Efficient HEVC intra-frame prediction using curved angular modes

R. Fernandes[✉], G. Sanchez, R. Cataldo, T. Webber and C. Marcon

Blending curved and straight lines on angular modes are proposed for intra-frame prediction in high-efficiency video coding (HEVC). All 33 angular modes in HEVC have received an offset-based displacement calculation to each predicted sample, so that the resulting prediction block models image regions with curved textures. The proposal includes a negligible overhead in the syntax elements of the encoded bitstream (up to 2.56%) to transmit the curvature parameter along with the angular mode of a predicted block. Results demonstrate increased prediction accuracy and consequent reduction in residuals. There is a fair trade-off between prediction accuracy and computational complexity even considering few curvature parameters. Further evaluations show an average reduction in Bjøntegaard delta rate of 3.19% for the HEVC test sequences.

Introduction: Recent video applications require high quality and high definition. High compression ratio with minimal perceivable quality degradation is one of the commonly found goals in the video coding state-of-the-art. Modern video encoders such as the high-efficiency video coding (HEVC) [1] adopt advanced methods that exploit the best trade-off among encoding quality, bitrate, and computational complexity. HEVC employs the hybrid encoding of prediction and its residuals. The predictive step adopts intra- and inter-frame predictions for encoding video frames. Intra-frame prediction explores the spatial redundancy of neighbouring blocks [2], whereas inter-frame prediction is based on redundant information of past encoded frames [3]. After the predictive step, the encoder transforms and quantises the prediction error (called residuals) [4]. Finally, these residuals along with the high-level syntax elements of the encoding process are entropy encoded into the video bitstream, using context adaptive binary arithmetic coding.

Prediction plays a crucial role in the encoded video efficacy and efficiency, as a high-quality prediction reduces the residuals and consequently the bitstream size [5]. Thus, the investigation of efficient prediction methods becomes essential for the current and next-generation standards. This Letter focuses strictly on the intra-prediction step of video coding using HEVC as the baseline for the remaining explanations. Our work extends the prediction capabilities of HEVC by introducing new angular intra-prediction modes. The proposal relies on ways of capturing texture characteristics more accurately using curved lines instead of assuming only angular straight lines. The approach brings better prediction quality reducing the residuals; therefore, improving the encoding efficiency.

HEVC intra-frame prediction: The HEVC intra-frame prediction exploits the natural spatial redundancy inside a frame to compress information [2]. HEVC employs a range of prediction methods to efficiently predict different kinds of contents. These methods can be divided into two classes: (i) angular prediction, totalling 33 modes (17 horizontal and 18 vertical, each one with an angular parameter A) that model the directional edges in images and (ii) planar and DC prediction, modelling smooth image content. The HEVC intra-prediction uses a *reference array* computed from adjacent reconstructed blocks to extrapolate the predicted block contents. The prediction process implies a high computational effort since all modes are recursively evaluated to obtain an efficient encoding, with block sizes ranging from 64 × 64 down to 4 × 4. Additionally, the prediction incurs residuals that need to be transformed and quantised, the latter a lossy compression step. Still, increasing the encoding complexity is often acceptable, as most video content is only encoded once, but decoded countless times. Thus, a more accurate prediction reduces residuals, improving encoding efficiency.

Curve-based intra-frame prediction: This proposal extends the 33 angular intra-prediction HEVC modes by shifting samples during the prediction step, based on the predicted sample position together with one additional information: the curve offset weight. Equation (1) demonstrates how to compute samples p in a predicted block [6], where t assumes values of y or x for horizontal and vertical modes, respectively, with ref as the *reference array*. The projected integer displacement i, shown in (2), depends on the row $t = y$ for vertical or the column $t = x$ for horizontal modes, and the angular parameter A. Finally, (3) shows the fractional part of the projected displacement f that depends on the row $t = y$ for vertical modes and the column $t = x$ for horizontal modes, and the angular parameter A

$$
\begin{aligned}
p[x][y] = ((32 - f) \times \mathrm{ref}[t + i + 1] \\
+ f \times \mathrm{ref}[t + i + 2] + 16) \gg 5
\end{aligned}
\tag{1}
$$

$$i = ((t + 1) \times \mathrm{A}) \gg 5 \tag{2}$$

$$f = ((t + 1) \times \mathrm{A})\ \&\ 31 \tag{3}$$

Let Hb be half of the predicted block size, d the distance from the predicted sample to the centre line of the block, given by (4), and ω the curve offset weight. Equation (5) describes the *shift* applied to the reference sample computed by HEVC intra-prediction. Thus, samples near the centre are most affected, since $\mathrm{Hb} - d$ is maximum in the centre of the block, as it can be seen in Fig. 1 for an 8 × 8 prediction block. We have chosen this approach so that curves resemble an arc, bending at the outer edges of the predicted block

$$
d = \begin{cases}
\mathrm{Hb} - y - 1, & y < \mathrm{Hb},\ \text{vertical modes} \\
y - \mathrm{Hb}, & y \geq \mathrm{Hb},\ \text{vertical modes} \\
\mathrm{Hb} - x - 1, & x < \mathrm{Hb},\ \text{horizontal modes} \\
x - \mathrm{Hb}, & x \geq \mathrm{Hb},\ \text{horizontal modes}
\end{cases}
\tag{4}
$$

$$\mathrm{shift} = (\mathrm{Hb} - d) \times \omega/\mathrm{Hb} \tag{5}$$

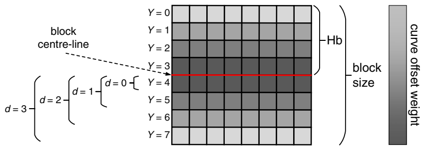

Fig. 1 *Curve offset weights (ω) for 8 × 8 prediction block in vertical mode. Samples located near centre line are more influenced by curve offset weight, while samples further away from centre line are less affected*

Solving (5), the *shift* parameter is included in (1) at the reference sample indexing step, resulting in (6), which is used to compute reference sample values with curve offset weights. Note that when $\omega = 0$, the prediction behaves exactly like in standard HEVC. This approach results in minimal computational overhead, as the only required step is solving (5) and the modified equation below:

$$
\begin{aligned}
p[x][y] = ((32 - f) \times \mathrm{ref}[t + i + \mathrm{shift} + 1] \\
+ f \times \mathrm{ref}[t + i + \mathrm{shift} + 2] + 16) \gg 5
\end{aligned}
\tag{6}
$$

Regarding the high-level syntax elements, some elements are added in the bitstream for signalling the usage of the ω parameter: *curve_offset_flag* – a 1 bit flag, which specifies if ω is used in the predicted block; and *curve_offset* – an identifier of the ω value. The *curve_offset* is n bits long as our experiments assume varying ranges of ω values. Let θ be the number of ω values, then $n = \lceil \log_2 \theta \rceil$, e.g. $\theta = 2$ requires 1 bit for representing ω values −1 and 1.

Whenever $\omega = 0$ in the intra-predicted block, *curve_offset_flag = 0* is set and *curve_offset* is omitted from the bitstream. Therefore, the syntax element that identifies the ω value is only used when required, optimising the encoded bitstream by omitting unnecessary information.

Setup and results: The HEVC test Model version 16.18 [7] was applied to conduct tests with all of HEVC's test sequences in all-intra main profile. All tests adopt an increasing amount of ω values to cover a wide range of testing scenarios. Our results evaluate the utilisation of intra-predicted blocks with curve offset weights, as well as an analysis of the Bjøntegaard delta rate (BD-rate) in all test sequences.

Intra-prediction mode distribution: The proposal requires the encoder to evaluate additional modes according to θ. When employing two values for ω (−1 and +1), the encoder evaluates a total of 101 modes for intra-prediction: (i) two for planar and DC; (ii) 33 for the standard HEVC angulars; (iii) 33 for $\omega = -1$; and (iv) 33 for $\omega = +1$. Thus, having θ as the amount ω values, the amount of intra-prediction modes is given by $2 + (33 \times (\theta + 1))$.

Despite angular modes with $\omega = 0$ (i.e. default HEVC angular modes) being more used than angular modes with curves, the latter is still significantly employed in our test results, shown in Table 1. With

1350911x, 2018, 21, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/el.2018.6051 by CAPES, Wiley Online Library on [11/11/2022]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

$\theta = 2$, 34.47% of the chosen modes correspond to the curved angular modes, whereas 65.53% of the intra modes used are standard HEVC. This proportion increases in favour of the curved angular modes with higher θ values, up to 50.97% with $\theta = 20$. Thus, the additional prediction modes provided by curves can model more situations than the 35 modes of HEVC.

Table 1: Utilisation of prediction modes for varied θ values

Curve offsets (θ)	Intra/DC modes (%)	HEVC angular modes (%)	Curved modes (%)	Syntax bitstream overhead (%)
0	28.47	71.53	—	—
2	19.74	45.79	34.47	0.58
4	16.19	44.37	39.44	0.99
6	14.85	42.80	42.35	1.42
8	13.60	41.99	44.41	1.49
10	13.24	40.76	46.01	1.92
12	12.98	39.71	47.31	1.98
14	12.65	38.93	48.42	2.02
16	11.90	38.73	49.37	2.06
18	12.21	37.57	50.22	2.52
20	11.98	37.05	50.97	2.56

BD-rate analysis and encoding effort: Table 1 shows the HEVC encoder quite often uses modes with $\omega \neq 0$. Thus, better encoding efficiency is expected, as these modes have been selected by offering a better prediction than that of standard HEVC. Tables 2 and 3 illustrate, for the luma component, the average BD-rate results for all test sequence classes (A–F) varying the θ value (even numbers from 2 to 20). The adoption of angular modes with curves has reduced the measured BD-rate with varying improvements. Increasing θ does not imply rising BD-rate gains due to the bitstream overhead required to encode the syntax elements.

Table 2: Average BD-rate reduction with $2 \leq \theta \leq 10$

Sequence	Curve offsets (θ)				
	2 (%)	4 (%)	6 (%)	8 (%)	10 (%)
Class A	−1.78	−2.17	−2.02	−2.22	−1.85
Class B	−1.69	−2.42	−2.48	−2.89	−2.63
Class C	−2.73	−3.74	−3.89	−4.42	−4.24
Class D	−2.48	−3.41	−3.56	−4.03	−3.84
Class E	−3.13	−3.78	−3.65	−3.93	−3.57
Class F	−2.99	−3.85	−3.90	−4.38	−4.16
average A–F	−2.41	−3.17	−3.20	−3.60	−3.34

Table 3: Average BD-rate reduction with $12 \leq \theta \leq 20$

Sequence	Curve offsets (θ)				
	12 (%)	14 (%)	16 (%)	18 (%)	20 (%)
Class A	−1.87	−1.95	−1.98	−0.49	−0.75
Class B	−2.78	−2.93	−3.11	−1.83	−2.14
Class C	−4.44	−4.67	−4.91	−3.21	−3.96
Class D	−4.06	−4.22	−4.45	−3.01	−3.45
Class E	−3.61	−3.70	−3.74	−1.69	−2.00
Class F	−4.38	−4.55	−4.78	−3.21	−3.82
average A–F	−3.52	−3.67	−3.80	−2.24	−2.69

Going from standard HEVC with $\theta = 0$ to 2 incurs an average BD-rate reduction of 2.41%. Meanwhile, using $\theta = 4$ reduces the average BD-rate by 3.17%, which is still a significant improvement over standard HEVC, and 0.76% better than the average BD-rate achieved with the $\theta = 2$. However, $\theta = 6$ improves the BD-rate by only 0.03% compared with $\theta = 4$. This minimal gain is due to the poor utilisation of the additional syntax elements. In other words, while $\theta = 2$ needs one additional bit, $\theta = 4$ requires two extra bits, and $\theta = 6$ requires three, whose binary combinations are not entirely used for encoding the binary representation of the six ω values. Meanwhile, $\theta = 8$ introduces further BD-rate gains (3.60 or 0.40% when compared with $\theta = 6$) by compensating the bitstream overhead with a better prediction quality. Fig. 2 shows this characteristic. Still, there is a saturation point where no further encoding gains can be observed considering the bitstream overhead. Considering experiments with $\theta > 20$, results for $\theta = 32$ showed an average BD-rate reduction of 3.85%, which is an insignificant improvement when compared with $\theta = 16$. Although $\theta = 32$ provides a better prediction, it implicates in twice the computational complexity of $\theta = 16$.

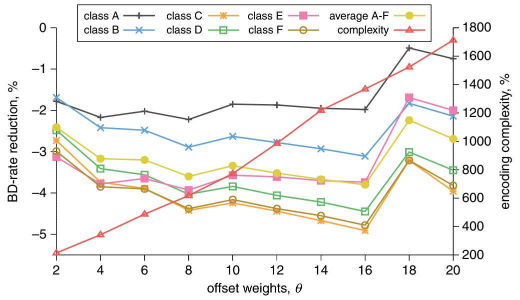

Fig. 2 *BD-rate reduction with θ ranging from 2 to 20, for each test sequence class*

Fig. 2 illustrates the encoding effort associated with the proposed model for increasing θ values. Using the additional encoding modes directly translates to the encoding time, triplicating the computational effort for $\theta = 2$, and linearly increasing with higher θ values. As such, the usage of many curve offset parameters becomes less attractive; nevertheless, if the goal is encoding efficiency, this might be a worthwhile trade-off. Regarding the decoder complexity, this model proposal does not require additional computing effort than in standard HEVC.

Conclusion: This work proposes an extension of the HEVC angular intra-prediction modes by adopting a curve offset weight. The angular predicted blocks could model samples being displaced as a curve instead of a straight line, providing a better fit to their original content, thus reducing residuals and consequently improving the encoding efficiency. Results demonstrate an average BD-rate reduction of 3.19% for all tests, with at most 4.91% in Class C videos with $\omega = 16$. This Letter shows that $\theta \leq 4$ offers a fair trade-off in BD-rate gains versus encoding complexity. Moreover, experiments showed that better encoding efficiency can be achieved with higher θ values.

© The Institution of Engineering and Technology 2018

Submitted: *26 June 2018* E-first: *12 September 2018*
doi: 10.1049/el.2018.6051
One or more of the Figures in this Letter are available in colour online.

R. Fernandes, G. Sanchez, R. Cataldo and C. Marcon (*Escola Politécnica, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), Porto Alegre – RS, Brazil*)

✉ E-mail: ramon.fernandes@acad.pucrs.br

T. Webber (*Departamento de Computação, Universidade de Santa Cruz do Sul (UNISC), Santa Cruz do Sul – RS, Brazil*)

References

1 Sullivan, G., Ohm, J., Han, W.-J., *et al.*: 'Overview of the high efficiency video coding (HEVC) standard', *Trans. Circuits Syst. Video Technol.*, 2012, **22**, (12), pp. 1649–1668, doi: 10.1109/TCSVT.2012.2221191

2 Lainema, J., Bossen, F., Han, W.-J., *et al.*: 'Intra coding of the HEVC standard', *Trans. Circuits Syst. Video Technol.*, 2012, **22**, (12), pp. 1792–1801, doi: 10.1109/TCSVT.2012.2221525

3 Rhee, C., Lee, K., Kim, T., *et al.*: 'A survey of fast mode decision algorithms for inter-prediction and their applications to high efficiency video coding', *Trans. Consum. Electron.*, 2012, **58**, (4), pp. 1375–1383, doi: 10.1109/TCE.2012.6415009

4 Winken, M., Helle, P., Marpe, D., *et al.*: 'Transform coding in the HEVC test model'. Int. Conf. Image Processing (ICIP), Brussels, Belgium, December 2011, pp. 3693–3696, doi: 10.1109/ICIP.2011.6116521

5 Li, J., Li, B., Xu, J., *et al.*: 'Efficient multiple-line-based intra prediction for HEVC', *Trans. Circuits Syst. Video Technol.*, 2016, **28**, (4), pp. 947–957, doi: 10.1109/TCSVT.2016.2633377

6 Vivienne, S., Budagavi, M., and Sullivan, G.-J.: 'High efficiency video coding (HEVC): algorithms and architectures' (Springer International Publishing, Cham, Switzerland, 2014, 1st edn.)

7 JCT-VC: 'HEVC test model HM-16.8'. Available at https://hevc.hhi.fraunhofer.de/svn/svnHEVCSoftware/, accessed May 2018

1350911x, 2018, 21, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/el.2018.6051 by CAPES, Wiley Online Library on [11/11/2022]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License