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 that require low bitrates, for example, for content distribution on a mass scale. For these applications, the performance of conventional coding methods is 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 paper. 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 this 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 using feature-dependent lookup 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*— Frequency estimation, High Efficiency Video Coding (HEVC), predictive coding, video compression.**

I. Introduction

MOST of the video compression standards available today, including the recently ratified state-of-the-art H.265/High Efficiency Video Coding (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 that 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 that attempts to further reduce spatial redundancies using a more suitable representation and successively quantizing 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 using 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. 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/Advanced Video Coding (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

Manuscript received January 20, 2014; revised May 6, 2014 and July 23, 2014; accepted September 14, 2014. Date of publication September 19, 2014; date of current version May 1, 2015. This paper was recommended by Associate Editor F. Wu.

S. G. Blasi and E. Izquierdo are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K. (e-mail: s.blasi@qmul.ac.uk; e.izquierdo@qmul.ac.uk).

M. Mrak is with the Research and Development Department, British Broadcasting Corporation, London W12 7SB, U.K. (e-mail: marta.mrak@bbc.co.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2014.2359097

very high frame rates and high bit depths, the quality of the delivered videos is becoming an extremely important issue even in the context of consumer applications. Users expect 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 that 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.

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 behaviors 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 this paper. In particular, an additional stage of processing is performed on the transformed prediction signal prior to the residual computation. The processing is performed using appropriately defined masking patterns and lookup 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 organized 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 the conclusion is 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 neighboring 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 using 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–E in Fig. 1(a).

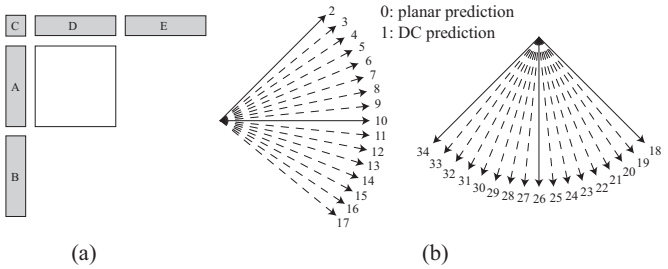


Fig. 1. (a) Intra prediction reference samples and (b) available modes in HEVC.

The standard allows up to 35 intra prediction modes [6], each labeled by an index from 0 to 34. Among these modes, dc prediction (labeled 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 (labeled 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 Fig. 1(a) as $s_A(i)$, and in D as $s_D(i)$ with $i = 0, \ldots, N-1$. 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 shown in Fig. 1(b). Modes labeled from 2 to 17 are referred to as horizontal directions, and modes labeled 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 [6]). Denote the samples in the currently predicted block as $p(i, j)$, where $i, j = 0, \ldots, N-1$. Each sample is predicted as the weighted interpolation of two reference samples as in

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

The weighting factor is computed as $w_j = |jd|\%$ [32], with $\%[\cdot]$ 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 Fig. 1(b). The index t is computed as

$$t = i + c$$

where

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

and $\lfloor \cdot \rfloor$ 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 artifacts 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 artifacts. 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 shown in the example in Fig. 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 Fig. 2(a), and when it is disabled in Fig. 2(b). In the second case, clearly, the residual sample magnitude tends to increase toward 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 nonsquare 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.

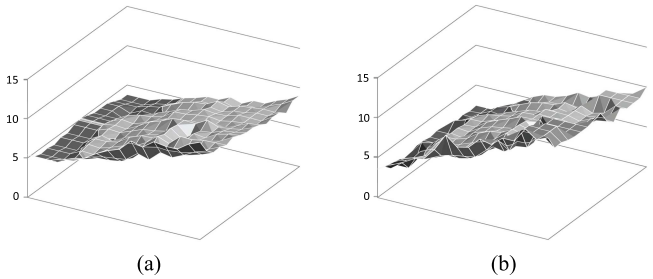


Fig. 2. Example of average per-sample absolute residual magnitude. Smoothing is (a) enabled and (b) disabled.

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 2-D 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 of samples obtained with horizontal angular intra prediction.

Samples toward 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 behavior 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 quantization. Each coefficient is quantized to a given step [depending on a parameter usually referred to as the quantization parameter (QP)]. The higher the QP, the coarser is the quantization. In the case of HEVC, QP is allowed to assume values between 0 and 52. When QP is set to 0, no quantization 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 quantization. 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. 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.

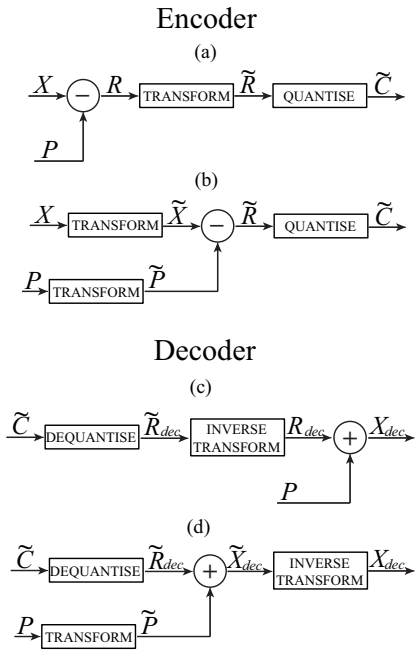


Fig. 3. Conventional (a) encoder and (c) decoder schemes compared with the proposed (b) encoder and (d) decoder schemes with direct transformation of prediction blocks.

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 quantized to obtain the coefficients $\widetilde{C}$. These steps are shown in the scheme in Fig. 3(a). At the decoder side, the coefficients are extracted from the bitstream, dequantized (i.e., rescaled), and inverse transformed. Due to the fact that the quantization is a nonreversible 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 Fig. 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 shown in Fig. 3(c). These are used to obtain the residuals $\widetilde{R}$, which are then quantized 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 dequantized to obtain $\widetilde{R}_{\mathrm{dec}}$. These are used to obtain the reconstructed block $\widetilde{X}_{\mathrm{dec}}$ in the frequency domain, which is finally inverse transformed to return X_{dec}, as in the scheme in Fig. 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 precomputed 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 Fig. 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

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

In practice, due to the truncation of the variables during the transform stages, the residual signal obtained using the proposed encoder scheme is different from the signal obtained using conventional schemes. It is worth clarifying how such truncations affect both schemes in Fig. 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 Fig. 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. 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 Fig. 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 Fig. 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 $\times$ 512 images.

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

The compression performance was measured in terms of Bjontegaard Delta (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 -0.18% and $+0.31\%$, respectively.

Note also that using the schemes in Fig. 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 Fig. 3(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 (SADs) or sum of squared differences, 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 analyzed to formulate appropriate processing methods to improve the coding efficiency.

To perform such analysis, the modified encoder scheme in Fig. 3(b) was implemented in the context of HEVC intra prediction and a few sequences were encoded to collect test 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 colocated 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, $K_{N,\text{mode}}$ 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, $K_{N,\text{mode}}$ is denoted as K. Refer to each transformed original or prediction block as $\widetilde{X}_i$ or $\widetilde{P}_i$, respectively, where $i = 0, 1, \ldots, K-1$. Finally, denote as $\widetilde{x}_i(m,n)$ and $\widetilde{p}_i(m,n)$ the samples at location (m,n) in the blocks $\widetilde{X}_i$ and $\widetilde{P}_i$, respectively. The correlation between the arrays $[\widetilde{x}_0(m,n), \widetilde{x}_1(m,n), \ldots, \widetilde{x}_{K-1}(m,n)]$ and $[\widetilde{p}_0(m,n), \widetilde{p}_1(m,n), \ldots, \widetilde{p}_{K-1}(m,n)]$ for a given transform size $N \times N$ and a certain intra prediction mode *mode* can be defined as

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

where the expected values $E\{\cdot\}$ and standard deviations σ are estimated from the samples. Values of $R_{\{N,\text{mode}\}}(m,n)$ 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.

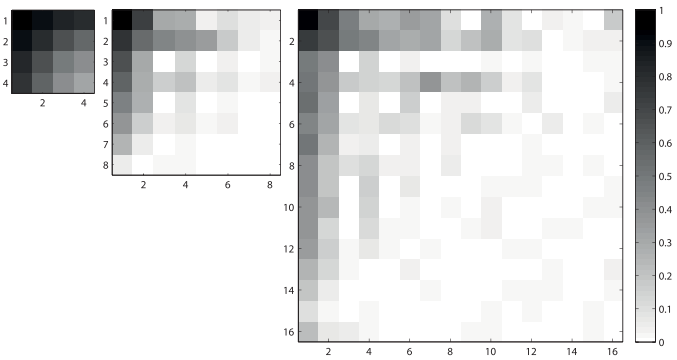


Fig. 4. Per-sample correlation between original and prediction samples for blocks of different sizes using planar mode.

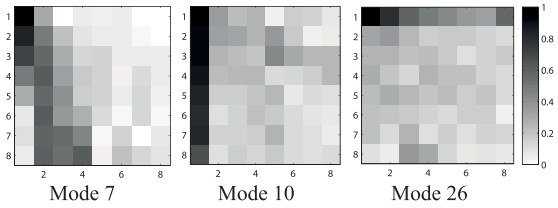


Fig. 5. Per-sample cross-correlation for 8×8 blocks predicted using different intra prediction modes.

The correlation values for TUs of different sizes (4×4, 8×8, and 16×16) are shown in Fig. 4 for the planar mode. Clearly, the transform size has an evident impact on the correlation values especially at higher frequencies (i.e., toward 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 Fig. 5. Correlation values for three different angular directions (namely, modes 7, 10, and 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 Fig. 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

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

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 behavior 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 shown in Fig. 8). The correlation values at these three locations is reported in 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 the fact that intra prediction works better on smaller blocks is a well-known behavior, 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.

Interestingly, the analysis presented in this paper does not only confirm this behavior but also highlights that these effects have a direct impact on different frequency components. For instance, the sample at the highest frequency [labeled 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 analyzing 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 labeled 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 behavior is reported in the case of the bottom-left sample labeled 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 quantization but needs to be transmitted in the bitstream.

To limit these effects and possibly improve compression efficiency, a different approach is proposed in this paper to

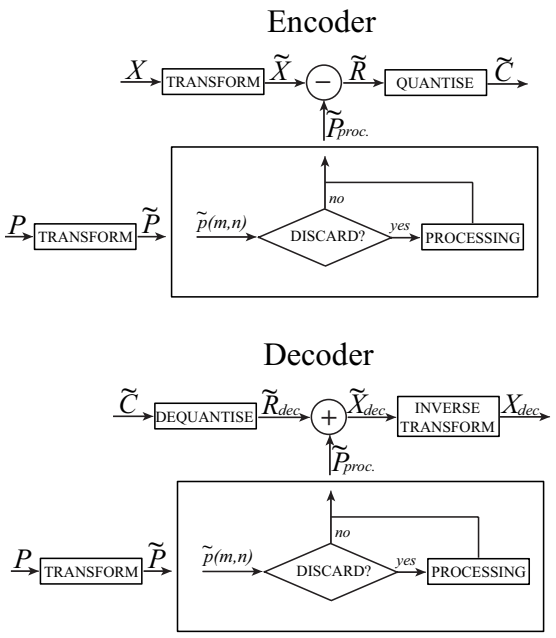


Fig. 6. Proposed encoder and decoder schemes including processing of the transformed prediction blocks.

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 shown in Fig. 6. Each of these two steps is detailed in the rest of this section.

A. Discarding Coefficients Using Patterns

The strongly localized distribution of correlation values obtained when using particular intra prediction modes (as shown in Fig. 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 behaviors 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 to follow these behaviors can be easily formalized 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

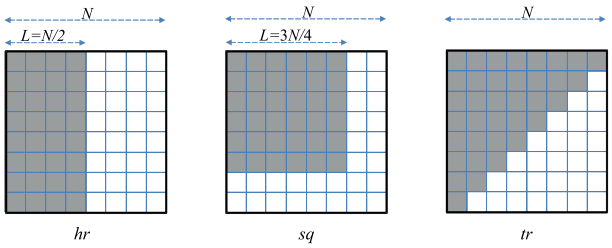


Fig. 7. Example of patterns used for frequency-domain prediction processing. Coefficients in shaded locations are preserved; coefficients in white locations are discarded.

is preserved or it is discarded and replaced. Although more complex options are possible, only four classes of patterns are considered in this paper. 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 the composition of various pattern is shown as follows.

1) 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

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

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

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

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

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

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 using appropriate methods, as illustrated later in this section. Some example patterns are shown in Fig. 7.

The schemes in Fig. 6 were implemented in HEVC. 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 $\widetilde{p}_{\text{proc}}(m,n)$ the elements of $\widetilde{P}_{\text{proc}}$, this corresponds to

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

The following algorithm, referred to as Algorithm 1, was then implemented at the encoder side. A list of all considered patters $H_1, H_2, \ldots, H_M$ 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 $m = 0, \ldots, N-1$, $n = 0, \ldots, N-1$, 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 initialized to zero and the following actions are performed.

1) The pattern H_j is extracted from the list and applied to $\widetilde{P}$ to obtain $\widetilde{P}_{\text{proc}}$. The residual samples are computed as $\widetilde{R} = \widetilde{X} - \widetilde{P}_{\text{proc}}$ and quantized to obtain $\widetilde{C}$. This is dequantized and inverse transformed to obtain $\widetilde{R}_{\text{dec}}$, which is finally used to compute the reconstruction $\widetilde{X}_{\text{rec}} = \widetilde{R}_{\text{dec}} + \widetilde{P}_{\text{proc}}$.
2) $\widetilde{C}$ and $\widetilde{X}_{\text{rec}}$ 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 signaled 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 Fig. 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	$H_{sq,N/4}$	$H_{sq,N/4}$	$H_{hr,N/4}$

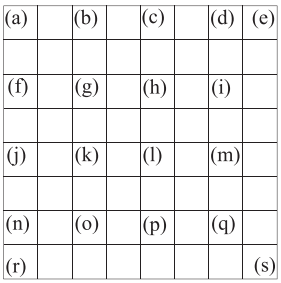


Fig. 8. Sample location labels.

B. Replacing Coefficients With Lookup 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 Fig. 3(b). Consider also that the transformed original blocks $\widetilde{X}$ are collected while encoding, and classified depending on the optimal intra prediction mode and TU size selected by the encoder. The histograms in Fig. 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 Fig. 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 Fig. 9.

Consider now that the same content is encoded using the schemes in Fig. 6. Assuming that such block is processed using

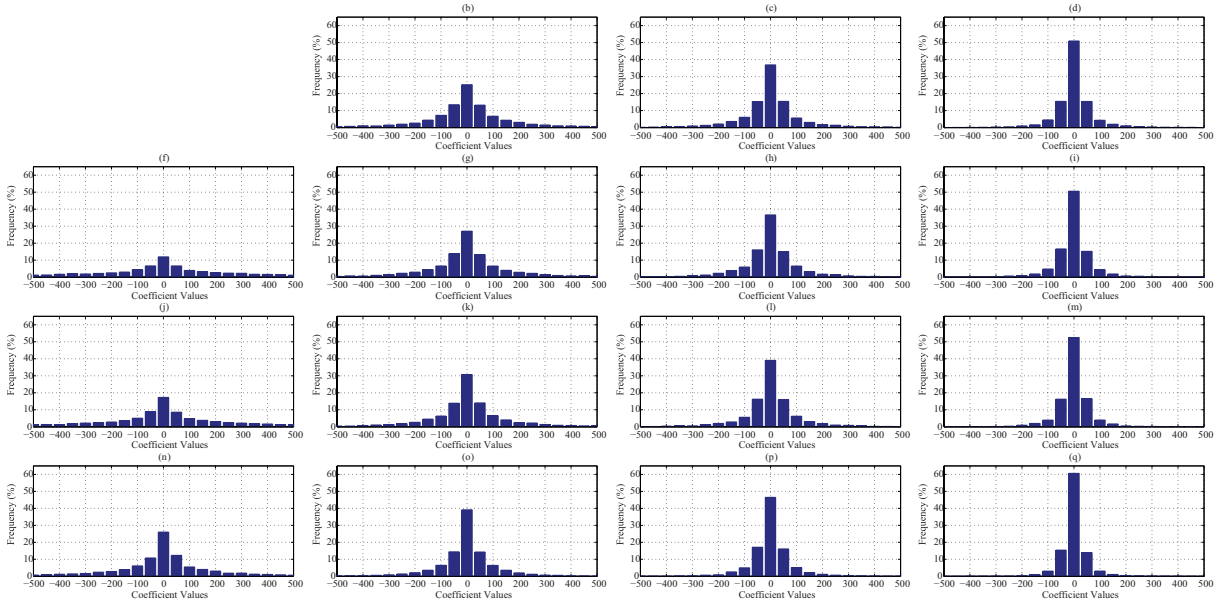


Fig. 9. Frequency of occurrence of coefficient values at different locations in the transformed blocks, for intra prediction mode 9, block size 8×8.

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. 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 behavior clearly is not optimal in a large number of cases, as highlighted by the histograms in Fig. 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 Fig. 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 Fig. 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. 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 formalizing and optimizing 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 $\alpha_0, \ldots, \alpha_{T-1}$. These values are selected to be representative of the range spanned by the actual coefficients at high frequencies. A tradeoff 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 $+128$.

To correctly select and use a certain element from the dictionary, this must be signaled 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 Fig. 6. Unfortunately, high bitrates may result as a consequence of this signaling, 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 lookup tables.

In particular, for each TU, the feature set $\Omega = \{N, H, \text{mode}\}$ 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 lookup table F_Ω is defined as an indexed array $f_\Omega(k)$, where $k = 0, \ldots, K-1$; the K elements in each table are a particular subset of the elements $\alpha_0, \ldots, \alpha_{T-1}$ in the dictionary. The length of the lookup 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 lookup table and their order can be derived from statistical analysis. 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. To derive these statistics, experiments were performed using Algorithm 1 described in Section IV-A on some test sequences as follows.

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

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

Given a feature set $\Omega = \{N, H, \text{mode}\}$, 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 $P_0, P_1, \ldots, P_{K-1}$ 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, $k = 0, \ldots, K-1$, 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 signaling the index to select the correct element in the lookup 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 lookup table is used. Each element $f_\Omega(k)$ in the lookup 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 lookup tables must be available to both the encoder and decoder.

C. Proposed Algorithm

The two steps of the proposed approach described, respectively, in Sections 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 using 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 using feature-dependent lookup tables. The feature set $\Omega = \{N, H, \text{mode}\}$ is derived for the current block, and a lookup table F_Ω is considered. An element $f_\Omega(k^o)$ must be appropriately extracted, and finally $\widetilde{p}_{\text{proc}}(m,n) = f_\Omega(k^o)$ if $h_{j^o}(m,n) = 0$.

TABLE V
LOOKUP 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

In theory, these steps should be performed in such a way that the optimal combination of best pattern and best element in the corresponding lookup table is selected. Algorithm 1, as presented in Section 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 lookup 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 using 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 $H_{N,\text{mode}}$ 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 $\Omega = \{N, \text{mode}\}$ due to the fact that the pattern depends on the other two features. Consequently, a much smaller number of lookup tables need to be computed and stored in the encoder and decoder resources. Two example lookup 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 lookup 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 $H_{N,\text{mode}}$ is considered from Table V. In addition, the lookup table F_Ω

is extracted according to the block features. An index k is initialized to zero. This is used to identify the elements in the lookup 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).

1) If $k \neq 0$ the element $f_\Omega(k)$ is extracted from the lookup table. Pattern $H_{N,\text{mode}}$ is applied to $\widetilde{P}$ and $\widetilde{P}_{\text{proc}}$ is obtained as
$$\widetilde{p}_{\text{proc}}(m,n) = \begin{cases} \widetilde{p}(m,n) & \text{if } h_{N,\text{mode}}(m,n) = 1 \\ f_\Omega(k) & \text{otherwise.} \end{cases}$$
2) Otherwise, if $k = 0$, the trivial pattern H_0 is used, namely, $\widetilde{P}_{\text{proc}} = \widetilde{P}$.
3) The residual samples are computed in the frequency domain using $\widetilde{P}_{\text{proc}}$, quantized (to obtain $\widetilde{C}$), dequantized and inverse transformed (to obtain $\widetilde{X}_{\text{rec}}$). $\widetilde{C}$ and $\widetilde{X}_{\text{rec}}$ 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 $\widetilde{P}_{\text{proc}}$ 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

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-rext 2 was used as the anchor.

All tests were performed on proposed approach and conventional HEVC using the configuration parameters and encoder settings specified in Joint Collaborative Team on Video Coding (JCT-VC) common test conditions [15]. Notice that the intra prediction scheme implemented in the HM reference software makes use of some speedups implemented and enabled by default to reduce coding complexity. Some of these speedups are not compatible with the proposed approach and were therefore disabled in all tests in this section. 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. The proposed algorithm involves an RD decision to choose whether a given TU should be coded using the modified encoder scheme or conventional HEVC (where the latter is signaled with an index $k = 0$, as shown 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. In addition, 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

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%

TUs are not tested during the encoding), the approach was shown providing in average -1% additional BD-rate gains when tested on the same images, as shown 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 ultrahigh-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 Basketball drive 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, and scrolling subtitles. 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 ultrahigh-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 ultrahigh-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 lookup tables; obviously, this results in some additional computational complexity with encoding times up to four times higher than conventional HEVC when testing the all-intra-configurations, or up to two 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. CONCLUSION

An analysis of intra prediction methods for video compression under high quality conditions was reported in this paper. This paper 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 behavior 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. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[3] J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—Including High Efficiency Video Coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[4] K. Ugur and J. Lainema, *Updated Results on HEVC Still Picture Coding Performance*, document JCTVC-M0041, Incheon, Korea, 2012.

[5] K. Sharman, N. Saunders, and J. Gamei, *CE1: Test 1—Rectangular Transform Units for 4:2:2 (and AHG7 Benchmarks)*, document JCTVC-L0182, Geneva, Switzerland, 2013.

[6] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792–1801, Dec. 2012.

[7] X. Cao, C. Lai, Y. Wang, and Y. He, "Short distance intra coding scheme for HEVC," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 501–504.

[8] A. Gabriellini, D. Flynn, M. Mrak, and T. Davies, "Combined intra-prediction for high-efficiency video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 7, pp. 1282–1289, Nov. 2011.

[9] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.

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

[11] W.-J. Han *et al.*, "Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1709–1720, Dec. 2010.

[12] C. Yeo, Y. H. Tan, and Z. Li, "Low-complexity mode-dependent KLT for block-based intra coding," in *Proc. 18th IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2011, pp. 3685–3688.

[13] A. Saxena and F. C. Fernandes, "DCT/DST-based transform coding for intra prediction in image/video coding," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3974–3981, Oct. 2013.

[14] K. Ugur and O. Bici, *Performance Evaluation of DST in Intra Prediction*, document JCTVC-I0582, Geneva, Switzerland, 2012.

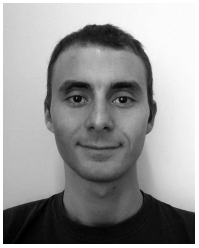
[15] Y. Zheng, M. Coban, and M. Karczewicz, *Common Test Conditions and Software Reference Configurations*, document JCTVC-H1100, San Jose, CA, USA, 2012.

[16] J. Han, V. Melkote, and K. Rose, "Transform-domain temporal prediction in video coding: Exploiting correlation variation across coefficients," in *Proc. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 953–956.

[17] J. Han, V. Melkote, and K. Rose, "Transform-domain temporal prediction in video coding with spatially adaptive spectral correlations," in *Proc. IEEE 13th Int. Workshop Multimedia Signal Process.*, Oct. 2011, pp. 1–6.

[18] Kodak. *Kodak Test Image Set*. [Online]. Available: http://r0k.us/graphics/kodak/, accessed Jan. 2015.

[19] G. Bjontegaard, *Improvements of the BD-PSNR Model*, document ITU-T SG16/Q6, 2008.



Saverio G. Blasi (S'14) was born in Rome, Italy. He received the B.Sc. degree in control systems engineering and the M.Sc. degree in automation engineering from University of Rome "La Sapienza," Rome, in 2006 and 2008, respectively, and the M.Sc. degree in signal processing for multimedia technologies from King's College London, London, U.K., in 2010. He is currently working toward the Ph.D. degree in video and image coding, and compression with the Multimedia and Vision Research Group, Queen Mary University of London, London.

Marta Mrak (SM'13) received the Dipl.-Ing. and M.Sc. degrees in electronic engineering from University of Zagreb, Zagreb, Croatia, and the Ph.D. degree from Queen Mary University of London, London, U.K.

She was a Post-Doctoral Researcher with University of Surrey, Surrey, U.K., and Queen Mary University of London, before joining the Research and Development Department, British Broadcasting Corporation, London, in 2010, where she has been involved in video compression research and the H.265/HEVC standardization. She has been involved in several projects funded by the European and U.K. research councils in roles ranging from Researcher to Scientific Coordinator. She has co-authored over 100 papers, book chapters, and standardization contributions, and also co-edited a book entitled *High-Quality Visual Experience* (Springer, 2010).

Dr. Mrak is a member of the Multimedia Signal Processing Technical Committee of the IEEE, an Area Editor of *Signal Processing: Image Communication* (Elsevier), and a Guest Editor of several special issues in relevant journals. She received the 2002 German DAAD Scholarship for Video Compression Research at the Heinrich Hertz Institute, Germany.

Ebroul Izquierdo (SM'03) received the M.Sc., C.Eng., Ph.D., and Dr. Rerum Naturalium (Ph.D.) degrees from Humboldt University, Berlin, Germany.

He is currently the Chair of the Multimedia and Computer Vision, and the Head of the Multimedia and Vision Group with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. He has been a Senior Researcher with Heinrich Hertz Institute for Communication Technology, Berlin, and the Department of Electronic Systems Engineering, University of Essex, Colchester, U.K. He has authored over 450 technical papers, including chapters in books, and holds several patents in multimedia signal processing.

Prof. Izquierdo is a Chartered Engineer, a Fellow Member of the Institution of Engineering and Technology (IET), a member of the British Machine Vision Association, the Chairman of the IET Professional Network on Information Engineering, a member of the Visual Signal Processing and Communication Technical Committee of the IEEE Circuits and Systems Society, and a member of the Multimedia Signal Processing Technical Committee of the IEEE. He has been an Associated and Guest Editor of several relevant journals, including IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, EURASIP *Journal on Image and Video processing*, *Signal Processing: Image Communication* (Elsevier), EURASIP *Journal on Applied Signal Processing*, IEEE PROCEEDINGS ON VISION, IMAGE AND SIGNAL PROCESSING, *Journal of Multimedia Tools and Applications*, and *Journal of Multimedia*.