Modeling the Time—Varying Subjective Quality of HTTP Video Streams With Rate Adaptations

Chao Chen, *Student Member, IEEE*, Lark Kwon Choi, *Student Member, IEEE*, Gustavo de Veciana, *Fellow, IEEE*, Constantine Caramanis, *Member, IEEE*, Robert W. Heath Jr., *Fellow, IEEE*, and Alan C. Bovik, *Fellow, IEEE*

***Abstract*— Newly developed hypertext transfer protocol (HTTP)-based video streaming technologies enable flexible rate-adaptation under varying channel conditions. Accurately predicting the users' quality of experience (QoE) for rate-adaptive HTTP video streams is thus critical to achieve efficiency. An important aspect of understanding and modeling QoE is predicting the up-to-the-moment subjective quality of a video as it is played, which is difficult due to hysteresis effects and non-linearities in human behavioral responses. This paper presents a Hammerstein–Wiener model for predicting the time-varying subjective quality (TVSQ) of rate-adaptive videos. To collect data for model parameterization and validation, a database of longer duration videos with time-varying distortions was built and the TVSQs of the videos were measured in a large-scale subjective study. The proposed method is able to reliably predict the TVSQ of rate adaptive videos. Since the Hammerstein–Wiener model has a very simple structure, the proposed method is suitable for online TVSQ prediction in HTTP-based streaming.**

***Index Terms*— QoE, HTTP-based streaming, time-varying subjective quality.**

I. Introduction

BECAUSE the Hypertext Transfer Protocol (HTTP) is firewall-friendly, HTTP-based adaptive bitrate video streaming has become a popular alternative to its Real-Time Transport Protocol (RTP)-based counterparts. Indeed, companies such as Apple, Microsoft and Adobe have developed HTTP-based video streaming protocols [1]–[3], and the Moving Picture Experts Group (MPEG) has issued an international standard for HTTP based video streaming, called Dynamic Adaptive Streaming over HTTP (DASH) [4].

Another important motivation for HTTP-based adaptive bitrate video streaming is to reduce the risk of playback interruptions caused by channel throughput fluctuations. When a video is being transmitted, the received video data are first buffered at the receiver and then played out to the viewer. Since the channel throughput generally varies over time, the amount of buffered video decreases when the channel throughput falls below the video data rate. Once all the video data buffered at the receiver has been played out, the playback process stalls, significantly impacting the viewer's Quality of Experience (QoE) [5], [6]. In HTTP-based rate-adaptive streaming protocols, videos are encoded into multiple representations at different bitrates. Each representation is then partitioned into segments of lengths that are several seconds long. At any moment, the client can dynamically select a segment from an appropriate representation to download, in order to adapt the downloading bitrate to its channel capacity. Although HTTP-based streaming protocols can effectively reduce the risk of playback interruptions, designing rate-adaptation methods that could optimize end-users' QoE is difficult since the relationship between the served bitrate and the users' viewing experience is not well understood. In particular, when the video bitrate is changed, the served video quality may also vary. If the impact of quality variations on QoE is not accurately predicted, the rate adaptation method will not provide the optimal QoE for the users.

One important indicator of QoE is the *time-varying subjective quality* (TVSQ) of the viewed videos. Assuming playback interruptions are avoided, the TVSQ is *a continuous-time record of viewers' judgments of the quality of the video as it is being played and viewed*. The TVSQ depends on many elements of the video including spatial distortions and temporal artifacts [7], [8]. What's more, human viewers exhibit a hysteresis [9] or recency [10] "after effect", whereby the TVSQ of a video at a particular moment depends on the viewing experience before the moment. The quantitative nature of this dependency is critical for efficient rate adaptation. For example, as observed in our subjective study (see Section II-E for more detail), a viewer suffering a previous unpleasant viewing experience tends to penalize the perceived quality in the future. One approach to combat this is to force the rate controller to provide higher video quality in the future to counterbalance the negative impact of a prior poor viewing experience. But, without a predictive model for TVSQ, it is impossible to qualitatively assess how much quality improvement is needed. Another important property of the TVSQ is its nonlinearity. In particular, the sensitivity of the TVSQ to quality variation is not constant. This property should also be utilized for resource allocation among users sharing a network resource (such as transmission time in TDMA systems). For example, when the TVSQ of a user is insensitive to quality variations, the rate-controller could reserve some transmission resources by reducing the bitrate

Manuscript received June 20, 2013; revised November 16, 2013 and March 4, 2014; accepted March 4, 2014. Date of publication March 19, 2014; date of current version April 8, 2014. This work was supported in part by Intel Inc. and in part by Cisco Corporation through the VAWN Program. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Stefan Winkler.

The authors are with the Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX 78712-0240 USA (e-mail: tochenchao@gmail.com; larkkwonchoi@gmail.com; gustavo@ece.utexas.edu; constantine@utexas.edu; rheath@utexas.edu; bovik@ece.utexas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2014.2312613

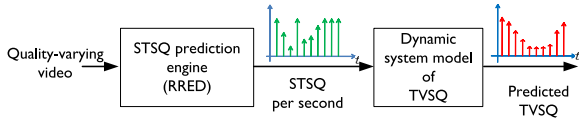


Fig. 1. Proposed paradigm for TVSQ prediction.

without lowering the user's TVSQ. The reserved resources could then be used to increase the bitrate of other users and thus improve their TVSQs. A predictive model for TVSQ is an essential tool to assess the sensitivity of TVSQ and to achieve quality-efficient rate adaptation.

The goal of this paper is to develop a predictive model that captures the impact of quality variations on TVSQ. The model predicts the average TVSQ every second and can be used to improve rate-adaptation algorithms for HTTP-based video streaming.

We propose to predict TVSQ in two steps (see Fig. 1). The two steps capture the spatial-temporal characteristics of the video and the hysteresis effects in human behavioral responses, respectively. In the first step, quality-varying videos are partitioned into one second long video chunks and the short-time subjective quality (STSQ) of each chunk is predicted. Unlike TVSQ, which is a temporal record, the STSQ is *a scalar prediction of viewers' subjective judgment of a short video's overall perceptual quality*. A STSQ prediction model such as those in [7] and [11]–[14], operates by extracting perceptually relevant spatial and temporal features from short videos then uses these to form predictions of STSQ. Hence, STSQ contains useful, but incomplete evidence about TVSQ. Here, the Video-RRED algorithm [14] is employed to predict STSQs because of its excellent quality prediction performance and fast computational speed. In the second step, the predicted STSQs are sent to a dynamic system model, which predicts the average TVSQ every second. The model mimics the hysteresis effects with a linear filter and captures the nonlinearity in human behavior with nonlinear functions at the input and the output of the linear filter. In HTTP-based streaming protocols, the interval between consecutive video data rate adaptations is usually several seconds long.[1] Since the proposed model predicts the average TVSQ per second, the prediction timescales are suitable for HTTP-based streaming.

The contributions of this paper are summarized as follows:

1) *A new database for the TVSQ of HTTP-based video streams.* A database of rate-varying video sequences is built to simulate quality fluctuations commonly encountered in video streaming applications.[2] Then, a subjective study was conducted to measure the TVSQs of these video sequences. This database is useful for developing and validating TVSQ models and thus is important in its own right, as it may contribute to future research efforts.
2) *An effective TVSQ prediction method.* Using the new database, a dynamic system model is proposed to predict the average TVSQ per second of video. Experimental results show that the proposed model reliably tracks the TVSQ of video sequences with time-varying qualities. The dynamic system model has a simple structure and is computationally efficient for TVSQ prediction. It is in fact suitable for online TVSQ-optimized rate adaptation. In HTTP-based video streaming protocols, the video is encoded into multiple representations at different video data rates. These representations are stored on the video server before transmission. Thus, the rate-STSQ function for each second of the video can be computed off-line before transmission. Since the proposed dynamic system model predicts the TVSQ from the STSQ, we may combine the rate-STSQ function with the dynamic system model to obtain a rate-TVSQ model. This rate-TVSQ model can then be used to determine the video data rate that optimizes the TVSQ.

Related Work: TVSQ is an important research subject in the realm of visual quality assessment [9], [10], [15]–[18]. In [10], the relationship between STSQ and TVSQ for packet videos transmitted over ATM networks was studied. A so-called "recency effect" was observed in their subjective experiments. At any moment, the TVSQ is quite sensitive to the STSQs over the previous (at least) 20-30 seconds [10]. Thus, the TVSQ at any moment depends not only on the current video quality, but also on the preceding viewing experience. In [15], Tan *et al.* proposed an algorithm to estimate TVSQ. They first applied an image quality assessment algorithm to each video frame. Then they predicted the TVSQ with per-frame qualities using a "cognitive emulator" designed to capture the hysteresis of the human behavioral responses to visual quality variations. The performance of this model was evaluated on a database of three videos, on which the encoding data rates were adapted over a slow time scale of 30-40 seconds [15]. In [16], a first-order infinite impulse response (IIR) filter was used to predict the TVSQ based on per-frame distortions, which were predicted by spatial and temporal features extracted from the video. This method was shown to track the dynamics of the TVSQ on low bit-rate videos. In [17], an adaptive IIR filter was proposed to model the TVSQ. Since the main objective of [17] was to predict the *overall* subjective quality of a long video sequence using the predicted TVSQ, the performance of this model was not validated against the measured TVSQ. In [9], a temporal pooling strategy was employed to map the STSQ to the overall video quality using a model of visual hysteresis. As an intermediate step, the STSQ was first mapped to the TVSQ, then the overall quality was estimated as a time-averaged TVSQ. Although this pooling strategy yields good predictions of the overall video quality, the model for the TVSQ is a non-causal system, which contradicts the fact that the TVSQ at a moment only depends on current and previous STSQs. In [18], a convolutional neural network was employed to map features extracted from each video frame to the TVSQ. The estimated TVSQs were shown to achieve high correlations with the measured TVSQ values on constant bitrate videos.

In [9] and [17], estimated TVSQ was used as an intermediate result in an overall video quality prediction process. However, the performances of these models were not

[1]For example, in MPEG-DASH [4], the rate adaptation interval is at least two seconds.

[2]Since HTTP is based on TCP, which guarantees that the data is delivered without packet loss. Thus, only encoding distortions are considered.

validated against recorded subjective TVSQ. The TVSQ models proposed in [15], [16], and [18] mainly targeted videos for which the encoding rate was fixed or changing slowly. Newly proposed HTTP-based video streaming protocols, e.g., DASH, provide the flexibility to adapt video bitrates over timescales as short as 2 seconds. Thus the prior models cannot be directly applied to estimate the TVSQ for HTTP-based video streaming.

In this paper, a new video quality database is built and is specifically configured to enable the development of TVSQ prediction models of HTTP-based video streaming. The STSQs of the videos in the new database were designed to vary randomly over time scales of several seconds in order to simulate the quality variations encountered in HTTP-based video streaming. The database consists of 15 videos. Each video is 5 minutes long and is viewed by 25 subjects.

Organization and Notation: The remainder of this paper is organized as follows: Section II introduces the new TVSQ database and describe its construction. Section III explains the model for TVSQ prediction. In Section IV, the model is validated through extensive experimentation and by a detailed system theoretic analysis.

Some of the key notation are briefly introduced as follows. Let $\{x[t],\ t = 1, 2 \dots\}$ denote discrete time series. The notation $\left(x\right)_{t_1:t_2}$ denotes the column vector $(x[t_1], x[t_1+1], \dots, x[t_2])$. The zero-padded convolution of $\left(x\right)_{t_1:t_2}$ and $\left(y\right)_{t_1:t_2}$ is denoted by $\left(x\right)_{t_1:t_2} * \left(y\right)_{t_1:t_2}$. Lower-case symbols such as a denote scalar variables. Random variables are denoted by uppercase letters such as A. Boldface lowercase symbols such as $\mathbf{a}$ denote column vectors and $\mathbf{a}^\mathsf{T}$ is the transpose of $\mathbf{a}$. Calligraphic symbols such as $\mathcal{A}$ denote sets while $|\mathcal{A}|$ is the cardinality of $\mathcal{A}$. Finally, the function $\nabla_{\mathbf{a}} \mathrm{f}(\mathbf{a}, \mathbf{b})$ denotes the gradient of the multivariate function $\mathrm{f}(\mathbf{a}, \mathbf{b})$ with respect to variable $\mathbf{a}$.

II. Subjective Study for Model Identification

In this section, the construction of the database and the design of the subjective experiments is described first. Then, based on the experimental results, the dynamic system model for TVSQ prediction is motivated.

A. Quality-Varying Video Construction

Using the following 5 steps, 15 quality-varying videos were constructed such that their STSQs vary randomly across time.

1. Eight high quality, uncompressed video clips with different content were selected. These clips have a spatial resolution of 720p (1280×720) and a frame rate of 30 fps. A short description of these clips is provided in Table I. The content was chosen to represent a broad spectrum of spatial and temporal complexity (see sample frames in Fig. 2).
2. Using the video clips selected in the first step, 3 reference videos were constructed. They were used to generate quality-varying videos in the subjective study. Each reference video was constructed by concatenating 5 or 6 different clips (see Fig. 3). The reference video were constructed in this way because long videos with monotonous content can be boring to subjects. This could adversely impact the accuracy of the TVSQ measured in the subjective study. The length of each video is 300 seconds, which was chosen to agree with the value recommended by the ITU [19]. This is longer than the videos tested in [9], [15], [16], [18], [20], and [21]

TABLE I
A Brief Description of the Video Clips in Our Database

Name	Abbreviation	Description
Fountain	ft	Still camera, shows a fountain.
Turtles	tu	Still camera, a girl is feeding turtles.
Stick	st	Still camera, a man is waving a stick.
Bulldozer	bu	Camera span, a man is driving a bulldozer.
Singer&girl	sg	Camera zoom, a man is singing to a girl.
Volleyball	vo	Still camera, shows a volleyball game.
Dogs	do	Camera span, two dogs play near a pool.
Singer	si	Camera zoom, a singer is singing a song.

Fig. 2. Sample frames of the video clips involved in the subjective study. The abbreviations of the names of the videos can be found in Table. I. (a) ft. (b) tu. (c) st. (d) bu. (e) sg. (f) vo. (g) do. (h) si.

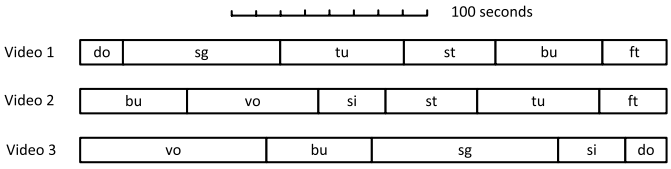


Fig. 3. The construction of the reference videos. The abbreviation of the names of the clips can be found in Table I.

thus is a better tool towards understanding the long-term behaviorial responses of human vision system.

3. For each reference video, 28 distorted versions were generated. Specifically, every reference video sequence was encoded into 28 constant bitrate streams using the H.264 encoder in [22] and then were decoded. To achieve a wide range of video quality exemplars, the encoding bitrates were chosen from hundreds of Kbps to several Mbps.
4. Every distorted version was partitioned into 1 second long video chunks and their STSQs were predicted with the computationally efficient and perceptually accurate RRED index [14]. Let the RRED index of the t^{th} chunk in the ℓ^{th} distorted version of the k^{th} reference video be denoted by $\mathrm{q}^{\text{rred}}_{\ell,k}[t]$, where $t \in \{1,\ldots,300\}$ second, $\ell \in \{1,\ldots,28\}$, and $k \in \{1,2,3\}$. Then the Difference Mean Opinion Score (DMOS, see [23]) of the STSQ for each chunk was predicted via logistic regression:

$$\mathrm{q}^{\text{dmos}}_{\ell,k}[t] = 16.4769 + 9.7111 \log\left(1 + \frac{\mathrm{q}^{\text{rred}}_{\ell,k}[t]}{0.6444}\right). \quad (1)$$

The regression model in (1) was obtained by fitting a logistic mapping from the RRED index to the DMOSs on the LIVE Video Quality Assessment Database [24]. Here, the predicted DMOS $\mathrm{q}^{\text{dmos}}_{\ell,k}[t]$ ranges from 0 to 100 where lower values indicate better STSQ. To represent STSQ more naturally, so that higher numbers indicate better STSQ, we define the Reversed DMOS (RDMOS) corresponding to a DMOS of x to be $100 - x$. Thus, the predicted RDMOS of the STSQ for each chunk is given by:

$$\mathrm{q}^{\text{rdmos}}_{\ell,k}[t] = 100 - \mathrm{q}^{\text{dmos}}_{\ell,k}[t]. \quad (2)$$

Broadly speaking, a RDMOS of less than 30 on the LIVE databases [24] indicates bad quality, while scores higher than 70 indicate excellent quality. As an example, Fig. 4(a) plots $\mathrm{q}^{\text{rdmos}}_{\ell,k}[t]$ for all of the distorted versions of the first reference video. Clearly, their STSQ covers a wide range of RDMOSs.
5. Finally, for each reference video, 6 quality-varying videos were constructed by concatenating the video chunks selected from different distorted versions. For the k^{th} reference video, 6 target STSQ sequences $\left\{\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)_{1:300}, j = 1,\ldots,6\right\}$ were designed to simulate the typical quality variation patterns in HTTP-based streaming (see section II-B for more details). Then, 6 quality-varying videos were constructed such that their STSQs approximate the target sequences. Specifically, the t^{th} chunk of the j^{th} quality-varying video was constructed by copying the t^{th} chunk in the $\ell^*_{t,j,k}$-th distorted version, where

$$\ell^*_{t,j,k} = \arg\min_{\ell} \left|\mathrm{q}^{\text{tgt}}_{j,k}[t] - \mathrm{q}^{\text{rdmos}}_{\ell,k}[t]\right|. \quad (3)$$

Denoting the STSQ of the t^{th} chunk in the obtained video by $\mathrm{q}^{\text{st}}_{j,k}[t]$, we have

$$\mathrm{q}^{\text{st}}_{j,k}[t] = \mathrm{q}^{\text{rdmos}}_{\ell^*_{t,j,k}}[t]. \quad (4)$$

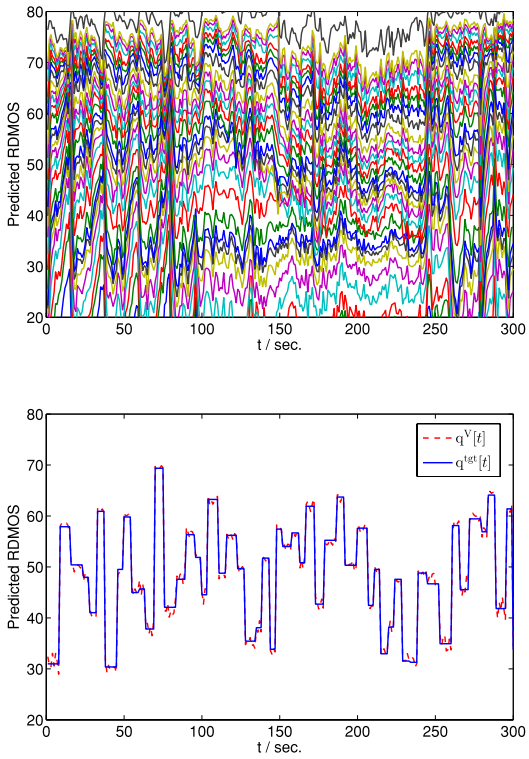


Fig. 4. (a) The STSQ of each compressed version of the reference video is shown in different colors. (b) A example of the designed target video quality $\mathrm{q}^{\text{tgt}}[t]$ and the actual video quality $\mathrm{q}^{\text{st}}[t]$ of the video sequence used in our database.

As can be seen in Fig. 4, since the RDMOS scale is finely partitioned by the RDMOSs of the compressed versions, the error between the obtained STSQ $\mathrm{q}^{\text{st}}_{j,k}[t]$ and the target STSQ $\mathrm{q}^{\text{tgt}}_{j,k}[t]$ is small. Among the 6 quality-varying videos generated from each reference video, 1 video is used for subjective training and the other 5 videos are used for subjective test. In all, $3 \times 1 = 3$ training videos and $3 \times 5 = 15$ test videos were constructed.

With this procedure, the pattern of quality variations in the test video sequences is determined by the target video quality sequence $\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)$. The design of $\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)$ is described next.

B. Target Video Quality Design

To obtain a good TVSQ prediction model for videos streamed over HTTP, the target video quality $\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)_{1:300}$ was designed such that the generated quality-varying videos can roughly simulate the STSQs of videos streamed over HTTP. In HTTP-based video streaming protocols such as those described in [1]–[4], videos are encoded into multiple representations at different bitrates. Each representation is then partitioned into segments, each several seconds long. The client dynamically selects a segment of a representation to download. Therefore, in our subjective study, $\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)_{1:300}$ was designed as a piece-wise constant time-series. Specifically, $\left(\mathrm{q}^{\text{tgt}}_{j,k}\right)_{1:300}$ was generated using two independent random processes. The first random process $\{\mathsf{D}(s) : s = 1,2,\ldots\}$ simulates the length of the video segments. The second random process

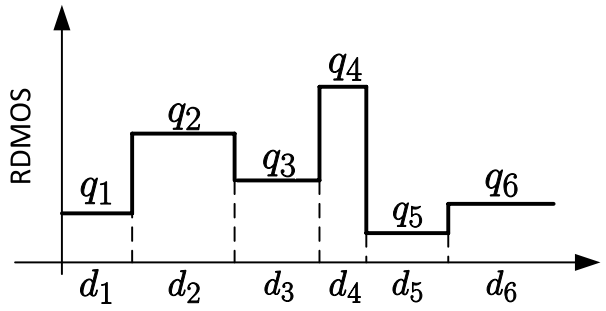


Fig. 5. The design of the target STSQs. The durations of each segment $d_1, d_2, \ldots$ were realizations of $\mathsf{D}(1), \mathsf{D}(2), \ldots$. The STSQ levels $q_1, q_2, \ldots$ were realizations of $\mathsf{Q}(1), \mathsf{Q}(2), \ldots$.

$\{\mathsf{Q}(s) : s = 1, 2, \ldots\}$ simulates the STSQs of segments. The sequence $\left(\mathrm{q}^{\mathrm{tgt}}_{j,k}\right)_{1:300}$ was constructed as a series of constant-value segments where the durations of the segments were given by $\mathsf{D}(s)$ and the RDMOSs of the segments were given by $\mathsf{Q}(s)$ (see Fig. 5).

In HTTP-based video streaming protocols, the duration of video segments can be flexibly chosen by the service provider. Shorter durations allow more flexibility for rate adaptation when the channel condition is varying rapidly. For example, due to the mobility of wireless users, the wireless channel throughput may vary on time scales of several seconds [25]. Consequently, this work focus on applications where the lengths of the segments are less than 10 seconds. TVSQ modeling for videos undergoing slowly varying data rates has been investigated in [15], [16], and [18]. In a subjective experiment, there is always a delay or latency between a change in STSQ and a subject's response. During the experimental design, we found that if the video quality varied too quickly, subjects could not reliably track their judgments of quality to the viewed videos. Specifically, when the video quality changes, a subject may take 1-2 seconds to adjust his/her opinion on the TVSQ. If the quality is adapted frequently, the quality variations that occur during this adjustment process can annoy the subject and thus reduce the accuracy of the measured TVSQs. Thus, we restricted the length of each segment to be at least 4 seconds, which is comfortably longer than the subjects' latency and short enough to model quality variations in adaptive video streaming. In sum, the random process $\{\mathsf{D}(s) : s = 1, 2, \ldots\}$ takes values from the set $\{4, 5, 6, 7, 8, 9, 10\}$.

The distribution of STSQs of a video transported over HTTP depends on many factors including the encoding bitrates, the rate-quality characteristics, the segmentation of each representation, the channel dynamics, and the rate adaptation strategy of the client. To sample uniformly from among all possible patterns of STSQ variations, the random processes $\mathsf{D}(s)$ and $\mathsf{Q}(s)$ were designed as i.i.d. processes, which tend to traverse all possible patterns of quality variations. Also, the distributions of $\mathsf{D}(s)$ and $\mathsf{Q}(s)$ were designed to "uniformly" sample all possible segment lengths and STSQ levels, respectively. To this end, we let $\mathsf{D}(s)$ take values in the set $\{4, 5, 6, 7, 8, 9, 10\}$ with equal probability. Similarly, the distribution of $\mathsf{Q}(s)$ was designed such that the sample values of $\mathsf{Q}(s)$ would be distributed as if the videos were uniformly sampled in the LIVE database, because that set of videos is carefully chosen to represent a wide range of perceptually separated STSQ [24]. The RDMOSs of videos in the LIVE database are distributed as approximately obeying a normal distribution $\mathcal{N}(50, 10^2)$ [24]. Therefore, we let the distribution of $\mathsf{Q}(s)$ be $\mathcal{N}(50, 10^2)$. In the LIVE database, almost all of the recorded RDMOSs fall within the range $[30, 70]$. Videos with RDMOS lower than 30 are all very severely distorted while videos with RDMOS higher than 70 are all of high quality. Due to saturation of the subjects' scoring capability outside these ranges, the recorded qualities of videos with RDMOSs lower than 30 or higher than 70 are difficult to distinguish. Therefore, we truncated $\mathsf{Q}(s)$ to the range $[30, 70]$.

C. Subjective Experiments

A subjective study was conducted to measure the TVSQs of the quality-varying videos in our database. The study was completed at the LIVE subjective testing lab at The University of Texas at Austin. The videos in our database were grouped into 3 sessions. Each session included one of the three reference videos and the 6 quality-varying videos generated from the reference video. The videos in each session were each viewed and scored by 25 subjects. One of the quality-varying videos was used as a training sequence. The other six videos, including 5 quality-varying videos and the reference video, were used for subjective study. The subjects were not notified about the existence of the reference videos. The subjective scores obtained from these reference videos were then used for the computation the RDMOSs of the TVSQs [23].

A user interface was developed for the subjective study using the Matlab XGL toolbox [26]. The user interface ran on a Windows PC with an Intel Xeon 2.93GHz CPU and a 24GB RAM. The XGL toolbox interfaced with ATI Radeon X300 graphics card on the PC to precisely display video frames without latencies or frame drops, by loading each video into memory before display. Video sequences were displayed to the viewers on a 58 inch Panasonic HDTV plasma monitor at a viewing distance of about 4 times the picture height. During the play of each video, a continuous scale sliding bar was displayed near the bottom of the screen. Similar to the ITU-R ACR scale [19], the sliding bar was marked with five labels: "Bad", "Poor", "Fair", "Good", and "Excellent", equally spaced from left to right. The subject could continuously move the bar via a mouse to express his/her judgment of the video quality as each video is played. The position of the bar was sampled and recorded automatically in real time as each frame is displayed (30 fps). No mouse clicking was required in the study. Fig. 6 shows the subjective study interface including a frame of a displayed video.

During the training period, each subject first read instructions describing the operation of the user interface (see Appendix A), then practiced on the training sequence. The subject then started rating the test videos (reference video and five quality-varying videos) shown in random order. The subjects were unaware of the presence of the reference video.

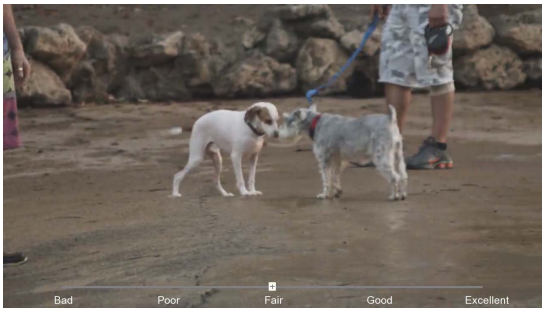


Fig. 6. User interface used in the subjective study.

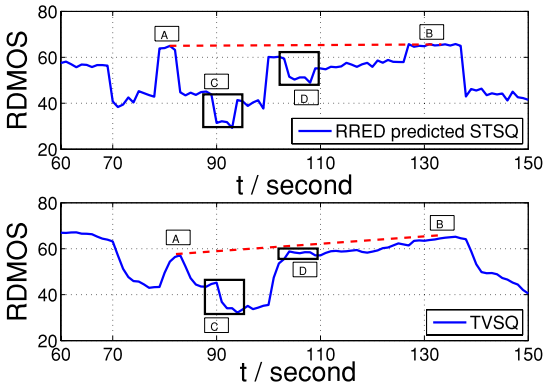


Fig. 7. (Upper) the STSQs predicted using Video-RRED. (Lower) the TVSQs measured in the subjective study.

D. Data Preprocessing

Denote the average score assigned by the i^{th} subject to the frames of the t^{th} chunk of the j^{th} quality-varying video in the k^{th} session by $\mathrm{c}_{i,j,k}[t]$. Let the score assigned to the reference video be denoted by $\mathrm{c}^{\text{ref}}_{i,k}[t]$. The impact of video content was offseted on the TVSQs of the test videos using

$$\mathrm{c}^{\text{offset}}_{i,j,k}[t] = 100 - \left(\mathrm{c}^{\text{ref}}_{i,k}[t] - \mathrm{c}_{i,j,k}[t]\right). \tag{5}$$

In (5), we subtracted $\left(\mathrm{c}^{\text{ref}}_{i,k}[t] - \mathrm{c}_{i,j,k}[t]\right)$ from 100 to compute the RDMOS from $\mathrm{c}^{\text{offset}}_{i,j,k}[t]$. Let T denote the length of the test videos and J denote the number of quality-varying videos in each session. In our experiment, $\mathrm{T} = 300$ and $\mathrm{J} = 5$. Note that the subjects deliver their quality judgments in real-time as the test video is being displayed. To avoid distracting the subjects from viewing the video, we did not require them to use the full scale of the sliding bar. Moreover, such an instruction may tend to bias the recorded judgments from their natural response. Thus, each subject was allowed to freely deploy the sliding bar when expressing their judgments of TVSQ. To align the behavior of different subjects, paralleling to prior work such as [27]–[31], we normalize $\left(\mathrm{c}^{\text{offset}}_{i,j,k}\right)$ by computing the Z-scores [32] as follows:

$$\begin{aligned} m_{i,k} &= \frac{1}{\mathrm{J}}\frac{1}{\mathrm{T}}\sum_{j=1}^{\mathrm{J}}\sum_{t=1}^{\mathrm{T}}\mathrm{c}^{\text{offset}}_{i,j,k}[t]; \\ \sigma^2_{i,k} &= \frac{1}{\mathrm{JT}-1}\sum_{\mathrm{j}=1}^{\mathrm{J}}\sum_{t=1}^{\mathrm{T}}\left(\mathrm{c}^{\text{offset}}_{i,j,k}[t] - m_{i,k}\right)^2; \\ \mathrm{z}_{i,j,k}[t] &= \frac{\mathrm{c}^{\text{offset}}_{i,j,k}[t] - m_{i,k}}{\sigma_{i,k}}. \end{aligned} \tag{6}$$

In (6), the values of $m_{i,k}$ and $\sigma_{i,k}$ are respectively the mean and the variance of the scores assigned by the i^{th} subject in the k^{th} session. The value of $\mathrm{z}_{i,j,k}[t]$ is the normalized score. Let I denote the number of subjects. We have $\mathrm{I} = 25$. Then for the t^{th} second of the j^{th} test video, the average and standard deviation of the Z-scores assigned by the subjects were computed

$$\begin{aligned} \mu_{j,k}[t] &= \frac{1}{\mathrm{I}}\sum_{i=1}^{\mathrm{I}}\mathrm{z}_{i,j,k}[t]; \\ \eta^2_{j,k}[t] &= \frac{1}{\mathrm{I}-1}\sum_{i=1}^{\mathrm{I}}\left(\mathrm{z}_{i,j,k}[t] - \mu_{j,k}[t]\right)^2. \end{aligned} \tag{7}$$

If $\mathrm{z}_{i,j,k}[t] > \mu_{j,k}[t] + 2\eta_{j,k}[t]$ or $\mathrm{z}_{i,j,k}[t] < \mu_{j,k}[t] - 2\eta_{j,k}[t]$, $\mathrm{z}_{i,j,k}[t]$ was marked as an outlier because the Z-score given by subject i deviates far from the Z-scores given by the other subjects. The outliers were excluded and the Z-scores were recomputed using (6). Let $\mathcal{O}_{j,k,t}$ denote the set of subjects who assigned outlier Z-scores to the t^{th} chunk of the j^{th} video in the k^{th} session. The averaged Z-score of the TVSQ for the t^{th} chunk is then

$$\bar{\mathrm{z}}_{j,k}[t] = \frac{1}{\mathrm{I} - |\mathcal{O}_{j,k,t}|}\sum_{i\notin\mathcal{O}_{j,k,t}}\mathrm{z}_{i,j,k}[t]. \tag{8}$$

The 95% confidence interval of the average Z-scores is $\bar{\mathrm{z}}_{j,k}[t] \pm 1.96\eta_{j,k}[t]/\sqrt{\mathrm{I} - |\mathcal{O}_{j,k,t}|}$. We found that the values of the averaged Z-scores all lie in the range $[-4, 4]$. Therefore, $\bar{\mathrm{z}}_{j,k}[t]$ was mapped to the range $[0, 100]$ using the following formula:

$$\mathrm{q}^{\text{tv}}_{j,k}[t] = \frac{\bar{\mathrm{z}}_{j,k}[t] + 4}{8} \times 100. \tag{9}$$

Correspondingly, the 95% confidence interval of TVSQ is $\mathrm{q}^{\text{tv}}_{j,k}[t] \pm \epsilon_{j,k}[t]$, where

$$\epsilon_{j,k}[t] = \frac{1.96\eta_{j,k}[t]/\sqrt{\mathrm{I} - |\mathcal{O}_{j,k,t}|} + 4}{8} \times 100. \tag{10}$$

In all, the TVSQ for $\mathrm{N} = 3 \times 5 = 15$ quality-varying videos were measured. In the following, we replace the subscript (j, k) with a subscript $1 \le n \le \mathrm{N}$ to index the quality-varying videos and denote by $\mathrm{q}^{\text{tv}}_n[t]$ and $\epsilon_n[t]$ the measured TVSQ and the confidence interval of the n^{th} video, respectively. Similarly, the STSQ of the n^{th} video predicted by the Video-RRED algorithm [14] is denoted by $\mathrm{q}^{\text{st}}_n[t]$.

E. Preliminary Observations

Since $\left(\mathrm{q}^{\text{st}}\right)$ is the predicted STSQ, we expect $\left(\mathrm{q}^{\text{st}}\right)$ to contain useful evidence about the TVSQ. The $\left(\mathrm{q}^{\text{st}}\right)$ and the corresponding $\left(\mathrm{q}^{\text{tv}}\right)$ of the 6^{th} quality-varying video from $t = 61$ to $t = 150$ is plotted in Fig. 7. It is seen that both the $\left(\mathrm{q}^{\text{st}}\right)$ and the $\left(\mathrm{q}^{\text{tv}}\right)$ follow the similar trend of variation. But it should be noted that the relationship between $\left(\mathrm{q}^{\text{st}}\right)$ and $\left(\mathrm{q}^{\text{tv}}\right)$ cannot be simply described by a static mapping. For example, at point A ($t = 29$) and point B ($t = 85$), the $\mathrm{q}^{\text{st}}[t]$ takes similar values.

But the corresponding $q^{tv}[t]$ is lower at point A than point B. This observation could be explained by the hysteresis effects. Prior to point A, $q^{st}[t]$ is around 40 (see $(q^{st})_{20:28}$). But, prior to point B, $q^{st}[t]$ is around 65 (see $(q^{st})_{76:84}$). Thus, the previous viewing experience is worse at point A, which gives rise to a lower TVSQ. Such hysteresis effects should be considered in HTTP-based rate adaptations. For example, if the "previous viewing experience" is bad (such as point A), the server should send the video segment of higher quality to counterbalance the impact of bad viewing experience on the TVSQ.

It may be observed that the $q^{st}[t]$ experiences the similar level of drop in region C and region D. The drop of $q^{st}[t]$ in region C results in a significant drop in $q^{tv}[t]$. But, in region D, $q^{st}[t]$ is not as affected by the drop of $q^{st}[t]$. In other words, the sensitivity of TVSQ to the variation in (q^{st}) is different in region C and region D. This is probably due to the non-linearities of human behavioral responses. Including such nonlinearities is critical for efficient HTTP-based adaptation. Specifically, when the TVSQ is insensitive to the STSQ (such as in region D), the server may switch to a lower streaming bitrate to reserve some resources (such as transmission time) without hurting the TVSQ. Those reserved resources can then be used to maintain a good TVSQ when the TVSQ is sensitive (such as region C).

In sum, quantitatively modeling the hysteresis effects and the nonlinearities are critical for TVSQ-optimized rate adaptations. This motivate us to propose a non-linear dynamic system model, which is described in more detail below.

III. System Model Identification

In this section, the model for TVSQ prediction is presented in Section III-A. Then, the algorithm for model parameter estimation is described in Section III-B. The method for model order selection is introduced in Section III-C.

A. Proposed Model for TVSQ Prediction

Due to the hysteresis effect of human behaviorial responses to quality variations, the TVSQ at a moment depends on the viewing experience prior to the current moment. A dynamic system model can be used to capture the hysteresis effect using the "memory" of the system state. The simplest type of dynamic system is a linear filter. The human vision system, however, is non-linear in general [33]–[35]. Although introducing intrinsic nonlinearities into the dynamic system model could help to capture those nonlinearities,[3] the dynamic system would become too complicated to provide guidance on the design of TVSQ-optimized rate-adaptation algorithms. More specifically, due to the randomness of channel conditions, the TVSQ-optimized rate-adaptation algorithm design is essentially a stochastic optimization problem. For a linear dynamic model with input (x), its output (y) is given by $(\mathrm{y}) = (\mathrm{h}) * (\mathrm{x})$, where (h) is the impulse response. Due to the linearity of expectation, the expectation of the TVSQs can be characterized using $\mathbb{E}[y] = ||\mathbf{h}||_1 \mathbb{E}[x]$. For a dynamic model with intrinsic nonlinearities, however, linearity of expectation cannot be applied and analyzing the average behavior of the TVSQ becomes difficult. Therefore, we employed a Hammerstein-Wiener (HW) model [36], which captures the nonlinearity with extrinsic nonlinear functions. The model is illustrated in Fig. 8. The core of the HW model is a linear filter (see [36]) which is intended to capture the hysteresis. At the input and output of the HW model, two non-linear static functions are employed to model potential non-linearities in the human response. We call these two functions input nonlinearity and output nonlinearity, respectively.

[3]We say a nonlinear system has intrinsic nonlinearity if its current system state is a nonlinear function of the previous system state and input. Otherwise, we say the system has extrinsic nonlinearity.

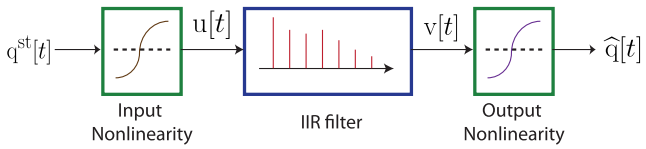


Fig. 8. Proposed Hammerstein-Wiener model for TVSQ prediction.

The linear filter has the following form:

$$\begin{aligned} \mathrm{v}[t] &= \sum_{d=0}^{r} b_d \, \mathrm{u}[t-d] + \sum_{d=1}^{r} f_d \, \mathrm{v}[t-d] \\ &= \mathbf{b}^{\mathsf{T}} (\mathrm{u})_{t-r:t} + \mathbf{f}^{\mathsf{T}} (\mathrm{v})_{t-r:t-1}, \end{aligned} \tag{11}$$

where the parameter r is the model order and the coefficients $\mathbf{b} = (b_0, \ldots, b_r)^{\mathsf{T}}$ and $\mathbf{f} = (f_1, \ldots, f_r)^{\mathsf{T}}$ are model parameters to be determined. At any time t, the model output $\mathrm{v}[t]$ depends not only on the previous r seconds of the input $\mathrm{u}[t]$, but also on the previous r seconds of $\mathrm{v}[t]$ itself. Thus this filter has an infinite impulse response (IIR). We employed this model rather than a finite impulse response (FIR) filter because the IIR filter can model the long-term impact of quality variations with a lower model order and thus using fewer parameters. To train a parameterized model, the size of the training data set increases exponentially with the number of the parameters [36]. Therefore, it is easier to train an IIR model. A drawback of the IIR filter (11) is its dependency on its initial state. Specifically, to compute $(\mathrm{v})_{t>r}$, the initial r seconds of output $(\mathrm{v})_{1:r}$ need to be known. But $(\mathrm{v})_{1:r}$ is the TVSQ of the user, which is unavailable. Actually, it can be shown that this unknown initial condition only has negligible impact on the performance of the proposed model. A more detailed analysis is presented in Section IV-B.

To model the input and output nonlinearities of the HW model, we have found that if the input and output static functions are chosen as generalized sigmoid functions [37], then the proposed HW model can predict TVSQ accurately. Thus, the input and output functions were set to be

$$\mathrm{u}[t] = \beta_3 + \beta_4 \frac{1}{1 + \exp\left(-(\beta_1 \mathrm{q}^{\mathrm{st}}[t] + \beta_2)\right)}, \tag{12}$$

and

$$\widehat{\mathrm{q}}[t] = \gamma_3 + \gamma_4 \frac{1}{1 + \exp\left(-(\gamma_1 \mathrm{v}[t] + \gamma_2)\right)}, \tag{13}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_4)^{\mathsf{T}}$ and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_4)^{\mathsf{T}}$ are model parameters and $\widehat{\mathrm{q}}$ is the predicted TVSQ.

Let $\boldsymbol{\theta} = (\mathbf{b}^{\mathsf{T}}, \mathbf{f}^{\mathsf{T}}, \boldsymbol{\beta}^{\mathsf{T}}, \boldsymbol{\gamma}^{\mathsf{T}})^{\mathsf{T}}$ be the parameters of the proposed HW model, and let $\widehat{\mathrm{q}}$ be regarded as a function both of time t and parameter $\boldsymbol{\theta}$. Thus, in the following, we explicitly rewrite $\widehat{\mathrm{q}}$ as $\widehat{\mathrm{q}}(t, \boldsymbol{\theta})$. To find the optimal HW model for TVSQ prediction, two things need to be determined: the model order r and the model parameter $\boldsymbol{\theta}$. In the following, we first show how to optimize the model parameter $\boldsymbol{\theta}$ for given model orders. Then, we introduce the method for model order estimation.

B. Model Parameter Training

This section discusses how to optimize the model parameter $\boldsymbol{\theta}$ such that the error between the measured TVSQ and the predicted TVSQ can be minimized. In system identification and machine learning, the mean square error (MSE) is the most widely used error metric. Denoting the predicted TVSQ of the n^{th} video by $\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta})$, the MSE is defined as $\frac{1}{\mathrm{NT}} \sum_{n=1}^{\mathrm{N}} \sum_{t=1}^{\mathrm{T}} \left(\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta}) - \mathrm{q}_n^{\mathrm{tv}}[t]\right)^2$. The MSE always assigns a higher penalty to a larger estimation error. For the purposes of tracking TVSQ, however, once the estimated TVSQ deviates far from the measured TVSQ, the model fails. There is no need to penalize a large error more than another smaller, yet still large error. Furthermore, since the $\mathrm{q}_n^{\mathrm{tv}}[t]$ is just the average subjective quality judgment, the confidence interval of TVSQ $\epsilon_n[t]$ (see the definition in (10)) should also be embodied in the error metric to account for the magnitude of the estimation error. We chose to use the outage rate, also used in [18], as the error metric. Specifically, the outage rate of a TVSQ model is defined as the frequency that the estimated TVSQ deviates by at least twice the confidence interval of the measured TVSQ. More explicitly, outage rate can be written as

$$\mathrm{E}(\boldsymbol{\theta}) = \frac{1}{\mathrm{NT}} \sum_{n=1}^{\mathrm{N}} \sum_{t=1}^{\mathrm{T}} \mathbb{1}\left(\left|\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta}) - \mathrm{q}_n^{\mathrm{tv}}[t]\right| > 2\epsilon_n[t]\right), \quad (14)$$

where $\mathbb{1}(\cdot)$ is the indicator function.

Gradient-descent parameter search algorithms are commonly used for model parametrization. In our case, however, the gradient of the indicator function $\mathbb{1}(\cdot)$ in (14) is zero almost everywhere and thus the gradient algorithm cannot be applied directly. To address this difficulty, we approximated the indicator function $\mathbb{1}(|x| > 2\epsilon)$ by a penalty function

$$\mathrm{U}_\nu(x, \epsilon) = \mathrm{h}(x, \nu, -2\epsilon) + (1 - \mathrm{h}(x, \nu, 2\epsilon)), \quad (15)$$

where $\mathrm{h}(x, \alpha, \zeta) = 1/(1 + \exp(-\alpha(x + \zeta))$ is a logistic function. In Fig. 9, $\mathrm{U}_\nu(x, \epsilon)$ with different configurations of the parameter ν are plotted. It can be seen that, as $\nu \to \infty$, $\mathrm{U}_\nu(x, \epsilon)$ converges to $\mathbb{1}(|x| > 2\epsilon)$. The outage rate $\mathrm{E}(\boldsymbol{\theta})$ can thus be approximated by $\mathrm{E}(\boldsymbol{\theta}) = \lim_{\nu \to \infty} \mathrm{E}_\nu^{\mathrm{apx}}(\boldsymbol{\theta})$, where

$$\mathrm{E}_\nu^{\mathrm{apx}}(\boldsymbol{\theta}) = \frac{1}{\mathrm{NT}} \sum_{n=1}^{\mathrm{N}} \sum_{t=1}^{\mathrm{T}} \mathrm{U}_\nu\left(\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta}) - \mathrm{q}_n^{\mathrm{tv}}[t], \epsilon[t]\right). \quad (16)$$

The iterative algorithm used for model parameter identification is described in Algorithm 1. In the i^{th} iteration, a gradient-descent search algorithm is applied to minimize $\mathrm{E}_\nu^{\mathrm{apx}}(\boldsymbol{\theta})$. The obtained parameter $\boldsymbol{\theta}^i$ is then used as the starting point for the gradient-descent search in the $(i+1)^{\text{th}}$ iteration.

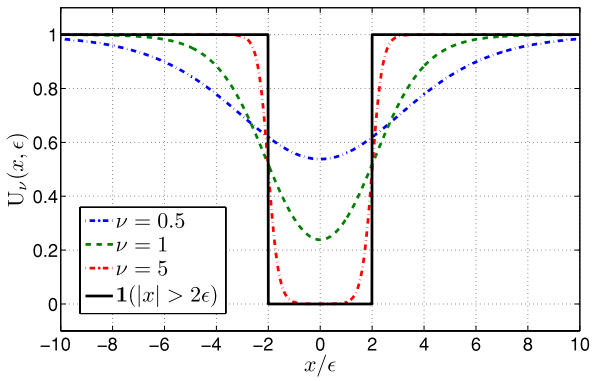


Fig. 9. $\mathrm{U}_\nu(x, \epsilon)$ with $\nu = 0.5$, 1 and 5.

Algorithm 1 Parameter Optimization Algorithm

Input: $\mathrm{q}_n^{\mathrm{st}}[t]$, $\mathrm{q}_n^{\mathrm{tv}}[t]$, $\epsilon_n[t]$, $i = 1$, and $\nu = 0.8$

while $\nu < 20$ **do**
  $\boldsymbol{\theta}^i = \arg\min_{\boldsymbol{\theta}} \mathrm{E}_\nu^{\mathrm{apx}}(\boldsymbol{\theta})$ via gradient-descent search starting from $\boldsymbol{\theta}^{i-1}$.
  $i := i + 1$
  $\nu := 1.2\nu$
end while

Algorithm 2 Gradient-Descent Algorithm

Input: $\mathrm{q}^{\mathrm{st}}[t]$, $\mathrm{q}^{\mathrm{tv}}[t]$, $\epsilon[t]$, ν, and $j = 1$

while $\mathrm{E}_\nu^{\mathrm{apx}}\left(\boldsymbol{\theta}^{j-1}\right) - \mathrm{E}_\nu^{\mathrm{apx}}\left(\boldsymbol{\theta}^{j}\right) \geq 10^{-5}$ **do**
  $\Delta\boldsymbol{\theta} := -\nabla_{\boldsymbol{\theta}} \mathrm{E}_\nu^{\mathrm{apx}}\left(\boldsymbol{\theta}^j\right)$
  while $\mathrm{E}_\nu^{\mathrm{apx}}\left(\boldsymbol{\theta}^j + \omega\Delta\boldsymbol{\theta}\right) > \mathrm{E}_\nu^{\mathrm{apx}}\left(\boldsymbol{\theta}^j\right) - 0.1\omega||\Delta\boldsymbol{\theta}||_2^2$ or $\rho(\mathbf{f}) \geq 1$ **do**
    $\omega := 0.7\omega$
  end while
  $\boldsymbol{\theta}^{j+1} := \boldsymbol{\theta}^j + \omega\Delta\boldsymbol{\theta}$
  $j := j + 1$
end while

At the end of each iteration, the parameter ν is increased by $\nu := 1.2\nu$.[4] Using this algorithm, the penalty function $\mathrm{U}_\nu(x, \epsilon)$ is gradually modified to $\mathbb{1}(|x| > 2\epsilon)$ such that the estimated TVSQ is forced into the confidence interval of the measured TVSQ. Note that when $\nu \geq 20$, $\mathrm{U}_\nu(x, \epsilon)$ is very close to $\mathbb{1}(|x| > 2\epsilon)$. Hence, the iteration is terminated when $\nu \geq 20$.[5]

The gradient-descent mechanism in Algorithm 1 is described by Algorithm 2. The algorithm contains two loops. In the outer loop, $\boldsymbol{\theta}$ is moved along the direction of negative gradient $-\nabla_{\boldsymbol{\theta}} \mathrm{E}_\nu^{\mathrm{apx}}(\boldsymbol{\theta})$ with a step-size ω. The loop is terminated when the decrement of the cost function between consecutive loops is less than a small threshold δ. On our database, we found that setting $\delta = 10^{-5}$ is sufficient.

[4]The choice of the multiplicative factor 1.2 is to balance the efficiency and accuracy of the algorithm. Any number less than 1.2 gives rise to similar performance. Any number larger than 1.2 results in worse performance.

[5]Since $\mathrm{E}(\boldsymbol{\theta})$ is not a convex function of θ, gradient-descent can only guarantee local optimality.

The inner loop of Algorithm 2 is a standard backtracking line search algorithm (see [38]), which determines an appropriate step-size ω. To calculate the gradient $\nabla_{\boldsymbol{\theta}} \mathrm{E}_v^{\mathrm{apx}}(\boldsymbol{\theta})$, we have

$$\nabla_{\boldsymbol{\theta}} \mathrm{E}_v^{\mathrm{apx}}(\boldsymbol{\theta}) = \frac{1}{\mathrm{NT}} \sum_{n=1}^{\mathrm{N}} \sum_{t=1}^{\mathrm{T}} \left[\frac{\mathrm{dU}_v(x, \epsilon_n[t])}{\mathrm{d}x} \bigg|_{x=\widehat{\mathrm{q}}_n(t,\boldsymbol{\theta})-\mathrm{q}_n^{\mathrm{tv}}[t]} \right] \nabla_{\boldsymbol{\theta}} \widehat{\mathrm{q}}_n(t, \boldsymbol{\theta}). \tag{17}$$

In (17), $\dfrac{\mathrm{dU}_v(x,\epsilon)}{\mathrm{d}x}$ can be directly derived from (15). The calculation of $\nabla_{\boldsymbol{\theta}} \widehat{\mathrm{q}}_n(t, \boldsymbol{\theta})$ is not straightforward since the dynamic model has a recurrent structure. Specifically, the input-output relationship of the HW model can be written as:

$$\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta}) = \mathrm{g}\left(\boldsymbol{\theta}, \left(\widehat{\mathrm{q}}_n\right)_{t-r:t-1}, \left(\mathrm{q}_n^{\mathrm{st}}\right)_{t-r:t}\right), \tag{18}$$

where the function $\mathrm{g}(\cdot)$ is the combination of (11), (12), and (13). The model output $\widehat{\mathrm{q}}_n(t, \boldsymbol{\theta})$ depends not only on $\boldsymbol{\theta}$ but also on previous system outputs $\left(\widehat{\mathrm{q}}_n\right)_{t-1:t-r}$, which depend on $\boldsymbol{\theta}$ as well.[6] Denoting by θ_i the i^{th} component of $\boldsymbol{\theta}$, differentiating both side of (18), we have

$$\frac{\partial \widehat{\mathrm{q}}_n(t, \boldsymbol{\theta})}{\partial \theta_i} = \frac{\partial \mathrm{g}}{\partial \theta_i} + \sum_{d=1}^{r} \frac{\partial \mathrm{g}}{\partial \widehat{\mathrm{q}}_n(t-d, \boldsymbol{\theta})} \frac{\partial \widehat{\mathrm{q}}_n(t-d, \boldsymbol{\theta})}{\partial \theta_i}. \tag{19}$$

Because of the structure of (19), computing $\frac{\partial \widehat{\mathrm{q}}_n(t,\boldsymbol{\theta})}{\partial \theta_i}$ is equivalent to filtering $\frac{\partial \mathrm{g}}{\partial \theta_i}$ by a filter with a transfer function

$$\mathrm{H}(z) = \frac{1}{1 - \sum_{d=1}^{r} \frac{\partial \mathrm{g}(\cdot)}{\partial \widehat{\mathrm{q}}_n(t-d,\boldsymbol{\theta})} z^{-d}}, \tag{20}$$

If $\boldsymbol{\theta}$ is not appropriately chosen, the filter $\mathrm{H}(z)$ can be unstable. The computed gradient $\frac{\partial \widehat{\mathrm{q}}_n(t,\boldsymbol{\theta})}{\partial \theta_i}$ could diverge as t increases. It is proved in Appendix B that, if the root radius[7] $\rho(\mathbf{f})$ of the polynomial $z^r - \sum_{d=1}^{r} f_d z^{r-d}$ is less than 1, the filter $\mathrm{H}(z)$ is stable. Therefore, in the line search step Algorithm 2, the step-size ω is always chosen to be small enough such that the condition $\rho(\mathbf{f}) < 1$ is satisfied (see line 3-5 in Algorithm 2). For further details about the calculation of $\frac{\partial \widehat{\mathrm{q}}_n(t,\boldsymbol{\theta})}{\partial \theta_i}$, see Appendix B.

C. Model Order Selection

Using Algorithm 1, the optimal parameter $\boldsymbol{\theta}$ for a given model order r can be determined. This section discusses how to select the model order. First, a possible range of model orders is estimated by inspecting the correlation between the input and output of the HW model, i.e., $\left(\mathrm{q}^{\mathrm{st}}\right)_{1:\mathrm{T}}$ and $\left(\mathrm{q}^{\mathrm{tv}}\right)_{1:\mathrm{T}}$. Then, the model order is determined in the estimated range using the principle of Minimum Description Length.

The TVSQ at any time depends on the previous viewing experience. In the proposed TVSQ model (18), $\boldsymbol{\phi}_r[t] = \left(\left(\mathrm{q}^{\mathrm{st}}\right)_{t-r:t}^{\mathsf{T}}, \left(\mathrm{q}^{\mathrm{tv}}\right)_{t-r:t-1}^{\mathsf{T}}\right)^{\mathsf{T}}$ has been employed as the model input to capture the previous viewing experience. Thus, identifying the model order r is essentially estimating how much previous viewing experience is relevant to the current TVSQ. In [39], the Lipschitz quotient was proposed to quantify the relevance of $\boldsymbol{\phi}_r$ by

$$\mathrm{Q}^{\mathrm{lip}}(r) = \max_{1 \le t_1 < t_2 \le \mathrm{T}} \left(\frac{\left|\mathrm{q}^{\mathrm{tv}}[t_1] - \mathrm{q}^{\mathrm{tv}}[t_2]\right|}{||\boldsymbol{\phi}_r[t_1] - \boldsymbol{\phi}_r[t_2]||_2} \right). \tag{21}$$

A large $\mathrm{Q}^{\mathrm{lip}}(r)$ implies that a small change in $\boldsymbol{\phi}_r$ could cause a significant change in q^{tv} and thus $\boldsymbol{\phi}_r$ is relevant to TVSQ. Conversely, if $\mathrm{Q}^{\mathrm{lip}}(r)$ is small, the model order r may be larger than necessary. Using $\mathrm{Q}^{\mathrm{lip}}(r)$, the necessary model order can be estimated. In Fig. 10(a), the Lipschitz quotients for different values of r are plotted. It can be seen that, as the model order increases, the corresponding Lipschitz quotient decreases significantly when r is less than 10. This means the viewing experience over the previous 10 seconds is closely related to the TVSQ. Therefore, the model order r should be at least 10.

According to the parameterizations of the HW model in (11), (12), and (13), models of lower order are special cases of the model of higher order. Therefore, in principle, the higher the order, the better performance can be achieved by the model. A large model order, however, may result in over-fitting the model to the training dataset. To select an appropriate order for the HW model, we employed the Minimum Description Length (MDL) criterion, which is widely used in the realm of system identification [36], [40]. The description length of an r-order model is defined in [36] as

$$\mathrm{L}^{\mathrm{des}}(r) = \mathrm{E}(\boldsymbol{\theta}_r^*) \left(1 + (2r+1) \frac{\log(N(\mathrm{T}-r))}{N(\mathrm{T}-r)}\right), \tag{22}$$

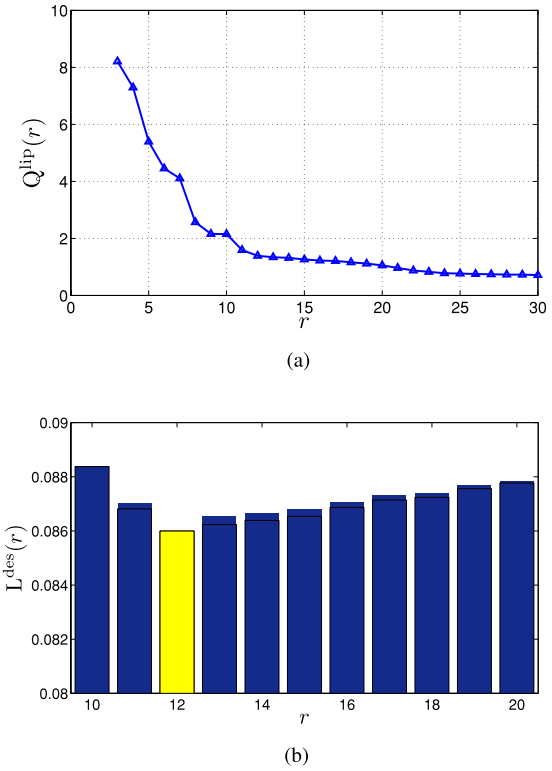


Fig. 10. Model order selection via (a) Lipschitz quotient and (b) description length.

[6] Here $\left(\widehat{\mathrm{q}}_n\right)$ also depends on $\left(\mathrm{q}^{\mathrm{st}}\right)$. But in parameter training, $\left(\mathrm{q}^{\mathrm{st}}\right)$ is treated as a known constant.

[7] The root radius of a polynomial is defined as the maximum radius of its complex roots.

TABLE II
PERFORMANCE OF THE PROPOSED MODEL ON THE DATABASE

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	mean
outage rate(%)	12.15	11.46	9.38	18.06	9.72	4.17	10.76	8.33	8.33	8.33	3.82	7.64	1.74	6.25	0.69	**8.06**
linear correlation	0.868	0.897	0.862	0.785	0.919	0.936	0.859	0.896	0.845	0.863	0.938	0.898	0.892	0.916	0.906	**0.885**
rank correlation	0.881	0.857	0.875	0.814	0.897	0.943	0.872	0.901	0.833	0.859	0.911	0.899	0.870	0.927	0.866	**0.880**

where $\boldsymbol{\theta}_r^*$ is the model parameter of the r-order model determined through Algorithm 1. The first multiplicative term in (22), which is defined in (14) as the outage rate, represents the ability of a model to describe the data. The second multiplicative term increases with the number of parameters $(2r+1)$ and decreases with the size of training set $N(T-r)$. Thus, this term roughly indicate whether the training set is sufficiently large for training a r-order model. The definition of (22) balances the accuracy and the complexity of the model. In Fig. 10(b), the description lengths of the proposed models under different model orders are plotted. It is seen that the minimum description length is achieved at $r = 12$. Therefore, $r = 12$ was selected.

IV. Model Evaluation and Analysis

In this section, the efficiency of the proposed HW model is studied first. Then, four important properties of the proposed model are studied. They are the impact of the initial state, the stability for online TVSQ prediction, the input and output nonlinearities, and the impulse response of the IIR filter. Finally, we analyze the computational complexity of the model for realtime TVSQ prediction.

A. Model Evaluation and Validation

The model parameters were trained using our database via Algorithm 1. Table II list the outage rate of the trained model on all of the 15 test videos. The average outage rate is 8.06%. This means that the model can accurately predict 91.94% of the TVSQs in the database. Furthermore, Table II also list the linear correlation coefficient and the Spearman's rank correlation coefficient between the predicted TVSQ and the measured TVSQ values. The average linear correlation and rank correlation achieved by our model is 0.885 and 0.880, respectively. In Fig. 11, the predicted TVSQs and the 95% confidence interval of the measured TVSQs are plotted. The proposed model effectively tracked the measured TVSQs of the 15 quality-varying videos.

In the proposed method, the TVSQ is estimated by the Hammerstein-Wiener model using the RRED-predicted STSQs of the previous twelve seconds. In Table III, the proposed method is compared with several basic pooling methods, i.e., the maximum, the minimum, the median, and the mean of the RRED-predicted STSQs in the previous twelve seconds. It is seen that the proposed method achieves a significantly lower outage rate and a much stronger correlation with the measured TVSQs.

Table IV shows the performance of the proposed TVSQ prediction method when the STSQ predictor is PSNR, MS-SSIM, and RRED. It may be seen that the RRED-based model outperforms both the MS-SSIM-based model and the PSNR-based model. This can be attributed to the high accuracy of RRED in STSQ prediction. It can also be observed that the performance of the MS-SSIM-based model is close to that of RRED. Since MS-SSIM has lower complexity, it may be attractive as a low-complexity alternative to RRED in the TVSQ prediction model if slightly lower prediction accuracy is acceptable.

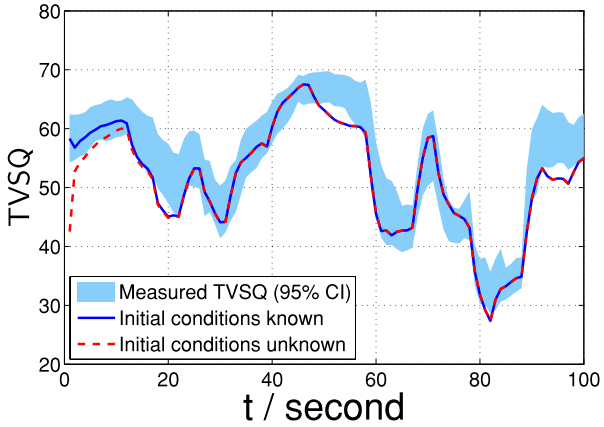


Fig. 12. An illustration of the impact of initial state on predicted TVSQ. Dashed Line: Initial condition $(\mathrm{v})_{1:r}$ is set to be zero. Solid Line: Initial condition is assumed to be known, i.e., $(\mathrm{v})_{1:r} = \left(\mathrm{k}_{\boldsymbol{\gamma}}^{-1}(\mathrm{q}^{\mathrm{tv}})\right)_{1:r}$, where $\left(\mathrm{q}^{\mathrm{tv}}\right)_{1:r}$ is the measured TVSQ in the subjective study.

To rule out the risk of over-fitting the model to the TVSQ database, a leave-one-out cross-validation protocol were employed to check whether the model trained on our database is robust. Each time, the 5 videos corresponding to the same reference video were selected as the validation set and trained the model parameters on the other 10 videos. This procedure was repeated such that all the videos are included once in the validation set. The results are summarized in Table V. Comparing with the models trained on the whole database, the performance of the models in the cross-validation is only slightly degraded. Therefore, the model obtained from our database appears to be robust.

B. Impact of Initial State

As indicated in Section III-A, the initial conditions $\left(\mathrm{v}\right)_{1:r}$ are required to estimate TVSQ. For online video streaming applications, however, $\left(\mathrm{v}\right)_{1:r}$ is unavailable because $\left(\mathrm{v}\right)_{1:r}$ is given by $\left(\mathrm{q}^{\mathrm{tv}}\right)_{1:r}$ and the latter is the TVSQ of the first r seconds of the video. This section studies the impact of the unavailability of the initial conditions.

The transfer function of the linear filter is

$$\mathrm{H}(z) = \frac{\sum_{d=0}^{r} b_d z^{-d}}{1 - \sum_{d=1}^{r} f_d z^{-d}}. \tag{23}$$

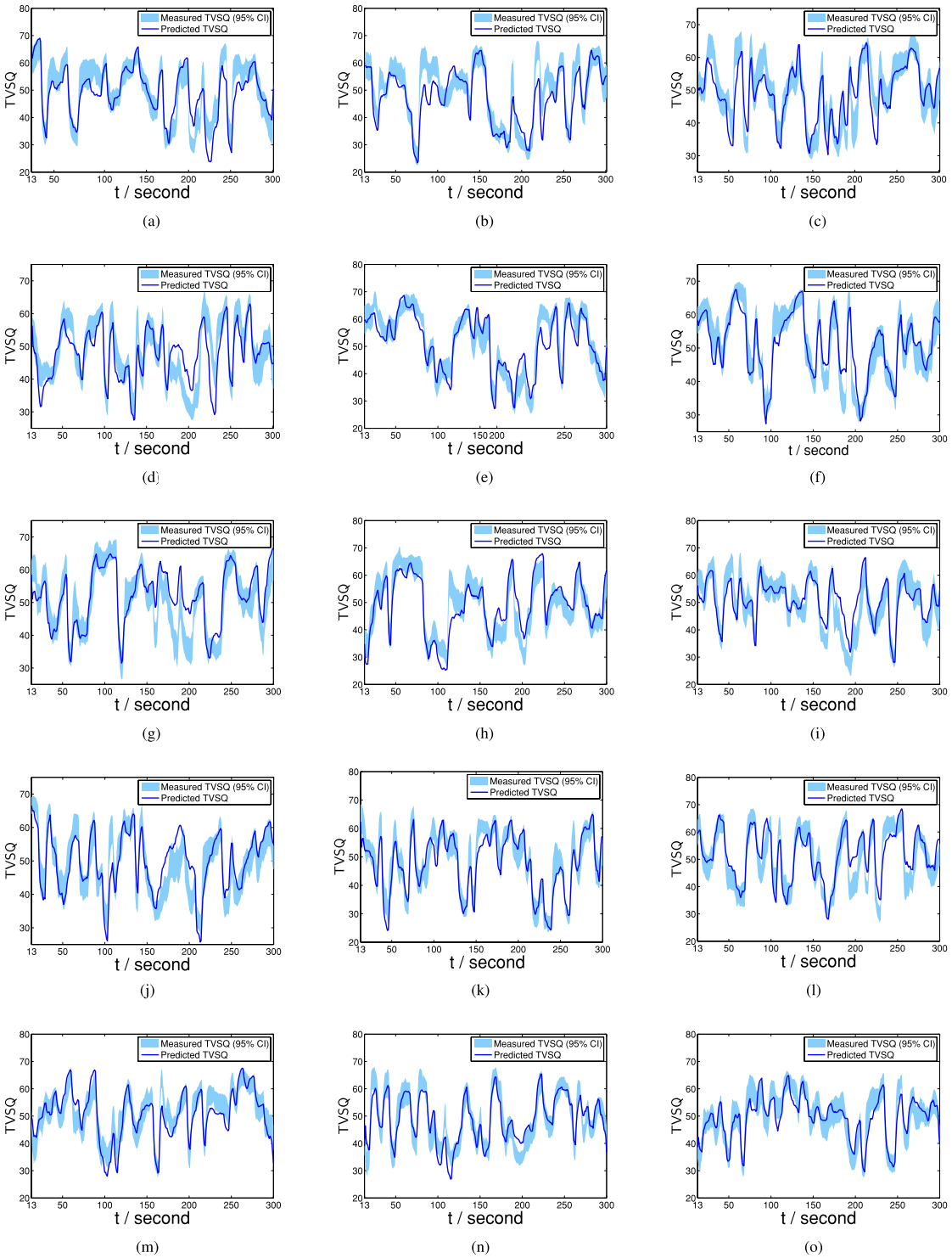


Fig. 11. The predicted TVSQ and the 95% confidence interval (CI) of the TVSQs measured in the subjective study. Since the Hammerstein-Wiener model predicts TVSQs using the STSQs of previous 12 seconds, the plots start from $t = 13$. (a) Video #1. (b) Video #2. (c) Video #3. (d) Video #4. (e) Video #5. (f) Video #6. (g) Video #7. (h) Video #8. (i) Video #9. (j) Video #10. (k) Video #11. (l) Video #12. (m) Video #13. (n) Video #14. (o) Video #15.

According to classical results from system theory, if the root radius of the denominator polynomial $z^r - \sum_{\mathrm{d}=1}^{r} f_d z^{r-d}$, is less than 1, the impact of the initial condition fades to 0 as $t \to \infty$ exponentially fast. Denoting by $\rho(\mathbf{f})$ the root radius of $z^r - \sum_{d=1}^{r} f_d z^{r-d}$, the fading speed is $\rho(\mathbf{f})^t$. Here, we define the quantity $\tau(\mathbf{f}) = -3/\ln \rho(\mathbf{f})$. Over every $\tau(\mathbf{f})$ seconds, the impact of the initial state fades to $e^{-3} \approx 5\%$ of its original level. Therefore, $\tau(\mathbf{f})$ indicates the delay before our TVSQ model starts to track TVSQ. For the model trained on the TVSQ database, $\tau(\mathbf{f}) = 15.1895$ seconds. This means that our model cannot accurately predict the TVSQs of the first 15.1895 seconds of the video. For quality monitoring of

TABLE III
PERFORMANCE COMPARISON WITH DIFFERENT TVSQ POOLING METHODS

	max	min	median	mean	proposed
outage rate(%)	34.26	32.06	26.76	22.22	**8.06**
linear correlation	0.497	0.541	0.589	0.702	**0.885**
rank correlation	0.475	0.515	0.611	0.693	**0.880**

TABLE IV
PERFORMANCE OF THE TVSQ PREDICTION MODEL WITH DIFFERENT STSQ PREDICTORS

STSQ predictors	PSNR	MS-SSIM	RRED
outage rate(%)	21.8	11.5	**8.06**
linear correlation	0.754	0.855	**0.885**
rank correlation	0.744	0.862	**0.880**

long videos, this delay is tolerable. In Fig. 12, the impact of the initial state on one of the quality-varying videos is illustrated. The figure shows the predicted TVSQ when the initial state $(\mathrm{v})_{1:r}$ is simply set to zero. For comparison, the predicted TVSQs when the initial state is assumed to be perfectly known is also shown in the figure. It can be seen that the predicted TVSQs in both cases coincide with each other when $t > 15$ seconds. It also justifies that, for long videos, the impact of the initial condition diminishes over time.

C. Stability for Online TVSQ Prediction

The goal of our TVSQ model is for the online TVSQ prediction. Different from our video database, where each video is 5 minutes long, the videos streamed over HTTP can be much longer. Therefore, it is necessary to check the long-term stability of the proposed model. Specifically, for any quality-varying video, the estimated TVSQ should be bounded within the RDMOS scale of [0, 100]. Since the filter is a linear system, we have $(\mathrm{v})_{1:\mathrm{T}} = (\mathrm{h})_{1:\mathrm{T}} * (\mathrm{u})_{1:\mathrm{T}}$, where $\mathrm{h}[t]$ is the impulse response of the linear filter. It is well-known that $||\mathbf{v}||_\infty = ||\mathbf{h}||_1 ||\mathbf{u}||_\infty$, i.e., that the dynamic range of $\mathrm{v}[t]$ is a dilation of the dynamic range of $\mathrm{v}[t]$. For our TVSQ model, we found that $||\mathbf{h}||_1 \approx 0.3853$. Given that the dynamic range of $\mathrm{q}^{\mathrm{st}}[t]$ is [0, 100], then the dynamic range of $\widehat{\mathrm{q}}[t]$ is found to be [10.2661, 78.9525]. Therefore, the proposed model has bounded output in RDMOS scale.

D. The Input and Output Nonlinearities

In Fig. 13(a), the input nonlinearity of the TVSQ model is plotted. As the input $\mathrm{q}^{\mathrm{st}}[t]$ increases, the gradient of the input nonlinearity diminishes. In particular, the slope of the input nonlinearity is much larger when $\mathrm{q}^{\mathrm{st}}[t] < 50$. As discussed in Section II-A, an RDMOS of 50 indicates acceptable STSQ. Therefore, the concavity of the input non-linearity implies that, the TVSQ is more sensitive to quality variations when viewers are watching low quality videos. This also explains why the TVSQ is more sensitive in region C than region D in Fig. 7 (see section II-E).

In Fig. 13(b), the output nonlinearity of our TVSQ model is plotted. It can be observed that, when $30 \leq \widehat{\mathrm{q}}[t] \leq 70$, the function is almost linearly increasing with the input. This observation inspired us to further simplify the model by replacing the sigmoid output nonlinearity function with a linear function. Table VI shows the performance of the model when the output nonlinearity is replaced by

$$\widehat{\mathrm{q}}[t] = a\mathrm{v}[t] + b, \tag{24}$$

where $a = 0.7013$ and $b = 49.9794$. Comparing with Table II, it can be seen that the outage rate is increased slightly but that the linear correlation coefficients and Spearman's rank correlation coefficients are almost the same. Hence, the simplified model can also predict TVSQ reasonably well. An important advantage of this simplified model is its concavity. Indeed, since the input nonlinearity function is a concave function and the filter is linear, then at any time t, the mapping between $\mathrm{q}^{\mathrm{st}}[t]$ and $\widehat{\mathrm{q}}[t]$ is also concave. Hence, the simplified model can thus be easily incorporated into a convex TVSQ optimization problem, which can be easily solved and analyzed.

E. Impulse Response of the IIR Filter

The impulse response of the IIR filter in the simplified Hammerstein-Wiener model is shown in Fig. 14. Denoting the impulse response by $\mathrm{h}[d]$, we have $\mathrm{v}[t] = \sum_{d=0}^{\infty} \mathrm{h}[d]\mathrm{u}[t-d]$. Thus, $\mathrm{h}[d]$ indicates to what extent the current TVSQ depends on the STSQ of the d seconds prior to the current time. In Fig. 14, it can be seen that $\mathrm{h}[d]$ is maximized at $d = 2$. This means that there is a 2 seconds delay before the viewers respond to a variation in STSQ. That is a natural physiological delay, or latency, between a human subject's observation of STSQ variations and her/his manual response that is given via the human interface. Fig. 14 also shows that $\mathrm{h}[d]$ takes very small values when $d \geq 15$. This implies that the current TVSQ value depends mainly on the STSQs over the immediately preceding 15 seconds. In other words, the visual memory of TVSQ perception on the videos in our database is around 15 seconds. This observation coincides with our analysis that the impact of the initial states of the IIR filter persists for about 15 seconds.

F. Computational Complexity

The proposed model can predict the TVSQ for HTTP-based video streaming in realtime. Since the video source is stored at the server, the STSQs of each second of the video source can be computed off-line before video transmission starts. During video transmission, the proposed Hammerstein-Wiener model can predict the TVSQ in realtime using the pre-computed STSQs.

For example, on a computer with Intel - Xeon X5680 CPU (6-Core, 3.33GHZ) and 12GB RAM, RRED takes 49.9 seconds to predict the STSQ of one second of 720p video. Since RRED is used for offline STSQ prediction, this speed is acceptable. Using the same computer, the proposed Hammerstein-Wiener model takes 9.9×10^{-5} seconds to predict the TVSQ of one second of video. Thus the proposed model is suitable for real-time TVSQ prediction.

TABLE V

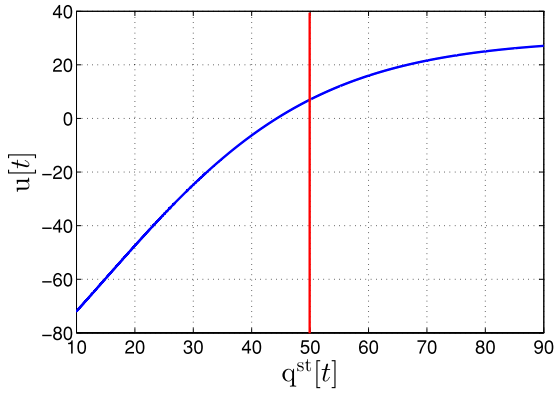
RESULTS OF LEAVE-ONE-OUT CROSS-VALIDATIONS. HERE $\{n_1,\ldots,n_2\}$ DENOTES THE SET OF VIDEO SEQUENCES WITH SEQUENCE NUMBERS FROM n_1 TO n_2. THE PERFORMANCE OF THE MODELS OBTAINED IN CROSS-VALIDATION IS SHOWN IN BOLDFACE. THE PERFORMANCE OF THE MODEL THAT IS TRAINED ON THE WHOLE DATABASE IS ALSO LISTED FOR COMPARISON

validation set	{1,...,5}		{6,...,10}		{11,...,15}	
training set	{1,...,15}	{6,...,15}	{1,...,15}	{1,...,5,11,...,15}	{1,...,15}	{1,...,11}
outage rate (%)	12.154	**13.75**	7.98	**10.00**	4.03	**5.00**
linear correlation	0.866	**0.860**	0.881	**0.875**	0.910	**0.903**
rank correlation	0.864	**0.862**	0.882	**0.879**	0.895	**0.889**

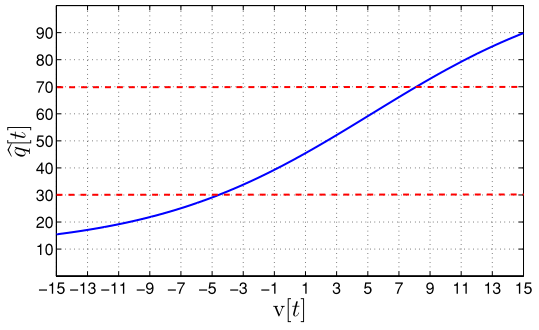
TABLE VI

PERFORMANCE OF THE MODEL IF OUTPUT NONLINEARITY IS REPLACED WITH A LINEAR FUNCTION

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	mean
outage rate(%)	12.00	10.91	10.55	16.00	9.82	5.45	10.18	11.64	10.55	10.18	4.00	10.91	1.45	6.91	1.09	**8.78**
linear correlation	0.840	0.896	0.864	0.787	0.920	0.930	0.869	0.876	0.854	0.842	0.937	0.886	0.883	0.914	0.897	**0.879**
rank correlation	0.866	0.845	0.876	0.818	0.906	0.939	0.883	0.887	0.851	0.840	0.916	0.890	0.853	0.935	0.853	**0.877**



(a)



(b)

Fig. 13. Input and output nonlinearities of the HW model. (a) Input nonlinearity. (b) Output nonlinearity.

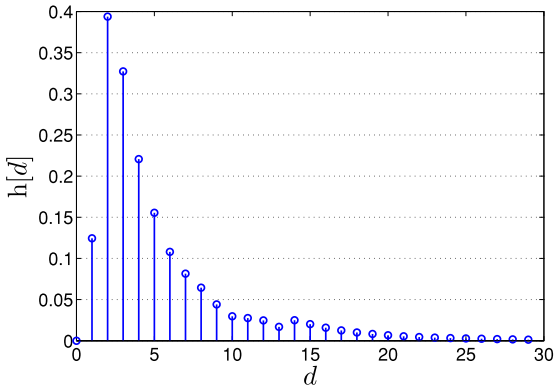


Fig. 14. The impulse response of the IIR filter in the first 30 seconds.

V. CONCLUSION AND FUTURE WORK

In this paper, a TVSQ prediction model is proposed for the rate-adaptive videos transmitted over HTTP. The model was trained and validated on a new database of quality-varying videos, which simulate the true rate-adaptive videos commonly encountered in HTTP-based streaming. Two important conclusions are drawn based on our model. First, the behavioral response of viewers to quality variation is more sensitive in the low quality region than in the high quality region. Second, the current TVSQ can affect the TVSQ in the next 15 seconds. Based on our analysis of the proposed model, the mapping from STSQ and TVSQ is not only monotone but also concave. This property is desirable in solving TVSQ optimization problems.

The proposed TVSQ model can be used to characterize the mapping between video data rate and TVSQ. The rate-adaptation algorithm can then use the rate-TVSQ mapping to select an optimal video data rate that not only avoids playback interruptions but also maximizes the TVSQ.

In this paper, we focus on modeling the impact of quality fluctuations on TVSQ. Of course, frame freezes and re-buffering events caused by playback interruptions can also significantly affect the viewer's QoE. These events, however, are quite distinctive in their source and effect on QoE relative to the types of distortions studied herein. Studying the impact of playback interruptions on TVSQ is an important future work, but is certainly beyond the scope of the work presented here.

APPENDIX A
INSTRUCTIONS FOR SSCQE

You are taking part in a study to assess the quality of videos. You will be shown a video at the center of the monitor and there will be a rating bar at the bottom, which can be controlled by a mouse on the table. You are to provide feedback on

how satisfied you are with your viewing experience up to and including the current moment, i.e., by moving the rating bar in real time based on your satisfaction. The extreme right on the bar is 'excellent' and the extreme left is 'bad'. There is no right or wrong answer.

APPENDIX B
GRADIENT CALCULATION FOR MODEL IDENTIFICATION

For the parameter $\boldsymbol{\gamma}$, we have $\nabla_{\boldsymbol{\gamma}} \mathrm{q}^{\mathrm{tv}}[t] = \left(\frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_1}, \frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_2}, \frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_3}, \frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_4}\right)^{\mathsf{T}}$, where

$$\begin{aligned}
\frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_1} &= \frac{\gamma_4 \mathrm{v}[t] \exp(-(\gamma_1 \mathrm{v}[t] + \gamma_2))}{(1 + \exp(-(\gamma_1 \mathrm{v}[t] + \gamma_2)))^2}, \\
\frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_2} &= \frac{\gamma_4 \exp(-(\gamma_1 \mathrm{v}[t] + \gamma_2))}{(1 + \exp(-(\beta_1 \mathrm{v}[t] + \beta_2)))^2}, \\
\frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_3} &= 1, \\
\frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \gamma_4} &= \frac{1}{1 + \exp\left(-(\gamma_1 \mathrm{v}[t] + \gamma_2)\right)}.
\end{aligned} \tag{25}$$

For the parameter $\mathbf{b}$, $\mathbf{f}$ and $\boldsymbol{\beta}$, we have

$$\begin{aligned}
\nabla_{\boldsymbol{\xi}} \mathrm{q}^{\mathrm{tv}}[t] &= \frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \mathrm{v}[t]} \nabla_{\boldsymbol{\xi}} \mathrm{v}[t] \\
&= \frac{\gamma_1 \gamma_4 \exp(-(\gamma_1 \mathrm{v}[t] + \gamma_2))}{(1 + \exp(-(\gamma_1 \mathrm{v}[t] + \gamma_2)))^2} \nabla_{\boldsymbol{\xi}} \mathrm{v}[t],
\end{aligned} \tag{26}$$

where $\boldsymbol{\xi}$ can be $\mathbf{b}$, $\mathbf{f}$ or $\boldsymbol{\beta}$. Thus we only need to compute $\nabla_{\boldsymbol{\xi}} \mathrm{v}[t]$. For $\mathbf{b}$ and $\mathbf{f}$, we have

$$\begin{aligned}
\nabla_{\mathbf{b}} \mathrm{v}[t] &= (\mathrm{u}[t])_{t-1:t-r} + \sum_{d=1}^{r} f_d \nabla_{\mathbf{b}} \mathrm{v}[t-d] \\
\nabla_{\mathbf{f}} \mathrm{v}[t] &= (\mathrm{v}[t])_{t-1:t-r} + \sum_{d=1}^{r} f_d \nabla_{\mathbf{f}} \mathrm{v}[t-d].
\end{aligned} \tag{27}$$

For $\boldsymbol{\beta}$, we have

$$\nabla_{\boldsymbol{\beta}} \mathrm{v}[t] = \sum_{d=0}^{r} b_d \nabla_{\boldsymbol{\beta}} \mathrm{u}[t-d] + \sum_{d=1}^{r} f_d \nabla_{\boldsymbol{\beta}} \mathrm{v}[t-d], \tag{28}$$

where $\nabla_{\boldsymbol{\beta}} \mathrm{u}[t] = \left(\frac{\partial \mathrm{u}[t]}{\partial \beta_1}, \frac{\partial \mathrm{u}[t]}{\partial \beta_2}, \frac{\partial \mathrm{u}[t]}{\partial \beta_3}, \frac{\partial \mathrm{u}[t]}{\partial \beta_4}\right)^{\mathsf{T}}$ can be computed similarly as (25).

It may be seen from (27) and (28) that $\nabla_{\mathbf{b}} \mathrm{v}[t]$, $\nabla_{\mathbf{f}} \mathrm{v}[t]$ and $\nabla_{\boldsymbol{\beta}} \mathrm{v}[t]$ can be recursively computed. The stability of the recursions can be ensured by the following lemma.

Lemma 1 (Stability of recursive gradient calculation)**.** *If the roots of polynomial $1 - \sum_d^r f_d z^{-d}$ are confined within the unit circle of the complex plane, the recursive gradient calculation is stable.*

Proof: In (27) and (28), the gradients of $\nabla_{\mathbf{b}} \mathrm{v}[t]$, $\nabla_{\mathbf{f}} \mathrm{v}[t]$ and $\nabla_{\boldsymbol{\beta}} \mathrm{v}[t]$ are actually the outputs of IIR filters, where the denominator of the transfer function is $1 - \sum_d^r f_d z^{-d}$. Thus the roots of $1 - \sum_d^r f_d z^{-d}$ determines the stability of the recursive calculation process and the lemma is proved. ∎

To calculate the gradient using (19), we also need to know the initial values of $\left(\widehat{\mathrm{q}}(t, \boldsymbol{\theta})\right)_{1:r}$ to calculate $\left(\frac{\partial \widehat{\mathrm{q}}(t, \boldsymbol{\theta})}{\partial \theta_{\mathrm{i}}}\right)_{1:r}$. For the purpose of model training, we simply set $\left(\widehat{\mathrm{q}}(t, \boldsymbol{\theta})\right)_{1:r} = \left(\mathrm{q}^{\mathrm{tv}}(t, \boldsymbol{\theta})\right)_{1:r}$. Thus, we have $\frac{\partial \widehat{\mathrm{q}}(t, \boldsymbol{\theta})}{\partial \theta_{\mathrm{i}}} = \frac{\partial \mathrm{q}^{\mathrm{tv}}[t]}{\partial \theta_{\mathrm{i}}} = 0, \forall\, t \le r$.

REFERENCES

[1] Microsoft Corporation, Albuquerque, NM, USA. (2009, Sep.). *IIS Smooth Streaming Technical Overview* [Online]. Available: http://www.microsoft.com/en-us/download/default.aspx

[2] R. Pantos and E. W. May. (2011, Mar.). HTTP live streaming. IETF Internet Draft [Online]. Available: http://tools.ietf.org/html/draft-pantos-http-live-streaming-12

[3] Adobe Systems, San Jose, CA, USA. (2012, Mar.). *HTTP Dynamics Streaming* [Online]. Available: http://www.adobe.com/products/hds-dynamic-streaming.html

[4] MPEG Requirements Group, Washington, DC, USA. (2011, Jan.). *ISO/IEC FCD 23001-6 Part 6: Dynamics Adaptive Streaming Over HTTP (DASH)* [Online]. Available: http://mpeg.chiariglione.org/working_documents/mpeg-b/dash/dash-dis.zip

[5] F. Dobrian *et al.*, "Understanding the impact of video quality on user engagement," in *Proc. ACM SIGCOMM Conf.*, 2011, pp. 362–373.

[6] T. Zinner, T. Hoßfeld, T. N. Minhas, and M. Fiedler, "Controlled vs. uncontrolled degradations of QoE—The provisioning-delivery hysteresis in case of video," in *New Dimensions in the Assessment and Support of Quality of Experience (QoE) for Multimedia Applications*, Tampere, Finland: Univ. Tampere, Jun. 2010.

[7] K. Seshadrinathan and A. C. Bovik, "Motion-tuned spatio-temporal quality assessment of natural videos," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 335–350, Feb. 2010.

[8] M. Zink, O. Künzel, J. Schmitt, and R. Steinmetz, "Subjective impression of variations in layer encoded videos," in *Proc. 11th Int. Workshop Quality Service*, 2003, pp. 137–154.

[9] K. Seshadrinathan and A. C. Bovik, "Temporal hysteresis model of time varying subjective video quality," in *Proc. IEEE ICASSP*, May 2011, pp. 1153–1156.

[10] D. E. Pearson, "Viewer response to time-varying video quality," *Proc. SPIE*, vol. 3299, pp. 16–25, Jul. 1998.

[11] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. Conf. Rec. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2. Nov. 2003, pp. 1398–1402.

[12] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Trans. Broadcast.*, vol. 50, no. 3, pp. 312–322, Sep. 2004.

[13] P. V. Vu, C. T. Vu, and D. M. Chandler, "A spatiotemporal most-apparent-distortion model for video quality assessment," in *Proc. IEEE ICIP*, Sep. 2011, pp. 2505–2508.

[14] R. Soundararajan and A. Bovik, "Video quality assessment by reduced reference spatio-temporal entropic differencing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 4, pp. 684–694, Apr. 2013.

[15] K. T. Tan, M. Ghanbari, and D. E. Pearson, "An objective measurement tool for MPEG video quality," *Signal Process.*, vol. 70, no. 3, pp. 279–294, Jul. 1998.

[16] M. A. Masry and S. S. Hemami, "A metric for the continuous quality evaluation of video with severe distortions," *J. Signal Process., Image Commun.*, vol. 19, no. 2, pp. 133–146, Feb. 2004.

[17] M. Barkowsky, B. Eskofier, R. Bitto, J. Bialkowski, and A. Kaup, "Perceptually motivated spatial and temporal integration of pixel based video quality measures," in *Proc. Welcome Mobile Content Quality Exper.*, Mar. 2007, pp. 1–7.

[18] P. Le Callet, C. Viard-Gaudin, and D. Barba, "A convolutional neural network approach for objective video quality assessment," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1316–1327, Sep. 2006.

[19] (2012, Jan.). *Methodology for the Subjective Assessment of the Quality of Television Pictures* [Online]. Available: http://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.500-13-201201-I!!PDF-E.pdf

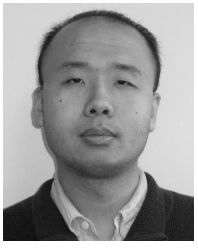
[20] A. Moorthy, K. Seshadrinathan, R. Soundararajan, and A. Bovik, "Wireless video quality assessment: A study of subjective scores and objective algorithms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 587–599, Apr. 2010.

[21] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "A subjective study to evaluate video quality assessment algorithms," *Proc. SPIE*, vol. 7527, pp. 1–10, Feb. 2010.

[22] F. Bellard and M. Niedermayer. (2012). *FFmpeg* [Online]. Available: http://ffmpeg.org/

[23] ITU-T Tutorial, Geneva, The Switzerland. (2004). *Objective Perceptual Assessment of Video Quality: Full Reference Television* [Online]. Available: http://http://www.itu.int/dms_pub/itu-t/opb/tut/T-TUT-OPAVQ-2004-FRT-PDF-E.pdf

[24] Laboratory for Image and Video Engineering, The Univ. Texas at Austin, Austin, TX, USA. (2012). *LIVE Video Quality Database* http://live.ece.utexas.edu/research/quality/live_video.html
[25] A. Goldsmith. *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[26] Center for Perceptual Systems, The Univ. Texas at Austin, Austin, TX, USA. (2012). *The XGL Toolbox* [Online]. Available: http://svi.cps.utexas.edu/software.shtml
[27] H. Liu, N. Klomp, and I. Heynderickx, "A no-reference metric for perceived ringing artifacts in images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 529–539, Apr. 2010.
[28] F. Ribeiro, D. Florencio, and V. Nascimento, "Crowdsourcing subjective image quality evaluation," in *Proc. IEEE 18th ICIP*, Sep. 2011, pp. 3097–3100.
[29] L. Ma, W. Lin, C. Deng, and K. N. Ngan, "Image retargeting quality assessment: A study of subjective scores and objective metrics," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 6, pp. 626–639, Oct. 2012.
[30] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010.
[31] H. Sheikh, M. Sabir, and A. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.
[32] A. M. van Dijk, J. B. Martens, and A. B. Watson, "Quality assessment of coded images using numerical category scaling," *Proc. SPIE*, vol. 2451, pp. 90–101, Feb. 1995.
[33] A. B. Watson and J. A. Solomon, "Model of visual contrast gain control and pattern masking," *J. Opt. Soc. Amer. A*, vol. 14, no. 9, pp. 2379–2391, Sep. 1997.
[34] P. Teo and D. J. Heeger, "Perceptual image distortion," in *Proc. IEEE ICIP*, vol. 2. Nov. 1994, pp. 982–986.
[35] S. Daly, "The visible differences predictor: An algorithm for the assessment of image fidelity," *Digital Images Human Vis.*, vol. 1, pp. 179–206, Jun. 1993.
[36] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ, USA: Prentice-Hall, 1986.
[37] J. A. Nelder, "The fitting of a generalization of the logistic curve," *Biometrics*, vol. 17, no. 1, pp. 89–110, 1961.
[38] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.
[39] X. He and H. Asada, "A new method for identifying orders of input-output models for nonlinear dynamic systems," in *Proc. Amer. Control Conf.*, Jun. 1993, pp. 2520–2523.
[40] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2743–2760, Oct. 1998.



Chao Chen (S'11) received the B.E. and M.S. degrees in electrical engineering from Tsinghua University in 2006 and 2009, respectively. In 2009, he joined the Wireless Systems Innovation Laboratory and the Laboratory for Image and Video engineering, University of Texas at Austin, where he received the Ph.D. degree in 2013. Since 2014, he has been with Qualcomm Incorporated at San Diego. His research interests include visual quality assessment, system identification, and network resource allocation.

Lark Kwon Choi received the B.S. degree in electrical engineering from Korea University, Seoul, Korea, and the M.S. degree in electrical engineering and computer science from Seoul National University, Seoul, in 2002 and 2004, respectively. He was with Korea Telecom as a Senior Engineer from 2004 to 2009 on IPTV platform research and development. He contributed to IPTV standardization in International Telecommunication Union Telecommunication Standardization Sector, Internet Engineering Task Force, and Telecommunications Technology Association.

He is currently pursuing the Ph.D. degree as a member with the Laboratory for Image and Video Engineering and the Wireless Networking and Communications Group, University of Texas at Austin under Dr. Alan C. Bovik's supervision. His research interests include image and video quality assessment, spatial and temporal visual masking, motion perception, and perceptual image and video enhancement.

Gustavo de Veciana (S'88–M'94–SM'01–F'09) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California at Berkeley in 1987, 1990, and 1993, respectively. He is currently the Joe. J. King Professor with the Department of Electrical and Computer Engineering. He served as the Director and Associate Director of the Wireless Networking and Communications Group, University of Texas at Austin, from 2003 to 2007.

His research focuses on the analysis and design of wireless and wireline telecommunication networks, architectures, and protocols to support sensing and pervasive computing, and applied probability and queuing theory. He has served as the Editor for the IEEE TRANSACTIONS ON NETWORKING/*ACM Transactions on Networking*. He was a recipient of the National Science Foundation CAREER Award in 1996, and a co-recipient of the IEEE William McCalla Best ICCAD Paper Award in 2000, the Best Paper in *ACM Transactions on Design Automation of Electronic Systems* from 2002 to 2004, the Best Paper in the International Teletraffic Congress in 2010, and the Best Paper in ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems in 2010. In 2009, he was designated IEEE Fellow for his contributions to the analysis and design of communication networks. He is on the technical advisory board of IMDEA Networks.

Constantine Caramanis (M'06) received the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology in 2006. Since 2006, he has been with the faculty of the Department of Electrical and Computer Engineering, University of Texas at Austin. He received the NSF CAREER Award in 2011. His current research interests include robust and adaptable optimization, machine learning and high-dimensional statistics, with applications to large scale networks, and computer aided design.

Robert W. Heath Jr. (S'96–M'01–SM'06–F'11) is a Cullen Trust endowed Professor at The University of Texas at Austin and is Director of the Wireless Networking and Communications Group. He is also President and CEO of MIMO Wireless Inc. and Chief Innovation Officer at Kuma Signals LLC.

Dr. Heath has been an Editor for the IEEE Transactions on Communication, an Associate Editor for the IEEE Transactions on Vehicular Technology, and lead guest editor for an IEEE Journal on Selected Areas in Communications special issue on limited feedback communication, and lead guest editor for an IEEE Journal on Selected Topics in Signal Processing special issue on Heterogenous Networks. He currently serves on the steering committee for the IEEE Transactions on Wireless Communications. He was a member of the Signal Processing for Communications Technical Committee in the IEEE Signal Processing Society and is a former Chair of the IEEE COMSOC Communications Technical Theory Committee. He was a technical co-chair for the 2007 Fall Vehicular Technology Conference, general chair of the 2008 Communication Theory Workshop, general co-chair, technical co-chair and co-organizer of the 2009 IEEE Signal Processing for Wireless Communications Workshop, local co-organizer for the 2009 IEEE CAMSAP Conference, technical co-chair for the 2010 IEEE International Symposium on Information Theory, the technical chair for the 2011 Asilomar Conference on Signals, Systems, and Computers, general chair for the 2013 Asilomar Conference on Signals, Systems, and Computers, founding general co-chair for the 2013 IEEE GlobalSIP conference, and is technical co-chair for the 2014 IEEE GLOBECOM conference.

Dr. Heath is a co-recipient of the 2010 and 2013 EURASIP Journal on Wireless Communications and Networking best paper awards, the 2012 Signal Processing Magazine best paper award, a 2013 Signal Processing Society best paper award, and the 2014 EURASIP Journal on Advances in Signal Processing best paper award. He was a 2003 Frontiers in Education New Faculty Fellow. He is also a licensed Amateur Radio Operator and is a registered Professional Engineer in Texas.

Alan C. Bovik (F'96) is the Curry/Cullen Trust Endowed Chair Professor with the University of Texas at Austin, where he is the Director of the Laboratory for Image and Video Engineering. He is a Faculty Member with the Department of Electrical and Computer Engineering and the Center for Perceptual Systems, Institute for Neuroscience. His research interests include image and video processing, computational vision, and visual perception. He has authored more than 650 technical articles in these areas and holds two U.S. patents. His several books include the recent companion volumes *The Essential Guides to Image and Video Processing* (Academic Press, 2009).

He was named the SPIE/IS&T Imaging Scientist of the Year in 2011. He has also received a number of major awards from the IEEE Signal Processing Society, including the Best Paper Award in 2009, the Education Award in 2007, the Technical Achievement Award in 2005, and the Meritorious Service Award in 1998. He received the Hocott Award for Distinguished Engineering Research from the University of Texas at Austin, the Distinguished Alumni Award from the University of Illinois at Champaign-Urbana in 2008, the IEEE Third Millennium Medal in 2000, and two journal paper awards from the International Pattern Recognition Society in 1988 and 1993. He is a fellow of the Optical Society of America, the Society of Photo-Optical and Instrumentation Engineers, and the American Institute of Medical and Biomedical Engineering. He has been involved in numerous professional society activities, including Board of Governors of the IEEE Signal Processing Society from 1996 to 1998, a co-founder and Editor-in-Chief of the IEEE TRANSACTIONS ON IMAGE PROCESSING from 1996 to 2002, Editorial Board of the PROCEEDINGS OF THE IEEE from 1998 to 2004, has been a Series Editor for *Image, Video, and Multimedia Processing*, Morgan and Claypool Publishing Company, since 2003, and the Founding General Chairman of the First IEEE International Conference on Image Processing, held in Austin, TX, USA, in 1994.

Dr. Bovik is a registered Professional Engineer in the State of Texas and a frequent consultant to legal, industrial, and academic institutions.