# Constructing a No-Reference H.264/AVC Bitstream-based Video Quality Metric Using Genetic Programming-based Symbolic Regression

Nicolas Staelens, *Associate Member, IEEE,* Dirk Deschrijver, *Member, IEEE,* Ekaterina Vladislavleva, *Member, IEEE,* Brecht Vermeulen, Tom Dhaene, *Senior Member, IEEE,* and Piet Demeester, *Fellow, IEEE*

***Abstract*—In order to ensure optimal quality of experience toward end users during video streaming, automatic video quality assessment becomes an important field-of-interest to video service providers. Objective video quality metrics try to estimate perceived quality with high accuracy and in an automated manner. In traditional approaches, these metrics model the complex properties of the human visual system. More recently, however, it has been shown that machine learning approaches can also yield competitive results. In this paper, we present a novel no-reference bitstream-based objective video quality metric that is constructed by genetic programming-based symbolic regression. A key benefit of this approach is that it calculates reliable white-box models that allow us to determine the importance of the parameters. Additionally, these models can provide human insight into the underlying principles of subjective video quality assessment. Numerical results show that perceived quality can be modeled with high accuracy using only parameters extracted from the received video bitstream.**

***Index Terms*—H.264/AVC, high definition, no-reference, objective video quality metric, quality of experience (QoE).**

## I. Introduction

DURING real-time transmission of digital video over best-effort internet protocol (IP)-based networks, packet losses can severely degrade the overall quality of experience (QoE) of the end users [1]. This, in turn, influences willingness to pay and customer satisfaction [2], [3]. Furthermore, QoE is considered a key factor for the success or failure of new broadband video services [4]. Therefore, service providers strive toward maximizing and maintaining adequate QoE at all times. In the case of video streaming, this requires continuous monitoring and measuring of perceived video quality in order to get an indication of end users' QoE.

Subjective video quality assessment is commonly used to measure the influence of visual degradations on perceived quality of video sequences [5]. During subjective quality assessment, real human observers evaluate the visual quality of a number of short video sequences by, for example, providing a score between 1 (bad) and 5 (excellent) after or while watching each video sequence. Afterwards, the mean opinion score (MOS) is calculated per video sequence as the average quality rating provided by the different observers. Several assessment methodologies have already been standardized by the International Telecommunications Union (ITU) in ITU-T Recommendation P.910 [6] and ITU-R Recommendation BT.500-12 [7], and describe in detail how subjective video quality experiments should be set up and conducted. However, subjective experiments are time consuming, expensive and need to be conducted in controlled environments. Furthermore, it is clear that subjective quality assessment cannot be used in the case of real-time quality monitoring and measuring.

Over the past years, a lot of research has been conducted toward the construction of objective video quality metrics. As stated in [8], "The goal of objective image and video quality assessment research is to design quality metrics that can predict perceived image and video quality automatically." This rather broad definition can also be refined by stating that this prediction should be reliable and correlate well with scores of subjective quality assessment (= MOS scores).

In this paper, the use of a robust machine learning (ML) technique, called symbolic regression, is proposed to derive a new no-reference bitstream-based objective video quality metric for estimating perceived quality of high definition (HD) H.264/AVC encoded videos. The new approach has two distinctive features that make it particularly attractive for the analysis of perceived video quality: 1) it allows us to perform an automated selection of the most important variables, and 2) it provides predictive models that are interpretive and provide insight in the relation between encoder settings, loss location, and video content characteristics and the perceived quality.

Manuscript received June 5, 2012; revised August 23, 2012; accepted December 6, 2012. Date of publication January 28, 2013; date of current version July 31, 2013. The research activities that have been described in this paper were funded by Ghent University, iMinds, and the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT). This paper is the result of research carried out as part of the OMUS project funded by the iMinds. OMUS is being carried out by a consortium of the industrial partners: Technicolor, Televic, Streamovations, and Excentis in cooperation with iMinds research groups: IBCN & MultimediaLab & WiCa (UGent), SMIT (VUB), PATS (UA), and COSIC (KUL). This work was also supported by the Research Foundation in Flanders (FWO-Vlaanderen). This paper was recommended by Associate Editor E. Steinbach.

N. Staelens, D. Deschrijver, B. Vermeulen, T. Dhaene, and P. Demeester are with Ghent University—iMinds, Department of Information Technology, Technology, Ghent 9050, Belgium (e-mail: nicolas.staelens@intec.ugent.be; dirk.deschrijver@intec.ugent.be; brecht.vermeulen@intec.ugent.be; tom.dhaene@intec.ugent.be; piet.demeester@intec.ugent.be).

E. Vladislavleva is with Evolved Analytics Europe BVBA, Wijnegem 2110, Belgium (e-mail: katya@evolved-analytics.com).

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/TCSVT.2013.2243052

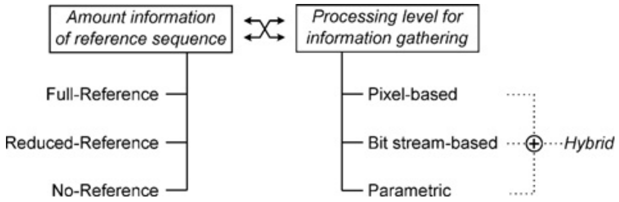

Fig. 1. Different categories of video quality metrics based on the amount of information which is used from the reference sequence or based on the processing level for extracting information in order to model perceived quality.

The remainder of this paper is outlined as follows. Section II gives an overview of the state-of-the-art and describes how ML algorithms can be used to derive different types of objective video quality metrics. In Section III, the modeling principles of the genetic-programming-based symbolic regression approach are outlined, and a detailed description of the algorithmic aspects is provided. In order to validate the effectiveness of the proposed method, an extensive subjective experiment was conducted. The procedure that is followed to collect the subjective quality ratings from human observers is discussed in Section IV. Then, Section V presents the main results and describes how the new modeling approach can be applied to derive a new no-reference bitstream-based metric for video quality assessment. First, the different parameters that are extracted from the received encoded video bitstream are listed and the symbolic regression approach is used to determine the most important parameters. Next, this subset of parameters is selected in the modeling process to compute a final model that predicts the perceived video quality in a reliable way. An interpretation of the model and a comparison with alternative machine learning techniques is also provided. Numerical results confirm that the perceived quality can be predicted accurately using only parameters extracted from the received video bitstream. Section VI concludes the article.

## II. Machine Learning-Based Metrics

### A. Objective Video Quality Metrics

In general, objective video quality metrics can be categorized into three main classes, based on the availability of the original video sequences as depicted in Fig. 1.

Full-reference (FR) quality metrics [9] require access to the complete original video stream. Research has already shown that FR metrics can predict perceived quality with high accuracy. However, due to their dependency on the original video, FR metrics cannot be used for real-time video quality evaluation. Reduced-reference (RR) metrics, recently standardized in [10], perform quality predictions by comparing features extracted from the original and received video sequence. In order to transmit these features from the points they are extracted to the point where the quality evaluation is performed, an ancillary error-free channel is needed. Both FR and RR metrics usually predict quality based on a frame-by-frame evaluation. As such, the received video stream requires proper alignment with the original sequence. From a real-time monitoring point of view, no-reference (NR) video quality metrics are the most interesting ones as they neither need access to the original sequence nor rely on feature extraction [11].

Another criterion to categorize these metrics is the type of information or processing level where the information from the video sequences is extracted. As such, pixel-based metrics require access to the decoded video stream whereas bitstream-based metrics only perform a parsing of the encoded data. A last category of video quality metrics are the parametric metrics, which only use high level information accessible through the packet headers in the case of video streaming. Quality metrics combining information from the network-, pixel-, and/or bitstream-level are also called hybrid metrics. For real-time video quality evaluation, metrics that do not require a complete decoding of the received video stream are of particular interest. The interested reader is referred to [12] for more details on the classification of objective video quality metrics and a performance comparison.

The performance of FR and RR video quality metrics has already widely been investigated by the Video Quality Experts Group (VQEG) through several projects [13]–[15]. The results of this study resulted in the standardization of a number of objective video quality metrics. However, research is still ongoing toward the construction of NR metrics.

### B. Overview of State-of-the-art

Machine learning techniques such as neural networks (NN), support vector machines (SVM), support vector regression (SVR), and decision trees have already successfully been applied for constructing objective video quality metrics. In general, these metrics either use regression or classification for estimating perceived quality. Regression is commonly used for estimating MOS whereas classification is typically used for predicting error visibility by means of a binary decision.

In Mohamed *et al.* [16], [17], used NNs for constructing an objective video quality metric capable of continuous quality monitoring and measuring. The stream bitrate, sequence frame rate, network loss rate and burst size, and the ratio of encoded intra- to intermacroblocks are used as inputs to the NN. However, only a single low resolution [352 × 288 pixels[1]] video sequence was used for training and validating the model. Consequently, the influence of video content is not considered in the proposed model. Nevertheless, results indicated that quality can be estimated with a high accuracy without the need for modeling the human visual system (HVS). A similar approach is followed in [18], where an NN is used to estimate perceived quality of H.264/AVC encoded CIF resolution sequences. However, the authors do not consider the influence of network impairments on the perception of quality. NNs have further also been used in [19], [20], and [21] for modeling video quality.

Recently, SVMs have been used to predict video quality. In [22], different NR and RR parameters are extracted from the decoded video stream and used to build an SVM. The performance of the model is evaluated based on an existing FR objective video quality metrics, VQM [23]. Compared to their previous work [24], the authors found that video quality

[1] Common interchange format (CIF) resolution.

can be predicted with a higher accuracy using SVMs. Also Narwaria *et al.* [25]–[27] model visual quality using SVR. More specifically, ML is used for modeling the interaction effect between spatial and temporal quality factors affecting perceived video and image quality. In [27], they evaluate the performance of SVR on a number of video databases against eight different existing visual quality predictors, and the results show a significant improvement in prediction accuracy.

Rather than estimating the perceived quality, Argyropoulos *et al.* [28], [29] use SVMs to build a classifier for estimating the probability that an impairment in the video stream will result in a visible impairments for the end-user.

In [30] and [31], packet loss visibility is estimated using decision trees. In this case, a binary classification is performed labeling an impairment in the video bitstream as visible or invisible to the average end user. A decision tree is also used by Menkovski *et al.* [32] to determine whether the QoE of a video service is acceptable or not acceptable. As a decision tree is a white box model, the internal structure of the classification process is completely visible and can thus be used to gain better insights in the modeling process. This is not the case for SVMs and NNs, which are black-box models.

In our previous work [33], we investigated and modeled impairment visibility in HD H.264/AVC encoded video sequences using decision trees. Our results showed that it is possible to reliably predict impairment visibility using only a limited number of parameters extracted from the received video bitstream. As a decision tree was used, a binary classification is made. The work presented in this paper further elaborates on the data obtained during our previous work. However, instead of determining impairment visibility we are now interested in estimating how end users would rate the visual quality of the video sequences. As such, our goal is to construct a video quality metric which predicts perceived quality and correlates well with the MOS obtained during subjective quality assessment.

## III. GP-Based Symbolic Regression

In this section, a novel ML technique [genetic programming (GP)-based symbolic regression method [34]] is proposed to model the perceived quality, as an alternative to modeling the different complex properties of the HVS. As the name suggests, this method is applied to model the MOS score by means of a regression approach (not classification). A key advantage of this method is that the resulting metrics are essentially white-box models that comprise only the variables that are truly influential. Moreover, the metric can provide human insight into the underlying principles of subjective video quality assessment.

### A. Goal Statement and Notation

GP-based symbolic regression offers the unique capability to compute nonlinear white-box models that predict the MOS quality rating $q$ of a video fragment in terms of several input variables $\vec{v} = \{v_n\}_{n=1}^{N}$. These variables $\vec{v}$ identify the characteristics of the sequence and quality degradation factors, and comprise only those parameters that can be extracted from the received video bitstream without the need for complete decoding. Given a limited set of $M$ video sequences, a sparse set of data samples $\mathcal{S}$ is obtained, which can be represented as a set of tuples $\mathcal{S} = \{(\vec{v}_m, q_m)\}_{m=1}^{M}$. Symbolic regression is then used to compute a set of models $f$ that predict the MOS quality scores $q$ in terms of the parameters $\vec{v}$ [35]

$$f\colon \mathbb{R}^N \to \mathbb{R},\ f(\vec{v}_m) \approx q_m. \quad (1)$$

A set of models is calculated, because this allows us to determine the importance of variables in a reliable way. As suggested by El Khattabi *et al.* [21], one should only retain the relevant input variables, in order to reduce the computational cost and to limit the model complexity. The importance of carefully selecting the input variables for subjective video quality assessment was also highlighted in [35]. After discarding the redundant variables, a new set of models is computed using only the most important variables and the best model is returned as the final solution. Note that for the actual implementation, we made use of the DataModeler package [36] for Mathematica, because it offers the integrated functionality for automatic variable selection and dimensionality analysis, variable contribution analysis and set-based predictions.

### B. Outline of Evolutionary Algorithm

This section explains some algorithmic details on how the set of models can be computed in a reliable way. GP-based symbolic regression is a biologically inspired method that mimics the process of Darwin's evolution theory and the mechanisms of genetic variation and natural selection [37]. It is based on the concept of genetic programming, and computes a set of tree-based regression models that give a good approximation of the sparse subjective video quality data $\mathcal{S}$. The evolutionary algorithm consists of the following steps.

1) **Model initialization**: In the first generation step ($t = 1$), an initial population of $K$ randomly generated parse trees $P^t = \{f_k(\vec{v})\}_{k=1}^{K}$ (also called models or individuals) with a maximal arity of 4 is formed. Each parse tree $f_k(\vec{v})$ represents a potential solution to the approximation problem, and is composed of multiple nodes that comprise primitive functions and terminals. The primitive functions are represented by the standard arithmetic operators ($+, -, *, /, inv, pow, sqrt, \ln, \exp$), whereas the terminals consist of the input variables $\vec{v}$ and real constants drawn from the interval $[-5, 5]$. All the input variables $\vec{v}$ are scaled into the range 0–2.
2) **Model evaluation**: In order to measure the fitness of a particular individual, an operator $Z$ is defined that maps each model onto the space of two design objectives

$$Z : f_k(\vec{v}) \in P^t \to (z_1(f_k(\vec{v})), z_2(f_k(\vec{v}))) \in \Theta \quad (2)$$

a) Objective 1 aims to minimize the prediction error

$$z_1(f_k(\vec{v})) = 1 - R^2(q, f_k(\vec{v})) \quad (3)$$

where $R$ represents the correlation coefficient

$$R(q, f_k(\vec{v})) = \frac{\mathrm{cov}(q \cdot f_k(\vec{v}))}{\mathrm{std}(q) \cdot \mathrm{std}(f_k(\vec{v}))}. \quad (4)$$

TABLE I

PARETO GP EXPERIMENTAL SETTINGS

| Setting | Values |
|---|---|
| No. of replicates | 5 |
| No. of generations | 310 |
| Population size | 1000 |
| Archive size | 100 |
| Crossover rate | 0.9 |
| Subtree mutation rate | 0.1 |
| Population tournament | 5 |

b) Objective 2 aims to minimize the expressional model complexity $z_2(f_k(\vec{v}))$, which is defined as the sum of the number of nodes in all subtrees of a given tree. This objective penalizes complexity and avoids the excessive growth of models over time.

Both criteria are often conflicting, so the goal is to obtain models that make a good tradeoff and perform well on both objectives. This idea is motivated by Occam's razor, which states that simpler models are preferable to more complex ones if they explain the data sufficiently well.

3) **Model archiving and elitism**: Next to the population $P^t$, the algorithm also maintains a fixed-size archive $A^t$ that contains the best performing models discovered so far. This archive serves as an elitism strategy to ensure that the fittest models of the population are carried forward from one generation to the next. In each generation step $t$, the archive $A^t$ is updated by selecting the least-dominated models from the joint set $A^{t-1} \cup P^t$, where initially $A^0 = \phi$. Note that a model $f_1(\vec{v})$ is said to dominate a model $f_2(\vec{v})$ in the objective space $\Theta$ if $f_1(\vec{v})$ is no worse than $f_2(\vec{v})$ in all the objectives, and strictly better in at least one of the objectives

$$\forall i = 1, 2 : z_i(f_1(\vec{v})) \leq z_i(f_2(\vec{v})) \tag{5}$$

$$\exists j \in \{1, 2\} : z_j(f_1(\vec{v})) < z_j(f_2(\vec{v})) \tag{6}$$

(Models that are not outperformed by any other model in terms of both objectives are Pareto-optimal models)

4) **Model evolution**: In each step $t$ of the algorithm, a set of individuals is chosen by means of a Pareto tournament selection operator. These individuals are exposed to genetic operators (such as crossovers and mutations), in order to create the population $P^{t+1}$ of the next generation step. The crossover operator selects two parent individuals and combines them to create new offspring by swapping subtrees, whereas the mutation operator makes a random alteration to nodes of a subtree. At each generation, archive members ($A^t$) are merged with the newly created population ($P^t$), and variation operators are applied to the aggregated set of models. Selection of individuals for crossovers and mutations happens by means of Pareto tournaments. This archive-based selection preserves genotypic diversity of the individuals. New individuals are generated using a subtree crossover with rate 0.9, and subtree mutation with rate 0.1. Every 10 generations, the population gets re-initialized to provide diversity and avoid inbreeding.

TABLE II

CHARACTERISTICS OF THE EIGHT SELECTED TEST SEQUENCES

| Sequence | Source | Description |
|---|---|---|
| *basketball* | CDVL | Basketball game with score. Camera pans and zooms to follow the action. |
| *BBB** | Big Buck Bunny | Computer-Generated Imagery. Close-up of a big rabbit. Slight camera pan while following a butterfly in front to the rabbit. |
| *cheetah* | *CDVL* | Cheetah walking in front of a chainlink fence. Camera pans to follow the cheetah. |
| *ED** | Elephants Dream | Computer-Generated Imagery. Fixed camera focusing on two characters. Motion in the background. |
| *foxbird3e* | CDVL | Cartoon. Fox running toward a tree and falling in a hole. Fast camera pan with zoom. |
| *purple4e* | CDVL | Spinning purple collage of objects. Many small objects moving in a circular pattern. |
| *rush hour* | TUM | Rush hour in Munich city. Many cars moving slowly, high depth of focus. Fixed camera. |
| *SSTB** | Sita Sings the Blues | Cartoon. Close-up of two characters talking. Slight camera zoom in. |

This evolutionary process is repeated over many generation steps ($t = 1, ..., T$), in order to create models with increasing fitness, based on the survival-of-the-fittest principle. The settings of the algorithm are illustrated in Table I. After a certain stopping criterion is met (e.g., a time budget), the algorithm is terminated. All the archives $A^t$ in each run are aggregated into a compound archive $A$, and the nondominated individuals in $A$ are used to form a super Pareto front of models. This set of models is then returned as the final result.

## IV. SUBJECTIVE VIDEO QUALITY EXPERIMENT

In order to validate the method, a subjective video quality experiment was conducted. During the experiment, a number of test subjects had to evaluate the visual quality of a number of impaired video sequences. This section provides details of the experiment, as well as the selection, encoding, and impairing of the different video sequences that were used.

### A. Source Video Sequence Selection

As a base for the subjective experiment, eight freely available source video sequences were selected. These sequences were obtained from open source movies, the Consumer Digital Video Library (CDVL) [38] and the Technical University of Munich (TUM). All sequences were in full 1080p HD resolution (1920 × 1080), with a frame rate for 25 f/s and a duration of exactly 10 s. An overview of the different source sequences is shown in Fig. 2, and a short description of their characteristics is provided in Table II. Marked sequences ($*$) were taken from an open source movie.

It is generally known that certain video characteristics, i.e., the amount of motion and the amount of spatial details, influence the perceptibility of visual degradations [39], [40]. Therefore, it was ensured during the subjective video quality assessment that these video sequences have different amounts

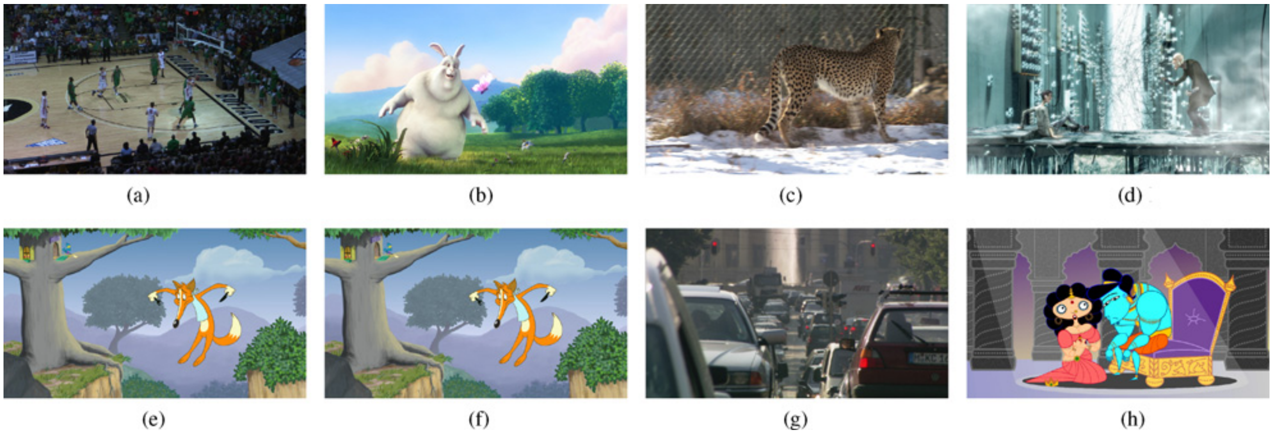

Fig. 2. Overview of the eight selected source video sequences, taken from open source movies, CDVL and TUM. (a) *basketball*. (b) *BBB*. (c) *cheetah*. (d) *ED*. (e) *foxbird3e*. (f) *purple4e*. (g) *rush hour*. (h) *SSTB*.

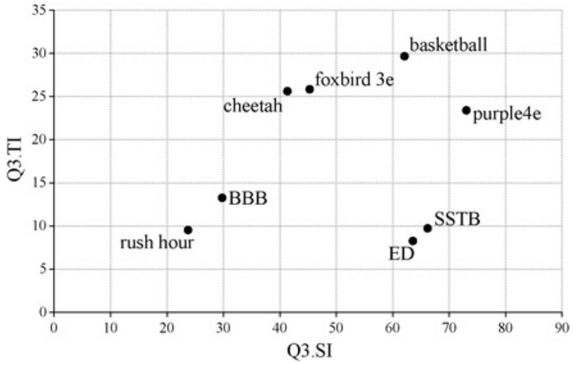

Fig. 3. Calculated Q3.SI and Q3.TI values for each sequence [41].

of motion and textures [6]. For quantifying the amount of motion and spatial details in a video sequence, two perceptual metrics are defined in ITU-T Recommendation P.910: the spatial perceptual information (SI) and the temporal perceptual information (TI). These measurements are calculated frame-by-frame and the maximum value is taken as overall SI and TI value for a particular video sequence. However, Ostaszewska *et al.* [41] showed that the overall SI and TI of a video sequence can be better approximated by taking the upper quartile value instead of the maximum value as this eliminates the influence of peak values (caused by, e.g., scene cuts). Hence, the SI and TI value of a video sequence were calculated as follows:

$$Q3.SI = \text{Upperquartile}_{\text{time}}\{\text{stdev}_{\text{space}}[\text{Sobel}(F_n)]\}. \quad (7)$$

$$Q3.TI = \text{Upperquartile}_{\text{time}}\{\text{stdev}_{\text{space}}[M_n(i, j)]\}, \quad (8)$$

where $M_n(i, j) = F_n(i, j) - F_{n-1}(i, j)$.

Fig. 3 visualizes the calculated *Q3.SI* and *Q3.TI* values for each of the selected sequences.

### B. Encoding and Impairment Generation

The paper is focused on the estimation of perceived quality for HD H.264/AVC encoded video sequences. In order to use realistic encoder settings, the settings used for HD content available from online video services and websites were analyzed. Furthermore, the default settings recommended by a number of commercially available H.264/AVC encoders were investigated. Based on this analysis, *x264* was used with the following settings for encoding the video sequences:

1) number of slices: 1, 4, and 8;
2) number of B-pictures: 0, 1, and 2;
3) GOP size [42]: 15 (0 or 1 B-picture) or 16 (2 B-pictures);
4) closed GOP structure;
5) bit rate: 15 Mb/s.

This results in a total number of nine different encoder configurations. Each encoded video sequence was also carefully visually inspected to ensure no encoding artifacts were present.

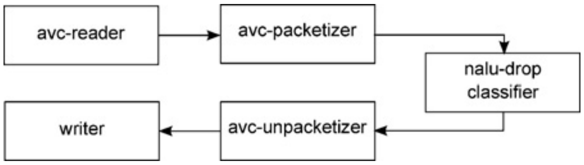

Fig. 4. RTP packets, which carry data from particular slices, are dropped using the nalu-drop classifier component. After unpacketizing, the resulting impaired sequence is saved to a new file.

The open source streamer Sirannon [43] was used to impair the encoded video sequences. First, the raw H.264/AVC Annex B bitstream was packetized into RTP packets according to RFC3984. Then, slice losses were simulated by dropping all RTP packets carrying data from that particular slice.[2]Finally, the stream was unpacketized and the resulting impaired bitstream was saved to a new file. This process is illustrated in Fig. 4.

It is also possible to use different configurations for dropping slices, by considering on the following parameters:

1) number of B-pictures (0, 1, 2);
2) type of first lost slice (I, P, B);
3) location within the GOP of the loss (begin, middle, end);
4) number of consecutive slice drops (1, 2, 4);

[2] Entire slices should be dropped in order for the bitstream to remain compliant with Annex B as specified in the H.264/AVC video coding standard.

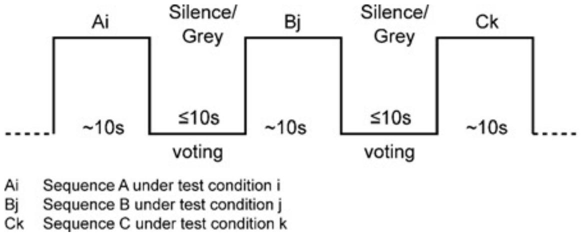
Ai
Silence/ Grey
Bj
Silence/ Grey
Ck
~10s
≤10s
voting
~10s
≤10s
voting
~10s
Ai Sequence A under test condition i
Bj Sequence B under test condition j
Ck Sequence C under test condition k

Fig. 5. Typical trail structure of an SS subjective quality experiment defines the order how sequences are displayed and rated by the subjects.

5) location within the picture of the loss (top, middle, bottom);
6) number of consecutive entire picture drops (0, 1).

An experimental design was used to select a subset of 48 representative impairment scenarios which are applied to the eight selected sequences. This resulted in a total amount of $M = 384$ impaired video sequences. Note that no visual impairments were injected in the first and last two seconds of video playback. The interested reader is referred to [33] for more details on the experimental design.

For decoding the impaired sequences, a modified version of the JM Reference Software version 16.1 was used, which implements frame copy as concealment strategy [44], [45].

### C. Subjective Quality Assessment Methodology

The different encoded and impaired video sequences were presented to human subjects using a single stimulus (SS) absolute category rating (ACR) subjective assessment methodology, as specified in ITU-T Rec. P.910 [6]. The SS methodology implies that all sequences are presented one-after-another, as depicted in Fig. 5. Immediately after watching each sequence, subjects are required to evaluate the quality of that particular sequence using a five-grade ACR scale with adjectives.

Before the start of the experiment, all subjects received specific instructions on how to evaluate the different video sequences. Ishihara plates and a Snellen chart were used to test the users for visual acuity and normal vision. Three training sequences were presented to indicate the typical impairments that they could perceive during the experiment and to get the subjects familiarized with the test software. The quality ratings that were assigned to these test sequences are not taken into account when analyzing the results. The sequences that had to be evaluated were divided into six distinct datasets, each containing 76 sequences. This limited the experiment duration to 20 min. Subjects were encouraged to evaluate more than one dataset, although not necessarily on the same day. The order in which the sequences were presented was randomized at the start of the experiment. This way, no two subjects evaluated the sequences in exactly the same order.

The experiment was conducted inside an ITU-R BT.500 [7] compliant test environment. A 40 inch full HD LCD television was used to display the sequences. The test subjects were seated at four times the picture height (4H) from the screen.

Forty nonexpert subjects participated in the experiment: 11 were females and 29 were male subjects. The age of the subjects ranged from 18 to 34 years old. Most subjects evaluated more than one dataset, and each dataset was evaluated by exactly 24 subjects. The postexperiment screening method detailed in Annex V of the VQEG HDTV report [15] was used to ensure no outliers were present in the response data.

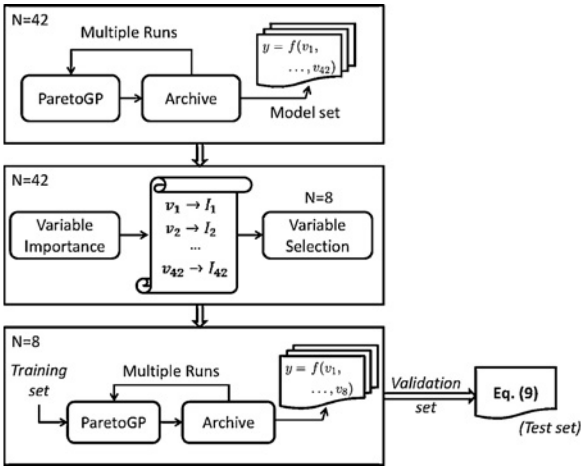

Fig. 6. Approach of using GP-based symbolic regression for estimating perceived video quality. After determining variable importance, a final set of candidate models is generated after which, the best model is selected for estimating video quality.

## V. Modeling Perceived Video Quality

This section presents the results of using GP-based symbolic regression to model the perceived video quality.

First, all parameters extracted from the received video bitstream are made available to the modeling process. Next, based on the set of generated GP models, variable importance is determined. A new set of models is then generated using only the most important variables. Finally, from the resulting set of GP models, the best model is selected for predicting video quality. This approach is visually presented in Fig. 6. This section is concluded by a comparison between the presented approach and other existing ML techniques. The performance of the metric is also validated on existing benchmark databases.

### A. Listing of the Parameters

The focus of this paper is the construction of an NR bitstream-based objective video quality metric. Hence, only parameters that can be extracted or calculated from the received video bitstream, without the need for complete decoding, are considered. This set comprises $N = 42$ different parameters that are subdivided into the following three categories:

1) describe the encoder settings;
2) identify the location and severity of the loss;
3) characterize the video content.

A complete listing of the parameters is provided in Table III.

Note that the parameter `drift` represents the temporal duration (extent) of the loss, i.e., the number of frames which are affected by the loss. If a loss occurs in an I-picture, the loss is propagated through the entire GOP. B-pictures are in

TABLE III

OVERVIEW OF PARAMETERS EXTRACTED FROM RECEIVED VIDEO BITSTREAM IN ORDER TO IDENTIFY LOCATION OF LOSS AND TO CHARACTERIZE VIDEO CONTENT

| Parameter | Description |
|---|---|
| *Encoder settings* | |
| **B-pictures**, **slices**, GOP | Number of B-pictures, slices per picture and GOP size as specified during encoding. |
| *Loss location and severity* | |
| **i_loss**, **p_loss**, b_loss | Indication (1 or 0) whether the loss originates from an I-, P- or B-picture. |
| **perc_pic_lost** | Percentage of slices lost of the picture where the loss originates. |
| imp_in_gop_pos, imp_in_pic_pos | Temporal location within the GOP (begin, middle, end) and spatial location within the picture (top, bottom, middle) of the first lost slice. |
| imp_in_gop_idx, imp_in_pic_idx | Absolute position within the GOP and within the picture of the first lost slice. |
| **imp_cons_slice_drops**, imp_cons_b_slice_drops, imp_pic_drops | Number of consecutive slice drops, number of consecutive B-slice drops and number of entire picture drops. |
| drift | Temporal duration of the loss. |
| *Video content characteristics* | |
| perc_pb_4 × 4, perc_pb_8 × 8, perc_pb_16 × 16, perc_pb_8 × 16, perc_pb_16 × 8, perc_i_4 × 4, **perc_i_8 × 8**, perc_i_16 × 16 | Percentage of I, P & B macroblocks of type 4 × 4, 8 × 8, 16 × 16, 8 × 16, and 16 × 8, averaged over the pictures in the GOP containing the loss. |
| perc_i_mb, perc_skip, perc_ipcm | Percentage of macroblocks encoded as I, skip and PCM, averaged over the pictures in the GOP containing the loss. |
| I_perc_4 × 4, **I_perc_8 × 8**, I_perc_16 × 16 | Percentage of macroblocks of type 4 × 4, 8 × 8 and 16 × 16 in the first I or IDR picture of the GOP containing the loss. |
| abs_avg_coeff, avg_qp | Absolute average value of the macroblock coefficients and QP value, averaged over the P or B pictures in the GOP containing the loss. |
| I_abs_avg_coeff, I_avg_qp | Absolute average value of the macroblock coefficients and QP value in the first I or IDR picture of the GOP containing the loss. |
| perc_zero_coeff, I_perc_zero_coeff | Percentage of zero coefficients, averaged over the P or B pictures in the GOP containing the loss and average of zero coefficients in the first I or IDR picture of the GOP containing the loss. |
| avg_mv_x, avg_mv_y, stdev_mv_x, stdev_mv_y<br>avg_mv_xy, stdev_mv_xy | Average absolute motion vector length and standard deviation in *x*- and *y*-direction, averaged over the P or B pictures in the GOP containing the loss. Motion vector magnitudes have quarter pixel precision.<br>Average and standard deviation of the sum of the motion vector magnitudes in *x*- and *y*-direction, averaged over the P or B pictures in the GOP containing the loss. Motion vector magnitudes have quarter pixel precision. |
| perc_zero_mv | Average percentage of zero motion vectors, calculated over the P or B pictures in the GOP containing the loss. |

Subset of eight influential parameters is marked in bold.

TABLE IV

CALCULATED AVERAGE DRIFT (WITH STANDARD DEVIATION) CAUSED BY LOSSES IN P-PICTURES, IN RELATION TO THE LOCATION WITHIN THE GOP OF THE P-PICTURE

| | Location within GOP | | |
|---|---|---|---|
| | BEGIN | MIDDLE | END |
| avg(drift) | 14 | 9 | 4 |
| stdev(drift) | 1 | 2 | 2 |

our case never used as reference. Hence, losses in B-pictures only affect one picture and do not propagate any further. The temporal extent caused by losses in P-pictures depends on the location of that particular picture within the GOP. Based on our created video sequences (as detailed in Section IV-B), we calculated the average drift caused by losses in P-pictures, in relation to the position of that picture within the GOP. Hence, drift is calculated in the pixel domain as the number of pictures containing a visual distortion. The calculated values are listed in Table IV.

As indicated in Table III, all parameters are calculated based on the GOP containing the loss. If a loss occurs in an I-picture, statistics are calculated using the remaining P- and B-pictures. In case the loss occurs in a P-picture, the parameters are calculated using the I-picture and the remaining P-pictures in the GOP (except for the P-pictures where the loss originates from). This is similar when the loss originates from a B-picture, in which case the I- and B-pictures are used for calculating the statistics.

### B. Identifying the Importance of Variables

First, all parameters are used during the modeling procedure, and the resulting set of models is shown in Fig. 7. Models that lie on the Pareto front are marked in black and represent the best individuals in the population [46]. The plot illustrates that an increase in model complexity often results in more accurate predictions, and it shows that the model prediction error saturates around 0.09. It is, however, found that variables that are not truly significant are often present in reasonable quantities in the final models. The presence of insignificant variables in regression models is usually undesired, because it can lead to overfitting and models that are very complex to interpret. There are multiple reasons why such variables can be present in models: due to the stochastic nature of the GP algorithm, insignificant variables that disappeared from models during the evolutionary run still have a chance to come back by means of the random mutation operator. On some occasions, insignificant variables are present in low-order metavariables evaluating to an important constant.

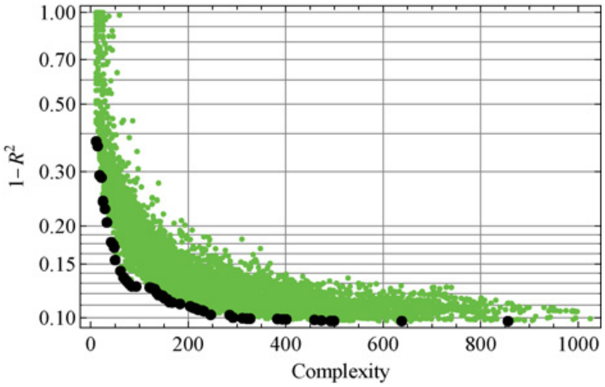

Fig. 7. Set of GP models generated using all parameters extracted from the video bitstream. Models on the Pareto front are marked in black.

Fortunately, there is a robust way to overcome this problem. The DataModeler environment offers the variable contribution analysis function that estimates the contribution of each variable into the prediction error of each individual symbolic regression model, based on the rate of change in the relative prediction error when the variable is present or removed from the model. It estimates the contribution of each variable to each model in the set and aggregates all the results. Fig. 8 demonstrates the quantitative characteristics of the variable contribution. For example, a variable contribution of 120% for variable `drift` means that the removal of this variable from a model causes on average a 120% increase in the prediction error. This implies quantitatively that `drift` is a highly important variable, which clearly agrees with the common sense and domain knowledge. In order to determine the actual drift accurately, pixel data should be reconstructed. However, in this paper, we are targeting a bitstream-based video quality metric which does not require a decoding (= pixel reconstruction) of the video stream. Therefore, it was decided to omit this parameter as well and to use only the parameters that can exactly be extracted and calculated from the received encoded video bitstream.

The results of the variable contribution analysis show that there are $N = 8$ influential parameters for modeling perceived video quality: `perc_pic_lost`, `i_loss`, `slices`, `p_loss`, `B_pictures`, `imp_cons_slice_drops`, `I_perc_8x8` and `perc_i_8x8`. Interestingly, these parameters largely correspond with the variables that were used in our previous work [33] for modeling the impairment visibility in H.264/AVC encoded HD video sequences.

## C. Final Modeling of Perceived Quality

The variable contribution analysis is most beneficial, since it identifies that only 20% of video bitstream parameters are causing significant changes in the video quality perception. This information is of high value because it significantly decreases the problem dimensionality, and focuses future research. In this section, a final modeling step is applied to construct an objective quality metric using only the most important variables as determined in the previous section.

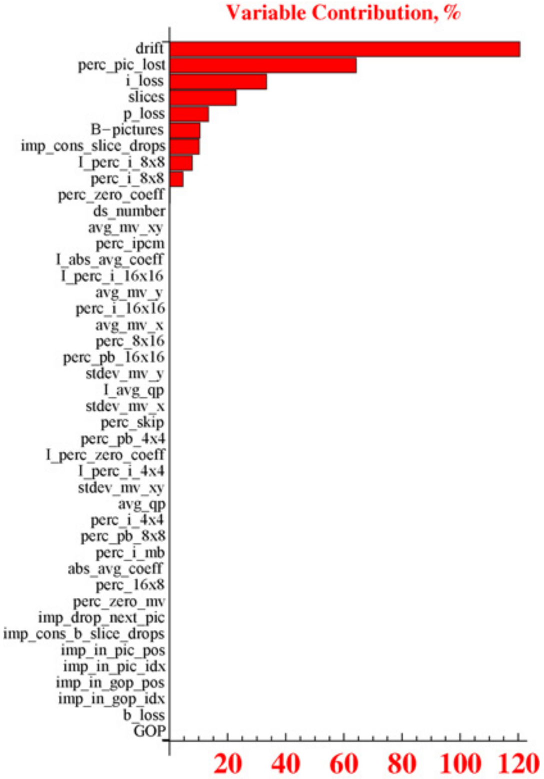

Fig. 8. Contribution of each variable into the prediction error of the regression models when removing that particular variable from the model.

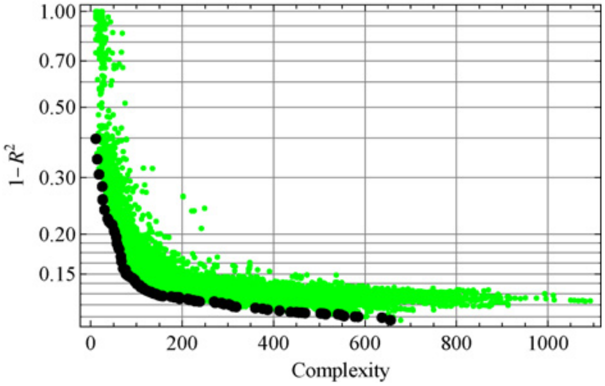

Fig. 9. Set of GP models generated using only the selected influential parameters. Models on the Pareto front are marked in black.

In order to compute an interpretable model that uses only these influential variables, the data set $\mathcal{S}$ is divided into a disjoint training (60%), validation (20%), and test (20%) set. The training set is used to generate new models based on the subset of eight parameters and the results are depicted in Fig. 9. It shows that the models are able to fit the data using much less variables without suffering a significant loss in accuracy.

All models that lie on the Pareto front are now candidate models for predicting video quality. In order to pick the best model from the set, all Pareto-optimal models in Fig. 9 are evaluated on the validation set. Fig. 10 plots the prediction error between predicted and actual MOS against model's complexity for each Pareto optimal models evaluated on the validation set. Model complexity is computed as the total sum of nodes in all the subtrees of the parse tree representation

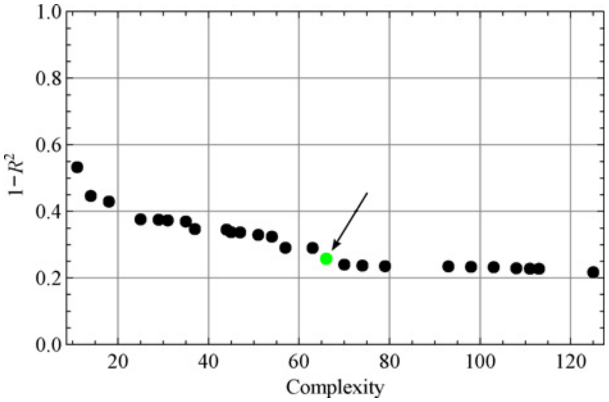

Fig. 10. Prediction error versus model complexity for each Pareto efficient model identified using the validation set. The arrow indicates the final selected model.

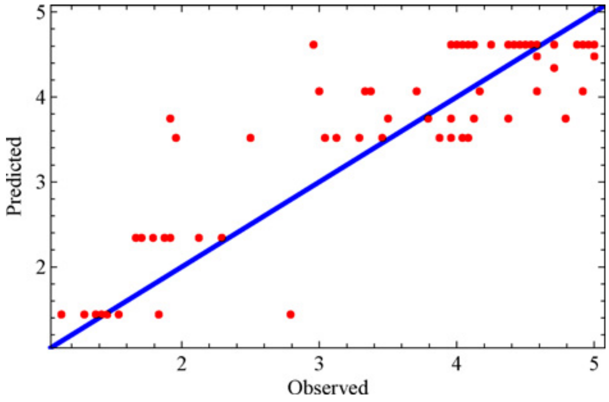

Fig. 11. Predicted MOS equation (9) versus actual MOS over the test set.

of that particular model. In order to select the final model, a tradeoff must be made between model complexity and accuracy. Based on the plot in Fig. 10, it can be seen that the performance saturates of the Pareto optimal models with a complexity of 60 or more. Therefore, we selected the final model (indicated by the arrow in Fig. 10) as the point in the graph after which there is no significant gain in prediction accuracy. This corresponds with the point located near the 'elbow' of the plot.

The performance of the final model is then assessed by evaluating it over the test set that contains previously unseen data. For each data sample in the test set, the predicted MOS ($MOS_p$) is compared to the actual MOS ($q$) and the result is depicted in Fig. 11. The Pearson correlation coefficient $R$ over the test set is 0.9003 with a root mean squared error (RMSE) of 0.5663, which confirms that the final model is indeed capable of predicting perceived quality with a high accuracy, using only a limited number of parameters extracted solely from the received encoded video bitstream.

### D. Objective Video Quality Metric

The parse tree corresponding with the final selected model is depicted in Fig. 12. This tree can easily be translated to the

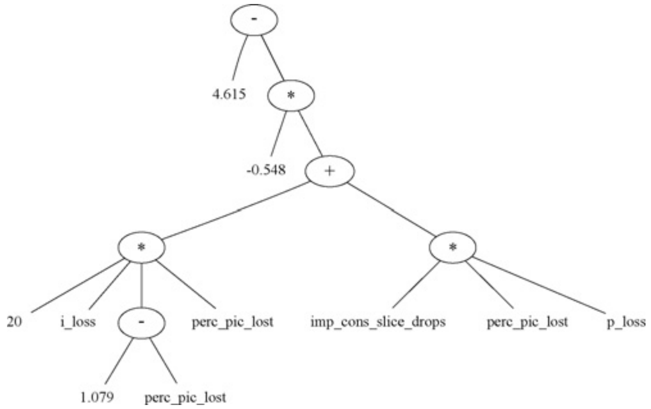

Fig. 12. Parse tree corresponding with the selected GP model indicated in Fig. 10.

algebraic expression shown in (9).

In this model, only four parameters are present for estimating video quality, since only models with a higher complexity use all eight parameters. This formula computes the predicted MOS as a large constant from which multiple terms are subtracted. Each term is weighted by the type of picture where the loss originates from. Losses originating in I- or P-pictures cause a drop in perceived quality.

In the case losses occur in B-pictures, perceived quality equals 4.615. In previous research [33], we found that losses in B-pictures are never perceived. As such, the quality is not influenced. The fact that, in this case, perceived quality does not equal 5 (i.e., excellent quality) can be explained by the fact that subjects tend to avoid using the extremes of the rating scales [47] during the subjective video quality evaluation. This effect is also known as the saturation effect [48].

In general, losses in I-pictures will result in a higher drop in quality due to the drift (=spatial extent) caused by the decoding dependencies with other pictures in the GOP. Losses of entire I-pictures are rated higher quality compared to losing only a certain portion of the picture. This matches the conclusions of [33] where it was found that dropping an entire picture and relying on the error concealment strategy might benefit quality perception. In the case of similar consecutive pictures, frame freezes can be used as an efficient concealment technique [49].

When losses originate from P-pictures, the drop in perceived quality is further depending on the amount of slices per picture and the amount of slices lost. For the same amount of the picture lost (`perc_pic_lost`), a higher number of slices per picture will result in a slightly higher drop in quality. This again matches with the earlier findings [33] that impairment visibility of loosing up to half a picture depends on the number of encoded slices in that particular picture, i.e. impairments are easier detected in sequences encoded with multiple slices.

### E. Performance Comparison

In this last section, the results of the GP-based symbolic regression approach are compared to several state-of-the-art

$$MOS_p = 4.615 - 0.548 \cdot (20 \cdot i\_loss \cdot (1.079 - perc\_pic\_lost) \cdot perc\_pic\_lost + imp\_cons\_slice\_drops \cdot perc\_pic\_lost \cdot p\_loss) \qquad (9)$$

TABLE V

PEARSON LINEAR CORRELATION COEFFICIENT (PLCC), SPEARMAN RANK-ORDER CORRELATION COEFFICIENT (SROCC) AND PREDICTION ERROR $1 - R^2$ USING DIFFERENT MODEL TYPES

| | Training | | | Validation | | | Test | | | All | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Model type | PLCC | SROCC | Error | PLCC | SROCC | Error | PLCC | SROCC | Error | PLCC | SROCC | Error |
| GP metric (4) | 0.9047 | 0.75107 | 0.1814 | 0.8619 | 0.8288 | 0.2571 | 0.9003 | 0.8447 | 0.1895 | 0.8961 | 0.7961 | 0.1969 |
| ANN (42) | 0.9680 | 0.9164 | 0.0629 | 0.8975 | 0.8827 | 0.1945 | 0.8551 | 0.8363 | 0.2688 | 0.9310 | 0.8990 | 0.1333 |
| ANN (8) | 0.9330 | 0.8522 | 0.1294 | 0.8540 | 0.8284 | 0.2707 | 0.8712 | 0.8115 | 0.2411 | 0.9057 | 0.8447 | 0.1797 |
| ANN (4) | 0.9111 | 0.7829 | 0.1699 | 0.8567 | 0.8330 | 0.2661 | 0.8931 | 0.8231 | 0.2023 | 0.8977 | 0.8077 | 0.1941 |
| SVR (42) | 0.9665 | 0.9341 | 0.0660 | 0.9159 | 0.8746 | 0.1612 | 0.9270 | 0.8829 | 0.1407 | 0.9486 | 0.9076 | 0.1002 |
| SVR (8) | 0.9225 | 0.8489 | 0.1490 | 0.8590 | 0.8390 | 0.2621 | 0.9012 | 0.8394 | 0.1879 | 0.9065 | 0.8489 | 0.1783 |
| SVR (4) | 0.9107 | 0.7959 | 0.1706 | 0.8609 | 0.8202 | 0.2588 | 0.8713 | 0.8188 | 0.2408 | 0.8946 | 0.8089 | 0.1998 |

ML techniques. To this end, the same training, validation, and test sets are used as in the previous section. Two different model types are investigated and the results are listed in Table V.

1) The first model type is a two-layer feedforward artificial neural network (ANN) with sigmoid hidden neurons and linear output neurons. It is found that this network topology is able to approximate the nonlinear function $f(\vec{v})$ sufficiently well. Based on experimental results, the number of neurons in the hidden layer is set to 4 and the weights of the neural network are computed with the Levenberg–Marquardt backpropagation algorithm [50].
2) The second model type is the SVR model used by Narwaria *et al.* [27]. Each attribute of the input vector $\vec{v}$ is scaled to [0, 1] and the following model is computed:

$$f(\vec{v}) = \sum_{i=1}^{k}(\alpha_i^* - \alpha_i)K(\vec{v}_i, \vec{v}) + b \qquad (10)$$

where $K$ is chosen to be a radial basis function kernel that maps the problem from a lower dimensional space to a higher feature space and $b$ is a real constant.

$$K(\vec{v}_i, \vec{v}) = \exp(-\gamma \left\| \vec{v}_i - \vec{v} \right\|^2), \gamma > 0 \qquad (11)$$

The variables $\alpha_i$ and $\alpha_i^*$ are optimized by maximizing a constrained quadratic function, and the constants $\gamma$ and $C$ are selected from a grid of increasing values. The reader is referred to [51] for a detailed discussion of the algorithm.

As can be seen from Table V, the accuracy of the GP-based symbolic regression metric [see (9)] yields performance results that are comparable to, or better than the ANN and SVR algorithms. A key advantage of the GP approach is that it provides a natural way for variable selection and yields interpretable models. Discarding redundant variables is important, as it reduces the dimensionality of the problem. Numerical results in Table V confirm that models which are based on the subset of 8 (or even 4) parameters are indeed sufficiently accurate to characterize the perceived quality.

When providing all available parameters to the modeling process, the SVR model achieves a slightly higher accuracy. However, this requires an in-depth processing of the received video stream. This, in turn, increases model complexity. In the case of real-time monitoring, models using parameters which do not require in-depth processing are preferred.

TABLE VI

PERFORMANCE EVALUATION OF OUR PROPOSED METRIC, PSNR AND VQM AGAINST THE EPFL-POLIMI VIDEO DATABASE

| | PLCC | SROCC | Pred. error |
|---|---|---|---|
| GP metric | 0.8816 | 0.8830 | 0.2227 |
| PSNR | 0.7374 | 0.7463 | 0.4562 |
| VQM | 0.8127 | 0.8344 | 0.3395 |

### *F. Model Validation*

The validity of the proposed metric (9) has also been checked by applying it to the publicly available EPFL-PoliMI video quality assessment database [52], [53]. This database contains, 72 4CIF resolution (704x576 pixels) and 72 CIF resolution H.264/AVC encoded video sequences impaired at different packet loss rates. The MOS scores which are predicted by our metric [see (9)] are compared to the MOS scores in the database, and the PLCC, the SROCC and the prediction error $(1 - R^2)$ are computed. The performance of our metric is also compared against two well known FR quality metric, namely PSNR and VQM [23], for benchmarking.

It is seen from Table VI that the metric yields a very good agreement, which confirms that the metric has good generalization properties and that it also works well on similar video sequences. Comparing the performance of our metric against PSNR and VQM measurements show that our NR bitstream-based metric achieves a higher accuracy in estimating perceived quality.

## VI. CONCLUSION

In this paper, a novel machine learning technique for constructing a no-reference bitstream-based objective video quality metric was proposed. Genetic programming-based symbolic regression was used to generate sets of white box models for estimating perceived quality. This, in turn, yielded interpretable models and allowed automatic selection of the most quality-affecting parameters. The modeling technique does not make any *a priori* assumptions on the functional form or the complexity of the final model(s). Since the focus of the paper is a no-reference bitstream-based metric, only parameters that can be extracted from the received encoded video bitstream without the need for complete decoding are taken into account during the modeling process.

In total, 42 different parameters were extracted from the bitstream characterizing the encoding settings, the type of loss and the video content. Based on the variable contribution analysis of the modeling toolkit, it was found that only 20% of these parameters significantly influence perceived quality. Modeling results confirmed that the perceived quality can be estimated accurately using only a very limited number of parameters. This enables real-time no-reference-based objective video quality monitoring for video service providers.

## References

[1] N. Staelens, S. Moens, W. Van den Broeck, I. Mariën and, B. Vermeulen, P. Lambert, R. Van de Walle, and P. Demeester, "Assessing quality of experience of IPTV and video on demand services in real-life environments," *IEEE Trans. Broadcast.*, vol. 56, no. 4, pp. 458–466, Dec. 2010.

[2] N. Staelens, K. Casier, W. Van den Broeck, B. Vermeulen, and P. Demeester, "Determining customer's willingness to pay during in-lab and real-life video quality evaluation," in *Proc. 6th Int. Workshop VPQM*, Jan. 2012.

[3] G. Cermak, "Consumer Opinions about frequency of artifacts in digital video," *IEEE J Selected Topics Signal Process.*, vol. 3, no. 2, pp. 336–343, Apr. 2009.

[4] DSL Forum Technical Report TR-126, "Triple-play services quality of experience (QoE) requirements," DSL Forum, 2006.

[5] M. Pinson and S. Wolf, "Comparing subjective video quality testing methodologies," Proc. SPIE, vol. 5150. no. 1. 2003, pp. 573–582.

[6] ITU-T Recommendation P.910, *Subjective Video Quality Assessment Methods for Multimedia Applications,* ITU, 1999.

[7] ITU-R Recommendation BT.500, *Methodology for the Subjective Assessment of the Quality of Television Pictures,* ITU, 2009.

[8] Z. Wang, H. R. Sheikh, and A. C. Bovik, "Objective video quality assessment," in *The Handbook of Video Databases: Design and Applications*, B. Furht and O. Marqure, Eds. Boca Raton, FL, USA: CRC Press, 2003, pp. 1041–1078.

[9] ITU-T Recommendation J.341, *Objective Perceptual Multimedia Video Quality Measurement of HDTV for Digital Cable Television in the Presence of a Full Reference,* ITU, 2011.

[10] ITU-T Recommendation J.342, *Objective Multimedia Video Quality Measurement of HDTV for Digital Cable Television in the Presence of a Reduced Reference Signal,* ITU, 2011.

[11] M. Naccari, M. Tagliasacchi, and S. Tubaro, "No-reference video quality monitoring for H.264/AVC coded video," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 932–946, Aug. 2009.

[12] S. Chikkerur, V. Sundaram, M. Reisslein, and L. J. Karam, "Objective video quality assessment methods: A classification, review, and performance comparison," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 165–182, Jun. 2011.

[13] Video Quality Experts Group (VQEG), "Final report from the Video Quality Experts Group on the validation of objective models of video quality assessment, Phase II," Aug. 2003 [Online]. Available: http://www.its.bldrdoc.gov/vqeg/project-pages/frtv-phase-ii/frtv-phase-ii.aspx

[14] Video Quality Experts Group (VQEG), *Final Report of VQEG's Multimedia Phase I Validation Test,* Sep. 2008 [Online]. Available: http://www.its.bldrdoc.gov/vqeg/project-pages/multimedia-phase-i/multimedia-phase-i.aspx

[15] Video Quality Experts Group (VQEG), *Report on the Validation of Video Quality Models for High Definition Video Content,* Jun. 2010 [Online]. Available: http://www.its.bldrdoc.gov/vqeg/projects/hdtv/

[16] S. Mohamed, G. Rubino, H. Afifi, and F. Cervantes, "Real-time video quality assessment in packet networks: A neural network model," in *Proc. Int. Conf. Parallel Distributed Process. Techniques Appl.*, Jun. 2001.

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

[18] J. Choe, K. Lee, and C. Lee, "No-reference video quality measurement using neural networks," in *Proc. 16th Int. Conf. Digital Signal Process.*, Jul. 2009, pp. 1–4.

[19] P. Gastaldo, S. Rovetta, and R. Zunino, "Objective quality assessment of MPEG-2 video streams by using CBP neural networks," *IEEE Trans. Neural Netw.*, vol. 13, no. 4, pp. 939–947, Jul. 2002.

[20] 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.

[21] H. El Khattabi, A. Tamtaoui, and D. Aboutajdine, "Video quality assessment measure with a neural network," *Int. J. Comput. Inform. Eng.*, vol. 4, no. 3, pp. 167–171, 2010.

[22] B. Wang, D. Zou, and R. Ding, "Support vector regression based video quality prediction," in *Proc. IEEE ISM*, Dec. 2011, pp. 476–481.

[23] ITU-T Recommendation J.144, *Objective Perceptual Video Quality Measurement Techniques for Digital Cable Television in the Presence of a Full Reference,* ITU, 2004.

[24] N. Narvekar, T. Liu, D. Zou, and J. Bloom, "Extending G.1070 for video quality monitoring," in *Proc. IEEE ICME*, Jul. 2011, pp. 1–4.

[25] M. Narwaria and W. Lin, "Machine learning based modeling of spatial and temporal factors for video quality assessment," in *Proc. Int. Conf. Image Process.*, vol. 1. Sep. 2011, pp. 2513–2516.

[26] M. Narwaria, W. Lin, and L. Anmin, "Low-complexity video quality assessment using temporal quality variations," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 525–535, Jun. 2012.

[27] M. Narwaria and W. Lin, "Svd-based quality metric for image and video using machine learning," *IEEE Trans. Systems, Man, Cybern.—Part B: Cybern.*, vol. 42, no. 2, pp. 347–364, Apr. 2012.

[28] S. Argyropoulos, A. Raake, M.-N. Garcia, and P. List, "No-reference bit stream model for video quality assessment of H.264/AVC video based on packet loss visibility," in *Proc. IEEE ICASSP*, May 2011, pp. 1169–1172.

[29] S. Argyropoulos, A. Raake, M. Garcia, and P. List, "No-reference video quality assessment for sd and hd H.264/AVC sequences based on continuous estimates of packet loss visibility," in *Proc. 3rd Int. Workshop QoMEX*, Sep. 2011, pp. 31–36.

[30] A. Reibman, S. Kanumuri, V. Vaishampayan, and P. Cosman, "Visibility of individual packet losses in MPEG-2 video," in *Proc. Int. Conf. Image Process.*, vol. 1. Oct. 2004, pp. 171–174.

[31] S. Kanumuri, P. Cosman, A. Reibman, and V. Vaishampayan, "Modeling packet-loss visibility in MPEG-2 video," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 341–355, Apr. 2006.

[32] V. Menkovski, G. Exarchakos, and A. Liotta, "Machine learning approach for quality of experience aware networks," in *Proc. 2nd Int. Conf. INCOS*, Nov. 2010, pp. 461–466.

[33] N. Staelens, G. Van Wallendael, K. Crombecq, N. Vercammen, J. De Cock, B. Vermeulen, R. Van de Walle, T. Dhaene, and P. Demeester, "No-reference bitstream-based visual quality impairment detection for high definition H.264/AVC encoded video sequences," *IEEE Trans. Broadcast.*, vol. 58, no. 2, pp. 187–199, Jun. 2012.

[34] K. Vladislavleva, K. Veeramachaneni, M. Burland, J. Parcon, and U.-M. O'Reilly, "Knowledge mining with genetic programming methods for variable selection in flavor design," in *Proc. 12th Annu. Conf. GECCO*, 2010, pp. 941–948.

[35] P. Gastaldo and J. A. Redi, "Machine learning solutions for objective visual quality assessment," in *Proc. 6th Int. Workshop VPQM*, Jan. 2012.

[36] Evolved Analytics LLC, *DataModeler Release 8.0 Documentation*. Evolved Analytics LLC, 2010 [Online]. Available: http://www.evolved-analytics.com

[37] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection* (A Bradford Book Series). Cambridge, MA, USA, London: The MIT Press, 1992.

[38] M. Pinson, S. Wolf, N. Tripathi, and C. Koh, "The consumer digital video library," in *Proc. 5th Int. Workshop VPQM*, Jan. 2010.

[39] G. E. Legge and J. M. Foley, "Contrast masking in human vision," *J. Opt. Soc. America*, vol. 70, no. 12, pp. 1458–1471, Dec. 1980.

[40] Y. Zhong, I. Richardson, A. Sahraie, and P. McGeorge, "Influence of task and scene content on subjective video quality," in *Proc. Image Anal. Recognit.*, LNCS 3211. 2004, pp. 295–301.

[41] A. Ostaszewska and R. Kloda, "Quantifying the amount of spatial and temporal information in video test sequences," in *Recent Advances in Mechatronics*. Berlin, Heidelberg: Springer, 2007, pp. 11–15.

[42] G. O'Driscoll, *Next Generation IPTV Services and Technologies*. New York, USA: Wiley-Interscience, 2008.

[43] A. Rombaut, N. Staelens, N. Vercammen, B. Vermeulen, and P. Demeester, "xStreamer: Modular Multimedia Streaming," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 929–930.

[44] N. Staelens, N. Vercammen, Y. Dhondt, B. Vermeulen, P. Lambert, R. Van de Walle, and P. Demeester, "ViQID: A no-reference bit stream-based visual quality impairment detector," in *Proc. 2nd Int. Workshop QoMEX*, Jun. 2010, pp. 206–211.

[45] N. Staelens and G. Van Wallendael, "Adjusted JM Reference Software 16.1 with XML Tracefile Generation Capabilities," VQEQ_JEG_Hybrid_2011_029_jm with xml tracefile_v1.0, Hillsboro, OR, USA, Dec. 2011.

[46] G. Smits and F. Kotanchek, *Pareto-Front Exploitation in Symbolic Regression* (Genetic Programming Theory and Practice II Series). Berlin: Springer, 2004, ch. 17, pp. 283–299.

[47] P. Corriveau, *Video Quality Testing*, (Digital Video Image Quality and Perceptual Coding Series). Boca Raton, FL, USA: CRC Press, 2006, ch. 4, pp. 125–153.

[48] Q. Huynh-Thu, M.-N. Garcia, F. Speranza, P. Corriveau, and A. Raake, "Study of rating scales for subjective quality assessment of high-definition video," *IEEE Trans. Broadcast.*, vol. 57, no. 1, pp. 1–14, Mar. 2011.

[49] N. Staelens, B. Vermeulen, S. Moens, J.-F. Macq, P. Lambert, R. Van de Walle, and P. Demeester, "Assessing the influence of packet loss and frame freezes on the perceptual quality of full length movies," in *Proc. 4th Int. Workshop VPQM*, Jan. 2009.

[50] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA: MIT Press, 2002.

[51] S. Haykin, *Neural Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[52] F. De Simone, M. Naccari, M. Tagliasacchi, F. Dufaux, S. Tubaro, and T. Ebrahimi, "Subjective assessment of H.264/AVC video sequences transmitted over a noisy channel," in *Proc. 1st Int. Workshop QoMEX*, Jul. 2009, pp. 204–209.

[53] F. De Simone, M. Tagliasacchi, M. Naccari, S. Tubaro, and T. Ebrahimi, "A H.264/AVC video database for the evaluation of quality metrics," in *Proc. IEEE ICASSP*, Mar. 2010, pp. 2430–2433.

**Nicolas Staelens** (A'11) received the Master's degree in computer science from Ghent University, Ghent, Belgium, in 2004.

He started his career in 2004 as a Research and Development Engineer with Televic (a Belgian company that develops, designs, and manufactures high-end network systems and software applications for the healthcare market). In 2006, he joined the IBCN-group (Internet Based Communication Networks and Services) at Ghent University—iMinds as a Ph.D. student. His current research interests include studying the effects of network impairments on the perceived quality of audiovisual sequences. As of 2007, he was also actively participating within the Video Quality Experts Group (VQEG) and is currently Co-Chair of the Tools and Subjective Labs support group and the JEG-Hybrid project.

**Dirk Deschrijver** (M'08) was born in Tielt, Belgium, on September 26, 1981. He received the Master's (Licentiaat) degree and the Ph.D. degree in computer science from the University of Antwerp, Antwerp, Belgium, in 2003 and 2007, respectively.

From May to October 2005, he was a Marie Curie Fellow with the Scientific Computing Group, the Eindhoven University of Technology, Eindhoven, The Netherlands. He is currently an FWO Post-Doctoral Research Fellow with the Department of Information Technology (INTEC), iMinds, Ghent University, Gent, Belgium. His current research interests include robust parametric macromodeling, rational least-squares approximation, orthonormal rational functions, system identification, and broadband macromodeling techniques.

**Ekaterina Vladislavleva** (S'05-M'08) received the Master of Science degree in mathematics (in mathematical theory of intelligent systems) from the Moscow State University of Lomonosov, Moscow, Russia, the Ph.D. degree in symbolic regression from Tilburg University, The Netherlands, and the Professional Doctorate degree in engineering (in industrial mathematics) from the Eindhoven University of Technology, Eindhoven, The Netherlands.

She is a Chief Data Scientist and Partner at Evolved Analytics and a Co-Owner of Evolved Analytics Europe. Her current research interests include data-driven modeling, high-performance computing, and industrial optimization, particularly, in the industrial scale data analysis and predictive modeling for process and research analytics.

**Brecht Vermeulen** received the electronic engineering degree from Ghent University, Ghent, Belgium in 1999. He received the Ph.D. degree for the work entitled "Management architecture to support quality of service in the internet based on IntServ and DiffServ domains" from the Department of Information Technology, Ghent University, Ghent, in 1999.

Since June 2004, he has lead a research team within the IBCN group led by Prof. Demeester who investigates network and server performance and quality of experience in the fields of video, audio/voice, and multiple play. Since the start of iMinds in 2004, he has also lead the iMinds Technical Test Centre in Ghent, Belgium.

**Tom Dhaene** (M'94–SM'05) was born in Deinze, Belgium, on June 25, 1966. He received the Ph.D. degree in electrotechnical engineering from the University of Ghent, Ghent, Belgium, in 1993.

From 1989 to 1993, he has been a Research Assistant with the Department of Information Technology, University of Ghent, Ghent, where his research focused on different aspects of full-wave electromagnetic circuit modeling, transient simulation, and time-domain characterization of high-frequency and high-speed interconnections. In 1993, he joined the EDA company Alphabit (now part of Agilent). He was one of the key developers of the planar EM simulator ADS Momentum. Since September 2000, he has been a Professor with the Department of Mathematics and Computer Science, the University of Antwerp, Antwerp, Belgium. Since October 2007, he has been a Full Professor with the Department of Information Technology (INTEC), Ghent University, Ghent, Belgium. As author or co-author, he has contributed to more than 220 peer-reviewed papers and abstracts in international conference proceedings, journals, and books.

**Piet Demeester** (F'09) is a Professor in the faculty of Engineering at Ghent University.

He is Head of the research group "Internet Based Communication Networks and Services" (IBCN) that is part of the Department of Information Technology (INTEC) of Ghent University. He is also leading the Future Internet (Networks, Media and Service) Department of iMinds.