ON SECONDARY TRANSFORMS FOR INTRA PREDICTION RESIDUAL

Ankur Saxena and Felix C. Fernandes

Samsung Telecommunications America
1301 E. Lookout Drive, Richardson, TX -75082, USA
Email: {asaxena,ffernandes}@sta.samsung.com

ABSTRACT

In this paper, we present a secondary transform that is applied in a codec after the conventional DCT for all the video-coding intra prediction modes. Our approach is applicable to any block-based intra prediction scheme that employs transforms along the horizontal and vertical direction separably. The secondary transform is applied to the lower 8x8 frequency coefficients of the output of conventional DCT at block sizes 8x8 and higher. The proposed transform scheme has low complexity as it is applied only to the top-left portion of DCT output, especially in the context of large blocks such as 32x32 where an alternate transform of size 32x32 other than DCT would be expensive to be implemented in hardware. The proposed technique works in single-pass, and the choice of when to use the secondary transform is solely based on the intra prediction mode and requires no additional signaling information or R-D search. Our simulation results show that the proposed transform scheme provides significant BD-Rate improvement over the conventional DCT-based coding scheme for intra prediction of video sequences in the ongoing HEVC standardization.

Index Terms— Video coding, secondary transform, DCT, compression, HEVC.

1. INTRODUCTION

Most image and video-coding standards such as JPEG, H.264/AVC, VC-1, and the ongoing HEVC standardization employ block-based transform coding as a framework for efficient image and video compression. The pixel domain data is transformed to frequency domain using a transform process on a block-by-block basis. For typical images, most of the energy is concentrated in low-frequency transform coefficients. Following the transform, a relatively large step-size quantizer can be used for high-frequency transform coefficients in order to compact energy more efficiently and to attain better compression. Hence it is required to devise the optimal transform for each image block to fully decorrelate the transform coefficients. The Karhunen-Loeve Transform (KLT) is known to be optimal under certain conditions for transform coefficients. However, practical use of KLT is limited due to its high computational complexity and it has been shown in [9] that DCT provides an attractive alternative to KLT in terms of energy compaction and performance close to KLT. But with the advent of intra prediction (i.e., prediction within the same image), this is no longer the case and the optimal transform should adapt to the prediction residue characteristics. In the HM reference software [3] for High-Efficiency Video Coding (HEVC) standardization, after intra prediction, various mode-dependent intra prediction transforms such as mode-dependent directional transform (MDDT) [18] were tested. However, the MDDT requires 9 different transforms along each of the horizontal and vertical directions at a block-size, and it would therefore require multiple number of transform cores in hardware at a particular block size. Furthermore MDDTs were derived using training-based residues and the actual coefficients in the transform matrix are dependent on the video sequences used for training, as well as other techniques used for intra prediction etc. Han, Saxena & Rose in [6] analytically derived DST Type-7 with frequency and phase components different from the conventional DCT to be the actual KLT along the prediction direction for intra modes in H.264/AVC. They also showed that if prediction is not performed along a particular direction, then DCT performs close to KLT. The idea was applied to the vertical and horizontal modes in intra-prediction in H.264/AVC and a combination of the DST and conventional DCT was used for the vertical and horizontal modes. More recently, in the current HEVC standardization, the mode-dependent DCT/DST scheme in [14] was adopted at size 4x4. However, the gains by applying the DCT/DST transform scheme to square blocks of larger sizes such as 8x8, 16x16 and 32x32 as presented in [10], [13] are relatively less as compared to 4x4 DCT/DST scheme and may not justify the introduction of additional DST transform at larger block sizes, such as 32x32 from the hardware complexity viewpoint. To avoid using new large transforms, and motivated by the fact that most of the signal energy is concentrated only in the low-frequency transform coefficients, secondary transforms which are applied to the low-frequency coefficients of DCT are being currently investigated in the HEVC standardization [5].

Prominent among the secondary transforms is the Rotational Transforms (ROT) [1], [7], wherein a 8x8 secondary transform is chosen from a dictionary of 4 sets of 8x8 transforms by determining the best Rate-Distortion (R-D) cost for all the 5 choices (4 transform, and 1 no transform case, when DCT alone provides better R-D cost) during encoding. The 8x8 ROT is applied as a secondary transform to the top 8x8 low-frequency coefficients of the DCT output at blocks 8x8, 16x16 and 32x32 (see Fig. 1). The choice of ROT transform is sent via additional bits in the video bitstream. Though closer to optimality in R-D sense, the R-D computation process in ROT causes a lot of encoding time overhead as we will see in Results section. In fact, the information in the intra prediction mode, which is already sent in the encoded video bit-stream can be efficiently used to indicate whether a secondary transform is required. In this paper, we present an analytically derived 8x8 "mode-dependent" secondary transform (implying no signaling information is required), that can be applied for all the modes in unified intra prediction [8] in HEVC on blocks of size 8x8 and larger. No R-D search is performed for the proposed secondary transform, and this results in almost no increase in the encoding time.

The rest of the paper is organized as follows: Sec. 2 reviews unified intra prediction in the current HEVC standardization. The derivation of the proposed mode-dependent secondary transform is outlined in Sec. 3. followed by experimental results in Sec. 4 and Conclusions in Sec. 5.

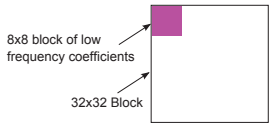

Fig. 1. Low frequency components of a DCT transformed output

2. UNIFIED INTRA PREDICTION

The ongoing HEVC standardization uses unified intra prediction in which up to 34 different directional intra-prediction modes at a particular block size are defined. These 34 directional intra-prediction modes can be divided into 3 categories as follows:

1. Category 1 oblique modes (Fig. 2): Here prediction is performed from the decoded pixels from either the top row or the left column. The vertical mode and the horizontal mode in [8] are special cases of this oblique mode when prediction direction is vertical or horizontal respectively.
2. Category 2 oblique modes (Fig. 2): Here prediction is performed from both the top row and the left column pixels.
3. DC mode: Here prediction is performed from an average of all available decoded pixels similar to H.264/AVC [17].

We next outline the derivation of the secondary transform for the vertical mode, which is a special case of Category 1 mode.

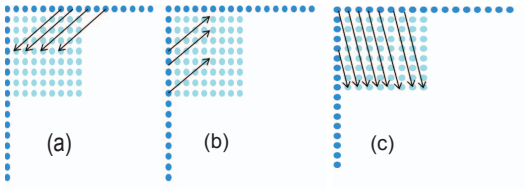

Fig. 2. Examples of Category 1 oblique modes are shown in (a) and (b) where prediction is performed from one direction only: either the top row or left column. (c) shows example of Category 2 oblique mode where prediction is performed from both directions: top row and left column

3. DERIVATION OF SECONDARY TRANSFORM

3.1. Analysis for Vertical Mode

Consider a vertical intra prediction mode in Fig. 3. Here prediction is performed from the top row in the vertical direction. After intra prediction, we need to take the vertical (column) transform of each of the column, and horizontal (row) transform of each row. In [6],[12] the following Gauss-Markov model for the pixels was assumed in the context of 1-d: $x_k = \rho x_{k-1} + e_k$, where ρ is the correlation coefficient between the pixels, e_k is a white-noise process with zero mean, and variance $1-\rho^2$, and $k = 0..N$. Here x_0 denotes the boundary pixel and x_1 to x_N are the pixels to be encoded. The correlation between pixels x_k and x_l is given by $R_{kl} = \rho^{|k-l|}$. The model can be straightforwardly extended to the 2-d separable model (separability is assumed along the horizontal and vertical directions). Hence, for 2-d case, the correlation between pixels x_{ij} and x_{mn} will be $\rho^{(|i-m|)}\rho^{(|j-n|)} = \rho^{(|i-m|+|j-n|)}$.

In Fig. 3, the pixels x_{01},. . ., x_{0N} and x_{10}, . . ., x_{N0} denote the boundary pixels that have already been encoded. Pixels $x_{ij}(i,j \in \{1...N\})$ denote the pixels to be encoded. The prediction for a pixel x_{ij} be given by $\tilde{x}_{ij} = x_{0j}$. Hence the prediction error for a pixel is given as: $e_{ij} = x_{ij} - \tilde{x}_{ij} = x_{ij} - x_{0j}$. The overall matrix for the error-residues for the NxN image block is: $\mathbf{E} = \mathbf{X} - \tilde{\mathbf{X}}$ where $\mathbf{X}$ is

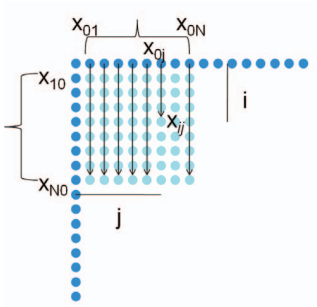

Fig. 3. Vertical Intra prediction mode

the original NxN image block and $\tilde{\mathbf{X}}$ is its prediction. Assuming the separable pixel model, we seek to find the optimal transforms in both the vertical and horizontal directions for the above prediction residue matrix. Specifically for finding the vertical (respectively horizontal) transform of a column of $\mathbf{E}$, we require to find a matrix which diagonalizes the autocorrelation matrix of the corresponding columns (respectively row) of $\mathbf{E}$.

We first consider E_j, column j of $\mathbf{E}$. The autocorrelation matrix of E_j can be obtained via Equations (5) to (10) in Section 3.1 of [10]. The NxN correlation matrix in [10] is simplified to an equivalent matrix M_N with element $\{i,k\} \equiv \min(i,k)$. This, in fact, works very well for $N = 4$. However, for blocks of size 8x8 and larger, the correlation matrix needs to be smoothened as shown in [16] and [11]. The reason is as follows: If a larger block (e.g., 32x32) is chosen for intra prediction, it would be very smooth and the pixel values should not vary a lot. In such a case, the correlation between the neighboring pixels decays slowly (and not as rapidly as in [10]) and an appropriate smoothing factor should be applied for such blocks. In [16], for the 8x8 correlation matrix, a smoothing factor of 50% was proposed and well-supported by extensive simulations, i.e., the entries of the matrix M_8 were modified as follows:

$$\mathbf{M_8} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 \\ 1 & 1.5 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1.5 & 2 & 2.5 & 2.5 & 2.5 & 2.5 & 2.5 \\ 1 & 1.5 & 2 & 2.5 & 3 & 3 & 3 & 3 \\ 1 & 1.5 & 2 & 2.5 & 3 & 3.5 & 3.5 & 3.5 \\ 1 & 1.5 & 2 & 2.5 & 3 & 3.5 & 4 & 4 \\ 1 & 1.5 & 2 & 2.5 & 3 & 3.5 & 4 & 4.5 \end{bmatrix} \quad (1)$$

The above correlation matrix was used to derive a 4x4 and 8x8 secondary transform in [16] and [11] respectively. In fact, for all correlation matrices M_N of size NxN, the smoothing for the intra prediction residues can be generalized as follows, also shown in [15]:

$$p = \min(i,k); \qquad M_N(i,k) = 1 + (p-1)/(N/4) \quad (2)$$

Note that the 4x4 matrix used for deriving DST Type-7 as the optimal transform for 4x4 blocks in [6] and the 8x8 matrix in [16] and [11] are special cases of the smoothened matrix for general N in (2). We next outline the steps for deriving the optimal KxK secondary transform from the NxN autocorrelation matrix M_N in (2):

1. Obtain the correlation matrix after applying DCT on the intra-prediction residuals, i.e., the resulting correlation matrix denoted as $U_N = C_N^T * M_N * C_N$, where C_N is the conventional 2-d DCT matrix of size NxN and '*' is the standard multiplication operator.
2. Obtain the matrix for the top K rows and left-most K columns $V_{K,N} = U_N(1:K, 1:K)$ where the sub-scripts

K and N in $V_{K,N}$ denote that $V_{K,N}$ is obtained from the K top rows and K left columns of NxN correlation matrix U_N.

3. Find the KLT of $V_{K,N}$ of dimension KxK denoted as $W_{K,N}$. The resulting matrix $W_{K,N}$ is the secondary matrix of dimension K that can be used after DCT.
4. In case an integer based approximation of $W_{K,N}$ with m-bit precision (defined as $Y_{K,N}$) is required, multiply $W_{K,N}$ by 2^m and then round the matrix elements to the nearest integer, i.e., $Y_{K,N} = round(2^m * W_{K,N})$.

As an example, we show how to derive an 8x8 secondary transform $Y_{8,32}$ with 7-bit precision from M_{32}. Following the above steps, we have: $U_{32} = C_{32}^T * M_{32} * C_{32}$; $V_{8,32} = U_{32}(1:8, 1:8)$; $W_{8,32} = KLT(V_{8,32})$; and $Y_{8,32} = round(128 * W_{8,32})$ =

$$\begin{bmatrix} 123 & -35 & -8 & -3 & -2 & -1 & -1 & -1 \\ 32 & 120 & -29 & -10 & -5 & -3 & -2 & -1 \\ -14 & -24 & -123 & 21 & 8 & 4 & 3 & 2 \\ 7 & 11 & 17 & 125 & -16 & -7 & -4 & -2 \\ -4 & -7 & -8 & -13 & -126 & 13 & 5 & 3 \\ 3 & 4 & 5 & 6 & 11 & 127 & -10 & -5 \\ -2 & -3 & -3 & -4 & -5 & -9 & -127 & 9 \\ 2 & 2 & 3 & 3 & 3 & 5 & 8 & 128 \end{bmatrix} \quad (3)$$

The above transform can be applied as the vertical secondary transform following the DCT for vertical intra prediction mode. The optimal transform in the horizontal direction will still be DCT as shown in [6] and no secondary transform needs to be applied in the horizontal direction. Next, we show how to apply the secondary transform for all the other intra prediction modes.

3.2. Application of Secondary Transform Based on Intra Prediction modes

Fig. 4 shows the decoder operations when the derived K=8-point secondary transform is applied as a row or column transform depending on the intra prediction mode for a block of size NxN ($N = 8, 16, 32$), where $N \geq K$. The trigger conditions when the secondary transform is used are shown in the right column in Fig. 4 and depend on the categorization of the intra prediction modes.

For category 1 intra prediction modes in Sec. 2, if the prediction was performed from pixels only from the top row, i.e., in vertical direction, and the intra prediction modes are one of the VER, VER+1,. . ., VER+8 as specified in [8], then the secondary transform is used only in the vertical direction. The proof for this comes directly from [14] and [10] where the 4x4 DST was shown to be the optimal vertical transform for all these modes. The analysis in [10] can be trivially extended to the particular case of secondary transform, and due to space limitations we omit the detailed mathematical derivation here. Similarly when the prediction was performed from pixels only from the left column, i.e., in horizontal direction, and intra prediction modes are HOR, HOR+1,. . ., HOR+8 as specified in [8], then the secondary transform is used only in the horizontal direction.

For the DC mode (a non-directional mode), no secondary transform needs to be applied in horizontal and vertical directions. Finally for Planar mode in [3] and Category 2 intra prediction modes, when prediction is performed using both the left column and the top row, i.e., intra prediction modes are VER-1,. . . VER-8 or HOR-1,. . . HOR-7 as specified in [8], the secondary transform is applied in both the horizontal and vertical directions. Again the mathematical derivation is based on the derivation of 4x4 DCT/DST scheme for Category 2 modes in [10]. The encoder instantiation is a straightforward inverse of the decoder implementation.

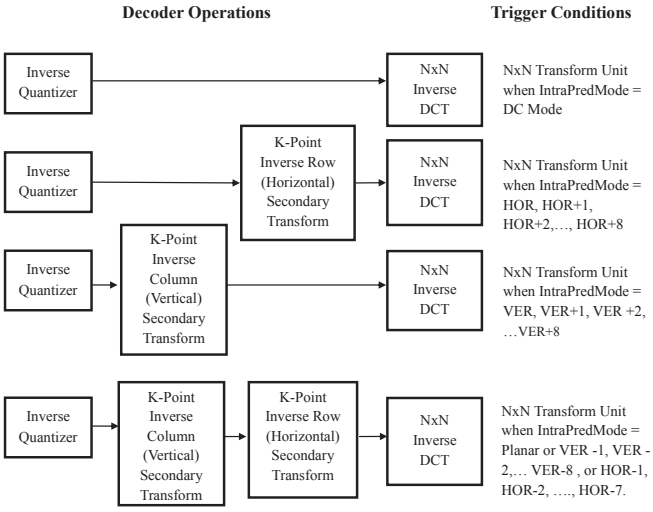

Fig. 4. Example decoder operations for secondary transform

4. EXPERIMENTAL RESULTS

We encoded full length sequences (which had 150 to 600 images) at various resolutions varying from 416x240 to 2560x1600. The anchor (reference) was HM 3.0 [3] with DCT applied to intra-prediction residuals for blocks 8x8 and higher. For 4x4 blocks, we retain the mode-dependent DCT/DST scheme in HM 3.0. The performance of the proposed 8x8 secondary transform was evaluated for the following 4 settings: Intra High Efficiency (HE), Intra Low Complexity (LC), Random Access (RA) HE, and RA-LC configurations as specified in [4]. In the Intra and RA settings, all the images were respectively encoded as I-I-I- and I-B-B- respectively. These video sequences are being tested as part of HEVC standardization. Full details about the GOP size, Intra period, coding structure of these video sequences etc. are available in [4]. Note that we present, here the results for only the evaluation of the secondary transform and retain all other test settings as in [4].

Table 1 presents the BD-Rate [2] gains for Luma component for various video sequences. Here the proposed secondary transform scheme is applied at block sizes 8x8, 16x16 and 32x32. From the table, our proposed algorithm gains upto 1.19% and 1.10% BD-Rate in the Intra HE and Intra LC settings. Note that the results for HE and LC settings for a particular video sequence should not be compared, since there are different tools that are ON and OFF in these settings. For example, in HE settings, entropy coding scheme is CABAC, while in LC setting, entropy coding scheme is CAVLC. The gains in RA-HE and RA-LC setting are upto 0.74 and 0.80 % BD-Rate respectively, since the proposed algorithm is applied only for I Transform Units (TU) (a TU is equivalent to a block in H.264/AVC on which a 4x4 or 8x8 transform is applied) and for the B blocks only DCT is used. Finally note that the results for Vidyo4 sequence are only for Intra configurations and not for RA settings, as specified in the common conditions for HEVC standardization [4].

Table 2 provides the BD-Rate gains when the R-D optimized Rotational Transform is applied at block sizes from 8x8 to 32x32. Here the maximum gains are 1.12% , 1.40% for Intra configurations and 0.86% and 0.95% for RA configurations across all the sequences. Note that the gains vary across the video sequences for both the proposed secondary transform and ROT, and this is typically the case in video coding depending on the sequence characteristics. Also, gains such as 0.5-1% in these advanced stages of HEVC are difficult to come by, and in general, considered to be significant unlike the days

Sequence Name	Intra HE	Intra LC	Random Access HE	Random Access LC
PeopleOnStreet 2560x1600	-1.19	-1.10	-0.54	-0.62
ParkScene 1920x1080	-0.87	-0.62	-0.47	-0.39
Cactus 1920x1080	-0.71	-0.64	-0.54	-0.47
BasketballDrill 832x480	-0.76	-0.87	-0.74	-0.80
RaceHorses 832x480	-0.58	-0.59	-0.43	-0.35
Vidyo4 416x240	-0.99	-0.81	N/A	N/A
Average	-0.85	-0.77	-0.54	-0.52

Table 1. BD-Rate gains when proposed secondary transform is applied at sizes 8x8 to 32x32 for different video sequences under various settings. Note that negative BD-Rate means compression gain.

Sequence Name	Intra HE	Intra LC	Random Access HE	Random Access LC
PeopleOnStreet 2560x1600	-0.90	-1.40	-0.23	-0.56
ParkScene 1920x1080	-1.12	-1.27	-0.86	-0.95
Cactus 1920x1080	-0.95	-1.13	-0.76	-0.77
BasketballDrill 832x480	-0.41	-0.60	-0.61	-0.65
RaceHorses 832x480	-1.04	-1.04	-0.54	-0.41
Vidyo4 416x240	-0.84	-1.11	N/A	N/A
Average	-0.88	-1.09	-0.60	-0.67

Table 2. BD-Rate gains when Rotational Transform is applied at sizes 8x8 to 32x32 for different video sequences under various settings. Note that negative BD-Rate means compression gain.

of H.264/AVC, where 5-10% gains were considered significant.

4.1. Discussion

We begin the discussion by a note on complexity of the transforms: The average increase over the reference (no secondary transform case) in the encoder/decoder run-times for the proposed secondary transform is within the range of 1-4 % across all tested configurations. This is expected as the proposed transform is mode-dependent, and does not require any R-D search. On the other hand, for ROT, the average increase in decoder run-time over the reference is within 1-2 % which is as expected since the choice of ROT transform is signaled to the decoder. However, the increase in run-times for the encoder is around 25 %, 57 %, 5 %, 5 % for the Intra HE, LC, RA-HE and RA-LC configurations. The reason for such an increase is the R-D search over 5 possible transforms (4 ROT transform, and 1 no transform case) at the encoder for Intra blocks. Given such an increase in Intra encoding times for ROT, mode-dependent secondary transform provides an attractive option in terms of compression efficiency at almost negligible additional complexity. Other advantages of the 8x8 secondary transform are:

1. The same secondary transform can be applied on all blocks: 8x8 and larger, thereby eliminating the need of storing and applying different secondary transforms for different blocks: 8x8, 16x16 and 32x32. In our experiments, when different secondary transforms designed for 8x8, 16x16 and 32x32 blocks are respectively applied to these blocks, the difference in compression efficiency is almost negligible. However, in that case three matrices would be required to be stored.
2. The secondary transform can be applied to any non-square block such as 8x32 as well in a similar fashion, after the 8-point DCT and 32-point DCT for the 8x32 block.

We should also mention that in this paper, secondary transform is applied as a full matrix multiplication. Future work includes finding a fast algorithm for the proposed secondary transform.

5. CONCLUSIONS

In this paper, a mode-dependent secondary transform scheme is presented as the transform for intra prediction residual at block sizes 8x8 and higher. The proposed transform scheme requires the storage of only one additional 8x8 matrix, is based on the intra prediction mode, and does not require any explicit signaling. Simulation results show significant gains in compression performance as compared to HM 3.0 anchors at almost negligible complexity increase.

6. REFERENCES

[1] E. Alshina, A. Alshin, and F. C. Fernandes, "Rotational transform for image and video compression," in *IEEE ICIP*, Sept 2011.

[2] G. Bjontegaard, "Calculation of average PSNR Differences between RD curves," *ITU-T SG16/Q6, VCEG-M33*, April 2001.

[3] F. Bossen, D. Flynn, and K. Suhring, "JCT-VC AHG report: Software development and HM software technical evaluation (AHG 3)," *ITU-T & ISO/IEC JCTVC-F003*, July 2011.

[4] F. Bossen, "Common test conditions and software reference configurations," *ITU-T & ISO/IEC JCTVC-B300*, July 2010.

[5] R. Cohen *et al.*, "Description of Core Experiment 7: Additional transforms," *ITU-T & ISO/IEC JCTVC-F907*, July 2011.

[6] J. Han, A. Saxena, and K. Rose, "Towards jointly optimal spatial prediction and adaptive transform in video/image coding," in *IEEE ICASSP*, March 2010, pp. 726–729.

[7] Z. Ma *et al.*, "Experimental results for the Rotational transform," *ITU-T & ISO/IEC JCTVC-F294*, July 2011.

[8] J. Min *et al.*, "Unification of the directional intra prediction methods in TMuC," *ITU-T & ISO/IEC JCTVC-B100*, July 2010.

[9] K. R. Rao and P. Yip, *Discrete Cosine Transform-Algorithms, Advantages and Applications*. Academic Press, 1990.

[10] A. Saxena and F. C. Fernandes, "Mode Dependent DCT/DST for intra prediction in block-based image/video coding," in *IEEE ICIP*, Sept 2011, pp. 1721–1724.

[11] ——, "On secondary transforms for intra prediction residual," *ITU-T & ISO/IEC JCTVC-F554*, July 2011.

[12] ——, "Jointly optimal intra prediction and adaptive primary transform," *ITU-T & ISO/IEC JCTVC-C108*, October 2010.

[13] ——, "Mode-dependent 8x8 DCT/DST for intra prediction," *ITU-T & ISO/IEC JCTVC-F282*, July 2011.

[14] ——, "Mode-dependent DCT/DST without 4*4 full matrix multiplication for intra prediction," *ITU-T & ISO/IEC JCTVC-E125*, March 2011.

[15] A. Saxena, Y. Shibahara, F. C. Fernandes, and T. Nishi, "On secondary transforms for intra prediction residual," *ITU-T & ISO/IEC JCTVC-G108*, Nov 2011.

[16] Y. Shibahara and T. Nishi, "Mode dependent 2-step Transform for intra coding," *ITU-T & ISO/IEC JCTVC-F224*, July 2011.

[17] T. Wiegand *et al.*, "Overview of the H.264/AVC video coding standard," *IEEE Trans. on CSVT*, vol. 13, no. 7, pp. 560–576, July 2003.

[18] Y. Ye and M. Karczewicz, "Improved H.264 intra coding based on bi-directional intra prediction, directional transform, and adaptive coefficient scanning," in *IEEE ICIP*, Oct 2008, pp. 2116–2119.