Wavelet-Like Lifting-Based Transform for Decomposing Images in Accordance with the Inter-prediction Principles of Video Coding

Marek Parfieniuk(✉)

Department of Digital Media and Computer Graphics,
Bialystok University of Technology, Wiejska 45A, 15-351 Bialystok, Poland
m.parfieniuk@pb.edu.pl

Abstract. In this paper, an innovative approach to image analysis-synthesis is presented, which follows the prediction principles of video coding. It consists in decomposing an image into four polyphase components, which are processed like video frames. One of them is made the essential reference frame, whereas each of the remaining components is predicted using the reference or any of previously encoded components. Such hierarchical prediction is adapted, being similar to the bidirectional motion estimation-compensation using two reference frames, known of the MPEG-4 AVC standard for video coding. On the other hand, obtainable residuals are similar to the results of lifting-based subband decompositions of images, or even to wavelet transforms, if the algorithm is applied iteratively to the reference. But, surprisingly, our computational scheme is most related to the known PLT and GTD algorithms, conceptually distant from both wavelets and video coding, and thus it can be called the hierarchical adaptive spatial triangular decomposition (HASTD). Owing to implementation advantages, our solution forms an interesting basis for developing a new class of image codecs.

Keywords: Image · Decomposition · Prediction · Polyphase · Adaptive · Transform · Lifting

1 Introduction

From several works [10][15], it is known that video codecs can be used to compress still images to good effect. This has been verified mainly by applying a video codec to the single-frame sequence that consists only of a given image. But in [10], we have shown that a video codec can as well be applied to a sequence that comprises four polyphase components of the image. In the first approach, images are compressed using only the intra-coding mechanisms of the video codec, whereas the latter solution additionally allows for employing motion estimation-compensation and bidirectional inter-prediction.

The second method has given no notable advantages, but it seems possible to significantly improve its performance by adjusting inter-prediction and entropy

V. Murino and E. Puppo (Eds.): ICIAP 2015, Part II, LNCS 9280, pp. 162–171, 2015.
DOI: 10.1007/978-3-319-23234-8_16

coding to processing polyphase components. As this requires very deep changes in a video codec, we decided to develop from scratch a novel image codec that follows the inter-prediction principles of video coding.

In this paper we present the main part of such a codec — a transform that does that with polyphase components what happens with frames in a video encoder. A similar idea was proposed in [19], where the authors noticed analogies between motion compensation and prediction among image blocks. Block prediction was considered also in [6][18][7], but image blocks are much less correlated than blocks taken from polyphase components, which are predicted in our approach. Our solution has also some connection with adaptive lifting-based wavelet transforms [3][4][8][16][11][2][5][1], but is most similar to the prediction-based lower triangular (PLT) transform [12] and generalized triangular decomposition (GTD) [17]. For this reason, the proposed transform could be called the hierarchical adaptive spatial triangular decomposition (HASTD).

The general computational scheme of the HASTD is almost the same as for the PLT and GTD, but there are fundamental conceptual differences. Firstly, both those transforms have been developed by factorizing autocorrelation matrices and by looking for approximations of the Karhunen–Loeve transform (KLT). No analogies to video coding have been noticed. Secondly, the PLT and GTD are aimed at processing 1-D signals, and their theory cannot easily be extended to images, or to the spatial domain. Thirdly, neither adaptive nor iterative variants of the transforms have been considered. Finally, multiplications are necessary to compute both PLT and GTD.

2 Proposed Principle of Image Analysis-Synthesis

The HASTD algorithm for image analysis and synthesis can be explained using Fig. 1. In the scheme, if some symbol $\mathbf{S}$ denotes a signal, then $\underline{\underline{\mathbf{S}}}$ stands for a prediction-based estimate of this signal, and $\boldsymbol{\Delta}\mathbf{S} = \mathbf{S} - \underline{\underline{\mathbf{S}}}$ is the prediction residual. The quantized version of the residual is denoted as $\underline{\boldsymbol{\Delta}\mathbf{S}}$, and the related approximation of the signal is denoted as $\underline{\mathbf{S}} = \underline{\underline{\mathbf{S}}} + \underline{\boldsymbol{\Delta}\mathbf{S}}$.

In our approach, an image $\mathbf{X}$ is firstly split into four polyphase components, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$, in accordance with Fig. 2. Then $\underline{\underline{\mathbf{A}}}$, a quantized version of $\mathbf{A}$, is used to predict $\mathbf{B}$. The prediction residual $\boldsymbol{\Delta}\mathbf{B} = \mathbf{B} - \underline{\underline{\mathbf{B}}}$ is quantized, and its approximate $\underline{\boldsymbol{\Delta}\mathbf{B}}$ and the estimate of $\underline{\underline{\mathbf{B}}}$ from $\underline{\mathbf{A}}$ are added so as to reconstruct $\underline{\mathbf{B}}$, an approximate of $\mathbf{B}$. The $\mathbf{C}$ component is processed in the same way, but being predicted using $\underline{\mathbf{A}}$, and $\underline{\mathbf{B}}$. Finally, $\mathbf{D}$ is predicted using $\underline{\mathbf{A}}$, $\underline{\mathbf{B}}$, and $\underline{\mathbf{C}}$. We do not apply any additional transform to the residuals, like in the MPEG standards for video coding [13][14], or in [9].

The analysis results comprise a quantized version of the reference component, the quantized residuals of prediction of the remaining polyphase components, and the prediction-control data. The data point out samples of approximate components used to obtain residual blocks, as explained in Section 3.

The synthesis algorithm is shown in the right-hand side of Fig. 1. The control data allow for reversing the prediction done during analysis. Given $\underline{\mathbf{A}}$ and $\underline{\boldsymbol{\Delta}\mathbf{B}}$,

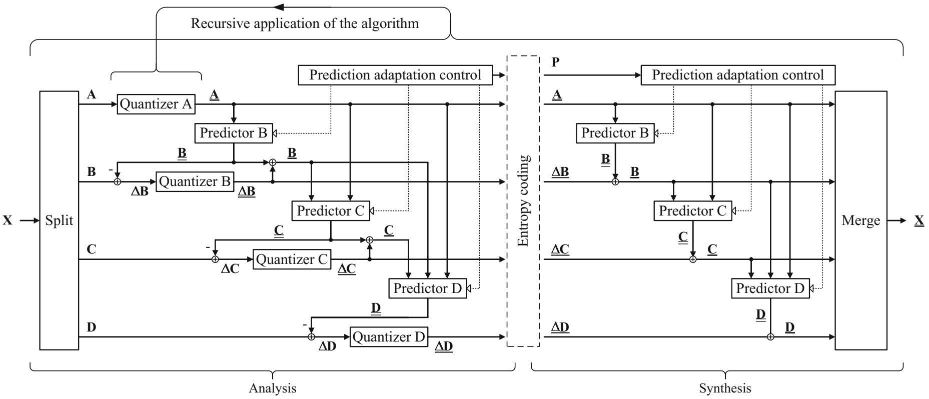


Fig. 1. Data flow graphs of the HASTD.

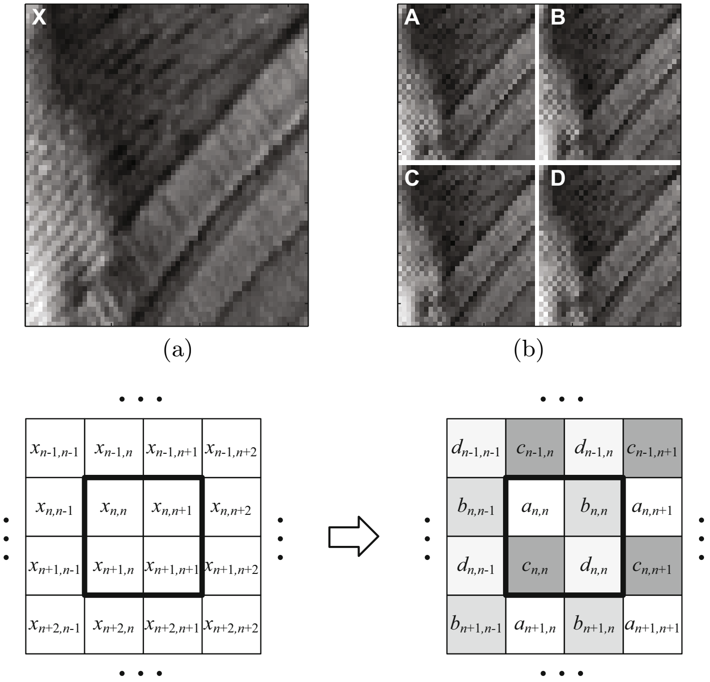
one can restore **B**. Then **C** can be reconstructed from **A**, **B**, and **ΔC**, and **D** — from **A**, **B**, **C**, and **ΔD**.

In Fig. 1, the "Split" and "Merge" blocks represent the conversions of an image into its polyphase components and vice versa, respectively. Obviously, they symbolize only conceptual reorganization of image samples, not real memory data movements.

Each quantizer in Fig. 1 represents an arbitrary algorithm for approximating samples at reduced bit rate. It is expected to produce a representation of approximated values that is more suitable to entropy coding. In particular, one can realize a hierarchical transform by using the scheme in Fig. 1 to decompose the **A** component in the same way. Thus, decompositions obtainable using our approach can be visualized as in Fig. 3, similarly to wavelet transforms.

Our solution mimics the motion estimation-compensation used in video coding, where previously encoded frames serve as references for predicting a subsequent frame. The scheme is also similar to lifting schemes that are commonly used to implement wavelet transforms, but the differences between the solutions are considerable. Firstly, prediction lifting steps are done after quantization of signals used as the prediction basis. Secondly, there is no update steps, in our approach.

It can by surprising for some people that we omit the update step in the lifting scheme and that we process polyphase components like smooth signals, even though they contain aliasing. However, from a pragmatic point of view, if a signal contains aliasing terms, then this only means that this signal is "difficult" to encode using methods that assume its smoothness, as severe artifacts are more probable in its decoded versions , and it is risky to reconstruct the original signal using its alias-contaminated quantized components. Undoubtedly, an untouched polyphase component carries useful information about an image — the values of every fourth pixel. In our transform, quantization of a polyphase component directly introduces inaccuracies to the reconstructed image, and the


X
A
B
C
D
$x_{n-1,n-1}$ $x_{n-1,n}$ $x_{n-1,n+1}$ $x_{n-1,n+2}$
$x_{n,n-1}$ $x_{n,n}$ $x_{n,n+1}$ $x_{n,n+2}$
$x_{n+1,n-1}$ $x_{n+1,n}$ $x_{n+1,n+1}$ $x_{n+1,n+2}$
$x_{n+2,n-1}$ $x_{n+2,n}$ $x_{n+2,n+1}$ $x_{n+2,n+2}$
$d_{n-1,n-1}$ $c_{n-1,n}$ $d_{n-1,n}$ $c_{n-1,n+1}$
$b_{n,n-1}$ $a_{n,n}$ $b_{n,n}$ $a_{n,n+1}$
$d_{n,n-1}$ $c_{n,n}$ $d_{n,n}$ $c_{n,n+1}$
$b_{n+1,n-1}$ $a_{n+1,n}$ $b_{n+1,n}$ $a_{n+1,n+1}$

(a) (b)

(c)

Fig. 2. Polyphase decomposition of an image: (a) a fragment of the “Lena” image, (b) its polyphase components, and (c) the relations among pixels of the image and its components.

error in a polyphase component limits reconstruction quality. But this error does not automatically spread on other polyphase components, or adjacent pixels, as in the conventional transform coders. It only increases prediction residuals among this and other components. If a residual is not quantized, then the related polyphase component can be exactly recovered regardless of what happened with the remaining components.

This establishes two important key properties of our algorithm. Firstly, the effect of coefficient quantization on the reconstruction accuracy can be determined more easily than in the known transform coders, as errors in reconstructed images are directly determined by errors in polyphase components. Secondly, each residual might be quantized in a different way, for the same reconstruction accuracy. If a component can be predicted using more other components, and/or the reference components are quantized less, then the resulting residual is smaller and can be quantized more coarsely.

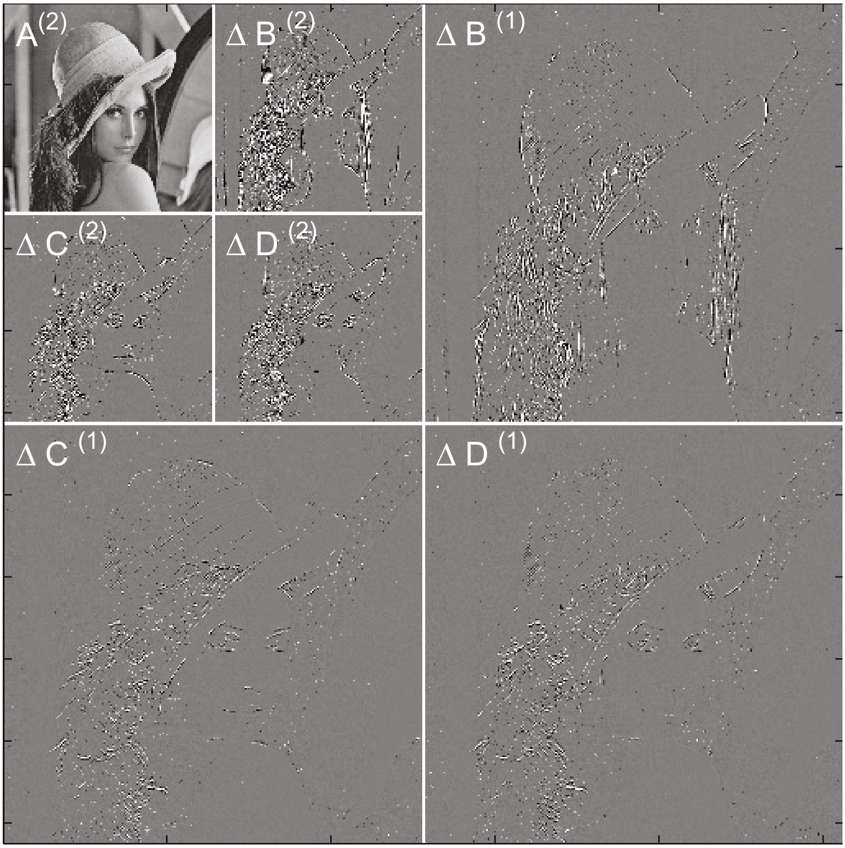


Fig. 3. Non-quantized two-level HASTD of the "Lena" image. The residuals have been amplified, so as to better show distributions of their non-zero values.

3 Adaptive Prediction Mechanisms

Predicting a polyphase component using other ones is equivalent to predicting a pixel using its neighbours, which is the well known effective means of image compression. In our algorithm, the prediction is adapted, because this allows to reduce the energies of the residuals about ten times, compared to simply taking the average of all accessible adjacent pixels. The unit of adaptation is the square block of pixels of a polyphase component. Our experiments showed that the block size is not very important and can be large, as there is no big difference between residuals obtained using 4×4 and 16×16 blocks.

We have limited the adaptation to switching reference samples that are arguments for a fixed prediction function. The function is the average of adjacent pixels, so can be computed without multiplications, using only one addition, one bit shift, and one subtraction. In the preliminary experiments, we considered the following models of a pixel block: only one prediction mode

$$b_{m,n} = \tfrac{1}{2}\left(a_{m,n} + a_{m,n+1}\right) + \Delta b_{m,n} \tag{1}$$

for the B component; three prediction modes:

$$c_{m,n} = \tfrac{1}{2}\left(a_{m,n} + a_{m+1,n}\right) + \Delta c_{m,n} \tag{2}$$

$$c_{m,n} = \tfrac{1}{2}\left(b_{m,n-1} + b_{m+1,n}\right) + \Delta d_{m,n} \tag{3}$$

$$c_{m,n} = \tfrac{1}{2}\left(b_{m,n} + b_{m+1,n-1}\right) + \Delta c_{m,n} \tag{4}$$

for the C component; and four modes

$$d_{m,n} = \tfrac{1}{2}\left(b_{m,n} + b_{m+1,n}\right) + \Delta d_{m,n} \tag{5}$$

$$d_{m,n} = \tfrac{1}{2}\left(c_{m,n} + c_{m,n+1}\right) + \Delta d_{m,n} \tag{6}$$

$$d_{m,n} = \tfrac{1}{2}\left(a_{m,n} + a_{m+1,n+1}\right) + \Delta d_{m,n} \tag{7}$$

$$d_{m,n} = \tfrac{1}{2}\left(a_{m,n+1} + a_{m+1,n}\right) + \Delta d_{m,n} \tag{8}$$

for the D component, where $m = iM + (0, \ldots, M-1)$, $n = jM + (0, \ldots, M-1)$, i and j determine a pixel block, and M denotes the block size. Thus, the algorithm predicts the pixels of a block using their horizontal, vertical, or diagonal neighbours of other polyphase components, so as to follow the direction of local changes of image intensity. This can be conceptually connected with the bidirectional prediction of video coding.

The adaptation consists in searching for the prediction variant that minimizes the Sum of Absolute Differences (SAD), the criterion that is often used in motion estimation. In the preliminary experiments, the optimum prediction settings for a block were determined by brute-force testing all possibilities. Therefore, SAD evaluations represent the main computational load related to our transform, but the quantity can be computed without multiplications, and it is possible to skip further evaluations for a given block after reaching the SAD below some threshold.

It is noteworthy that the indexes of prediction variants, which must accompany HASTD residuals, can be effectively encoded. Firstly, for large blocks, a single index codeword is shared by many pixels. In particular, the related bitrate is only $4/256 = 0.016$ bpp, if 4 bits, necessary to straightforwardly represent the index, are assigned to each 16 × 16 block. Secondly, the indexes of adjacent blocks are usually correlated, so they can be compressed using known methods for coding motion vectors in video compression.

It should also be mentioned that similar prediction equations can be found in many papers on adaptive wavelet transforms, see eg. [4][8], but in contexts different from our idea. In those works, contrary to our approach, prediction is usually done among rows or columns, so as to preserve the transform separability, a single pixel is the adaptation unit, and the optimum prediction model is determined by analyzing adjacent reference pixels, so that no side data must accompany transform coefficients. But, the main difference between our approach and adaptive wavelet transforms is that, in the latter, prediction, or lifting, is based on non-quantized signals, and thus the encoding order and reconstruction quality of these signals need not to be taken into account.

Table 1. Statistics of the HASTD coefficients of the “Lena” image

Category	Min	Max	Mean	Var	Entropy
$A^{(1)}$	-101,0	113,0	-4,0	2260,1	7,4
$\Delta B^{(1)}$	-56,0	73,0	0,0	40,8	4,4
$\Delta C^{(1)}$	-48,0	63,0	0,0	21,1	4,1
$\Delta D^{(1)}$	-48,0	68,0	0,0	20,7	4,0
$A^{(2)}$	-101,0	106,0	-4,0	2291,4	7,4
$\Delta B^{(2)}$	-99,0	152,0	0,0	143,5	5,0
$\Delta C^{(2)}$	-81,0	106,0	0,0	78,5	4,6
$\Delta D^{(2)}$	-76,0	100,0	0,0	78,4	4,6

Table 2. Statistics of the wavelet coefficients of the “Lena” image

Category	Min	Max	Mean	Var	Entropy
$A^{(1)}$	-207.6	228.0	-8.2	8907.8	8.4
$H^{(1)}$	-49.3	47.7	0.0	17.4	3.9
$V^{(1)}$	-74.2	63.6	0.0	42.4	4.3
$D^{(1)}$	-33.7	27.9	0.0	8.7	3.5
$A^{(2)}$	-395.8	402.8	-17.2	33651.2	9.4
$H^{(2)}$	-204.4	141.0	0.0	166.7	4.9
$V^{(2)}$	-299.6	238.1	-0.2	430.3	5.6
$D^{(2)}$	-106.4	117.6	0.1	113.1	4.8

4 Experimental Evaluation of the HASTD

The key properties of the HASTD can be demonstrated empirically. We use the “Lena” image through this paper, but we obtained similar results also for other standard test images. The presented plots and tables refer to the 2-level decomposition with the prediction adaptation block size of 16 × 16.

Figure 3 illustrates the results of the HASTD, in the way commonly used to wavelet transforms. The residuals generally look like ordinary wavelet subbands. The image energy is packed into the reference polyphase component at the second level, an the variance of this component is orders of magnitude greater than those of the residuals. The second-level residuals are larger than the first-level ones, but both have significant values located at the same places, related to image edges. Thus, the HASTD can be used to compress images using the principle of zero-tree coding. The main difference compared to the wavelet transform is that that the **C** and **D** residuals are very similar and cannot be identified with horizontal and diagonal details.

More information about the HASTD results is revealed in Table 1 and Fig. 4. For comparison purposes, the analogous data for the two-level wavelet transform with the “9-7” filters has been shown in Table 2 and Fig. 5, where $\mathbf{A}^{(n)}$ stands for the approximation subband of the n-th level, whereas **H**, **V**, and **D** denote horizontal, vertical, and diagonal details, respectively.

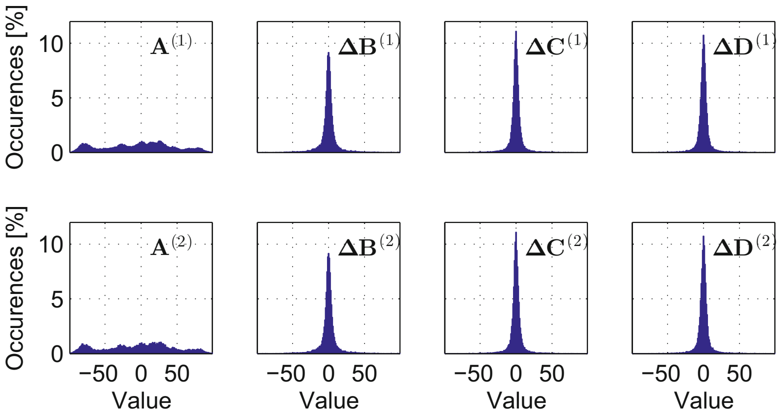


Fig. 4. Coefficient histograms of the 2-level HASTD of the "Lena" image.

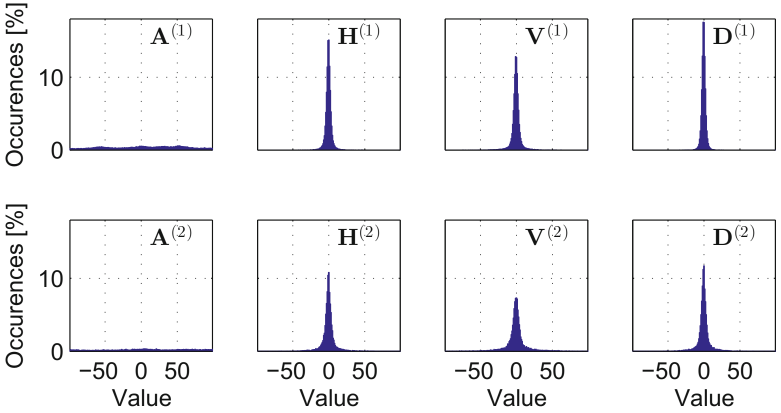
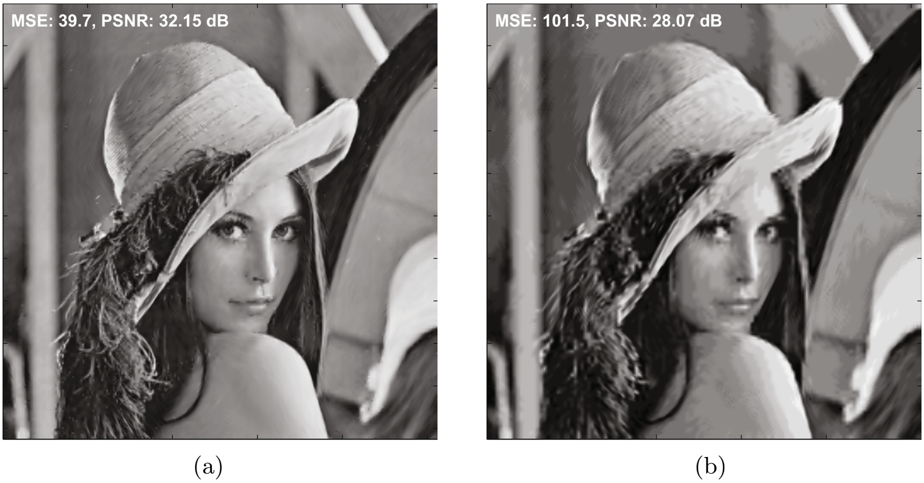


Fig. 5. Coefficient histograms of the 2-level wavelet transform obtained for the "Lena" image using the "9–7" filters.

From a compression point of view, there is not much difference between the histograms of the HASTD residuals and the histograms of wavelet coefficients. Unlike for wavelet coefficients, the dynamic range and variance of HASTD results increase only slightly from level to level. The HASTD seems to posses the property of the unit gain, which characterizes the known triangular transforms [12]. Thus, it seems possible to store HASTD results in the place of image samples, so as to save memory.

It is also of interest to evaluate the achievable accuracy of reconstruction of an image from its quantized HASTD representations. The image in Fig. 6a has been restored using only the second-level data: $\mathbf{A}(2)$ was quantized to 7 bits, whereas $\mathbf{\Delta B}(2)$, $\mathbf{\Delta C}(2)$, and $\mathbf{\Delta D}(2)$ were quantized to 3 bits (the total bitrate of 1 bpp). In Fig. 6b, the image is presented that has been restored using only $\mathbf{A}(2)$ samples quantized to 4 bits (the total bitrate of 0.25 bpp). It is notable that



(a) (b)

Fig. 6. Reconstructions of the “Lena” image from its quantized HASTDs.

the compression artifacts are different from those that characterize the known, DCT- as well as wavelet-based codecs: they are a combination of blocking and blurring.

The results are not impressive compared to the state-of-art algorithms but prove that the HASTD can be used in image coding. Obviously, better compression ratios can be obtained using fine-tuned quantization and entropy coding algorithms, which are subjects for future works.

5 Conclusion

The HASTD is conceptually simple, even though it is difficult to describe it using mathematical notation. Nevertheless, it decorrelates image samples nearly as well as the known transforms that are based on well-developed theories or have been designed using advanced numerical optimization. Our algorithm can be implemented without multiplications, using only additions and bit shifts, and can be computed without wasting memory for intermediate results. Therefore, our transform can be very economically implemented in hardware and is able to substitute the conventional transforms in many applications.

Acknowledgments. This work was supported by the Polish National Science Centre under Decision No. DEC-2012/07/D/ST6/02454.

References

1. Blackburn, J., Do, M.N.: Two-dimensional geometric lifting. In: Proc. 16th IEEE Int. Conf. Imag. Process. (ICIP), Cairo, Egypt, pp. 3817–3820 (November 7–10, 2009)

2. Fang, Z., Xiong, N., Yang, L., Sun, X., Yang, Y.: Interpolation-based direction-adaptive lifting DWT and modified SPIHT for image compression in multimedia communications. IEEE Systems J. **5**(4), 584–593 (2011)
3. Gerek, O.N., Cetin, A.E.: Adaptive polyphase subband decomposition structures for image compression. IEEE Trans. Image Process. **9**(10), 1649–1659 (2000)
4. Gerek, O., Cetin, A.: A 2-D orientation-adaptive prediction filter in lifting structures for image coding. IEEE Trans. Image Process. **15**(1), 106–111 (2006)
5. Hattay, J., Benazza-Benyahia, A., Pesquet, J.: Adaptive lifting for multicomponent image coding through quadtree partitioning. In: Proc. 30th IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP), Philadelphia, PA, vol. 2, pp. 213–216. (March 19–23, 2005)
6. Huang, J., Liu, S.: Block predictive transform coding of still images. In: Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP). vol. 5, Adelaide, Australia, pp. V-333–V-336 (April 19–22, 1994)
7. Jiao, L., Wang, L., Wu, J., Bai, J., Wang, S., Hou, B.: Shape-adaptive reversible integer lapped transform for lossy-to-lossless ROI coding of remote sensing two-dimensional images. IEEE Geosci. Remote Sens. Lett. **8**(2), 326–330 (2011)
8. Kaaniche, M., Pesquet-Popescu, B., Benazza-Benyahia, A., Pesquet, J.C.: Adaptive lifting scheme with sparse criteria for image coding. EURASIP J. Advances Sig. Process. **2012**(1), 10 (2012)
9. Kamisli, F., Lim, J.: 1-D transforms for the motion compensation residual. IEEE Trans. Image Process. **20**(4), 1036–1046 (2011)
10. Parfieniuk, M.: Polyphase components of an image as video frames: a way to code still images using H.264. In: Proc. Picture Coding Symp. (PCS), Cracow, Poland, pp. 189–192 (May 7–9, 2012)
11. Peng, X., Xu, J., Wu, F.: Directional filtering transform for image/intra-frame compression. IEEE Trans. Image Process. **19**(11), 2935–2946 (2010)
12. Phoong, S.M., Lin, Y.P.: Prediction-based lower triangular transform. IEEE Trans. Signal Process. **48**(7), 1947–1955 (2000)
13. Rao, K., Kim, D., Hwang, J.: Video Coding Standards: AVS China, H.264/MPEG-4 PART 10, HEVC, VP6, DIRAC and VC-1. Springer (2014)
14. Richardson, I.: The H.264 Advanced Video Compression Standard, 2 edn. Wiley (2010)
15. Tran, T., Liu, L., Topiwala, P.: Performance comparison of leading image codecs: H.264/AVC Intra, JPEG2000, and Microsoft HD Photo. In: Proc. SPIE 6696 (Applications of Digital Image Processing XXX), 66960B (2007)
16. Vrankic, M., Sersic, D., Sucic, V.: Adaptive 2-D wavelet transform based on the lifting scheme with preserved vanishing moments. IEEE Trans. Image Process. **19**(8), 1987–2004 (2010)
17. Weng, C.C., Chen, C.Y., Vaidyanathan, P.: Generalized triangular decomposition in transform coding. IEEE Trans. Signal Process. **58**(2), 566–574 (2010)
18. Xu, J., Wu, F., Zhang, W.: Intra-predictive transforms for block-based image coding. IEEE Trans. Signal Process. **57**(8), 3030–3040 (2009)
19. Zhao, H., He, Z.: Lossless image compression using super-spatial structure prediction. IEEE Signal Process. Lett. **17**(4), 383–386 (2010)