same strategy as outlined in Section IV above, where we tentatively add each new factor individually and decide to include it in our model only if the average prediction error Q decreases, where this average is across 4 starting random seeds for each of the four training and test set partitions in our validation procedure. The resulting average prediction error Q is shown in Table VIII as each new factor is added to the model sequentially. As can be seen, each new factor continues to improve the model. (However, AfterScene is not incorporated into the model. Even though it was present in one of the PRIM boxes, incorporating it into the logistic regression model never yielded a statistically significant improvement.) While the error, signal, and signal+error factors are most important to the models, adding the new factors decreases the average prediction error for both Models 1 and 2 by over 10%. Also, even though the final factor, FarConceal does not help Model 3, adding the other factors to Model 3 improves it by almost 13%.

In addition, for Model 2, we show in Figure 6 the average performance within each of the Datasets in Table I. As can be seen, the new factors improve the fit for Dataset 1 most substantially, and adding the final FarConceal factor actually degrades performance on Dataset 3. However,

	Model 1: 1-sec pooling	Model 2: Init. err.	Model 3: Max-MB err.
Error/SSIM/SBM	0.04824	0.05152	0.05286
Plus signal factors	0.03924	0.04155	0.04652
Plus SceneCut factors	0.03733	0.03900	0.04369
Plus StillCamera	0.03647	0.03775	0.04052
Plus FarConceal	0.03522	0.03693	—

TABLE VIII

AVERAGE RESIDUAL ERROR Q ACROSS ALL 3 DATASETS AS NEW FACTORS ARE ADDED TO MODELS

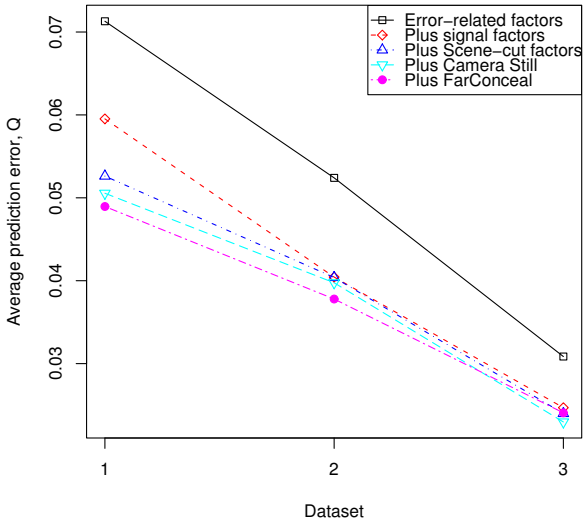

Fig. 6. Residual error within Datasets for models with IMSE and ISSIM.

overall, incorporating this factor into the model helps the overall performance across all three Datasets.

Table IX presents the factors included in our final models, and the associated coefficients for one selected model designed during our cross-validation. (The coefficients for the other models during cross-validation are similar.) These can be used in the logistic regression model,

$$\log\left(\frac{p_i}{1-p_i}\right) = \gamma + \sum_j \beta_j x_{ji}$$

where p_i is the probability of visibility for the ith packet loss, γ is the intercept term and β_j is the coefficient for factor x_{ji}, the jth factor associated with the ith loss.

VII. CONCLUSIONS

In this paper, we extend our generic packet loss visibility model with new factors that account for scene-level characteristics. We integrate both camera motion and proximity to scene cuts. We show that packet losses in scenes with a still camera are more likely to be invisible. Packet losses immediately at a scene change are significantly more visible, however packet losses shortly before or shortly after a scene change are less visible. In addition, our analysis using PRIM also shows interesting interactions between MSE and the distance between the current frame and the frame used for concealment. For the same MSE, PLI are less visible when they have been concealed from a temporally-closer reference frame, since this causes smaller temporal artifacts. We were able to incorporate all these findings (except the finding on reduced visibility after a scene cut) into our PLI visibility model to improve its prediction accuracy by over 10%.

Factor	Coeffs. for Model 1: 1-sec pooling	Coeff. for Model 2: Init. err.	Coeff. for Model 3: Max-MBerr
Intercept	9.21880	7.933943	2.596257
log(MSE1)	0.501144	—	—
log(IMSE)	—	0.651037	—
log(MaxIMSE)	—	—	0.504702
log(1 − SSIM1)	0.207579	—	—
log(1 − ISSIM)	—	0.187520	—
log(MinISSIM)	—	—	0.391291
log(MaxSBM)	0.182920	—	—
Duration	-2.63622	—	—
(Duration< 0.05)	—	-0.613950	-0.811756
InitialSpatialExtent=2	-0.30792	-0.208028	—
InitialSpatialExtent=30	-2.01809	-1.748874	—
|SigMean − 128|	-0.007474	-0.005021	-0.005391
log(SigVar)	-0.38353	-0.159266	-0.226153
MotMean > $1/\sqrt{2}$	0.168632	—	0.096248
MotionVar	-0.003649	—	—
log(ResidEng)	-0.406089	-0.664350	-0.403958
AtScene	0.861301	1.157734	1.578054
BeforeScene	-1.19976	-1.474047	-1.566542
StillCamera	-0.597119	-0.778593	-0.963693
log(MSE1)*FarConceal	0.277193	—	—
log(IMSE)*FarConceal	—	0.160639	—

TABLE IX

COEFFICIENTS FOR ONE MODEL DURING CROSS-VALIDATION

The results of our study show that a scene cut following a PLI suppresses its visibility up to about 0.35 seconds, and a scene cut immediately before a PLI suppresses its visibility slightly for a few frames. Previous studies [15], [16] regarding the impact of scene changes on video compression artifacts came to different conclusions. First, we are the only ones to study or report non-trivial backward masking, where the scene change hides what came before it. Second, Seyler and Budrikis [15] show significant forward masking to about 0.78 seconds, whereas our forward masking is less than 0.25 seconds. In contrast, Tam et al. [16] show forward masking only within the first frame or two immediately following the scene cut.

We postulate that the difference among these findings is due, at least in part, to the different types of impairments injected. Seyler and Budrikis [15] examine blurriness that decreases slowly. Tam et al. [16] examine high-frequency compression artifacts, and we examine PLI. Also, Tam uses the much more strenuous 2-alternative-forced-choice detection criterion, whereas both Seyler and we consider the less stringent question: "Did you see an impairment?".

Our results indicate a noticeable decrease in visibility immediately prior to a scene change, for at least one-third of a second. This is well beyond what is predicted from studies on backward temporal masking [29]. However, the backward masking that we see prior to a scene cut may not be due to the typical integration masking that

occurs in the early stages of visual processing. Instead, it may be an example of interruption masking [30]. Interruption masking does not involve the early stages of visual processing but takes place during higher-level processing, for example during object recognition. The mask (i.e., scene cut) interrupts the amount of time that can be spent processing the target (i.e., the packet loss impairment).

Clearly, more study is necessary for these issues to be better understood.

ACKNOWLEDGEMENTS

The authors would like to thank S. Kanumuri and P. Cosman for the subjective test results in [13].

REFERENCES

[1] Z. Wang et al., "Image Quality Assessment: From Error Visibility to Structural Similarity", *IEEE Trans. Im. Proc.*, vol. 13, Apr. 2004.

[2] H. R. Sheikh et al., "A Statistical Evaluation of Recent Full Reference Image Quality Assessment Algorithms", *IEEE Trans. Image Proc.*, vol. 15, no. 11, pp. 3440-3451, Nov. 2006.

[3] M. A. Masry, S. S. Hemami, "A Metric for Continuous Quality Evaluation of Compressed Video With Severe Distortions," *Signal Proc.: Image Comm.*, vol. 19, no. 2, Feb. 2004.

[4] S. Wolf and M. H. Pinson, "Low bandwidth reduced reference video quality monitoring system", *First Int'l Workshop on Video Proc. and Quality Metrics*, Jan 2005.

[5] P. Marziliano et al, "Perceptual blur and ringing metrics: Application to JPEG2000", *Sig. Proc.: Image Comm.*, pp. 163–172, Feb. 2004.

[6] S. Mohamed and G. Rubino, "A study of real-time packet video quality using random neural networks", *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 12, no. 12, pp. 1071-1083, Dec. 2002.

[7] S. Tao et al., "RealTime Monitoring of Video Quality in IP Networks", *Proceedings NOSSDAV'05*, pp. 129–134, June 2005.

[8] A. R. Reibman et al. "Quality monitoring of video over a packet network", *IEEE Trans. Multimedia*, vol. 6, pp. 327-334, April 2004.

[9] R. R. Pastrana-Vidal and J.-C. Gicquel, "Automatic quality assessment of video fluidity impairments using a no-reference metric", *Int'l Workshop on Video Proc. and Quality Metrics*, Jan 2006.

[10] R. V. Babu et al., "No-Reference metrics for video streaming applications", *International Workshop on Packet Video*, Dec 2004.

[11] H. Rui et al., "Evaluation of packet loss impairment on streaming video", *J. of Zhejiang University SCIENCE*, vol. 7, April 2006.

[12] S. Kanumuri et al., "Modeling packet-loss visibility in MPEG-2 Video", *IEEE Trans. Multimedia*, April 2006.

[13] S. Kanumuri et al., "Packet-loss visibility in H.264 videos using a reduced reference method", *IEEE Int. Conf. Image Proc.*, Oct. 2006.

[14] A. R. Reibman and D. Poole, "Characterizing packet loss impairments in compressed video", *IEEE Int. Conf. Image Proc.*, Sept. 2007.

[15] A. J. Seyler and Z. L. Budrikis, "Detail Perception after Scene Changes in Television Image Presentations", *IEEE Transactions on Information Theory*, vol. 11, no. 1, pp. 31–43, Jan. 1965.

[16] W. J. Tam et al. "Visual masking at video scene cuts", Proceedings of SPIE Human Vision, Visual Processing, and Digital Display VI, vol. 2411, pp. 111–119, Apr. 1995.

[17] J. H. Friedman and N. I. Fisher, "Bump hunting in high-dimensional data", *Statistics and Computing*, vol. 9, pp. 123143, 1999.

[18] Y. Sermadevi and A. R. Reibman, Unpublished subjective test results, Sept. 2002.

[19] K. Stühlmuller et. al. "Analysis of video transmission over lossy channels", *IEEE J. on Selected Areas in Comm.*, vol. 18, no. 6, pp. 1012-1032, June 2000.

[20] M. Budagavi and J. D. Gibson, "Multiframe video coding for improved performance over wireless channels", *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 252–265, February 2001.

[21] O. Nemethova et al., "PSNR-Based Estimation of Subjective Time-Variant Video Quality for Mobiles", Proc. of the MESAQUIN.

[22] A. R. Reibman et al., "Visibility of Individual Packet Losses in MPEG-2 Video, *IEEE Int. Conf. Image Proc.*, Oct. 2004.

[23] A. Hanjalic, *Content-based analysis of digital video*, Kluwer Academic Publishers, Boston, 2004.

[24] Z. Liu, et al., "AT&T Research at TRECVID 2006", TRECVID 2006.

[25] Y.-P. Tan et al., "Rapid estimation of camera motion from compressed video withapplication to video annotation", *IEEE Trans. Circuits and Syst. for Video Tech.,*, vol. 10, no. 1, pp. 133–146, Feb. 2000.

[26] D. W. Hosmer and S. Lemeshow, *Applied Logistic Regression*, 2nd ed., Wiley-Interscience.

[27] T. Hastie et al., *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer-Verlag, New York, 2001.

[28] J. W. Tukey, *Exploratory data analysis*, Addison-Wesley, Boston, MA, 1977.

[29] B. Girod, "The information theoretical significance of spatial and temporal masking in video signals.", *Proc. SPIE*, vol. 1077, pp. 178-187, 1989.

[30] J. T. Enns and V. Di Lollo, "Whats new in visual masking?", *Trends in Cognitive Sciences*, vol. 4, no. 9, pp. 345–352, Sept. 2000.