Adaptive Precision Motion Estimation for HEVC Coding

Saverio G. Blasi, Ivan Zupancic and Ebroul Izquierdo
School of Electronic Engineering and Computer Science
Queen Mary University of London
United Kingdom
{s.blasi, i.zupancic, e.izquierdo}@qmul.ac.uk

Eduardo Peixoto
Departamento de Engenharia Eletrica
Universidade de Brasilia
Brazil
eduardopeixoto@ieee.org

***Abstract*—Most video coding standards, including the state-of-the-art High Efficiency Video Coding (HEVC), make use of sub-pixel Motion Estimation (ME) with Motion Vectors (MV) at fractional precisions to achieve high compression ratios. Unfortunately, sub-pixel ME comes at very high computational costs due to the interpolation step and additional motion searches. In this paper, a fast sub-pixel ME algorithm is proposed. The MV precision is adaptively selected on each block to skip the half or quarter precision steps when not needed. The algorithm bases the decision on local features, such as the behaviour of the residual error samples, and global features, such as the amount of edges in the pictures. Experimental results show that the method reduces total encoding time by up to 17.6% compared to conventional HEVC, at modest efficiency losses.**

***Index Terms*—HEVC, Video Coding, Sub-Pixel Motion Estimation**

I. INTRODUCTION

The state-of-the-art H.265/High Efficiency Video Coding (HEVC) standard [1] makes use of sub-pixel precision block-based Motion Estimation (ME) and motion compensation to achieve very high compression ratios. A frame is partitioned in square blocks of different sizes, referred to as Coding Units (CU). Each CU is then sub-partitioned for inter-prediction in up to four Prediction Units (PUs), and finally uni-directional and bi-directional ME are performed independently on each PU. Sub-pixel ME up to quarter precision can be considered in HEVC, similar to its predecessor H.264/Advanced Video Coding (AVC) [2]. In particular, reference frames are first interpolated to a given sub-pixel precision, and ME is performed on the interpolated frames to obtain a fractional MV refinement.

The interpolation and motion search steps needed during sub-pixel ME come at a very high computational price, which may limit the usage of HEVC especially when coding high resolution content. In this paper, a novel algorithm is presented to reduce complexity of HEVC sub-pixel ME. The scheme allows interpolation and motion search steps at selected precisions to be skipped in certain PUs. Both global and local features of the content are used at this purpose. The method can be applied on top of any integer precision fast ME algorithm. The rest of this paper is organised as follows. An overview of related work is presented in Section II. An analysis of the impact of sub-pixel ME on coding complexity and efficiency is presented in Section III. The proposed Adaptive Precision ME approach is detailed in Section IV. Experimental results are presented in Section V. Finally, Section VI concludes the paper.

II. RELATED WORK

Due to the fact that ME is responsible for a large portion of the coding time of typical encoders, many efforts have been spent to define fast ME methods and algorithms. The HEVC reference encoder implementation (HEVC Model, HM) [3] already includes a fast and efficientinteger precision ME algorithm based on Enhanced Predictive Zonal Search (EPZS) [4]. EPZS makes use of a combination of MV prediction algorithms and pattern searches to greatly reduce the number of tested MV candidates. EPZS is typically used in HM for uni-directional ME, whereas bi-directional MVs are found using a full motion search in a small window surrounding the uni-directional solution.

Instead of performing a time-consuming full motion search at sub-pixel precision, typical encoder implementations also rely on fast sub-pixel ME algorithms to reduce complexity. HM makes use of a Hierarchical Search (HS) approach where the MV precision is successively refined. After the optimal integer precision MV is obtained, this is first refined with half precision components: the half precision samples are computed by interpolating the reference samples in the region surrounding the integer precision solution; then, half precision ME is performed searching in the 8 locations around the integer precision solution. After the optimal half-pixel solution is obtained, the process is repeated at quarter-pixel precision.

Other approaches have been proposed specifically to reduce complexity of sub-pixel ME. Most methods in the literature were developed in the context of AVC. A fast fractional precision ME algorithm was proposed [5] in which AVC 16×16 macroblocks are classified in two classes as smooth or not. Sub-pixel ME is completely skipped in sub-blocks extracted from smooth macroblocks; conversely, an extended motion search is performed to find fractional refinements in non-smooth blocks. Similarly, another technique was proposed [6] in which the optimal sub-pixel ME location is predicted based on the behaviour of the residual error surface at integer precision. The prediction requires the integer precision solution to be derived using a full search ME algorithm on a minimum of 33×33 samples. A model is used to analytically compute the minimum of the surface, and correspondingly select fractional components. Methods were presented [7] [8] based on the idea that sub-pixel ME may be completely skipped in some blocks under particular conditions, using a 5×5 window of integer precision residual samples. Finally, recently a method was proposed [9] in the context of HEVC, based on error surface approximation. When using this algorithm, the error surface at integer precision is approximated by a second order function, and its minimum is used as the solution at sub-pixel precision.

Unfortunately, all these methods impose some restrictions on the integer precision ME algorithm being used. In particular, most methods rely on the availability of a complete window of residual error samples at integer precision. When using fast integer precision ME (such as EPZS or other similar techniques), only a reduced number of MV candidates are tested, which may be very distant one from the other. Approaches based on residual error surface models, or manipulation of a window of integer precision motion costs, can

The authors would like to thank Technology Strategy Board (TSB), UK, for co-funding this work as part of the THIRA consortium project (http://thiraproject.org.uk/).
This research utilised Queen Mary's MidPlus computational facilities, supported by QMUL Research-IT and funded by EPSRC grant EP/K000128/1. This work was partially supported by CNPq under grant 444864/2014-8.

not be applied in these cases unless the missing integer precision samples are tested, which would require additional computational time and therefore limit the benefits of these methods. Moreover, many of these techniques only target reducing complexity of the motion search step, while not considering the interpolation. In most HEVC implementations interpolation is computed on-the-fly directly on the interested region in the reference frames, and for this reason it accounts for a consistent part of the sub-pixel ME complexity, as further shown in the next section.

The method proposed in this paper is instead independent on the integer precision algorithm being used, and can be applied on top of any fast ME algorithm with no adaptation. Moreover, the proposed method targets all sub-pixel ME steps (namely interpolation and motion search, at both half and quarter precision), hence providing consistent speedups for each PU.

III. THE IMPACT OF SUB-PIXEL ME

The development of the algorithm proposed in this paper follows from an analysis of the complexity and efficiency of HEVC with respect to various encoding tasks. The HM reference encoder implementation was profiled while coding a number of sequences from well-known test sets [10] [11] [12] at different formats, spanning from Ultra High Definition (UHD, corresponding to a spatial resolution of 3140×2160 luma samples) to 1920×1080 (Full HD) and 1280×720 (720p) luma samples. Sequences were encoded at 4 different values of the Quantization Parameter (QP), namely 22, 27, 32 and 37, respectively. Encoding was performed according to the Common Test Conditions [13] specified while developing HEVC, where Random-Access Main configuration was used in all tests.

The execution times of several HEVC coding modules were measured while encoding. These were averaged for all QPs and all sequences. Fig. 1 (a) shows the average time spent by the HM encoder while performing inter-prediction, transform and quantization. It is clear from the figure that sub-pixel ME is the second most computationally expensive module while encoding inter-frames, taking on average 33.88% of the total encoding time. Fig. 1 (b) shows a breakdown of this time, depending on the particular sub-pixel ME task. Sub-pixel ME time is almost equally distributed between interpolation and motion search, and between half-pixel and quarter-pixel precision tasks.

While sub-pixel ME is very computationally complex, disabling it is not a viable solution because the coding efficiency gains provided by the MV fractional refinements are too high. The same sequences used to generate the statistics in Fig. 1 were encoded again at the same QP values using the HEVC reference encoder, but the encoder was modified to either completely skip sub-pixel ME in all blocks, or to only compute half-pixel precision refinements (and skip quarter precision refinements). Results of these tests were obtained in terms of BD-rates [14], a performance metric which computes the bitrate difference at the same quality levels with respect to an anchor, in percentage. Positive BD-rate values represent efficiency losses. Conventional HEVC was used as anchor. When completely disabling sub-pixel ME, average BD-rates of 7.01%, 6.27% and 16.33% were obtained for UHD, Full HD and 720p content, respectively. In case only quarter-pixel ME is disabled, BD-rates of 2.13%, 1.32% and 4.63% were obtained for each resolution, respectively.

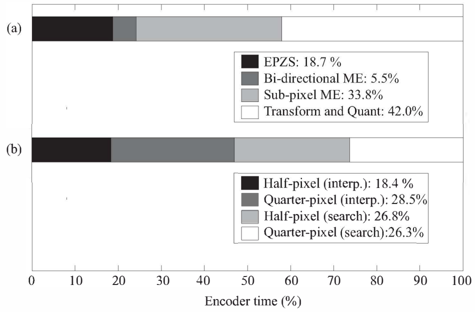

Fig. 1. (a) Average times spent by an HEVC encoder performing inter-prediction tasks. (b) Average times spent by an HEVC encoder performing sub-pixel ME tasks.

IV. ADAPTIVE PRECISION MOTION ESTIMATION

It has already been shown [8] that the behaviour of the residual error samples at integer precision can be used to predict the optimal MV precision. Residual samples can be interpolated to define an error surface. High curvatures of this surface correspond to cases in which the residual errors are changing rapidly in the surrounding of the integer solution: in these cases, a fractional solution could exist which considerably decreases the prediction error. Conversely, an almost flat residual error surface might indicate that fractional refinements are unlikely to impact the outcomes of ME. Unfortunately, most of the methods in the literature to estimate the residual error curvature require that all MVs in the surrounding of the optimal solution are tested at integer precision. Moreover, trying to model the residual error surface by means of complex interpolations or analytic models may be too time consuming, affecting the overall performance of the algorithm.

In this paper a different approach is proposed. Instead of trying to model the residual error surface, each block is characterised locally by means of a very simple and computationally inexpensive metric, which can be applied during ME regardless of the algorithm used for the motion search. The approach is then refined by taking into consideration global features of the frame, to better adapt to the characteristics of the encoded content.

A. *Block Characterisation Using Cost Slopes*

Consider that a PU is currently being tested with uni or bi-directional ME. The encoder tests a sequence of K MV candidates MV_i, $i = 0, ..., K-1$; the number K, order and components of such MVs depend on the current ME algorithm being used. Denote as x_i, y_i the horizontal and vertical components of MV_i respectively. For each MV, the encoder extracts the corresponding prediction block R_i from a reference frame. Denote as $X[h, w]$ and $R_i[h, w]$ the samples in the original block X and prediction R_i respectively, where $h = 0, ..., H-1$, $w = 0, ..., W-1$ and where H and W are the blocks height and width respectively. The distortion between these two blocks is computed, usually by means of the Sum of Absolute Differences (SAD), to obtain SAD_i.

The SAD is then usually adjusted with an estimate of the rate necessary to encode MV_i, to obtain a Rate-Distortion (RD) cost C_i. This is typically computed as:

$$C_i = SAD_i + \lambda B_{MV_i}, \tag{1}$$

where λ is a predefined Lagrangian multiplier, and B_{MV_i} is an estimate of the number of bits necessary to encode x_i and y_i.

Consider now that the MVs are referred to as MV_i, $i = 0, ..., K-1$ in ascending order depending on their costs, namely MV_i, $i = 0, ..., K-1$ such that $C_i \leq C_j$, $\forall i < j$. Conventional ME algorithms keep track of the minimum solution MV_0, namely the MV candidate such that $C_0 \leq C_k$, $\forall k$, and then proceed to computing fractional refinements, discarding all other costs and MVs tested during integer precision ME. On the other hand, the information collected while testing the K integer precision MVs might be manipulated and used to decide whether it is worth computing half precision refinements or not. In particular, consider that the encoder also keeps track of the following two MVs while performing integer precision ME:

1) The MV at maximum cost MV_{K-1}, namely the candidate such that $C_{K-1} \geq C_k, \forall k$.
2) The MV at second minimum cost MV_1, namely the candidate such that $C_0 \leq C_1 \leq C_k \ \forall k : k \geq 1$

Notice that these two MVs can be found during ME by means of simple cost comparisons, similarly to the computation of the minimum MV_0. Consider now for example that the maximum cost C_{K-1} is very similar to the minimum C_0: this means that all K MVs returned very similar costs (because all other costs are smaller or equal than C_{K-1}). It is therefore very unlikely that a fractional solution exists which returns a cost very different than C_0. Therefore C_0 is likely to be a good enough solution for the current PU. Conversely, consider that the second minimum cost C_1 is considerably higher than C_0: this means that all other MVs returned considerably higher costs than the minimum C_0, namely the residual error changes rapidly in the surrounding of the optimal solution. In this case it is worth computing fractional solutions because they might return costs which are considerably lower than C_0.

Extending this idea, MV_1 and MV_{K-1} can be compared with respect to MV_0 based on their costs found during integer precision ME, to adaptively select whether to perform half precision ME. A trivial way to obtain such comparison might consist in using the cost differences $C_1 - C_0$ (or $C_{K-1} - C_0$). Unfortunately, such value is not always significant. Especially when using fast ME algorithms, the MV candidates tested during the motion search may point to locations very far one from the other. In these cases, large values of the cost difference may not be indicative of the behaviour of the residual error in the surrounding of MV_0.

For this reason, comparison between MV_1 (or MV_{K-1}) and MV_0 is obtained in this paper in terms of the slope of their residual costs. Consider again the cost difference $\Delta(0,1) = |C_0 - C_1|$ (and similarly $\Delta(0, K-1)$ for MV_{K-1}). Define then the Euclidean distance between the locations pointed by MV_1 and MV_0 as:

$$D(0,1) = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}, \tag{2}$$

or $D(0, K-1)$ for MV_{K-1}. The cost slope between MV_1 and MV_0 is then defined as:

$$S_1 = \frac{\Delta(0,1)}{D(0,1)}, \tag{3}$$

and similarly S_{K-1} can be obtained for MV_{K-1}.

Ideally, PUs which return small values of S_{K-1} and S_1 at integer precision should correspond to blocks where half precision ME can be skipped without affecting too much the encoding efficiency. Conversely, large values of S_{K-1} and S_1 might indicate rapid variations in the residual errors at integer precision, which could correspond to a strong impact of sub-pixel ME on coding efficiency. Such characterisation by means of cost slopes can be used to define a fast algorithm for HEVC sub-pixel ME, as will be illustrated in the rest of this paper. Notice that, while the aforementioned characterisation was introduced in the context of integer precision ME, the same process can be applied during half precision ME, to decide whether to perform quarter precision ME or not. In particular, assuming that the encoder tests L MVs at half precision MV_i^h, $i = 0, ..., L-1$, the MVs at minimum, second minimum and maximum costs can be computed among such candidates, referred to as MV_0^h, MV_1^h and MV_{L-1}^h respectively in the rest of this paper. Finally, the half cost slopes S_1^h between MV_0^h and MV_1^h and S_{L-1}^h between MV_0^h andMV_{L-1}^h can be computed using the expressions in Eq. 2 and 3.

The cost slopes defined in the previous subsection can be used to adaptively select the precision of ME for each PU. The integer cost slopes S_{K-1} and S_1 can be compared against two predefined thresholds T_{K-1} and T_1: only if both cost slopes are above the thresholds, half precision ME is computed. Similarly, in case a half precision solution is computed, the half cost slopes S_1^h and S_{L-1}^h can be compared against two predefined thresholds T_{L-1}^h and T_1^h to decide whether to perform quarter precision ME.

This method requires the derivation of four thresholds T_{K-1}, T_1, T_{L-1}^h and T_1^h. Such derivation is crucial for the performance of the proposed method. On one hand, the computation of half or quarter precision MV refinements should be avoided in the largest number of cases to achieve the highest speedups (namely thresholds should be as high as possible). On the other hand, too high values of the thresholds could lead to large efficiency losses, due to the fact that the encoder may skip the computation of fractional refinements too often. Thresholds should be selected to achieve the best trade-off between these conflicting factors. Ideally such trade-off could be achieved by defining thresholds in such a way that fractional refinements are skipped in roughly a certain predefined percentage P of the total number of encoded PUs. By appropriately defining the optimal value of P, consistent speedups may be obtained for relatively low efficiency losses.

In order to perform such derivation, the distribution of the cost slopes defined in the previous section was investigated. Notice that all tests and results performed at this purpose were derived on a different test set than the one used to validate the proposed method in the Results section. In particular, a set of sequences was encoded with conventional HEVC, and the cost slopes were computed for each PU. The frequency of occurrence of each of the 4 cost slopes was then studied after the encoding. Fig. 2 shows the results of such analysis. Each plot presents the values of P, namely the cumulative number of PUs which resulted in a cost slope smaller than a certain value, in percentage with respect to the total number of encoded PUs. Values of P were clustered in percentage bins, each spanning a step of 5%. For instance, the plot in the top-right portion of the figure shows that around 90% of the PUs resulted in values of S_{L-1}^h smaller than 2, and around 95% resulted in values smaller than 3. Assuming a certain target probability P, the corresponding thresholds can in theory be obtained from the distributions in the figure. In particular these distributions can be easily embedded in an encoder by means of appropriate look-up tables, which assign a specific threshold to each percentage bin.

B. Frame Characterisation using Edge Estimation

The approach proposed in the previous subsection was tested using two different values of P (equal to 30 and 40, respectively). The same sequences used to generate the statistics in Fig. 2 were used at this purpose. Results of these tests in terms of BD-rate losses can be found in the second and third columns in Table I, for $P = 30$ and $P = 40$ respectively. It is clear from these results that using a fixed target percentage P across all sequences is not ideal. The BD-rate losses drastically vary across different sequences. In case of $P = 40$, up to 11.10% BD-rate losses are reported for the Station sequence, whereas only 0.22% losses are reported using the same thresholds on the Mobcal sequence. These numbers indicate that the impact of fractional refinements on coding efficiency is highly content dependent: even though sub-pixel ME is skipped on roughly the same number of PUs, very different results are obtained across the test set.

These effects have already been studied in the literature, where global features of each frame were shown to be correlated with the impact of sub-pixel ME on coding efficiency. In particular, it was shown [15] that sequences containing computer generated imagery or screen content do not typically benefit from sub-pixel ME, therefore sub-pixel ME may be completely disabled when coding this kind of content. This behaviour can be easily explained by noticing that these signals contain a high amount of sharp edges and high frequency content. On the other hand, the interpolation filters used to compute fractional reference samples act as low-pass filters and therefore are not good at predicting this high frequency content. Hence, fractional refinements are rarely used in these cases.

These findings can be exploited for the purpose of the approach proposed in this paper. In theory, frames containing high amount of edges should correspond to cases where sub-pixel ME can be skipped in a large percentage of PUs without affecting much the coding efficiency. Conversely, frames with low amount of edges may benefit from fractional refinements. In order to validate this assumption and possibly exploit it, the encoder should be able to estimate the number of edges in each frame in the sequence. A simple and effective technique to obtain such estimate consists in using the Sobel operator [16], a well-known filter widely used in edge detection algorithms. The Sobel operator is obtained by convolving two 3×3 kernels to the original image, to obtain an estimate of the horizontal and vertical gradients of the image, respectively. These can then be combined to obtain the gradient magnitude. High values of the gradient magnitude might correspond to the picture edges. The gradient magnitude can then be binarised (where all magnitudes above a predefined value are set to 1, and all others are set to 0). The percentage of 1s in the binarised gradient magnitude can then be taken as a rough estimate of the amount of edges in the frame, and it is referred to as the edge ratio in the rest of this paper.

The edge ratio was computed for each frame in the sequences in Table I and then averaged among all frames. Results are shown in the right-most column. Clearly, sequences which produced high efficiency losses generally present very low values of the average edge ratio; conversely, high values of the edge ratio seem to indicate that sub-pixel refinements can be skipped more often. This behaviour can be easily exploited by defining a correspondence between edge ratios and target percentages P. In this paper, such relationship was derived from empirical analysis, and it was defined in the form of a linear function $P = aE + b$, where E is the edge ratio computed on a frame, P is the output target percentage, and a and b were set to 4 and 10, respectively.

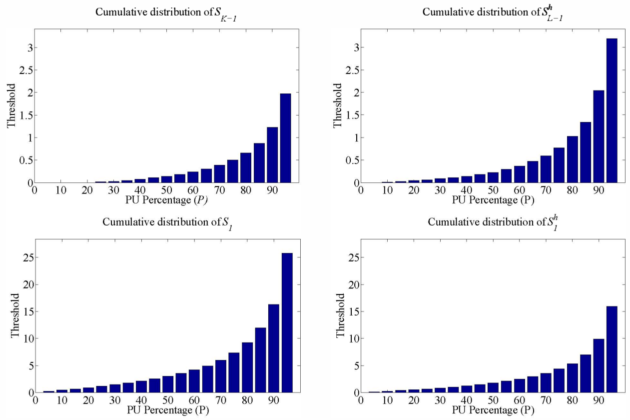

Fig. 2. Cumulative distributions of the cost slopes. For each value of P in the horizontal axis, the height of the corresponding bar represents the cost slope value T such that P PUs (in percentage) are smaller or equal than T.

C. Proposed Algorithm

The steps illustrated in this section can be combined to obtain an algorithm to reduce complexity of HEVC inter-prediction, referred to as Adaptive Precision ME (APM). The proposed algorithm works as follows. First, the edge ratio E is computed on a frame using the Sobel filter. This is then used to derive the corresponding value of P as illustrated in the previous subsection. P is quantised to 5%-wide bins, and the obtained valued is used to derive the thresholds T_{K-1}, T_1, T^h_{L-1} and T^h_1 based on the available look-up tables.

The encoder starts encoding CUs. Each CU is split into PUs according to an inter-prediction mode. Then for each PU:

1) Integer precision ME is performed to obtain the minimum solution MV_0. MV_1 and MV_{K-1} are also computed during the motion search. The integer cost slopes S_{K-1} and S_1 are computed.
2) If $S_{K-1} < T_{K-1}$ and $S_1 < T_1$, the algorithm exits. Else, next step is performed.
3) Half precision ME is performed to obtain the minimum solution MV^h_0. MV^h_1 and MV^h_{L-1} are also computed during the motion search. The half cost slopes S^h_{L-1} and S^h_1 are computed.
4) If $S^h_{L-1} < T^h_{L-1}$ and $S^h_1 < T^h_1$, the algorithm exits. Else, next step is performed.
5) Quarter precision ME is performed to obtain the MV solution at quarter-pixel resolution.

The algorithm is applied twice per PU, for uni-directional and bi-directional ME respectively. The rest of the encoding continues as in conventional HEVC.

V. RESULTS

The proposed algorithm was implemented on top of a conventional HEVC encoder making use of EPZS (for uni-directional integer precision ME) and windowed full search (for bi-directional integer precision ME). In case half or quarter precision refinements are computed on a PU (namely if steps 4 or 5 in the algorithm in Subsection IV-C is performed), these are obtained searching on the 8 candidates closest to the solution at previous precision, as in conventional HS. The obtained modified encoder is referred to here as HEVC-AMP. Performance of HEVC-AMP was validated on sequences at UHD, 2560×1600 (1600p), Full HD and 720p resolutions from well known test sets [17] [11] [12]. Common Test Conditions [13] in the Random-Access Main configuration were used for these experiments. To further evaluate performance of the approach, HEVC-AMP was compared with a modified HEVC encoder in which fractional refinements are skipped in all cases (i.e. MVs are forced to integer precision). Such an encoder provides the

TABLE I

CODING EFFICIENCY LOSSES USING THE PROPOSED ALGORITHM WITH FIXED THRESHOLDS, AND AVERAGE EDGE RATIO.

	Sequence	**BD-rates** ($P = 30\%$)	**BD-rates** ($P = 40\%$)	**Edge Ratio**
UHD	Book	3.60	4.82	0.57
	MenAndPlants	2.86	3.84	0.92
	ParkAndBuildings	1.14	2.13	5.72
	Wood	0.20	0.25	14.70
Full HD	Station	7.89	11.10	0.95
	Sunflower	2.82	3.24	0.63
	Parkscene	1.71	2.68	4.38
	DucksTakeOff	0.67	1.02	16.16

theoretical upper bound limits to the efficiency losses and speedups which can be provided by HEVC-AMP, and is referred to here as HEVC-INT. Coding efficiency of HEVC-AMP and HEVC-INT is presented in terms of BD-rates, in percentage, where HEVC with HS was used as anchor in all tests (referred to as HEVC-HS), to obtain BD_{AMP} and BD_{INT}, respectively. Similarly, the total encoding times required by HEVC-AMP, HEVC-INT and HEVC-HS were measured and averaged across all QPs for each sequence, to obtain T_{APM}, T_{INT} and T_{HS} respectively. Finally, speedups (in percentage) of HEVC-AMP with respect to the anchor were computed as:

$$\Delta_{AMP} = 100\frac{T_{HS} - T_{APM}}{T_{HS}},$$

and similarly speedups of HEVC-INT were computed as Δ_{INT}.

Full results are presented in Table II. The proposed HEVC-AMP encoder is considerably faster than the anchor in all tests, achieving average speedups of 13.0% Δ_{AMP} on the entire test set, at a very modest cost of 1.3% BD_{AMP} losses. Up to 17.6% speedups were obtained in the Vehicles sequences, at a cost of 1.95% BD_{AMP} losses. On the contrary, completely disabling sub-pixel ME provides average 11.2% BD_{INT} losses for average speedups of 31.2% Δ_{INT}. The proposed method is therefore capable of obtaining average 42% of the maximum achievable speedups (generated by HEVC-INT), but at a fraction of the costs in terms of efficiency losses. Also, while HEVC-INT is very content dependent, the proposed ADP approach is instead capable of adapting to different content thanks to considering both global and local features. For instance, BD_{INT} losses of 1.5% and 8.0% were obtained in the Bosphorus and BundNightscape sequences, highlighting that fractional refinements have a much higher impact on the latter sequence. ADP can adapt to such variations, providing low BD_{ADP} losses of 0.3% and 0.7%, respectively in these two sequences. It is also interesting to report that no significant variation of performance of the method was observed in the tests across different resolutions, highlighting the fact that ADP can be used with no adaptation on content at different resolutions.

Finally, notice that while the approach requires the encoder to perform some additional computations with respect to conventional HEVC (for the Sobel filter and cost slopes computation), the complexity of these operations is relatively very small and is completely compensated by the time savings obtained by the method. To further validate this assumption, the proposed APM encoder was modified to always perform sub-pixel refinements, even when cost slopes are found below the thresholds. Such an encoder performs all the additional operations required by APM, but does not benefit from the corresponding speedups. The time overhead required by such an encoder with respect to HEVC with HS to encode the sequences in Table II is on average only 0.5%.

TABLE II
RESULTS OF APM WITH RESPECT TO CONVENTIONAL HEVC USING HS.

	Sequence	BD_{AMP}	Δ_{AMP}	BD_{INT}	Δ_{INT}
UHD	Vehicles	1.95	17.6	16.13	30.63
	TallBuildings	0.97	15.38	14.29	34.66
	Library	1.31	14.73	10.21	35.93
	Bosphorus	0.27	12.45	1.49	32.95
	BundNightscape	0.68	12.26	8	35.53
	RainFruits	0.54	12.03	6.85	33.2
	CalendarAndPlants	0.71	11.91	6.95	33.78
	ConstructionField	0.52	11.85	8.53	36.52
	LupoCandlelight	0.44	10.35	8.45	33.49
1600p	Traffic	1.67	14.3	13.11	32.7
	ParkJoy	1.84	12.1	7.91	24.6
Full HD	BQTerrace	2.73	15.7	21.71	29.8
	ChristmasTree	2.31	13.0	12.19	25.8
	Wisley2	2.47	12.2	18.45	28.1
	Kimono	0.69	10.0	5	28.6
720p	Stockholm	2.49	16.9	19.08	31.8
	ParkRun	0.83	8.9	11.52	22.9
	Average	**1.31**	**13.0**	**11.17**	**31.2**

VI. CONCLUSIONS

A fast algorithm for reducing complexity of HEVC sub-pixel ME was proposed in this paper, based on local features of each PU (the cost slopes of residual errors) and global features of each frame (the amount of edges in the picture). The method was shown achieving consistent speedups with respect to previously proposed algorithms at very small efficiency losses.

A detailed analysis may be performed to find other global features correlated with the outcomes of fractional ME, to possibly obtain a more complex characterization of each frame. This may be used to improve the threshold selection used during the proposed algorithm, and possibly enhance performances of the approach.

REFERENCES

[1] G. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 560–576, 2003.

[3] ITU. HM Reference Software. [Online]. Available: https://hevc.hhi.fraunhofer.de/HM-doc/

[4] A. Tourapis, O. Au, and M.-L. Liou, "Highly efficient predictive zonal algorithms for fast block-matching motion estimation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 12, no. 10, pp. 934–947, Oct 2002.

[5] S. Zhu and J. Tian, "An improved fast fractional pel motion estimation algorithm based on h.264," in *Industrial Technology (ICIT), 2010 IEEE International Conference on*, March 2010, pp. 179–182.

[6] Q. Zhang, Y. Dai, and C.-C. Kuo, "Fast sub-pel motion vector prediction via block classification," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, Nov 2009, pp. 1569–1572.

[7] S. G. Blasi and E. Izquierdo, "Residual error curvature estimation and adaptive classification for selective sub-pel precision motion estimation," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, March 2012, pp. 1193–1196.

[8] S. G. Blasi, I. Zupancic, and E. Izquierdo, "Fast motion estimation discarding low-impact fractional blocks," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*, Sept 2014, pp. 201–205.

[9] W. Dai, O. Au, C. Pang, L. Sun, R. Zou, and S. Li, "A novel fast two step sub-pixel motion estimation algorithm in hevc," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, March 2012, pp. 1197–1200.

[10] R. Weerakkody, M. Naccari, and M. Mrak, "UHD test sequences," JCTVC-O0332, Tech. Rep., November 2013.

[11] L. Song, X. Tang, W. Zhang, X. Yang, and P. Xia, "The SJTU 4K video sequence dataset," in *Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop on*, July 2013, pp. 34–35.

[12] L. Haglund, "The SVT high definition multi format test set," Sveriges Television, Tech. Rep., February 2006.

[13] F. Bossen, "Common test conditions and software reference configurations," JCTVC-L1100, Tech. Rep., October 2012.

[14] G. Bjontegaard, "Improvements of the BD-PSNR model," ITU-T SG16/Q6, 35th VCEG Meeting, Doc.VCEG-AI11, 2008.

[15] Y. Zhou, B. Li, J. Xu, G. Sullivan, and B. Lin, " Motion Vector Resolution Control for Screen Content Coding," JCTVC-P0277, Tech. Rep., January 2014.

[16] I. Sobel, *An isotropic* 3×3 *image gradient operator.* Academic Press, 1990.

[17] R. Weerakkody and M. Mrak, "High efficiency video coding for ultra high defnition television," in *Proc. 2013 NEM Summit*, October 2013, pp. 9–14.