Coding Gain Optimized 8-Point DST with Fast Algorithm for Intra-frames in Video Coding

M. Hüsrev Cılasun[†] and Fatih Kamışlı[*]
[*†] Department of Electrical and Electronics Engineering
Middle East Technical University, Ankara, Turkey
Email: {husrev.cilasun, kamisli}@metu.edu.tr
[†] Aselsan Inc., Ankara, Turkey
Email: mhcilasun@aselsan.com.tr

***Abstract*—In modern video codecs, such as HEVC and VP9, intra-frame blocks are decorrelated using DCT or DST. The optimal separable transform for intra prediction residual blocks has been determined to be a hybrid transform composed of the DCT and the odd type-3 DST (ODST-3), independent of block size. However, the ODST-3 has no fast algorithm like the DCT. Hence its use in HEVC and VP9 has been limited to only 4x4 blocks. For larger blocks such as 8x8 or 16x16, HEVC replaces the optimal ODST-3 with the conventional DCT while VP9 replaces it with the even type-3 DST (EDST-3), both of which have fast algorithms. The EDST-3 has better coding gain than the DCT but it has still a coding gain loss with respect to the optimal ODST-3. This paper attempts to optimize some parameters of the EDST-3 to reduce this coding gain loss while still retaining the fast algorithm. In particular, the 8-point EDST-3 is represented as a cascade of Givens rotations and some rotation angles are optimized to reduce the coding gain loss with respect to the optimal 8-point ODST-3. By replacing only the 8-point EDST-3 in VP9 with this optimized transform, while leaving other transforms with different sizes unchanged, average Bjøntegaard-Delta bitrate savings of -0.13% are achieved with respect to the standard VP9 codec.**

I. Introduction

Standard video coding approaches, such as VP9 [9] or HEVC [11], are based on a hybrid coding architecture that combines spatial or temporal prediction with transform coding. In particular, a block of pixels are first predicted using pixels either from a previously coded frame (temporal/inter prediction) or from previously coded regions of the current frame (spatial/intra prediction). The prediction error (residual) block typically carries significant spatial correlation and is transform-coded. Specifically, the transform coefficients of the residual block are quantized and entropy coded together with other relevant side information such as prediction modes.

Transform coding is an important part of the hybrid coding approach in video and image compression. Many research efforts have been devoted to optimize the transform to fully exploit the correlation in the transformed signal. In inter-frame coding, the transformed signal is the temporal prediction residual while in intra-frame coding the transformed signal is the spatial prediction residual. These residuals can have different spatial correlation and therefore the transform used in transform-coding them can be different and needs to be optimized for the correlation of the signal that is transform-coded.

This paper focuses on lossy intra coding, i.e. transform-coding of the spatial prediction residual. In the old video compression standard H.264/AVC [12], the spatial prediction residual was transform-coded with the conventional discrete cosine transform (DCT) that is optimal for first-order Markov processes [5], which could model image pixels or temporal prediction residuals well [3]. However, it was shown in [13, 6] that the spatial prediction residual is not first-order Markov along the prediction direction, and the odd type-3 DST (ODST-3) is optimal. Along the perpendicular direction of the principal prediction direction, the DCT or the ODST-3 are optimal (or close to optimal) depending on whether the prediction is performed from only left or upper boundaries or both. In summary, a hybrid DCT/ODST-3 transform approach was proposed and shown to perform better than the traditional 2D DCT at transform-coding intra prediction residual blocks, independent of block size.

While the hybrid DCT/ODST-3 transform improves the transform-coding spatial prediction residuals, the ODST-3 has no fast algorithm like the DCT. Hence its use in new video coding standards, such HEVC or VP9, has been limited to only the smallest 4x4 block size. For such small block size, the computational complexity between a fast algorithm and brute-force computation (e.g. matrix multiplication) is not significant. However, as block size increases, the computational complexity difference grows significantly, creating a significant drawback for the use of ODST-3 in video codecs. Therefore, for block sizes larger than 4x4, instead of using the the optimal ODST-3, HEVC uses the conventional DCT while VP9 uses the even type-3 DST (EDST-3). Like the DCT, the EDST-3 has a fast algorithm but also has better coding gain than the DCT. However, the EDST-3 still has a coding gain loss with respect to the optimal ODST-3.

This paper explores whether it is possible to optimize some parameters of the EDST-3 to reduce the coding gain loss with respect to the optimal ODST-3 while still retaining the fast algorithm. In particular, the 8-point EDST-3 is represented as a cascade of plane rotations and some rotation angles are optimized to reduce the coding gain loss with respect to the optimal ODST-3. We explore three cases and obtain

ISBN 978-90-827970-1-5

three transforms. These optimized 8-point transforms are integrated to the VP9 codec and experiments with different video sequences are performed. By replacing only the 8-point EDST-3 in VP9 with one of these optimized transform, while leaving other transforms with different sizes unchanged, average Bjøntegaard-Delta (BD) bitrate savings of -0.13% are achieved (in the best case obtained by optimizing all sixteen rotations) with respect to the standard VP9 codec.

The remainder of this paper is organized as follows. Section II reviews related previous work in the literature. Section III presents the optimization problem proposed in this paper and the obtained transforms together with their theoretical coding gains. Section IV presents experimental results with various video sequences using the original VP9 codec and its modified version that includes the obtained transforms. Finally, the paper is concluded in Section V.

II. Related Work

In block-based spatial (intra) prediction, a block of pixels are predicted by copying the block's boundary pixels (which reside in the previously reconstructed left and upper blocks) along a predefined direction inside the block. While HEVC supports 33 such directional modes [8], VP9 supports 8 such directional modes [9], as shown in Figure 1. The prediction residual block, obtained by subtracting the prediction block from the original block, is then transform-coded.

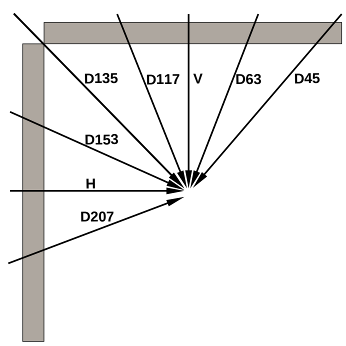

Fig. 1. Angular intra-frame prediction modes in VP9. Maximum efficiency is achieved when EDST-3 is preferred for modes V, D117, D135, D153, and H while EDST-3/DCT combination is preferred for the prediction direction and the other direction, respectively.

The optimal transform for the coding of the spatial prediction residual block was determined as the hybrid DCT/ODST-3 based on modeling the image pixels with a first-order Markov process [13, 6]. Depending on the copying direction of the prediction mode, the DCT or the ODST-3 is applied in either the horizontal and/or vertical direction forming a hybrid 2D transform. In particular, if the copying direction of the prediction mode is near-horizontal using only the left boundary pixels, the ODST-3 is applied along the horizontal direction and the DCT is applied along the vertical direction. Similarly, if the copying direction of the prediction mode is near-vertical using only the upper boundary pixels, the ODST-3 is applied along the vertical and the DCT along the horizontal direction. If the copying direction of the prediction mode uses both the left and upper boundary pixels, then the exact optimal transform can not be analytically derived but applying the ODST-3 along both directions is proposed in [10].

Now, we briefly review the derivation of the auto-correlation of the block-based spatial prediction residual because the coding gains of the various transforms we discuss and derive in the next section are all based on this auto-correlation expression. We use a 1D signal in our discussions for simplicity and because the result can be used for 2D signals by constructing separable 2D transforms.

A first-order Markov process, which is used to model image pixels horizontally within a row or vertically within a column, is represented recursively as

$$u(i) = \rho \cdot u(i-1) + w(i) \tag{1}$$

where ρ is the correlation coefficient, $u(i)$ are zero-mean, unit variance process samples (i.e. image pixels) and $w(i)$ are zero-mean, white noise samples with variance $1-\rho^2$. The auto-covariance or correlation of the process is given by

$$E[u(i) \cdot u(j)] = \rho^{|i-j|}. \tag{2}$$

The DCT is the optimal transform for the first-order Markov process as its correlation coefficient ρ approaches 1 [5].

The spatial prediction block is obtained by copying the boundary pixel of the block, i.e. $u(0)$, inside the block. In other words, the spatial prediction pixels $\hat{u}(i) = u(0)$, $i = 1, .., N$, where N is the block length. The residual block pixels $r(i)$, $i = 1, .., N$, are obtained by subtracting the spatial prediction pixels $\hat{u}(i)$ from the original pixels $u(i)$:

$$\begin{aligned} r(i) &= u(i) - \hat{u}(i) \\ &= u(i) - u(0). \end{aligned} \tag{3}$$

The auto-correlation of the residual pixels is given by $E[r(i)r(j)]$ and is obtained as follows :

$$\begin{aligned} E[r(i)r(j)] &= E[(u(i)-u(0))(u(j)-u(0))] \\ &= \rho^{|i-j|} - \rho^i - \rho^j + 1, \qquad i,j \in \{1, ..., N\} \end{aligned} \tag{4}$$

Such an auto-correlation expression results in a special auto-correlation matrix as the correlation coefficient ρ approaches 1. The eigenvectors of this correlation matrix have been determined to be the basis vectors of the odd type-3 discrete sine transform (ODST-3) given by [13, 7, 6]

$$[S]_{m,n} = \frac{2}{\sqrt{2N+1}} \, sin(\frac{(2m-1)n\pi}{2N+1}), \quad m,n \in \{1, ..., N\} \tag{5}$$

where m and n are integers representing the frequency and time index of the basis functions, respectively. Hence, the optimal transform for the spatial prediction residual block $r(i)$, $i = 1, .., N$ is the ODST-3, as ρ approaches 1.

While the ODST-3 is the optimal transform for spatial prediction residual blocks, it has no fast algorithm like the DCT. Therefore HEVC and VP9 use only the 4-point ODST-3. For larger residual blocks, HEVC uses the conventional DCT

ISBN 978-90-827970-1-5 © EURASIP 2018

and VP9 uses the EDST-3, both of which have fast algorithms. The basis functions of the EDST-3 are given by

$$[E]_{m,n} = \sqrt{\frac{2}{N}}\, sin(\frac{(2m-1)(2n-1)\pi}{4N}), \quad m,n \in \{1,...,N\} \tag{6}$$

While the EDST-3 improves on the coding gain of the DCT, it still has a coding gain loss with respect to ODST-3. The comparison of all these transforms in terms of coding gain losses with respect to the optimal Karhunen Loève Transform (KLT) of the prediction residual with a correlation parameter $\rho = 0.95$ for various block sizes N are shown in Table I.

TABLE I
CODING GAIN LOSSES (IN DB) OF ODST-3, EDST-3 AND DCT RELATIVE TO THE KLT FOR THE BLOCK-BASED SPATIAL PREDICTION RESIDUAL WITH CORRELATION PARAMETER $\rho = 0.95$ AT VARYING BLOCK SIZES.

Block size	4	8	16	32
ODST-3	-0.0009	-0.0024	-0.0045	-0.0072
EDST-3	-0.2174	-0.1376	-0.0797	-0.0468
DCT	-0.6211	-0.5611	-0.4108	-0.2640

The coding gain in dB of a transform T is defined as

$$\mathcal{G}(T) = 10\log_{10}(D_I/D_T) \tag{7}$$

where D_T is proportional to the geometric mean of the transform coefficients $\sigma^2_{T,i}$ $i = 1,...,N$ obtained with transform T as given by

$$D_T \propto (\prod_{i=1}^{N} \sigma^2_{T,i})^{1/N} \tag{8}$$

and D_I is the similarly defined geometric mean of the residual block pixels.

The coding gain loss of a transform T with respect to the KLT is then defined as

$$\mathcal{G}^{loss}(T) = \mathcal{G}(T) - \mathcal{G}(KLT). \tag{9}$$

As can be seen from Table I, there can be a significant coding gain difference between the EDST-3 (which has a fast computation algorithm via butterfly decomposition) and the near-optimal ODST-3 (which has not fast algorithm). The largest coding gain difference for block size larger than 4, is for block size of 8 and thus we focus on the block size of 8 for the remainder of this paper. In the following section, we retain the butterfly decomposition structure of the 8-point EDST-3 to retain its fast algorithm but optimize its butterfly angles to reduce its coding loss relative to the ODST-3 or KLT.

III. CODING GAIN OPTIMIZATION OF EDST-3

The decomposition of the 8-point EDST-3 into a cascade of plane rotations is shown in Figure 2. Each of the sixteen plane rotations in the decomposition have the structure shown in Figure 3. Such a plane rotation structure that has a rotation angle α and processes the i^{th} and j^{th} branches out of the N available signal branches can be represented with the following NxN matrix :

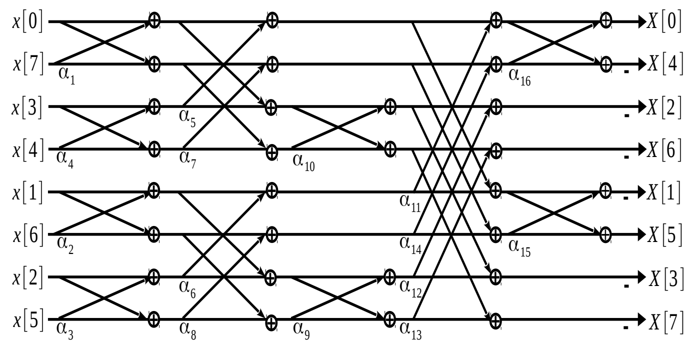

Fig. 2. Decomposition of EDST-3 into cascade of plane rotations.

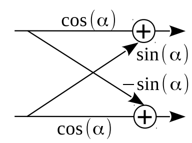

Fig. 3. Plane rotation structure.

$$P(i,j,\alpha) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \cos\alpha & \cdots & \sin\alpha & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -\sin\alpha & \cdots & \cos\alpha & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \tag{10}$$

where all diagonals are 1 and all off-diagonals are 0, except the four sinusoidal terms which appear at the intersections of the i^{th} and j^{th} rows and columns. Hence, a transform formed from a cascade of such L plane rotations can be presented as a product of such matrices : $\Pi_{k=1}^{L} P(i_k, j_k, \alpha_k)$.

Out of the sixteen plane rotations of the EDST-3 in Figure 2, ten rotations have angle $\alpha_k = 45°$ and the remaining six rotations (with angles $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_9, \alpha_{10}$) have different angles as shown in the first column of Table II. Rotation angles of $45°$ are preferable since rotation can in this case be implemented with only addition and subtraction and multiplications can be avoided. This is because the factors $cos(\alpha)$ and $\pm\sin(\alpha)$ (see Figure 3) become equal and thus can be factored out into the outgoing branches of the rotation structure. These factors can then be merged into following rotations in the cascade, allowing thus rotations with angle of $45°$ to be implemented without multiplications.

Our major goal in this paper is to obtain an 8-point transform that has better coding gain than the 8-point EDST-3 on spatial prediction residuals, while at the same time having a fast computation algorithm, i.e. a decomposition into plane rotations similar to EDST-3 in Figure 2. One simple approach towards this goal, which we pursue in this paper, is to keep

ISBN 978-90-827970-1-5

TABLE II
GIVENS PLANE ROTATION ANGLES OF THE OPTIMIZED TRANSFORMS

Angle	EDST-3	6-rot-opt-EDST-3	10-rot-opt-EDST-3	16-rot-opt-EDST-3
α_1	84.38°	80.36°	80.23°	79.28°
α_2	73.13°	70.36°	70.26°	69.08°
α_3	61.86°	60.48°	60.61°	59.39°
α_4	50.63°	50.72°	50.81°	49.94°
α_5	45°	45°	45°	46.88°
α_6	45°	45°	45°	45.66°
α_7	45°	45°	45°	57.48°
α_8	45°	45°	45°	48.06°
α_9	22.50°	20.79°	17.99°	16.89°
α_{10}	67.50°	62.34°	59.45°	54.07°
α_{11}	45°	45°	45.31°	45.30°
α_{12}	45°	45°	53.70°	56.71°
α_{13}	45°	45°	46.66°	48.15°
α_{14}	45°	45°	54.23°	50.05°
α_{15}	45°	45°	45°	35.23°
α_{16}	45°	45°	45°	35.25°

the overall decomposition structure (i.e. the order and the connections of all rotations) of the EDST-3 and only change the angles α_r of some of the plane rotations so that the theoretical coding gain of the overall transform is maximized. This approach can be formalized as the following optimization problem :

$$\max_{\alpha_r \in S} \quad \mathcal{G}(\ \Pi_{k=1}^{16} P(i_k, j_k, \alpha_k)\ ). \qquad (11)$$

Here, the set S contains the angle parameters α_r of those plane rotations that are allowed to be modified. We define three different sets S, leading to 3 different optimization problems and resulting transforms :

- $S_6 = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_9, \alpha_{10}\}$. This set contains the angle parameters of the rotations which do not have 45° angles in the decomposition of the EDST-3 in Figure 2.
- $S_{10} = S_6 \cup \{\alpha_{11}, \alpha_{12}, \alpha_{13}, \alpha_{14}\}$. This set contains additional four rotation angle parameters.
- S_{16}, which contains all sixteen rotation angle parameters of the decomposition of EDST-3.

We call the resulting transforms as follows in the remainder of the paper :

- 6-rot-opt-EDST-3
- 10-rot-opt-EDST-3
- 16-rot-opt-EDST-3.

To solve these optimization problems, the interior point algorithm [2] in MATLAB is used. The resulting transforms are listed in Table II, parametrized by the angles of all their sixteen plane rotations. The coding gains of the transforms are given in Table III using a correlation parameter of $\rho = 0.95$, as it is common in the literature [5, 6]. To present further coding gain comparisons, their coding gains are also plotted as a function of the correlation parameter $\rho = 0.95$ in Figure 4.

The results in Table III show that the obtained transforms can indeed increase the theoretical coding with respect to the EDST-3. By modifying six angles of the EDST-3, a coding gain increase of about 0.04 dB is obtained, by modifying ten angles, a coding gain increase of about 0.06 dB is obtained. By modifying all sixteen angles of the EDST-3, a coding gain increase of about 0.10 dB is obtained, which is only 0.03 dB less than that of the ODST-3. Opt-EDST-3 maintains the same skeleton as EDST-3, yet 45° rotations of VP9 do not require multiplications since they are compensated by bit shifts. Increase of non-45° rotations might cause the complexity of opt-EDST-3 to exceed ODST-3. Thus, the trade off should be addressed carefully.

TABLE III
THEORETICAL CODING GAINS AND CODING LOSSES FOR CORRELATION COEFFICIENT $\rho = 0.95$ (dB)

Transform	Coding Gain	Coding Loss
KLT	10.0087	0
ODST-3	10.0063	-0.0024
16-rot-opt-EDST-3	9.9715	-0.0372
10-rot-opt-EDST-3	9.9330	-0.0757
6-rot-opt-EDST-3	9.9096	-0.0991
EDST-3	9.8711	-0.1376
DCT	9.4476	-0.5611

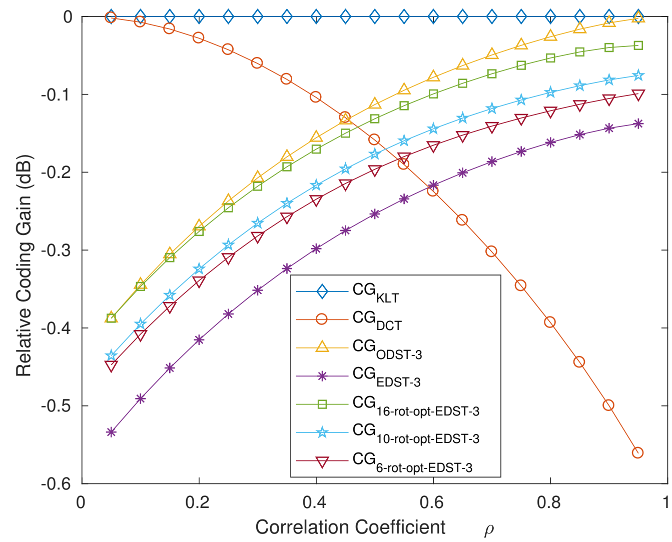

Fig. 4. Relative theoretical coding gains of KLT, ODST-3, several optimized 8-point EDST-3 transforms, EDST-3, and DCT versus correlation coefficient ρ.

IV. EXPERIMENTAL RESULTS

To test the practical compression performance of these transforms, the VP9 codec is used. The 8-point EDST-3 in VP9 is replaced with one of the obtained 8-point transforms while leaving other transforms, such as the 8-point DCT and 4 or 16-point transforms unchanged. Note that both the original and optimized EDST-3 transforms are implemented as integer transforms. Alongside conventional test sequences of HEVC (classes A-E), 120fps 8-bit 4K (3840x2160) sequences from [4] are also used in the experiments. Only intra-frame coding is used by disabling inter-frame prediction with parameter *–kf-max-dist=1*, and the default two-pass frame encoding is used.

ISBN 978-90-827970-1-5 © EURASIP 2018

TABLE IV
BD BITRATE SAVINGS WITH RESPECT TO STANDARD VP9

Class	Sequences	BD-Bitrate(%) 6-rot-o EDST-3	10-rot-o EDST-3	16-rot-o EDST-3	ODST-3
4K Seq. (3840x2160)	Beauty	0.01	-0.02	-0.02	-0.02
	Bosphorus	0.03	-0.05	-0.19	-0.26
	HoneyBee	-0.02	-0.07	-0.06	-0.06
	Jockey	-0.04	-0.09	-0.08	-0.06
	ReadySteadyGo	-0.06	-0.23	-0.33	-0.46
	ShakeNDry	0.02	-0.03	-0.06	-0.03
	YachtRide	-0.12	-0.18	-0.37	-0.60
	Mean	**-0.03**	**-0.10**	**-0.16**	**-0.21**
Class A (2560x1600)	Traffic	0.05	-0.11	-0.10	-0.47
	PeopleOnStreet	0.10	0.01	-0.12	-0.46
	Mean	**0.08**	**-0.05**	**-0.11**	**-0.46**
Class B (1920x1080)	Kimono1	-0.09	-0.04	-0.18	-0.19
	ParkScene	0.05	0.04	-0.01	-0.09
	Cactus	-0.03	-0.01	-0.06	-0.06
	BQTerrace	-0.01	-0.02	-0.05	-0.40
	BasketballDrive	-0.16	-0.14	-0.29	-1.04
	Mean	**-0.05**	**-0.03**	**-0.12**	**-0.35**
Class C (832x480)	RaceHorses	0.08	0.00	-0.03	-0.06
	BQMall	-0.03	-0.13	-0.23	-0.23
	PartyScene	-0.04	-0.01	-0.07	-0.07
	BasketballDrillText	0.03	0.02	-0.06	-0.21
	Mean	**0.01**	**-0.03**	**-0.09**	**-0.15**
Class D (416x240)	RaceHorses	0.10	0.03	-0.04	-0.17
	BQSquare	-0.03	-0.07	-0.07	-0.08
	BlowingBubbles	-0.03	-0.16	-0.15	-0.13
	BasketballPass	-0.23	-0.20	-0.13	-0.47
	Mean	**-0.05**	**-0.10**	**-0.10**	**-0.21**
Class E (1280x720)	FourPeople	-0.11	-0.20	-0.32	-0.72
	Johnny	0.08	-0.08	-0.22	-0.35
	KristenAndSara	0.01	-0.00	-0.10	-0.32
	Mean	**-0.01**	**-0.10**	**-0.21**	**-0.46**
	Overall Mean	**-0.02**	**-0.09**	**-0.13**	**-0.28**

The used quantization parameters are 28,24,20, and 16. The experimental results are evaluated with respect to the default VP9 codec using the Bjøntegaard-Delta (BD) bitrate metric [1] and are presented in Table IV.

The results in Table IV indicate that when any of the obtained 8-point transforms replaces the EDST-3, the compression performance of the VP9 codec can slightly increase, which is consistent with the theoretical coding gain results in Table III. The BD bitrate savings are on average -0.02%, -0.09% and -0.13% when the 6-rot-opt-EDST-3 or the 10-rot-opt-EDST-3 or the 16-rot-opt-EDST-3, respectively, replaces the 8-point EDST-3 in VP9. When the ODST-3 with brute-force matrix multiplication is used to replace the 8-point EDST-3 in VP9, an average BD bitrate savings of -0.28% is achieved.

V. CONCLUSION

The hybrid DCT/ODST-3 transform has been proposed to transform-code the spatial prediction residuals independent of the block size. However, the ODST-3 has no fast algorithm like the DCT and thus its use in new video coding standards has been limited to only the smallest 4x4 block size. For larger block sizes, VP9 uses instead the EDST-3, which has a fast algorithm due to its decomposition into plane rotations. However, the EDST-3 has still a coding gain loss with respect to the ODST-3. This paper focused on the 8-point transform and optimized the angles of some of the plane rotations of the 8-point EDST-3 to reduce its coding loss relative to the 8-point ODST-3. The resulting transform(s) provided slight improvements in both the theoretical coding gain and practical compression tests with the VP9 codec.

REFERENCES

[1] Gisle Bjontegarrd. "Calculation of average PSNR differences between RD-curves". In: *VCEG-M33* (2001).

[2] Richard H Byrd, Jean Charles Gilbert, and Jorge Nocedal. "A trust region method based on interior point techniques for nonlinear programming". In: *Mathematical Programming* 89.1 (2000), pp. 149–185.

[3] C-F Chen and Khee K Pang. "The optimal transform of motion-compensated frame difference images in a hybrid coder". In: *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 40.6 (1993), pp. 393–397.

[4] *Digiturk 4K Test Sequences [Online]. Available:* http://ultravideo.cs.tut.fi/testsequences. Accessed: 2018-13-02.

[5] M.D. Flickner and N. Ahmed. "A derivation for the discrete cosine transform". In: *Proceedings of the IEEE* 70.9 (1982), pp. 1132–1134. ISSN: 0018-9219.

[6] Jingning Han et al. "Jointly optimized spatial prediction and block transform for video and image coding". In: *IEEE Transactions on Image Processing* 21.4 (2012), pp. 1874–1884.

[7] Anil K Jain. "A sinusoidal family of unitary transforms". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 4 (1979), pp. 356–365.

[8] J. Lainema et al. "Intra Coding of the HEVC Standard". In: *Circuits and Systems for Video Technology, IEEE Transactions on* 22.12 (Dec. 2012), pp. 1792–1801. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2012.2221525.

[9] Debargha Mukherjee et al. "A technical overview of VP9 - The latest open-source video codec". In: *SMPTE Motion Imaging Journal* 124.1 (2015), pp. 44–54.

[10] Ankur Saxena and Felix C Fernandes. "Mode dependent DCT/DST for intra prediction in block-based image/video coding". In: *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE. 2011, pp. 1685–1688.

[11] G.J. Sullivan et al. "Overview of the High Efficiency Video Coding (HEVC) Standard". In: *Circuits and Systems for Video Technology, IEEE Transactions on* 22.12 (Dec. 2012), pp. 1649–1668. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2012.2221191.

[12] T. Wiegand et al. "Overview of the H.264/AVC video coding standard". In: *Circuits and Systems for Video Technology, IEEE Transactions on* 13.7 (July 2003), pp. 560–576. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2003.815165.

[13] Chuohao Yeo et al. "Mode-Dependent Transforms for Coding Directional Intra Prediction Residuals". In: *Circuits and Systems for Video Technology, IEEE Transactions on* 22.4 (Apr. 2012), pp. 545–554. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2011.2168291.