Frequency Domain Intra-Prediction Analysis and Processing for High Quality Video Coding

Saverio G. Blasi, *Student Member, IEEE,* Marta Mrak, *Senior Member, IEEE,*
and Ebroul Izquierdo, *Senior Member, IEEE*

***Abstract*—Most of the advances in video coding technology focus on applications which require low bitrates, for example for content distribution on a mass scale. For these applications the performance of conventional coding methods are typically sufficient. Such schemes inevitably introduce large losses to the signal, which are unacceptable for numerous other professional applications such as capture, production and archiving. To boost the performance of video codecs for high quality content, better techniques are needed especially in the context of the prediction module. An analysis of conventional intra-prediction methods used in the state-of-the-art High Efficiency Video Coding (HEVC) standard is reported in this paper, in terms of the prediction performance of such methods in the frequency domain. Appropriately modified encoder and decoder schemes are presented and used for this study. The analysis shows that conventional intra-prediction methods can be improved especially for high frequency components of the signal which are typically difficult to predict. A novel approach to improve the efficiency of high quality video coding is also presented in the paper based on such analysis. The modified encoder scheme allows for an additional stage of processing performed on the transformed prediction to replace selected frequency components of the signal with specifically defined synthetic content. The content is introduced in the signal by means of feature-dependant look-up tables. The approach is shown to achieve consistent gains against conventional HEVC with up to -5.2% coding gains in terms of bitrate savings.**

***Index Terms*—Video compression, predictive coding, frequency estimation, HEVC**

I. Introduction

Most of the video compression standards available today, including the recently ratified state-of-the-art H.265/HEVC (High Efficiency Video Coding, referred to in this paper as HEVC) [1], follow a block-based scheme involving three successive stages. The current picture in the sequence is first partitioned into blocks of a given size which are sequentially processed by the encoder. Each block is input to a prediction module, which attempts to remove temporal and spatial redundancies present in the sequence to obtain a compressed signal using previously coded content. The residual signal is then input to a transform module, which attempts to further reduce spatial redundancies using a more suitable representation and successively quantising the data. Finally the resulting signal is input to an entropy coding unit, which exploits statistical redundancy to represent the signal in a compact form by means of short binary codes.

The prediction module of a video encoder provides the prediction signal for a given block. The way this signal is computed depends on the current coding conditions (such as the temporal order of the current frame in the sequence or the coding configuration), and it is in general based on rate-distortion (RD) decisions. Typically two schemes can be used at this purpose: inter-prediction makes use of previously encoded frames to compute a prediction for the current block, based on the assumption that the content of these frames may be similar to the current frame; intra-prediction makes use of content extracted from the same frame as the currently encoded block. While typically inter-prediction provides higher compression efficiency, intra-prediction is useful in case of high spatial correlation within the current picture, and it is necessary in case the current frame is the only available information (e.g. while encoding the first frame in the sequence or in the case of still image coding, or when a decoder refresh is required).

Video coding standards are mostly designed for efficient usage for a mass scale distribution and use such prediction schemes to deliver very high compression of medium to low quality content. Most of the efforts in the video coding community are dedicated to improving the efficiency of video codecs at these levels of quality. In fact under these conditions the HEVC standard is reportedly achieving more than 50% bitrate savings while preserving the same visual quality of its predecessor H.264/AVC (Advanced Video Coding, referred to in this paper as AVC) [2] [3]. Interestingly, HEVC is also considerably more efficient than JPEG2000 when coding still images (on average 44% higher efficiency in terms of bitrate savings at the same objective quality) [4].

While such levels of quality are acceptable for some purposes, there are many applications in which higher levels of quality are necessary. In these cases, it is even more important that the decoded video is as visually similar to the original as possible. Typical examples of such kind of applications can be found in medical imaging applications, in the transmission of signals from cameras throughout the production chain, in real-time screenshot sharing or in screen mirroring systems (when the content on the screen of a device is mirrored in real-time to a different screen). Moreover with the increasing demand for high-definition televisions capable of displaying content at very high framerates and high bitdepths, the quality of the delivered videos is becoming an extremely important issue even in the context of consumer applications. Users expect

Saverio G. Blasi and Ebroul Izquierdo are with the School of Electronic Engineering and Computer Science, Queen Mary, University of London, Mile End Road, E1 4NS, London, (UK) email: s.blasi@qmul.ac.uk, e.izquierdo@qmul.ac.uk

Marta Mrak is with the British Broadcasting Corporation, Research and Development Department, 56 Wood Lane, W12 7SB, London, (UK) email: marta.mrak@bbc.co.uk

video content at as good quality as possible, with the lowest visible coding distortion.

Under these constraints it is difficult to predict the fine granularity details of the signal needed to preserve such levels of quality. Consequently, even the most advanced compression schemes are less efficient and provide high bitrates. As a result the efficiency of HEVC decreases becoming closer to that of its predecessor AVC, as it was recently shown via experimental validation [5]. Similarly when coding still images at such levels of quality, HEVC results in less improvements compared with JPEG2000 [4]. Conventional prediction methods rely on spatial interpolation, which typically provides a soft prediction signal. Such signals might not be optimal for high quality coding as they do not always deliver the high frequency content.

In order to improve video and still image coding under these high quality constraints, an analysis of conventional intra-prediction methods is presented in this paper, focused on evaluating the impact of each intra-prediction mode on the prediction accuracy of different frequency components of the signal. The analysis shows that intra-prediction methods typically provide less accurate prediction of high frequency components of the signal in many cases, and also highlights the different behaviours of each mode in terms of prediction accuracy in the frequency domain. Based on this analysis, a novel approach to improve the efficiency of high quality video coding is also presented in the paper. In particular an additional stage of processing is performed on the transformed prediction signal prior to the residual computation. The processing is performed by means of appropriately defined masking patterns and look-up tables, to possibly improve the high-frequency content in the prediction signal introducing synthetic components with the goal of reducing the bits needed to encode the residual coefficients. The analysis and proposed method are implemented in this paper in the context of the intra-prediction schemes used in HEVC.

The rest of this paper is organised as follows. Some background on state-of-the-art intra-prediction methods and transform methods for video compression is presented in Section II, mainly focusing on techniques proposed and used in the context of HEVC. The modified encoder and decoder schemes with direct transformation of the predictors are illustrated in Section III, followed by an analysis of conventional intra-prediction methods in the frequency domain. In Section IV the proposed method to improve coding efficiency under high quality constraints is presented, based on prediction processing in the frequency domain. Finally, results of the approach are shown in Section V and some conclusions are presented in Section VI.

II. Background

Intra-prediction, sometimes referred to as predictive image coding, consists of computing a prediction for the current block using a number of pixels (referred to as reference samples) extracted from the same frame. To ensure that the process can be repeated at the decoder side, only content that has already been coded can be used for this purpose. Typically the highest redundancy appears among neighbouring pixels, and for this reason only pixels in the surrounding of the currently encoded block are used as reference samples. In HEVC a block of $N \times N$ luma samples is predicted by means of up to $4N+1$ reference samples located immediately at the top and on the left of the current block, in the regions denoted as A, B, C, D and E in Figure 1 (a).

The standard allows up to 35 intra-prediction modes [6], each labelled by an index from 0 to 34. Among these modes, DC prediction (labelled as mode 1) simply consists of predicting the samples in the prediction block using a single value obtained by averaging all available reference samples. Due to the fact that the signal is subsequently transformed to the frequency domain, and given the nature of such transformed signals, typically the largest coefficient can be expected at the zero-frequency (DC) component. DC prediction attempts at predicting this coefficient limiting its impact on the related bitrates.

A technique was proposed already in the context of the AVC standard and is also used in HEVC, referred to as planar prediction (labelled as mode 0 in the standard). The idea is that of finding a plane (namely a polynomial surface of order 1) that optimally fits the available reference samples, and using integer approximations of values extracted from such plane as prediction. Refer to the reference samples in A in Figure 1 (a) as $s_A(i)$, and in D as $s_D(i)$ with $i = 0, ..., N-1$. Denote with $s_A = s_A(N-1)$ and $s_D = s_D(N-1)$. For each sample $p(i,j)$ two linear interpolations are first computed as:

$$p_A(i,j) = (N-j)s_A(i) + (j)s_D$$

and:

$$p_D(i,j) = (N-i)s_D(j) + (i)s_A$$

Finally the predicted sample $p(i,j)$ is obtained as the average of the two linear interpolations:

$$p(i,j) = \frac{p_A(i,j) + p_D(i,j)}{2}$$

approximated to the nearest integer.

Finally HEVC makes use of another class of intra-prediction methods based on the idea that visual content often follows a direction of propagation. Reference samples can be projected inside the prediction block according to such direction, possibly returning a good approximation of the original content. Up to 33 angular directions are considered for the luma component as illustrated in Figure 1 (b). Modes labelled from 2 to 17 are referred to as horizontal directions, and modes labelled from 18 to 34 are referred to as vertical directions. The reference samples are arranged in a row or column reference array whose elements $s(t)$ depend on the angular direction (more details on this process can be found in the literature [6]). Denote the samples in the currently predicted block as $p(i,j)$ where $i,j = 0, ..., N-1$. Each sample is predicted as the weighted interpolation of two reference samples as in:

$$p(i,j) = \frac{w_j}{32}s(t) + \frac{(32-w_j)}{32}s(t+1) \tag{1}$$

The weighting factor is computed as $w_j = |jd| \% [32]$, with $\% [\bullet]$ being the modulo operator and d being a parameter allowed to assume a fixed number of possible values depending on the direction. For instance, in the case of horizontal directions, these span from $d = +32$ (for mode 2) to $d = -26$ (for mode 17). Values of $w_j = 0$ correspond to exactly vertical, horizontal or diagonal modes in which samples are simply copied throughout the block (namely no weighted interpolation is involved), marked with a solid line in Figure 1 (b). The index t is computed as:

$$t = i + c$$

where:

$$c = \left\lfloor \frac{jd}{32} \right\rfloor$$

and $\lfloor \bullet \rfloor$ corresponds to rounding to the nearest integer smaller than its argument.

Due to the fact that a relatively large number of samples is predicted using a small amount of information strongly localized on a particular area in the frame, the aforementioned intra-prediction methods might still introduce unwanted prediction artefacts and in general might not provide sufficiently accurate predictions, as will also be shown in the rest of this paper. In the case of angular prediction this is mostly evident when using modes with a strong directionality (e.g. exactly vertical or horizontal). Particularly in large blocks, original samples located in the block edges distant from the reference samples might not be accurately predicted, returning considerably high residuals in such locations. An attempt to reduce the related bitrates might result in blocking artefacts. To limit these effects HEVC makes use of a smoothing filter which interpolates reference samples prior to intra-prediction with the goal of more uniformly distributing the residual error among the samples in the block. The filter is selectively applied only in particular intra-prediction modes and block sizes. The effect of the smoothing filter used in HEVC is illustrated in the example in Figure 2. Average absolute values of the residual samples obtained in the case of 16×16 blocks predicted using mode 12 in a test sequence are presented when the smoothing filter is enabled in Figure 2 (a), and when it is disabled in Figure 2 (b). In the second case clearly the residual sample magnitude tends to increase towards the edges of the block, while a more uniform distribution of the residual magnitude is obtained when smoothing is enabled.

It is worth noting here that intra-prediction methods can still be considerably improved, depending on the kind of content and targeted application. For example a method [7] was proposed to perform intra-prediction on non-square block partitioning, implemented in the context of the HEVC standard for medium to low quality applications. Similarly, combined intra-prediction [8] can be used to improve prediction exploiting spatial redundancies within the block.

The intra-predicted samples are subtracted to the original samples to obtain a residual signal. This is then input to the transform module with the main goal of finding a representation more suitable for the purpose of data compression. A well known and successful way of obtaining such representation consists in using the discrete cosine transform (DCT), a member of a particular family of sinusoidal unitary transforms derived from discrete Fourier analysis. Different types of DCT have been proposed, where the two-dimensional versions of types referred to as II and III [9] are typically used in image and video compression applications for the forward and inverse transform respectively.

Due to the fact that entries in the DCT base matrix are irrational numbers, rounding is necessary before they can be stored in a digital representation. To limit the effects of such approximation, these entries are also scaled to reduce rounding errors. The transform base matrix used in HEVC was derived following this process approximating to the nearest integer DCT coefficients appropriately scaled [10]. Notice that while HEVC allows the transform to be applied to blocks of different sizes (referred to as transform units, TUs [11]) ranging in size from 32×32 to 4×4 samples, to limit the resources needed while coding a single transform base matrix Q_{32} is defined for transforming the largest 32×32 TUs. Transform base matrices for smaller blocks are simply obtained by downsampling Q_{32}. For instance the matrix used for 8×8 TUs is:

$$Q_8 = \begin{bmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{bmatrix}$$

The DCT has many desirable characteristics, but it might not be the optimal transform to decorrelate the residual signal in some cases [12]. Consider for instance the case of a block

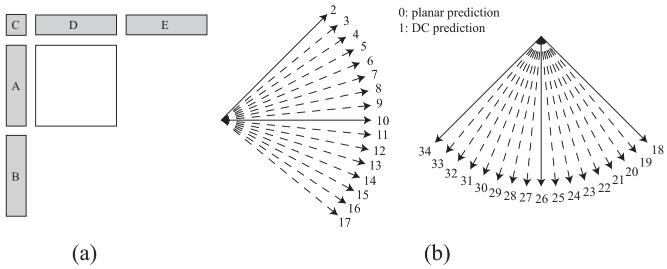


Fig. 1: Intra-prediction reference samples (a) and available modes (b) in HEVC.

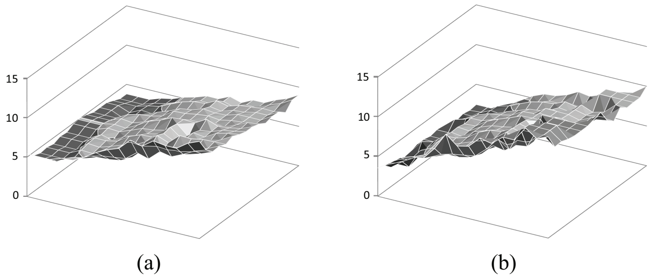


Fig. 2: Example of average per-sample absolute residual magnitude. Smoothing is enabled in (a) and disabled in (b).

of samples obtained with horizontal angular intra-prediction. Samples towards the left of the block (closer to the reference samples used for the prediction) are likely to be predicted more accurately than samples closer to the right side of the block. Consequently the residual magnitudes can be expected to increase with the distance of a sample from the left boundary of the block. Conversely, DCT basis functions behave in an opposite way: for instance the function corresponding to first frequency component decreases monotonically. In these cases a better representation may be obtained using a transform whose basis functions are more correlated with the behaviour of the signal.

The discrete sine transform (DST) was originally proposed at this purpose, to be used in HEVC on all intra-predicted blocks. Later, a study on the compression performance provided by different transforms in the case of angular intra prediction was presented [13], showing that while more efficient compression is obtained using DST in intra-predicted blocks, the benefits of DST against DCT in large blocks are generally limited and do not counterbalance the disadvantages of its generically higher computational complexity and lack of fast algorithms (such as partial butterfly). For this reason in the first version of HEVC, DST is only used on small 4×4 TUs of luma samples [14].

The transform is generally followed by quantisation. Each coefficient is quantised to a given step (depending on a parameter usually referred to as the quantisation parameter, QP). The higher the QP, the coarser is the quantisation. In the case of HEVC, QP is allowed to assume values between 0 and 52. When QP is set to 0 no quantisation is performed (transform is also skipped in this case as it would not bring any benefits) and the resulting encoder performs lossless coding which means that the reconstructed decoded signal is mathematically identical to the original signal. High values of the QP result instead in a degraded decoded signal. The common test conditions [15] of the standard define four QP values (22, 27, 32 and 37) to be used to measure the compression performance of the encoder at medium to low levels of quality. Conversely, high quality image and video coding requires a moderate quantisation. QP values of 2, 5, 7 and 12 were used in this paper.

III. Analysis of Intra-Prediction Methods in Frequency Domain

An analysis of conventional intra-prediction methods is presented in this section, with the goal of evaluating the performance of each mode on predicting different frequency components of the original signal. In order to allow the analysis, modified encoder and decoder schemes making use of direct transformation of the prediction blocks are first introduced at the beginning of the section as the essential base for the methodology presented in this paper.

A. Direct Transformation of Prediction Blocks

Consider that a certain $N \times N$ square block of samples X is being encoded. Consider also that an equally sized block of samples P is being considered as a prediction for X, obtained from one of the possible intra-prediction modes. Denote as Q the $N \times N$ transform base matrix.

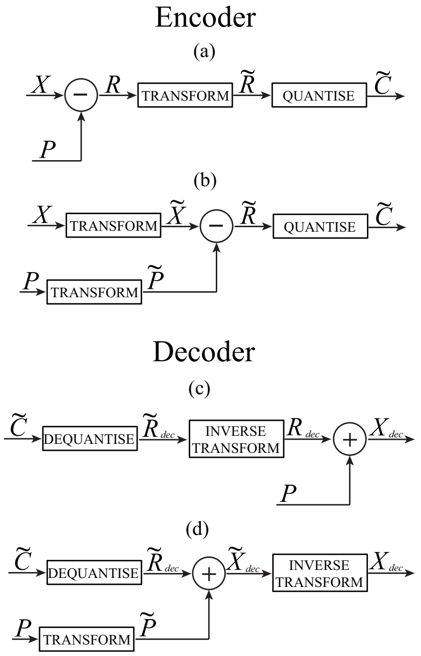


Fig. 3: Conventional encoder (a) and decoder (c) schemes compared with the proposed encoder (b) and decoder (d) schemes with direct transformation of prediction blocks.

In conventional video codecs the residual samples R are computed in the spatial domain from the samples in X and P as $R = X - P$. The transformed residual block is then obtained as:

$$\widetilde{R} = QRQ^{\top}$$

The transformed samples are successively quantised to obtain the coefficients $\widetilde{C}$. These steps are illustrated in the scheme in Figure 3 (a). At the decoder side the coefficients are extracted from the bitstream, dequantised (i.e. rescaled), and inverse transformed. Due to the fact that the quantisation is a non-reversible operation, the block R_{dec} is different than the block of residuals R. R_{dec} is added to P in the spatial domain to obtain the reconstructed block X_{dec}, as in the scheme in Figure 3 (c).

In this paper different encoder and decoder schemes are considered as follows. The prediction and original signals are directly transformed to the frequency domain, to obtain respectively $\widetilde{P}$ and $\widetilde{X}$, as illustrated in Figure 3 (c). These are used to obtain the residuals $\widetilde{R}$, which are then quantised as in conventional coding. At the decoder side the prediction block is first transformed to the frequency domain to obtain $\widetilde{P}$, and the coefficients $\widetilde{C}$ are dequantised to obtain $\widetilde{R}_{dec}$. These are used to obtain the reconstructed block $\widetilde{X}_{dec}$ in the frequency domain, which is finally inverse transformed to return X_{dec}, as in the scheme in Figure 3 (d).

Note that similar schemes have already been proposed in video coding, though they have been applied with different purposes and in different modules of encoder and decoder. A method was presented [16] in which motion compensated prediction and original signals are separately transformed. In such method the inter-predicted samples are transformed and scaled using pre-computed weights, before calculating the residuals directly in the frequency domain. The weights are fixed on a sequence basis, and are transmitted in the bitstream to be used at the decoder side. The method was further extended [17] to include a recursive calculation of the weights avoiding the need for additional side information.

If the same X and P are used as input to the two schemes in Figure 3 (a) and (b) exactly the same residual $\widetilde{R}$ should be obtained in the frequency domain. The linearity of the transform is easily shown as follows:

$$\begin{aligned}\widetilde{R} = \widetilde{X} - \widetilde{P} &= \left(QXQ^\top - QPQ^\top\right)\\ &= Q\left(X - P\right)Q^\top = QRQ^\top\end{aligned}$$

In practice due to the truncation of the variables during the transform stages, the residual signal obtained by means of the proposed encoder scheme is different from the signal obtained using conventional schemes. It is worth clarifying how such truncations affect both schemes in Figure 3 (a) and (b). A 16-bit representation of variables between and after the transform stages is supported in HEVC. To meet these requirements, the output of each transform stage has to be carefully scaled. Consider as an example that a block of 4×4 residual samples is being transformed using a DCT as in the scheme in Figure 3 (a), and assume an input 8-bit data representation. The dynamic range of the residual samples goes from -255 to $+255$ requiring 9 bits (to account for the sign). The first stage of the transform consists of multiplying such a block to the right by the transpose of the 4×4 DCT base matrix Q_4. The L1 norm of the transpose of Q_4 is $(64 \times 4 = 256)$, therefore the dynamic range of the variables after the first stage of the transform goes from $-(256 \times 255)$ to $+(256 \times 255)$. This range would require 17 bits to be exactly represented. In order to keep variables within 16-bit representation, such variables must be scaled by a factor of 2 (i.e. 1-bit binary right shift). Extending this concept to blocks of arbitrary size $N \times N$ and input variables of arbitrary bitdepth B (with $B \geq 8$), the variables after the first stage of the transform must be shifted to the right by a number of bits equal to $s = \log_2(N) - 1 + (B - 8)$.

Consider now using the scheme in Figure 3 (b). In this case instead of transforming the block of residual samples, the 4×4 original and prediction blocks are directly transformed. The representation of the dynamic range of the samples input to the first stage of the transform remains 8 bits (e.g. from 0 to $+255$). Consequently in theory there is no need for scaling the output variables in this case. In practice this is difficult to implement due to the fact that different frequency components after the first stage of the transform have different dynamic ranges, and for this reason the same adjustments used in conventional HEVC are used in this paper when considering the proposed scheme in Figure 3 (b). These are shown in Table I in the case of DCT for 8-bit input data representation, for the two stages of transform.

Some tests were performed to quantify the effects of using the modified encoder scheme compared with conventional HEVC. While the analysis and methods are mainly presented in the context of encoding of video sequences, they only directly affect intra-prediction and as such they can be also tested on still images. In particular the Kodak test set [18] was used at this purpose, comprising 24 768×512 images.

The compression performance was measured in terms of BD-rate [19], a well known metric which computes the average bitrate difference relative to an anchor in percentage, where conventional HEVC was used as the anchor. Tests were performed under high quality constraints using QP values equal to 2, 5, 7 and 12 respectively. Results of these tests are reported in Table II where negative values correspond to an improvement with respect to the anchor. In average, a negligible 0.1% BD-rate difference was obtained between the two codecs, with minimum and maximum BD-rates of respectively -0.18% and $+0.31\%$.

Note also that using the schemes in Figure 3 (b) and (d) implies that the transform operation is performed twice for each block (both at encoder and decoder side), instead of only once as in the schemes in (a) and (c). This additional transform clearly adds some computational complexity to the encoding and decoding. The computational complexity of the proposed encoder and decoder were compared with the complexity of conventional HEVC encoder and decoder in terms of additional coding time, in percentage. In average 6.7% and 2.8% increases in encoding and decoding time were reported respectively. Using such schemes has negligible effects on the coding efficiency and acceptable impacts in terms of complexity, while it provides the essential base for the analysis and proposed method presented in the rest of this paper.

B. *Per-coefficient intra-prediction correlation*

In general by providing a more accurate prediction of the current block, a better encoder performance can be expected (due to the smaller residual samples which require less bits to be coded). While common distortion metrics in the spatial domain such as the sum of absolute differences (SAD) or sum of squared differences (SSD) can be used to estimate the accuracy of a prediction, these types of metrics fail in measuring the impact of each intra-prediction method on different frequency components of the signals. It is instead reasonable to expect particular effects of certain prediction modes on specific frequency components. These effects can be captured and analysed to formulate appropriate processing methods to improve the coding efficiency.

To perform such analysis, the modified encoder scheme in Figure 3 (b) was implemented in the context of HEVC intra-prediction and a few sequences were encoded to collect test

TABLE I: Data ranges and adjustments during HEVC forward transforms with 8-bit input/output.

TU Size	Max. input	Input Bits	L1 norm	Max. output	Output bits	Binary Shift
First DCT stage						
4×4	$+255$	9	256	$+65280$	17	$\gg 1$
8×8	$+255$	9	512	$+130560$	18	$\gg 2$
16×16	$+255$	9	1024	$+261120$	19	$\gg 3$
32×32	$+255$	9	2048	$+522240$	20	$\gg 4$
Second DCT stage						
4×4	$+32767$	16	256	$+8388352$	24	$\gg 8$
8×8	$+32767$	16	512	$+16776704$	25	$\gg 9$
16×16	$+32767$	16	1024	$+33553408$	26	$\gg 10$
32×32	$+32767$	16	2048	$+67106816$	27	$\gg 11$

TABLE II: Comparison of proposed encoder and decoder schemes and conventional HEVC.

Image	BD-rates (%)
Stone building	0.14
Red door	-0.02
Hats	0.03
Girl in red	0.14
Motocross bikes	0.27
Sailboat	0.07
Window	0.16
Market place	0.24
Spinnakers	-0.18
Sailboat race	0.10
Pier	-0.17
Couple on beach	0.26
Mountain stream	0.14
Water rafters	0.13
Girl	0.20
Tropical key	-0.05
Monument	-0.10
Model in black	0.05
Lighthouse	0.31
Mustang	0.15
Portland headlight	0.11
Barn and pond	0.11
Parrots	0.07
Chalet	0.13

data. Coding was performed under high quality constraints (namely the QP was set to 5). All pairs of transformed original and prediction blocks computed during the encoding were collected, grouped in terms of the transform size and intra-prediction mode used.

Given a certain transform size and intra-prediction mode and considering all corresponding pairs available in the test data, a measure of the performance of the prediction at different frequency components can be obtained by studying the similarity between the two samples co-located in the transformed prediction and original blocks respectively. A well known method for computing such similarity consists of computing the per-coefficient correlation between the time series of prediction coefficients and corresponding original coefficients at each specific location in the blocks.

Assume that in total $K_{N,mode}$ pairs of transformed original and prediction blocks of a certain size $N \times N$ using a certain intra-prediction mode $mode$ are available in the test data. For simplicity in the following $K_{N,mode}$ is denoted as K. Refer to each transformed original or prediction block as $\widetilde{X}_i$ or $\widetilde{P}_i$ respectively, where $i = 0, 1, ..., K-1$. Finally denote as $\widetilde{x}_i(m,n)$ and $\widetilde{p}_i(m,n)$ the samples at location (m,n) in the block $\widetilde{X}_i$ and $\widetilde{P}_i$ respectively. The correlation between the arrays $[\widetilde{x}_0(m,n), \widetilde{x}_1(m,n), ..., \widetilde{x}_{K-1}(m,n)]$ and $[\widetilde{p}_0(m,n), \widetilde{p}_1(m,n), ..., \widetilde{p}_{K-1}(m,n)]$ for a given transform size $N \times N$ and a certain intra-prediction mode $mode$ can be defined as:

$$R_{\{N,mode\}}(m,n) = \\ = \frac{1}{K}\sum_{i=0}^{K-1} \frac{[\widetilde{p}_i(m,n) - E\{\widetilde{p}(m,n)\}]\,[\widetilde{x}_i(m,n) - E\{\widetilde{x}(m,n)\}]}{\sigma_{\widetilde{p}(m,n)}\sigma_{\widetilde{x}(m,n)}}$$

where the expected values $E\{\bullet\}$ and standard deviations σ are estimated from the samples. Values of $R_{\{N,mode\}}(m,n)$ close to $+1$ indicate that the intra-prediction mode $mode$ is good at predicting the coefficient in (m,n) when the TU size is $N \times N$. Values of the correlation close to zero indicate instead that the predicted samples in (m,n) carry almost no information on the original samples when using a specific intra-prediction mode on TUs of a specific size.

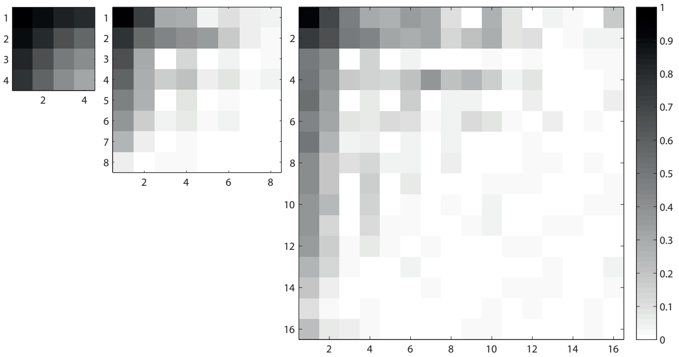


Fig. 4: Per-sample correlation between original and prediction samples for blocks of different sizes using planar mode.

The correlation values for TUs of different sizes (4×4, 8×8 and 16×16) are illustrated in Figure 4 for the planar mode. Clearly the transform size has an evident impact on the correlation values especially at higher frequencies (i.e. towards the bottom-right corner of the blocks). Relatively high correlation values are reported in 4×4 blocks at all locations (minimum correlation of 0.35). This can be taken as an indication that the planar mode performs relatively well at predicting any frequency components of the signal in the case of the 4×4 transform size. Conversely very low values of the correlation are obtained at high frequency components in the case of larger transform sizes. In particular in the case of the 16×16 transform size, all correlation values are smaller than 0.3 for locations with $m \geq 4$ and $n \geq 4$. This can be taken as an indication that the prediction performance of the planar mode at such high frequency components is relatively poor.

Similarly the correlation values are strongly influenced by the type of intra-prediction being used. This is particularly evident for angular intra-prediction modes, and in fact the prediction direction has a direct impact on the prediction performance at different frequency components as shown in Figure 5. Correlation values for three different angular directions (namely mode 7, mode 10 and mode 26) are illustrated, in the case of a fixed 8×8 transform size. In the case of mode 7, namely a horizontal mode as in Figure 1 (b), high values of the correlation are obtained in the left half of the block. Low correlation values are reported elsewhere in the block. Interestingly in the case of mode 10 (pure horizontal prediction) very high correlations are reported in the left region of the block concentrated in the first few columns of the block. Mode 26 (pure vertical prediction) results in a similar behaviour in the vertical direction.

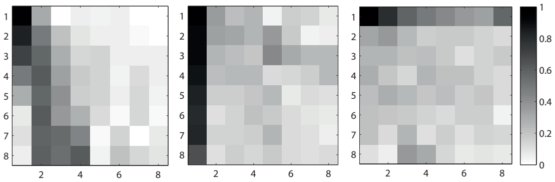
A report of the results of the per-coefficient correlation analysis is presented here for selected frequency components. In particular, refer to locations at the top-left, bottom-left and bottom-right corners in the block as (e), (r) and (s) respectively (the labels used in the rest of this paper to identify particular locations in the blocks are illustrated in Figure 8). The correlation values at these three locations is reported in

TABLE III: Correlation values at the top-right corner (e), bottom-left corner (r) and bottom-right corner (s).

	4×4			8×8		
mode	(e)	(r)	(s)	(e)	(r)	(s)
0	0.90	0.86	0.53	0.17	0.46	-0.01
1	0.89	0.85	0.56	0.12	0.27	0.20
2	0.91	0.71	0.15	0.14	0.01	0.14
3	0.92	0.66	0.31	0.10	0.04	0.09
4	0.92	0.67	0.31	0.03	0.10	0.01
5	0.91	0.72	0.45	0.05	0.07	0.16
6	0.93	0.69	0.55	0.06	0.13	-0.01
7	0.94	0.56	0.54	0.02	-0.10	0.00
8	0.94	0.72	0.61	0.00	-0.10	0.04
9	0.96	0.83	0.65	0.00	0.72	0.00
10	0.96	0.90	0.73	0.12	0.87	0.20
11	0.96	0.86	0.67	0.00	0.79	0.02
12	0.94	0.78	0.62	-0.01	0.10	0.04
13	0.94	0.61	0.52	0.03	-0.13	0.02
14	0.94	0.80	0.52	0.05	0.20	-0.04
15	0.92	0.84	0.34	0.04	0.11	0.11
16	0.92	0.86	0.02	0.12	0.10	-0.12
17	0.92	0.87	0.26	0.18	0.12	-0.02
18	0.91	0.89	0.24	0.21	0.16	0.24
19	0.89	0.89	0.31	0.29	0.08	0.22
20	0.88	0.90	0.03	0.29	0.08	-0.07
21	0.84	0.89	0.35	0.29	0.08	0.17
22	0.83	0.90	0.52	0.32	0.04	-0.03
23	0.77	0.90	0.46	0.16	0.00	0.05
24	0.79	0.90	0.55	0.28	0.01	0.03
25	0.78	0.90	0.55	0.32	0.02	0.01
26	0.80	0.91	0.63	0.55	0.36	0.15
27	0.75	0.91	0.57	0.29	0.00	0.03
28	0.76	0.91	0.55	0.17	-0.02	0.05
29	0.74	0.91	0.49	0.18	0.02	-0.01
30	0.79	0.91	0.56	0.26	0.06	-0.04
31	0.80	0.90	0.48	0.26	0.06	0.23
32	0.79	0.88	0.27	0.23	0.05	-0.02
33	0.78	0.89	0.35	0.22	0.07	0.18
34	0.78	0.88	0.22	0.14	0.06	0.16

Table III for all HEVC intra-prediction modes, for transform sizes of 4×4 and 8×8 samples. The values of the correlation at these corner locations can be taken as an indication of the performance of the accuracy of each intra-prediction mode when predicting selected frequency components of the signal.

A first conclusion can be immediately highlighted from these results: the size of the blocks has an evident impact on the performance of intra-prediction. Much higher correlation values are obtained in the case of 4×4 TUs than in 8×8 blocks. Note that the fact that intra-prediction works better on smaller blocks is a well-known behaviour, which can be easily explained considering that intra-prediction techniques make use of a few samples close to the top-left boundary of the block to predict all samples within the block. In smaller blocks such reference samples are obviously closer to the locations in which they are used for prediction, therefore it can be expected that they are more correlated with the original content of such locations.



Mode 7 Mode 10 Mode 26

Fig. 5: Per-sample cross-correlation for 8×8 blocks predicted using different intra-prediction modes.

Interestingly, the analysis presented in this paper does not only confirm this behaviour but also highlights that these effects have a direct impact on different frequency components. For instance, the sample at the highest frequency (labelled as (s) in Table III) is still predicted relatively well in almost all cases when coding 4×4 TUs, with an average correlation of 0.44. Conversely, average 0.06 correlation was obtained in the same location in 8×8 TUs.

Another important conclusion can be obtained by analysing such results. Considering only vertical angular modes (from 18 to 34) and referring to results obtained in 8×8 blocks, average correlation of 0.26 was obtained for the top-right sample labelled as (e) in Table III. Conversely, average correlation of 0.05 was obtained in the same location for horizontal angular modes (from 2 to 17). An opposite behaviour is reported in the case of the bottom-left sample labelled as (r) in Table III: average correlation of 0.06 is obtained for vertical modes, whereas an average value of 0.19 is obtained for horizontal modes. These results confirm that the directionality of intra-prediction has predictable effects on the prediction accuracy at different frequency components in the blocks.

Following from these observations, it is clear that conventional intra-prediction methods may not be sufficiently accurate in predicting some frequency components of the original signal (depending on the intra-prediction mode being used), and as a result high bitrates can be expected particularly when targeting high quality video coding. These effects are evident in large blocks, but instead are very limited in case of 4×4 TUs. Note that in HEVC these are the only blocks that are transformed using DST instead of DCT. Such transform is noticeably more computational complex than DCT (mostly due to the lack of fast algorithms such as partial butterfly). Using the proposed schemes implies that the transform and inverse transform operations need to be performed twice on each block with respect to conventional schemes, which means that enabling the approach while using DST would have a considerable impact on computational complexity. Due to these effects and also considering the relatively already good performance of conventional methods when coding these small blocks, the approach illustrated in the rest of this paper is only enabled on TUs larger or equal than 8×8 samples, and therefore it is only studied in the context of the DCT transform. Conversely, 4×4 TUs are coded as in conventional HEVC.

IV. Frequency Domain Prediction Processing

Average correlation of 0.06 as found for instance in 8×8 TUs at location (s) in Table III means that intra-prediction modes under high quality constraints provide a signal whose highest frequency is almost completely uncorrelated with the same component in the original signal. The residual sample at this location is consequently likely to assume a high value. In high quality coding this value cannot be discarded by quantisation but needs to be transmitted in the bitstream.

In order to limit these effects and possibly improve compression efficiency, a different approach is proposed in this paper to deliver the high frequency components of the original signal. The first step of such approach consists of selectively discarding frequency components of the prediction signal which are almost completely uncorrelated with the original signal, under the assumption that these components provide no benefits to the encoding. The second step consists then in replacing these discarded components with more informative content capable of limiting the impact of residual samples at these frequencies, possibly reducing the related bitrates. The encoder and decoder schemes can be further modified to include such additional frequency-domain prediction processing, as illustrated in Figure 6. Each of these two steps is detailed in the rest of this section.

A. Discarding Coefficients using Patterns

The strongly localised distribution of correlation values obtained when using particular intra-prediction modes (as illustrated in Figure 5) can be exploited to selectively discard coefficients in the transformed prediction block. For instance in the case of mode 7 in the figure, clearly relatively high correlations were obtained in samples in the left half portion of the block, whereas very low correlation was obtained in almost all samples located in the other half of the block. Similar behaviours were obtained for other modes highlighting the fact that correlation values are generally distributed in a predictable manner depending on the coding conditions.

The process of selecting coefficients in the transformed prediction block in order to follow these behaviours can be easily formalised through the definition of a set of masking matrices, referred to as patterns in the rest of this paper. A pattern is a matrix H of a given size $N \times N$, whose elements $h(m,n)$ are binary elements (namely either 1 or 0). The value of an element in a certain location determines whether the corresponding coefficient in the transformed prediction block is preserved or it is discarded and replaced. Although more complex options are possible, only four classes of patterns are considered in this work. Formally consider an integer parameter L, referred to as pattern size, where $0 \leq L \leq N$. Three values of L were considered, $L = N/4$, $L = N/2$ and $L = 3N/4$. Then:

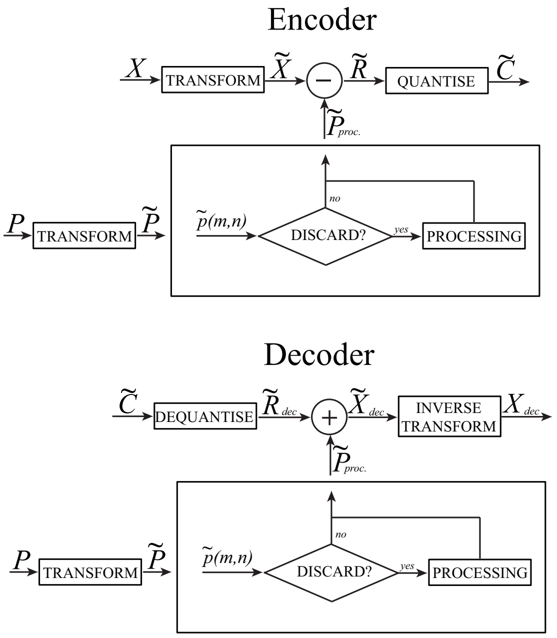


Fig. 6: Proposed encoder and decoder schemes including processing of the transformed prediction blocks.

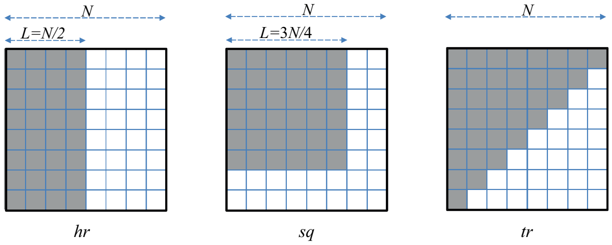


Fig. 7: Example of patterns used for frequency domain prediction processing. Coefficients in shaded locations are preserved, coefficients in white locations are discarded.

- Vertical rectangular patterns, referred to as H_{vr}, consist of L consecutive rows of preserved coefficients in the top-side portion of the block, or:

$$h_{(vr,L)}(m,n) = \begin{cases} 1 & \text{if } n \leq L; \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

- Horizontal rectangular patterns, referred to as H_{hr}, consist of L consecutive columns of preserved coefficients in the left-side portion of the block ,or:

$$h_{(hr,L)}(m,n) = \begin{cases} 1 & \text{if } m \leq L; \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

- Square patterns, referred to as H_{sq}, consist of $L \times L$ preserved coefficients in the top-left portion of the block, or:

$$h_{(sq,L)}(m,n) = \begin{cases} 1 & \text{if } m \leq L \text{ and } n \leq L; \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

- Triangular patterns, referred to as H_{tr}, consist of a triangular region of preserved coefficients in the top-left portion of the pattern, or:

$$h_{(tr,L)}(m,n) = \begin{cases} 1 & \text{if } (m+n) \leq N; \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

A certain pattern H is applied to a transformed prediction block $\widetilde{P}$ by Hadamard (entrywise) product. Coefficients $\widetilde{p}(m,n)$ that are discarded can be either left to zero or replaced with other values by means of appropriate methods, as illustrated later in this section. Some example patterns are shown in Figure 7.

The schemes in Figure 6 were implemented in HEVC. In order to identify which patterns should be used depending on features of the current block and prediction block, first experiments were performed under the condition that the processing in the schemes in the figure is performed by simply setting the discarded masked coefficients to zero. Denoting as $\widetilde{p}_{proc}(m,n)$ the elements of $\widetilde{P}_{proc}$, this corresponds to:

$$\widetilde{p}_{proc}(m,n) = 0 \text{ if } h(m,n) = 0$$

The following algorithm, referred to as Algorithm 1, was then implemented at the encoder side. A list of all considered patters H_1,H_2, ...,H_M is considered, where M is the number of available patterns; an additional element H_0 was included at the first position in the list to identify the trivial pattern, where $h_0(m,n) = 1$ for $m = 0, ...N-1$, $n = 0, ...N-1$, namely this is the case when no coefficients are discarded in the prediction block. After a block of samples is intra-predicted using a given mode, prediction and original signals are independently transformed obtaining $\widetilde{P}$ and $\widetilde{X}$ respectively. An index j is initialised to zero and:

1) The pattern H_j is extracted from the list and applied to $\widetilde{P}$ to obtain $\widetilde{P}_{proc}$. The residual samples are computed as $\widetilde{R} = \widetilde{X} - \widetilde{P}_{proc}$ and quantised to obtain $\widetilde{C}$. This is dequantised and inverse transformed to obtain $\widetilde{R}_{dec}$, which is finally used to compute the reconstruction $\widetilde{X}_{rec} = \widetilde{R}_{dec} + \widetilde{P}_{proc}$.
2) $\widetilde{C}$ and $\widetilde{X}_{rec}$ are used to compute the RD cost relative to the current element j. A temporary solution is considered as the index j^o such that pattern H_{j^o} corresponds to minimum RD cost.
3) If $j < M$, the index j is incremented and the algorithm goes back to step 1. Otherwise the pattern at minimum RD cost is output, identified by its optimal index j^o.

The index j^o to select the correct pattern in the list is signalled to the decoder in the bitstream for each block in which the algorithm is enabled. At the decoder side, such index is decoded and used to extract H_{j^o}. This is then applied to the transformed prediction as in the scheme at the bottom of Figure 6.

The approach was tested again in the Kodak image test set. In Table IV, the most frequently selected pattern is shown for each HEVC intra-prediction mode and for each TU size in which the algorithm is enabled. In case the most frequently selected pattern is the trivial pattern H_0, the second most frequently selected pattern is reported.

Clearly, the patterns are chosen according to the directionality of the intra-prediction mode used in the blocks. Horizontal patterns are most likely selected in horizontal modes, and vertical patterns are most likely selected in vertical modes. The triangular pattern H_{tr} is chosen relatively rarely apart from the case of the planar prediction (mode 0).

TABLE IV: Patterns at minimum distortion according to transform size and intra-prediction mode.

mode	8×8	16×16	32×32
0	H_{tr}	H_{tr}	H_{tr}
1	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{sq,N/4}$
2	$H_{sq,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
3	$H_{sq,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
4	$H_{sq,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
5	$H_{hr,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
6	$H_{hr,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
7	$H_{hr,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
8	$H_{hr,N/2}$	$H_{hr,N/4}$	$H_{hr,N/4}$
9	$H_{hr,N/2}$	$H_{hr,N/2}$	$H_{hr,N/4}$
10	$H_{hr,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
11	$H_{hr,N/4}$	H_{tr}	$H_{hr,N/4}$
12	$H_{hr,N/2}$	$H_{hr,N/4}$	$H_{hr,N/4}$
13	$H_{hr,N/2}$	$H_{hr,N/4}$	$H_{hr,N/4}$
14	$H_{hr,N/2}$	$H_{hr,N/4}$	$H_{hr,N/4}$
15	$H_{sq,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
16	$H_{sq,N/4}$	$H_{hr,N/4}$	$H_{hr,N/4}$
17	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{hr,N/4}$
18	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{sq,N/4}$
19	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{sq,N/4}$
20	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{sq,N/4}$
21	$H_{sq,N/4}$	$H_{vr,N/4}$	$H_{vr,N/4}$
22	$H_{vr,N/4}$	$H_{vr,N/4}$	$H_{vr,N/4}$
23	$H_{vr,N/4}$	$H_{vr,N/4}$	$H_{vr,N/4}$
24	$H_{vr,N/2}$	$H_{vr,N/4}$	$H_{vr,N/4}$
25	$H_{vr,N/2}$	$H_{vr,N/4}$	$H_{vr,N/4}$
26	$H_{vr,N/4}$	$H_{vr,N/4}$	$H_{vr,N/4}$
27	$H_{vr,N/4}$	$H_{sq,N/4}$	H_{tr}
28	$H_{vr,N/2}$	$H_{vr,N/2}$	$H_{vr,N/4}$
29	$H_{vr,N/2}$	$H_{vr,N/2}$	$H_{vr,N/4}$
30	$H_{vr,N/2}$	$H_{vr,N/2}$	$H_{vr,N/4}$
31	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{sq,N/4}$
32	$H_{sq,N/4}$	$H_{vr,N/4}$	$H_{vr,N/4}$
33	$H_{sq,N/4}$	$H_{vr,N/4}$	$H_{sq,N/4}$
34	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{hr,N/4}$

B. Replacing Coefficients with Look-up Tables

While the analysis in Section III is helpful in determining which frequency components of the prediction signal should be preserved and which may instead be discarded, it gives no information regarding the real content of the original blocks at these frequency components. It is instead reasonable to assume that such content is correlated with encoder decisions on the currently encoded block.

Consider for instance that HEVC is used to encode some test content (the Kodak image set was used again in this example) using the scheme in Figure 3 (b). Consider also that the transformed original blocks $\widetilde{X}$ are collected while encoding, classified depending on the optimal intra-prediction mode and TU size selected by the encoder. The histograms in Figure 9 show then the frequency of occurrence of coefficient values extracted at 15 locations in the block, in the case of 8×8 TUs that are intra-predicted with mode 9. The locations are marked following the labels in Figure 8.

It is reasonable to expect that the content in blocks that are well predicted by the almost horizontal mode 9 presents a strong directionality. In fact such directionality reflects in larger coefficients toward the left-most portion in the blocks and conversely smaller coefficients in the right-most portion in the blocks, as evident from the histograms in Figure 9.

Consider now that the same content is encoded using the schemes in Figure 6. Assuming that such block is processed using an horizontal pattern (which is the most frequently selected option according to Table IV), prediction coefficients toward the right-most portion in the block would be discarded.

(a)		(b)		(c)		(d)	(e)
(f)		(g)		(h)		(i)	
(j)		(k)		(l)		(m)	
(n)		(o)		(p)		(q)	
(r)							(s)

Fig. 8: Sample location labels.

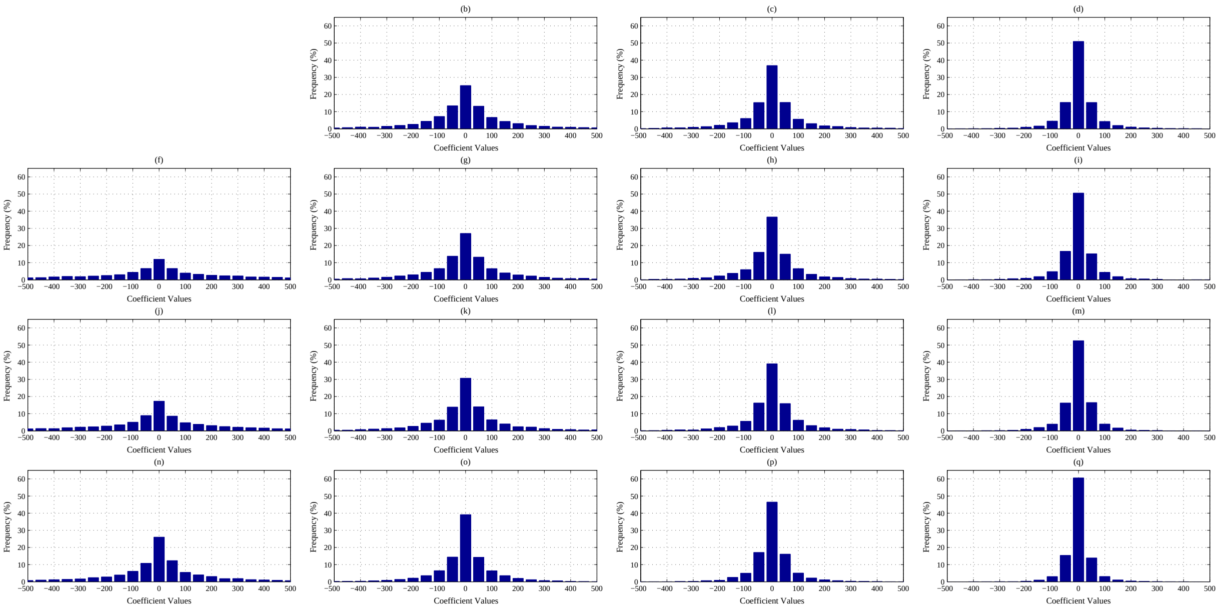


Fig. 9: Frequency of occurrence of coefficient values at different locations in the transformed blocks, for intra-prediction mode 9, block size 8×8.

Unless the encoder can provide a prediction of such coefficients in a different way, the transformed original samples in this portion of the block would directly go in the residual signal. This behaviour clearly is not optimal in a large number of cases, as highlighted by the histograms in Figure 9: for instance, while 52% of coefficients are valued between -25 and 25 in location (m), still around 35% of coefficients in this location result in an absolute value between 25 and 75, and the remaining 13% result in an absolute value even larger than 75. Attempting to code such large values with conventional methods would provide very high bitrates and inefficient coding.

A method is proposed in this paper to solve this issue, based on the assumption that completely new content can be inserted in the prediction signal within the processing block in the schemes in Figure 6, specifically with the goal of reducing residual samples providing synthetic high frequency components. Such synthetic content can be defined studying histograms as those in Figure 9, obtained for different intra-prediction modes and TU sizes. The values in such histograms which appear with high relative frequency can be tested as possible replacements for the discarded coefficients in the transformed prediction signal after the application of a given pattern. In order to reduce as much as possible the overhead required to signal the parameters necessary to apply the approach, and also to limit the complexity needed to perform the prediction processing, in this paper all discarded coefficients in a block are replaced with the same synthetic value. The problem is then that of formalising and optimising the process of defining and using these synthetic values.

At this purpose, a dictionary can be defined by considering a set of T different values $\alpha_0, ..., \alpha_{T-1}$. These values are selected to be representative of the range spanned by the actual coefficients at high frequencies. A trade-off between frequency of occurrence of coefficients and their effects on the coding efficiency should be considered. While large coefficients tend to appear less often, they also have a higher impact on the related bitrates when they are not accurately predicted: in these cases very high residual samples are obtained, which are inefficiently compressed by conventional methods. For this reason it makes sense to include in the dictionary many small values, but also some sparse large values to deal with the cases when they might be needed. A total of 33 elements in the dictionary was considered in the implementation used in this paper, where $\alpha_0 = 0$, and values span from -128 to $+128$.

In order to correctly select and use a certain element from the dictionary, this must be signalled in the bitstream so that the same element can be extracted and applied also at the decoder side as in the scheme at the bottom of Figure 6. Unfortunately, high bitrates may result as a consequence of this signalling especially if the number T of elements in the dictionary is large. For this reason, following again from the assumption that the frequency of occurrence of coefficient values is dependent on encoder decisions on the currently encoded block, it makes sense to restrict and adapt the number of allowed elements in the dictionary to these features. Instead of considering all possible dictionary elements in each block, subsets of such dictionary can be used in the form of look-up tables.

In particular for each TU the feature set $\Omega = \{N, H, mode\}$ is considered, where N is the TU height and width, H is the currently used pattern and $mode$ is the intra-prediction mode being used. For each instance of Ω, a look-up table F_Ω is defined as an indexed array $f_\Omega(k)$ where $k = 0, ..., K-1$; the K elements in each table are a particular sub-set of the elements $\alpha_0, ..., \alpha_{T-1}$ in the dictionary. The length of the look-up tables K can be set to allow the testing of a sufficient number of coefficient values in each block, while at the same time limiting the rates needed for transmitting the corresponding index. A value $K = 8$ was used in the implementation described in this paper.

The K elements to form each look-up table and their order can be derived from statistical analysis. In fact such elements should represent the entire range of values assumed by the coefficients in the transformed original blocks, spanning from very probable small values to more rare large values. In order to derive these statistics, experiments were performed using Algorithm 1 described in Subsection IV-A on some test sequences as follows.

For each transformed original block $\widetilde{X}$ with the corresponding feature set $\Omega\{N, H, mode\}$, all T values in the dictionary were tested. For a given α_i the block $\widetilde{P}_{proc,\alpha_i}$ was computed by applying pattern H to the transformed prediction block $\widetilde{P}$ and then replacing all discarded coefficients with α_i, or $\widetilde{p}_{proc,\alpha_i}(m,n) = \alpha_i$ if $h(m,n) = 0$. The element in the dictionary at minimum prediction distortion (computed using SAD) was selected as in:

$$\underset{\alpha_i}{\operatorname{argmin}} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} |\widetilde{p}_{proc,\alpha_i}(m,n) - \widetilde{x}(m,n)|$$

Given a feature set $\Omega = \{N, H, mode\}$, the probability of occurrence of each element in the dictionary $P(\alpha_i|\Omega)$ was then estimated as the number of times the element α_i was selected over the total number of blocks coded with Ω.

A set of K target probabilities P_0, P_1, ..., P_{K-1} was also defined. Probability values spanning from 0.4 to 0.02 were used in the implementation described in this paper. Finally for each instance of Ω, K elements were selected from the dictionary to be included in F_Ω: for each target probability P_k, $k = 0, ..., K-1$, the element α_i such that $P(\alpha_i|\Omega)$ is closer to P_k was selected. To improve the efficiency of entropy coding when signalling the index to select the correct element in the look-up tables, elements are sorted by decreasing probability, or $P(f_\Omega(i)|\Omega) \geq P(f_\Omega(j)|\Omega)$ if $i < j$.

When coding a certain block, the feature Ω is computed and the appropriate look-up table is used. Each element $f_\Omega(k)$ in the look-up table is tested in a RD sense, and finally the optimal index k^o to identify the correct element is selected and transmitted in the bitstream to be used at the decoder side. Note that clearly all look-up tables must be available to both the encoder and decoder.

C. Proposed Algorithm

The two steps of the proposed approach described respectively in Subsection IV-A and IV-B can be eventually integrated within a single algorithm. The first step consists in classifying coefficients within the transformed prediction block by means of masking patterns, to select those that should be preserved and those that can be discarded. An appropriate pattern H_{j^o} must be selected at this purpose for each block. In the second step, discarded coefficients are replaced with more meaningful synthetic content by means of feature-dependent look-up tables. The feature set $\Omega = \{N, H, mode\}$ is derived for the current block, and a look-up table F_Ω is considered. An element $f_\Omega(k^o)$ must be appropriately extracted, and finally $\widetilde{p}_{proc}(m,n) = f_\Omega(k^o)$ if $h_{j^o}(m,n) = 0$.

In theory these steps should be performed in such a way that the optimal combination of best pattern and best element in the corresponding look-up table is selected. Algorithm 1 as presented in Subsection IV-A would need to be modified accordingly: a nested loop to test the K elements in F_Ω should be considered within the main loop described in Step 1 of such algorithm. Selecting and transmitting the optimal value of both indexes k and j in a RD sense is not feasible, because it would likely result in very high bitrates, and it is also considerably expensive in terms of computational complexity. In total $(M \times K) + 1$ iterations would need to be performed (this also includes testing of the trivial pattern). Performing this number of iterations is not optimal even if very few elements are considered in the look-up tables.

To solve these issues a different approach can be formulated. Instead of selecting in a RD sense and transmitting in the bitstream the index j^o to identify the best pattern, the choice of pattern can be fixed by means of statistical analysis, depending on features of the current block such as intra-prediction mode $mode$ and TU size N. While more complex statistics might be used, for simplicity in this paper only intra-prediction mode and TU size were considered at this purpose. Table IV already presents the most frequently selected pattern (excluding the trivial pattern H_0) for each combination of these features. Such pattern $H_{N,mode}$ as reported in the table can be used any time a block of size $N \times N$ is encoded using intra-prediction mode $mode$.

Following from this restriction, the feature set reduces to $\Omega = \{N, mode\}$ due to the fact that the pattern depends on the other two features. Consequently a much smaller number of look-up tables need to be computed and stored in the encoder and decoder resources. Two example look-up tables as considered in the implementation used in this paper are presented in Table V in the case $N = 8$, $mode = 10$ and in the case $N = 32$, $mode = 0$ respectively. Interestingly, these examples suggest that values included in the look-up tables used for small blocks span a much wider range than those selected for larger TUs.

Following this adaptation an algorithm to perform frequency domain prediction processing can be defined which only needs $K + 1$ iterations for each block, referred to as Algorithm 2 in the rest of this paper and defined as follows. After a block of samples X is intra-predicted using a given mode, prediction and original signals are independently transformed obtaining $\widetilde{P}$ and $\widetilde{X}$ respectively. The pattern $H_{N,mode}$ is considered from Table V. Also the look-up table F_Ω is extracted according to the block features. An index k is initialised to zero. This is used to identify the elements in the look-up tables, with the exception of a value $k = 0$, reserved to signal the case when the trivial pattern H_0 is used and no coefficient is discarded (and consequently no processing is performed).

Then:

1) If $k \neq 0$ the element $f_\Omega(k)$ is extracted from the look-up table. Pattern $H_{N,mode}$ is applied to $\widetilde{P}$ and $\widetilde{P}_{proc}$ is

TABLE V: Look-up tables for two example feature sets.

$\Omega = \{8, 10\}$		$\Omega = \{32, 0\}$	
Index	**Element**	**Index**	**Element**
0	*reserved*	0	*reserved*
1	0	1	0
2	$+2$	2	$+1$
3	-2	3	$+2$
4	16	4	-1
5	-16	5	$+4$
6	-96	6	-2
7	$+128$	7	$+8$

obtained as:

$$\widetilde{p}_{proc}(m,n) = \begin{cases} \widetilde{p}(m,n) & \text{if } h_{N,mode}(m,n) = 1; \\ f_{\Omega}(k) & \text{otherwise.} \end{cases}$$

2) Otherwise if $k = 0$, the trivial pattern H_0 is used, namely $\widetilde{P}_{proc} = \widetilde{P}$.
3) The residual samples are computed in the frequency domain using $\widetilde{P}_{proc}$, quantised (to obtain $\widetilde{C}$), dequantised and inverse transformed (to obtain $\widetilde{X}_{rec}$). $\widetilde{C}$ and $\widetilde{X}_{rec}$ are used to compute the RD cost relative to the current element defined by k. A temporary solution is considered as k^o such that $\widetilde{P}_{proc}$ returns the current minimum RD cost.
4) If $k < K$, the index k is incremented by 1 and the algorithm goes back to step 1. Otherwise the element at minimum RD cost is output, identified by its optimal index k^o.

Only the optimal index k^o needs to be encoded in the bitstream when using this algorithm, to be extracted at the decoder side and used to select the optimal solution. Note that this is the only overhead required by the entire proposed method.

V. Results

In order to evaluate the approach presented in this paper and consequently validate the conclusions of the analysis carried out in the previous sections, several tests were performed to compare the performances of the proposed method with conventional HEVC coding. Results are mainly presented in terms of the BD-rate measure in percentage, a well known metric used to compare the efficiency of an encoder with respect to an anchor. Negative values of the BD-rates correspond to a more efficient encoding; this can be an effect of achieving higher qualities of the decoded signal while preserving the bitrate, or achieving lower bitrates while preserving the quality, or both decreasing bitrates and increasing quality of the signal at the same time. Conventional HEVC based on the reference software version $HM10.1 - rext2$ was used as the anchor.

All tests were performed on proposed approach and conventional HEVC using the configuration parameters and encoder settings specified in JCT-VC common test conditions [15]. Notice that the intra-prediction scheme implemented in the HM reference software makes use of some speed-ups implemented and enabled by default to reduce coding complexity. Some of these speed-ups are not compatible with the proposed approach and were therefore disabled in all tests in this section. In order to obtain a fair comparison, these tools were disabled also when testing conventional HEVC. Tests were performed under high quality conditions, namely using four low QP values of 2, 5, 7 and 12. Notice that even though the proposed approach is disabled on the smallest 4×4 TUs, still all allowed TU sizes (from 4×4 to 32×32 samples) were enabled to be tested in these experiments both when using the modified encoder and conventional HEVC. When testing 4×4 TUs using the modified encoder, such TUs are encoded following the conventional HEVC scheme in which no frequency domain prediction processing is applied.

TABLE VI: Still Image Coding.

Image	**BD-rates**
Stone building	-1.2%
Red door	-1.6%
Hats	-2.5%
Girl in red	-2.0%
Motocross bikes	-0.9%
Sailboat	-1.5%
Window	-2.0%
Market place	-1.1%
Spinnakers	-2.1%
Sailboat race	-1.9%
Pier	-1.5%
Couple on beach	-1.9%
Mountain stream	-1.3%
Water rafters	-1.4%
Girl	-2.3%
Tropical key	-1.8%
Monument	-1.9%
Model in black	-1.8%
Lighthouse	-1.9%
Mustang	-2.0%
Portland headlight	-2.0%
Barn and pond	-1.9%
Parrots	-3.0%
Chalet	-1.2%

While the approach is mainly proposed as a tool for compressing video sequences, it only directly affects intra-predicted blocks and for this reason its performance can also be tested on still images. For this reason first tests shown here were performed on the Kodak image data set as reported in Table VI. All images were encoded more efficiently using the proposed approach than the anchor. On average -1.8% gains were reported with up to -3.0% and -2.5% gains obtained for the Parrots and Hats images respectively. It is interesting to report some statistics on the percentages of TUs that were encoded using the modified encoder scheme with respect to the total. In fact, the proposed algorithm involves a RD decision to choose whether a given TU should be coded using the modified encoder scheme or conventional HEVC (where the latter is signalled with an index $k = 0$ as illustrated in Table V). As an example consider the Hats image encoded with $QP = 7$. Out of all TUs encoded in which the approach is enabled (namely larger or equal than 8×8 samples), 80% were encoded using the modified encoder scheme instead of conventional HEVC. This means that only in 20% of the cases, a conventional encoder scheme without frequency domain prediction processing was selected by the encoder. Also, it is interesting to mention some results obtained by restricting the encoder to only test TU sizes in which the approach is enabled, i.e. those larger or equal than 8×8 samples. When using such restriction (namely 4×4 TUs are not tested during

TABLE VII: Video Coding in All-Intra Configuration

Resolution	Sequence	BD-rates
Standard sequences		
2560 × 1600	Steamlocomotive	−2.3%
	Nebuta	−1.5%
1920 × 1080	Basketballdrive	−4.6%
	BQTerrace	−4.3%
	Kimono	−1.5%
1280 × 720	Johnny	−1.7%
	KristenAndSara	−1.7%
	FourPeople	−1.5%
832 × 480	Mobisode2	−5.2%
	Keiba	−2.7%
	Partyscene	−2.3%
	BasketballDrill	−2.1%
416 × 240	Basketballpass	−3.2%
	Racehorses	−2.4%
	BQSquare	−1.8%
Screen content		
1024 × 768	ChinaSpeed	−1.4%
	SlideShow	−1.2%
Ultra high-definition content		
3830 × 2860	Lupo Boa	−2.0%
	Veggie Fruits	−1.5%

the encoding), the approach was shown providing in average -1% additional BD-rate gains when tested on the same images as in Table VI.

Similar results were obtained in the case of video coding. Test material used in this set consists of test sequences used in JCT-VC common test conditions [15], and also screen-content sequences and ultra high-definition sequences. Results of the approach for the all-intra configuration are shown in Table VII. This configuration consists of encoding all frames in the sequence using solely intra-predicted blocks and is mostly used in high quality applications, for instance in digital camcorders when storing a sequence immediately after capturing. The proposed approach consistently increases the coding efficiency compared with conventional HEVC, obtaining on average -2.7% gains in test sequences in the JCT-VC standard test conditions. The performances of the approach are influenced by the original resolution of the encoded sequences. Best results were obtained in particular when coding sequences at 1920×1080 resolution (-4.6% gains obtained in the Basketballdrive sequence) and at 832×480 resolution (-5.2% gains obtained in the Mobisode2 sequence).

The approach was also tested with other types of content specifically interesting for high quality conditions. In particular, two screen-content sequences were tested, namely sequences containing computer generated scenery such as graphic overlays, large amounts of texts, scrolling subtitles and so on. High quality coding is particularly relevant for this kind of content, for instance in the case of screen mirroring applications or in medical imaging. Gains were reported for both tested sequences as shown in Table VII.

Finally the approach was also tested on two ultra high-definition sequences. These are sequences at a resolution of 3840×2860 luma samples. High quality video coding is relevant in this case mostly due to the increasing demand of the general public for ultra high-resolution content at very high levels of quality. Again gains were reported in both tested sequences as reported in Table VII.

TABLE VIII: Video Coding in Random Access and Low Delay Configurations.

Random access		
Resolution	**Sequence**	**BD-rates**
1920 × 1080	Basketballdrive	−4.1%
	BQTerrace	−3.1%
	Kimono	−0.7%
Low delay		
Resolution	**Sequence**	**BD-rates**
1920 × 1080	Basketballdrive	−4.3%
	BQTerrace	−3.2%
	Kimono	−0.6%

While the method directly affects only intra-predicted blocks (and as such has the greatest impact when testing the all-intra configuration), its effects have a considerable impact even when using the low delay or random access configurations. Even though when using such configurations most of the blocks are predicted using inter-prediction, improving intra-prediction has a strong impact by providing more accurate reference frames that can be exploited for improving motion compensation in subsequent frames. Some example results for these configurations are reported in Table VIII for test sequences at full HD resolution (1920×1080) from the JCT-VC standard test conditions. Again the results are presented in terms of BD-rates, where the approach is shown always achieving higher efficiencies than conventional HEVC with up to -4.3% coding gains.

Eventually some considerations can be reported regarding the complexity of the approach. In its current implementation, the method requires the modified encoder to inverse-transform and entropy code each TU once for each entry in the corresponding look-up tables; obviously this results in some additional computational complexity with encoding times up to 4 times higher than conventional HEVC when testing the all-intra configurations, or up to 2 times higher than conventional HEVC when testing the low delay configuration. On the other hand though the proposed method has a very small impact on the decoding complexity: less than 3% increase in decoding times was reported in the all-intra configuration, and even less was reported in the random access and low delay configurations.

VI. Conclusions

An analysis of intra-prediction methods for video compression under high quality conditions was reported in this paper. The study was based on modified encoder and decoder schemes in which original and prediction blocks are directly transformed, with the goal of highlighting the performance of each method in the frequency domain. The state-of-the-art HEVC standard was used as a base for the implementation. The analysis showed that high frequency components are difficult to predict using conventional intra-prediction methods, often resulting in high bitrates of the encoded signal. A novel approach was also proposed in this paper to improve the efficiency of high quality video coding based on such analysis. An additional stage of frequency domain processing was introduced during encoding and decoding before the residual computation, to selectively discard frequency components of the prediction signal and replace these with predefined synthetic content. Tests showed that the approach always outperformed conventional HEVC coding achieving up to -5.2% coding gains on video sequences.

The study presented in this paper provides a valuable insight on the behaviour of intra-prediction methods directly in the frequency domain. While the approach proposed here already outperforms conventional state-of-the-art video coding, more complex approaches may be formulated based on such analysis to further improve the efficiency of video coding especially under high quality constraints.

References

[1] G. Sullivan, J. Ohm, W.-J. 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] J. Ohm, G. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC)," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1669–1684, 2012.

[4] K. Ugur and J. Lainema, "Updated results on HEVC still picture coding performance," document n. JCTVC-M0041, Incheon, 2012.

[5] K. Sharman, N. Saunders, and J. Gamei, "CE1: Test 1 - Rectangular transform units for 4:2:2 (and AHG7 benchmarks)," document n. JCTVC-L0182, Geneva, 2013.

[6] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra Coding of the HEVC Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1792–1801, 2012.

[7] X. Cao, C. Lai, Y. Wang, and Y. He, "Short Distance Intra Coding Scheme for HEVC," in *Picture Coding Symposium (PCS)*, 2012, pp. 501–504.

[8] A. Gabriellini, D. Flynn, M. Mrak, and T. Davies, "Combined Intra-Prediction for High-Efficiency Video Coding," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 7, pp. 1282–1289, 2011.

[9] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete Cosine Transform," *IEEE Transactions on Computers*, vol. C-23, no. 1, pp. 90–93, 1974.

[10] A. Fuldseth, G. Bjntegaard, M. Budagavi, and V. Sze, "CE10: Core Transform Design for HEVC," document n. JCTVC-G495, Geneva, 2011.

[11] W.-J. Han, J. Min, I.-K. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, M.-S. Cheon, N. Shlyakhov, K. McCann, T. Davies, and J.-H. Park, "Improved Video Compression Efficiency Through Flexible Unit Representation and Corresponding Extension of Coding Tools," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 12, pp. 1709–1720, 2010.

[12] C. Yeo, Y. H. Tan, and Z. Li, "Low-complexity Mode-dependent KLT for Block-based Intra Coding," in *Image Processing (ICIP), 18th IEEE International Conference on*, 2011, pp. 3685–3688.

[13] A. Saxena and F. Fernandes, "DCT/DST-Based Transform Coding for Intra Prediction in Image/Video Coding," *Image Processing, IEEE Transactions on*, vol. 22, no. 10, pp. 3974–3981, 2013.

[14] K. Ugur and B. O., "Performance Evaluation of DST in Intra-Prediction," document n. JCTVC-I0582, Geneva, 2012.

[15] Y. Zheng, M. Coban, and M. Karczewicz, "Common Test Conditions and Software Reference Configurations," document n. JCTVC-H1100, San Jose, 2012.

[16] H. Jingning, V. Melkote, and K. Rose, "Transform-domain Temporal Prediction in Video Coding: Exploiting Correlation Variation across Coefficients," in *IEEE International Conference on Image Processing*, 2010, pp. 953–956.

[17] ——, "Transform-domain Temporal Prediction in Video Coding with Spatially Adaptive Spectral Correlations," in *IEEE 13th International Workshop on Multimedia Signal Processing*, 2011, pp. 1–6.

[18] Kodak. Kodak Test Image set. [Online]. Available: http://r0k.us/graphics/kodak/

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

IEEE Transactions on Circuits and Systems for Video Technology

Paper Title: Frequency Domain Intra-Prediction Analysis and Processing for High Quality Video Coding
Authors Names: Saverio G. Blasi, Marta Mrak and Ebroul Izquierdo

Reply to the reviewers

We would like to thank the reviewers for their time spent reviewing our paper, and for these additional comments and remarks. We appreciate that the reviewers recognised our efforts in improving our work following from their previous reviews. In this document we are providing detailed answers to the new comments included in their reviews; in the remainder of this document, the reviewers comments are expressed in normal text, whereas the authors replies are highlighted in italic.

The major change we included in the paper consists in removing any restriction on the TU size used to provide the results. In the previously submitted version of the paper, results in section V A were presented by limiting TUs to sizes larger than 8×8*, namely the encoder was forced to skip testing of the smallest* 4×4 *TUs. We previously included these data to show a comparison between conventional HEVC and proposed method only in the cases in which the approach is actually used. As clearly detailed in the paper, the method is disabled on TUs smaller than* 8×8*, which means that in case* 4×4 *TUs are allowed during the encoding, they are tested and possibly encoded only using conventional HEVC. By including results of the approach without testing such smaller TUs, we were forcing the encoder to enable the proposed approach in all cases.*

Unfortunately, the inclusion of these results was also misleading and led to confusion with respect to the provided numbers and for this reason we decided to completely remove it from the currently submitted version of the paper. All results provided in the tables are now obtained without using such restriction. For the sake of completeness, when commenting the results obtained on still images, we briefly mention in the new version of the paper that by enabling this restriction (and limiting the encoder to test only TUs larger or equal than 8×8 *samples), higher gains may be obtained. Notice that all other results in the paper (such as those in section V B) did not make use of such limitation already in the previous version of the paper, and were obtained without any kind of restriction on the TU size. Due to the fact that results are now all obtained under the same configuration and encoder settings, we also removed the separation between subsections and put all results in the same section. We are sorry for the confusion on this matter and we hope that we solved this issue in this new submitted version of the paper.*

I. Review Number 1

- The authors have addressed all the concerns of the review and answered the questions. The technique descriptions are much clearer in the revised manuscript.

R: *Thank you very much for this comment.*

- It seems in Table VI, the authors still disabled 4×4 blocks, being the same as former edition. It would be better that 4×4 blocks are enabled even if your method is not performed on 4×4 blocks since the performance on that test set in comparison with HEVC can be known by authors. .

R: *As already mentioned in the previous part of this document, we are now presenting all results in the tables without any limitation on the TU size. The results included in Table VI are now obtained allowing all TU sizes, including testing of* 4×4 *blocks.*

- On page 11, Table IV presents the most frequently selected pattern for each combination of those features. What sequences the statistics are obtained from? The distribution percentage under given mode for different pattern could be given.

R: *The results in Table IV were obtained using the Kodak image test set, as specified in the resubmitted version of the paper. The authors believe the table to be already considerably large, due to the consistent amount of information presented (following from the large number of modes available in HEVC and in order to show results for all TU sizes in which the approach is enabled). Including also figures related with distribution percentage might decrease the clarity of presentation of the paper. For this reason the authors decided to leave such table as is in the current version of the paper.*

II. Review Number 2

- The authors have addressed the comments adequately and the presentation is much clear in the revision. One additional suggestion is that Table VIII only contains a subset of the test sequences as in Table VII, it may be better to show the matched set.

R: *First of all, we would like to thank the reviewer for their suggestions and for their appreciation of our efforts in addressing their previous comments. The approach is only applied to intra-predicted frames and as such its impact on other configurations (such as low delay or random access) is by definition limited. This is clear from the relatively modest results obtained in some cases in Table VII. We only included such results to show that the approach does have an impact in general during the encoding also in these other configurations, but we believe that providing additional results in such cases would not improve the paper. For this reason we decided to leave such results as they are in the current version of the paper.*

III. Review Number 3

- First, I would like to show my appreciation of the authors' efforts to make the algorithm clearer. The part on how to generate the synthesized content is added in the current version. The experimental result with all TU sizes enabled can still not be found in this version, not as said in the reply letter. In Section V A. and B., 4×4 TU is still disabled.

R: *We would like to thank the reviewer for this comment. As already mentioned in this document, we are sorry for the confusion generated by the results previously included in the paper. The results included in this new version of the paper are now obtained without any restriction on the TU size. Notice that, in the previous version of the paper, such limitation only applied to results included in subsection V A: results in subsection V.B were already obtained under the most challenging conditions and without considering any such limitation on the TU size. Due to the fact that results are all obtained now under the same encoder settings, we completely removed distinction between the two subsections and put all results under the same section V.*

- Without this result, we cannot see the efficiency of the proposed scheme considering the extensive encoder complexity increase and the fact that only disabling the fast mode decision of intra prediction part in the existing HEVC reference software will bring $1 - 2\%$ coding gains.

R: *We are sorry for the confusion on this matter. It is true that in order to apply the proposed method, one of the encoder speed-ups pre-implemented in the HM reference software has to be disabled. In particular we are referring to the speed-up which consists in only considering the maximum allowed TU size when testing the intra-modes to select an optimal mode, and then only testing full RQT on this optimal mode. Such speed-up is somehow not compatible with the proposed approach, because the patterns and corresponding processing would only be applied to the maximum allowed TU size in each block. This is not optimal in that the approach works at its best when tested on all TU sizes to select the optimal configuration of intra-mode, TU size and look-up table entry. For this reason, such speed-up was disabled from all tests. Notice that to make the comparison completely fair the speed-up was also disabled when testing conventional HEVC. The comparison is therefore fair in that any gain obtained as a result of disabling such speed-up would reflect in both the proposed method and conventional HEVC. Notice that we extended the text in the paper to clarify this matter and to specify that we are disabling some speed-up tools during the tests. We did not include many details regarding such speed-ups in the paper, as this would unnecessarily increase the length of the paper while such description would go out of the scope of the paper. We believe that the current version of the paper is much clearer thanks to the reviewer suggestion.*

- Why use constant value for all frequencies of the synthesized part for each given feature is still not addressed.

R: *In order for the method to work, the overhead added to each block must be as small as possible, and the processing must take as little time as possible (not to increase complexity too much). Using a constant value for replacing such coefficients in the blocks that can be discarded is a way to address both these aforementioned issues. In particular, by considering look-up tables, the overhead required by the approach is considerably small. Also, the complexity necessary to apply a pattern and process a block is relatively very small and only consists in replacing a coefficient with a pre-computed value (extracted from the table) according to a boolean decision (depending on the pattern). The text in section IV was extended to also include this clarification.*

- Some logics are not correct according to my understanding. For example, in the first paragraph of Page 4, ”Samples towards the right of the block () are likely to be predicted more accurately...” might be ”Samples towards the left of the block ()...”. In the paragraph just above Fig. 8,”prediction coefficients toward the left-most portion in the block would be discarded” might be ”...the right-most portion in the block...”.

R: *Thank you very much for these comments. We are sorry for including such mistakes in the paper; we have now corrected both these errors and thoroughly proof-read the paper.*