Improved Video Compression Efficiency Through Flexible Unit Representation and Corresponding Extension of Coding Tools

Woo-Jin Han, *Member, IEEE,* Junghye Min, Il-Koo Kim, Elena Alshina, Alexander Alshin, Tammy Lee, Jianle Chen, Vadim Seregin, Sunil Lee, *Member, IEEE,* Yoon Mi Hong, Min-Su Cheon, Nikolay Shlyakhov, Ken McCann, Thomas Davies, and Jeong-Hoon Park

***Abstract*—This paper proposes a novel video compression scheme based on a highly flexible hierarchy of unit representation which includes three block concepts: coding unit (CU), prediction unit (PU), and transform unit (TU). This separation of the block structure into three different concepts allows each to be optimized according to its role; the CU is a macroblock-like unit which supports region splitting in a manner similar to a conventional quadtree, the PU supports nonsquare motion partition shapes for motion compensation, while the TU allows the transform size to be defined independently from the PU. Several other coding tools are extended to arbitrary unit size to maintain consistency with the proposed design, e.g., transform size is extended up to 64×64 and intraprediction is designed to support an arbitrary number of angles for variable block sizes. Other novel techniques such as a new noncascading interpolation filter design allowing arbitrary motion accuracy and a leaky prediction technique using both open-loop and closed-loop predictors are also introduced. The video codec described in this paper was a candidate in the competitive phase of the high-efficiency video coding (HEVC) standardization work. Compared to H.264/AVC, it demonstrated bitrate reductions of around 40% based on objective measures and around 60% based on subjective testing with 1080p sequences. It has been partially adopted into the first standardization model of the collaborative phase of the HEVC effort.**

***Index Terms*—Flexible unit representation, HEVC, video coding.**

I. Introduction

RECENTLY, with the ever increasing popularity of high-definition (HD) video content, video compression technologies which can provide higher coding efficiency than existing video coding standards have received increased attention. Early video coding standards such as MPEG-1 [1] and MPEG-2 [2] employ fixed-size block motion compensation method with a macroblock size of 16×16. In the state-of-the-art standard H.264/AVC [3], this macroblock size is retained but with the addition of a depth-2 quadtree, which significantly improves coding efficiency. However, it has been observed that the maximum 16×16 block size is too small when applied to high resolution video and causes inefficiency [4], [5]. To solve this problem, recent Visual Coding Experts Group (VCEG) proposals have extended the maximum motion block size up to 64×64 by adding additional large motion block modes on top of the conventional 16×16 H.264/AVC macroblock syntax [6]–[8].

Although those approaches suggested that the coding efficiency for the high resolution video materials can be improved by providing a better tradeoff between prediction performance and side information, they simply tried to add the large size motion partition while maintaining the existing 16×16 macroblock sizes, which imposes restrictions on the flexibility of the quadtree structure. For example, spatial/temporal prediction mode switching, intracoding tools and spatial transforms are still defined based on the 16×16 macroblock sizes, thus the schemes generally suffer from lower coding efficiency in intracoded blocks.

There has been extensive research on introducing a general multidepth quadtree for variable block-size motion compensation [9]–[12]. A frame is typically divided into fixed-size superblocks, with each superblock able to be further split. This kind of generic quadtree can provide a consistent codec design; however, focusing only on motion compensation does not exploit the full potential of the technique.

This paper describes a video compression framework where the whole codec has been designed to exploit flexible block sizes. A highly flexible hierarchical structure based on the generic quadtree scheme is defined by three independent block concepts: coding unit (CU), prediction unit (PU), and transform unit (TU). CU is the basic unit of region splitting. CU is analogous to the concept of macroblock, but it does not restrict the maximum size and it allows recursive splitting into four equal size CUs to improve the contents adaptivity. PU is the

Manuscript received July 23, 2010; revised October 11, 2010; accepted October 19, 2010. Date of publication November 15, 2010; date of current version January 22, 2011. This paper was recommended by Associate Editor T. Wiegand.

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, and J. H. Park are with the Digital Media and Communication Research and Development Center, Samsung Electronics, Suwon 443-742, Korea (e-mail: wjhan.han@samsung.com; jh643.min@samsung.com; ilkoo.kim@samsung.com; elena_a.alshina@samsung.com; alexander_b.alshin@samsung.com; tammy.lee@samsung.com; jianle.chen@samsung.com; v.seregin@samsung.com; sunil.lee@samsung.com; yoonmi.hong@samsung.com; minsu.cheon@samsung.com; n.shlyakhov@samsung.com; jeonghoon@samsung.com).

K. McCann is with ZetaCast, Bath, BA2 6NZ, U.K. (e-mail: ken@zetacast.com).

T. Davies is with British Broadcasting Corporation, London, W1A 1AA, U.K. (e-mail: thomas.davies@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.2010.2092612

1051-8215/$26.00 © 2010 IEEE

Nokia Exhibit 2006, p. 1
Amazon.com v. Nokia
IPR2024-01505

basic unit of inter/intraprediction and it may contain multiple arbitrary shape partitions in a single PU to effectively code irregular image patterns. TU is the basic unit of transform. It can be defined independently from the PU; however, its size is limited to the CU which the TU belongs to. This separation of the block structure into three different concepts allows each to be optimized according to its role, which results in the improved coding efficiency.

From these unit definitions, all coding processes can be defined in a very consistent way; large block sizes are supported in both motion compensation and intraprediction schemes, prediction mode switching is allowed at any block sizes and spatial transforms larger than 16×16 are also supported in both inter and intracoded blocks. Due to this consistency, every syntax element can be defined in exactly the same way independent of the unit sizes, which makes the specification and parsing process much simpler than would be the case with an extension scheme on top of a conventional 16×16 macroblock syntax.

Fig. 1 shows all proposed building blocks of the decoder. All the coding tools are extended to support the proposed flexible architecture. Intra prediction is extended to support an arbitrary number of angles rather than the conventional 9-modes defined in H.264/AVC. Fast integer transforms larger than 8×8 are developed to support large TU sizes. Edge definition for the deblocking filter is modified according to the CU, PU, and TU concepts. Partition information in quadtree-based adaptive loop filter (QALF) [13] is replaced with CU splitting information.

The rest of this paper is organized as follows. Section II introduces the details of the unit definitions and Section III presents the proposed coding tools for the interframe prediction. The remaining coding tools for the intraframe prediction, spatial transform, in-loop filtering, and entropy coding are introduced in Sections IV, V, VI, and VII, respectively. Section VIII summarizes the coding efficiency of the proposed techniques based on the objective and subjective results. Finally, Section IX concludes the paper. For a more complete description for the coding tools, please refer to [14] and [15].

II. Unit Definition

For efficient and flexible representation of video content with various resolutions, a triplet of CU, PU, and TU are introduced. The following subsections describe the details of each unit.

A. Coding Unit

The picture is divided into slices, with each slice composed of a sequence of largest coding units (LCUs). The LCU size is specified in the sequence parameter set (SPS) or slice header; the optimum LCU size is dependent on the application.

Fig. 2 shows quad-tree based CU structure. The LCU can be divided into four CUs by a 1-bit syntax, split_flag. Each split CU can be divided into four CUs recursively up to the maximum allowed hierarchical depth, also specified in the SPS or slice header. For example, if LCU size is equal to 128 and the maximum hierarchical depth is equal to 5, then

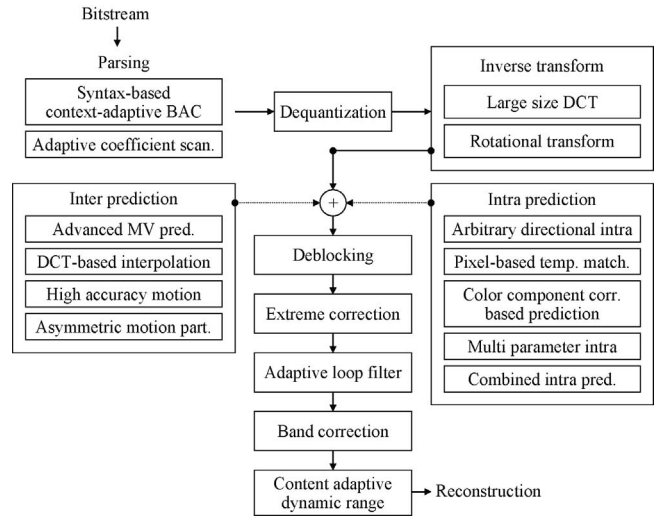


Fig. 1. All building blocks to decode the input bit-stream.

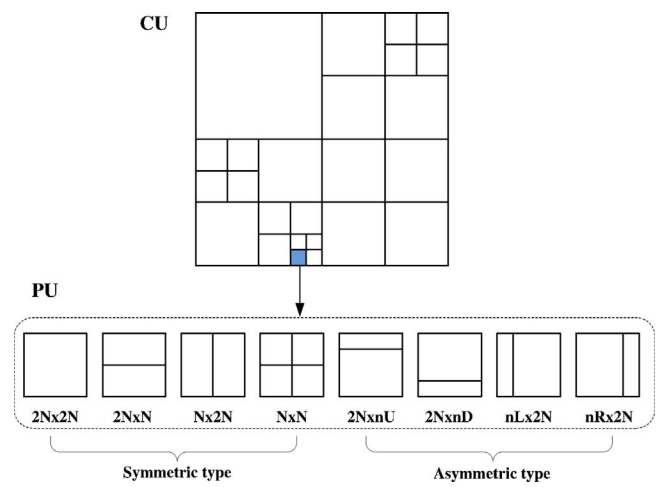


Fig. 2. Relationship between CU and PU.

5 kinds of CU sizes are possible: 128×128 (LCU), 64×64, 32×32, 16×16, and 8×8. The three syntax elements, LCU size, maximum hierarchical depth, and split_flags, completely specify the generic multidepth quadtree.

B. Prediction Unit

Coupled with CU, this paper introduces a basic unit for prediction: the PU. The PU is defined only for the leaf node CU which is not split in quadtree structure, and the size of the PU is limited to that of the CU. The prediction method is specified by the prediction mode and the PU splitting. As with H.264/AVC, the prediction mode can be one of the values among skip, intra, and inter. Fig. 2 shows different PU splitting strategy for a CU of size $2N \times 2N$, which is dependent on the property of each prediction mode.

Skip mode can be thought of as a special intermode in which no motion vector and residue information are transmitted. Typically it is used in an area of the image with smooth motion, thus no further splitting is necessary and only $2N \times 2N$ partition is used. For the normal intermode, a set of eight splitting options are defined to cover possible motion boundaries

Nokia Exhibit 2006, p. 2
Amazon.com v. Nokia
IPR2024-01505

inside the PU. Given a PU of size $2N \times 2N$, four symmetric splittings ($2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$) and four asymmetric splittings ($2N\times$nU, $2N\times$nD, nL$\times 2N$, nR$\times 2N$) are defined. The asymmetric PU splittings are included in the design to improve the coding efficiency for the irregular object boundaries, which otherwise would be constrained to being represented less efficiently by further CU splittings to the deeper depth. For the intramode, only two possible splittings $2N \times 2N$ and $N \times N$ are defined.

All information related to the prediction is signaled on a PU basis, for example, the prediction mode and PU splitting type are specified for each PU. Once the prediction mode and PU splitting are given, motion vector difference, intraprediction direction, reference indices for each PU partition are specified on a PU partition basis.

C. Transform Unit

The TU, for transform and quantization, is the third basis unit. Although, in principle, TU can be defined in a similar way to PU to support arbitrary shape transforms, only two TU partitions are used in this paper considering the implementation complexity.

When the tu_size_flag is equal to zero, the TU size is set equal to that of the CU which it belongs to. When tu_size_flag is equal to one, the TU size is set as $N \times N$ for symmetric PU splittings and $N/2 \times N/2$ for asymmetric PU splittings, respectively. This ensures that the transform which is not applied across motion boundaries can be tested in the rate-distortion optimization process for asymmetric PU partitions.

It should be noted that the transform may be applied to the residue generated by multiple PU partitions with different motion vectors or prediction methods. This approach is especially beneficial for coding semirandom residuals generated by complex texture areas where neighboring partitions share common signal characteristics, which can exploit the better energy compaction of larger transform [14].

D. Size-Independent Syntax Representation

The proposed unit definitions allow us to design a size-independent syntax representation which specifies all syntax elements in a consistent way independent of the unit size. By contrast, in H.264/AVC and its extension in [6]–[8], block level syntax elements such as prediction_mode, transform_flag, coded_block_flag, and intra_pred_mode are coded differently depending on block size. This size-independent method of representation allows syntax elements to be specified generically for all the remaining coding tools. This property can greatly simplify the specification effort as well as the actual parsing process, especially if a large number of hierarchical depths is allowed.

III. Interprediction

In this paper, three aspects of interframe prediction are improved: motion accuracy, interpolation filter, and motion vector prediction. For motion accuracy, quarter-pel precision with 1/12-pel optional refinement is used to improve the coding efficiency while minimizing the additional overhead for signaling the motion information. For motion compensation, noncascaded spatial filtering using a discrete cosine transform (DCT)-based interpolation filter is used for every motion accuracy position. For motion vector prediction, advanced motion vector prediction (AMVP), which is a motion vector competition method [16] adapted to the flexible unit representation, is used to reduce overhead for motion information.

A. High Accuracy Motion

Even though higher motion accuracy results in less prediction error, it does not guarantee improved coding efficiency, since it leads to an increase in the bit rate to convey the motion information. For example, 1/8-pel motion accuracy may often result in a reduction in the overall coding efficiency, except for particular sequences [17].

In this paper, high accuracy motion (HAM), which adaptively refines the motion information to an arbitrary accuracy, is proposed. In the HAM method, motion information at quarter-pel accuracy is followed by a refinement flag and optional refinement information at 1/12-pel accuracy. The refinement flag is set equal to 1 only when the refinement provides the gain in the rate-distortion optimization sense, otherwise, the flag is set to 0 and no refinement information is signaled. Although the refinement information can represent any accuracy, the current design uses 1/12-pel, since refinement in 1/8-pel accuracy results in the redundancy in the refinement information.

B. DCT-Based Interpolation Filter

Motion compensation with fractional pel accuracy requires interpolation of the reference signal. In H.264/AVC standard, the prediction values at half-pel positions are obtained using a 6-tap Wiener filter, while those at quarter-pel positions are obtained by bilinear combination of the samples at integer and half-pel positions [3]. This paper proposes a DCT-based interpolation filter (DCT-IF), which can directly provide the interpolated value at the desired fractional accuracy from the samples at integer-pel positions. Since the DCT-IF can provide the interpolated value without cascaded filtering, the motion compensation process can be simplified and the complexity can also be reduced for samples at quarter-pel accuracy. Furthermore, the DCT-IF provides a unified way to generate the interpolation filters supporting motion accuracy higher than quarter-pel, e.g. 1/8-pel, 1/12-pel, etc. Note that 12-tap and 6-tap DCT-IFs are used for luma and chroma, respectively.

1) *Filter Coefficient Generation:* In this paper, trigonometric interpolation, which has been widely used for the other applications [18], is exploited to construct DCT-IF for high accuracy motion compensation. Let $p(\alpha)$ denote the sample value at the fractional position $\alpha \in [-(M-1), M]$, where M is a positive integer. Then, $p(\alpha)$ can be derived as follows:

$$p(\alpha) = \mathbf{f}_{(\alpha)}^T \mathbf{p} = \left(\mathbf{w}_{(\alpha)}^T D\right) \mathbf{p} \tag{1}$$

where $\mathbf{p} = [p_{-(M-1)}, \ldots, p_M]^T$ is the vector of the given samples at the integer-pel positions, D is the $2M \times 2M$ DCT

Nokia Exhibit 2006, p. 3
Amazon.com v. Nokia
IPR2024-01505

TABLE I
FILTER COEFFICIENTS OF 6-TAP DCT-IF

α	$2M = 6$ (6-tap filter)						
−1/12	{ −4,	19,	254,	−19,	8,	−2	}
1/12	{ 4,	−16,	252,	22,	−8,	2	}
2/12	{ 6,	−28,	242,	48,	−17,	5	}
3/12	{ 9,	−37,	227,	75,	−25,	7	}
4/12	{ 11,	−42,	208,	103,	−33,	9	}
5/12	{ 12,	−44,	184,	132,	−39,	11	}
6/12	{ 11,	−43,	160,	160,	−43,	11	}
7/12	{ 11,	−39,	132,	184,	−44,	12	}
8/12	{ 9,	−33,	103,	208,	−42,	11	}
9/12	{ 7,	−25,	75,	227,	−37,	9	}
10/12	{ 5,	−17,	48,	242,	−28,	6	}

matrix whose element is given as

$$d_{ij} = \frac{1}{M}\cos\left(\frac{(2j-1+2M)i\pi}{4M}\right) \tag{2}$$

where $0 \le i \le (2M-1)$, $(-M+1) \le j \le M$, and $\mathbf{w}_{(\alpha)} = [w_0^{(\alpha)}, \ldots, w_{(2M-1)}^{(\alpha)}]^T$ is given as

$$w_i^{(\alpha)} = \begin{cases} \frac{1}{2}, & i = 0 \\ \cos\left(\frac{\pi(2\alpha-1+2M)i}{4M}\right), & 1 \le i \le (2M-1). \end{cases} \tag{3}$$

By using $\mathbf{f}_{(\alpha)}$ of (1), one can easily calculate the filter coefficients for an arbitrary number of filter taps ($2M$) and arbitrary fractional position α. Table I shows the coefficients of the 6-tap filters generated for 1/12-pel motion accuracy. Note that the filter coefficients are scaled by 256, H.264/AVC uses 5 bits for filter coefficients.

2) *Low Complexity Optimization:* An efficient hardware implementation requires multiplication-free representation for all the calculations. In order to minimize the number of arithmetical operations (shifts and additions), the filter coefficients in Table I can be slightly changed without significant loss in coding efficiency. Table II shows 5 bits representation for optimized filter coefficients for the quarter-pel positions, together with the required number of arithmetical operations. The proposed DCT-IF is a noncascaded interpolation scheme, i.e. only one spatial filtering is performed for all the possible positions. It can be implemented by a multiplication-free design. Thus, the 6-tap DCT-IF has the smaller number of operations compared to the interpolation filter of H.264/AVC since samples at the quarter-pel position can be generated by one 6-tap filtering operation whereas H.264/AVC uses a combination of 6-tap and bi-linear filters. The 12-tap DCT-IF filter requires twice as many operations as the 6-tap DCT-IF, however it provides similar performance to the well-known adaptive interpolation filter (AIF) scheme [19] while being less complex and more suitable for hardware implementation.

TABLE II
OPTIMIZED FILTER COEFFICIENTS FOR 6-TAP DCT-IF AT QUARTER-PEL POSITIONS

α	2M = 6 (6-tap filter)							Add	Shift
1/4	{ 1,	−4,	28,	9,	−3,	1	}	8	2
2/4	{ 1,	−5,	20,	20,	−5,	1	}	6	2
3/4	{ 1,	−3,	9,	28,	−4,	1	}	8	2

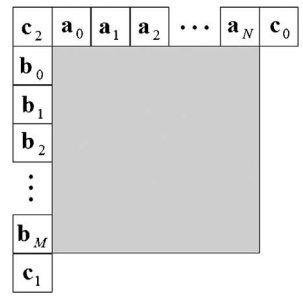


Fig. 3. Spatially adjacent motion vectors for the AMVP.

C. Advanced Motion Vector Prediction

In H.264/AVC standard, motion vectors are predicted from a median of the motion vectors of the spatially adjacent blocks [3]. To further improve the efficiency of the motion vector prediction, a few competition-based prediction methods have been proposed [16], [20], where the best predictor is selected from a given set through rate-distortion optimization.

This paper proposes the AMVP method, which is adapted to the proposed flexible unit representation. It allows the selection of the best predictor from the set which consists of three spatially adjacent motion vectors, their median, and a temporal motion vector.

Fig. 3 shows the spatially adjacent motion vectors which can be considered the candidate for the prediction. Let $A = \{\mathbf{a}_0, \mathbf{a}_1, \ldots, \mathbf{a}_N\}$, $A' \subset A$, and $I_{A'}$ denote the set of motion vectors above the current PU, the set of the available motion vectors in A, and the set which includes the indices of the elements in A', respectively. The set of the available motion vectors, A', is derived from A by verifying whether the corresponding reference index is the same as that of the current PU partition. When $A' = \emptyset$, the candidate motion vector $\mathbf{a}$ from A' is excluded from the candidate set, and a zero vector is used when taking the median of spatial motion vector predictors. Otherwise, $\mathbf{a}$ is determined as $\mathbf{a} = \mathbf{a}_n$, where n is the smallest element of $I_{A'}$. The candidate motion vectors $\mathbf{b}$ and $\mathbf{c}$ from $B = \{\mathbf{b}_0, \ldots, \mathbf{b}_M\}$ and $C = \{\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2\}$ are also determined using the same algorithm, and the resulting spatially adjacent motion vectors are used in the proposed AMVP method.

The overhead for signaling the index of the best predictor is reduced by re-organizing the set of candidate predictors. First, the predictors in the candidate set are reordered based on the PU splitting, so that the most probable motion predictor appears first in the set. Then, duplicate predictors and the predictors which can be excluded based on the parsed MVD values at the decoder side are eliminated from the set [20].

Nokia Exhibit 2006, p. 4
Amazon.com v. Nokia
IPR2024-01505

Note that, when only one predictor remains as a result of the re-organization, no overhead is signaled, since it can be also derived at the decoder.

Although AMVP is developed to predict the motion vector, it is also applied to the skip and direct modes, which do not transmit the additional motion vector differences. It should be noted that exactly the same process is applied to all cases to simplify the design. For comparison, H.264/AVC uses a slightly different scheme for the skip mode.

IV. Intraprediction

To improve the performance of intraprediction, five new intracoding tools are included. Arbitrary directional intra (ADI) extends the conventional nine-direction intraprediction to support any arbitrary number of angles in a unified way. Multiparameter intra (MPI) tries to apply post processing filters to the prediction samples from the ADI. To exploit the interchannel correlation between luminance and chrominance, color component correlation based prediction (CCCP) is introduced. Pixel-based template matching (PTM) is included to improve the coding efficiency for repeated regular patterns. Finally, the combined intraprediction (CIP) allows combining the closed-loop predictor generated from the above methods with an open-loop predictor to further improve the performance.

A. Arbitrary Directional Intra (ADI)

In H.264/AVC intracoding, nine intraprediction modes are supported for 4×4 and 8×8 blocks but only four intraprediction modes for 16×16 blocks. Intra-coded 16×16 blocks were intended to code homogeneous regions in low resolution video. However, even relatively large blocks ($\geq 16 \times 16$) cover only a small part of objects or background in high resolution video, which may contain strong directional patterns.

Arbitrary directional intra (ADI) is a coding tool designed to extend the directional intraprediction scheme used in H.264/AVC to give more accurate representation of directions and better suit larger prediction units. In the ADI scheme, up to 30 angles are generated using the integer pair (dx, dy) to represent the direction of lines which each mode uses for context pixel extrapolation as illustrated in Fig. 4.

Like H.264/AVC, ADI uses the reconstructed pixels. However, in some cases, the down-left region is available for the prediction process as depicted in Fig. 4. When the down-left region is not available, that region is generated using the nearest reconstructed pixel value.

Due to the large variety of the possible unit sizes, it is beneficial to adjust the number of angles according to the unit size. For example, PU partitions with 64×64 and 128×128 sizes are usually used for homogeneous areas which can be generated by using only some prediction modes, such as DC and plane modes. In this case, the number of supported prediction modes of the current PU partition may differ from that of the neighboring PU partitions. Intra prediction modes of the neighboring PU partitions are converted to the nine most frequently used modes to predict the current intraprediction mode in a unified way.

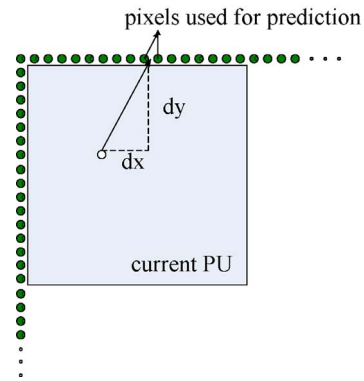


Fig. 4. Neighboring pixels and angles defined by (dx, dy).

B. Multiparameter Intra

Directional intraprediction sometimes results in nonsmooth and unnatural patterns which are not transform-friendly. This problem can be alleviated by applying a smoothing filter to the prediction signal obtained by the ADI scheme described in Section IV-A. Multiparameter intra (MPI) allows the use of the 4-point filter in (4) for each pixel to increase the probability of generating a prediction signal well-suited to the following transform stages. For each PU partition, mpi_flag is used to specify whether or not the MPI scheme is applied.

$$p'[x, y] = (p[x, y] + p[x-1, y] + p[x, y-1] + p[x, y+1] + 2\ ) >> 2 \quad (4)$$

where $p[x, y]$ and $p'[x, y]$ are the original and the smoothed prediction value at (x, y) coordinates, respectively, and $>>$ is the bit-wise shift to the right.

C. Color Component Correlation Based Prediction

There have been some studies on predicting chrominance pixels using the correlation with luminance [21]. Although most approaches try to predict the pixel values of one color channel by another color channel, the color component correlation based prediction (CCCP) scheme derives an object segmentation map of the chrominance channel from the already coded luma channel, and applies it to intraprediction.

The CCCP procedure is as follows. Firstly, the size of the reconstructed luma samples including the neighboring luma pixels are adjusted to match that of the chroma samples according to the given chroma format. Then, simple thresholding using the mean value is exploited to generate the binary segmentation map. After that, two mean values are computed from the neighboring chroma pixels according to the segmentation map and assigned to the inner part of the chroma block. Finally, a 3×3 averaging filter is applied to improve the smoothness of the prediction. Fig. 5 shows a brief flow of the CCCP process.

D. Pixel Based Template Matching (PTM)

Whilst directional extrapolation based intraprediction tools are very efficient for regions with directional patterns, they are not well-suited to regions with repeated regular patterns. To overcome this limitation, several intratechniques using some

Nokia Exhibit 2006, p. 5
Amazon.com v. Nokia
IPR2024-01505

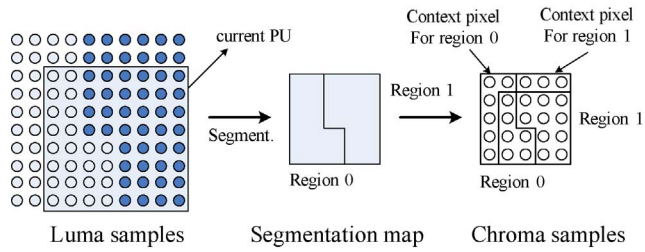


Fig. 5. Processing flow of color component correlation based prediction (CCCP).

form of template matching based on the already decoded pixels have been proposed [22], [23]. Although the reported schemes were proven to provide good coding gains for the targeted areas, a major drawback was the high complexity at the decoder side.

In this paper, a simple pixel based template matching (PTM) approach is proposed. PTM is designed to apply template matching at the pixel level to maximize the degrees of freedom. PTM uses only three pixels as template. Search candidates are also restricted to neighboring 27 pixels, which results in complexity reduction in matching process. Another difference of PTM compared with previous methods is that already prediction samples in the same prediction unit are also used as template and search candidate. This allows use very limited search area even for large block prediction.

E. Combined Intraprediction

Combined intraprediction (CIP) is a coding tool for providing improved prediction, especially in large blocks where a directional linear prediction may not work well, even with the large range of angles available to ADI. CIP predictions comprise a weighted combination of the prediction samples provided by other intracoding tools together with a pixel-by-pixel local mean prediction of pixels within the block to be transformed.

The local mean is constructed as the average of pixel values to the left, top-left and above the current pixel. In the decoder and within the local decoder in the encoder, these values are the reconstructed values after inverse quantization and transform. In the encoder, the reconstructed values are not available in the interior of the block because this would require quantization, transform and inverse transform to have been performed, and the pixels used for prediction lie within the block. Instead original values are used, although the encoder may select any suitable values. This local mean prediction is therefore different between forward prediction (encoder only) and reconstruction (encoder and decoder). Due to this property, CIP can be seen as a leaky prediction method to combine the conventional closed-loop predictor and an open-loop predictor. The advantage of CIP is that it provides pixel-by-pixel adaptation but is a simpler tool than, for example, local template matching approaches.

The combination factor is a 5-bit number between 0 and 31/32 which determines the proportion of local mean in the prediction. Since this number is less than 1, the prediction/reconstruction difference noise is damped, and pixel-by-pixel prediction adaptation is possible without noise blow-up. The ADI component provides an overall prediction direction and the local mean prediction provides local adaptation.

V. Spatial Transforms

It is well known that transforms larger than 8×8 provides better energy compaction and less quantization error for smooth and coarse textured video areas at the expense of computational complexity [24], [25]. In this paper, 16×16, 32×32 and 64×64 fast integer transforms are presented in addition to the conventional 4×4 and 8×8 transforms to support large TUs. In addition, a rotational transform (ROT) tool, which can adaptively compact the energy into low frequency components by rotating the transform basis, is also proposed as a second transform.

A. Large Integer Transform (16×16, 32×32, and 64×64)

The DCT is independent of input signal while it still provides near optimal decorrelation properties. Moreover, many fast DCT algorithms have been developed. All large transforms proposed in this paper are scaled integer DCTs based on Chen's fast factorizations [26]. Chen's fast DCT has a regular butterfly structure which is beneficial to hardware implementation.

Similarly to the H.264/AVC 4×4 core transform, each multiplication factor in the flow graph of the fast algorithm is scaled and approximated by some fixed precision values p, i.e., each constant is approximated by the dyadic rational of the form $q/2^p$, which can be implemented with only additions and bit-shifting operations. The precision value p can be chosen according to the dynamic range limitation and coding efficiency. As all multiplication factors are approximated, the core transform may not be truly orthogonal. However, this minor nonorthogonality does not result in any perceptible negative effect on the compression performance, whilst the complexity can be significantly reduced. Furthermore, it is still possible to achieve perfect reconstruction using a lifting scheme that can be applied additionally to the butterfly structure [27].

After all multiplication factors are determined, the forward 2-D transform is performed separately in vertical and horizontal direction as follows:

$$Y = C_f(C_f X^T)^T = C_f X C_f^T \tag{5}$$

where X is $N \times N$ residual data, C_f is the core N-point forward transform, and Y is $N \times N$ transformed coefficients. The calculation of $C_f X^T$ is performed from left to right in the flow graph of the fast algorithm. For the inverse transform, the computation order is reversed from right to left. Since the core transform C_f is scaled DCT, the remaining scaling factors should be integrated into the quantization stage.

B. Rotational Transform

While DCT is the most widely used block transform for video and image codecs, the DCT basis functions are not optimal for some types of residual signal. For example, a residual signal having strong diagonal components cannot

Nokia Exhibit 2006, p. 6
Amazon.com v. Nokia
IPR2024-01505

be represented efficiently with the DCT basis vectors. Typically, directional transform schemes are proposed to exploit this problem such as mode dependent directional transform (MDDT) [28], [29].

However, the direct use of directional transform is not appropriate to the proposed flexible unit definitions since it is difficult to design the directional transform by a fast butterfly structure such as DCT. In addition, the number of transforms would be almost doubled because both DCT and the directional transform for various TU sizes would be kept in the codec design.

In this paper, ROT is proposed as a second transform that could be applied after the core integer transform instead of introducing completely new transform cores. The main idea of ROT is to change the coordinate system of the transform basis, instead of direct rotation of the input residual. For example, the rotation matrices of horizontal and vertical directions for a 4×4 block with the rotation angles of $(\alpha_1, \alpha_2, \alpha_3)$ and $(\alpha_4, \alpha_5, \alpha_6)$ are defined as

$$R_{\text{ver}}(\alpha_1, \alpha_2, \alpha_3) = \begin{pmatrix} C_1C_3 - S_1C_2S_3 & -S_1C_3 - C_1C_2S_3 & S_2S_3 & 0 \\ C_1S_3 + S_1C_2C_3 & -S_1S_3 + C_1C_2C_3 & -S_2C_3 & 0 \\ S_1S_2 & C_1S_2 & C_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{6}$$

$$R_{\text{hor}}(\alpha_4, \alpha_5, \alpha_6) = \begin{pmatrix} C_4C_6 - S_4C_5S_6 & -S_4C_6 - C_4C_5S_6 & S_5S_6 & 0 \\ C_4S_6 + S_4C_5C_6 & -S_4S_6 + C_4C_5C_6 & -S_5C_6 & 0 \\ S_4S_5 & C_4S_5 & C_5 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{7}$$

where $C_k = \cos\alpha_k$ and $S_k = \sin\alpha_k$ for $k = 1, \ldots, 6$. The rotation matrices for 8×8 block can be defined similar to those for 4×4 block.

Each set of angles $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}$ specifies one element in the ROT dictionary. In ROT, only four out of all possible rotational angles sets are used. These RT matrices minimize the discrepancy between forward-inverse RT in integer arithmetic realization and provide better energy compaction. These sets of rotational angles were selected using the training with VCEG test sequences, and they provide similar performance for other video sequences including the test sequences for the HEVC standardization. This proves that these angles sets capture and exploit general characteristic of natural images. The index of rotational angles in the dictionary is signaled to the decoder for each CU explicitly.

Furthermore, since ROT is a second transform, not all of the DCT coefficients are processed. For TUs larger than 4×4, only 8×8 low-frequency areas are processed. It has been found that this restriction does not degrade the coding efficiency since most of the coefficients are already compacted into these low-frequency areas.

The final 2-D spatial transform process when the ROT is used can be summarized as

$$Y = R_{\text{ver}}(C_f X C_f^T) R_{\text{hor}} \tag{8}$$

where X is residual data, C_f is the core forward transform, and Y is output coefficients.

VI. In-Loop Filtering

After reconstruction, several in-loop filters are applied to reduce the distortion between the original and the reconstructed video frames. In addition to a modified H.264/AVC deblocking filter for the CU/PU/TU structure, a CU-synchronized adaptive loop filter (CU-ALF) is introduced to reduce the average distortion of the reconstructed frame. Extreme correction (EXC) and band correction (BDC) techniques are applied to reduce the distortion further by considering object-based pixel classifications. Finally, content adaptive dynamic range (CADR) is proposed to reduce the performance loss from the rounding effects of the various internal processes.

A. Deblocking Filter

The same process as used in H.264/AVC is employed for the deblocking filter. However, the use of a large size transform can effectively reduce the complexity of the deblocking filter, by reducing the number of deblocked pixels. It was reported that the number of deblocked pixels was reduced by 43% on average [14] for the 1080p sequences used in [30].

B. CU-Synchronized Adaptive Loop Filter (CU-ALF)

QALF [13] is known to provide significant coding gain by effectively combining both the Wiener filter approach and the quadtree based on/off selections. In QALF, along with filter coefficients, the information representing the quadtree structure is coded independently with the prediction quadtree structure, which results in large amount of side information in the slice header.

Since the proposed unit representations provides a generic quadtree structure similar to the original design of QALF, this information can be reutilized for the adaptive loop filter process. CU-synchronized adaptive loop filter (CU-ALF) is based on the QALF approach; however, it uses the already coded CU hierarchy as the on/off control for the adaptive loop filter process instead of coding separate quadtree information in the slice header.

It has been reported that the proposed CU-ALF provides comparable coding gain to QALF, while eliminating the need for any side information related to the quadtree partition in the slice header and reducing the encoder complexity [14].

C. Extreme Correction and Band Correction

EXC is designed to compensate the distortion according to the identified pixel classes. Each pixel is categorized as local maxima, local minima, or object edge by simple computation based on the four neighboring pixel values: left, above, right, and bottom. For each pixel class, the mean difference between the original and the reconstructed signals is computed and transmitted to the decoder. A fuller description is given in [14].

BDC is another form of pixel classification, based on pixel value ranges. The main motivation is to equalize the different probability distribution functions of the reconstructed signal with the original. In this paper, up to 16 value ranges are defined; the mean difference for each band is computed and transmitted to the decoder.

Nokia Exhibit 2006, p. 7
Amazon.com v. Nokia
IPR2024-01505

D. Content Adaptive Dynamic Range

Internal bit depth increase (IBDI) is a technique to reduce the rounding error by enlarging the internal data precision [31]. Typically, IBDI is implemented by increasing the source bit-depth, which makes the encoder implementation more difficult. CADR is related to IBDI, but is a more general form of source scaling. Using the CADR, a pixel value x is adaptively scaled to $F(x)$ as follows:

$$F(x) = (2^{\text{bit}} - 1)(x - \text{Min})/(\text{Max} - \text{Min}) \tag{9}$$

where bit, Max, and Min are the bit-depth, the maximum value, and the minimum value of the input source, respectively. It should be noted that the parameters Min and Max can be adjusted at sequence, slice, or even block level.

CADR can be used to reduce the rounding error without requiring increased bit-depth in the encoder. For example, source material that is known to be constrained by the [16, 235] safe range of BT.709 can be mapped to [0, 255] to reduce rounding errors in the internal representation.

VII. Entropy Coding

In this paper, syntax-based context-adaptive binary arithmetic coder (SBAC) is used for the entropy coding of syntax elements. Additionally, adaptive coefficient scanning (ACS), which provides multiple scanning patterns for the transform coefficients, is also proposed to improve the coding efficiency while reducing the complexity.

A. Syntax-Based Context-Adaptive Binary Arithmetic Coding

As with the context-adaptive binary arithmetic coding (CABAC) of H.264/AVC [32], the SBAC consists of four major steps: syntax-based binarization, context modeling, probability estimation, and binary arithmetic coding. The syntax-based binarization and the context modeling are adapted to the syntax elements obtained from the flexible unit representation and the extended coding tools. For example, the assignment of context models for the syntax elements related to transform coefficients (e.g. significance map, coefficient level, etc.) is dependent on the hierarchical depth of the CU. The probability estimation and binary arithmetic coding are based on an improved version of the arithmetic coding method specified in Annex D of the JPEG still image coding standard [33]. Specifically, the table-based probability update and the multiplication-free design are improved, and the bypass mode for equiprobable binary symbols is added to reduce complexity [14].

B. Adaptive Coefficient Scanning

Since large transforms frequently have very sparse coefficient distributions, a fixed zigzag scanning method may result in long coefficient scanning lists including many zeroes, resulting in inefficient coding. ACS that adaptively chooses the best scanning method for each TU is used to avoid this problem.

Given the quantized transform coefficients, the best scanning pattern is chosen from conventional zigzag, horizontal, and vertical scanning patterns, and then its index is explicitly signaled in each TU. Note that the ACS index is signaled only when nonzero AC coefficients exist in the TU, since there is no difference between the patterns when the TU includes only the DC coefficient.

When the horizontal or vertical scanning methods are chosen as the best scanning method, the mapping process from the coefficient array indices to the scanning order indices is not necessary. Thus, the complexity for the coefficient indexing can be reduced and also the memory bandwidth to access the zigzag mapping table can be saved.

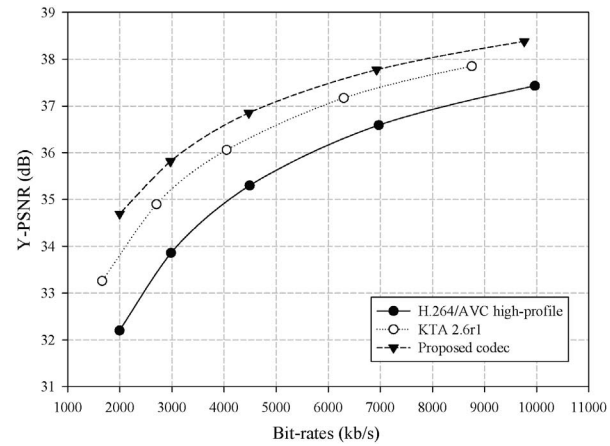


Fig. 6. Rate distortion curve of BasketballDrive 1080p sequence.

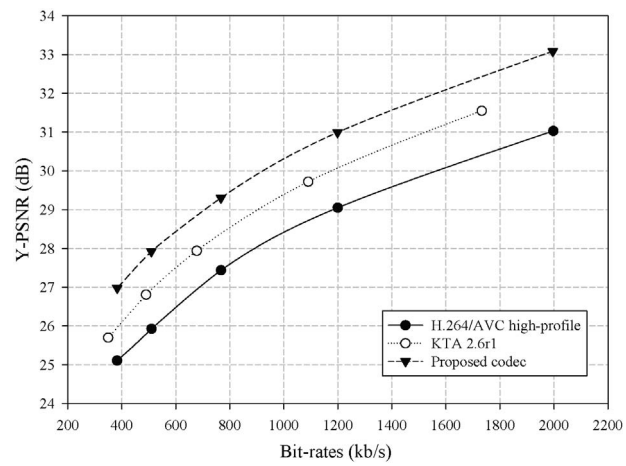


Fig. 7. Rate distortion curve of PartyScene WVGA sequence.

VIII. Performance Analysis

To verify the performance of the proposed coding tools, they were implemented into one complete video codec software which is available at [14]. This section provides both objective performance based on the peak signal-to-noise ratio (PSNR) measure and the subjective performance based on the mean opinion score (MOS) values for the proposed video codec.

A. Objective Compression Efficiency

1) *Overall Performance:* The coding efficiency of the proposed system is verified using two sets of coding configurations defined in the Call for Proposals [30]—constraint

Nokia Exhibit 2006, p. 8
Amazon.com v. Nokia
IPR2024-01505

set 1 (CS1) and constraint set 2 (CS2) which correspond to a random access case and a low delay case, respectively. Figs. 6 and 7 show typical examples of rate-distortion curves when video sequences are coded using CS1 configuration. As shown in the figures, the proposed codec consistently showed significant coding benefits over H.264/AVC High Profile. For comparison with other recent advances in the video coding area, the results of the key technology area (KTA) 2.6r1 [34] with the same test configurations are also included. This test configuration is same as in [35], where all major tools of KTA including enhanced AIF, MV competition, MDDT, high precision filter, ALF, and extended block size are enabled. It should be noted that the improved coding efficiency can be observed with both high resolution and low resolution video material.

Table III summarizes the performance of the proposed codec relative to the H.264/AVC High Profile. The average compression performance improvement over all sequences is 39.49% in CS1 and 39.48% in CS2, respectively. Class B (1080p resolution) achieves 43.29% (CS1) and 43.66% (CS2) bit saving on average, with BQTerrace as the best sequence in both configurations, showing 54.06% (CS1) and 60.62% (CS2) performance improvement, respectively. RaceHorses in Class D (WQVGA resolution) shows the lowest performance improvement in both configurations, at 27.08% in CS1 and 21.59% in CS2, respectively. The performance summarized in Table III shows that the proposed codec consistently achieves the excellent coding performance regardless of the resolution of input video sequences and the coding configuration.

2) *Performance of Each Set of Coding Tools:* It is hard to evaluate the performance of each individual coding tool, since the coding tools interact with each others and the tool-wise evaluation cannot measure the performance improvement introduced by the interaction. Therefore, the proposed coding tools are categorized to five sets—large transform (16×16, 32×32, and 64×64 transforms), AMVP, inter (DCT-IF and HAM), intra (ROT, MPI, ADI, CCCP, TMI, and CIP), and in-loop (EXC, ALF, and CADR)—and the performance of each set of coding tools is evaluated by measuring the performance loss introduced by turning off the tools in the corresponding set. The performance of the large motion partition is not evaluated since its performance has been already verified in the literatures [6]–[8]. Table IV summarizes the performance of each set of coding tools and the alternative setting which is enabled when the corresponding tool set is turned off. Note that only 1-second tests using CS1 configuration were carried out to reduce the experiment time. The results show that all the sets of coding tools contribute to the overall performance improvement almost equally.

TABLE III
SUMMARIZED BD-RATE REDUCTIONS FOR VARIOUS RESOLUTIONS

Class (Size)	Sequence Name	CS1 (%) BD-Rate	CS1 (%) Average	CS2 (%) BD-Rate	CS2 (%) Average
A (4K)	*Traffic*	40.59	35.21	N/A	N/A
	PeopleOnStreet	29.84		N/A	
B (1080p)	*Kimono*	44.42	43.29	44.45	43.66
	ParkScene	32.96		31.18	
	Cactus	40.78		34.74	
	BasketballDrive	44.21		47.29	
	BQTerrace	54.06		60.62	
C (WVGA)	*BasketballDrill*	43.09	39.94	36.82	35.58
	BQMall	39.69		37.06	
	PartyScene	41.13		40.45	
	RaceHorses	35.86		27.99	
D (WQVGA)	*BasketballPass*	30.55	36.45	26.93	32.16
	BQSquare	54.19		51.87	
	BlowingBubbles	33.97		28.24	
	RaceHorses	27.08		21.59	
E (720p)	*Vidyo 1*	N/A	N/A	48.38	47.48
	Vidyo 3	N/A		46.26	
	Vidyo 4	N/A		47.79	
Total average		39.49		39.48	

TABLE IV
PERFORMANCE OF EACH SET OF CODING TOOLS

Tool Set	Alternative Setting	Class (%) A	B	C	D	Average
Large tr.	Max tr. size = 8×8	2.2	7.2	1.9	1.0	3.5
AMVP	Median predictor [3]	4.4	3.7	2.9	2.4	3.2
Inter	6-tap/bi-linear filter [3]	0.4	1.3	4.2	7.6	3.6
Intra	Directional intra [3]	3.5	3.3	4.4	2.4	3.4
In-loop	Deblocking only	8.8	6.1	3.6	2.9	4.3

B. Subjective Compression Efficiency

A large scale subjective test was carried out for all the proposals submitted to the Call for Proposals. In this test, it was shown that the proposed codec provided excellent subjective improvements over H.264/AVC in all configurations and it gave one of the best overall results of all of the proposals that were submitted, based on MOS values. The official report of subjective test results can be found in [36].

Table V shows several pairs of MOS values for H.264/AVC High Profile and the proposed codec for the five 1080p sequences. Each pair has been selected to give similar MOS values, to verify the bit-rate reduction that the proposed codec provides for similar subjective visual quality. As shown in the table, the proposed codec operating at an average of 2.4 Mbit/s provides similar visual quality to H.264/AVC High Profile operating at an average of 6.4 Mbit/s. This corresponds to approximately 60% bit-rate savings, indicating that the proposed codec provides even greater improvement in subjective visual quality than would be expected from objective metrics such as PSNR.

Nokia Exhibit 2006, p. 9
Amazon.com v. Nokia
IPR2024-01505

TABLE V
SUMMARIZED MOS VALUES FOR 1080P SEQUENCES

Sequence	H.264/AVC		Proposed		
	Bit rates (kb/s)	MOS	Bit rates (kb/s)	MOS	Reduction (%)
Kimono	4000	8.86	1600	8.56	60.0
ParkScene	4000	7.41	1600	7.78	60.0
Cactus	7000	8.61	3000	8.24	57.1
BasketballDrive	7000	7.44	3000	8.17	57.1
BQTerrace	10 000	8.77	3000	9.17	70.0
Average	6400	8.22	2440	8.38	60.9

IX. CONCLUSION

In this paper, a new flexible unit representation composed of CU, PU, and TU was proposed. The proposed scheme provides a generic quadtree structure while defining the relationship between region splitting, prediction method, and transform sizes with minimal signaling overhead.

In addition, various other coding tools were extended in a very consistent way according to the proposed unit representation. For example, the number of directions in the directional intraprediction scheme was extended to arbitrary angles and the supported transform size was extended up to 64×64.

By combining other novel coding tools designed under the proposed unit representation, the proposed codec showed about 40% performance improvement over the state-of-the-art H.264/AVC standard based on objective PSNR measurements. The large-scale subjective testing results show that an average of about 60% coding gain can be achieved for the 1080p sequences.

The video codec described in this paper was a candidate in the competitive phase of the HEVC standardization work and it has been partially adopted into the first standardization model of the collaborative phase of the HEVC effort.

ACKNOWLEDGMENT

The authors would like to thank their colleagues including, but not limited to, H. K. Jung, J. C. Lee, S. R. Lee, D. Y. Kim, K. H. Lee, Y. M. Sohn, Y. Y. Lee, B. K. Lee, S. M. Park, H. S. Song, J. B. Choi, B. D. Choi, C. H. Lee, S. S. Jung, and D. S. Cho for their invaluable contributions and continuous comments on this paper.

REFERENCES

[1] *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbit/s: Video*, ISO/IEC 11172-2, ISO/IEC JTC1/SC29/WG11, Aug. 1993.

[2] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, ISO/IEC 13818-2, ISO/IEC JTC1/SC29/WG11, 1996.

[3] *Final Draft International Standard of Joint Video Specification*, ITU-T Rec. H.264, ISO/IEC 14496-10, ISO/IEC JTC1/SC29/WG11, Mar. 2003.

[4] K. H. Lee, E. Alshina, J. H. Park, W. J. Han, and J. H. Min, "Technical considerations for ad hoc group on new challenges in video coding standardization," in *Proc. 85th MPEG Meeting*, Jul. 2008, no. M15580.

[5] E. Alshina, K. H. Lee, W. J. Han, and J. H. Park, "Technical considerations for ad hoc group on new challenges in video coding standardization," in *Proc. 86th MPEG Meeting*, Oct. 2008, no. M15899.

[6] P. Chen, Y. Ye, and M. Karczewicz, "Video coding using extended block sizes," in *Proc. 36th VCEG Meeting*, Oct. 2008, no. VCEG-AJ23.

[7] T. Yoshino, S. Naito, and S. Sakazawa, "Preliminary response for draft call for evidence on high performance video coding," in *Proc. 87th MPEG Meeting*, Feb. 2009, no. M16082.

[8] S. Sekiguchi and S. Yamagishi, "On coding efficiency with extended block size for UHDTV," in *Proc. 87th MCEG Meeting*, Feb. 2009, no. M16019.

[9] A. Puri, H. M. Hang, and D. Schilling, "An efficient block-matching algorithm for motion-compensated coding," in *Proc. ICASSP*, vol. 12. Aug. 1997, pp. 1063–1066.

[10] M. H. Chan, Y. B. Yu, and A. G. Constantinides, "Variable size block matching motion compensation with applications to video coding," *Commun. Speech Vis. IEE Proc. I*, vol. 137, no. 4, pp. 205–212, Aug. 1990.

[11] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 327–331, May 1994.

[12] J. J. Zhang, M. O. Ahmad, and M. N. Swamy, "Quadtree structured region-wise motion compensation for video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 5, pp. 808–822, Aug. 1999.

[13] T. Chujoh, N. Wada, and G. Yasuda, *Quadtree-Based Adaptive Loop Filter*, document VCEG-C181, ITU-T Study Group 16, Jan. 2009.

[14] K. McCann, W. J. Han, I. K. Kim, J. H. Min, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, M. S. Cheon, and N. Shlyakhov, "Samsung's response to the call for proposals on video compression technology," in *Proc. 1st JCT-VC Meeting*, Apr. 2010, no. JCTVC-A124.

[15] JCT of ISO/IEC MPEG & ITU-T VCEG, "Test model under consideration," in *Proc. 1st JCT-VC Meeting*, Apr. 2010, no. JCTVC-A205.

[16] J. Jung and G. Laroche, "Competition-based scheme for motion vector selection and coding," in *Proc. 29th VCEG Meeting*, Jul. 2006, no. VCEG-AC06.

[17] T. Wedi, "Hybrid video coding based on high-resolution displacement vectors," in *Proc. VCIP*, Jan. 2001, pp. 186–193.

[18] C. C. Tseng and S. L. Lee, "Design of fractional delay FIR filter using discrete cosine transform," in *Proc. APCCAS*, Nov. 2008, pp. 858–861.

[19] T. Wedi and H. G. Musmann, "Motion- and aliasing-compensated prediction for hybrid video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 577–586, Jul. 2003.

[20] S.-D. Kim and J. B. Ra, "An efficient motion vector coding scheme based on minimum bitrate prediction," *IEEE Trans. Image Process.*, vol. 8, no. 8, pp. 1117–1120, Aug. 1999.

[21] S. H. Lee and N. I. Cho, "Intra prediction method based on the linear relationship between the channels for YUV 4:2:0 intra coding," in *Proc. ICIP*, 2009, pp. 1037–1040.

[22] T. K. Tan, C. S. Boon, and Y. Sujuki, "Intra prediction by template matching," in *Proc. ICIP*, 2006, pp. 1693–1696.

[23] J. Balle and M. Wien, "Extended texture prediction for H.264/AVC intra coding," in *Proc. ICIP*, 2007, pp. VI-93–VI-96.

[24] S. Ma and C.-C. J. Kuo, "High-definition video coding with super-macroblocks," in *Proc. VCIP*, Jan. 2007.

[25] J. Dong, K. N. Ngan, C. K. Fong, and W. K. Cham, "2-D order-16 integer transforms for HD video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 10, pp. 1462–1474, Oct. 2009.

[26] W.-H. Chen, C. H. Smith, and S. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. Commun.*, vol. 25, no. 9, pp. 1004–1009, Sep. 1977.

[27] J. Liang and T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme," *IEEE Trans. Signal Process.*, vol. 49, no. 12, pp. 3032–3044, Dec. 2001.

[28] Y. Ye and M. Karczewicz, *Improved Intra Coding*, Contribution C257, ITU-T Q.6/SG16, Geneva, Switzerland, Jun. 2007.

[29] Y. Ye and M. Karczewicz, *Improved Intra Coding*, VCEG Contribution VCEG-AG11, no. 20, Shenzhen, China, Oct. 2007.

[30] ISO/IEC JTC1/SC29/WG11, "Joint call for proposals on video compression technology," in *Proc. 91st MPEG Meeting*, Apr. 2010, no. N11113.

[31] T. Chujoh and R. Noda, "Internal bit depth increase for coding efficiency," in *Proc. 31st VCEG Meeting*, Jan. 2007, no. VCEG-AE13.

Nokia Exhibit 2006, p. 10
Amazon.com v. Nokia
IPR2024-01505

[32] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620–636, Jul. 2003.

[33] *Coding of Still Pictures*, ISO/IEC IS 10918-1, ITU-T Rec. T.81, ISO/IEC JTC1/SC29/WG1, Sep. 1993.

[34] ITU-T VCEG. (2010). *Key Technology Area (KTA) Software* [Online]. Available: http://iphome.hhi.de/suehring/tml/download/KTA

[35] T. K. Tan, G. Sullivan, and T. Wedi, *Recommended Simulation Common Condition for Coding Efficiency Experiments Revision 1*, VCEG Contribution VCEG-AJ10, Jul. 2008.

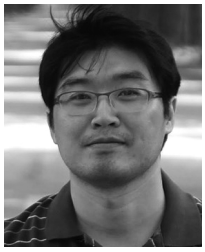
[36] V. Baroncini, J. R. Ohm, and G. Sullivan, "Report of subjective test results of responses to the joint call for proposals (cfp) on video coding technology for high efficiency video coding (HEVC)," in *Proc. 1st JCT-VC Meeting*, Apr. 2010, no. JCTVC-A204.

Woo-Jin Han (M'02) received the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1997 and 2002, respectively.

He is currently a Principal Engineer with the Multimedia Platform Laboratory, Digital Media and Communication Research and Development Center, Samsung Electronics, Suwon, Korea. Since 2002, he has contributed successfully to the ISO/IEC Moving Pictures Experts Group, Joint Video Team, and Joint Collaborative Team standardization efforts. His current research interests include high-efficiency video compression techniques, scalable video coding, multi-view synthesis, and visual contents understanding.

Junghye Min received the B.S. degree in mathematics from Ewha Women's University, Seoul, Korea, in 1995, and the M.E. and Ph.D. degrees in computer science and engineering from Pennsylvania State University, University Park, in 2003 and 2005, respectively.

From 1995 to 1999, she was a S/W Engineer with Telecommunication Systems Business, Samsung Electronics, Suwon, Korea. She is currently a Principal Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics. Her current research interests include video content analysis, motion tracking, pattern recognition, and video coding.



Il-Koo Kim received the M.S. and Ph.D. degrees from the School of Electrical Engineering, Seoul National University, Seoul, Korea, in 2001 and 2006, respectively.

After joining the Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea, in 2006, he has worked to develop high efficiency video codec. His current research interests include video compression and transmission.

Elena Alshina received the M.S. degree in physics from Lomonosov Moscow State University (MSU), Moscow, Russia, in 1995. She continued her postgraduate study at the Faculty of Computational Mathematics and Cybernetics, MSU, and received the Ph.D. degree in mathematical modeling in 1998.

Since that time, she has been with the Institute for Mathematical Modeling, Russian Academy of Sciences, Moscow, performing research on high accuracy numerical methods and new finite-difference scheme development. Simultaneously, she was an Associate Professor for the Moscow Institute of Electronic Technology, Moscow. In 2006, she joined Samsung Electronics (first at the Moscow Research Center, before moving to Korea). Currently, she is a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea, where she makes high efficiency video codecs.

Alexander Alshin received the M.S. and Ph.D. degrees in mathematical physics from Moscow State University (MSU), Moscow, Russia, in 1995 and 1998, respectively.

From February 1998 to January 2006, he was a Senior Researcher with the Physical Department, MSU. Since 2006, he has been with Samsung Electronics, Suwon, Korea, and currently works with the Multimedia Platform Laboratory, DMC Research and Development Center, Samsung Electronics, as a Senior Engineer. His current research interests include theory of non-classical PDEs, nonlinear functional analysis, applied mathematics, numerical analysis, and their applications to video compression.

Tammy Lee received the B.S. degree in mathematics education from Ewha Women's University, Seoul, Korea, in 2000, and the M.S. degree in computer science from New York University, New York, in 2005.

She is currently a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. Her current research interests include image and video compression.

Jianle Chen received the B.S. and Ph.D. degrees in information and communication system from Zhejiang University, Hangzhou, China, in 2001 and 2006, respectively.

Currently, he is a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. His current research interests include video/image compression, computer vision, and image/video processing.

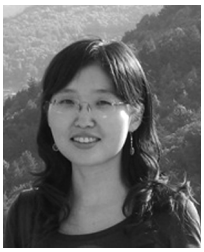
Vadim Seregin received the M.S. (honors) degree in physics from Moscow State University (MSU), Moscow, Russia, and the Ph.D. degrees in mathematical modeling, and numerical methods and software systems from MSU, in 2002 and 2005 respectively.

Currently, he is a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. His current research interests include quasi-gasdynamics, numerical methods, and the field of image and video processing with a focus on video compression.

Sunil Lee (S'02–M'08) received the B.S. degree in electrical engineering from Yonsei University, Seoul, Korea, in 2001, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2002 and 2008, respectively.

From March 2008 to August 2008, he was a Post-Doctoral Researcher with the Division of Electrical Engineering, School of Electrical Engineering and Computer Science, KAIST. He is currently a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. His current research interests include multimedia retrieval, multimedia security, digital watermarking, multirate signal processing, and video coding.

Nokia Exhibit 2006, p. 11
Amazon.com v. Nokia
IPR2024-01505



Yoon Mi Hong received the B.S. degree in mathematics from Yonsei University, Seoul, Korea, in 2003, and the M.S. degree in applied mathematics and the Ph.D. degree in mathematical sciences from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2005 and 2009, respectively.

She is currently a Senior Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. Her current research interests include time-frequency analysis, sampling theory, and its applications in signal processing, as well as video coding.

Min-Su Cheon received the B.S. and M.S. degrees in electrical engineering from Kyunghee University, Suwon, Korea, in 2005 and 2007, respectively.

He is currently an Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon. His current research interests include high-efficient video coding, multiview video coding, 3-D video coding, scalable video coding, and bioimage processing.

Nikolay Shlyakhov received the M.S. degree in applied mathematics from the Moscow Institute of Physics and Technology (MIPT), Moscow, Russia, in 1999. From 1999 to 2007, he was pursuing the Ph.D. degree from the Institute for Mathematical Modeling of the Russian Academy of Sciences, Moscow.

From 2003 to 2007, he was an Assistant Professor with the Department of Higher Mathematics, MIPT. Since 2007, he has been an Engineer with the Multimedia Platform Laboratory, Digital Media and Communications Research and Development Center, Samsung Electronics, Suwon, Korea. His current research interests include numerical mathematics, theory of approximation, and video coding.

Ken McCann graduated from Glasgow University, Glasgow, U.K., with the B.S. degree in natural philosophy in 1979. He received the M.S. degree in systems engineering from Surrey University, Surrey, U.K., in 1982.

He is currently a Director and Co-Founder of ZetaCast, Bath, U.K., an independent U.K.-based technology consultancy company specializing in digital TV and related systems. Prior to founding ZetaCast, he was the Technology Director with NTL Broadcast (now Arqiva), the main provider of broadcasting infrastructure in the U.K. Before that, he was with Symbionics Instruments, Cambridge, U.K., the Independent Broadcasting Authority and Philips Research Laboratories. He contributed significantly to the development of the MPEG-2 standard, has chaired the DVB technical group responsible for audio-visual coding specification since its inception over 15 years ago, and is currently an active member of the Joint Collaborative Team on Video Coding, working on the new high efficiency video coding standard.

Mr. McCann has been a member of the IET in the U.K. (and its predecessor, the IEE) since 1982.

Thomas Davies graduated from Oxford University, Oxford, U.K., with the M.A. degree in mathematics and philosophy in 1991. He received the M.S. and Ph.D. degrees in mathematics from Warwick University, Coventry, U.K., in 1992 and 1996, respectively.

He is currently a Lead Research and Development Engineer with Research and Development, British Broadcasting Corporation, London, U.K., having joined in 2000. He specializes in video compression, and image and signal processing, and was responsible for developing the open source Dirac video compression algorithms, the intra coding part of which is standardized as SMPTE VC-2. He is actively participating in the development of the MPEG high efficiency video coding standard under the auspices of the Joint Collaborative Team on Video Coding. Prior to joining the BBC, he was a Technology Consultant in systems engineering and satellite networking.

Jeong-Hoon Park received the B.S. and M.S degrees in electrical engineering from Hanyang University, Seoul, Korea, in 1991 and 1994, respectively.

He is currently a Principal Engineer with the Multimedia Platform Laboratory, Digital Media and Communication Research and Development Center, Samsung Electronics, Suwon, Korea. In 1998, he was a Visiting Scholar with the University of California, Los Angeles, where he researched error resilient video compression and transmission. Since then, he has contributed to the ITU-T Video Coding Experts Group and ISO/IEC Moving Pictures Experts Group. From 2002 to 2004, he has been involved in developing mobile broadcasting technology as an active member of WorldDMB forum. His current research interests include audio and video compression technology, medical image processing, and multimedia systems including video broadcasting and communications.

Nokia Exhibit 2006, p. 12
Amazon.com v. Nokia
IPR2024-01505