Temporal Video Quality Model Accounting for Variable Frame Delay Distortions

Margaret H. Pinson, Lark Kwon Choi, and Alan Conrad Bovik, *Fellow, IEEE*

***Abstract*—We announce a new Video Quality Model (VQM) that accounts for the perceptual impact of variable frame delays (VFD) in videos with demonstrated top performance on the Laboratory for Image & Video Engineering (LIVE) Mobile Video Quality Assessment (VQA) database. This model, called VQM_VFD, uses perceptual features extracted from spatial-temporal blocks spanning fixed angular extents and a long edge detection filter. VQM_VFD predicts video quality by measuring multiple frame delays using perception based parameters to track subjective quality over time. In the performance analysis of VQM_VFD, we evaluated its efficacy at predicting human opinions of visual quality. A detailed correlation analysis and statistical hypothesis testing show that VQM_VFD accurately predicts human subjective judgments and substantially outperforms top-performing Image Quality Assessment (IQA) and VQA models previously tested on the LIVE Mobile VQA database. VQM_VFD achieved the best performance on the mobile and tablet studies of the LIVE Mobile VQA database for simulated compression, wireless packet-loss, and rate adaptation, but not for temporal dynamics. These results validate the new model and warrant a hard release of the VQM_VFD algorithm. It is freely available for any purpose, commercial, or noncommercial at http://www.its.bldrdoc.gov/vqm/.**

***Index Terms*—Edge detection, video quality model, video quality assessment, variable frame delay, video quality database, VQM_VFD**

I. Introduction

MODERN video transmission systems contain different impairments than those seen two decades ago. Back in the 1990s, video codecs operated with one system delay. Difficult-to-code segments resulted in lower frame rates and more delay; easy-to-code segments resulted in higher frame rates and less delay. These delays always varied around a single system delay. Changes to delay occurred gradually, making them difficult for a naïve viewer to notice.

Today, video transmitted over the internet contains occasional, systematic change to the delay. That is, the system varies around one delay for a while, an event occurs, then the system varies around a different delay, and so on. Example events are rebuffering and decoder buffer overflow / underflow. These changes are often abrupt and easy to perceive (e.g., the video freezes without loss of content).

Manuscript received May 1, 2013.

M. H. Pinson is with the Institute for Telecommunication Sciences (ITS), Boulder, CO, 80305 USA (e-mail: mpinson@its.bldrdoc.gov). ITS is the research and development office of the National Telecommunications & Information Administration (NTIA).

L. K. Choi and A. C. Bovik are with the Laboratory for Image and Video Engineering (LIVE), and the Wireless Networking and Communications Group (WNCG), Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-1084 USA. (e-mail: larkkwonchoi@gmail.com; bovik@ece.utexas.edu).

The Laboratory for Image & Video Engineering (LIVE) Mobile Video Quality Assessment (VQA) database [1] is a tool to investigate this "multiple system delays" problem. It contains a variety of video impairments that are typical of heavily loaded wireless networks, including dynamically varying distortions such as frame freeze and time varying compression rates, as well as static distortions such as compression and wireless packet loss. In August of 2012, LIVE made these video sequences and subjective scores available upon request to researchers. One goal is to encourage development of improved video quality models that are appropriate for mobile video applications.

Objective video quality models are struggling to catch up with the impact of multiple system delays on users' perception of video quality. Most models were designed under the one system delay paradigm. Two examples are Peak Signal to Noise Ratio (PSNR, see the Appendix) and the NTIA General Model, released in 2001 under the name Video Quality Metric (VQM) [2],[3].

In August of 2011, Wolf and Pinson [4] issued a soft release of a new model: the video quality model for variable frame delay (VQM_VFD). This model was designed to accommodate the reality of multiple system delays. Code implementing VQM_VFD is freely available for any purpose, commercial or non-commercial [5]. VQM_VFD was soft released with a small announcement, while independent analyses were being sought.

Another goal of the LIVE Mobile VQA database was to analyze the performance of existing objective video quality models for mobile applications. Moorthy *et al.* [6] analyzed the performance of eleven objective video quality models. Their conclusion was that existing VQA algorithms are not well-equipped to handle distortions that vary over time. This analysis did not include VQM_VFD, as the authors were not aware of each other's work.

We have recently employed the LIVE Mobile VQA database to independently analyze the performance of the VQM_VFD model. The VQA database was not made available to NTIA until after the analyses listed in this report were completed (to ensure impartial analysis). The good performance of VQM_VFD on this database verifies the value of the new model, which thus warrants a hard release.

II. VQM_VFD

A. Background and Design Goals

In 2001, NTIA finalized the General Video Quality Model (VQM) [2], [3]. VQM was trained on 11 datasets, containing a

This article was accepted by and will appear in *IEEE Transactions on Broadcasting* in December 2014.

total of 1,536 subjectively rated video sequences [2]. VQM is one of the first of four models developed for digital video codecs that passed scrutiny when independently examined by the Video Quality Experts Group (VQEG). Of these four models, only VQM showed equally strong performance for both American and European frame rates. VQM gained popularity and is widely used.[1] However, VQM has the following known flaws:

- Training data limited to standard definition television and CIF resolution progressive video
- Few examples of transmission errors in the training data
- Assumes the "single system delay" paradigm

VQM is a reduced reference (RR) metric, meaning low bandwidth features are extracted from the original video and compared to the processed video. For practical reasons, such as the difficulty of getting in-service access to original videos, the software implementations of VQM are full reference (FR). This means that the entire original video and processed video are available at one location. An overview of these and other model types is provided by Wang and Jiang [7].

By 2010, NTIA had access to 83 datasets, containing a total of 11,255 subjectively rated video sequences. These datasets include five image sizes: Quarter Common Intermediate Format (QCIF), Common Intermediate Format (CIF), Video Graphics Array (VGA), Standard Definition (SD), and High Definition (HD). Five combined datasets were created, each with one image size. The Iterative Nested Least Squares Algorithm (INLSA) was used to map the subjective scores onto the nominal (0, 1) common scale [8]. This enabled the combined datasets to be used for developing and testing the output mapping.

NTIA decided to develop a new FR model to replace VQM. The design goals were as follows:

- Include the multi-system delay paradigm
- Allow different viewing distances
- Improve accuracy for transmission error impairments
- 0.90 Pearson Correlation on training data for each of five resolutions: QCIF, CIF, VGA, SD & HD

Like VQM and PSNR, this new FR model requires calibrated video sequences. Calibration algorithms estimate and remove systematic differences between the original and received sequence that do not impact quality:

- A constant spatial shift, horizontally and/or vertically
- A small amounts of spatial scaling (e.g., $\leq 10\%$)
- A constant delay
- A small, constant gain and offset applied to the luma component / Y in the YCbCr colorspace (e.g., $\leq 10\%$)
- A change to the overscan size

NTIA developed two sets of calibration routines that can be used for this purpose. The first are FR calibration routines defined in [2] and [3]. The second are reduced reference (RR) calibration routines defined in [9].

[1] As of the date this article was submitted for publication, Google Scholar finds 655 citations associated with [3]. This does not capture papers that cite VQM with [2], ITU-T Rec. J.144, or ITU-R Rec. BT.1683.

B. VFD: Measurement of Multiple Frame Delays

Digital video transmission systems can produce pauses in the video presentation, after which the video may continue with or without skipping video frames. Sometimes sections of the original video stream may be missing entirely (skipping without pausing).

Time varying delays of the output (or processed) video frames with respect to the input (i.e., the original or reference) video frames present significant challenges for FR video quality measurement systems. Time alignment errors between the output video sequence and the input video sequence can produce measurement errors that greatly exceed the perceptual impact of these time varying video delays.

Wolf [10] describes an algorithm that finds the best matching original frame for each received frame. This variable frame delay (VFD) algorithm does pixel-by-pixel comparisons between each received frame and a range of original video frames. A heuristic algorithm chooses the set of most likely matching frames. The VFD algorithm steps are as follows.

- Normalize each original and processed frame (or field) for zero mean and unit variance.
- Compute mean squared error (MSE) between each processed frame (or field).
- Choose a threshold below which MSE indicates a likely candidate for correct alignment. This threshold is set empirically, based on the range of MSE for the current frame (or field). This produces a fuzzy set of likely alignments for each frame (or field).
- Compute frame (or field) update patterns that are likely and ensure causality. This produces a set of alignment alternatives, some of which may not span the entire duration of the clip.
- Sort these update patterns by length. Compute the most probable alignment pattern for the entire sequence, based upon the assumption that longer update patterns are more likely to be correct than shorter update patterns.
- If the longest update pattern does not span the entire sequence, fill gaps using a multi-stage set of heuristics.

The original video sequence is then modified so it matches the processed video sequence (i.e., VFD-matched original video). For instance, if the received video sequence repeats every other frame, then the original sequence would match this behavior. The VFD information generated from this step, together with the calibrated processed video, and the VFD-matched original video are sent to the objective model. In this way, the objective model predicts quality based on correctly aligned original and distorted frames, and on the estimated annoyance of frame delay variations and frame repetition.

The VFD algorithms act as a pre-filter for two objective video quality models: VQM_VFD and PSNR_VFD.

C. PSNR_VFD

PSNR is probably the most well-known objective video quality model. PSNR is a logical extension of signal-to-noise ratio, which is a long standing electrical engineering measurement. PSNR is on a logarithmic decibel scale, which is not a perceptual scale. PSNR is widely accepted by industry

and has value for that alone.

There are multiple variations of the PSNR in use. The NTIA author is aware that a proprietary implementation of this algorithm calculates and removes the impact of variable frame delays before calculating PSNR. This motivated NTIA to develop a freely available variant, PSNR_VFD.

PSNR_VFD [10] is calculated by comparing the received video with the VFD-matched original video. PSNR is then calculated as:

$$PSNR = 10 \times log_{10}\left(\frac{255^2}{\frac{1}{N}\sum_x\sum_y\sum_t\left(O_{x,y,t}-P_{x,y,t}\right)^2}\right) \quad (1)$$

where

- O is the luma plane of the original video
- P is the luma plane of the received video
- x, y, and t index the video horizontally, vertically, and in time
- N is the total number of pixels used in the calculation

Like most versions of PSNR, this model is very sensitive to calibration errors. PSRN_VFD is intended to be run in three steps: first calibrate the received video, second calculate VFD information, and third calculate PSNR_VFD. In our experiments, PSNR_VFD is run twice: once with the FR calibration routines [2] and once with the RR calibration routines [9].[2]

PSNR_VFD does not capture errors due to temporal misalignments of the video frames, or indeed any artifacts whose perceptual impact is primarily temporal (such as flicker). Instead of measuring overall video quality as perceived by a person, PSNR_VFD isolates one element: the amount of distortion in individual frames.

The goal of PSNR_VFD is to enable subsequent root cause analysis. PSNR_VFD focuses on one aspect of video quality: how well individual frames replicate the original picture. Root cause analysis may provide useful indicators as to *why* the video system is producing the given quality level.

The disadvantage is that PSNR_VFD does not always track subjective opinion, as we will see in Section IV. PSNR_VFD is used by the VQM_VFD model, as one of its parameters.

D. VQM-VFD Filters

A core component of both VQM and VQM_VFD is a spatial information (SI) filter that detects long edges. This filter is similar to the classical Sobel filter in that separate horizontal and vertical filters are applied, then the total edge energy is computed as the Euclidean distance:

$$SI_n(i,j,t) = \sqrt{H_n(i,j,t)^2 + V_n(i,j,t)^2} \quad (2)$$

where the filter size is $(n \times n)$, i is the row, j is the column, t is the time (frame number), H_n is the horizontal bandpass filtered video, and V_n is the vertical bandpass filtered video. Unlike Sobel, each line of the horizontal bandpass filter is identical, and likewise each column of the vertical bandpass filter.

Next, SI_n is separated into HV_n and $\overline{HV_n}$, such that HV_n contains the horizontal-vertical edges (and zero otherwise), and $\overline{HV_n}$ contains the diagonal edges. Low energy edges are omitted.

Filter SI_n assumes that subjects focus on long edges and tend to ignore short edges. As the filter size increases (e.g., SI_5, SI_7, SI_9), individual pixels and small details have a decreasing impact on the edge strength and angle calculation. By contrast, Sobel (3 × 3) responds identically to short and long edges.

The optimal SI_n filter size depends upon the resolution of the target video and, consequently, the length of interesting edges. The filter sizes used by VQM_VFD were chosen empirically, based on the training databases: SI_5 for QCIF resolution video, SI_9 for CIF, SI_{13} for standard definition, and SI_{13} for HD. Naturally there are diminishing returns. SI_{21} showed slightly improved performance over SI_{13} for HD, but the performance difference was too small to justify the slower run speed.

The SI_n, HV_n and $\overline{HV_n}$ filters have potential value for other video or image processing applications. The advantage of SI_n is the ability to detect long edges. HV_n and $\overline{HV_n}$ provide a means to detect a shift of energy from diagonal edges to horizontal & vertical edges (e.g., blocking or tiling) or the opposite (e.g., blurred vertical edges). Here, we have only summarized the filters. Source code is available online at http://www.its.bldrdoc.gov/resources/video-quality-research/guides-and-tutorials/guides-and-tutorials.aspx.

E. VQM_VFD Model Parameters

VQM_VFD computes video quality by comparing the received video sequence to the VFD-matched original video. This new video quality model accounts for the perceptual impact of variable frame delays, by using features extracted from spatial-temporal (ST) blocks spanning a fixed angular extent as seen by the eye. Thus, the ST block sizes change in response to the viewing distance. This enables VQM_VFD to track subjective quality over a wide range of viewing distances and image sizes.

Features and parameters are extracted from ST blocks. Each ST block has a fixed angular extent θ, as seen by the viewer, plus a time extent in seconds. The viewing distance is an input parameter to the model. The ST block size is translated from angular degrees and seconds into pixels and frames using the current viewing distance and video sequence's frame rate. For VQM_VFD, θ is 0.4 degrees. The time extent is 0.2 sec, which is identical to VQM.

A "feature" is a quantity of information associated with, or extracted from, an ST block. A "parameter" is a measure of video distortion that is the result of comparing two parallel streams of features, one stream from the original video and the corresponding stream from the processed video. VQM_VFD contains the following eight parameters. The eight parameters of VQM_VFD are briefly summarized below. The reader is directed to the source code for additional details, including algorithms not given here for clipping functions, thresholds, and weighting.

1) HV_Loss detects a loss in horizontal and vertical spatial edge energy, compared to diagonal edge energy. The

[2] ITU-T Rec. P.340 calculates $PSNR_{const}$ by combining equation (1) with an exhaustive search calibration algorithm.

computation begins by estimating the edge energy in each ST block in both the original and processed video:

$$fHV = mean(HV_n)/mean\left(\overline{HV_n}\right) \quad (3)$$

where *mean* computes the average over the pixels within a particular ST block. A minimum threshold is applied separately to $mean(HV_n)$ and $mean\left(\overline{HV_n}\right)$ to eliminate erratic behavior from imperceptible impairments. The filter adapts in size to the video resolution (e.g., HV_{13} for HD, SD and VGA; HV_9 for CIF; and HV_5 for QCIF).

The differences between original and processed features are computed by estimating the change in HV edge energy:

$$pHVL = min(log10\left(fHV_{orig}/fHV_{proc}\right), 0) \quad (4)$$

where fHV_{orig} is (3) calculated on the original video, fHV_{proc} is (3) calculated on the processed video, and *min* computes minimum. This produces one parameter value per ST block, where decreasing (negative) values of *pHVL* indicate the processed video has lost horizontal & vertical edge energy. The visual masking function in (4) implies that impairment perception is inversely proportionate to the amount of local activity.

The HV_Loss parameter in VQM was oversensitive to impairments for scenes with low and high luma levels and low and high motion levels (i.e., HV_Loss values were too large, so the quality predicted was too low). Thus, VQM_VFD's HV_Loss parameter includes a quadratic weighting function that de-weights ST blocks containing low and high luma levels and/or low and high motion levels. These weighting functions reduce the magnitude of impairments detected in individual ST-blocks. Fig 1 depicts the luma de-weighting function.

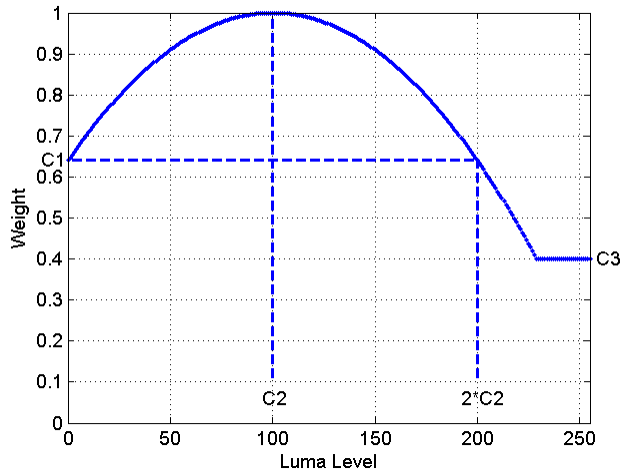


Fig.1. De-weighting function that reduces the HV_Loss parameter ST-blocks for ST-blocks with low and high luma levels. C1=0.64, C2=100, and C3=0.40.

After the de-weighting function, the three-dimensional matrix of parameter values is reduced by a single number by:

$$HV\_Loss = \left[mean_{time}(below5\%_{space}(pHVL_{dw}))\right]^2 \quad (5)$$

where $pHVL_{dw}$ are the de-weighted *pHVL* values, $below5\%_{space}$ computes the average of the 0th through 5th percentile values for all ST blocks associated with the same time segment, and $mean_{time}$ computes the average over time. Put another way, *below5%* detects the areas of the video that contain the greatest loss in HV edge energy. As a final step, a clipping function is applied to eliminate small values. This reduces the parameter's sensitivity to small impairments

2) HV_Gain detects an increase in horizontal and vertical spatial edge energy, compared to diagonal edge energy. Both HV_Loss and HV_Gain can be caused by edge coding noise. The computation is identical to HV_Loss through (4), except that *minimum* is replaced by *maximum*:

$$pHVG = max(log10\left(fHV_{original}/fHV_{processed}\right), 0) \quad (6)$$

where *max* computes the maximum. This produces one parameter value per ST block, where increasing values of *pHVG* indicate the processed video has gained horizontal and vertical edge energy.

The three-dimensional matrix of parameter values is reduced by a single number by:

$$HV\_Gain = rms_{time}(rms_{space}(pHVG)) \quad (7)$$

where rms_{space} computes the root mean square (RMS) for all ST blocks associated with the same time segment, and rms_{time} computes the RMS over time.

3) SI_Loss detects a general decrease in spatial edge energy over time due, for example, to blurring. The computation begins by calculating SI_n feature values for each ST block in both the original and processed video:

$$fSI = stdev(SI_n) \quad (8)$$

where *stdev* computes standard deviation over a particular ST block. A minimum threshold eliminates erratic behavior from imperceptible impairments. The SI_n filter adapts in size to the video resolution as per HV_Loss.

The difference between original and processed video is computed by estimating the loss in SI edge energy:

$$pSIL = min\left[\left(\left(fSI_{proc} - fSI_{orig}\right)/fSI_{proc}\right), 0\right] \quad (9)$$

where fSI_{orig} is (8) calculated on the original video, and fSI_{proc} is (8) calculated on the processed video. This produces one parameter value per ST block, where decreasing (negative) values of *pSIL* indicate the processed video has lost edge energy. The visual masking function in (9) acts similarly to that seen in (4) and (6).

The three-dimensional matrix of parameter values is reduced by a single number by:

$$SI\_Loss = above90\%_{time}(mean_{space}(pSIL)) \quad (10)$$

where $mean_{space}$ computes the average for all ST blocks associated with the same time segment, and $above95\%_{time}$ averages the 90th through 100th percentile values over time. Function $above90\%_{time}$ focuses on the time segments with the worst impairments.

4) SI_Gain detects a general increase in spatial edge energy over time using the same adaptive edge filter. The SI_Gain parameter is sensitive to transient added edges in the picture. SI_Gain uses the same features as SI_loss, from (8), but applies a different visibility threshold. The difference between original and processed video is computed by estimating the gain in SI edge energy:

$$pSIG = max\left[\left(\left(fSI_{proc} - fSI_{orig}\right)/fSI_{proc}\right), 0\right] \quad (11)$$

This produces one parameter value per ST block, where increasing values of *pSIG* indicate the processed video has gained edge energy. The three-dimensional matrix of parameter values is reduced by a single number by:

$$SI\_Loss = rms_{time}(above98\%tail_{space}(pSIG)) \quad (12)$$

where *above98%tail*$_{space}$ computes the difference between two values: (a) the average of the 98th through 100th percentile values over space and (b) the 98th percentile value over space. This measures the spread of the worst quality levels seen in one time segment.

5) TI_Gain computes temporal information (TI) of an ST block by computing the pixel-by-pixel difference between the current frame and the previous frame.

$$fTI = rms(Y(i,j,t) - Y(i,j,t-1)) \quad (13)$$

where *rms* computes RMS over a particular ST block and *Y(i,j,t)* is the luma plane. A minimum threshold on *fTI* eliminates erratic behavior from imperceptible impairments.

The difference between original and processed video is computed by estimating the gain in TI edge energy:

$$pTIG = max\left[log10\left(fTI_{orig}/fTI_{proc}\right), 0\right] \quad (14)$$

where *fTI*$_{orig}$ is (13) calculated on the original video, and *fTI*$_{proc}$ is (13) calculated on the processed video. This produces one parameter value per ST block, where increasing values of *pTIL* indicate the processed video has gained motion energy.

The three-dimensional matrix of parameter values is reduced by a single number by:

$$TI\_Gain = STabove95\%tail(pTIG) \quad (15)$$

where *STabove95%tail* computes difference between two values: (a) the average of the 95th through 100th percentile values and (b) the 95th percentile value. Equation (15) pools all values of *pTIG* into a single ST collapsing function. This measures the spread of the worst quality levels seen over the entire sequence.

Since the original video is VFD-matched to the processed clip, the TI_Gain parameter does not have a large sensitivity to dropped or repeated frames—these are compensated for by the VFD matching process. Rather, the TI_Gain parameter measures added transient distortions in the processed video (such as error blocks) that are not compensated for by the VFD correction. TI_Gain is sensitive to transient-added errors in the picture.

6) RMSE_Gain is a full reference parameter that is computed by comparing pixels within an ST block of the received clip and the VFD-matched original clip.

$$pDiff = Y_{proc}(i,j,t) - Y_{orig}(i,j,t) \quad (16)$$

where Y_{proc} is the luma plane of the processed video, and Y_{orig} is the luma plane of the original video. RMSE_Gain is calculated as follows:

$$RMSE\_Gain = STmean[max(rmse(pDiff), 0)] \quad (17)$$

where *STmean* takes the average over all parameter values in space and time, and *rmse* is root mean square error.

7) VFD_Par1 is extracted from variable frame delay (VFD) information. This temporal distortion parameter is only triggered by delay changes (e.g., received frame *N* aligns to original frame *N*, but received frame *N*+1 aligns to original frame *N*+3). VFD_Par1 is weighted by the duration of the freeze preceding the delay changes (e.g., long freezes are more heavily penalized than many small frame freezes). VFD_Par1 ignores pure frame freezes, for example from a constant reduction to the frame rate, and errs on the side of detecting no impairment when the VFD alignments are ambiguous.

8) VFD_Par1·PSNR_VFD is the product of VFD_Par1 and the full reference metric PSNR_VFD. This parameter is triggered by video clips that contain both temporal distortions impacting the pattern of frames (e.g., pauses and skips detected by VFD_Par1) and spatial distortions impacting individual frames (e.g., fine details detected by PSNR_VFD).

F. VQM_VFD Model Description and Training

The VFD algorithm and VFD_Par1 were developed using a small number of clips known to contain variable frame delays. This training emphasized manual inspection of individual received sequences and VFD delay traces. VFD_Par1 and VFD_Par1·PSNR_VFD were tested on portions of the QCIF, CIF and VGA combined subsets (see [10]).

The remaining parameters were chosen for consistent performance across all five combined datasets, either in isolation or as a complement to the other parameters. The HV_Loss, HV_Gain, SI_Loss and SI_Gain parameters are similar to parameters used in the prior model, VQM, with improvements that appear in the Fast Low Bandwidth Models [11]. Variants of TI_Gain and RMSE_Gain were considered for inclusion in those prior models. The final form for each parameter was determined by calculating numerous variations (e.g., different values for θ and the time extent; see [2] for other examples). The parameter variant and parameter combinations were experimentally determined via searches of the five combined databases using Pearson's correlation coefficient.

A neural network (NN) is used to combine these eight objective video quality parameters. The video sequences from the 83 databases were randomly divided into 70% NN training and 30% NN testing. The MATLAB® NN training tool (nntraintool) was used to train and test the NN.[3]

The eight-parameter input vector is multiplied by an 8 × 8 weighting matrix, which is added to a bias vector, and sent to a hidden layer consisting of eight tan-sigmoid (tansig) neurons. The outputs of these eight tansig neurons are then weighted, summed together with a bias, and sent to a pure-linear (purelin) output neuron. There are thus 72 weights and nine biases in the NN, for a total of 81 free parameters, which are determined in the training phase. A tansig/purelin NN was chosen because of its ability to act as a generalized function approximator (i.e., be similar to nearly any function).

[3] Certain commercial equipment, materials, and/or programs are identified in this report to specify adequately the experimental procedure. In no case does such identification imply recommendation or endorsement by the National Telecommunications and Information Administration, nor does it imply that the program or equipment identified is necessarily the best available for this application.

VQM_VFD achieves a 0.9 Pearson correlation to subjective quality for each of the five subjective datasets.

G. Comments on VQM_VFD

While the VQM_VFD model achieves good performance in predicting subjective ratings, there is always room for improvement. One obvious improvement would be the addition of color distortion parameters.

One possible reason for the difficulty in obtaining a robust color distortion measure that brings added information to the VQM_VFD model might be the lack of independent color distortions in the subject datasets. Distortions that appear in the chroma channels (CB, CR) nearly always also appear in the luma channel (Y).

Another reason might be that some of the color distortions are actually pleasing to the eye (e.g., colors are made more vibrant). Thus, a color distortion metric probably needs to be bipolar, where some distortions produce increases in subjective quality while others produce decreases in subjective quality.

III. Testing On The LIVE Mobile VQA Database

A. Background and Motivation

The Laboratory for Image & Video Engineering (LIVE) at the University of Texas at Austin performs research on the human perception of video and images. LIVE is known for the LIVE image quality database [12], the LIVE video quality database [13], and the LIVE 3D image quality database [14]. These databases are available to the research community free of charge.

The recently-released LIVE Mobile VQA database focuses on video quality distortions typical of a heavily-trafficked wireless network. The goal was to make a dataset available to researchers that aids the development of perceptually optimized VQA algorithms for wireless video transmission on mobile devices and that helps the design of video streaming strategies for video network resource allocation and rate adaptation as a function of time. It is useful for our purposes since it includes systematic simulations of realistic distortion including changes in delay. The dataset contains:

- 720p 30 fps videos
- High quality original video sequences
- A large number of impaired video sequences
- A wide range of quality
- Examples of most common mobile video impairments

Combined, these characteristics were not available from preexisting video databases. This section provides an overview of the LIVE Mobile VQA database. For details, see [6]; and to obtain a copy, see [1].

B. Reference Videos and Distortion Simulation

The LIVE mobile VQA database reference video sequences are 720p (1280 × 720) at 30fps and 15 sec duration. These videos were filmed with the best acquisition quality option (42MB/s). The final scene pool contains 12 videos that depict a variety of content types. Two of these videos were used for training the human subjects, while the rest were used in the actual study.

For each scene, four encoding levels were chosen that show unmistakably different quality levels. The JM reference implementation of H.264 scalable video codec (SVC) [13], [14] was used with fixed Quantization Parameter (QP) encoding. The QP parameter / scene content interaction produces a unique bitrate. The four QP levels are R_1 (highest QP), R_2, R_3 & R_4 (lowest QP). The goal was to ensure perceptual separation of the subjective scores (i.e., perceived quality of R_i < perceived quality of R_{i+1}). This perceptual separation makes it possible for people (and algorithms alike) to produce consistent judgments of visual quality [11], [15]. Because the source video content is quite varied, the resulting bitrates vary between 0.7 Mbps and 6 Mbps.

The LIVE Mobile VQA database consists of 10 reference videos and 200 distorted videos. The distortions simulate most common mobile video impairments as follows:

Compression: This subset contains coding-only impairments R_1, R_2, R_3 and R_4 for each sequence.

Rate adaptation: This subset explores the quality impact of rate changes of different magnitudes (i.e., large or small). The video sequence began with an encoding rate of either R_1, R_2 or R_3 then after 5 seconds switched to the highest rate (R_4), then again after 5 seconds switched back down to the original rate. Three rate adaptations are illustrated in Fig. 2.

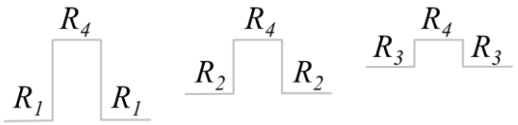


Fig.2. Rate Adaptation: Schematic diagram of the three different rate-switches in a video stream simulated in this study.

Temporal dynamics: This subset was designed to evaluate the effect of multiple rate switches, using five patterns:

pattern 1) multiple rate switches between R_1 and R_4
pattern 2) $R_1 - R_2 - R_4$
pattern 3) $R_1 - R_3 - R_4$
pattern 4) $R_4 - R_2 - R_1$
pattern 5) $R_4 - R_3 - R_1$

These patterns were designed to evaluate two types of switch patterns: abrupt (pattern 1) and smooth (patterns 2 to 5). Each new rate was presented for between 3 and 5 seconds.

Wireless packet loss: The H.264 bitstream (e.g., R_1, R_2, R_3, and R_4) was impaired using a Rayleigh fading channel, which was modeled by an IEEE 802.11 based wireless channel simulator. Bit errors due to attenuation, shadowing, fading and multiuser interference in wireless channels cause spatiotemporal transient distortions which appear as glitches in videos.

Frame-freezes: This subset models two types of frame-freeze impairments:

- Frame-freezes that did not result in the loss of a video segment, to simulate stored video delivery
- Frame-freezes that resulted in a loss of video segments

and lacked temporal continuity, to simulate live video delivery

Three frame-freeze patterns were designed, such that the total duration of all freeze events was held constant:

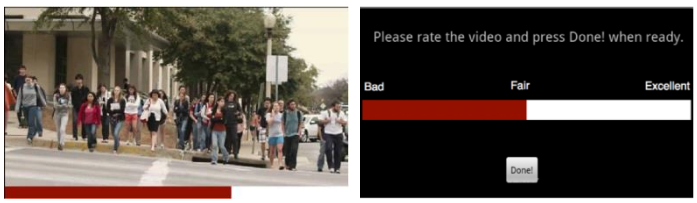
- Eight 1 sec frame-freezes
- Four 2 sec frame-freezes
- Two 4 sec frame-freezes

This subset uses uncompressed video sequences.

C. Test Methodology

Subjects rated the videos using the single-stimulus continuous quality evaluation (SSCQE) method [19] with hidden reference [11], [17], [20]. Subjects watched 200 test videos on a 4" touchscreen Motorola Atrix™ with a resolution of 960 × 540 and 100 different test videos on a 10.1" touchscreen Motorola Xoom with a resolution of 1280 × 800. Because these platforms do not support uncompressed video playback, videos were lightly compressed (> 18Mbps MPEG-4). The experimenters were unable to detect any differences between the visual quality of the uncompressed video files and quality of the compressed video streams. The video files used by objective models do not include this compression, nor do they include the resolution due to the monitor or playback software.

Testing took place at the LIVE subjective testing lab, using software that was specially created for the Android platform to display videos. The subjects rated the videos as a function of time during the playback, yielding continuous temporal quality scores using an uncalibrated bar that spanned the bottom of the screen (see Fig 3a). Subjects also rated the overall quality at the end of each video, using a similar bar (see Fig. 3b).



(a) **(b)**

Fig.3. Subjective study interface: (a) video display and a temporal score rating bar, (b) overall score rating bar.

A total of 36 subjects attended the mobile study, and 17 subjects participated in the tablet study. Most of the subjects were undergraduate students between 22 and 28 years old. Although no vision test was executed, a verbal confirmation of soundness of (corrected) vision was obtained from each subject. Each subject attended two separate sessions. Each session lasted less than 30 minutes, and consisted of the subject viewing 55 videos in randomized order (5 reference and 50 distorted videos). A short training set (6 videos) preceded the study.

Differential Mean Opinion Scores (DMOS) were calculated as the difference between the score that the subject gave the reference video and the score for the distorted video. The overall scores were used to evaluate the Image Quality Assessment and Video Quality Assessment (IQA/VQA) models.

D. Evaluation of Subjective Opinion

This section summarizes trends indicated by the subjective scores. This analysis uses the overall scores.

The design goal of the compression subset was achieved. Subjective opinion of each compression rate (R_i) was statistically better than of the next lower rate (R_{i-1}) for all contents. For example, for the following scenes, the four DMOS values from R_1 to R_4 were:

- "bulldozer with fence" [3.24, 2.09, 1.04, 0.36]
- "two swan dunking" [3.23, 2.55, 1.39, 0.31]

Subjects preferred fewer freezes of long duration to more frequent yet short duration freezes, perhaps because the latter lead to choppy playback. Subjects also preferred not to lose content after a frame-freeze, however that preference was less pronounced. For example, subjects preferred two 4 sec frame-freezes with loss of content over eight 1 sec frame freezes with no loss of content.

A Student's *t*-test on the DMOS results for the rate adaptation and temporal dynamics subsets showed that the time-varying quality of a video had a definite and quantifiable impact. When variations in quality occurred, the opinion scores were influenced by the magnitude, order, and duration of those quality level changes.

The rate adaptation subset analysis indicated that it is preferable to switch from a low rate to a higher rate when the higher rate segment lasts at least half as long as the lower rate. This study only included rate increases that lasted at least 5 seconds, so further study is needed. Nonetheless, this conclusion parallels a speech quality subjective test that analyzed time varying quality in talk-spurts [22]. A change in the lowest rate has a clear impact on visual quality.

The temporal dynamics subset analysis indicated that it is preferable to switch to an intermediate rate before switching to a higher or lower rate (patterns 2 to 5). An abrupt change of bitrate received a statistically significantly lower score (pattern 1).

A comparison between the coding subset and the temporal dynamics subset showed a preference for constant bitrates. For example, R_3 is favored over $R_2 - R_4 - R_2$. This preference is not explained by a weighted sum of the compression-only DMOS scores for rates R_2 and R_4. This behavior could indicate a quality penalty for changing video bitrates, as Voran and Catellier [22] demonstrated can occur when the audio coding bitrate of a talk-spurt is increased.

An analysis of the temporal dynamics subset showed that multiple bitrate switches were preferred over fewer switches. For example, bitrate switches every 3 sec with the pattern $R_1 - R_4 - R_1 - R_4 - R_1$ was preferred over bitrate switches every 5 sec with the pattern $R_1 - R_4 - R_1$; and this preference could not be explained by the 1 sec difference in the duration of the R_4 level. We interpret this to mean that humans perceive multiple changes in quality level as attempts to provide better quality and appear to reward those endeavors.

The overall quality scores were impacted by the quality at the end of the clip. This supports the forgiveness effect theory

proposed by Hands [23].

Regarding the comparison of subjective opinions between the mobile and the tablet study, subjects seemed to be more sensitive to dynamically varying distortions displayed on the tablet device. The higher resolution or larger screen size of the display probably caused those distortions to be more perceptible.

E. Evaluation of Algorithm Performance

The overall performance of various leading FR IQA/VQA algorithms on the LIVE Mobile VQA database indicates that none of the contemporary FR IQA/VQA algorithms are able to predict video quality accurately for the time varying dynamic distortions.

A useful lesson from the correlation coefficient analysis of algorithms is that true multiscale processing (as in VQM_VFD) is recommended to achieve scalability against variations in video resolutions, display sizes, and viewing distance. Another valuable reflection is that the variable frame delay approach is beneficial for the prediction of video quality.

F. Comments on LIVE Mobile VQA Database

The new LIVE Mobile VQA database opens fertile ground for researchers to test and develop perceptually improved VQA algorithms as well as providing analysis of human behavior to support successful video streaming strategies.

Due to limitations of the study session durations, the dataset could not include several other interesting scenarios, such as multiple rate changes between different quality levels, a large number of rate changes, a single change with a high quality segment at the end (e.g., $R_4 - R_1 - R_4$) and so on.

Longer video sequences (e.g., five to thirty minutes) with rate switch simulations to analyze time varying quality would also be beneficial for better understanding human perception of visual quality. We looked at short term effects in this current study. Future work will step towards longer studies including more possible scenarios.

IV. Model Performance Analysis

Moorthy *et al.* [6] analyze the performance of models 1 through 11 in Table I on the LIVE Mobile VQA database. All eleven are FR models. Models 1 to 9 are IQA models, while 10 and 11 are VQA models. This paper extends that work to include FR VQA models 12 to 15.

The intended use of the IQA models is to predict image quality. The IQA scores for each video sequence were calculated by averaging the frame-by-frame scores across time. Since it is not clear how FR IQA algorithms may be used for frame-freeze, we did not include this case in our evaluation. This paper presents the performance of PSNR_VFD [10] and VQM_VFD [4] with two additional calibration options: Reduced Reference (RR) calibration version 2 [9] and Full Reference (FR) calibration [2]. FR calibration is more accurate but does not check for spatial scaling. RR calibration version 2 checked whether or not the codec spatially scales the video. Estimation of spatial scaling can be achieves with RR calibration, but the problem is ill-suited for FR calibration. We chose the "with spatial scaling" option for the RR calibration version 2. Since the version we used (BVQM ver2 [5]) requires input videos in YUV422p format, the YUV420p videos were converted to YUV422p without compression.

TABLE I
List of FR 2D IQA/VQA Algorithms Evaluated

No.	Algorithm
1.	Peak Signal-to-Noise ratio (PSNR), as defined in the Appendix
2.	Structural Similarity Index (SS-SSIM) [21]
3.	Multi-scale Structural Similarity Index (MS-SSIM) [24]
4.	Visual Signal-to-Noise ratio (VSNR) [25]
5.	Visual Information Fidelity (VIF) [26]
6.	Universal Quality Index (UQI) [27]
7.	Noise Quality Measure (NQM) [28]
8.	Signal-to-Noise-ratio (SNR)
9.	Weighted Signal-to-Noise ratio (WSNR) [29]
10.	Video Quality Metric (VQM) [2],[3]
11.	MOtion-based Video Integrity Evaluation (MOVIE) index [30]
12.	Peak Signal-to-Noise ratio with Variable Frame Delay [10] with Reduced Reference calibration version 2 [9] (PSNR_VFD_RR)
13.	Peak Signal-to-Noise ratio with Variable Frame Delay [10] with Full Reference calibration [2] (PSNR_VFD_FR)
14.	Video Quality Model for Variable Frame Delay [4] with Reduced Reference calibration version 2 [9] (VQM_VFD_RR) [5]
15.	Video Quality Model for Variable Frame Delay [4] with Full Reference calibration [2] (VQM_VFD_FR) [5]

A. Correlations Against Subjective Opinion

The wide variety of FR IQA/VQA algorithms listed in Table I were compared using the Spearman Rank Order Correlation Coefficient (SROCC), the Pearson's (Linear) Correlation Coefficient (LCC), and the root mean-squared-error (RMSE). The SROCC measures the monotonicity of the objective algorithm prediction with human scores, while the LCC assesses the prediction accuracy. The LCC and the RMSE were computed after performing a non-linear regression on the objective algorithm scores using a logistic function prescribed in [26].[4] Table II shows the SROCC and LCC for the entire LIVE Mobile VQA database—except for the frame-freeze subset, which, as explained earlier, was omitted from the FR IQA/VQA algorithm analysis.

[4] There were two exceptions. The fitting failed for MOVIE; instead the logistic in [33] was used. There was a discrepancy in the logistic function used for the computation of the LCC for VQM. Here, we use the logistic function defined in [29].

TABLE II
SPEARMAN'S RANK ORDERED CORRELATION COEFFICIENT (SROCC) AND LINEAR (PEARSON'S) CORRELATION COEFFICIENT (LCC) BETWEEN THE ALGORITHM SCORES AND THE DMOS FOR VARIOUS IQA/VQA ALGORITHMS: MOBILE STUDY AND TABLET STUDY

	Mobile Study		Tablet Study	
	SROCC	LCC	SROCC	LCC
PSNR	0.6780	0.6909	0.5886	0.6348
SS-SSIM	0.6498	0.6637	0.4300	0.4893
MS-SSIM	0.7425	0.7077	0.5678	0.6213
VSNR	0.7517	0.7592	0.5929	0.6444
VIF	0.7439	0.7870	0.7261	0.7635
UQI	0.4894	0.6619	0.3642	0.3256
NQM	0.7493	0.7622	0.6614	0.7178
WSNR	0.6267	0.632	0.6255	0.6665
SNR	0.5836	0.5189	0.5474	0.5544
VQM	0.6945	0.6917[4]	0.5552	0.5816[4]
MOVIE	0.6420	0.7157	0.6792	0.7828
PSNR-VFD-RR	0.1109	0.0178	0.0351	0.0939
PSNR-VFD-FR	0.1020	0.0906	0.0030	0.0404
VQM-VFD-RR	0.8301	0.8645	0.8133	0.8110
VQM-VFD-FR	0.8295	0.8631	0.8385	0.8347

TABLE III
ROOT MEAN-SQUARED-ERROR (RMSE) BETWEEN THE ALGORITHM SCORES AND THE DMOS FOR VARIOUS IQA/VQA ALGORITHMS: (A) MOBILE STUDY, (B) TABLET STUDY

	Compression	Rate Adaptation	Temporal Dynamics	Wireless	All
PSNR	0.7069	0.5733	**0.4179**	0.7279	0.6670
SS-SSIM	0.7566	0.6023	**0.4228**	0.7670	0.6901
MS-SSIM	0.7316	0.4792	**0.4199**	0.7160	0.6518
VSNR	0.6021	0.5115	**0.4157**	0.5932	0.6005
VIF	0.5354	0.5078	**0.4572**	0.4945	0.5692
UQI	0.9283	0.6496	**0.4445**	0.7542	0.6916
NQM	0.6374	0.4999	**0.4280**	0.5463	0.5972
WSNR	0.7458	0.5733	**0.4592**	0.7707	0.7150
SNR	0.8654	0.6230	**0.4580**	0.8944	0.7887
VQM	0.7312	0.4840	0.4141	0.7279	0.6663
MOVIE	0.6674	0.4974	**0.4458**	0.7719	0.6444
PSNR-VFD-RR	1.1389	0.6297	**0.4592**	1.1227	0.9225
PSNR-VFD-FR	1.1365	0.6774	**0.4464**	1.1163	0.9188
VQM-VFD-RR	**0.4523**	**0.4029**	**0.4443**	**0.4219**	**0.4638**
VQM-VFD-FR	**0.4469**	**0.4029**	**0.4478**	**0.4207**	**0.4660**

(A)

	Compression	Rate Adaptation	Temporal Dynamics	Wireless	All
PSNR	0.7057	0.5810	**0.2510**	0.7205	0.6630
SS-SSIM	0.8985	0.5855	**0.2585**	0.8538	0.7483
MS-SSIM	0.7896	0.5332	**0.2533**	0.7489	0.6724
VSNR	0.7004	0.5562	**0.2530**	0.7216	0.6562
VIF	**0.5820**	0.5195	**0.2590**	0.5500	**0.5541**
UQI	1.0080	0.6261	**0.2470**	0.8683	0.8113
NQM	0.6477	0.5884	**0.2575**	0.5902	0.5974
WSNR	0.6424	0.4792	**0.2532**	0.7281	0.6397
SNR	0.7741	0.5164	0.2429	0.8349	0.7141
VQM	0.8047	0.5922	**0.2593**	0.7594	0.6980
MOVIE	**0.6224**	0.3855	**0.2593**	0.5087	**0.5342**
PSNR-VFD-RR	1.0961	0.6155	**0.2544**	0.9881	0.8543
PSNR-VFD-FR	1.1038	0.6324	**0.2539**	0.9859	0.8574
VQM-VFD-RR	**0.5750**	**0.3133**	**0.2593**	**0.3205**	**0.5020**
VQM-VFD-FR	**0.5318**	**0.3137**	**0.2452**	**0.3099**	**0.4726**

(B)

Table III tabulates the RMSE between the algorithm scores and DMOS for each distortion subset. For each column of Table III, the bold font highlights the top performing model (i.e., minimum RMSE) and all statistically equivalent models. RMSE is used to compare model performance on different subsets, because these RMSE values can be directly compared to each other. Pinson *et al.* [31] demonstrate how LCC drops as the range of quality narrows.

VQM_VFD showed the best performance for the entire LIVE Mobile VQA database in both the mobile and tablet studies. Since FR and RR calibration options showed almost no determinant differences on high correlation coefficients, we analyzed the performance across calibrations. The tables indicate that the new VQM_VFD model takes into account time varying video delays, and thus is a notable improvement on the previous VQM model. VQM_VFD also outperforms the two top performing models from [6], VSNR and VIF, which are true wavelet decomposition based IQA algorithms. VQM_VFD achieved 0.8301 (SROCC) and 0.8645 (LCC) in the mobile study and 0.8385 (SROCC) and 0.8347 (LCC) in the tablet study. These results imply that VQM_VFD is quite well correlated with human opinion and properly accounts for the importance of modeling variable frame delays in perceptual VQA.

VQM_VFD was either the top performing model or statistically equivalent to the top performing model for each data subset. Looking horizontally across Table III, notice that the RMSE values for the temporal dynamics subset are similar to those received by the other subsets.

Nonetheless, the temporal dynamics subset identifies a limitation of VQM_VFD. Fig. 4 shows a scatter plot between the VQM_VFD model with FR calibration and DMOS for the mobile and tablet studies. Notice that the VQM_VFD scores for the temporal dynamics subset are nearly identical (see the magenta diamonds in Fig. 4). The VQM_VFD time collapsing functions do not take the order of events into account (e.g., average the 10% of ST blocks containing the largest impairment values). Almost all other algorithms exhibit a similar behavior.

This demonstrates that there remains much work to be done on VQA algorithms to enable them to better handle temporal distortions. The implication is that these models should take into consideration the order of events. This might have a systematic impact on IQA/VQA models, because a scene's coding complexity can change over time.

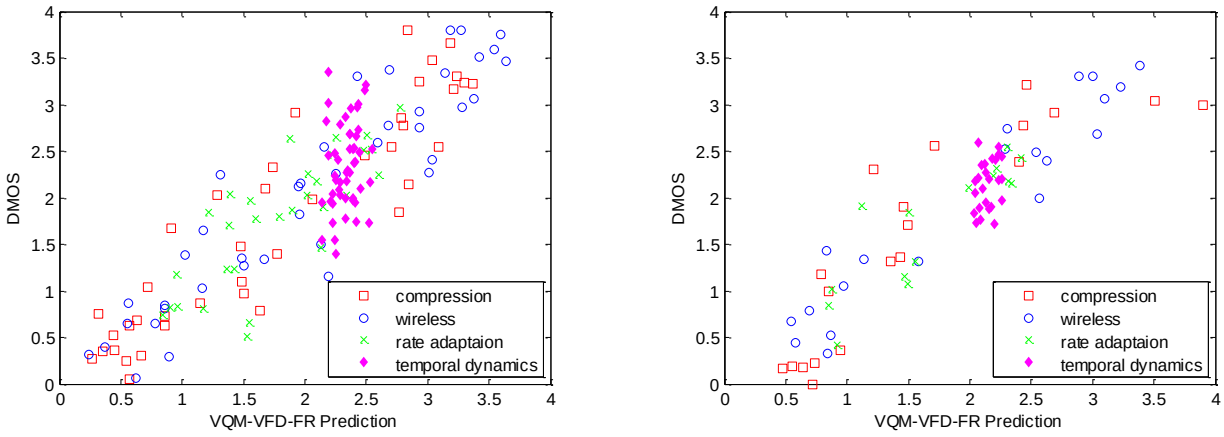


Fig. 4. Scatter plots of VQM-VFD prediction: (a) Mobile study. (b) Tablet study. Each square, circle, cross, or diamond marker indicates compression, wireless, rate adaptation, and temporal dynamics distortion, respectively.

TABLE IV
MOBILE STUDY: STATISTICAL ANALYSIS OF ALGORITHM PERFORMANCE. WITHIN EACH ELEMENT THE MATRIX, THE SYMBOLS CORRESPOND TO [COMPRESSION, RATE ADAPTATION, TEMPORAL DYNAMICS, WIRELESS, AND ALL]

	PSNR	SS-SSIM	MS-SIM	VSNR	VIF	UQI	NQM	WSNR	SNR	VQM	MOVIE	PSNR-VFD-RR	PSNR-VFD-FR	VQM-VFD-RR	VQM-VFD-FR
PSNR	- - - - -	- - - 1 1	- 1 - - 1	- - - - -	0 - 1 0 0	1 - - 1 -	- - - 0 0	1 1 1 1 1	1 1 - 1 1	1 - - - -	1 - 1 1 1	1 1 1 1 1	1 1 1 1 1	0 - 1 0 0	0 - 1 0 0
SS-SSIM	- - - 0 0	- - - - -	- 1 - 0 -	- - - 0 0	0 - - - 0	- - - - -	0 - - 0 0	- 1 - - -	1 1 - - 1	- - - 0 0	- - 1 - -	1 1 1 - 1	1 1 1 - 1	0 - 1 0 0	0 - 1 0 0
MS-SSIM	- 0 - - 0	- 0 - 1 -	- - - - -	- 0 - - 0	0 0 - 0 0	- 0 - - -	0 0 - 0 0	- - - - 1	1 - - 1 1	- 0 - - -	- 0 - - -	1 - - 1 1	1 - - 1 1	0 0 - 0 0	0 0 - 0 0
VSNR	- - - - -	- - - 1 1	- 1 - - 1	- - - - -	0 - 1 0 0	1 - - 1 1	- - - 0 -	1 1 1 1 1	1 1 - 1 1	1 0 - - -	1 - 1 1 1	1 - 1 1 1	1 - 1 1 1	1 1 1 1 1	0 - 1 0 0	0 - 1 0 0
VIF	1 - 0 1 1	1 - - 1 1	1 1 - 1 1	1 - 0 1 1	- - - - -	1 - - 1 1	1 - - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 - - 1 1	1 1 - 1 1	1 1 - 1 1	- - - 1 -	- - - 1 -
UQI	0 - - 0 -	- - - - -	- 1 - - -	0 - - 0 0	0 - - 0 0	- - - - -	0 - - 0 0	- 1 - - 1	- - - - 1	- 0 - - -	- - - - -	1 - - 1 1	1 - - 1 1	0 - - 0 0	0 0 - 0 0
NQM	- - - 1 1	1 - - 1 1	1 1 - 1 1	- - - 1 -	0 - - 0 0	1 - - 1 1	- - - - -	1 - - 1 1	1 - - 1 1	1 0 - 1 1	1 - - 1 1	1 - - 1 1	1 - - 1 1	0 0 - - 0	0 0 - - 0
WSNR	0 0 0 0 0	- 0 - - -	- - - - 0	0 0 0 0 0	0 0 - 0 0	- 0 - - 0	0 - - 0 0	- - - - -	- - - - -	- 0 - - 0	- 0 - - 0	- - - - 1	1 - - - 1	0 0 - 0 0	0 0 - 0 0
SNR	0 0 - 0 0	0 0 - - 0	0 - - 0 0	0 0 - 0 0	0 0 - 0 0	- - - - 0	0 - - 0 0	- - - - -	- - - - -	- 0 - 0 0	- 0 1 - 0	- - 1 - 1	- - 1 - 1	0 0 1 0 0	0 0 1 0 0
VQM	0 - - - -	- - - 1 1	- 1 - - -	0 1 - - -	0 - - 0 0	- 1 - - -	0 1 - 0 0	- 1 - - 1	- 1 - 1 1	- - - - -	- - 1 - -	1 1 1 1 1	1 1 1 1 1	0 - 1 0 0	0 - 1 0 0
MOVIE	0 - 0 0 0	- - 0 - -	- 1 - - -	0 - 0 0 0	0 - - 0 0	- - - - -	0 - - 0 0	- 1 - - 1	- 1 0 - 1	- - 0 - -	- - - - -	1 1 - - 1	1 1 - - 1	0 - - 0 0	0 - - 0 0
PSNR-VFD-RR	0 0 0 0 0	0 0 0 - 0	0 - - 0 0	0 - 0 0 0	0 0 - 0 0	0 - - 0 0	0 - - 0 0	- - - - 0	- - 0 - 0	0 0 0 0 0	0 0 - - 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
PSNR-VFD-FR	0 0 0 0 0	0 0 0 - 0	0 - - 0 0	0 0 0 0 0	0 0 - 0 0	0 - - 0 0	0 - - 0 0	0 - - - 0	- - 0 - 0	0 0 0 0 0	0 0 - - 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
VQM-VFD-RR	1 - 0 1 1	1 - 0 1 1	1 1 - 1 1	1 - 0 1 1	- - - 0 -	1 - - 1 1	1 1 - - 1	1 1 - 1 1	1 1 0 1 1	1 - 0 1 1	1 - - 1 1	1 1 - 1 1	1 1 - 1 1	- - - - -	- - - - -
VQM-VFD-FR	1 - 0 1 1	1 - 0 1 1	1 1 - 1 1	1 - 0 1 1	- - - 0 -	1 1 - 1 1	1 1 - - 1	1 1 - 1 1	1 1 0 1 1	1 - 0 1 1	1 - - 1 1	1 1 - 1 1	1 1 - 1 1	- - - - -	- - - - -

TABLE V
TABLET STUDY: STATISTICAL ANALYSIS OF ALGORITHM PERFORMANCE. WITHIN EACH ELEMENT THE MATRIX, THE SYMBOLS CORRESPOND TO [COMPRESSION, RATE ADAPTATION, TEMPORAL DYNAMICS, WIRELESS, AND ALL]

	PSNR	SS-SSIM	MS-SIM	VSNR	VIF	UQI	NQM	WSNR	SNR	VQM	MOVIE	PSNR-VFD-RR	PSNR-VFD-FR	VQM-VFD-RR	VQM-VFD-FR
PSNR	- - - - -	1 0 - - 1	1 0 - - 1	- 0 - - -	0 0 - 0 0	1 0 - 1 1	- 0 - 0 -	- 0 - - -	- 0 0 - -	1 0 - - 1	- 0 - 0 -	- 0 - - 1	- 0 - - 1	0 0 - 0 0	0 0 - 0 0
SS-SSIM	0 1 - - 0	- - - - -	- - - - -	0 - - - 0	0 0 - 0 0	- - - - -	0 - - 0 0	0 - - 0 0	0 - 0 - -	- - - - -	- - - 0 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
MS-SSIM	0 1 - - 0	- - - - -	- - - - -	0 - - - -	0 0 - 0 0	- - - - 1	0 - - 0 -	0 - - - 0	- - 0 - -	- - - - -	- 0 - 0 0	- - - - 1	- - - - 1	0 0 - 0 0	0 0 - 0 0
VSNR	- 1 - - -	1 - - - 1	1 - - - -	- - - - -	0 0 - 0 0	1 - - 1 1	- - - 0 -	- - - - -	- - - - -	1 - - - 1	- 0 - 0 -	- - - - 1	- - - - 1	0 0 - 0 0	0 0 - 0 0
VIF	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	- - - - -	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 - 0 1 1	1 1 - 1 1	1 - - 1 1	1 1 - 1 1	1 1 - 1 1	1 - - 0 1	1 - - 0 -
UQI	0 1 - 0 0	- - - - -	- - - - 0	0 - - 0 0	0 0 - 0 0	- - - - -	0 - - 0 0	0 - - 0 0	0 0 0 - 0	- - - - -	- 0 - 0 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
NQM	- 1 - 1 -	1 - - 1 1	1 - - 1 -	- - - 1 -	0 0 - 0 0	1 - - 1 1	- - - - -	- - - - -	- 0 - 1 -	1 - - 1 1	- 0 - - -	- 0 - 1 1	- - - 1 1	0 0 - 0 0	0 0 - 0 0
WSNR	- 1 - - -	1 - - 1 1	1 - - - 1	- - - - -	0 0 - 0 0	1 - - 1 1	- - - - -	- - - - -	- - - 1 -	1 - - 1 1	- - - 0 -	- - - - 1	- - - - 1	0 0 - 0 0	0 0 - 0 0
SNR	- 1 1 - -	1 - 1 - -	- - 1 - -	- - - - -	0 - 1 0 0	1 1 1 - 1	- 1 - 0 -	- - - 0 -	- - - - -	1 - 1 - -	- - 1 0 0	- - 1 - 1	- - 1 - 1	0 0 1 0 0	0 0 1 0 0
VQM	0 1 - - 0	- - - - -	- - - - -	0 - - - 0	0 0 - 0 0	- - - - -	0 - - 0 0	0 - - 0 0	0 - 0 - -	- - - - -	- 0 - 0 0	- - - - 1	- - - - 1	0 0 - 0 0	0 0 - 0 0
MOVIE	- 1 - 1 -	- - - 1 1	- 1 - 1 1	- 1 - 1 -	0 - - 0 0	- 1 - 1 1	- 1 - - -	- - - 1 -	- - 0 1 1	- 1 - 1 1	- - - - -	- - - 1 1	- - - 1 1	0 0 - 0 0	0 0 - 0 0
PSNR-VFD-RR	- 1 - - 0	- - - - -	- - - - 0	- - - - 0	0 0 - 0 0	- - - - -	- 1 - 0 0	- - - - 0	- - 0 - 0	- - - - 0	- - - 0 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
PSNR-VFD-FR	- 1 - - 0	- - - - -	- - - - 0	- - - - 0	0 0 - 0 0	- - - - -	- - - 0 0	- - - - 0	- - 0 - 0	- - - - 0	- - - 0 0	- - - - -	- - - - -	0 0 - 0 0	0 0 - 0 0
VQM-VFD-RR	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	0 - - 1 0	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 0 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	- - - - -	- - - - -
VQM-VFD-FR	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	0 - - 1 -	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 0 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	1 1 - 1 1	- - - - -	- - - - -

Among the tested objective IQA/VQA models, PSNR_VFD showed the worst result. PSNR_VFD focuses on one aspect of video quality: how well individual frames replicate the original picture. PSNR_VFD failed to predict subjective human opinion partly because it does not impose any penalties for dropped or repeated frames and variable video delays.

Note PSRN_VFD's extraordinarily large RMSE values for the Compression subset. This suggests that H.264 SVC makes significant changes to individual frames that people either do not notice or do not find objectionable.

B. Hypothesis Testing and Statistical Analysis

1) Inter-Algorithm Comparison: We executed a statistical analysis of the algorithm scores using the F-statistics as in [13] and [32] to evaluate whether the correlations of PSNR_VFD and VQM_VFD were significantly different from other algorithms. Specifically, the F-statistic was used to evaluate the variance of the residuals produced after a non-linear mapping between the two algorithms being compared. Tables IV and V list the results of this analysis for each distortion category and across all distortions for the mobile and the tablet studies, respectively. A value of '1' in the tables indicates that the row (algorithm) is statistically superior to the column (algorithm), while a value of '0' indicates that the row is worse than a column; a value of '–' indicates that the row and column are statistically indistinguishable. Within each entry of the matrix, the first four symbols correspond to the four distortions (ordered as in Section III.B: compression, rate adaptation, temporal dynamics, and wireless), while the last symbol represents significance across the entire database.

Tables IV and V indicate that VQM_VFD significantly outperforms other models. Only VIF is competitive with VQM_VFD for the entire database in the hypothesis test. This tells us that true multiscale and variable frame delay algorithms like VQM_VFD can improve the performance of objective VQA models for mobile video applications.

2) Comparison with the Theoretical Optimal Model: Seshadrinathan *et al.* [13] and Sheikh *et al.* [32] propose an alternate method for evaluating the accuracy of an objective video quality model. This technique is built on the premise that DMOS is an estimate of the underlying true mean of the entire population; and that an objective model should track this underlying true mean. The optimal (e.g., null) objective model displays this behavior.

Objective models estimate mean opinion score (MOS). Using the subjective data, we can calculate MOS and the variance of individual subjective ratings around this mean. Similarly, we can calculate the variance of individual subjective ratings round each estimated objective model value. An F-test will tell us if the latter variance is significantly greater than the former. If the two variances are statistically equivalent, then the model is equivalent to the optimal objective model.

The variance between the Differential Opinion Scores (DOS) and the DMOS is a measure of the inherent variance of subjective opinion (σ^2_{null}). This is compared to the variance between the DOS and the algorithm scores ($\sigma^2_{algorithm}$). The ratio of the two variables, $\sigma^2_{algorithm} / \sigma^2_{null}$ is evaluated with the F-statistic. A threshold F-ratio can be determined based on the degrees of freedom exhibited by the numerator and denominator at the 95% confidence level. If the F-statistic is larger than the threshold, the algorithm performance is statistically equivalent to the theoretical optimal model.

Table VI indicates that VQM_VFD is equivalent to the theoretical optimal model, when compared to the compression and the wireless subsets. However, none of the algorithms are equivalent to the optimal model when the entire database is considered. Obviously, despite the significant progress of VQM_VFD, there remains considerable opportunity to improve the performance of VQA algorithms and corresponding subjective human opinions.

TABLE VI
ALGORITHM PERFORMANCE VS. THE THEORETICAL OPTIMAL MODEL FOR (A) MOBILE STUDY, (B) TABLET STUDY. BOLD FONT INDICATES STATISTICAL EQUIVALENCE TO THE THEORETICAL OPTIMAL MODEL.

(A)	Compression	Rate Adaptation	Temporal Dynamics	Wireless	All
PSNR	0.8331	0.1365	0.0342	0.8391	0.3821
SS-SSIM	0.757	0.0212	0.0302	0.7722	0.3526
MS-SSIM	0.7959	0.2384	0.0327	0.8589	0.401
VSNR	0.9764	0.2054	0.036	1.0432	0.4614
VIF	1.0555	0.2094	0.0022	**1.1661**	0.4959
UQI	0.4549	0.0407	0.0128	0.7934	0.3507
NQM	0.7845	0.2172	0.0262	1.1043	0.4651
WSNR	0.7739	0.1365	0.0004	0.7658	0.3197
SNR	0.5727	0.0755	0.0014	0.5297	0.2156
VQM	0.7966	0.2337	0.037	0.8392	0.383
MOVIE	0.8897	0.2201	0.0117	0.7635	0.41
PSNR-VFD-RR	0.0002	0.0668	0.0004	0.0021	0.0003
PSNR-VFD-FR	0.0058	0.0027	0.0113	0.0184	0.0066
VQM-VFD-RR	**1.1413**	0.3075	0.013	**1.2424**	0.5982
VQM-VFD-FR	**1.1463**	0.3075	0.0101	**1.2431**	0.5963
Threshold F-ratio	1.139	1.1622	1.1234	1.139	1.0672

(B)	Compression	Rate Adaptation	Temporal Dynamics	Wireless	All
PSNR	0.9773	0.0859	0.0043	0.6947	0.2932
SS-SSIM	0.5638	0.0802	0.0005	0.4514	0.1743
MS-SSIM	0.8095	0.1434	0.0031	0.6463	0.2809
VSNR	0.9873	0.1163	0.0033	0.693	0.3022
VIF	1.1904	0.1589	0.0002	0.9459	0.4242
UQI	0.2844	0.0271	0.0063	0.4224	0.0771
NQM	1.0823	0.0766	0.0009	0.8928	0.3749
WSNR	1.0915	0.2023	0.0032	0.682	0.3233
SNR	0.8421	0.1623	0.0084	0.4884	0.2237
VQM	0.7773	0.0717	0.0000	0.628	0.2462
MOVIE	1.1253	0.2897	0.0000	0.9966	0.426
PSNR-VFD-RR	0.0365	0.0414	0.0025	0.1645	0.0064
PSNR-VFD-FR	0.014	0.0186	0.0027	0.1696	0.0012
VQM-VFD-RR	**1.2013**	0.3441	0.0000	1.1776	0.4786
VQM-VFD-FR	**1.2652**	0.3439	0.0057	1.1853	0.507
Threshold F-ratio	1.1956	1.2292	1.1732	1.1956	1.0831

V. CONCLUSION

We introduced a new video quality model (VQM_VFD) that is able to handle variable frame delays, and successfully captures multiple system delays of the processed video frames with respect to the reference video frames to track subjective quality. The performance of the VQM_VFD was evaluated on the recently-released LIVE Mobile VQA database, which

encompasses a wide variety of distortions, including dynamically-varying distortions as well as uniform compression and wireless packet loss. This confirms that variable frame delays have a definite impact on human subjective judgments of visual quality and that VQM_VFD significantly contributes to the progress of VQA algorithms. Based on non-optimized code, VQM_VFD takes five times as long to compute as PSNR.

Although VQM_VFD performed better than existing top-performing IQA/VQA models tested on the LIVE Mobile VQA database, there remains significant room for improvement. The temporal dynamics subset indicates that human subjective opinion is influenced by the time ordering of quality events within short video clips. Understanding the reactions of humans to time varying behavior and temporal dynamics may prove helpful in the design of future improved objective VQA algorithms that are appropriate for mobile video applications.

VSNR and VIF were the best performing IQA models. These image quality models were applied to rate video quality instead, by performing frame averages over time. The accuracy of these models implies that there is merit to the idea of an IQA model as the basis of a VQA model. The performance differential between VQM and VFD_VQM on the LIVE Mobile VQA database indicates that such IQA based VQA models could benefit by integrating the VFD algorithm [10]. Such integration would require separate training, which is beyond the scope of this paper. Note that the VFD algorithm and SI_n long edge detection filter can be used for any purpose, commercial or non-commercial.

In this article, we only summarized the portion of the LIVE Mobile database relevant to evaluating PSNR_VFD and VQM_VFD using a performance analysis mirroring the one that Moorthy *et al*. did in [6]. The reader is referred to [6] for a detailed description of the study including the evaluation of temporal quality scores.

APPENDIX

Here, PSNR was calculated using the MeTriX MuX Visual Quality Assessment Package from Cornell University [34]. PSNR is calculated as follows:

$$PSNR = \frac{1}{T}\sum_t 10 \times log_{10}\left(\frac{255^2}{\frac{1}{N}\sum_x\sum_y\left(O_{x,y,t}-P_{x,y,t}\right)^2}\right) \qquad (2)$$

where

- O is the luma plane of the original video
- P is the luma plane of the processed video
- x, y and t index the video horizontally, vertically and in time
- N is the number of pixels in each image
- T is the number of frames

ACKNOWLEDGMENT

M. H. Pinson thanks Stephen Wolf for his development of the VQM_VFD model, and Anush Krishna Moorthy for his work developing the LIVE Mobile VQA database.

REFERENCES

[1] http://live.ece.utexas.edu/research/quality/live_mobile_video.html
[2] Stephen Wolf, Margaret Pinson, "Video Quality Measurement Techniques", *NTIA Technical Report TR-02-392*, Jun. 2002.
[3] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol.50, no.3, Sep. 2004, pp. 312-322.
[4] S. Wolf and M. H. Pinson, "Video Quality Model for Variable Frame Delay (VQM_VFD)", *NTIA Technical Memo* TM-11-482, Sep. 2011.
[5] http://www.its.bldrdoc.gov/vqm/
[6] A. K. Moorthy, L. K. Choi, A. C. Bovik, and G. de Veciana, "Video Quality Assessment on Mobile Devices: Subjective, Behavioral and Objective Studies," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6 no. 6, Oct. 2012, pp. 652-671.
[7] C. Wang and X. Jiang, "Video quality assessment for IPTV services: A survey," *7th International Conference on Computing and Convergence Technology (ICCCT)*, 3-5 Dec. 2012, pp.182-186.
[8] M. Pinson, and S. Wolf, "An objective method for combining multiple subjective data sets," *SPIE Video Communications and Image Processing Conference*, Lugano, Switzerland, Jul. 2003.
[9] M. H. Pinson and S. Wolf, "Reduced Reference Video Calibration Algorithms", *NTIA Technical Report TR-08-433b*, Nov. 2007.
[10] Stephen Wolf, "Variable Frame Delay (VFD) parameters for video quality measurements," *NTIA Technical Memo TM-11-475*, Apr. 2011.
[11] M. Pinson and S. Wolf, "Fast low bandwidth model: a reduced reference video quality metric," *NTIA Technical Memo*, publication pending.
[12] Z. Wang, A.C. Bovik, H.R. Sheikh and E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol.13, no.4, pp. 600- 612, April 2004.
[13] K. Seshadrinathan, R. Soundararajan, A. C. Bovik and L. K. Cormack, "Study of Subjective and Objective Quality Assessment of Video", *IEEE Transactions on Image Processing*, vol.19, no.6, pp.1427-1441, June 2010.
[14] A.K. Moorthy, C.-C. Su, A. Mittal and A.C. Bovik, "Subjective evaluation of stereoscopic image quality," *Signal Processing: Image Communication*, publication pending.
[15] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the h. 264/avc standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
[16] SVC Reference Software (JSVM Software), Joint Video Team (JVT), [Online]. Available: http://www.hhi.fraunhofer.de/de/kompetenzfelder/image-processing/research-groups/image-video-coding/svc-extension-of-h264avc/jsvm-reference-software.html
[17] A. K. Moorthy, K. Seshadrinathan, R. Soundararajan, and A. C. Bovik, "Wireless video quality assessment: A study of subjective scores and objective algorithms'," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 513–516, Apr. 2010.
[18] B. Wandell, *Foundations of Vision*. Sunderland, MA: Sinauer, 1995.
[19] BT-500-11: Methodology for the Subjective Assessment of the Quality of Television Pictures, Int. Telecommunication Union Std.
[20] M. H. Pinson and S.Wolf, "Comparing subjective video quality testing methodologies," in *Proc. SPIE Vis. Commun. Image Process.*, vol. 5150, 2003.
[21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error measurement to structural similarity," *IEEE Signal Process. Lett.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[22] S. D. Voran and A. A. Catellier, "When should a speech coding quality increase be allowed within a talk-spurt?" *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, May 26-31, 2013.
[23] D. S. Hands, "Temporal characteristics of forgiveness effect," *Electronics Letters*, vol 37 no 12, Jun. 2001, pp 752-754.
[24] Z. Wang, L. Lu, and A. Bovik, "Foveation scalable video coding with automatic fixation selection," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 243–254, Feb. 2003.
[25] D. M. Chandler and S. S. Hemami, "VSNR: A wavelet-based visual signal-to-noise ratio for natural images," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2284–2298, Sep. 2007.
[26] H. R. Sheikh, and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[27] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.
[28] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, "Image quality assessment based on a degradation model," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 636–650, Sep. 2002.
[29] J. Mannos and D. Sakrison, "The effects of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 4, pp. 525–535, Jul. 1974.
[30] 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.
[31] M. Pinson *et al.*, "The influence of subjects and environment on audiovisual subjective tests: an international study," *Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, Oct. 2012, pp. 640–651.
[32] H. R. Sheikh, M. F. Sabir, and A. C. 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.
[33] Final Report from the Video Quality Experts Group on the Validation of Objective Quality Metrics for Video Quality Assessment Phase I, Video Quality Experts Group (VQEG), 2000, [Online]. Available: http://www.its.bldrdoc.gov/vqeg/projects/frtv_phaseI
[34] http://foulard.ece.cornell.edu/gaubatz/metrix_mux/