LOW COMPLEXITY TRANSFORM CODING FOR DEPTH MAPS IN 3D VIDEO

Fabian Jäger *Karam Naser*

Institut für Nachrichtentechnik
RWTH Aachen University, GERMANY
{jaeger,naser}@ient.rwth-aachen.de

ABSTRACT

3D video is a new technology, which requires transmission of depth data alongside conventional 2D video. The additional depth information allows to synthesize arbitrary viewpoints at the receiver for adaptation of perceived depth impression and for driving of multi-view auto-stereoscopic displays. Depth maps typically show different signal characteristics compared to textured video data. Piecewise smooth regions are bounded by sharp edges resembling depth discontinuities. These edges lead to strong ringing artifacts when depth maps are coded with DCT-based transform codecs, such as AVC or its successor HEVC.

In this paper alternative transforms are proposed to be used for coding depth maps for 3D video. By replacing the DCT with these transforms, ringing artifacts in the reconstructed depth maps are reduced and at the same time the complexity of the transform stage is lowered significantly. For high quality depth map coding the proposed alternative transforms can even increase coding efficiency.

Index Terms— depth map, video coding, transforms, complexity, 3d video

1. INTRODUCTION

3D video technology extends conventional stereo videos by the addition of the corresponding depth maps. Currently, stereoscopic displays, which require special glasses, are still used to display stereoscopic content. New technologies, such as auto-stereoscopic and depth-adapting stereoscopic displays, require more than two views of the same scene to allow for their enhanced 3D features. The Multi-View plus Depth (MVD) data format, which bundles multi view texture videos with their corresponding depth maps, was designed for these technologies.

Depth maps typically have different properties compared to texture images. First, they can be characterized by smooth areas of almost constant depth, which are bounded by sharp edges that resemble depth discontinuities. Second, depth maps are not viewed by the user, but are rather used to synthesize additional views using Depth Image Based Rendering. Third, any distortion to depth maps will be mapped to object deformations in the rendered views [1]. For these reasons, several approaches were investigated to efficiently code depth maps while retaining high view synthesis quality.

Recent compression algorithms approximate depth map signal characteristics by a partitioning the frame into triangular meshes [2] or platelets [3] and modeling each segment by an appropriate 2D function. These pure model-based approaches can also be combined with transform-based tools by introducing an additional coding mode, such as the sparse-dyadic (SD) mode [4]. An SD-coded block is divided into two segments with each one described by a constant depth value. This concept can further be extended by incorporating structural information from the corresponding texture component as proposed with Depth Modeling Modes [5].

In video coding, discrete cosine transform (DCT) is the state of the art block transform. Whereas alternative transforms have been developed for depth map coding to be better optimized for their unique signal characteristics. In [6] *Don't Care Regions* are defined as areas where distortions of the depth map cause a synthesis distortion smaller than a certain threshold (α). Based on this information a sparse representation of the depth signal is sought. The same approach was further developed in [7], allowing the synthesized image to be at arbitrary positions. For *Graph Based Transforms* (GBT) [8] the adjacency matrix of each block is used to calculate the degree matrix, from which the Laplacian matrix is calculated. The GBT is constructed based on the eigenvectors of the Laplacian matrix while the adjacency matrix is coded as additional side information.

All described methods can efficiently code depth maps, but they generally introduce a lot of complexity to the decoder compared to conventional DCT-based transform coding. In the proposed transform coding scheme for depth maps, alternative transforms are used, which significantly reduce the computational complexity for the transform stage while even improving coding efficiency.

The remainder of this paper is organized as follows: Section 2 explains major differences between the analyzed transforms. Section 3 presents simulation results with respect to coding performance when using the proposed alternative transforms. This analysis is further extended in Section 4 to investigate the resulting computational complexity. Finally,

Section 5 summarizes the results of the presented investigation and gives an outlook on potential future research topics in this field.

2. THEORETICAL BACKGROUND

In this section the proposed alternative transforms are explained and it is described why these transforms are better matched with typical depth map signal characteristics. Moreover, an example is given showing a reconstructed depth map coded with different transform types.

2.1. Haar Transform

The Haar transform is a relatively simple transform and was introduced by Alfred Haar in 1910 [9].

The discrete Haar transform uses rectangular basis functions, which are defined for any $2^N, N \in \mathbb{I}^+$. The $2n$ Haar transform matrix can be generated from n using the following formula:

$$H(2n) = \begin{bmatrix} H(n) \otimes \begin{bmatrix} 1 & 1 \end{bmatrix} \\ 2^{\frac{n-2}{2}} I \otimes I_{2^{n-2}} \otimes \begin{bmatrix} 1 & -1 \end{bmatrix} \end{bmatrix} \tag{1}$$

where

$$H(1) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2}$$

which needs to be normalized in case of an orthonormal transform.

An example of a Haar transform matrix is given in (3). It can be seen that the transform basis functions consist of two components: The DC component and the difference component. The first difference function is shifted and scaled $n-1$ times.

$$\frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \end{bmatrix} \tag{3}$$

2.2. Walsh-Hadamard Transform

The Walsh-Hadamard transform is based on the Hadamard matrix, which was introduced by the french mathematician Jacques Hadamard [10]. Hadamard matrices are square with entries of $\begin{bmatrix} 1 & -1 \end{bmatrix}$ and orthogonal rows.

The structure of a Hadamard matrix is given in (4). For any matrix X, the Hadamard matrix can be generated by replicating X horizontally and vertically, as described by (4). Hence, Hadamard matrices can be generated systematically starting from 1×1 matrix as shown in (5).

$$\begin{bmatrix} X & X \\ X & -X \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{5}$$

The proposed Walsh-Hadamard transform is the *sequency*-arranged variant of the Hadamard matrices with orthonormal rows. The *sequency* is half the number of sign changes in the basis functions. An example of this is given in (6).

$$\frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \tag{6}$$

2.3. Comparison of Transforms for Depth Map Coding

In Figure 1 the 8×8 2D basis functions of the two proposed transforms and the DCT are depicted. It can be seen that Haar and Walsh-Hadamard basis functions possess the property of sharp transitions between positive and negative values. Compared to these, to DCT shows smoother transitions. This property enables these transforms to better approximate blocks with discontinuities, as can be seen in typical depth map signals.

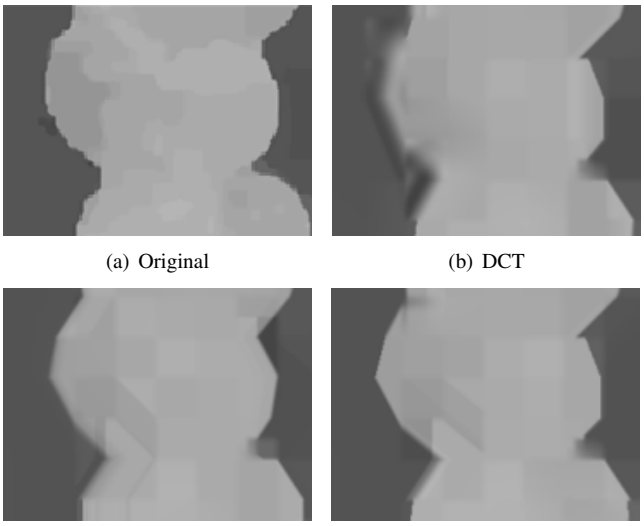
The DCT is widely used in coding of natural images and videos. For images with rapid signal changes it has been reported that the Haar transform performs better [11] with respect to coding efficiency. Depth maps, as stated before, are characterized by smooth areas with sharp edges. For such signals the DCT introduces strong ringing artifacts after the quantization of the resulting transform coefficients. The proposed transforms, however, do not tend to introduce such artifacts. An example of this ringing effect is shown in Figure 2 where the boundaries of the objects are extremely deformed in the DCT case and better kept when Haar or Walsh-Hadamard transforms are used. As a consequence, ringing artifacts around objects boundaries result in erosion artifacts in the reconstructed views, as shown in [1].

3. CODING EFFICIENCY ANALYSIS

The proposed alternative transforms were implemented into the JCT-3V HEVC Test Model (HTM 4.0) [12]. The encoder can be configured to either use the conventional DCT/DST transforms or use one of the proposed transforms for coding depth maps. All simulations are performed following the

(a) DCT (b) Haar (c) Walsh-Hadamard

Fig. 1: 2D basis functions of different transforms for 8×8 transform block sizes.



(a) Original (b) DCT

(c) Haar (d) Walsh-Hadamard

Fig. 2: Uncompressed and reconstructed depth maps with different transform functions. The quantization parameter for the depth map is $\mathrm{QP_D} = 50$.

JCT-3V common test conditions [13]. To get more meaningful results, additional simulations were performed and are presented hereafter.

3.1. General Evaluation Framework

As depth maps are typically not displayed and can be regarded as supplementary data to the corresponding texture videos, PSNR values are computed from synthesized views. The reference frames for this evaluation are synthesized views based on uncompressed texture and depth data. PSNR values are computed between synthesized views from uncompressed data and synthesized views based on decoded and reconstructed data. This evaluation procedure is also used within JCT-3V and complies with the common test conditions of the JCT-3V standardization activity [13].

In Table 1 simulation results of using the alternative transforms for depth map coding can be found. The results show that changing the transform basis functions only has a minimal impact on the overall coding performance.

Table 1: BD-Rate savings for proposed alternative transforms, following the JCT-3V common test conditions. BD-Rates are computed based on PSNR of synthesized views and total bitrate.

Sequence	Haar	Walsh-Hadamard
Balloons	0,13 %	0,15 %
Kendo	0,29 %	0,31 %
Newspaper	0,25 %	0,31 %
GT Fly	0,08 %	-0,10 %
Poznan Hall	-0,08 %	0,13 %
Poznan Street	0,11 %	0,12 %
Undo Dancer	0,09 %	-0,18 %
1024×768	0,22 %	0,25 %
1920×1088	0,05 %	-0,01 %
AVERAGE	**0,12 %**	**0,11 %**

3.2. Results with Fixed Texture Quality

The evaluation scheme following the common test conditions of JCT-3V is not optimal for evaluating pure depth coding tools, because the resulting bitrate savings are measured with respect to the total bitrate, which includes texture and depth bitrate. Moreover, the measured synthesis quality (in PSNR) is not only affected by distortions of the depth maps, but also by distortions to the corresponding texture views. Therefore it is relatively difficult to evaluate the impact of the tested depth coding tool alone.

To overcome this drawback of the evaluation framework an additional set of results is generated by using the same texture quality ($\mathrm{QP_T} = 20$) for all four depth map rate points. Only the depth map quality is varied by four different values of the depth map quantizer parameter ($\mathrm{QP_D}$). When measuring the PSNR values of the synthesized views, the reconstructed depth maps are used in combination with the uncompressed texture views and are compared to the synthesized reference views. Hence, distortions in the synthesized views are only influenced by coding artifacts in the reconstructed depth maps.

As the bitrate for the texture views is the same for all rate points of each sequence, it is also possible to show the amount of bitrate savings for the depth component. This better reflects the impact of the modified depth map coding as the BD-Rates are computed based on the depth bitrate and PSNR values of synthesized views, which are generated from reconstructed depth maps and uncompressed texture views.

The simulation results in Table 2 show that using the proposed alternative transform types does not lead to conclusive results as some sequences benefit from the different transform characteristics while others do not. The impact on coding efficiency with respect to the total bitrate (including texture) is negligible. That is due to the fact that the fixed and high tex-

Table 2: BD-Rate savings for proposed alternative transforms when coding the texture views with $QP_T = 20$. BD-Rates are computed based on PSNR of synthesized views and total or depth bitrate, respectively.

Sequence	Haar		Walsh-Hadamard	
	Total	Depth	Total	Depth
Balloons	0,03 %	1,21 %	0,04 %	1,51 %
Kendo	0,04 %	1,04 %	0,03 %	0,70 %
Newspaper	0,01 %	1,51 %	0,07 %	2,64 %
GT Fly	0,00 %	0,94 %	0,00 %	1,40 %
Poznan Hall	-0,01 %	-0.33 %	-0,01 %	-0,24 %
Poznan Street	0,02 %	1,88 %	0,02 %	2,00 %
Undo Dancer	-0,02 %	-1,95 %	-0,04 %	-2,53 %
1024×768	0,02 %	1,25 %	0,04 %	1,61 %
1920×1088	0,00 %	0,14 %	-0,01 %	0,16 %
AVERAGE	**0,01 %**	**0,62 %**	**0,02 %**	**0,78 %**

Table 3: BD-Rate savings for proposed alternative transforms coding the texture views with $QP_T = 20$ and the depth maps at high quality with $QP_D \in \{15, 20, 25, 30\}$. BD-Rates are computed based on PSNR of synthesized views and total or depth bitrate, respectively.

Sequence	Haar		Walsh-Hadamard	
	Total	Depth	Total	Depth
Balloons	-1,51 %	-3.70 %	-0.73 %	-1.77 %
Kendo	-1,86 %	-4.40 %	-1.05 %	-2.55 %
Newspaper	-2,89 %	-6.52 %	-1.32 %	-2.93 %
GT Fly	-1,75 %	-7.72 %	-1.46 %	-6.81 %
Poznan Hall	-1,31 %	-7.44 %	-0.56 %	-3.25 %
Poznan Street	-0.81 %	-3.77 %	-0.30 %	-1.51 %
Undo Dancer	-0.28 %	-2.97 %	-0.01 %	0.01 %
1024×768	-2,09 %	-4.87 %	-1.03 %	-2.41 %
1920×1088	-1,04 %	-5.48 %	-0.58 %	-2.89 %
AVERAGE	**-1,49 %**	**-5.22 %**	**-0.77 %**	**-2.68 %**

ture quality results in a very high bitrate for the texture component while the rate for the depth component is negligibly small.

3.3. Results with Fixed Texture and High Depth Quality

In a third set of simulations the performance of the alternative transforms for depth maps is evaluated with a configuration for higher reconstruction quality. When following the common test conditions of JCT-3V, the depth map quality is relatively low (QP values between 43 and 50). The proposed transforms for depth map coding are expected to perform even better in coding scenarios where higher depth map quality is targeted. Therefore, another set of simulations is performed, which follows the evaluation method explained in Section 3.2, but uses a QP_D range for the depth maps of 15 to 30 with a step size of 5. The QP for the texture (QP_T) is still fixed to 20.

From the simulations results in Table 3 it can be seen that in a coding scenario targeting high quality depth maps, using the proposed alternative transforms yields significant reductions in terms of BD-Rate. In the low QP_D coding scenario, more transforms are used (which is further investigated in Section 4) and consequently the impact of different transform characteristics become visible with respect to coding efficiency gains.

4. COMPLEXITY ANALYSIS

For the following complexity analysis, the number of additions/subtractions (ADDS), multiplications (MULTS) and binary shifts (SHIFTS) is used to compare different transform types. HEVC uses both, DCT and DST (discrete sine transform) transforms, depending on the block size and prediction mode. As the DST is only used in 4×4 intra-predicted blocks, the complexity analysis in this paper concentrates on the complexity comparison of DCT, Haar and Walsh-Hadamard. For all three transform types, fast implementations [14], [15], [16] are used to demonstrate the complexity impact for optimized algorithms. Moreover, a straight-forward matrix multiplication approach is added to the complexity analysis as a worst-case reference.

For each of the compared transform implementations the required number of operations is shown in Table 4.

For the presented complexity analysis, different transform types were implemented as alternative transforms into the HTM reference software of JCT-3V [12]. Simulations were performed according to the JCT-3V common test conditions (CTC) [13]. The average number of transforms of different sizes per depth frame are computed for the whole set of JCT-3V test sequences. Following the CTC, depth maps are coded with a relatively high quantizer setting (QP), which results in many skipped blocks in temporally predicted frames and consequently ends up in a very low number of transforms in depth frames. Therefore, an additional set of simulations was performed at a lower QP_D. This procedure and the QP range is the same as described in Section 3. In Table 5, it can be seen that for a finer depth quantization, the number of transforms increases significantly.

In a final step, these simulation results are combined with the number of arithmetic operations (Table 4) to obtain the overall complexity analysis of different transform types for the JCT-3V test sequences. The results are depicted in Figure 3 for the lower QP_D range and in Figure 4 for the default CTC simulation configuration. In the provided results, it can be seen that the two proposed alternative transforms (Haar and Walsh-Hadamard) can significantly reduce the computa-

Table 4: Number of mathematical operations required for different implementations and transform sizes.

Block Size	MULTS	ADDS	SHIFTS
Matrix			
4×4	128	96	32
8×8	1024	896	128
16×16	8192	7680	512
32×32	65536	63488	2048
Fast DCT (FDCT)			
4×4	0	136	40
8×8	0	1184	624
16×16	0	7424	4224
32×32	0	35072	15936
Fast Haar (FHT)			
4×4	32	48	32
8×8	128	224	128
16×16	512	960	512
32×32	2048	3968	2048
Fast Walsh-Hadamard (FWHT)			
4×4	0	64	32
8×8	0	384	128
16×16	0	2048	512
32×32	0	10240	2048

tional complexity, even compared to a highly optimized DCT implementation.

Table 5: Average number of transforms per depth frame for different transform sizes. The simulation setup follows the configuration described in Section 3.

Configuration	Number of Transforms			
	4×4	8×8	16×16	32×32
HighQP	62	48	72	30
LowQP	1648	460	216	64

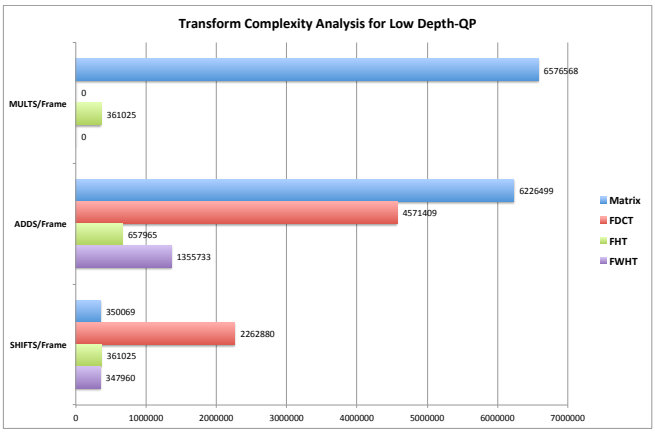


Fig. 3: Number of required arithmetic operations per depth frame for different transform types. The $\mathrm{QP_D}$ range follows the JCT-3V common test conditions while the $\mathrm{QP_T}$ was fixed to 20.

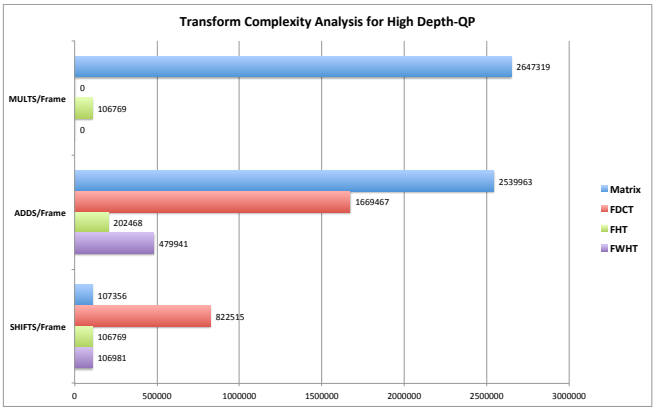


Fig. 4: Number of required arithmetic operations per depth frame for different transform types. While the $\mathrm{QP_T}$ was fixed to 20, the depth quantizer was configured to be $\mathrm{QP_D} \in \{15, 20, 25, 30\}$.

5. CONCLUSION

This paper proposes to replace the discrete cosine transform (DCT) with either the Haar or the Walsh-Hadamard transform when coding depth maps. Both alternative transform types are better matched with typical signal characteristics of depth maps and consequently tend to reduce the amount of ringing artifacts along depth discontinuities compared to conventional DCT/DST transforms. Especially for high quality coding of depth maps the alternative transforms outperform the DCT/DST-based coding of 3D video data. Moreover, it is shown that the two alternative transform types have a significantly lower complexity in terms of number of required multiplications and additions compared to the latest reference. This property is very important due to the fact that depth maps typically contribute to less than 10% of the total bitrate of a 3D video bitstream. Therefore, coding depth maps should consequently be coded with low complexity as it can be interpreted as additional side information to the decoder, similar to motion vectors.

Further study is needed to investigate whether the entropy coding stage for coding transform coefficients in 3D-HEVC can be further adapted and optimized for the proposed alternative transforms. It may be possible to further increase coding efficiency by such modifications.

6. REFERENCES

[1] PoLin Lai, Antonio Ortega, Camilo C. Dorea, Peng Yin, and Cristina Gomila, "Improving view rendering quality and coding efficiency by suppressing compression artifacts in depth-image coding," in *Proc. SPIE*, 2009, vol. 7257, p. 72570O.

[2] Michel Sarkis, Waqar Zia, and Klaus Diepold, "Fast depth map compression and meshing with compressed tritree," in *Computer Vision–ACCV 2009*, pp. 44–55. Springer, 2010.

[3] Yannick Morvan, Dirk Farin, et al., "Platelet-based coding of depth maps for the transmission of multiview images," in *Electronic Imaging 2006*. International Society for Optics and Photonics, 2006, pp. 60550K–60550K.

[4] Shujie Liu, PoLin Lai, Dong Tian, Cristina Gomila, and Chang Wen Chen, "Sparse dyadic mode for depth map compression," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 3421–3424.

[5] Philipp Merkle, Christian Bartnik, Karsten Müller, Detlev Marpe, and Thomas Wiegand, "3D video: Depth coding based on inter-component prediction of block partitions," in *Proceedings of IEEE Picture Coding Symposium*, May 2012, pp. 149–152.

[6] Gene Cheung, Akira Kubota, and Antonio Ortega, "Sparse representation of depth maps for efficient transform coding," in *Picture Coding Symposium (PCS), 2010*. IEEE, 2010, pp. 298–301.

[7] Gene Cheung, Junichi Ishida, Akira Kubota, and Antonio Ortega, "Transform domain sparsification of depth maps using iterative quadratic programming," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 129–132.

[8] G. Shen, W.-S. Kim, S.K. Narang, A. Ortega, Jaejoon Lee, and HoCheon Wey, "Edge-adaptive transforms for efficient depth map coding," in *Picture Coding Symposium (PCS), 2010*. IEEE, 2010, pp. 566–569.

[9] Radomir S. Stanković and Bogdan J. Falkowski, "The haar wavelet transform: its status and achievements," *Computers & Electrical Engineering*, vol. 29, no. 1, pp. 25–44, 2003.

[10] Jacques Hadamard, "Résolution dune question relative aux déterminants," *Bull. sci. math*, vol. 17, no. 1, pp. 240–246, 1893.

[11] O. Hunt and R. Mukundan, "A comparison of discrete orthogonal basis functions for image compression," 2004.

[12] Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) of ITU-T VCEG and ISO/IEC MPEG, "3D-HEVC test model 4," Tech. Rep., Doc. JCT3V-D1005, April 2013.

[13] Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) of ITU-T VCEG and ISO/IEC MPEG, "Common test conditions of 3DV core experiments," Tech. Rep., JCT3V-D1100, April 2013.

[14] Ashfaq Ahmed, Muhammad Usman Shahid, et al., "N point dct vlsi architecture for emerging hevc standard," *VLSI Design*, vol. 2012, pp. 6, 2012.

[15] Peter R. Roeser and M.E. Jernigan, "Fast haar transform algorithms," *Computers, IEEE Transactions on*, vol. 100, no. 2, pp. 175–177, 1982.

[16] Joseph W. Manz, "A sequency-ordered fast walsh transform," *Audio and Electroacoustics, IEEE Transactions on*, vol. 20, no. 3, pp. 204–205, 1972.