Int. J. Electron. Commun. (AEÜ) 114 (2020) 153000

Contents lists available at ScienceDirect

International Journal of Electronics and Communications (AEÜ)

journal homepage: www.elsevier.com/locate/aeue

Regular paper

# Pixelwise improvised blend of predictors in HEVC lossless mode

S. Shilpa Kamath [a,*], P. Aparna [a], Abhilash Antony [b]

[a] National Institute of Technology Karnataka, Karnataka, India
[b] Muthoot Institute of Technology and Science, Kerala, India

ARTICLE INFO

*Article history:*
Received 17 August 2019
Accepted 17 November 2019

*Keywords:*
HEVC
Lossless
Residuals
Blend
Sub-predictors

ABSTRACT

The commendable work by the two video coding pioneers ISO/IEC and ITU-T, to handle the next-generation of multimedia services has led to the evolution of High Efficiency Video Coding (HEVC) standard. The lossless mode of HEVC is essential when no loss in fidelity is desired to aide most of the real-world applications like video analytics, web collaboration, remote desktop sharing, etc. The proposed work intends to improvise the HEVC intra prediction scheme through the application of the heuristic history-based blend of predefined sub-predictors, while in lossless mode. The prime element of the locally adaptive mechanism is the derivation of the penalizing factors that are imposed on the sub-predictors, based on the neighborhood residuals. The experimental analysis highlights that the proposed method outperforms the lossless mode of HEVC anchor and the prevalent state-of-the-art prediction techniques in terms of savings in bit-rate which is achieved without any increase in run-time.



## 1. Introduction

The proliferation of multimedia services demands better compression strategies to meet the constraints in bandwidth and storage space. The associated challenges are commonly confronted when dealing with high-resolution video content. To address these issues globally, the Joint Collaborative Team on Video Coding (JCT-VC) was formed by the experts from ISO/IEC and ITU-T. High Efficiency Video Coding (HEVC) [1], a promising video coding standard developed conjointly by this team incorporates a sophisticated coding mechanism than the one present in its predecessor and the de facto standard H.264/Advanced Video Coding (AVC) [2]. This feature helps to accomplish a reduction in the bit-rate at the cost of augmented computational complexity while retaining the video perceptual quality. There are several decisions to be made that involves numerous computations [3–5] while encoding a given video content, thereby paving the way for an increase in the encoding latency.

The HEVC standard supports the lossless coding of the video sequences [6]. This feature is extremely beneficial whenever certain applications demand distortion-free reconstruction of the original content. Some of the applications that mandate lossless data reconstruction are video analytics, medical data compression, etc. Additionally, there are certain other multimedia applications such as cloud video gaming, video-conferencing, desktop sharing, etc. that have given rise to a class of video sequences termed as the screen content (SC) [7], requiring to maintain the original video integrity through lossless compression.

In lossless coding mode, as the quantization and transformation blocks are by-passed, sample-based strategies are proven to dominate over the conventional block-based prediction techniques. These strategies are based on the differential pulse code modulation (DPCM) techniques [8–11] that are mainly designed for lossless coding mechanisms. Zhou et al. [8] proposed a sample-based angular prediction (SAP) scheme to generate the prediction by exploiting the spatial redundancy of the proximally located samples corresponding to the target pixel. The DPCM based intra prediction improvement techniques mentioned in [9] are sample-based weighted prediction (SWP) and directional template matching (DTM) algorithms. While the SWP technique carries out the target pixel prediction by computing the weighted average of the surrounding pixels, DTM identifies the causal pixel with the closest similarity to the target pixel. In residual DPCM (RDPCM) [10], the DPCM is imposed on the generated residuals at the residue coding stage, after the application of the regular blockwise intra prediction. Another DPCM based technique proposed by Seemann and Tischer [11] has been applied to the image and audio data, by making use of the blending mechanism to create a compound predictor based on the regional adaptation. Most of the video content is heterogeneous in nature, comprising of several edges and textural structures. The DPCM based prediction in the simplest form cannot simultaneously deal with the pattern that includes smooth regions, regions with discontinuities, etc. within an image block. To handle

* Corresponding author.
*E-mail address:* shilpa.1107@yahoo.co.in (S. Shilpa Kamath).

https://doi.org/10.1016/j.aeue.2019.153000
1434-8411/

the diverse nature of the HEVC test video sequences, the history-based heuristic blend of sub-predictors is used to derive the optimum prediction mode for a target block. The mechanism of blending the sub-predictors has proven to be quite beneficial as it helps to identify the predominant nature within the current block in consideration with its neighborhood, thereby identifying the best-suited predictor that majorly contributes to the final predicted value.

The remainder paper is outlined as follows. Section 2 provides a brief insight into the intra prediction strategy in HEVC. While Section 3 presents the related prediction techniques in the literature, Section 4 elaborates the proposed method that helps to achieve additional intra coding gains over the conventional prediction strategy. The experimental results and analysis are discussed in Section 5, followed by the conclusion in Section 6.

## 2. HEVC intra prediction strategy

The coding tree unit (CTU) is the basic building block in the HEVC standard [12,13]. Each of the CTU is further partitioned as coding units (CU) using the quad-tree partitioning scheme. Each $L \times L$ CU comprises a square luma block, two chroma blocks and the syntax elements associated with the sample block elements. The prediction unit (PU) and transform unit (TU) together constitute the CU. While the PU holds the prediction information of all the color components, a TU contains the residual coding syntax structure for every color component. HEVC makes use of the conventional block-based prediction strategy and signals the prediction mode inside the bitstream for every CU, indicative of whether the CU will be subjected to intra or inter prediction. The intra prediction strategy in HEVC supports 35 prediction modes i.e. planar ($M0$), DC ($M1$) and 33 angular ($M2$-$M34$) modes, to handle the regions with gradual variation in intensities, smooth regions and regions with rich directional textures [14,15].

The planar ($M0$) mode in HEVC is designed to provide good approximations of the patterns with smooth variation in the luminance values. Additionally, it has the potential to overcome the discontinuities along the block boundaries. This is accomplished by means of sample-based averaging of the vertical and horizontal predictions as given in (1), involving a total of four reference samples. Fig. 1 illustrates the same with the previously reconstructed samples {a, b, c, d} representing the reference samples corresponding to the specified sample location of the target pixel $C_{x,y}$ in a block of size $L \times L$.

$$\hat{P}[x,y] = (\hat{P}_h[x,y] + \hat{P}_v[x,y] + L) \gg (log_2(L) + 1) \tag{1}$$

The horizontal prediction $\hat{P}_h[x,y]$ and vertical prediction $\hat{P}_v[x,y]$ in (1) are generated using (2) and (3). In general, $R[x,y]$ represents the reference samples from the blocks above and to the left of the current block.

$$\hat{P}_h[x,y] = [(L-1-x) * R[-1,y] + (x+1) * R[L,-1]] \tag{2}$$

$$\hat{P}_v[x,y] = [(L-1-y) * R[x,-1] + (y+1) * R[-1,L]] \tag{3}$$

The DC ($M1$) mode is meant to handle the smooth regions. The generated prediction value represents the average of the $2L$ reference samples from the previously reconstructed neighboring blocks on the top and to the left of the current block as stated in (4).

$$\hat{P}[x,y] = \frac{1}{2L}\left(\sum_{x=0}^{L-1} R[x,-1] + \sum_{y=0}^{L-1} R[-1,y]\right) \tag{4}$$

The 33 angular modes aim to generate suitable predictions corresponding to the directionally oriented structures. They are broadly categorized as vertical ($M18$-$M34$) and horizontal ($M2$-$M17$) modes. Each of these prediction modes is positioned at spatial increments of the displacement parameter ($d$) i.e. +/-{2, 5, 9, 13, 17, 21, 26, 32} with respect to the vertical and horizontal axis. The amount of displacement between the two vertical/horizontal modes increases as its directional alignment deviates from the vertical/horizontal axis respectively.

While in angular mode, the prediction process essentially requires the initial formulation of a 1-D array of samples from the previously reconstructed coding blocks in the closest proximity. The selection of these reference samples is based on the specified angular direction. Once the reference array is constructed, then the prediction value is generated using pixel interpolation of the two reference samples i.e. $R_i$ and $R_{i+1}$ as given in (5) at a sample position accuracy of (1/32). An illustration of the vertical angular prediction with a positive displacement parameter is elaborated in Fig. 2.

$$\hat{P}[x,y] = (((32 - w_y) * R_i + w_y * R_{i+1}) + 16) \gg 5 \tag{5}$$

The weighted coefficient ($w_y$) is computed as given in (6) and the operator $\gg$ performs the right bit-shift operation. While the variables $x$ and $y$ represent the sample coordinates varying from

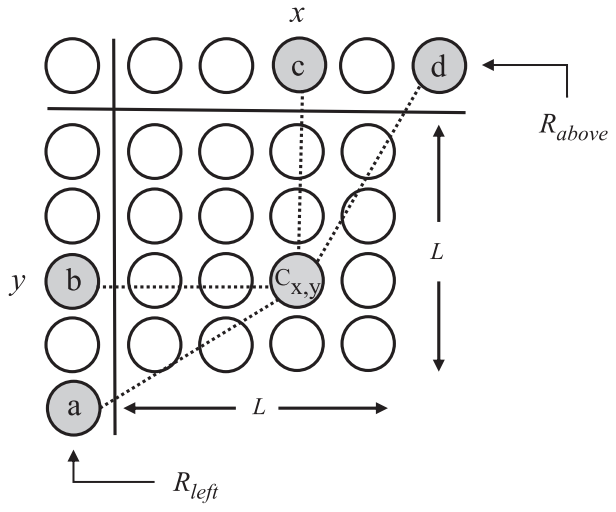


**Fig. 1.** Illustration for planar prediction at the sample location $[x,y]$.

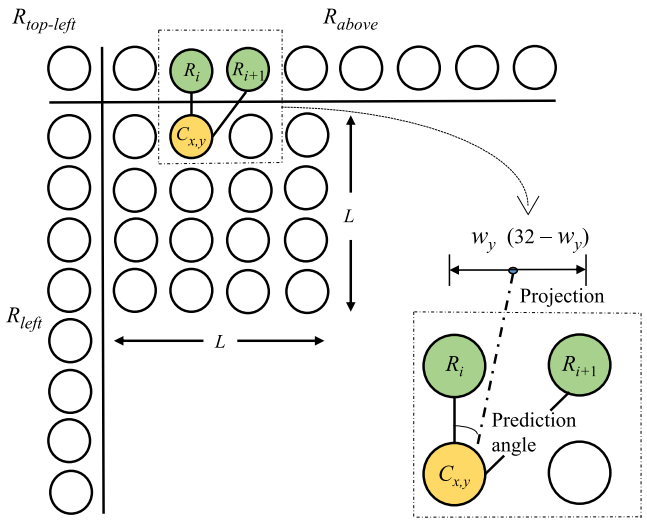


**Fig. 2.** Vertical angular prediction process in HEVC.

$0 \rightarrow (L-1)$, the coefficient $w_y$ is indicative of a fractional value that is pointed to by the projected displacement which lies between the two reference samples $R_i$ and $R_{i+1}$.

$$w_y = ((y+1) \cdot d) \ \& \ 31 \tag{6}$$

The reference sample index $i$ is computed using (7), where the parameter $C_y$ is derived as in (8). As the value of $d$ increases, it is evident that the integer value of $i$ also increases, resulting in shifting of the reference sample pair along the specified direction.

$$i = C_y + x + 1 \tag{7}$$

$$C_y = ((y+1).d) \gg 5 \tag{8}$$

To note that, for the generation of the horizontal prediction, index $y$ needs to be replaced with $x$ and vice versa in all the computations involved in (5)–(8).

## 3. Literature survey

The literature discusses the state-of-the-art lossless image compression strategies in addition to certain video compression techniques. Most of the algorithms aim supremacy in compression with reasonable computational complexity, while few of them relatively work on enhancing the compression efficiency regardless of the complexity. A few of the related prediction techniques are discussed below.

The low-complexity lossless coder for images (LOCO-I) [16] proposed by the Hewlett–Packard uses the median edge detection (MED) predictor. It mainly uses the North ($N$), West ($W$) and North West ($NW$) directional neighbors of the target pixel. The $N$ and $W$ pixels represent the prediction value in case of detection of vertical and horizontal edges respectively. Under any other conditions, the planar predictor ($N + W - NW$) is considered as the prediction value. However, the MED predictor limits the prediction value between $min\{N, W\}$ and $max\{N, W\}$. The context-based, adaptive, lossless image codec (CALIC) presented in [17] outperforms the remaining image compression techniques available in the literature, in terms of compression efficiency with relatively lesser time and space complexities. Robust performance is achieved by suitably modeling the image data. It makes use of the gradient adjusted predictor (GAP), an adaptive non-linear predictor that adapts to the local intensity gradients around the target pixel. The algorithm considers a neighborhood area of seven causal pixels in the $N, W, NW, NN, WW, NE$ and $NNE$ directional orientations. Threshold values are chosen after rigorous experimentation to identify and categorize the edges as strong, normal or weak. Seemann and Tischer in [11] propose a generalized algorithm that performs locally adaptive compound prediction incorporating the blending mechanism of a simple set of linear sub-predictors based on the penalty term. The suitable sub-predictors have been tested for the greyscale images, color images and audio data. The prediction scheme is designed to support noisy as well as smooth data, making use of minimal tunable parameters.

Wige et al. in [9] presented a sample-based weighted prediction (SWP), a low-complexity pixel-by-pixel mechanism that merges the linear predictors with the exponentially decreasing weights derived using the non-local means algorithm. A simple mathematical formulation and look-up table approach are used to derive the optimal integer weights, thereby minimizing the complexity. While the algorithm has proven to render better performance in case of natural video sequences, a noticeable reduction in the prediction efficiency is observed for the screen content sequences owing to the averaging operation performed around the sharp edges. To overcome this drawback, the SWP algorithm has been subtly modified. In the modified algorithm referred to as the directional template matching (DTM) technique, the concept of weighted averaging is replaced with the identification of the pixel that associates a close correlation with the target pixel, to be used as the prediction value. The gradient adaptive sample-based intra prediction (GASP) technique presented in [18] replaces the conventional block-based planar mode in the HEVC framework. The algorithm uses the GAP predictor [17] to identify the abrupt transitions and intends to exploit the sample-based benefits to improvise the prediction accuracy.

The conventional block-based prediction process for the angular modes in HEVC was replaced by the sample-based scheme to enhance the coding accuracy. The sample-based angular prediction (SAP) technique proposed by Zhou et al. [8] primarily depends on the two causal reference sample pair i.e. $R_i$ and $R_{i+1}$, lying in the immediate proximity to the target pixel. On formation of the reference pair, $R_i$ and $R_{i+1}$ are linearly interpolated to generate the prediction as provided in (5). In general, to predict a given row/column of pixels, a maximum of $L + 2$ samples are needed for any of the 33 angular modes. The pixels in the first row/column of the target block are predicted using the row/column in the previously reconstructed block on the top/left to the block under consideration. The reconstructed samples corresponding to the boundary pixels are padded in order to facilitate the $R_{i+1}$ sample for predicting the boundary pixel in the succeeding rows/columns, thereby resulting in the zero-angle prediction. To overcome the shortcoming of zero-angle prediction in SAP, a modified sample-based angular prediction (MSAP) for positive vertical/horizontal modes is proposed in [19]. The modification lies only in the formation of the reference pair $R_i$ and $R_{i+1}$ for the boundary pixels.

The concept of context-based prediction is employed in the improved sample-based angular prediction (ISAP) [20]. It mainly relies on the reference sample $R_i$ being identified as a peak or a valley sample, based on a certain threshold value. When the desired conditions are satisfied, the pixel intensity value corresponding to $R_{i+1}$ is determined as stated in [20]. Once the reference pair is available, linear interpolation of $R_i$ and $R_{i+1}$ is performed as given in (5). In all the remaining cases, the pixels are predicted using the SAP [8] technique. Meanwhile, Kamath et al. present the gradient-oriented directional prediction (GDP) [21], a low-complexity sample-based prediction scheme for angular and planar modes. The algorithm is mainly based on the gradient estimate gathered in the specified directional orientations. The computed gradient value is then used to derive a suitable prediction value on the basis of the neighborhood region around the target pixel as provided in the algorithm specified in [21].

The prediction generation stage plays a crucial role in the minimization of the entropy to enhance compression efficiency. Thus, it is imperative to find a suitable predictor that produces residual values with the entropy as low as possible. Improper selection of the prediction mechanism can result in inducing erroneous data that could risk permanent manipulation of the original content. All these factors serve as a motivation for the work in Section 4.

## 4. Proposed work

In this section, we propose to incorporate a blending strategy into the HEVC anchor that uses a set of sub-predictors to enhance the intra prediction process. It is mainly based on the history of the nearby causal samples. The scheme referred to as the improvised blend of predictors (IBP) for planar and angular prediction is embedded into the planar and angular modes of the HEVC framework. It is designed to locally adapt to the regions with smooth gradients and edges. Prior to the blending process, penalizing factors corresponding to each of the pre-selected sub-predictors are derived to serve as the weighted coefficients for the prediction

generation. A lower value of the penalizing factor signifies that the corresponding sub-predictor contributes more to the prediction process. Therefore, this factor intuitively serves as a local measure to suitably penalize the sub-predictor that results in larger residuals.

A total of six simple sub-predictors are considered in the present work, which shares a dependency on the pixels in the immediate proximity. Each of the sub-predictor depends solely on one or a combination of the four causal neighborhood pixels. The selection of these sub-predictors relies on the correlation between the target pixel and its neighborhood samples. They are meant to generate the prediction value based on their local adaptation. The categorization of these sub-predictors mainly intends to suit the regions that possess discontinuities and smooth gradients. They are grouped as Set 1: $[N, W, NE, NW]$ and Set 2: $[(N + W - NW), (W + NE - N)]$ to identify the patterns with edges and smooth gradients respectively. Hence, Set 1 and Set 2 group of predictors comprising of four ($S = 4$) and two ($S = 2$) sub-predictors are chosen to efficiently predict the angular and planar modes respectively. Here, $N, W, NE$ and $NW$ represent the samples at the specified directional orientation with respect to the target pixel $C_{x,y}$. Once the predictors are assigned based on the prediction mode, then the rest of the computations to calculate the penalizing factors ($G_i$) and correction term are common to both the modes. Set 1 and Set 2 predictors generate four and two penalizing factors corresponding to the angular and planar modes respectively.

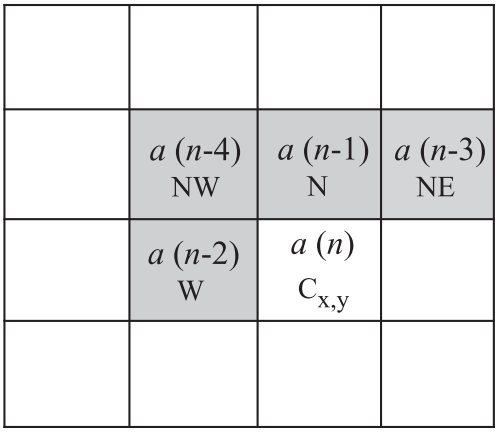


**Fig. 3.** Template for the four causal neighborhood samples $a(n - m)$ along with the directional notation.

For the ease of equation formulation, let $a(n)$ series denote the generalized pixel notation for the target pixel $C_{x,y}$ and its causal neighborhood pixels as shown in Fig. 3. To be precise, here $a(n)$ is the pixel to be predicted and $a(n - m)$ {$m$: 1 to $M$} represents the surrounding four causal pixels shaded in gray. Furthermore, $p_i(n - m)$ {$i$: 1 to $S$} represents the prediction estimate corresponding to the $i^{th}$ sub-predictor for the causal neighbor $a(n - m)$. Fig. 4 elaborates the pixels (shaded in green) that are involved in the derivation of $p_i(n - m)$ corresponding to the six sub-predictors. The penalizing factor $G_i$ for each of the chosen sub-predictors are derived as in (9).

$$G_i = \sum_{m=1}^{M} |a(n-m) - p_i(n-m)| \tag{9}$$

The penalizing factor $G_i$ for the $i^{th}$ sub-predictor is determined at the pixel-level based on the cumulative absolute residuals (CAR) of the causal neighboring samples around the $C_{x,y}$. In an illustration, consider deriving the penalizing factor $G_1$ for the first sub-predictor from the group Set 1 i.e. $N$, while in the angular mode. The computation of $G_1$ includes the absolute differential terms i.e. $(N - NN), (W - NW), (NE - NNE)$ and $(NW - NNW)$ clearly depicted in Fig. 4(a), corresponding to the four neighborhood causal pixels.

Once the $S$ penalizing factors are available, the prediction value $\tilde{P}[x, y]$ is computed as given in (10). Here, the term $P_f$ in the denominator of (10) is the normalization factor that makes the sum of the weighted coefficients as unity.

$$\tilde{P}[x,y] = \frac{1}{P_f} \sum_{i=1}^{S} \frac{p_i(n)}{G_i} \tag{10}$$

$$P_f = \sum_{i=1}^{S} \frac{1}{G_i}$$

The prediction value $\tilde{P}[x, y]$ derived using (10) is the sum of the $i^{th}$ predictor estimate of the target pixel $C_{x,y}$, with the inverse of the

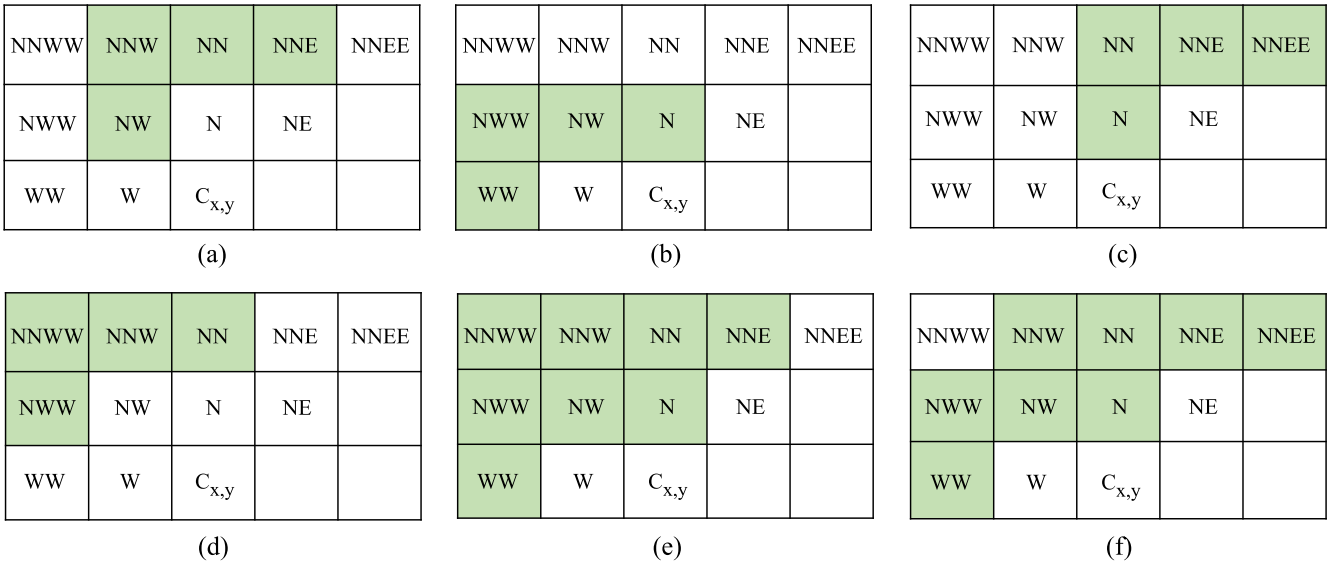


**Fig. 4.** Template for the causal neighborhood samples essential for deriving $p_i(n - m)$ corresponding to the sub-predictors: (a) $N$ (b) $W$ (c) $NE$ (d) $NW$ (e) $N + W - NW$ (f) $W + NE - N$.

relevant penalizing factors ($1/G_i$) serving as the weighted coefficients. This implies that if the sub-predictor predicts well, then the corresponding weighted coefficient will be substantially higher than the remaining. The blending process entirely depends on the value of $G_i$. While all the $G_i$ values are non-zero, the contribution of all the sub-predictors is vital for the prediction generation. In situations when exactly one of the $G_i$ value is zero, the implication is that the predictor estimate $p_i(n)$ corresponding to the sub-predictor is used to predict the neighborhood pixels. Hence, $p_i(n)$ is chosen as the prediction value in such scenarios. Meanwhile, when more than one $G_i$'s are zero, then arbitrarily any of the corresponding sub-predictors are chosen as the prediction value since it does not affect the entropy. The generated prediction value $\tilde{P}[x,y]$ is further fine-tuned to capture the influence of the blending predictor on the causal neighborhood around $C_{x,y}$. This is achieved by determining the correction term $\hat{E}[x,y]$ as given in (11). The term represents the average CAR of the neighborhood causal pixels around $C_{x,y}$. It is then added to $\tilde{P}[x,y]$ for deriving the final prediction value $\hat{P}[x,y]$ as provided in (12). The final predictor $\hat{P}[x,y]$ generated is best-suited to detect and adapt to the dominant characteristics in the local neighborhood region which may include edges or smooth variations in the intensity gradient.

$$\hat{E}[x,y] = \frac{1}{M}\sum_{m=1}^{M}(a(n-m) - p(n-m)) \tag{11}$$

$$\hat{P}[x,y] = \tilde{P}[x,y] + \hat{E}[x,y] \tag{12}$$

At instances when the previously reconstructed samples are unavailable for the prediction i.e. pixels along the block boundary, then the closest available reconstructed pixel is extended to serve as a reference sample for the subsequent encoding process.

The IBP prediction strategy replaces the conventional block-based planar ($M0$) and angular ($M25$) modes of the HEVC anchor. Based on statistical analysis and several set of trials performed on the test video sequences, mode $M25$ has emerged as the least frequently used prediction mode out of the 33 angular modes [22]. Hence, mode $M25$ is reserved for the IBP prediction, thereby resulting in minimal modifications to the HEVC framework. This mode is at an angular displacement of $-2$ from the vertical mode ($M26$) which is aligned along the vertical axis. Meanwhile, the remaining 32 angular modes are predicted using the SAP prediction [8] scheme described in Section 3. In general, the concept of the blending of sub-predictors restricts the amount of contribution by the non-dominant characteristic of the nearby region around the pixel under consideration. This is possible by suitably selecting a predictor with a minimal penalizing factor which implicitly helps to generate an optimum prediction.

## 5. Experimental results, analysis and discussions

The algorithmic validation is carried out on the YUV test dataset [23] that possesses diversely varying textural content ranging between Classes A-F. These sequences contain intense/moderate foreground motion along with the fixed/moderate background motion. The Classes A-E are simply referred to as naturally captured video sequences, while Class F sequences are computer-generated/screen content (SC) video sequences. Additionally, another class of natural sequences commonly referred to as the 4K sequences has been considered for the algorithmic evaluation which includes Beauty, Bosphorus, Honey Bee, Jockey, Ready Steady Go, Shake & Dry and Yacht Ride video sequences. The attributes of the HEVC test sequences possessing a bit-depth of 8 and 4:2:0 chroma sub-sampling are provided in Table 1. The validation is performed on the initial 100 frames of the test sequences, except the Class 4K sequences where only the initial 10 frames are considered, owing to the timing constraints.

**Table 1**
Details of the HEVC test sequences.

| Class | Video Resolution | Category |
|---|---|---|
| A | 2560 × 1600 | 4K × 2K ultra-HD at 30 fps (cropped) |
| B | 1920 × 1080 | 1080p HD at 24, 60 fps |
| C | 832 × 480 | WVGA at 30, 50, 60 fps |
| D | 416 × 240 | WQVGA at 30, 50, 60 fps |
| E | 1280 × 720 | 720p at 60 fps |
| F | 832 × 480 | WVGA at 50 fps |
| | 1024 × 768 | XGA at 30 fps |
| | 1280 × 720 | 720p at 20 fps |
| 4K | 3840 × 2160 | 4K × 2K ultra-HD at 120 fps |

**Table 2**
Simulation environment and test conditions for the lossless mode.

| | |
|---|---|
| CPU | 2.30 GHz Intel (R) Core (TM) i5–6200U processor |
| RAM | 8 GB |
| OS | Windows 10 |
| Profile/Configuration | Main - AI, LB, LP, RA<br>Main_SCC - AI, LB, RA |
| *QP* | 0 |
| *TransquantBypassEnable* | 1 |
| *CUTransquantBypassFlagForce* | 1 |

The HEVC Test Model (HM), a reference software [24] is provided by the JCT-VC to mimic the encoder and decoder. It serves as a common platform to incorporate the algorithms and validate them under the specified test conditions [25,26]. The screen content module (SCM) is essential to handle the SC sequences. The validation is performed using the HM software version HM 16.12 + SCM 8.3. The details on the coding tools included to create the bitstream are mentioned in the low computational complexity test settings i.e. Main and Main_SCC for natural and SC sequences respectively. The Main profile supports four configurations i.e. all intra (AI), low-delay B (LB), low-delay P (LP) and random access (RA), while Main_SCC profile does not extend support to the LP configuration. Thus, validation of Class F sequences is restricted to only three configurations. Now, to invoke the lossless mode in HEVC, the quantization parameter (*QP*) and status flags i.e. *TransquantBypassEnable* and *CUTransquantBypassFlagForce* are modified within the Main profile as specified in Table 2.

Tables 3–6 gives an insight into the savings (%) in bit-rate using the various prediction strategies in comparison with the HEVC anchor for the Main profile. From Tables 3 and 4, it can be observed that the proposed IBP predictor provides an overall percentage savings of 9.50, 2.20, 2.55 and 2.75 in terms of bit-rates for the Main-AI, Main-LB, Main-LP and Main-RA configuration profiles respectively, validated using the sequences from Classes A-E. Additionally, validation using Class 4K sequences provides bit-rate savings of 4.54%, 2.52%, 2.87% and 2.55% for the aforementioned configuration profiles as tabulated in Tables 5 and 6. Meanwhile, Table 7 states that bit-rate savings (%) of 1.29, 0.53 and 0.99 are obtained using the proposed method for the Main_SCC-AI, Main_SCC-LB and Main_SCC-RA configuration profiles respectively, validated using the Class F sequences. From these tables and their corresponding graphical representations in Figs. 5–7, it is evident that the proposed prediction scheme outperforms the HEVC anchor and existing state-of-the-art prediction techniques. The savings in bit-rate is achieved due to the drop in the residual magnitude, thereby resulting in an increase in the number of residuals whose magnitude is zero. Further, it helps to attain an at par latency, as

**Table 3**
Bit-rate savings (%) in the state-of-the-art prediction strategies and proposed IBP predictor for the Main-AI and Main-LB configuration settings using Class A-E sequences.

| Video Class | AI | | | | | | LB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RDPCM[a] | SAP[b] | ISAP[c] | SWP[d] | GDP[e] | IBP[f] | RDPCM | SAP | ISAP | SWP | GDP | IBP |
| A | 9.55 | 10.69 | 10.85 | 8.39 | 11.27 | 13.75 | 2.38 | 2.72 | 2.78 | 1.84 | 3.01 | 4.10 |
| B | 5.35 | 6.80 | 6.90 | 4.85 | 7.17 | 7.79 | 1.40 | 1.90 | 1.94 | 1.02 | 2.01 | 2.16 |
| C | 4.29 | 4.79 | 4.87 | 4.29 | 5.37 | 5.46 | 1.18 | 1.54 | 1.58 | 1.03 | 1.64 | 1.79 |
| D | 7.52 | 7.76 | 7.84 | 5.71 | 8.19 | 8.97 | 1.19 | 1.30 | 1.33 | 0.92 | 1.39 | 1.62 |
| E | 9.24 | 10.03 | 10.11 | 5.18 | 10.45 | 11.55 | 0.96 | 1.20 | 1.22 | 0.26 | 1.25 | 1.35 |
| Average Bit-rate Savings (%) | 7.19 | 8.01 | 8.11 | 5.68 | 8.49 | 9.50 | 1.42 | 1.73 | 1.77 | 1.01 | 1.86 | 2.20 |

Note-
[a] Residual DPCM [10].
[b] Sample-based angular prediction [8].
[c] Improved sample-based angular prediction [20].
[d] Sample-based weighted prediction [9].
[e] Gradient-oriented directional prediction [21].
[f] Improvised blend of predictors.

**Table 4**
Bit-rate savings (%) in the state-of-the-art prediction strategies and proposed IBP predictor for the Main-LP and Main-RA configuration settings using Class A-E sequences.

| Video Class | LP | | | | | | RA | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RDPCM | SAP | ISAP | SWP | GDP | IBP | RDPCM | SAP | ISAP | SWP | GDP | IBP |
| A | 2.82 | 3.20 | 3.28 | 2.10 | 3.53 | 4.68 | 2.77 | 3.15 | 3.23 | 2.20 | 3.47 | 4.60 |
| B | 1.67 | 2.36 | 2.40 | 1.32 | 2.50 | 2.64 | 1.68 | 2.24 | 2.26 | 1.27 | 2.37 | 2.54 |
| C | 1.27 | 1.69 | 1.73 | 1.12 | 1.79 | 1.91 | 1.40 | 1.78 | 1.81 | 1.27 | 1.92 | 2.05 |
| D | 1.27 | 1.42 | 1.46 | 0.97 | 1.48 | 1.70 | 1.71 | 1.82 | 1.85 | 1.31 | 1.94 | 2.21 |
| E | 1.35 | 1.23 | 1.34 | 0.43 | 1.38 | 1.83 | 1.71 | 2.08 | 2.13 | 0.63 | 2.20 | 2.34 |
| Average Bit-rate Savings (%) | 1.68 | 1.98 | 2.04 | 1.19 | 2.14 | 2.55 | 1.85 | 2.21 | 2.26 | 1.34 | 2.38 | 2.75 |

**Table 5**
Bit-rate savings (%) in the state-of-the-art prediction strategies and proposed IBP predictor for the Main-AI and Main-LB configuration settings using Class 4K sequences.

| Video Class | AI | | | | | | LB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RDPCM | SAP | ISAP | SWP | GDP | IBP | RDPCM | SAP | ISAP | SWP | GDP | IBP |
| 4K | 2.62 | 4.26 | 4.32 | 4.22 | 4.42 | 4.54 | 1.25 | 2.39 | 2.42 | 2.47 | 2.46 | 2.52 |

**Table 6**
Bit-rate savings (%) in the state-of-the-art prediction strategies and proposed IBP predictor for the Main-LP and Main-RA configuration settings using Class 4K sequences.

| Video Class | LP | | | | | | RA | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RDPCM | SAP | ISAP | SWP | GDP | IBP | RDPCM | SAP | ISAP | SWP | GDP | IBP |
| 4K | 1.46 | 2.81 | 2.84 | 2.81 | 2.85 | 2.87 | 1.32 | 2.50 | 2.50 | 2.53 | 2.52 | 2.55 |

**Table 7**
Bit-rate savings (%) in the state-of-the-art prediction strategies and proposed IBP predictor for the Main_SCC profile using Class F sequences.

| Video Class | AI | | | | | | LB | | | | | | RA | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RDPCM | SAP | ISAP | SWP | GDP | IBP | RDPCM | SAP | ISAP | SWP | GDP | IBP | RDPCM | SAP | ISAP | SWP | GDP | IBP |
| F | 0.65 | 0.59 | 0.82 | 1.06 | 0.71 | 1.29 | 0.40 | 0.37 | 0.40 | 0.17 | 0.32 | 0.53 | 0.48 | 0.40 | 0.44 | 0.33 | 0.39 | 0.99 |

lesser entropy coding time will be needed to code the fewer number of residuals.

Table 8 highlights the fact that the efficiency of the prediction strategy has resulted in the reduction of the total number of residuals. It indicates that there will be a substantial drop in the prediction error needed to represent a pixel on an average. This has been possible due to the smaller magnitude of the residuals, thus requiring a fewer bits for entropy coding.

Table 9 clearly states that embedding the proposed algorithm in the *M*0 and *M*25 modes of the HEVC intra prediction has favored the most number of pixels to be predicted using the IBP prediction strategy. This implies that there is a prominent increase in the total number of blocks being predicted using the modified *M*0 and *M*25 modes. This is possible due to the selection of the predictor type, targeting to capture the dominant characteristic in the causal neighborhood around the target pixel. The table also reveals that only a small percentage of blocks are coded using the prediction mode *M*25 in the HEVC anchor. The coding gain that is achieved using the IBP prediction, compensates for the subtle losses incurred if any, on the usage of the angular mode *M*25 as a specific prediction mode.

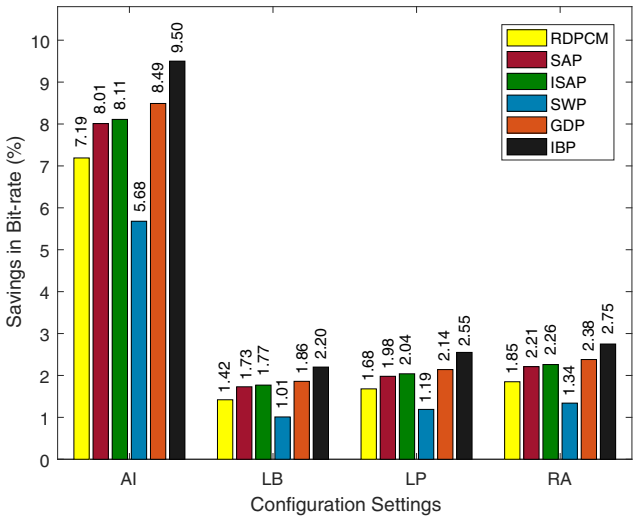


**Fig. 5.** Graphical representation of the bit-rate savings in the state-of-the-art techniques and proposed IBP predictor for the Main profile using Class A-E sequences.

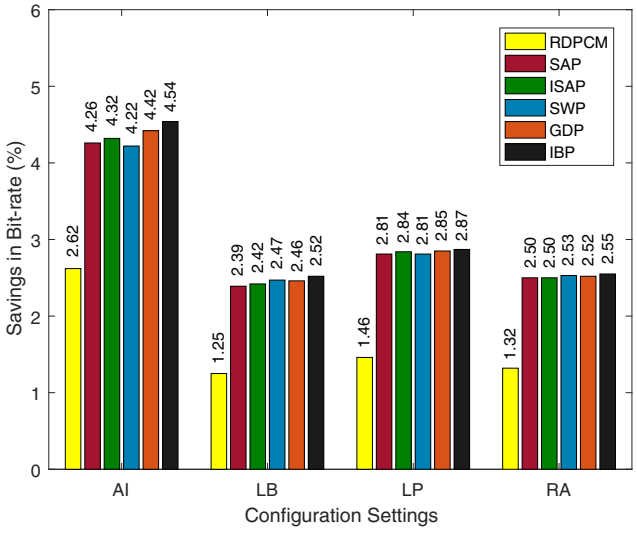


**Fig. 6.** Graphical representation of the bit-rate savings in the state-of-the-art techniques and proposed IBP predictor for the Main profile using Class 4K sequences.

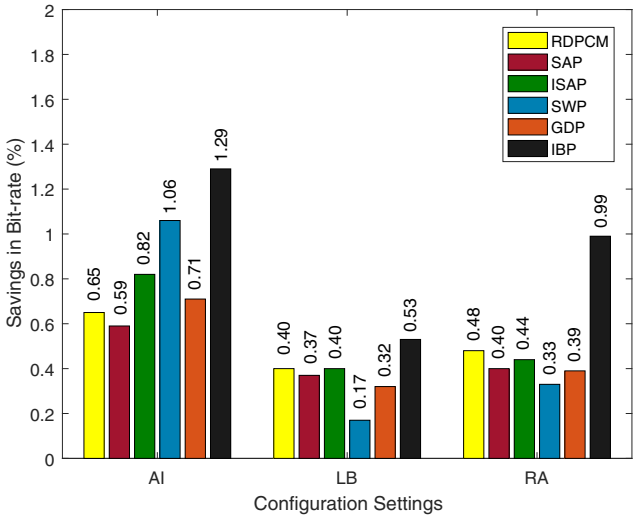


**Fig. 7.** Graphical representation of the bit-rate savings in the state-of-the-art techniques and proposed IBP predictor for the Main_SCC profile using Class F sequences.

**Table 8**
Prediction error analysis in the HEVC anchor and IBP predictor.

| Class | Video Sequence | Sum of absolute error/frame | | Average absolute error/sample | |
|---|---|---|---|---|---|
| | | Anchor | IBP | Anchor | IBP |
| A | Traffic | 13508500 | 11817152 | 3.30 | 2.89 |
| B | Park Scene | 8809124 | 8394252 | 4.25 | 4.05 |
| C | Race Horses | 2170520 | 2155640 | 5.43 | 5.40 |
| D | Race Horses | 605768 | 594960 | 6.07 | 5.96 |
| E | Vidyo1 | 1778940 | 1761016 | 1.93 | 1.91 |
| F | China Speed | 3690580 | 2125264 | 4.69 | 2.70 |
| 4K | Yacht Ride | 16588744 | 16132384 | 2.00 | 1.94 |

**Table 9**
Intra prediction mode distribution (%) in the HEVC anchor and proposed IBP predictor.

| Class | Video Sequence | Anchor | | | | IBP | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Planar (M0) | DC (M1) | Angular (M2-M34) | | Planar (M0) | DC (M1) | Angular (M2-M34) | |
| | | | | M25 | Others | | | M25 | Others |
| A | Traffic | 12.8 | 6.1 | 1.7 | 79.4 | 19.2 | 1.5 | 18.8 | 60.5 |
| | People on Street | 12.1 | 7.1 | 2.1 | 78.7 | 26.8 | 2.4 | 28.8 | 42.0 |
| B | Kimono | 21.0 | 8.7 | 1.4 | 68.9 | 12.1 | 9.2 | 11.6 | 67.1 |
| | Park Scene | 15.7 | 11.1 | 1.8 | 71.4 | 16.5 | 11.1 | 18.0 | 54.4 |
| | BQ Terrace | 8.6 | 6.5 | 15.3 | 69.6 | 7.2 | 8.8 | 5.0 | 79.0 |
| C | Race Horses | 15.8 | 7.9 | 2.0 | 74.3 | 17.9 | 3.9 | 12.5 | 65.7 |
| | BQ Mall | 11.9 | 7.8 | 5.9 | 74.4 | 5.1 | 12.3 | 3.7 | 78.9 |
| | Party Scene | 9.7 | 6.7 | 2.9 | 80.7 | 5.8 | 7.0 | 4.1 | 83.1 |
| | Basketball Drill | 7.3 | 5.6 | 0.8 | 86.3 | 1.8 | 2.3 | 1.6 | 94.3 |
| D | Blowing Bubbles | 11.3 | 5.8 | 4.4 | 78.5 | 3.8 | 2.8 | 3.4 | 90.0 |
| | Race Horses | 10.8 | 6.1 | 2.2 | 80.9 | 12.0 | 2.5 | 18.4 | 67.1 |
| | BQ Square | 7.4 | 5.9 | 2.8 | 83.9 | 2.5 | 6.8 | 1.9 | 88.8 |
| | Basketball Pass | 8.4 | 4.2 | 2.0 | 85.4 | 14.6 | 1.3 | 4.2 | 79.9 |
| E | Vidyo1 | 15.2 | 8.6 | 5.3 | 70.9 | 16.6 | 1.3 | 7.6 | 74.5 |
| | Vidyo3 | 14.2 | 10.9 | 4.1 | 70.8 | 17.2 | 2.3 | 8.7 | 71.8 |
| | Vidyo4 | 16.0 | 9.4 | 3.6 | 71.0 | 21.2 | 1.9 | 10.9 | 66.0 |
| | Johnny | 15.2 | 13.8 | 4.1 | 66.9 | 12.8 | 5.5 | 3.7 | 78.0 |
| F | Basketball Drill Text | 7.4 | 4.7 | 0.5 | 87.4 | 4.3 | 3.4 | 1.5 | 90.8 |
| | China Speed | 8.7 | 2.4 | 0.6 | 88.3 | 5.1 | 3.1 | 1.8 | 90.0 |
| | Slide Show | 8.5 | 3.0 | 0.8 | 87.7 | 19.5 | 2.5 | 7.1 | 70.9 |
| 4K | Beauty | 12.8 | 9.0 | 3.4 | 74.8 | 0.7 | 19.1 | 1.4 | 78.8 |
| | Bosphorus | 17.1 | 8.1 | 2.3 | 72.5 | 3.8 | 14.0 | 3.5 | 78.7 |
| | Honey Bee | 20.3 | 8.3 | 3.1 | 68.3 | 1.0 | 12.4 | 1.8 | 84.8 |
| | Jockey | 20.2 | 9.7 | 3.1 | 67.0 | 1.1 | 15.5 | 1.3 | 82.1 |
| | Ready Set Go | 12.7 | 6.5 | 7.3 | 73.5 | 5.0 | 12.4 | 9.0 | 73.6 |
| | Shake & Dry | 19.6 | 6.3 | 2.1 | 72.0 | 2.0 | 4.9 | 6.0 | 87.1 |
| | Yacht Ride | 17.7 | 8.8 | 2.3 | 71.2 | 4.1 | 17.0 | 1.7 | 77.2 |

Table 10 elaborates the distribution of the coded blocks for the various PU sizes. From the table, it is evident that in case of HEVC anchor, more number of blocks are coded using PU size of $4 \times 4$ followed by $8 \times 8, 16 \times 16$ and $32 \times 32$. The incorporation of the IBP prediction technique has resulted in a decrease in the number of blocks coded using $4 \times 4$ and a subsequent rise in the coded blocks using $32 \times 32$. Fig. 8 illustrates the distribution of the pixels for the various PU sizes in the Park Scene (Class B) video sequence. The illustration clearly states that more number of pixels

**Table 10**
Comparison of the total number of blocks coded using the various PU block sizes in the HEVC anchor and proposed IBP predictor.

| Class | Video Sequence | Method | Block Count | | | | |
|---|---|---|---|---|---|---|---|
| | | | 4 × 4 | 8 × 8 | 16 × 16 | 32 × 32 | Total |
| A | Traffic | Anchor | 233240 | 5458 | 54 | 1 | 238753 |
| | | IBP | 93732 | 19823 | 3066 | 530 | 117151 |
| B | Park Scene | Anchor | 99244 | 5165 | 402 | 51 | 104862 |
| | | IBP | 39596 | 3657 | 2239 | 618 | 46110 |
| C | Race Horses | Anchor | 23148 | 453 | 0 | 0 | 23601 |
| | | IBP | 10276 | 1779 | 325 | 37 | 12417 |
| D | Race Horses | Anchor | 6040 | 50 | 0 | 0 | 6090 |
| | | IBP | 2772 | 451 | 52 | 13 | 3288 |
| E | Vidyo1 | Anchor | 51200 | 1440 | 36 | 1 | 52677 |
| | | IBP | 20296 | 2870 | 942 | 168 | 24276 |
| 4K | Yacht Ride | Anchor | 415552 | 23756 | 457 | 8 | 439773 |
| | | IBP | 176168 | 55702 | 5464 | 500 | 237834 |

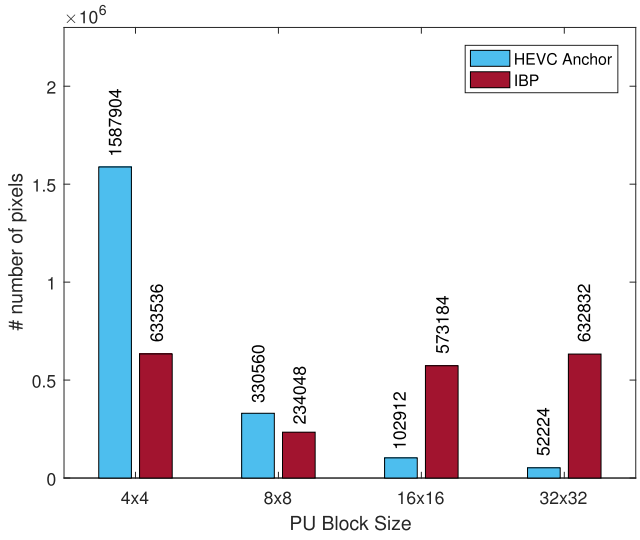


**Fig. 8.** Graphical representation of the pixel distribution for the various PU sizes in the Park Scene sequence.

**Table 11**
Run-time savings (%) in the proposed IBP predictor.

| Class | AI | | LB | | LP | | RA | |
|---|---|---|---|---|---|---|---|---|
| | ET | DT | ET | DT | ET | DT | ET | DT |
| A | 1.12 | −0.08 | 0.45 | 0.12 | 0.54 | 0.06 | 0.20 | 0.25 |
| B | 0.25 | 0.32 | 0.28 | −0.06 | 0.38 | 0.12 | 0.16 | −0.10 |
| C | 1.15 | 0.16 | 0.66 | 0.24 | 1.02 | 0.15 | 0.24 | 0.18 |
| D | 1.24 | 0.14 | 0.56 | 0.20 | 0.82 | 0.10 | 1.08 | 0.15 |
| E | 0.54 | 0.10 | 0.45 | 0.24 | 0.60 | 0.14 | 0.15 | 0.32 |
| F | 0.82 | −0.20 | 0.85 | 0.26 | – | – | 0.18 | 0.30 |
| 4K | 0.96 | 0.22 | 0.72 | 0.18 | 0.56 | 0.09 | 0.14 | 0.12 |

$((L \times L) \times BlockCount)$ are coded using larger PU sizes. Additionally, there is a reduction in the total number of coded blocks as compared to the HEVC anchor, which helps to achieve savings in the run-time. To note that, the analysis in the Tables 8–10 are performed on the first intra frame of the test video sequences.

The run-time analysis is presented in Table 11. Less complex computations and algorithmic simplicity have resulted in an at par latency at the encoder and decoder abbreviated as ET and DT respectively, in comparison with the HEVC anchor. Savings in bit-rate is achieved without any increase in run-time.

## 6. Conclusion

In this paper, we have presented an efficient pixel-wise prediction scheme to code the HEVC test video sequences in a lossless manner. It is achieved by suitably modifying the planar and angular modes of the HEVC anchor to embed the simple combination of sub-predictors that adapts to the causal neighborhood with smooth gradients and edges. Optimal prediction is achieved using this heuristic history-based blend of predictors over the conventional block-based planar and angular prediction strategies. The incorporated mechanism helps to select the best prediction mode for a given block based on the suitable regional adaptation. Prominent gains are achieved in terms of percentage savings in bit-rate, without any increase in run-time for the lossless coding of the natural and screen content videos. The comparative experimental results and the various analysis conducted reveal that the proposed IBP predictor mode outperforms the HEVC anchor as well as the state-of-the-art prediction techniques.

## Declaration of Competing Interest

The authors declared that there is no conflict of interest.

## Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.aeue.2019.153000.

## References

[1] Sullivan GJ, Ohm JR, Han WJ, Wiegand T. Overview of the high efficiency video coding (HEVC) standard. IEEE Trans Circ Syst Video Technol 2012;22(12):1648–67.
[2] Wiegand T, Sullivan GJ, Bjontegaard G, Luthra A. Overview of the H.264/AVC video coding standard. IEEE Trans Circ Syst Video Technol 2003;13(7):560–76.
[3] Pastuszak G, Trochimiuk M. Algorithm and architecture design of the motion estimation for the H.265/HEVC 4K-UHD encoder. J Real-Time Image Proc 2016;12(2):517–29.
[4] Zhang Y, Lu C. Efficient algorithm adaptations and fully-parallel hardware architecture of H.265/HEVC intra encoder. IEEE Trans Circ Syst Video Technol 2018.
[5] Zhang Y, Lu C. High-Performance Algorithm Adaptations and Hardware Architecture for HEVC Intra Encoders. IEEE Trans Circ Syst Video Technol 2019;29(7):2138–45.
[6] Sayood K. Lossless compression handbook. Academic press; 2002.
[7] Xu J, Toshi R, Cohen RA. Overview of the emerging HEVC screen content coding extension. IEEE Trans Circ Syst Video Technol 2015;26(1):50–62.
[8] Zhou M, Gao W, Jiang M, Yu H. HEVC lossless coding and improvements. IEEE Trans Circ Syst Video Technol 2012;22(12):1839–43.
[9] Wige E, Yammine G, Amon P, Hutter A, Kaup A. Sample-based weighted prediction with directional template matching for HEVC Lossless Coding. In: Proceedings of picture coding symposium PCS. p. 305–8.
[10] Lee S, Kim IK, Kim C. RCE2: Test 1 Residual DPCM for HEVC lossless coding. JCTVC-M0079, Incheon, Korea; 2013.
[11] Seemann T, Tisher P. Generalized locally adaptive DPCM. Department of Computer Science Technical Report CS97/301; 1997. p. 1–15.
[12] Sze V, Budagavi M, Sullivan GJ. High Efficiency Video Coding (HEVC): algorithms and architectures. Switzerland: Springer; 2014.
[13] Wein M. High efficiency video coding: coding tools and specification. Berlin: Springer; 2015.
[14] Lainema J, Bossen F, Han WJ, Min J, Ugur K. Intra coding of the HEVC standard. IEEE Trans Circ Syst Video Technol 2012;22(12):1792–801.
[15] Pastuszak G, Abramowski A. Algorithm and architecture design of the H.265/HEVC intra encoder. IEEE Trans Circ Syst Video Technol 2015;26(1):210–22.
[16] Weinberger MJ, Seroussi G, Sapiro G. LOCO-I: A low complexity, context-based, lossless image compression algorithm. In: Proceedings of data compression conference. p. 140–9.
[17] Wu X, Memon N. CALIC-a context based adaptive lossless image codec. In: Proceedings of IEEE international conference on acoustics, speech, and signal processing, vol. 4 ; 1996, p. 1890–3.

[18] Antony A, Ganapathy S. Performance enhancement of HEVC lossless mode using sample-based angular and planar predictions. SIViP 2017;11 (6):1057–64.
[19] Antony A, Ganapathy S. Highly efficient near lossless video compression using selective intra prediction for HEVC lossless mode. AEU-Int J Electron Commun 2015;69:1650–8.
[20] Antony A, Ganapathy S. HEVC-based lossless intra coding for efficient still image compression. Multimedia Tools Appl 2017;76(2):1639–58.
[21] Kamath SS, Aparna P, Antony A. Gradient-oriented directional predictor for HEVC planar and angular intra prediction modes to enhance lossless compression. AEU-Int J Electron Commun 2018;95:73–81.
[22] Guarda AF, Santos JM, da Silva Cruz LA, Assuncao PA, Rodrigues NM, de Faria SM. A method to improve HEVC lossless coding of volumetric medical images. Signal Process: Image Commun 2017;59:96–104.
[23] Test Sequences. Available:ftp://ftp.tnt.uni-hannover.de; 2012.
[24] HEVC Reference Software [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svnHEVCSoftware/tags/HM-16.12+SCM-8.3/; 2016.
[25] Bossen F. Common Test Conditions and Software Reference Configurations. JCTVC-H1100, San Jose, CA; 2012.
[26] Haoping Y, Robert C, Krishna R, Jizheng X. Common test conditions for screen content coding. JCTVC-U1015-r2, Warsaw; 2015.