HEVC Coding Optimisation for Ultra High Definition Television Services

Matteo Naccari, Andrea Gabriellini, Marta Mrak
British Broadcasting Corporation - R&D, United Kingdom

Saverio Giovanni Blasi, Ivan Zupancic, Ebroul Izquierdo
Queen Mary University of London, United Kingdom

***Abstract*—Ultra High Definition TV (UHDTV) services are being trialled while UHD streaming services have already seen commercial débuts. The amount of data associated with these new services is very high thus extremely efficient video compression tools are required for delivery to the end user. The recently published High Efficiency Video Coding (HEVC) standard promises a new level of compression efficiency, up to 50% better than its predecessor, Advanced Video Coding (AVC). The greater efficiency in HEVC is obtained at much greater computational cost compared to AVC. A practical encoder must optimise the choice of coding tools and devise strategies to reduce the complexity without affecting the compression efficiency. This paper describes the results of a study aimed at optimising HEVC encoding for UHDTV content. The study first reviews the available HEVC coding tools to identify the best configuration before developing three new algorithms to further reduce the computational cost. The proposed optimisations can provide an additional 11.5% encoder speed-up for an average 3.1% bitrate increase on top of the best encoder configuration.**

I. INTRODUCTION

With the advent of higher resolution cameras and displays, users will be provided with a more immersive and compelling experience when watching television programmes. The format which will deliver this improved quality of experience is called Ultra High Definition Television (UHDTV). The parameters for UHDTV are specified in the ITU Recommendation BT.2020 [1]. This recommendation standardises two spatial resolutions: 3840×2160 pixels/frame and 7680×4320 pixels/frame, both of which are integer multiples of the 1920×1080 (HDTV) picture size. Temporal resolutions for UHDTV are up to 120 frames per second (fps) and only progressive scanning is allowed. Pixel and picture aspect ratios are assumed to be square and 16:9, respectively. Allowed bit depths are 10- and 12-bit while the colourimetry system is wider than the one specified in Recommendation ITU-R BT.709 for HDTV signals and covers 75.8% of the CIE 1931 colour space. The chrominance sampling ratios included in BT.2020 range from 4:2:0 to 4:4:4. While BT.2020 defines the parameters of UHDTV services from the signal perspective, other organisations such as Digital Video Broadcasting (DVB) and European Broadcasting Union (EBU) have been working towards the definition of the parameters needed by applications which make use of UHDTV signals. The DVB has recently ratified the parameters for the delivery of UHDTV services using HEVC (they will be published as version 12 of ETSI TS 101 154): spatial resolution of 3840×2160, maximum bit depth of 10-bit, temporal resolution up to 60 fps, BT.709 colourimetry.

The volume of data associated with UHDTV signals is eight times the one associated with HDTV and this calls for efficient video compression technology. In answer to this need, ITU Video Coding Experts Group (VCEG) and ISO Moving Picture Experts Group (MPEG) joined efforts in a partnership called Joint Collaborative Team on Video Coding (JCT-VC) to develop the High Efficiency Video Coding (HEVC) standard [2]. Version 1 of HEVC was finalised in January 2013 and proved to outperform its predecessor Advanced Video Coding (AVC) by providing up to 50% bitrate reduction for the same perceived quality [3]. For UHD content, the reduction is even higher than 50% [4]. Given the superior performance, HEVC will be the best candidate in the deployment of UHDTV services. As expected, this improved compression efficiency comes at the expense of increased complexity, especially on the encoder side. Considering the reference implementations for AVC (JM) and HEVC (HM), it is estimated that an HEVC encoder can be up to 4 times more complex than an AVC one [5]. Therefore any practical implementation must optimise HEVC coding by reducing the complexity without sacrificing the compression performance.

In this context, this paper presents an analysis of the performance of HEVC for UHDTV content, assessing each coding tool for its impact on compression efficiency and encoder complexity. The main outcome of this analysis is a baseline encoder configuration suitable for UHDTV content compression. The paper then proposes three novel encoder optimisation techniques to further reduce the encoder complexity. The three novel optimisations are Multiple Early Termination (MET) for motion estimation, Adaptive Reference Frame Selection (ARFS) and Adaptive Partition Selection (APS) for motion compensation. These optimisation methods are integrated in the HM codec and their performance is assessed in terms of compression efficiency penalty and encoder complexity reduction.

The remainder of this paper is organised as follows: Section II presents the analysis of each coding tool for HEVC. The proposed encoder optimisations are described in Section III while their performance is analysed and discussed in Section IV. Finally, conclusions and future research directions are given in Section V.

II. ANALYSIS OF HEVC CODING TOOLS OVER UHD CONTENT

This section describes the experiments performed over UHDTV content to analyse the performance of each coding tool standardised by HEVC. The test material, coding conditions and performance indicators are presented first. Then the focus moves to describe the experiments and discuss their results.

A. Test Material and Coding Conditions

The test set is composed of sixteen sequences with 8 bits per component, 4:2:0 chroma format, 3840×2160 spatial resolution and frame rate of 50 and 60 fps. The name of these sequences, along with the type of content portrayed are listed in Table I. Each sequence is coded with four Quantisation Parameter (QP) values. They have been determined by visually inspecting the test set compressed with QP in the range 22 : 45 to determine a good coverage of different visual quality levels: from very good (i.e. coding artefacts unnoticeable) to fairly poor (i.e. coding artefacts visible and annoying). Content denoted as *outdoor* portrays external scenes. Some of these sequences contain water and complex motion (e.g. "*PetitBato*", "*Sedof*" and "*Manege*") or sharp details and camera panning (e.g. "*ParkAndBuildings*") and large area picturing grass (e.g. "*ParkAndBuildings*" and "*ParkDancers*"). Content denoted as *drama* corresponds to indoor

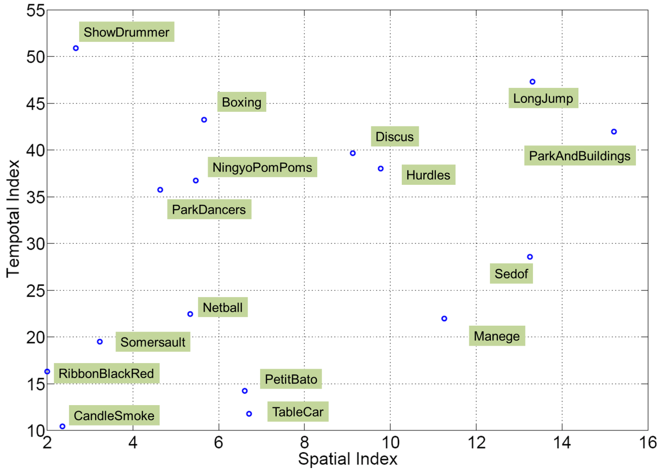


Fig. 1. Spatial and temporal indexes computed over the selected test material.

sequences representative of television drama. Content denoted as *objects* represents indoor scene with moving objects; this content is not fully representative of UHDTV material but, given its spatial and temporal features, is challenging from the compression point of view. Finally, content denoted as *sport*, was shot during the 2014 edition of the Commonwealth Games in Glasgow.

TABLE I

TEST MATERIAL DESCRIPTION.

Sequence name	Frame rate	Type	Sequence name	Frame rate	Type
ParkAndBuildings	50	outdoor	TableCar	50	objects
NingyoPompoms	50	objects	RibbonBlackRed	60	sport
ShowDrummer1	60	drama	Hurdles	50	sport
Sedof	60	outdoor	LongJump	50	sport
Petitbato	60	outdoor	Discus	50	sport
Manege	60	outdoor	Somersault	50	sport
ParkDancers	50	outdoor	Boxing	50	sport
CandleSmoke	50	drama	Netball	50	sport

The characteristics of the test material can be also described with the Spatial and Temporal Indexes (SI-TI) specified by the ITU Recommendation P.910 [6]. The scatter plot is depicted in Figure 1. As may be noted, the content spans a wide SI-TI feature space and thus represents a good selection of material to thoroughly assess HEVC performance and test new encoder optimisations. Finally, all the sequences have been encoded according to the JCT-VC Common Test Conditions (CTCs, [7]) using the selected QP values and the Random Access Main (RA-Main) configuration as this is representative of the encoding settings used in broadcasting services. For all experiments, the HEVC test Model (HM) software version 12.0 [8] has been used.

B. Performance Indicators

Compression efficiency and encoder complexity are used as performance metrics throughout this paper. For compression efficiency, the metric used is the Bjøntegaard Delta-rate (BD-rate) computed according to [9] between the anchor data (i.e. the sequences compressed with JCT-VC CTCs) and the sequences compressed according to the described experiments. In this context, negative BD-rate values will correspond to compression efficiency gains. Given the use of 4:2:0 chroma format, only the BD-rate for the luminance component will be considered. The running time of the encoder is used as complexity metric and the following sequence-based measures are reported.

- *Encoder Speed-up* (ES, in percentages), given as the arithmetic mean of four single speed-ups associated with the tested QP values. Positive values denote encoder complexity reductions while negative ones are associated with encoder slow-downs.
- *Delta Time difference* (ΔT in percentages), given as difference between the maximum and minimum speed-up. This measure indicates the encoding time variation across the coding points. The lower this value, the more effective an algorithm/coding tool is for a wide range of bitrates.

Finally, the percentage of the encoding time spent in each coding module has been estimated using the Google *gperf* tool [10].

C. Experiments Description

The HEVC standard specifies several novel tools for each coding module of a general motion compensated predictive codec architecture. Assuming a computationally constrained environment, it is important to investigate the trade-off between compression efficiency and encoder complexity associated with each tool. Therefore a set of experiments has been performed to quantify these trade-offs. The tools considered are listed in Table II. For each tool, the aforementioned performance indicators are computed between the anchors and the HM encoder when the tool is parametrised accordingly[1]. Besides testing HEVC coding tools, some non normative tools implemented in HM are assessed; these are Rate Distortion Optimised Quantisation (RDOQ), Early Coding Unit (ECU) termination, Early Skip Detection (ESD) and Coding Flag Mode (CFM) [11]. The results obtained will be then discussed in the following subsection.

TABLE II

HEVC CODING TOOLS CONSIDERED FOR TESTING.

Experiment name	Tested	CTC
Coding Tree Unit (CTU) sizes	32×32, 16×16	64×64
Residual QuadTree (RQT) depths	2 and 1	3
Max Transform Unit (TU) size	16×16, 8×8	32×32
Transform Skip (TS)	Disabled	Enabled
Sign Data Hiding (SDH)	Disabled	Enabled
RDOQ	Disabled	Enabled
Intra coding	Full rate distortion optimisation	HM implementation [11]
Quarter pel motion vector	Off	On
Half pel motion vector	Off	On
Asymmetric Motion Partition (AMP)	Disabled	Enabled
Merge mode candidates	3 and 2	5
Number of reference frames	1	Up to 4
ECU, ESD and CFM	Enabled	Disabled

D. Results Analysis

Table III lists the average results across the test set for the experiments performed. Starting with CTU size, it may be noted that for UHDTV material avoiding the use of large block sizes (e.g. 64×64) leads to high BD-rate penalties. Moreover, when HEVC adopts a block partitioning like AVC (i.e. the maximum CTU size is 16×16) the BD-penalty reaches its maximum values. These results are also in line with some published studies [3], [12]. Looking at RQT depth, it may be noted that limiting maximum depth to 1 incurs an

---

[1] Given the predictive nature of the codec specified by HEVC, disabling an individual coding tool may affect the performance and complexity of others.

TABLE III
HEVC CODING TOOLS: BD-RATE PENALTY AND ENCODER COMPLEXITY.

Experiment name	BD-rate [%]	ES [%]	ΔT [%]
CTU size 32×32	7.0	15.0	0.9
CTU size 16×16	34.0	35.0	1.6
RQT depth 2	0.3	9.0	2.4
RQT depth 1	1.0	15.0	4.3
TU size 16×16	4.3	10.0	2.7
TU size 8×8	13.0	11.0	3.2
TS off	-0.1	4.0	1.6
SDH off	0.8	0.6	1.0
RDOQ off	5.9	6.0	3.4
Intra coding	-1.3	-480.0	72.0
Quarter pel off	1.8	21.0	5.8
Half pel off	8.5	39.0	9.1
AMP off	0.5	5.0	6.0
Merge candidates 3	0.3	3.0	1.7
Merge candidates 2	0.9	6.0	2.6
One reference frame	2.5	31.0	4.0
ECU, ESD and CFM	1.4	68.0	26.2

average BD-rate loss of about 1% for a speed-up of 15%. These results seem to show that for UHDTV content both the spatial and temporal redundancy are mainly exploited by CTU partitioning while a deep recursion on transform units is less beneficial. This claim also holds for the results associated with the reduction of TU size to 16×16 and 8×8. In fact, if a large block (e.g. 32×32) is used for prediction, then a large transform may better capture all the remaining redundancy. Three tools operates at the transform stage: TS, SDH and RDOQ. TS does not bring any compression gain and its evaluation costs on average 4% of the encoder time. Transform skip proves to be efficient on sequences containing synthetic objects and text, both elements missing from the used test set. If instead these elements were present, fast algorithms should be devised to apply TS directly on the affected image areas. The remaining two tools provide coding gains for limited complexity (negligible for SDH).

Another very interesting result is provided by the "*Intra coding*" experiment. Using full Rate Distortion Optimisation (RDO) for all prediction modes leads to an average 1.3% BD-rate gain for a prohibitive 480% slow-down. This result proves that HM is already highly optimised for intra coding, especially when compressing UHDTV content.

With regards to inter coding, the results suggest that subpel motion compensation is needed (particularly at half pel resolution). The asymmetric partitions used for motion compensation do not prove to be very effective with a modest BD-rate penalty and a significant 5% complexity reduction when disabled. When the number of merge candidates is reduced limited losses are registered on average for significant complexity reduction. However, the high variability of the losses observed across the sequences suggest that rather than limiting the number of candidates an adaptive method should be devised. The same variability is also observed when the number of reference frames is limited to 1.

Finally, the performance of the built-in fast mode selection methods (ECU, ESD and CFM) are evaluated. A significant complexity reduction can be achieved for a limited 1.4% BD-rate penalty. It should be noted, however, that these methods lead to a rather high ΔT value. This is expected since ECU and ESD rely on the selection of the SKIP coding mode which is often selected at low bitrates. Therefore, ECU and ESD are effective mainly there while providing much smaller complexity reduction at higher bitrates. The same argument can be used for CFM since it checks whether all prediction residuals for one block have been quantised to zero, which is likely to happen at lower bitrates.

From the results obtained, only the tools which provided the best trade-off between coding efficiency and encoder complexity with low variation across the test set have been retained. The coding tool parameter values for the selected baseline are listed in Table IV. When the HM codec is run with this configuration an average BD-rate penalty of 3.0% is measured for a 76% encoder speed-up. Finally, it is also interesting to look at the profiling results for the baseline encoder. Hence, Table V reports the percentage of time spent in each coding stage estimated with the *gperf* profiler. The larger share of the encoder time is taken by inter coding which comprises different submodules among which motion estimation, merge mode and subpel motion compensation register the largest percentage. Given this result, the encoder optimisations proposed in this paper will focus on these coding submodules.

TABLE IV
HEVC CODEC CONFIGURATION FOR UHDTV CONTENT COMPRESSION.

Coding tool	Value	Coding tool	Value
CTU size	64×64	TS	Disabled
AMP	Disabled	RDOQ, SDH	Enabled
RQT depth	1	Merge mode	5 candidates
Reference frames	4	ECU, ESD, CFM	Enabled

TABLE V
TIME SPENT IN EACH CODING STAGE.

Coding stage	Time spent [%]
Motion estimation	31
RDOQ	7
Subpel motion compensation	14
Merge mode	10
Intra coding	9
Inter coding	76

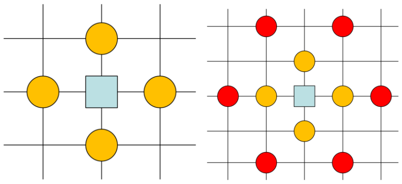
III. PROPOSED ENCODER OPTIMISATIONS

The proposed encoder optimisations will address inter coding, particularly motion estimation, RDO search for different motion partitions and reference frame selection for motion compensation. In the following the proposed methods are presented starting from the one addressing complexity reduction for motion estimation.

A. *Multiple Early Termination*

The Multiple Early Termination (MET, [13]) algorithm stops integer pel motion estimation search for one Prediction Unit (PU) if the starting point candidate has the best rate distortion cost. Otherwise, motion estimation proceeds as implemented in the HM codec ([14], [11]). The rationale for MET is that the prediction energy distribution can be approximated as nearly convex in the neighbourhood of one local minimum. If this minimum corresponds to one of the starting points, then the search can be terminated early. Considering a prediction unit at depth d of the CTU partitioning, the MET repeats the following sequence of steps for each starting point candidate i tested by Enhanced Predictive Zonal Search (EPZS):

1) Consider all pixel locations of the diamond search pattern, Figure 2(a), centred on i, and compute the rate distortion cost for all of them.
2) If the minimum cost corresponds to i, go to the next step. Else, move to the next starting point candidate $i + 1$.



(a) Diamond (b) Hexagonal

Fig. 2. Search patterns for motion estimation.

3) If the depth of the current PU is $d < 2$, then terminate early and move to the next PU. Otherwise, consider all pixel locations of a hexagonal search pattern, Figure 2(b), centred on i.
4) Compute the rate distortion cost for each pixel location.
5) If the minimum cost corresponds to i then terminate early and move to the next PU.
6) Else, move to the next starting point candidate $i+1$. Otherwise, if all the starting points have been examined, perform motion estimation as implemented in HM.

It should be noted that if the motion estimation cannot terminate early, the overall processing is increased by the additional cost computations required by Steps 1 and 4 above. Therefore, in order to minimise the impact of this additional processing the distortion for each tested pixel location is computed on an 8×8 fixed block resolution. That is, if the prediction unit size is larger than 8×8, then the block is resized accordingly. This block resizing is also performed when EPZS is not terminated early.

B. *Adaptive Reference Frame Selection*

JCT-VC CTCs require the use of multiple reference frames, up to 4 for the RA-Main configuration. Adaptive Reference Frame Selection (ARFS) attempts to reduce the number of reference frames without affecting the compression performance. ARFS collects statistics on the usage of reference frames for two consecutive Structure of Pictures (SOP). In particular, for each reference list and for each layer in the SOP, it registers the total number of prediction units and the number of units using the first reference frame. If the additional reference frames are not used often enough they are dropped from evaluation for future frames of the same type. Therefore, let $l \in \{0, 1\}$ denote the prediction list used for motion compensation. Let also N_F^l denote the number of times the first reference frame from l is used for inter prediction and let N_T^l denote the total number of prediction units predicting from l in the current frame. ARFS disables the use of additional reference frames only if the following condition is satisfied:

$$N_F^l > \theta \cdot N_T^l. \qquad (1)$$

where threshold θ is an experimentally determined value set to 0.75 for UHD content. The statistics collected by ARFS tend to become less correlated with the characteristics of the video content as they change over time and they are typically invalidated by a scene change. Therefore a scene change detection or a 10 second interval, whichever comes first, are used to refresh the statistics. ARFS does not disable any reference frame while collecting statistics.

C. *Adaptive Partition Selection*

During inter coding, HM performs RDO to decide the best partition size for each coding block (i.e. $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$). Table VI reports the frequency of different partition sizes for some test sequences. From these results three main facts may be observed: firstly $2N \times 2N$ is the most selected partition size. Secondly, for some sequences with complex motion and high spatial activity (e.g. "*Manege*" and "*Sedof*") partitions such as $2N \times N$ and $N \times 2N$ account for roughly 50% of the total while for sequences with regular motion (e.g. "*ParkDancers*") they account for a negligible percentage. Thirdly, the percentages reported in Table VI are independent from the coding rate and therefore a unified fast partition selection can be designed to work across all coding points. The proposed Adaptive Partition Selection (APS) algorithm checks whether the prediction residuals associated with $2N \times 2N$ are homogeneous. If this is the case, then RDO search for $2N \times N$ and $N \times 2N$ is omitted. More precisely, the $2N \times 2N$ prediction residuals are split in rectangles A and B as in Figure 3. The Sum of Absolute Difference (SAD) is computed for A and B and RDO search for their corresponding partitions is skipped if:

$$1 - \delta \leq \frac{SAD(A)}{SAD(B)} \leq 1 + \delta, \qquad (2)$$

where the value for δ is set to 25%. To further decrease the number of RDO searches, a threshold τ is used to filter the SAD values. Therefore, if for one block the normalised values of $SAD(A)$ and $SAD(B)$ are both less than τ then the RDO search for the associated partition is avoided. In this paper τ is set to 4 since it provided the best trade-off between BD-rate penalty and encoder speed-up.

TABLE VI
USE FOR DIFFERENT PARTITIONS (IN %) FOR SOME TEST SEQUENCES.

	Manege		**Sedof**		**ParkDancers**	
Size / QP value	**27**	**37**	**27**	**38**	**23**	**38**
$2N \times 2N$	48	48	55	57	78	75
$2N \times N$	11	9	16	12	8	9
$N \times 2N$	41	43	29	31	14	16
$N \times N$	0	0	0	0	0	0

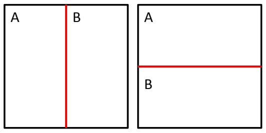


Fig. 3. Split considered by APS for residuals associated to $2N \times 2N$ partition.

IV. EXPERIMENTAL RESULTS

This section presents the performance of the proposed optimisations when implemented in the HM codec. The test material, coding conditions and performance indicators are the ones described in Section II while the encoder parameters are listed in Table IV.

Table VII reports the BD-rate penalty and encoder speed-up where the anchor is again the HM codec run according the JCT-VC CTCs. The proposed optimisations have been tested using the same order of Section III. Accordingly, the MET algorithm can provide 4.5% speed-up for an average loss of 1.1%. When the ARFS algorithm is implemented on top of MET, it yields an additional 2.9% speed-up for about 0.4% average BD-rate penalty. Finally, APS tested on top of MET plus ARFS brings a further 4.5% speed-up for 1.6% average BD-rate loss. It may be noted from the data in Table VII that also the value for ΔT has reduced which proves that all proposed methods are quite effective for a wide range of coding points.

In order to assess how the proposed optimisations compare with respect to other optimised HEVC implementations, the x265 codec [15] is considered. This codec is run using the random access very slow preset in single thread mode. To completely assess the performance of x265 and HM equipped with the proposed optimisations (hereafter denoted as HM-optimised), the BD-rate for luma and the encoder speed-up are reported for all sequences in Table VIII. As may be noted the HM-optimised codec outperforms x265 by providing 2.9% more encoder speed-up with a lower BD-rate penalty (3.8% less). Furthermore, for the x265 codec some high BD-rate losses (up to 81%) have been observed over the chroma components. It should be also noted from Table VIII that in some cases x265 provides significant compression gains (e.g. for "*TableCar*"). An inspection of the bitstreams coded with x265 revealed that the gains were mainly coming from a group of picture structure different from the one used in RA-Main and with different length to accommodate the varying temporal features of each sequence.

Finally, it is also interesting to show the results when the *gperf* profiler is run over HM-optimised. Thus, Table IX reports the percentage of time spent by each coding module. Compared to the numbers shown in Table V, it may be noted that the percentage of time taken by inter coding has been reduced by 21%. These numbers can be then used to develop further encoder optimisations to address some modules which are not influenced by MET, ARFS and APS.

TABLE VII
PROPOSED OPTIMISATIONS: BD-RATE PENALTY AND ENCODER SPEED-UP.

Optimisation tool	**BD-rate [%]**	**ES [%]**	**ΔT [%]**
Baseline codec	3.0	76.6	15.0
MET	4.1	80.8	13.5
ARFS	4.5	83.6	13.0
APS	6.1	88.1	7.5

TABLE VIII
COMPARISONS BETWEEN HEVC OPTIMISED IMPLEMENTATIONS.

	HM-optimised		**x265**	
Sequence	**BD-rate [%]**	**ES [%]**	**BD-rate [%]**	**ES [%]**
ParkAndBuildings	4.8	89.7	1.0	85.9
NingyoPompoms	3.0	82.0	9.6	73.7
ShowDrummer1	7.1	87.8	14.0	83.4
Sedof	9.5	87.3	18.8	83.2
Petitbato	8.6	88.5	4.0	84.6
Manege	10.4	84.5	11.6	82.5
ParkDancers	0.8	89.8	-9.1	89.1
CandleSmoke	4.1	90.8	10.5	89.0
TableCar	2.9	91.5	-15.7	92.3
TapeBlackRed	4.6	90.2	10.6	88.7
Hurdles	6.2	89.2	45.6	87.6
LongJump	9.9	87.6	-2.0	84.2
Discus	7.1	84.7	5.1	80.1
Somersault	7.0	91.4	14.8	92.0
Boxing	4.8	86.6	9.1	83.8
Netball	6.3	88.5	14.3	83.9
Average	**6.1**	**88.1**	**8.9**	**85.2**

V. CONCLUSION AND FUTURE WORK

This paper presented a study of the optimisations needed to reduce the complexity of an HEVC encoder when compressing UHDTV content. An optimised encoder baseline was defined which only includes a reduced set of coding tools. Moreover, three algorithms have been proposed to provide additional speed-up. The proposed implementation obtains competitive speed-up for overall lower BD-rate penalty when compared to another fast implementation. Future work will address the coding stages which are now taking a considerable encoder time as listed in Table IX. Particular attention should be devoted to the test of intra coding in inter coded slices. For many sequences, intra coding is not eventually selected while it still takes on average 20% of the encoding time.

TABLE IX
TIME SPENT IN EACH CODING STAGE FOR THE HM-OPTIMISED CODEC.

Coding stage	**Time spent [%]**
Motion estimation (unidirectional)	8.5
RDOQ	11.7
Half per motion compensation	3.5
Quarter pel motion compensation	7.2
Motion estimation (bi-directional)	5.0
Merge mode	18.0
Intra coding	20.0
Inter coding	55.0

ACKNOWLEDGEMENT

The authors would like to thank Technology Strategy Board (TSB), UK, for co-funding this work as part of the THIRA consortium project (www.thiraproject.org.uk).

REFERENCES

[1] ITU-R, *Recommendation ITU-R BT 2020, Parameter values for ultra-high definition television systems for production and international programme exchange*, August 2012.

[2] ISO/IEC and ITU-T, *Information technology - High efficiency coding and media delivery in heterogeneous environments - Part 2: High efficiency video coding and H.265: High Efficiency Video Coding*, November 2013.

[3] J. R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards - including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, December 2012.

[4] R. Weerakkody, M. Mrak, T. K. Tan, V. Baroncini, G. J. Sullivan, and J.-R. Ohm, "Verification testing of HEVC compression performance for UHD video," in *IEEE GlobalSIP*, Atlanta, GA, USA, December 2014.

[5] K. Sühring F. Bossen, B. Bross, "HEVC complexity and implementation analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 22, pp. 1685–1696, December 2012.

[6] ITU-T, *Recommendation ITU-R P.910, Subjective video quality assessment methods for multimedia applications*, September 1999.

[7] F. Bossen, "Common HM test conditions and software reference configurations," Tech. Rep. JCTVC-L1100, JCT-VC, Geneva, CH, January 2013, 12th meeting.

[8] Joint Collaborative Team on Video Coding (JCT-VC), "HEVC reference software version HM12.0," downloadable at http://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-12.0.

[9] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," Tech. Rep. VCEG-M33, VCEG, Austin, TX, USA, April 2001.

[10] Google Inc., "Gperf tools," downloadable at http://code.google.com/p/gperftools/.

[11] K. McCann, C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. J. Sullivan, "High efficiency video coding (HEVC) test model 16 (HM 16) improved encoder description," Tech. Rep. JCTVC-S1002, JCT-VC, Strasbourg, FR, October 2014.

[12] R. Weerakkody and M. Mrak, "High efficiency video coding for ultra high definition television," in *Networked and Electonic Media (NEM) summit*, Nantes, FR, October 2013.

[13] I. Zupancic, S. G. Blasi, and E. Izquierdo, "Multiple early termination for fast HEVC coding of UHD Content," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, AU, April 2015.

[14] A. M. Tourapis, O. C. Au, and M. L. Liou, "Highly efficient predictive zonal algorithms for fast block-matching motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 934–947, October 2002.

[15] The x265 consortium, "x265 software version 1.3," downloadable at https://bitbucket.org/multicoreware/x265/wiki/Home.