

Deriving video content type from HEVC bitstream semantics

James Nightingale^a, Qi Wang^a, Christos Grecos^a and Sergio R. Goma^b

^aCentre for Audio-Visual Communications and Networks, School of Computing, University of the West of Scotland, Paisley, PA1 2BE, United Kingdom; ^bMultimedia R&D, Qualcomm Inc., San Diego CA 92121, United States of America

ABSTRACT

As network service providers seek to improve customer satisfaction and retention levels, they are increasingly moving from traditional quality of service (QoS) driven delivery models to customer-centred quality of experience (QoE) delivery models. QoS models only consider metrics derived from the network however, QoE models also consider metrics derived from within the video sequence itself. Various spatial and temporal characteristics of a video sequence have been proposed, both individually and in combination, to derive methods of classifying video content either on a continuous scale or as a set of discrete classes. QoE models can be divided into three broad categories, full reference, reduced reference and no-reference models. Due to the need to have the original video available at the client for comparison, full reference metrics are of limited practical value in adaptive real-time video applications. Reduced reference metrics often require metadata to be transmitted with the bitstream, while no-reference metrics typically operate in the decompressed domain at the client side and require significant processing to extract spatial and temporal features.

This paper proposes a heuristic, no-reference approach to video content classification which is specific to HEVC encoded bitstreams. The HEVC encoder already makes use of spatial characteristics to determine partitioning of coding units and temporal characteristics to determine the splitting of prediction units. We derive a function which approximates the spatio-temporal characteristics of the video sequence by using the weighted averages of the depth at which the coding unit quadtree is split and the prediction mode decision made by the encoder to estimate spatial and temporal characteristics respectively. Since the video content type of a sequence is determined by using high level information parsed from the video stream, spatio-temporal characteristics are identified without the need for full decoding and can be used in a timely manner to aid decision making in QoE oriented adaptive real time streaming.

Keywords: Video streaming, HEVC, QoE, content type classification.

1. INTRODUCTION

The use of video related applications has grown exponentially in recent years and now accounts for a large proportion of all Internet traffic. Current forecasts¹ predict that this growth will continue unabated in the coming years. In such a challenging, highly competitive and fast moving market sector, content and service providers must continually introduce innovative services and improve current offerings. Given the proliferation of analogous services from many competing companies, service providers must also differentiate themselves by meeting or exceeding customer's expectations of quality and service. Increasingly, both network operators and content owners are turning to a quality approach which focuses on the customer experience rather than traditional models of service delivery². The problem of finding new models to accurately reflect the customer's Quality of Experience (QoE), although widely investigated, remains a significant challenge for the research community.

Existing Quality of Service (QoS) models can be considered to be network-centric or technology-centric^{3, 4}, where network resource allocation for a particular service or application is implemented as a function mapping network parameters of bandwidth, packet loss, delay and jitter to application/service requirements. QoS models are typically implemented by network operators to guarantee availability, reliability and integrity of services or applications with little regard to the customer's subjective perception of quality². This approach to quality provision dates from Internet Engineering Task Force (IETF) initiatives of the late 1990's³ and is now insufficient to meet the expectations of modern consumers in this age of ubiquitous Internet access and extremely capable consumer devices.

In 2007 the ITU-T defined QoE as “The overall acceptability of an application or service, as perceived subjectively by the end user”⁵. Although some research suggests that this definition should also include objective human-centric factors^{4,6}; the vast majority of recent QoE related research has concentrated on subjective perception of quality for multimedia (video, audio, voice etc.) applications. Existing approaches to designing QoE-aware video streaming metrics can be loosely split into two main categories of parametric models, those where a generalized linear model^{6,7,8} is employed and those that are modelled on the human vision system^{9,10}. Irrespective of which approach is taken, many QoE proposals have recognized the importance of video content type in subjective perception of video quality. The content type of video sequences is typically determined by extracting spatial and/or temporal features. Sequences with similar characteristics are grouped together, often using a clustering technique. Most approaches to content type classification have extracted features from the decompressed domain¹¹. Those proposals that have extracted information from the compressed domain¹² have been based on either the H.264 Advanced Video Coding standard (H.264/AVC)¹³ or older video compression standards. Another way in which service providers may seek to address the issues arising from this predicted increase in video traffic is to move towards adoption of High Efficiency Video Coding (HEVC), the new H.265¹⁴ video compression standard ratified by the ITU-T in 2013. HEVC/H.265 greatly reduces, by up to 50%, the bandwidth needed to store and transmit video content without any noticeable reduction in quality. It has been specifically designed to provide efficient compression of video content for ultra-high definition (UHD) 4k and 8k displays that are starting to become available in high-end consumer products. Although HEVC/H.265 is a potentially valuable component of future QoE-aware video streaming mechanisms, this use for the new standard is an emerging research area. Specifically, no published work has, as yet, investigated the extraction of spatial or temporal features from the compressed domain of the new HEVC/H.265 standard for the purpose of determining video content type.

In this paper we consider the problem of defining a video content type classification method for HEVC/H.265 encoded bitstreams suitable for deployment as part of a QoE-aware streaming system for real-time applications. Any viable solution to this problem needs to meet real-time constraints and must therefore have low complexity. Our proposed solution makes use of information contained in the compressed bitstream to approximate the spatial and temporal characteristics of the video sequence. The information is obtained by parsing an HEVC/H.265 bitstream, without the need for full decoding. We have fully implemented and evaluated the proposed method on a testbed. Results show that the scheme provides a fast method of content type classification suitable for use with parametric no-reference QoE models for real-time applications. It is also suitable for use at both client nodes and intermediate network nodes. The complexity is low, taking less than 20% of the time required to decode a sequence and is suitable to support timely decision making in real time applications.

The rest of this paper is organized as follows. In Section 2 we discuss HEVC/H.265 video compression and related work on both QoE aware video streaming and video content type classification. The proposed method of video content classification for HEVC and its implementation are described in Section 3 with results and analysis presented in Section 4. Finally Section 5 concludes the paper.

2. BACKGROUND AND RELATED WORK

2.1 High Efficiency Video Coding

HEVC can achieve bitrate reductions of up to 50% over the current H.264/AVC standard. These savings are achieved by employing variable-sized coding units and new, improved coding tools. The main features of HEVC/H.265 that differentiate it from previous encoding schemes include flexible coding block and transform block partitioning¹⁵. A more flexible approach to prediction mode decision making has also been adopted¹⁴. Mode and motion vector signaling, interpolation, de-blocking and prediction tools have all been improved in HEVC/H.265¹⁴. The proposed scheme, described in Section 3 of this paper, relies heavily on information derived from partitioning decisions made by the HEVC/H.265 encoder. Consequently, the basic operation of each of the partitioning mechanisms employed by the HEVC/H.265 encoder is described in this overview of the standard.

An HEVC/H.265 encoder partitions each picture into Coding Tree Units (CTU), the dimensions of which can be either application or encoder architecture specific. CTU dimensions are signaled in a Sequence Parameter Set (SPS) message within the bitstream and are the same for all CTU's in a video sequence. CTU's are square in shape and may have dimensions of 64x64, 32x32 or 16x16 luma samples. Generally choosing larger CTU sizes increases compression efficiency. The Coding Tree Unit comprises of a luminance Coding Tree Block (CTB) and two chroma CTB's, one for each of the two chroma components (Cr and Cb). Each CTB has a quadtree structure in which the CTB can be either be

kept as a single Coding Block (CB) or split into 4 equal subdivisions. Therefore the largest coding block size possible is equal to the CTB size. This process can be repeated until the minimum CB size of 8x8 luma samples is reached. An example of the quadtree structure is given in Figure 1(a) and the resultant partitioning of the CTB is shown in Figure 1(b). The CTB in this example has dimensions of 64x64 with a quadtree depth of 4. The lowest 'leaves' in the quadtree are minimum sized (8x8) CB's.

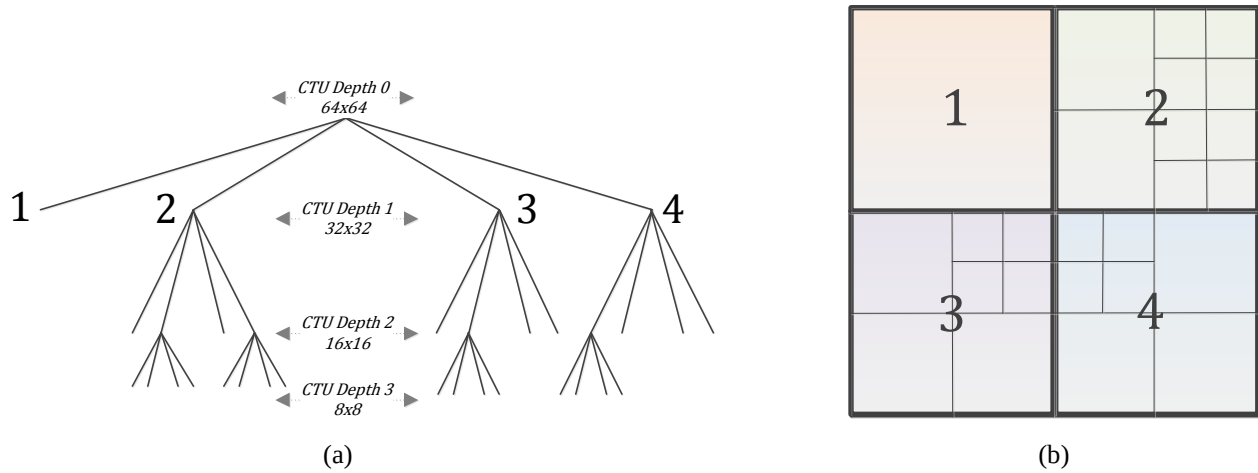


Figure 1. An example of code tree block partitioning in HEVC illustrating the quadtree hierarchical structure (a) and the resulting subdivisions within the CTB.

In HEVC the combination of a luma CB and its associated chroma CB's is called a coding unit (CU). CU dimensions are determined by the characteristics of the area of the picture contained in a CTB and are highly content dependent. HEVC/H.265 encoders perform an exhaustive search of all possible splitting combinations for each CTB where the rate distortion cost of keeping the larger coding block size is compared to that of further subdividing at each level of the quadtree. Homogenous areas of a picture (large areas with little activity) are encoded at low depths on the quadtree with large CU dimensions. Areas of a picture containing high levels of motion, texture or detail are encoded at deeper levels in the quadtree and have smaller CU dimensions. Each CU is split into one, two or four Prediction Units (PU). PU's are the units from which motion vectors are predicted and signaled. HEVC supports intra, uni-directional inter, shown in Figure 2(a) and bi-directional inter prediction shown in Figure 2(b). PU's must be rectangular in shape with boundaries chosen to match those of real objects in the picture. All of the possible PU partitioning options for a CU are shown in Figure 3.

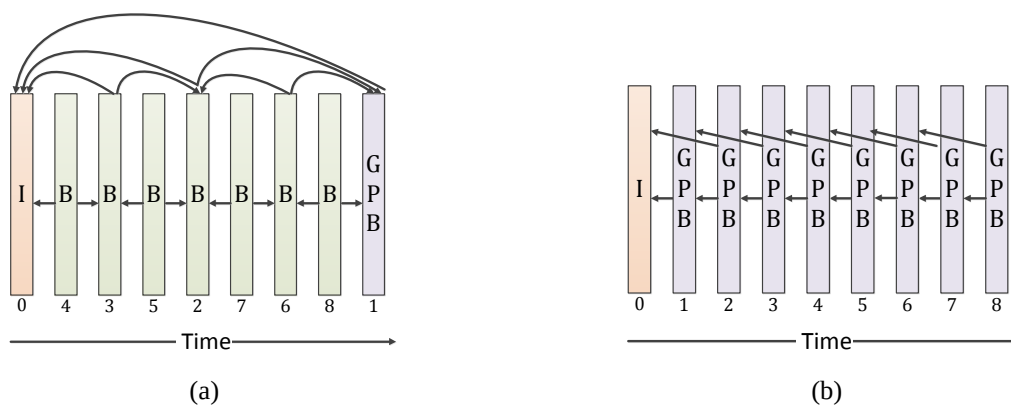


Figure 2. Inter prediction in HEVC/H.265. Bi-prediction is used in the Random Access configuration (a) and uni-prediction in the Low Delay configuration (b).

Transform Units (TU) are the third structure in HEVC. They consist of both square shaped luma and chroma transform blocks (TB) which may be split using a residual quadtree structure. The possible TB sizes 32x32, 16x16, 8x8 or 4x4 luma samples.

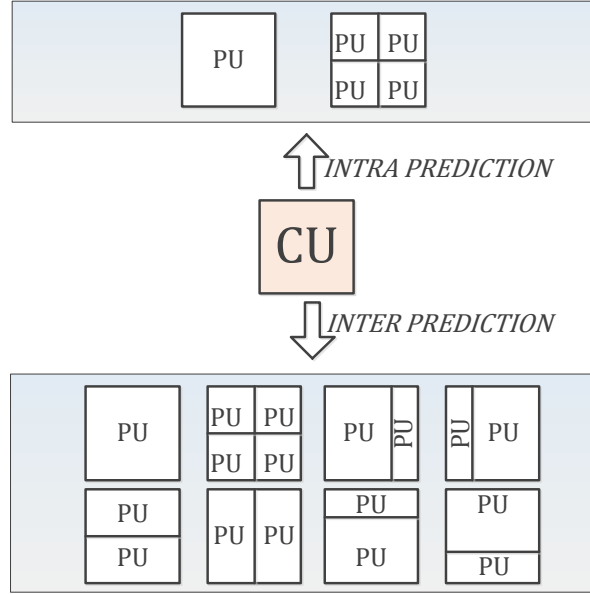


Figure 3. Possible prediction modes for each CU in an HEVC encoded stream.

2.2 QoE-Aware Video Streaming

In video streaming applications a user's Quality of Experience is generally interpreted as his or her perceptual evaluation of the quality of a video presented to them. Numerous models have proposed a means of estimating the quality of a video by using parameters obtained from the encoder, video stream and transmission network. These models fall into three categories. Full Reference schemes where the original video is directly compared to the received and decoded video are generally not very useful in real time situations as the original video is not available at the client. The MOVIE index⁹ is a prominent example of a full reference metric where both spatial and temporal components are considered in a perceptual Human Vision system (HVS) inspired approach. In addition to requiring the original video for comparison this method is also computationally intensive. Many full reference QoE schemes have been proposed that are based on the natural vision characteristics exploited in the Structural Similarity (SSIM) metric¹⁶ and its derivatives, Multi-Scale structural Similarity (MS-SSIM)¹⁷ and Video Structural Similarity (V-SSIM)¹⁸. Reduced reference and no-reference QoE models are more practical for use in real time situations. Reduced reference metrics such as the ITU-T Rec J.246¹⁹, extract some spatial or temporal features from both the original and the degraded video to produce a quality index or score. The extracted features from the original video may be carried as metadata with the encoded stream to facilitate comparison at the receiving node²⁰. No-reference metrics do not require access to the original video to perform quality assessment. Some no reference approaches²¹ use such as blocking and blurring present in the decompressed distorted image to estimate perceptual visual quality. Other no reference approaches such as those proposed by Lin et al²² and Raake et al.²³ use Quantisation Parameter (QP) and motion vector information extracted from the compressed bitstream to estimate video quality. The currently available QoE models are either general models that compare only the original to the decompressed received video or are based on either MPEG2, MPEG4 or H.264/AVC encoded bitstreams. No methods specifically targeting the compressed domain of the new HEVC/H.265 have been published.

2.3 Video Content Type Classification

Many of the QoE-aware parameter driven approaches to video streaming proposed in literature place emphasis on the significance of the content of a video sequence in human perception of quality. The methods used to classify video sequences into categories with similar content types have largely made use of spatial and temporal characteristics, such as motion vectors, extracted from the video sequence. Reis et al.²⁴ proposed a method of determining video content type

from a compressed bitstream. In this proposal pictures between two scene changes are assessed, motion vector and colour parameters are extracted from the bitstream and the content type grouped into one of five classes. Joskowicz et al.²⁵ employse the sum of the per-pixel absolute differences to classify content types and Khan et al.²⁶ group content into three types based on spatio-temporal features of edges, blurring and blocking artifacts extracted from the degraded video at the receiver. A K-means clustering approach is used in this paper to assign content type classifications.

3. PROPOSED SCHEME

3.1 Overview

In this work we propose a method of determining video content type classification from the compressed domain of HEVC/H.265 bitstreams. Our method explores the relationship between the encoder decision making process and the inherent characteristics of the video being encoded. It has been widely reported¹⁵ that HEVC/H.265 encoder's CTU splitting and prediction mode decisions are highly content dependent. Our method uses this relationship between content type and mode decisions to infer the content type of a video sequence from mode decision data contained in the compressed HEVC/H.265 bitstream. When an HEVC encoder performs its exhaustive searches of all possible combinations to determine Coding Unit size, Prediction Unit size and shape and Transform Unit size, the results of these searches are carried within syntax elements of the encoded bitstream. These syntax elements can be parsed from the bitstream without the need for full decoding. The proposed method consists of two components, a CTU quadtree depth component and a prediction component.

3.2 Coding Tree Depth Component

The spatial complexity of each picture within a video sequence is reflected in the depth level chosen by the encoder when partitioning each CTU in the picture. Large homogenous areas will be encoded at minimum CTU depths. Areas containing high detail, rich texture, edges of objects etc.; will be encoded at deeper depths in the quadtree using smaller areas. Figure 4 shows the partitioning decisions for the 6th frame of the (a) BlowingBubbles and (b) BasketballDrill sequences respectively. Both sequences were encoded using the Low Delay configuration of HEVC with uni-directional inter prediction and a QP of 22. The maximum CU size was set at 64x64 and the maximum quadtree depth set to 4. From the figure it can be seen that the BasketballDrill sequence has many more CTU's encoded at the minimum depth level of the quadtree (CTU depth = 0) with a CU size of 64x64 than the BlowingBubbles sequence does. This reflects the difference in spatial complexity between the sequences which can be explained by the large homogenous areas of basketball court, containing less spatial detail, in the BasketballDrill sequence.

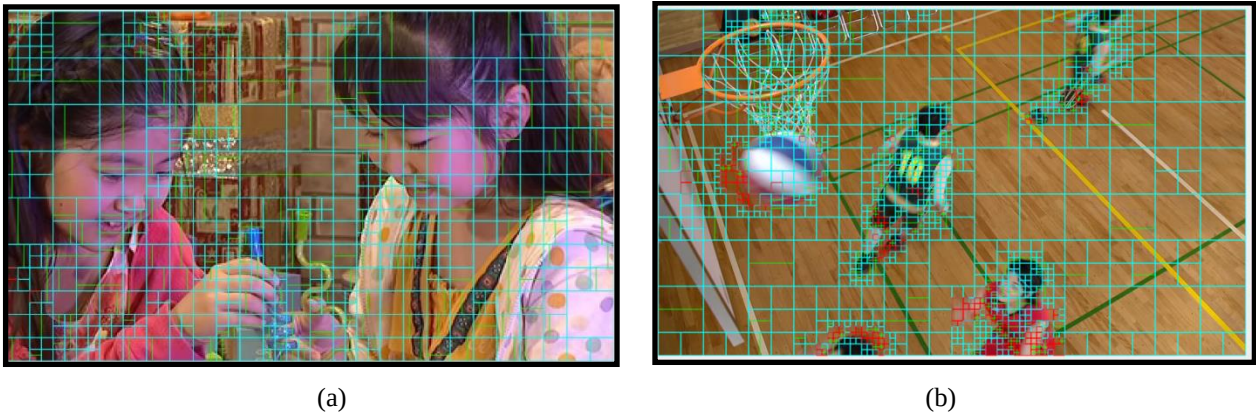


Figure 4. A comparison of HEVC coding tree splitting decisions for uni-directional inter predicted between (a) the BlowingBubbles sequence and (b) the BasketballDrill sequence.

The spatial component of the proposed scheme counts the percentage of CU's encoded at each depth level of the CTU quadtree structure. Depth level 0 represents the Largest Coding Unit (LCU) size of 64x64 luma samples and depth level 3 represents the smallest coding unit size of 8x8 luma samples.

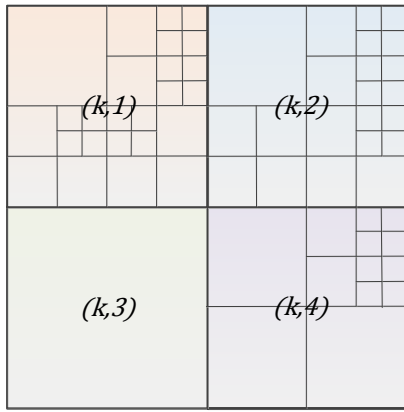
At each depth level i of CTU j of picture k in a video sequence, $a(i,j,k)$ is the actual number of CU's encoded at that level. $m(i,j,k)$ is the maximum number of CU's that could be encoded at that level and $r(i,j,k)$ the ratio of CU's encoded at depth i . In a quadtree structure where each level can be split into four subdivisions the maximum number of CU's is expressed by the formula shown in Equation 1.

$$m(i,j,k) = 4^i \quad (1)$$

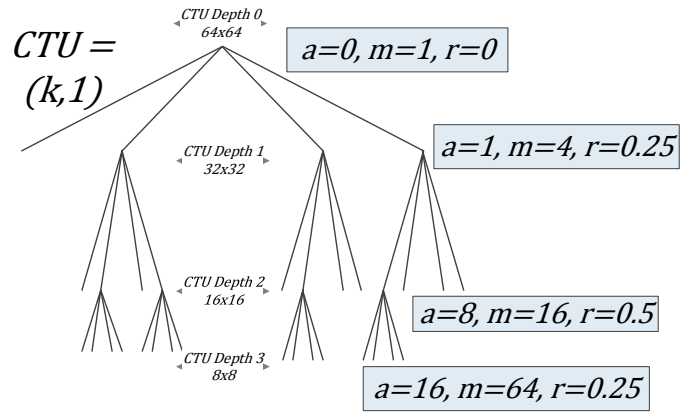
The ratio of CU's encoded at level i if the quadtree structure is expressed by Equation 2.

$$r(i,j,k) = a(i,j,k) / m(i,j,k) \quad (2)$$

A simple example illustrating the principle is given in Figure 5. A picture k , shown in Figure 5(a) has been partitioned into 4 CTU's each with a maximum size of 64x64 and a maximum partitioning depth of 4. The quadtree structure for CTU $(k,1)$ is shown in Figure 5(b). In this example the CTU has been split, therefore no CU's were encoded at quadtree depth 0 with an LCU size of 64x64 luma samples. At depth level 1, only one of the four subdivisions is encoded with a CU size of 32x32 luma samples. The ratio of CU's encoded at level 1 is therefore 0.25. Actual CU's, maximum CU's and the encoded ratio is also shown for depth levels 2 and 3. The values of the CU encoding ratios at every quadtree depth level for each of the CTU's shown in Figure 5(a) are given in Table 1. The average encoding CU ratio at each quadtree depth level for picture k is also shown in Table 1. In our proposed scheme these average values are weighted to represent the spatial complexity contributed by each quadtree level.



(a)



(b)

Figure 5. An example of deriving the spatial complexity component using CTU depth information from the encoded bitstream. The picture (a) is split into 4 CTU's with the quadtree structure (b) and derived encoding ratios for the upper left CTU $(k, 1)$ illustrated.

Table 1. Encoding ratios for each depth of the CTU's shown in Figure 5.

	$r(0)$	$r(1)$	$r(2)$	$r(3)$
CTU $(k,1)$	0.00	0.25	0.50	0.25
CTU $(k,2)$	0.00	0.25	0.56	0.19
CTU $(k,3)$	1.00	0.00	0.00	0.00
CTU $(k,4)$	0.00	0.75	0.00	0.25
Picture (k)	0.25	0.31	0.30	0.14

Figure 6 provides an example of an individual CTU taken from picture shown in Figure 4(b). In Figure 6(a) the area covered by the CTU is shown, in this illustration PU partitioning of each CU is also shown, while Figure 6(b) shows a detailed view of the area covered by the CTU with only the CU partitioning decisions shown.

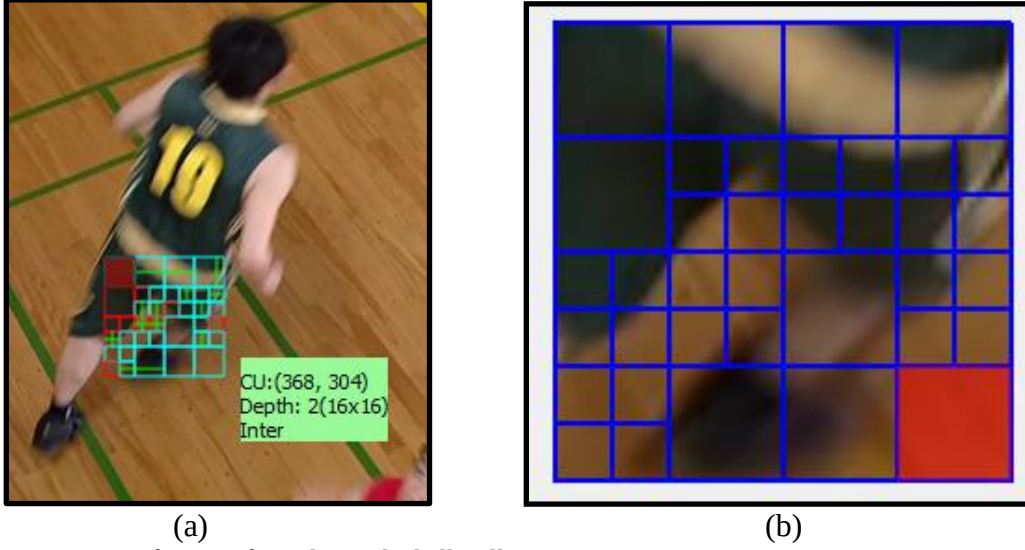


Figure 6. Partitioning of a CTU from the BasketballDrill sequence.

The spatial complexity, derived from quadtree information, $\gamma(k)$ of picture k is derived from the sum of the weighted averages of the CU encoding ratios at each quadtree depth, as shown in Equation 3. To obtain values of α , β , ρ and δ the contribution made by each depth level to the spatial perceptual information measurement (SI) from ITU-T REC 910²⁷ for each picture in the test sequences was calculated. The metric coefficients were calculated from the SI values of the 2280 pictures contained in the test sequences used in our experiments. The values for the coefficients are $\alpha = 0.113$, $\beta = 0.162$, $\rho = 1.42$ and $\delta = 3.11$, which indicates that areas encoded at the deepest levels of the quadtree contribute to spatial complexity to a much greater extent than those encoded at the highest levels.

$$\gamma(k) = \alpha \times r(0) + \beta \times r(1) + \rho \times r(2) + \delta \times r(3) \quad (3)$$

The spatial complexity S of a sequence consisting of N pictures is expressed in Equation 4.

$$S = \frac{\sum_{n=1}^N \gamma(n)}{N} \quad (4)$$

Where S is higher when the number of CU's encoded at deeper depths is higher, indicating a sequence containing picture with high detail and rich texture.

3.3 Temporal Prediction Component

The prediction component of our scheme is based on the following empirical observations of HEVC/H.265 encoder behavior in the Low Delay and Random Access configuration modes. Firstly we have observed that the edges of moving objects, particularly those objects that are moving quickly in relation to the background, are encoded using Intra prediction, whilst the body of the same objects is encoded using Inter prediction. Secondly we have observed that there is a relationship between the ratio of Intra encoded PU's and the number of Inter encoded PU's in an Inter encoded picture. This relationship varies in line with the number of moving objects or 'focal points' in the picture. This observation is illustrated for the BasketballDrill (a) and BlowingBubbles (b) sequences in Figure 7. In each case the upper image includes an overlay showing the direction and magnitude of the motion vectors and lower image shows the encoder prediction decision modes. In the lower image areas containing Intra encoded PU's are circled. In the BasketballDrill

sequence the ball and players are moving quickly in relation to the background, edges of each focal point are encoded using Intra prediction. In the picture shown (frame 6) 18.47% of all PU's are Intra encoded, 96% of which are encoded at the lowest (8x8) quadtree depth level. In the BlowingBubbles (frame 2), there is very little movement. Only a single bubble is floating gently in the top left of the picture and there is some minor movement of the apparatus used to generate the bubbles. This example has only 2.23% of PU's encoded using Intra prediction.

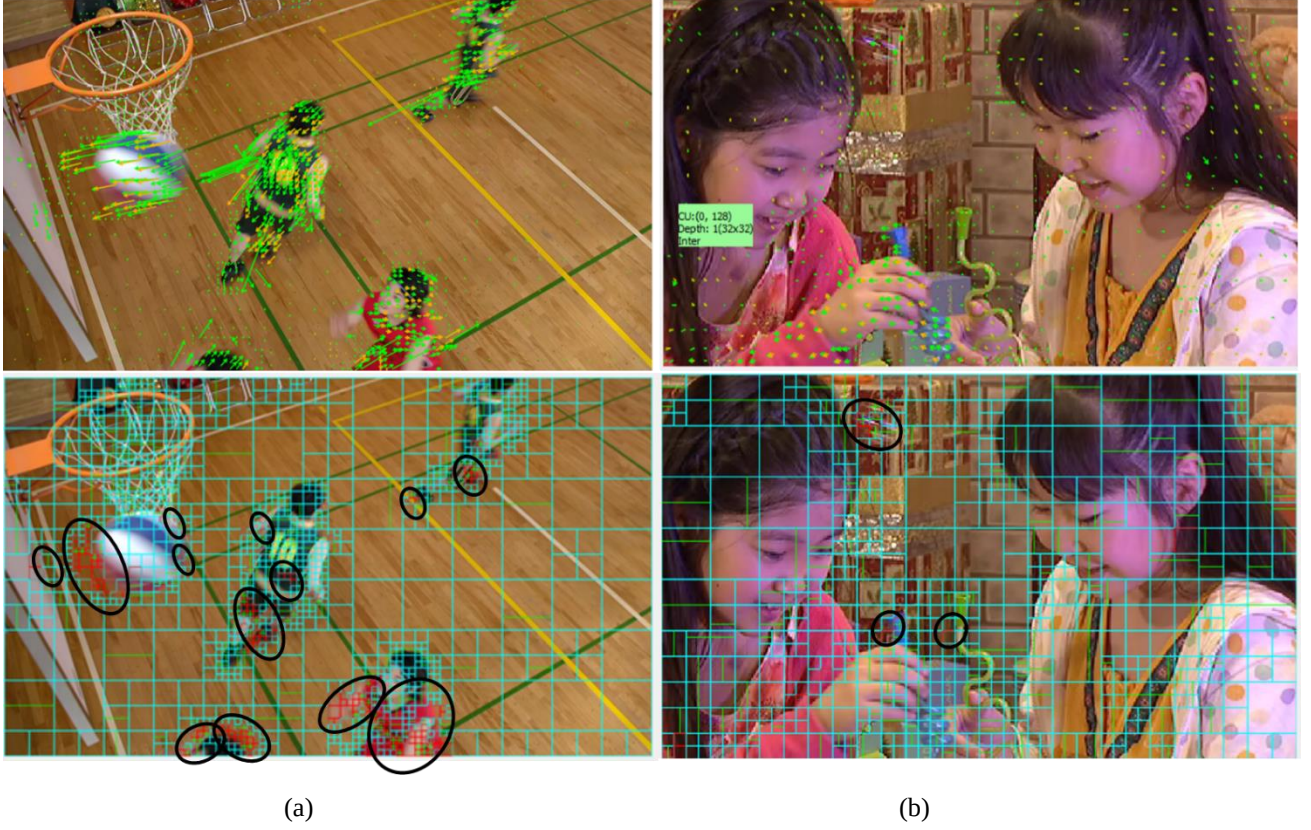


Figure7. Overlays showing motion vectors (upper image) and partitioning mode decisions (lower image) for the BasketballDrill (a) and the BlowingBubbles (b) sequences

Based on this observation, which holds true for all of the video sequences we have examined, we define the temporal component of an HEVC video sequence in terms of the ratio of Intra encoded PU's to that of Inter encoded PU's in the Inter predicted pictures (B, P, GPB) of the video sequence. For each picture k , in a video sequence the temporal index $\theta(K)$ is shown in Equation 5.

$$\theta(k) = \frac{Count(k)^{PU=Intra}}{Count(k)^{PU=Inter}} \quad (5)$$

The temporal complexity T of an Inter encoded video sequence consisting of N Inter encoded pictures is expressed in Equation 6. When the number of Intra encoded PU's is high the temporal complexity is assessed to be high.

$$T = \frac{\sum_{n=1}^N \theta(n)}{N} \quad (6)$$

We express the spatio-temporal complexity of a video stream C , as derived from the semantic information in and HEVC/H.265 bitstream as

$$C = \omega S + \tau T \quad (7)$$

The metric coefficients ω and τ were calculated as 0.48 and 1.76 respectively, indicating that the temporal component has more significance in determining content type. Coefficients in the temporal case were calculated using the tmoral complexity metric defined in ITU-T REC 910²⁷.

3.4 Implementation

The implementation and evaluation were performed using a PC with an Intel Core i7 processor, 8GB RAM and a solid state hard drive running the Ubuntu 12.04 operating system. Video sequences were encoded using the HEVC reference software (HM 11.0)²⁸. Bitsream semantic information was extracted using a version of the HM 11.0 reference decoder that had been modified to only parse the bitstream and allow access to the CTU, CU and PU data without performing any other decoding tasks. Any further implementation Figure 8 shows an example of the high level semantic data parsed from the bitstream for the CU previously shown in Figure 6. The example illustrates one branch of the CTU whose x and y coordinates are 320 and 256 respectively (top left corner) and a size of 64x64 luma samples (8.1). The split flag at dpeth level 0 of the CTU indicates that the CTU was split at that level with no encoding at 64x64 ($a(i,j,k) = 0$). The four subdivision at depth level 1 is detailed showing that this CTU is split (8.2) at level 1. The fourth subdivision of the level one split is further examined at depth level 2 where the split_cu_flag indicates (8.3) that this CTU is not split any further and is encoded at depth level 2 (16x16). As this is the final splitting decision, the resultant coding unit is examined (8.4) to determine the prediction mode which is Inter (pred_mode_flag = 0).

```
CTB Syntax
└─ coding_tree_unit()
  └─ sao(5, 4)
    └─ coding_quadtree(320, 256, 64, 0)
      └─ 1 <- split_cu_flag
      └─ coding_quadtree(320, 256, 32, 1)
      └─ coding_quadtree(352, 256, 32, 1)
      └─ coding_quadtree(320, 288, 32, 1)
      └─ coding_quadtree(352, 288, 32, 1)
        └─ 1 <- split_cu_flag
        └─ coding_quadtree(352, 288, 16, 2)
        └─ coding_quadtree(368, 288, 16, 2)
        └─ coding_quadtree(352, 304, 16, 2)
        └─ coding_quadtree(368, 304, 16, 2)
          └─ 0 <- split_cu_flag
          └─ coding_unit(368, 304, 16)
            └─ 0 <- skip_flag
            └─ 0 <- pred_mode_flag
            └─ prediction_unit(x0, y0, nPbW, nPbH)
              └─ 0 <- merge_flag
              └─ 1 <- inter_pred_idc
              └─ 3 <- ref_idx_l0
              └─ mvd_coding(x0, y0, 0)
              └─ 1 <- mvp_l0_flag
              └─ 1 <- no_residual_syntax_flag
              └─ transform_tree(368, 304, 368, 304, 16, 0, 0)
```

Figure 8. An example of high level semantic data available in the HEVC bitstream for a single CU.

It can be easily seen from Figure 8 that all of the semantic data needed to implement our scheme is readily available, with low complexity of extraction, in the HEVC/H.265 bitstream. Statistics were gathered using new C++ objects added to the modified decoder. In this implementation we made the assumption that no scene cuts occurred during the short duration of the test sequences. Any realistic implementation would require to select the number of pictures over which to

calculate the spatial and temporal characteristics in a way which takes account of the different scenes in a video sequence. Content type may then be dynamically calculated at intervals in a bitstream. Although exploration of a method to identify scene cuts in an HEVC encoded bitstream is outwith the context of this paper, we envisage that detection would be possible using a variation of the spatio-temporal characterisation methods presented here.

4. EVALUATION & RESULTS

4.1 Evaluation of Proposed Scheme

Our pilot evaluation conducted for this paper employed five video sequences which were each encoded at QP's of 22, 27, 32 and 37. The five sequences are shown in Table 2.

Table 2. Video sequences used for evaluation of the proposal.

	<i>Spatial Resolution</i>	<i>Frame Rate (fps)</i>
Kimono1	1920x1080	24
ParkScene	1920x1080	24
BasketballDrill	832x480	50
Vidyo1	1280x720	60
BQTerrace	1920x1080	60

4.2 Analysis of Results

The content type values (C) derived using our metric are shown in Figure 9 for each of the video sequences. The calculated content value varies inversely to the Quantisation Parameter (QP) used by the encoder. Although the values of (C) diminished as the quantisation parameter (QP) increased, the overall trends and differential between sequences with diverse spatio-temporal characteristics remained consistent.

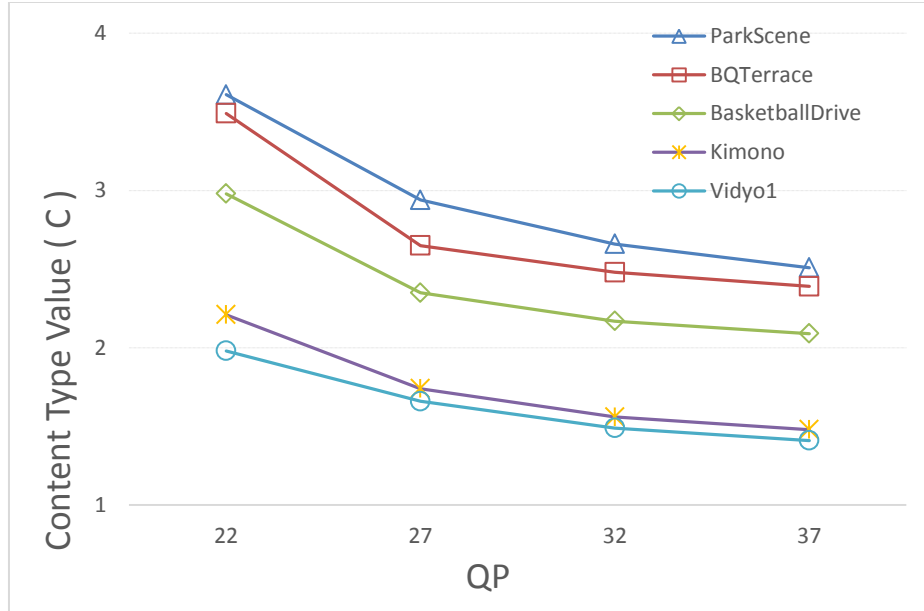


Figure 9. Content types derived using the proposed scheme for each of the five test sequences at a range of QP's.

We attribute this relationship between QP and content type value to the different CTU splitting and prediction mode decisions made by the encoder when different QP values were used. The proposed scheme was able to clearly differentiate between the different content types of the five test sequences which were chosen to reflect different temporal and spatial complexities. Further work on this metric may consider a means of grouping video sequences as

discrete classes based on their spatial and temporal complexity. This metric could be used as part of a QoE aware provisioning scheme for video streaming applications. Either by providing a no-reference method of classifying content type at the client node for use in end-to-end application layer mechanisms or at intermediate nodes as part of a service provider QoE provisioning scheme.

The complexity of our scheme was compared to that of simply decoding an HEVC bitstream without attempting to perform any extraction of features that would be required for content type derivation. In Figure 10 the results of this comparison are shown for each of the five sequences. Two QP testing points (22, 32) encoded with the Random Access configuration are shown as examples. Although there was significant difference in the decoding times between sequences and QP's, the time taken by the proposed scheme was always less than 20% of the decoding time alone and was always shorter than the duration of the video sequence. Our scheme did take longer for some sequences than others, however this was primarily due to the increased parsing time for the larger volumes of data carried in higher resolution sequences such as ParkScene. The proof of concept implementation of our proposal was based on an amended version of the relatively inefficient HEVC/H.265 reference decoder. A custom bitstream analysis tool using code optimization techniques would make this, already fast, technique suitable for real time content type analysis, even on modestly resourced client devices.

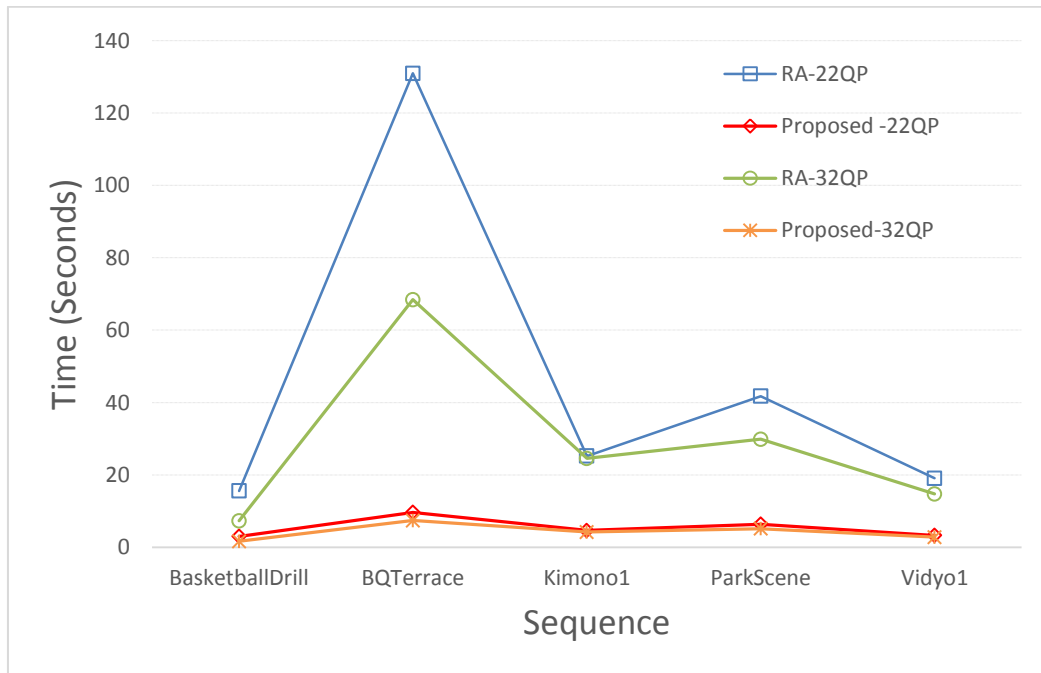


Figure 10. A comparison of complexity between the proposed scheme and decoding to permit extraction of spatio-temporal features. Each sequence is either decoded at one of two QP's or has content type derived using the proposed scheme.

5. CONCLUSIONS

In this paper we present a novel, heuristic method of determining the content type of a video sequence from semantic data contained within an encoded HEVC/H.265 bitstream. Our scheme leverages the relationship between an HEVC/H.265 encoder's partitioning and prediction mode decision making processes and the spatio-temporal features within the video stream. The spatial complexity of a video sequence is directly related to the CU partitioning of each picture in the stream as areas with high levels of detail or rich texture are invariably encoded with smaller CU sizes than large homogenous areas. In our proposal spatial complexity is calculated from the weighted averages of the number of Coding Units (CU) that have been encoded at each depth of the Coding Tree Unit (CTU) quadtree structure.

To determine the temporal complexity of a video sequence we leverage the observation that HEVC/H.265 encoders choose Intra prediction mode when encoding the edges of objects moving relative to the background. Our proposal employs the ratio of Intra encoded CU's to Inter encoded CU's in each picture to quickly estimate temporal complexity. The temporal method, in addition may also be used to determine focal points and regions of interest in an HEVC/H.265 encoded bitstream.

Our method is shown to take less than 20% of the time required to decode and extract spatio-temporal features from the decompressed domain and could be employed as the video content component of parametric Quality of Experience aware models for real time video streaming applications.

REFERENCES

- [1] Cisco, Visual Networking Index Forecast 2012 – 2017, 29 May 2013, < http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360_ns827_Networking_Solutions_White_Paper.html>, 18 December 2013.
- [2] NGNM, "Service delivery and quality measurement", Technical report, available online at http://www.ngmn.org/uploads/media/NGMN-P-SERQU_Service_Quality_Definition_and_Measurement_-_A_Technical_Report_by_NGMN_Alliance_v1_0_4.pdf.
- [3] Fiedler, M.; Hossfeld, T.; Tran-Gia, P., "A generic quantitative relationship between quality of experience and quality of service," *Network, IEEE*, 24(2), 36-41, (2010).
- [4] Kilkki K., "Quality of Experience in Communications Ecosystem," *Journal of Universal Computer Science*, 14(5), 615-624, (2008)
- [5] ITU-T: "Definition of Quality of Experience (QoE)", International Telecommunication Union, Liaison Statement, Ref.: TD 109rev2 (PLEN/12), (2007).
- [6] Kanumuri S. et al, "Predicting H.264 packet-loss visibility using a generalized linear model," *IEEE International Conference on Image Processing*, 2245-2248, (2006)
- [7] Chang Y-L, Lin T-L and Cosman P.C., "Network based H.264/AVC whole-frame loss visibility model and frame dropping methods" *IEEE Transactions on Image Processing*, 19(3), 722-735, (2010).
- [8] Ascenso J, Cruz H and Dias P, "Packet-header based no-reference quality models for H.264/AVC video transmission," *International conference on telecommunications and multimedia*, 174-151, (2012).
- [9] Seshadrinathan K, Bovik C, "Motion Tuned Spatio-temporal Quality Assessment of Natural Videos," *IEEE Transactions on Image Processing*, 19(2), 335-350, (2010).
- [10] Hekstra A et al, "PVQM-A perceptual Visual quality measure," *Signal Processing Image Communication*, 17 (10), 781-798, (2002).
- [11] Chan A, Pande A, Baik E and Mohapatra P, "Temporal Quality Assessment for Mobile Videos," *Proceedings of the 18th annual international conference on Mobile computing and networking*, 221-232, (2011).
- [12] Yen-Fu Ou; Zhan Ma; Tao Liu; Yao Wang, "Perceptual Quality Assessment of Video Considering Both Frame Rate and Quantization Artifacts," *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(3), 286-298, (2011).
- [13] Wiegand T., Sullivan G., Bjontegaard G., and Luthra A., "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.* 13(7), 560-576 (2003).
- [14] ITU, "High efficiency video coding", ITU-T H.265, (2013).
- [15] Sullivan G. J. et al, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.*, 22(12), 1649-1668 (2012).
- [16] Wang Z et al, "Image Quality Assessment: From Error Visibility to Structural Similarity" *IEEE Transactions on Image Processing* 13(4), 600-612, (2004).
- [17] Wang Z, Lu L. and Bovik A., "Video quality assessment based on structural distortion metric", *Signal processing Image Communication*, 19(2), 121-132, (2004).
- [18] Wang Z., Simoncelli E. and Bovik A., "Multiscale structural similarity for image quality assessment", *Signals, Systems and Computing*, 2, 1398-1402, (2003).
- [19] ITU-T, Rec J.246, Perceptual visual quality measurement techniques for multimedia services over digital cable television networks in the presence of a reduced bandwidth reference (2007).

- [20] Khan, A.; Lingfen Sun; Ifeachor, E, "Learning Models for Video Quality Prediction over WLAN and UMTS Networks", IET Communications, 4(12), 1380-1403, (2010).
- [21] Khan, A.; Lingfen Sun; Ifeachor, E., "Content Clustering Based Video Quality Prediction Model for MPEG4 Video Streaming over Wireless Networks," Communications, 2009. ICC '09. IEEE International Conference on , 1-5, (2009).
- [22] Xiangyu Lin; Ma, Hanjie; Lei Luo; Yaowu Chen, "No-reference video quality assessment in the compressed domain," Consumer Electronics, IEEE Transactions on , 58(2), 505-512, (2012).
- [23] Raake, A.; Garcia, M. -N; Moller, S.; Berger, J.; Kling, F.; List, P.; Johann, J.; Heidemann, C., "T-V-model: Parameter-based prediction of IPTV quality," Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on , 1149-1152, (2008).
- [24] Ries, M.; Crespi, C.; Nemethova, O.; Rupp, M., "Content Based Video Quality Estimation for H.264/AVC Video Streaming," Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE , 2668-2673, (2007).
- [25] Joskowicz, J.; Sotelo, R.; Lopez Ardao, J.C., "Towards a General Parametric Model for Perceptual Video Quality Estimation," Broadcasting, IEEE Transactions on , 59(4), 569-579, (2013).
- [26] Khan, A.; Lingfen Sun; Ifeachor, E., "QoE Prediction Model and its Application in Video Quality Adaptation Over UMTS Networks," Multimedia, IEEE Transactions on , 14(2), 431-442, (2012).
- [27] ITU-T, Rec P910, "Subjective video quality assessment methods for multimedia applications".
- [28] HEVC reference software version HM 11.0, <http://hevc.kw.bbc.co.uk/git/w/jctvc-hm.git/>.